

**PLATAFORMA WEB PARA EL SEGUIMIENTO, MONITOREO Y GESTIÓN DE
LA INFORMACIÓN EN TIEMPO REAL DIRIGIDA AL MODELO DE
TRANSPORTE URBANO BASADO EN BRT**

**LUIS ANTONIO CORTES RUEDA
RAFAEL EDUARDO SÁNCHEZ CUADROS**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA**

2017

**PLATAFORMA WEB PARA EL SEGUIMIENTO, MONITOREO Y GESTIÓN DE
LA INFORMACIÓN EN TIEMPO REAL DIRIGIDA AL MODELO DE
TRANSPORTE URBANO BASADO EN BRT**

**LUIS ANTONIO CORTES RUEDA
RAFAEL EDUARDO SÁNCHEZ CUADROS**

Trabajo de Grado para optar al título de Ingeniero de Sistemas e Informática

Director

**GABRIEL RODRIGO PEDRAZA FERREIRA
PhD, Ciencias de la computación**

Codirector

**LUIS EDUARDO BAUTISTA ROJAS
Mg, Ingeniería de sistemas e informática**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA**

2017

DEDICATORIA

Gracias a Dios, a mi familia en especial a mis padres por su amor y su apoyo incondicional, enseñándome que los sueños si se pueden cumplir con trabajo y perseverancia.

A mis queridos amigos, compañeros y profesores con quienes empecé esta etapa importante en mi vida y a quienes les agradezco por estos tiempos compartidos. Por las experiencias que me sirvieron para ser cada día mejor persona.

Luis Antonio Cortes Rueda

DEDICATORIA

A Dios ante todo, a mis padres Rafael Sánchez Prieto y Elvinia Cuadros Alarcón, que han sido un apoyo incondicional e invaluable en cada proyecto de mi vida. Por todo su amor, confianza y dedicación que han contribuido a formar la persona que soy hoy en día.

A mis profesores, por todas las experiencias y conocimientos que compartieron conmigo durante todas las clases. Mi gran respeto y admiración por tan importante labor.

A mis compañeros, con quienes compartí gran parte de este camino de aprendizaje, creación e investigación.

Rafael Eduardo Sánchez Cuadros

AGRADECIMIENTOS

Antes que todo agradecerle a Dios, por darnos salud e inteligencia y permitirnos vivir tan excelente experiencia.

Gracias a nuestro director de proyecto Gabriel Rodrigo Pedraza Ferreira por creer en nosotros y apoyarnos en la implementación de su tan maravillosa idea.

Gracias a nuestro codirector Luis Eduardo Bautista Rojas por guiarnos de la mejor manera en todo el desarrollo.

A nuestros familiares, amigos y compañeros que nos apoyaron en este proceso para culminar con satisfacción nuestro ciclo estudiantil.

CONTENIDO

	Pág.
INTRODUCCIÓN	15
1. PLANTEAMIENTO Y JUSTIFICACIÓN DEL PROBLEMA	16
2. OBJETIVOS	19
2.1 OBJETIVO GENERAL	19
2.2 OBJETIVOS ESPECÍFICOS	19
3. MARCO TEÓRICO	20
3.1 AUTOBÚS DE TRÁNSITO RÁPIDO (BRT)	20
3.2 SISTEMAS INTELIGENTES DE TRANSPORTE (ITS) Y SISTEMAS DE INFORMACIÓN GEOGRÁFICAS (SIG)	21
3.3 SISTEMAS HOMBRE-MÁQUINA – ERGONOMÍA	22
3.4 TECNOLOGÍAS WEB	23
3.4.1 Html (HipertText Markup Language).	24
3.4.2 JavaScript.	24
3.4.3 JSON (JavaScript Object Notation).	25
3.4.4 Web Services	25
3.4.5 WebSockets	26
4. METODOLOGÍA	27
4.1 PROTOTIPOS EVOLUTIVOS: ENFOQUE GENERAL	27
4.2 METODOLOGÍA UTILIZADA Y PLAN DE TRABAJO	28
4.2.1 Primera Fase: Ambientación a la tecnología	29
4.2.2 Segunda Fase: Identificación de las necesidades	29

4.2.3 Tercera Fase: Desarrollo	30
4.2.4 Cuarta Fase: Simulación y validación	30
4.2.5 Quinta Fase: Prototipo final	30
5. IMPLEMENTACIÓN	31
5.1 HERRAMIENTAS DE DESARROLLO	31
5.2 ANÁLISIS DE REQUERIMIENTOS	37
5.3 ARQUITECTURA GENERAL	40
5.4 ARQUITECTURA SIGMAT	41
5.5 PROTOTIPO INICIAL	43
5.5.1 Generalidades del prototipo inicial.	43
5.5.2 Diseño Interfaz del prototipo inicial - Primera iteración	44
5.5.3 Diseño Interfaz del prototipo inicial - Segunda iteración	47
5.6 PRUEBA DE USABILIDAD EN SIGMAT POR PARTE DE LOS USUARIOS	52
5.7 PROTOTIPO FINAL	60
6. PRUEBAS DE VALIDACIÓN	68
6.1 PRUEBA DEL SISTEMA DE MONITOREO WEB POR MEDIO DE UNA SIMULACIÓN DE UN SISTEMA BRT DE PEQUEÑA ESCALA.	68
6.2 PRUEBAS DE COMPATIBILIDAD CON DIFERENTES NAVEGADORES WEB Y OS.	71
6.3 EVALUACIÓN HEURÍSTICA	74
7. CONCLUSIONES	84
REFERENCIAS BIBLIOGRAFICAS	86
BIBLIOGRAFÍA	89

LISTA DE FIGURAS

	Pág.
Figura 1. Actividades del modelo evolutivo	28
Figura 2. Fases de actividades dentro de la metodología adecuada a utilizar	29
Figura 3. Arquitectura general del macro-proyecto - SIGMAT	41
Figura 4. Arquitectura MVC-SIGMAT	42
Figura 5. Prototipo inicial - primer prototipo inicial - primera iteración - mapa	45
Figura 6. Prototipo inicial - primer prototipo inicial - primera iteración - grafica	46
Figura 7. Prototipo inicial - primer prototipo inicial - primera iteración - tabla	47
Figura 8. Prototipo inicial - segunda iteración - interfaz pantalla principal – Modal	49
Figura 9. Prototipo inicial - segunda iteración - interfaz pantalla principal	50
Figura 10. Prototipo inicial -segunda iteración -interfaz línea de tiempo –única ruta	51
Figura 11. Prototipo inicial-segunda iteración-interfaz línea de tiempo –múltiple ruta	51
Figura 12. Prototipo inicial - segunda iteración - interfaz tablas	52
Figura 13. Prototipo final - interfaz principal	62
Figura 14. Prototipo final - interfaz de navegacion - L.Tiempo	63
Figura 15. Prototipo final - interfaz de navegacion - tablas	63
Figura 16. Prototipo final - interfaz multiples rutas - L.Tiempo	64
Figura 17. Prototipo final - interfaz unica ruta - L.Tiempo	65
Figura 18. Prototipo final - interfaz - tablas	67
Figura 19. Prototipo final - interfaz - tablas – pdf	67
Figura 20. Estructuración de las peticiones y el archivo json	69
Figura 21. Prueba del sistema de monitoreo web – web server local	71

LISTA DE TABLAS

	Pág.
Tabla 1. Prueba de usabilidad - datos	56
Tabla 2. Prueba de usabilidad – efectividad	56
Tabla 3. Prueba de usabilidad - eficiencia	57
Tabla 4. Prueba de usabilidad - satisfacción	57
Tabla 5. Pruebas de compatibilidad con diferentes navegadores	73
Tabla 6. Prueba de usabilidad - datos de análisis	75
Tabla 7. Prueba de usabilidad - objetivos	75
Tabla 8. Prueba de usabilidad - mediciones	76
Tabla 9. Prueba de usabilidad - generales	76
Tabla 10. Prueba de usabilidad – identidad e información	77
Tabla 11. Prueba de usabilidad – lenguaje y redacción	77
Tabla 12. Prueba de usabilidad - rotulado	77
Tabla 13. Prueba de usabilidad – estructura y navegación	78
Tabla 14. Prueba de usabilidad – layout de la página	79
Tabla 15. Prueba de usabilidad - búsqueda	79
Tabla 16. Prueba de usabilidad – elementos multimedia	79
Tabla 17. Prueba de usabilidad - accesibilidad	79
Tabla 18. Prueba de usabilidad – control y retroalimentación	80
Tabla 19. Prueba de usabilidad – conclusiones	80

RESUMEN

TÍTULO: PLATAFORMA WEB PARA EL SEGUIMIENTO, MONITOREO Y GESTIÓN DE LA INFORMACIÓN EN TIEMPO REAL DIRIGIDA AL MODELO DE TRANSPORTE URBANO BASADO EN BRT*.

AUTORES: RAFAEL EDUARDO SÁNCHEZ CUADROS
LUIS ANTONIO CORTES RUEDA**

PALABRAS CLAVE: Arquitectura, escalabilidad, metodología, tecnologías web.

DESCRIPCIÓN:

SIGMAT (**Sistema de Información para la gestión y monitoreo automático del transporte**) es un prototipo funcional de una plataforma web que busca transformar y cambiar la forma en la que se realiza el monitoreo automático del transporte público actualmente. Además este sistema de información puede ser implementado en las diferentes ciudades donde se tengan sistemas de transporte BRT (**Autobús de tránsito rápido**). SIGMAT fue desarrollada para un tipo de usuario en especial:

EL OPERADOR DEL CENTRO DE CONTROL

En este proyecto pero totalmente adaptable para otros usuarios.

SIGMAT cuenta con cuatro (4) tipos de monitoreo diferentes:

- 1- **Monitoreo en línea de tiempo - única ruta**
- 2- **Monitoreo en línea de tiempo - múltiples rutas**
- 3- **Monitoreo en tablas - única ruta**
- 4- **monitoreo en mapa - una y múltiples rutas.**

Este proyecto pretende atacar diversas problemáticas del sistema de información que manejan actualmente los operadores que afectan el sistema de transporte público. SIGMAT busca brindarle usuario-operador un SIG con el cual puedan realizar seguimiento y monitoreo en tiempo real del recorrido de todos los buses correspondientes a una o múltiples rutas, además de mejorar la disponibilidad, calidad y la forma en que se muestra la información correspondiendo a su recorrido total y entre estaciones, otorgando al operador tomar decisiones con mayor rapidez, realizar análisis al sistema tendientes a la mejora continua de la calidad del servicio, generar un beneficio social que se representa en la disminución del tiempo de viaje eliminando recorridos innecesarios e incentivando y promoviendo la inclusión de nuevos indicadores de movilidad.

* Trabajo de grado

** Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingeniería de Sistemas e Informática.
Director: PhD Gabriel Rodrigo Pedraza Ferreira, Codirector: Mg Luis Eduardo Bautista Rojas

ABSTRACT

TITLE: WEB PLATFORM FOR MONITORING AND MANAGING REAL-TIME INFORMATION DIRECTED TO THE URBAN TRANSPORT MODEL BASED ON BRT^{*}

AUTHORS: RAFAEL EDUARDO SÁNCHEZ CUADROS;
LUIS ANTONIO CORTES RUEDA^{**}

KEYWORDS: Architecture, scalability, methodology, web technologies.

DESCRIPTION:

SIGMAT (Information System for the management and automatic monitoring of transport) is a functional prototype of a web platform that seeks to transform and change the way in which the automatic monitoring of public transport is performed today. In addition, this information system can be implemented in the different cities where there are BRT (Rapid Transit Bus) transport systems. SIGMAT was developed for a particular user type:

- **THE OPERATOR OF THE CONTROL CENTER**
In this project But totally adaptable for other users.

SIGMAT has four (4) different types of monitoring:

- 1- Timeline monitoring - single route**
- 2- Timeline monitoring - multiple routes**
- 3- Monitoring in tables - single route**
- 4- Map monitoring - one and multiple routes.**

This project aims to tackle various problems of the information system currently operated by operators that affect the public transport system. SIGMAT seeks to offer a user-operator a GIS with which they can monitor and monitor in real time the route of all buses corresponding to one or multiple routes, in addition to improving the availability, quality and the way information is displayed corresponding to its total travel and between stations, giving the operator quicker decisions, performing system analysis aimed at continuous improvement of service quality, generating a social benefit that is represented by decreasing travel time eliminating unnecessary travel and encouraging And promoting the inclusion of new mobility indicators.

^{*} Bachelor thesis

^{**} Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingeniería de Sistemas e Informática.
Director: PhD Gabriel Rodrigo Pedraza Ferreira, Codirector: Mg Luis Eduardo Bautista Rojas.

INTRODUCCIÓN

La dudosa logística y la ineficiente toma de decisiones por parte de los operadores del sistema generan incertidumbre a la hora de esperar una ruta de transporte, junto con enfrentamientos que se presentan entre la planeación urbana y del transporte (ocasionando caos), hacen del transporte público en Colombia un problema de gran magnitud que merece pronta solución.

En consecuencia, estos primitivos sistemas no permiten mitigar fenómenos como el sobrecupo en algunos buses mientras que otros se encuentran vacíos, esto trae como resultado la alteración de las frecuencias de estos mismos e incomodidad a los usuarios, esto debido a que la demanda de usuarios en el transporte público no se comporta de forma constante sino de forma variable en el tiempo, lo que imposibilita modelar el control con los pocos datos obtenidos además cuando estos son recogidos muchas veces ya no son pertinentes, esto hace necesario un sistema que permita obtener mayor cantidad de información que posibilite modelar eficazmente la demanda del transporte público, con el fin de ofrecer una frecuencia de buses acorde a la demanda del servicio.

El propósito del desarrollo objeto de este proyecto de grado es diseñar, implementar e implantar un prototipo de plataforma web funcional, capaz de mostrar una visualización de la información eficiente para facilitarle al usuario-operador tomar decisiones que tengan impacto en el sistema actual del transporte público basado en BRT, para ello el sistema consta de diversos prototipos funcionales que fueron probados y evaluados por estudiantes y profesionales de la escuela de Ingeniería de Sistemas e Informática como usuarios y empleados del sistema de transporte público, según sus puntos de vista y opiniones.

1. PLANTEAMIENTO Y JUSTIFICACIÓN DEL PROBLEMA

Previamente a la implantación del sistema BRT los problemas más evidentes que se presentaban en el transporte público eran:

- La inconsistencia en la frecuencia de buses, que causaba en los automotores sobrecupo en horas pico y pocos pasajeros en horas valle.
- La carente planeación de rutas del sistema que obligaba en muchos casos a realizar transbordos o usar otros medios alternativos de transporte.
- El rápido crecimiento que estaban teniendo las más importantes urbes de Colombia debido a factores como el crecimiento poblacional urbano y al desplazamiento rural.

Este crecimiento demográfico ha hecho del transporte urbano de pasajeros en las principales urbes de Colombia un problema de gran magnitud que necesita una pronta solución ante un previsible colapso del servicio de transporte público.

En vista de los argumentos anteriores se consideró urgente para los mandatarios locales tomar medidas, como implantar el sistema de transporte BRT, un sistema que sobre el papel resolvería el principal problema el cual era que se pronosticaba que el transporte público convencional se quedaría pequeño en el corto-mediano plazo ante la creciente demanda de pasajeros en las ciudades con un alto crecimiento poblacional, por lo que en algunas de las principales ciudades de Colombia se le apostó al sistema BRT que en teoría, gracias al gran tamaño de los buses y al uso de un carril exclusivo en gran parte del trayecto, podrían mover más pasajeros y hacerlo de forma más eficiente.

Se esperaba que la implementación de este “moderno sistema” aumentará la cobertura del servicio y mejora la calidad del mismo, pero la realidad no puede estar más a dispar, no solo arrastró los problemas del sistema anterior sino que

puso en evidencia otros, como: la mala planeación de su implementación, los colados, poca información sobre puntos de recarga, pocos puntos de recarga de las tarjetas, el crecimiento del parque automotor, el fortalecimiento del transporte informal (pirata), consecuencia del pésimo servicio y la falta de alternativas.

La falta de información no permite tomar medidas tendientes a mitigar fenómenos como el sobrecupo en algunos buses, mientras que en otros se encuentran vacíos o hacer más coherente la frecuencia de los buses o realizar una adecuada planeación de los horarios. Existen varias variables que hacen complejo el problema como lo son: la demanda variable de pasajeros, el tráfico en las vías, el uso de las estaciones y la falta de información acerca del comportamiento del sistema.

Esto hace necesario una herramienta de monitoreo con capacidades de visualización geográfica que proporcione al operador del sistema la capacidad de gestionar una mayor cantidad de información por medio de representaciones visuales de los datos en tiempo real, facilitando su análisis durante y después de su visualización, esto le permitirá al operador gestionar adecuadamente el servicio y analizar el comportamiento del sistema, con el fin de ofrecer una frecuencia de buses acorde a la demanda del servicio que se sustente en datos del comportamiento del sistema.

Basados en el estudio realizado por la firma internacional EuroTest y ejecutado por el Departamento de Transporte, Planificación e Ingeniería de Tráfico de la Universidad de Stuttgart, Alemania. A partir de una de sus conclusiones sobre la mejor ciudad Europea en el tema de movilidad podemos rescatar. “La ciudad con mejor puntuación: Munich, en Alemania. Conexiones rápidas, especialmente en el centro de la ciudad, genera bastante información, tanto en las estaciones como dentro de los vehículos, una página Web impresionante y puntos adicionales por su accesibilidad, elevaron a esta ciudad que bordea el río Isar al primer lugar de la

clasificación. A pesar de esto, la estrella de Múnich pierde algo de brillo cuando se llega al tema de los precios” [1].

Podemos notar que parte del atractivo de este sistema de transporte, es la gran cantidad de información con la que cuentan los usuarios del sistema y los encargados de operarlo. Indiscutiblemente se hace imprescindible implementar soluciones TIC, tanto para mejorar la información general del sistema, como para que las personas encargadas de operar el sistema puedan implementar rápidamente planes de acción ante previsibles cambios en la demanda de usuarios.

Este proyecto pretende atacar esta problemática brindando al usuario-operador del sistema de transporte público, un SIG con el cual puedan realizar seguimiento y monitoreo en tiempo real del recorrido de todos los buses correspondientes a una o múltiples rutas, además de mejorar la disponibilidad, calidad y la forma en que se muestra la información correspondiendo a su recorrido total y entre estaciones, otorgando al operador tomar decisiones con mayor rapidez, realizar análisis al sistema tendientes a la mejora continua de la calidad del servicio, generar un beneficio social que se representa en la disminución del tiempo de viaje y aumentar su eficiencia y eficacia, reduciendo costes logísticos, de auditoría y eliminando recorridos innecesarios, dándole un valor estratégico a la empresa, haciendo uso de sistemas embebidos embarcados al medio de transporte.

2. OBJETIVOS

2.1 OBJETIVO GENERAL

Desarrollar un prototipo de plataforma Web que permita monitorizar en tiempo real sistemas de transporte urbano basados en BRT por medio de la georreferenciación y que mejore la calidad y disponibilidad de la información sobre el comportamiento del sistema.

2.2 OBJETIVOS ESPECÍFICOS

- Identificar las necesidades para un sistema de monitoreo involucrado en la prestación del servicio de transporte público, basado en (BRT).
- Plantear una arquitectura para el sistema de monitoreo web que provea propiedades como desacoplamiento, evolución y escalabilidad.
- Implementar un prototipo funcional que utilice tecnologías web para validar la arquitectura ya mencionada.
- Validar el funcionamiento del prototipo de la plataforma web, realizando una simulación del proceso de localización que permita verificar el desplazamiento de algunas rutas de transporte BRT sobre un SIG.

3. MARCO TEÓRICO

En este proyecto se ven inmersos diferentes campos, por lo cual se hace un especial énfasis en los que se consideran son los más importantes, destacando su impacto e inferencia en el sistema de monitoreo.

3.1 AUTOBÚS DE TRÁNSITO RÁPIDO (BRT)

Crear una ciudad inteligente, eficiente y eficaz en temas de movilidad es una obligación. Tanto para el crecimiento de la ciudad como para mejorar la calidad de vida de los ciudadanos. Implementar y optimizar adecuadamente los sistemas de transporte (BRT) se hace necesario para que las ciudades no se convierten en lugares menos sostenibles, con baja calidad de vida y con una visión a futuro poco prometedora.

Por tanto BRT es un sistema de transporte masivo basado en autobuses. En donde las ciudades de América Latina han liderado la implementación de Sistemas de Transporte Público Masivo de Autobuses tipo BRT, un modo de transporte que generalmente se caracteriza por el desarrollo de infraestructura que dan prioridad al transporte público en relación con el transporte en otros tipos de vehículos, ofrece la posibilidad de pagar la tarifa antes de tomar el autobús y permite un rápido acceso al mismo. Más de 45 ciudades de América Latina han realizado inversiones en sistemas tipo BRT, lo que representa el 63,6 por ciento del número de pasajeros en sistemas tipo BRT a nivel mundial [2].

Estas aplicaciones de BRT en América Latina demuestran que la innovación y adaptación es posible, y que el flujo de conocimiento no siempre va en la dirección

Norte (de alto desarrollo industrial) – Sur (en desarrollo). Hasta ahora, el éxito de todas estas aplicaciones fue el resultado de una rara combinación de voluntad política, equipos técnicos orientados a la implantación y recursos; los cuales, aunque fueran menores que alternativas “más tradicionales”, era necesario conseguirlos y asignarlos. Estas iniciativas son susceptibles de permanentes mejoras y expansión, pero son parte importante de la vida y movilidad de las ciudades latinoamericanas y ejemplo de buenas prácticas para el mundo [3].

3.2 SISTEMAS INTELIGENTES DE TRANSPORTE (ITS) Y SISTEMAS DE INFORMACIÓN GEOGRÁFICAS (SIG)

Esa pretensión hace que este proyecto se enfoque en dos tipos de sistemas (**ITS Y SIG**), fundamentales para el crecimiento de la plataforma generando datos óptimos, eficientes y eficaces para obtener el mejor resultado en el monitoreo de los (BRT).

La continua evolución de estos sistemas, que ya no solo obtienen la información sino que la relacionan con otras aplicaciones, analizándola por medio de sistemas expertos, está dando comienzo a una nueva era de Sistemas Inteligentes Interconectados, que sin duda van a suponer un gran salto cualitativo en la seguridad del transporte terrestre[4].

El primer sistema (ITS) se enfoca en obtener la información de los diferentes elementos de interés de las carreteras, que una vez procesada y analizada, se utiliza para mejorar la seguridad de los conductores, mejorando el tráfico y la comodidad en los desplazamientos.

Por ende se define el (ITS) al conjunto de aplicaciones informáticas y sistemas tecnológicos creados con el objetivo de mejorar la seguridad y eficiencia en el

transporte terrestre (carreteras y ferrocarriles), facilitando la labor de control, gestión y seguimiento por parte de los responsables [4].

Por otra parte existe otro tipo de sistema denominado (SIG) el cual nos permite y facilita la visualización de los datos obtenidos en un mapa con el fin de reflejar y relacionar fenómenos geográficos de cualquier tipo, desde mapas de carreteras hasta sistemas de identificación de parcelas agrícolas o de densidad de población. Además, permiten realizar las consultas y representar los resultados en entornos web y dispositivos móviles de un modo ágil e intuitivo, con el fin de resolver problemas complejos de planificación y gestión, conformándose como un valioso apoyo en la toma de decisiones [5].

3.3 SISTEMAS HOMBRE-MÁQUINA – ERGONOMÍA

La ergonomía estudia la interacción humana en cualquier sistema intervenible, es decir, en cualquier sistema que pueda ser objeto de la intervención humana para hacerle más confortable, de más fácil uso, para la actividad desempeñada por el hombre, abarcando los problemas del medio ambiente (puesto) de trabajo, así como las herramientas que son utilizadas para ello.

El objetivo de la ergonomía es mejorarla calidad de vida del trabajador en su ambiente de trabajo, mejorando la calidad en la interacción hombre-máquina, a través de acrecentar la eficacia funcional de las herramientas, se busca incrementar el bienestar del usuario final aumentando los valores de seguridad, salud y satisfacción en la labor realizada.

Para lograr incrementar la eficacia de las herramientas, es imprescindible considerar los aspectos humanos en su diseño. Esta máxima la comprendió muy bien la compañía Macintosh™, cuando interpretó que los usuarios de las

computadoras personales no tenían porque ser expertos informáticos, se buscó una herramienta para facilitar la interacción entre el hombre y la maquina creando la interfaz gráfica de usuario (GUI), la cual se basa en principios relativamente simples, como crear en la pantalla analogías graficas de objetos familiares del mundo real y así transmitir la idea de que la información digital es tan fácil de utilizar como un objeto cotidiano. El siguiente enlace contiene información sobre el diseño, mejoramiento e interacción de las interfaces [Semiótica de las interfaces] La usabilidad es parte del área de estudio de la ergonomía, más específicamente de la ergonomía de software, la cual busca encontrar los fundamentos teóricos para el diseño de interfaces de hombre-máquina, basándose en otras disciplinas que le permiten crear un marco teórico, tal es el caso de la teoría de Gestalt [Gestalt] y sus principios , mientras que la usabilidad es una disciplina del área de la ingeniería más práctica, y que se ha fundamentado en el análisis de casos de observaciones a través de test de usabilidad, que son procedimientos de análisis aplicados a los usuarios finales de un producto. La usabilidad es una propiedad de la interfaz hombre-máquina que confiere calidad al software, refiriéndose a la calidad de uso del producto [5.1].

3.4 TECNOLOGÍAS WEB

Es fundamental tener presente que, generalmente y más en el momento actual, los portales Web contienen elementos que interactúan facilitando la comunicación real entre la información y los usuarios, por ende la arquitectura Web es un conglomerado de acciones tendientes al desarrollo de páginas web y su optimización en torno al posicionamiento.

Una aplicación Web es proporcionada por un servidor Web y utilizada por usuarios que se conectan desde cualquier punto vía clientes Web (browsers o navegadores). La arquitectura de un Sitio Web tiene tres componentes principales:

- Un servidor Web, una conexión de red, uno o más clientes.

En algunos ambientes de desarrollo de aplicaciones Web, las páginas contienen código HTML y scripts dinámicos, que son ejecutados por el servidor antes de entregar la página [6].

3.4.1 Html (HiperText Markup Language). Se estructuran diferentes tipos de lenguajes para la calidad y eficiencia de la arquitectura. Logrando así obtener un alto nivel en el diseño de páginas web.

Las páginas web pueden ser vistas por el usuario mediante un tipo de aplicación llamada navegador. Podemos decir por lo tanto que el HTML es el lenguaje usado por los navegadores para mostrar las páginas web al usuario, siendo hoy en día la interfaz más extendida en la red, por lo tanto es el lenguaje con el que se escriben las páginas web.

Este lenguaje nos permite aglutinar textos, sonidos e imágenes y combinarlos a gusto propio. Además, y es aquí donde reside su ventaja con respecto a libros o revistas, el HTML nos permite la introducción de referencias a otras páginas por medio de los enlaces hipertexto [6.1].

3.4.2 JavaScript. Este tipo de lenguaje posee unas características que lo hacen especialmente idóneo para trabajar en Web.

Las dos principales características de JavaScript son, por un lado que es un lenguaje basado en objetos y por otro que es un lenguaje orientado a eventos, debido al tipo de entornos en los que se utiliza. Esto implica que gran parte de la programación en JavaScript se centra en describir objetos y escribir funciones que respondan a movimientos del ratón, pulsación de teclas, apertura y cerrado de ventanas o carga de una página, entre otros eventos [6.2].

3.4.3 JSON (JavaScript Object Notation). Es una sintaxis para almacenar e intercambiar datos. JSON es un texto escrito con la notación de objetos JavaScript.

- Intercambio de datos

Al intercambiar datos entre un navegador y un servidor, los datos sólo pueden ser texto. JSON es texto, y podemos convertir cualquier objeto JavaScript en JSON, y enviar JSON al servidor. También podemos convertir cualquier JSON recibido del servidor en objetos JavaScript.

De esta manera podemos trabajar con los datos como objetos JavaScript, sin análisis complicado y traducciones.

- Envío de datos

Si tiene datos almacenados en un objeto JavaScript, puede convertir el objeto en JSON y enviarlo a un servidor [6.3].

3.4.4 Web Services. El término Web Services describe una forma estandarizada de integrar aplicaciones WEB mediante el uso de XML, SOAP, WSDL y UDDI sobre los protocolos de la Internet. XML es usado para describir los datos, SOAP se ocupa para la transferencia de los datos, WSDL se emplea para describir los servicios disponibles y UDDI se ocupa para conocer cuáles son los servicios disponibles. Uno de los usos principales es permitir la comunicación entre las empresas y entre las empresas y sus clientes. Los Web Services permiten a las organizaciones intercambiar datos sin necesidad de conocer los detalles de sus respectivos Sistemas de Información [7].

3.4.5 WebSockets. Representan una evolución esperada en la tecnología web cliente / servidor. Permiten una conexión de socket TCP única de larga data que se establece entre el cliente y el servidor que permite bidireccional, full dúplex, mensajes para ser distribuidos al instante con poca sobrecarga que resulta en una conexión muy baja latencia.

Tanto el API Web Socket y el protocolo de WebSocket están estandarizados que significa que la web tiene ahora una norma acordada para la comunicación en tiempo real entre clientes y servidores de Internet. Originalmente considerada una tecnología de navegador, WebSockets están llegando mucho más allá de los navegadores web y se están convirtiendo en un estándar multiplataforma para la comunicación en tiempo real entre el cliente y el servidor [8].

Se debe tener en cuenta que SIGMAT hace parte de un macro proyecto y que este tipo de implementación no se manejan solo de una lado, si no que por el contrario se necesitan dos partes tanto el front end-SIGMAT como el back end-Web Server para hacer posible dicha implementación.

Aun así y pensando en las mejoras continuas que SIGMAT pueda tener, queda totalmente a disposición para implementar nuevas mejoras para el beneficio de un posible producto final.

4. METODOLOGÍA

4.1 PROTOTIPOS EVOLUTIVOS: ENFOQUE GENERAL

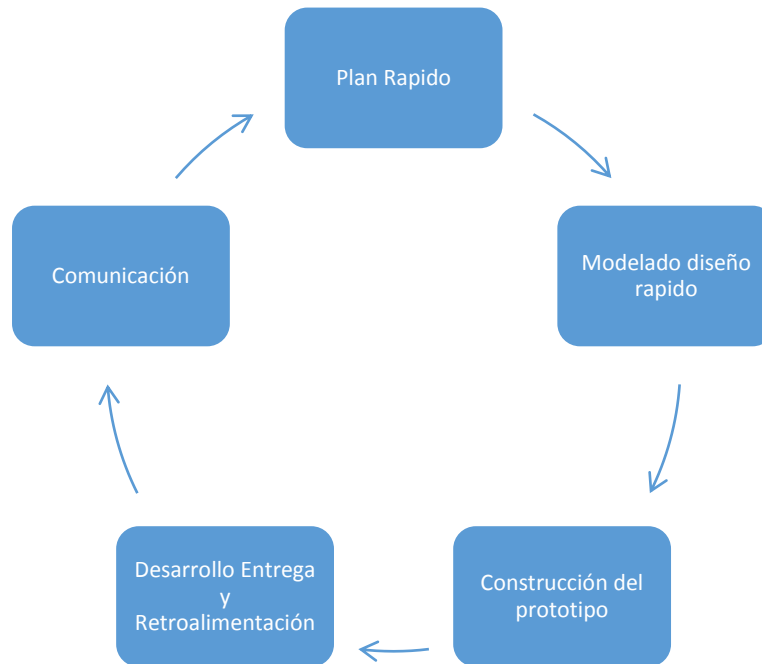
Este modelo se basa en la idea de desarrollar una implementación inicial, exponiéndola a los comentarios del usuario y refinándola a través de las diferentes versiones que se generan hasta que se desarrolle un sistema adecuado.

Las actividades de especificación, desarrollo y validación se entrelazan en vez de separarse, con una rápida retroalimentación entre estas. Existen dos tipos de desarrollo evolutivo:

- Desarrollo exploratorio, en este caso el objetivo del proceso es trabajar con el cliente para explorar sus requerimientos y entregar un sistema final. El desarrollo empieza con las partes del sistema que se comprenden mejor. El sistema evoluciona agregando nuevos atributos propuestos por el cliente.
- Prototipos desechables, el objetivo de este proceso de desarrollo evolutivo es comprender los requerimientos del cliente para así desarrollar una definición mejorada de los requerimientos para el sistema. El prototipo se centra en experimentar los requerimientos del cliente que no se comprenden del todo.

Haciendo referencia a la producción del software, un enfoque evolutivo suele ser más efectivo que el enfoque en cascada, ya que satisface las necesidades inmediatas de los clientes. La ventaja de un software que se basa en un enfoque evolutivo es que las especificaciones se pueden desarrollar de forma creciente. Tan pronto como los usuarios desarrollen un mejor entendimiento de su problema, esto se puede reflejar en el software [9].

Figura 1. Actividades del modelo evolutivo

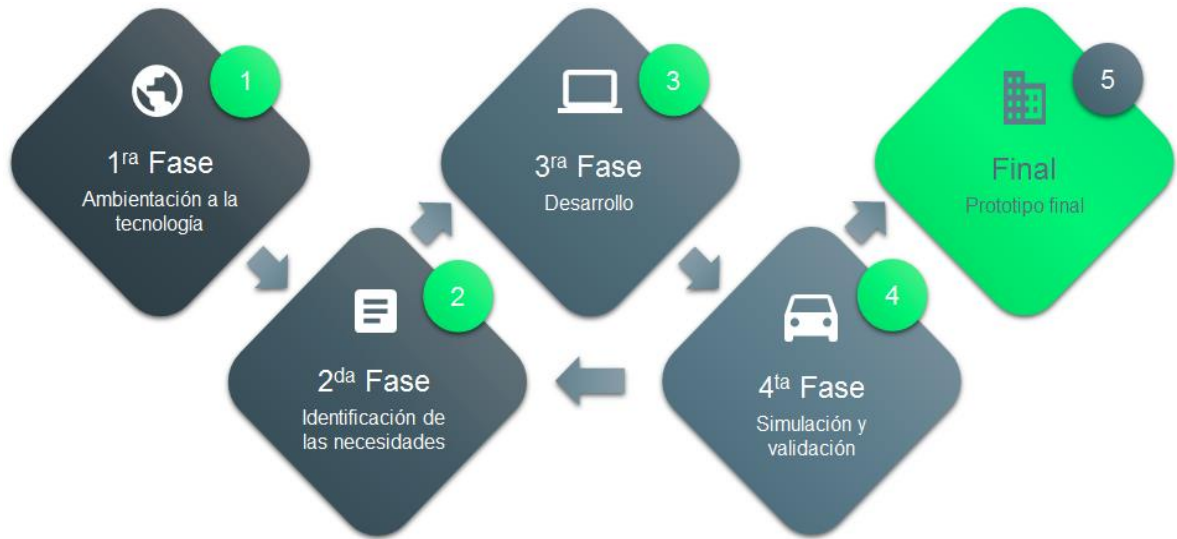


4.2 METODOLOGÍA UTILIZADA Y PLAN DE TRABAJO

La metodología con la que se llevará a cabo el proyecto, será una **adecuación de la metodología de prototipos evolutivos** para el desarrollo de software, buscando como finalidad una ejecución rápida que garantice el cumplimiento de los objetivos propuestos dentro del marco de referencia del proyecto, dando lugar a una mayor solidez y un componente de innovación al sistema de monitoreo web que se tiene como fin a realizar.

En el siguiente gráfico podemos observar el nombre de cada una de las fases que se llevarán a cabo junto con el orden en que se realizarán, en la fase cuatro podemos notar que hay dos posibles caminos, esto es debido a que serán realizados dos prototipos, por lo que las fases dos, tres y cuatro se realizarán dos veces.

Figura 2. Fases de actividades dentro de la metodología adecuada a utilizar



4.2.1 Primera Fase: Ambientación a la tecnología. En la fase uno del proyecto se enfatiza en la investigación estudio y selección de las herramientas software que se usarán en la ejecución del proyecto.

Se hará una investigación enfocada en buscar como fin, hacer uso de las tecnologías más adecuadas, buscando un factor diferenciador, atrayente e innovador.

4.2.2 Segunda Fase: Identificación de las necesidades. En esta fase nos concentramos primordialmente en el estudio y selección de la arquitectura de software, buscando que se adapte a las necesidades del sistema de monitoreo. Además identificar las necesidades, para lo cual haremos uso de diversas mecánicas para el descubrimiento de requisitos como:

- Observaciones y recolección de datos de rutas en un sistema de transporte BRT.
- Entrevistas con funcionarios de un sistema de transporte BRT.
- Lectura juiciosa de estudios de movilidad tanto nacionales como internacionales.

4.2.3 Tercera Fase: Desarrollo. Durante esta fase nos enfocaremos en el desarrollo, tanto de una interface gráfica básica como del desarrollo en general del sistema de monitoreo, realizando un esfuerzo mucho mayor en este último componente, priorizando las funcionalidades críticas por sobre los demás aspectos.

Luego en la siguiente iteración el foco pasará a darle funcionalidad adicionales al sistema de monitoreo, y crear una interface gráfica de usuario nueva, que integre de forma positiva todos los elementos y funcionales del sistema de monitoreo.

4.2.4 Cuarta Fase: Simulación y validación. En la cuarta fase se hará la validación del sistema por medio de innumerables pruebas, entre las que se destacan:

- Prueba del sistema de monitoreo web por medio de una simulación de un sistema BRT de pequeña escala.
- Pruebas de compatibilidad con diferentes navegadores web y SS.OO.
- Prueba evaluación heurística

4.2.5 Quinta Fase: Prototipo final. Por último, la quinta fase es la finalización y entrega del prototipo final funcional, garantizando la ejecución de cada fase y puesta en marcha de los objetivos propuestos al empezar.

5. IMPLEMENTACIÓN

5.1 HERRAMIENTAS DE DESARROLLO

Previamente a la inicialización de la implementación del prototipo web, se realizó una investigación de las posibles herramientas y tecnologías a utilizar, seguido de esto se hace una reunión con todo el grupo de trabajo y se establecen los software más óptimos y adecuados para este proyecto, garantizando escalabilidad en la plataforma web y brindando soporte en cada una de las herramientas seleccionadas y estableciendo como lenguaje de programación JavaScript. Se escoge el formato JSON para el intercambio de la información entre el cliente - (SIGMAT) y el servidor.

Las siguientes herramientas, serán empleados en la construcción e implementación del prototipo y montaje de todos los elementos que constituirán el contenido del sitio web:

- ***Framework AngularJS***

AngularJS es un framework *MVC* (Modelo Vista Controlador) de código abierto desarrollado por Google y escrito en Javascript, que trabaja del lado del cliente (*client-side*) y permite hacer más dinámica la aplicación web, trabajando de la mano con otras tecnologías como HTML y CSS, así como librerías de terceros.

Además AngularJS permite realizar aplicaciones de tipo **SPA**¹ (*Single Page Applications*), lo que significa que podemos construir una aplicación web en donde una parte de la misma cargue dinámicamente, sin que se tenga que recargar toda la página. Esto permite hacer una aplicación web más rápida y fácil.

Parte importante de este proyecto es poder consumir un servicio-API REST, es por eso que angular también permite realizar solicitudes HTTP de manera asíncrona con Javascript para obtener datos de un servidor y mostrarlos en el cliente sin tener que recargar la página entera.

El servicio `$http` (service en inglés, tal como se conoce en AngularJS) es una funcionalidad que forma parte del núcleo de Angular. Sirve para realizar comunicaciones con servidores por medio de HTTP a través de Ajax y vía el objeto XMLHttpRequest nativo de Javascript o vía JSONP.

`$http` tiene varios tipos de acciones posibles. Todos los puedes invocar a través de los parámetros de la función `$http` y además existen varios métodos alternativos (atajos o shortcuts) que sirven para hacer cosas más específicas [10].

Entre los shortcuts encuentras:

- **`$http.get()`**, `$http.post()`, `$http.put()`, `$http.delete()`, `$http.jsonp()`, `$http.head()`, `$http.patch()`.

En este sistema de información SIGMAT se inclinó a darle prioridad a el método **`$http.get()`**, este shortcuts permite realizar todas las peticiones pertinentes al servidor, obteniendo en un formato específico toda la información que se requiere

¹ Es un sitio web que cabe en una sola página con el propósito de dar una experiencia más fluida a los usuarios como una aplicación de escritorio.

del servidor, SIGMAT queda abierta a cualquier implementación de los métodos shortcuts restantes.

- **Framework JQuery**

jQuery es un peso ligero, "escribir menos, hacer más", biblioteca de JavaScript. El propósito de jQuery es que sea mucho más fácil de usar JavaScript en su página web. jQuery toma una gran cantidad de tareas comunes que requieren muchas líneas de código JavaScript para llevar a cabo y los envuelve en los métodos que se pueden llamar con una sola línea de código.

jQuery también simplifica mucho las cosas complicadas de JavaScript, como las llamadas AJAX y la manipulación del DOM [11].

La librería jQuery contiene las siguientes características:

- HTML manipulación / DOM
- la manipulación de CSS
- métodos de evento HTML
- Efectos y animaciones
- AJAX
- Utilidades

jQuery fue usado intensivamente, su corta curva de aprendizaje junto con las grandes posibilidades que ofrece para realizar una manipulación del DOM de forma sencilla, lo hizo imprescindible a la hora de dotar de funcionalidades la plataforma web SIGMAT.

Este Framework además da la posibilidad de manipular elementos de DOM, ofrece un sin número de funciones sobre estos mismo elementos que a medida que avanzaba el desarrollo y se hacían necesarias funciones más avanzadas y

específicas, JQuery iba un paso adelante y al consultar su documentación está casi siempre ofrecía o la funcionalidad necesaria o funcionalidades que usadas en conjunto permitían desarrollar más rápidamente la funcionalidades específicas que se requería para dotar de una mayor funcionalidad a la plataforma web.

- ***Framework Materialize***

El diseño, es un pilar fundamental en este proyecto, se sabe que la primera impresión de un sitio web puede determinar factores como credibilidad y grado de confianza. Para ello es fundamental que la información que se quiera transmitir sea clara, estructurada y atractiva para que el público pueda entenderla y sea atraído por ella en mayor o menor medida. Es por eso que este proyecto quiere brindar un aspecto externo atractivo, capaz de suscitar el interés y de llamar la atención del público objetivo, el usuario-operador.

Con materialize se quiere brindar un aspecto diferente de monitorear rutas, una alternativa diferente para visualizar la información con una perspectiva en línea de tiempo.

Es allí, donde nace la importancia de seleccionar una herramienta capaz de brindar funcionalidades que soporten cada una de las necesidades. Materialize es un framework Css que permite crear sitios y aplicaciones web con los principios de Material Design².

Puede ser usado de dos formas, Materialize Y Sass, dependiendo de las preferencias y la experiencia se puede seleccionar cualquiera de las dos versiones.

² Es un lenguaje de diseño que combina los principios básicos del diseño efectivo, con innovación y tecnología.

La versión estándar contiene Css y JavaScript y trae los CSS listos para trabajar. La otra versión, recomendada para los que están familiarizados con Sass, permite tener mayor control sobre los componentes que quieres incluir.

La herramienta incluye también flow text(responsive text), botones, check boxes, barras de navegación, estilos para tablas y cards. Además múltiples iconos de Material Design.

Toasts: Son contenidos que nos son visibles en la página pero aparecen bajo la forma de información adicional y/ alertas. También tienen diseño responsive.

Modals: Mensajes adicionales que oscurecen la pantalla principal y aparecen en primer plano. [12]

Este framework proporciona todo lo necesario para crear una plataforma web, totalmente basada en material Design.

- ***Ui-Grid de AngularJs***

SIGMAT, cuenta con diferentes tipos de monitoreo, uno de ellos es poder visualizar la información extraída del servidor en tablas, es por esto que se escoge Ui-grid un módulo desarrollado completamente por angularJS, que va a permitir mostrar la información adecuada de una manera ordena, totalmente dinámica y con estilo cumpliendo con las necesidad del usuario-operador.

Un elemento como un grid (traducido literalmente como rejilla) permite presentar en la interfaz de usuario grandes conjuntos de datos.

Las funcionalidades que se pueden encontrar en el módulo ui-grid son variadas dependiendo del nivel de complejidad que se necesite para mostrar los datos al usuario y de la interacción que se quiere ofrecer con el ui-grid.

Ui-Grid permite desde Ordenar, ocultar y filtrar, hasta incluir agrupación y expansión de filas y edición de campos, en muchas otras funcionalidades.

- ***Maps JavaScript API***

Parte del objetivo de SIGMAT es poder brindarle al tipo de usuario-operador, diversas maneras de monitorear el servicio de transporte publico basado en BRT, por ende este proyecto cuenta con diferentes tipos de monitoreo el ya mencionado en el ítem anterior (tablas) y ahora una manera visual de georreferenciación de buses y rutas dentro de un mapa.

Es allí donde se piensa en la Api de Google maps, una potente herramienta que permite la creación de mapas y con ello garantizando la creación de aplicaciones atractivas para la web.

Permitiendo integrar mapas básicos, mapas con estilo, diversos marcadores personalizados como sencillos, líneas, polilínea, y muchas más funcionalidades logrando estar al alcance del proyecto, estos factores cumplen con cada una de las necesidad para mostrar un monitoreo optimo y funcional sobre este segundo tipo de monitoreo que se va a implementar.

Además es una herramienta que cuenta con un soporte de alto nivel, garantizando estabilidad y escalabilidad a cada uno de sus usuarios, su eficiencia y eficacia en la demarcación de calles, carreras, etc., hace que este api sea la mejor opción para la plataforma web.

5.2 ANÁLISIS DE REQUERIMIENTOS

En reuniones sostenidas con el director de proyecto, apoyadas por lecturas de estudios de movilidad, una visita realizada al centro de control y monitoreo del servicio de transporte público BRT de la ciudad y una entrevista con algunos funcionarios del sistema de transporte, se identificaron las siguientes necesidades:

- **Seguimiento y control:** Se identificó que no cuentan con un software capaz de brindarle interfaces con una visualización adecuada de la información (tablas, tiempo, alertas). Limitando al usuario operador realizar un seguimiento y control oportuno de las rutas asignadas.
- **Georreferenciación:** Se identificó que el modulo grafico que utilizan actualmente nos les permite si no identificar un bus en un aspecto muy grande geográficamente, complicando el uso de ese modelo gráfico.
- **Perdida de vigencia:** Se identificó que gran parte de las herramientas tecnológicas utilizadas en el software actual del sistema no cuentan con un soporte activo, generando obsolescencia del software actual.

Se determinaron los siguientes requisitos que debe tener la solución software a implementar.

La Plataforma Web SIGMAT (Sistema de Información para la gestión y monitoreo automático del transporte), ofrecerá al usuario-operador en tiempo cuasi-real información general sobre rutas, buses, estaciones, conductores, horas, en general todo su itinerario correspondiente a todo un sistema de transporte en BRT.

Contará con un diseño capaz de determinar factores como credibilidad y grado de confianza, garantizando que la información mostrada sea atractiva, clara y estructurada y que le facilite al usuario-operador tomar decisiones tanto de largo

alcance y alto impacto como de bajo alcance y bajo impacto que influyan en un mejor servicio en todo el sistema de transporte.

Además contara con diferentes tipos de monitoreo distribuidos de la siguiente manera:

- **MONITOREO EN LÍNEA DE TIEMPO – MÚLTIPLES RUTAS:**

El monitoreo de rutas sobre líneas de tiempo es una forma muy usada para el monitoreo de buses. SIGMAT contara con un sistema de monitoreo denominado Línea de tiempo múltiple, un tipo de monitoreo que le brinda la posibilidad al usuario-operador visualizar información de sus rutas de una manera múltiple, otorgándole la posibilidad de seleccionar máximo cinco (5) rutas diferentes en una misma interfaz, ofreciendo una perspectiva diferente de monitoreo, la información a visualizar en esta interfaz es:

- Total de estaciones correspondiente a las rutas seleccionadas, distribuidas de manera equitativas.
- Flota de buses activas en cada una de las rutas seleccionada.
- Identificadores de cada uno de los buses en sus respectivas rutas.
- Estado de los buses en todo su recorrido, alterándose según los identificadores de movilidad planteados.

- **MONITOREO EN LÍNEA DE TIEMPO - ÚNICA RUTA:**

El monitoreo en líneas de tiempo también contara con un tipo de visualización individual e independiente de todos los buses de una misma ruta, la información que a visualizar en esta interfaz es:

- Total de estaciones que corresponden a la ruta seleccionada, distribuidas de manera equitativas.
- Flota de buses activas en la ruta seleccionada.

- Identificadores de cada uno de los buses en la ruta seleccionada.
- Estado de los buses entre cada estación, alterándose según los identificadores de movilidad planteados.
- Validación en la diferencia de tiempo entre las horas de llegada reales y estimadas entre dos estaciones.

- **MONITOREO EN TABLAS - ÚNICA RUTA:**

SIGMAT, contará con un tercer tipo de monitoreo, denominada monitoreo en tablas para una única ruta, en esta interfaz el usuario-operador podrá obtener información correspondiente a su itinerario como:

- Nombre del conductor, placa del bus, id del bus, estado del bus, sus respectivos horarios estimados y reales de su recorrido en cada estación.
- Estado del bus en su recorrido. (Aun en actividad) o (Finalizada actividad).
- Diferencia entre las horas estimadas y horas reales del bus en cada llegada a su respectiva estación, alterándose según los identificadores de movilidad planteados.

SIGMAT también le ofrecerá al usuario-operador una manera atractiva de realizar:

- Reportes

Estará a la disposición del usuario-operador que datos de la información mostrada en la tabla le sea más óptima para realizar un reporte, todo esto en formato pdf.

- **MONITOREO EN MAPA**

Por último SIGMAT contará con un cuarto tipo de monitoreo, en mapa y para ello este sistema de información va de la mano de una gran herramienta Maps JavaScript API.

SIGMAT ofrecerá un sistema de monitoreo bastante intuitivo e inteligente sobre el mapa, brindándole al usuario-operador una georreferenciación clara, ordenada, y bastante atractiva de todos los:

- Buses con su respectiva información (identificador, ruta)
- Estaciones con su respectiva información (nombre)

Correspondientes a una ruta, además el usuario-operador podrá consultar todas las:

- Estaciones con su respectiva información (nombre)
- Paradas con su respectiva información (nombre)

Existentes actualmente en el sistema de transporte público.

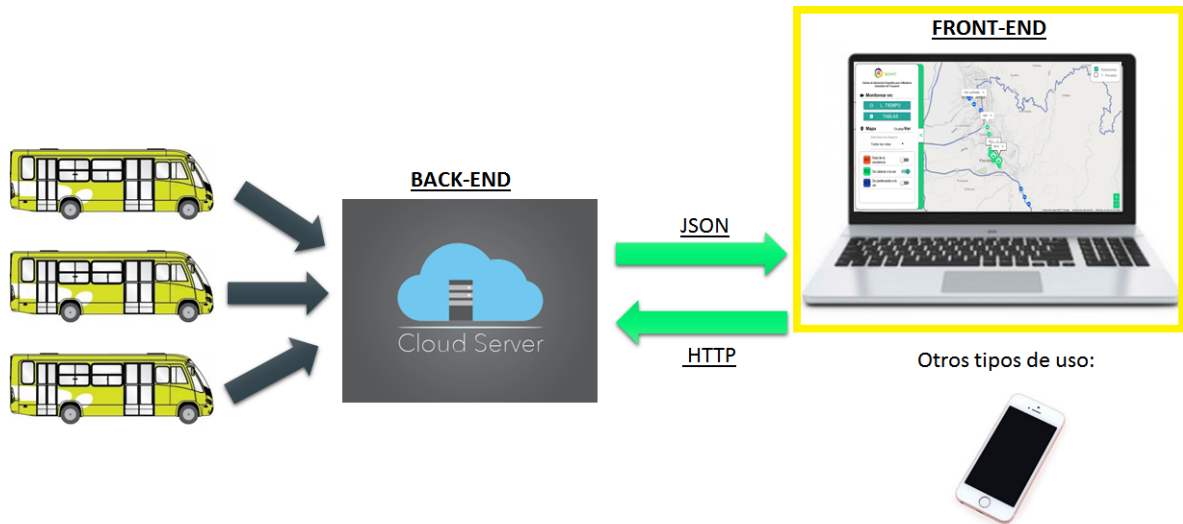
5.3 ARQUITECTURA GENERAL

SIGMAT es una plataforma web que no solo se alimenta de datos creados localmente por el programador, SIGMAT hace parte de una arquitectura general y sus componentes son mostrados en la figura 3.

Los buses están equipados con un sistema software embebido que se va a encargar de transmitir la información de los sensores que tienen dentro, esta información va a llegar a un plataforma Cloud que se encarga de procesar y coleccionar la información y ponerla disponible a diferentes aplicaciones del front-end a través de peticiones api rest.

En la arquitectura general de este proyecto estamos concentrados en la parte front-end (ordenadores) que corresponde a el monitoreo del servicio para un usuario operador.

Figura 3. Arquitectura general del macro-proyecto - SIGMAT



5.4 ARQUITECTURA SIGMAT

La arquitectura implementada para el desarrollo de la plataforma web fue un estilo arquitectural modelo - vista - controlador (MVC)³, esta decisión se tomó principalmente por la ventajas que ofrece en cuanto a separación de componentes diferenciales, como los son la parte grafica (la vista), la lógica del sistema y la recepción de los eventos resultado de la interacción del usuario con la interfaz gráfica (el controlador) y finalmente el modelo, que es el lugar donde la plataforma web se comunica con el servidor.

Indiscutiblemente el tener separados estos conceptos da un gran poder de maniobrabilidad al mismo tiempo que le da orden a la estructura del software, lo que permite realizar cambios de forma más eficiente y eficaz.

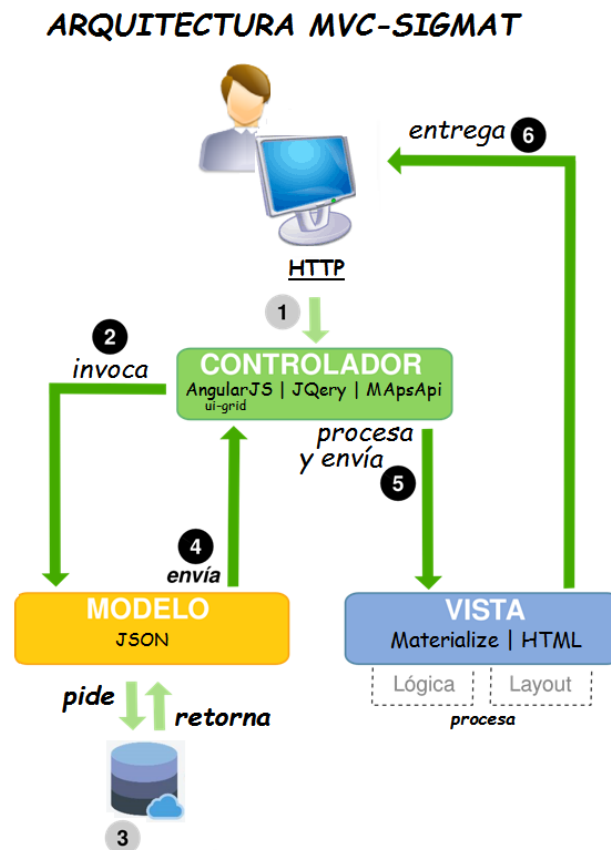
³ WIKIPEDIA LA ENCICLOPEDIA LIBRE Modelo Vista Controlador. [en línea] disponible en: <https://es.wikipedia.org/wiki/Modelo%E2%80%93vista%E2%80%93controlador>

El **Modelo** incorpora la capa del dominio y persistencia, es la encargada de guardar los datos en un medio persistente (En este proyecto el web server retorna los datos en formato JSON). En el modelo es donde se hace el levantamiento de todos los objetos que el sistema debe utilizar, es el proveedor de los recursos.

La **Vista** se encarga de presentar la interfaz al usuario en diferentes sistemas web, esto es típicamente HTML, aunque pueden existir otro tipo de vistas. En la vista se deben hacer operaciones simples, como ifs, ciclos, formateo, etc.

El **Controlador** es el que escucha los cambios en la vista y se los envía al modelo, el cual regresa los datos a la vista, es un ciclo donde cada acción del usuario causa que se inicie de nuevo un ciclo.

Figura 4. Arquitectura MVC-SIGMAT



Se definieron las clases dominio (Modelo) para no tener acoplamiento ni visibilidad directa respecto a las clases ventana (Vista) y para que los datos de la aplicación y de la funcionalidad se conservaran en las clases de dominio y no en las de ventana. Se definieron las clases manejadores (Controlador) para que procesen los eventos (Peticiónes) al sistema y re direccionen a las clases dominio y ventana tanto la visualización como el procesamiento de resultados respectivamente.

Obteniendo como beneficios:

- Menor acoplamiento
- las vistas proveen mayor flexibilidad y agilidad
- Mayor facilidad para el desarrollo de clientes en múltiples dispositivos
- Más claridad de diseño
- Facilita el mantenimiento
- Mayor escalabilidad

Y otorgando desacoplamiento, evolución y escalabilidad a la plataforma web.

5.5 PROTOTIPO INICIAL

5.5.1 Generalidades del prototipo inicial. El proceso de construcción del prototipo inicial comenzó desde el primer momento en que se realiza las reuniones con el director y codirector de proyecto, seguido de una serie de investigaciones sobre movilidad y sobre el transporte público en brt.

En las reuniones se escuchó y analizó cada uno de los requerimientos deseados para el sistema de información. Dicha información permitió construir un prototipo inicial el cual sirvió como guía para el desarrollo del software posteriormente.

El prototipo inicial se creó con una única página HTML, la cual permitía visualizar la estructura de navegación, reduciendo el nivel de complejidad en la labor de

determinar la información que se le iba a mostrar al usuario, el contenido de cada una de las vistas e intuir algunos procesos que se llevan a cabo dentro un centro y control de monitoreo.

Se manejó un proceso de refinamiento y perfección cíclico que permitiera llegar a un nivel de acercamiento detallado de lo que quiere alcanzar el usuario con el sistema, logrando reducir el tiempo de perfección del prototipo final agilizando de esta forma el ciclo de vida del proyecto.

5.5.2 Diseño Interfaz del prototipo inicial - Primera iteración. En esta primera iteración se realiza un diseño de interfaz básico, enfocado en priorizar las funcionalidades que va a tener el sistema de monitoreo.

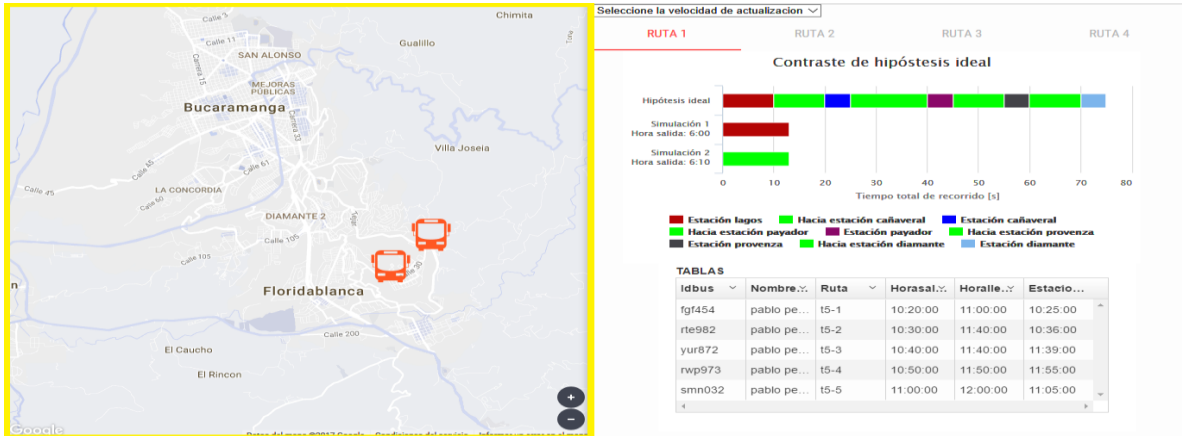
En este primer prototipo se establece una sola interfaz gráfica y tres (3) funcionalidades distribuidas de la siguiente manera y demarcadas por el color amarillo en sus respectivas figuras:

- **Visualización de la información sobre un mapa**

Esta primera funcionalidad es capaz de mostrar la información correspondiente de los recorridos con sus estaciones de todos los buses de su respectiva ruta, de una manera independizada y eficiente, agregando marcadores distintivos entre buses y estaciones según la ruta seleccionada.

En este primer foco se le da un nivel de importancia bastante alto a este primero tipo de monitoreo, estando por encima de las otras dos funcionalidades y otorgándole más prioridad visual en la interfaz gráfica.

Figura 5. Prototipo inicial - primer prototipo inicial - primera iteración - mapa

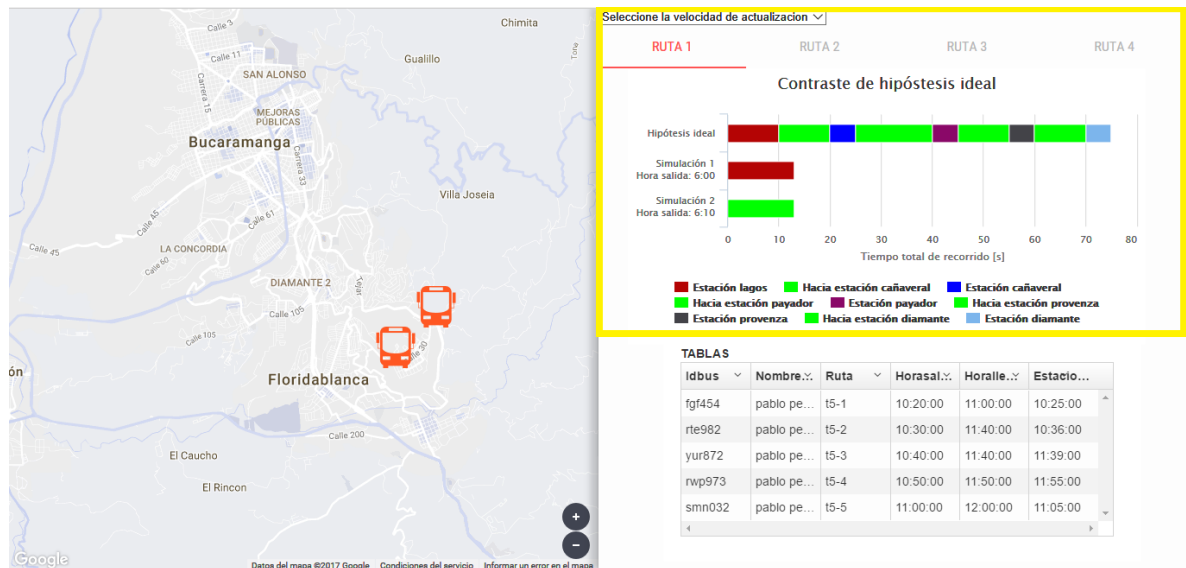


- **Visualización de la información sobre grafica**

La segunda funcionalidad se enfoca en crear y mostrar de manera eficiente y atractiva nuevos identificadores de movilidad, identificadores diferenciados por tres (3) tipos de colores, los cuales van cambiando e interactuando según el comportamiento que tengan los buses en su recorrido entre cada una de sus estaciones.

Ofreciendo un tipo de monitoreo diferente con un nivel de importancia alto y obteniendo un espacio visual medio en la interfaz gráfica.

Figura 6. Prototipo inicial - primer prototipo inicial - primera iteración - grafica

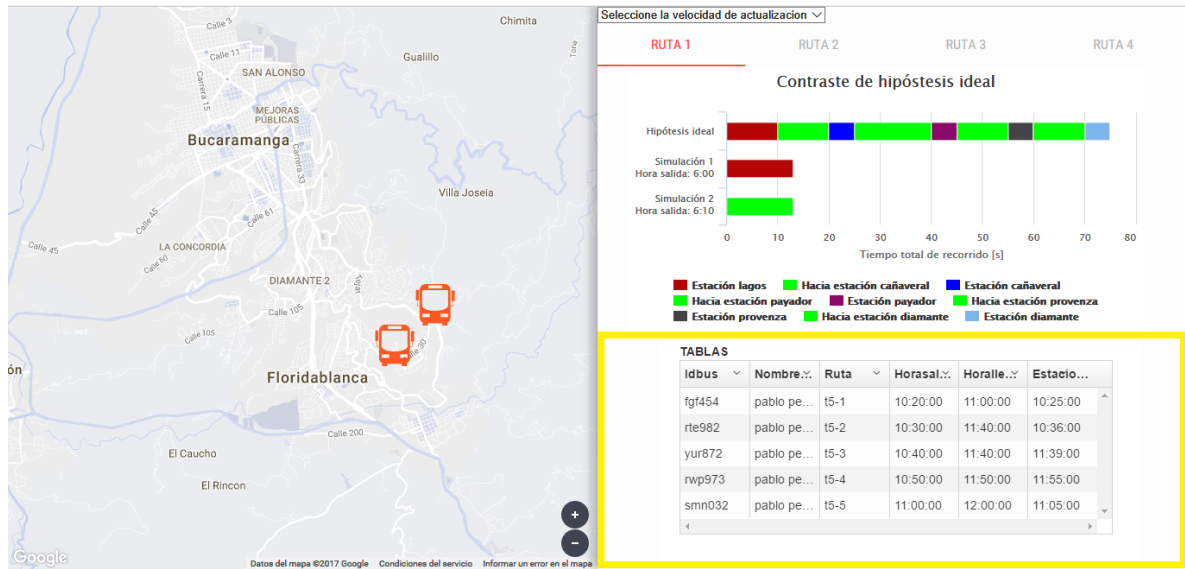


- **Visualización de la información sobre tablas**

Por último, SIGMAT incluye una funcionalidad capaz de mostrar los datos correspondientes al itinerario de cada una de las diferentes rutas, con este sistema de monitoreo se quiere dar un aspecto muy familiarizado y concurrido como lo es la información en tablas.

Se le da un nivel de importancia igual a la funcionalidad numero dos (2) otorgándole el mismo espacio de visualización en la interfaz gráfica.

Figura 7. Prototipo inicial - primer prototipo inicial - primera iteración - tabla



5.5.3 Diseño Interfaz del prototipo inicial - Segunda iteración. Dada por concluida la primera iteración, se inicia una segunda, cumplimiento en su totalidad la tercera (3) fase de desarrollo de la metodología.

Se realizan diversas reuniones con el director y codirector de proyecto en el transcurso de esta tercera fase, además como grupo de trabajo se tiene la oportunidad de asistir al centro de control y monitoreo de la empresa que brinda el servicio de transporte público en BRT en Santander y se llegan a algunos acuerdos para agregar, modificar y adicionar nuevas funcionalidades, además para crear un diseño lo suficientemente bueno y agradable para el prototipo final.

Los acuerdos son mencionados a continuación:

- Modificar la funcionalidad de visualización de la información en gráfica, por una funcionalidad de visualización en línea de tiempo en una única ruta y en múltiples rutas, conservando de la funcionalidad anterior los innovadores identificadores de movilidad.

- Independizar las funcionalidades en interfaces diferentes, permitiéndole al usuario-operador una mejor visualización de la información para los cuatro (4) tipos de monitoreo.
- Crear una interfaz principal lo suficientemente bien diseñada para intuirle al usuario-operador cada una de los sistemas de monitorio disponibles que ofrecería la plataforma web SIGMAT.
- Incluir características innovadoras en la interfaces del sistema de monitoreo que permitan un entorno de navegación agradable para el usuario y capaz de mostrar la información de la manera más simple y sencilla pero lo suficientemente eficiente para la toma de decisiones por parte del usuario-operador.

Posteriormente y después de aceptados los acuerdos mencionados anteriormente, se inicial el diseño de las nuevas interfaces y de la inclusión de las nuevas funcionalidades en el prototipo inicial funcional.

Los cambios son mostrados a continuación, las figuras están demarcadas de color amarillo según su edición:

- **Visualización de la información interfaz principal.**

En esta primera modificación con respecto al prototipo inicial-primera iteración, se enfoca en agregar una pantalla principal lo suficientemente dinámica, agregando características innovadoras como lo son las ventanas modal y los checks box de tipo switch, radio buttons y casillas de selección para una navegación más fluida y entendible para el usuario-operador.

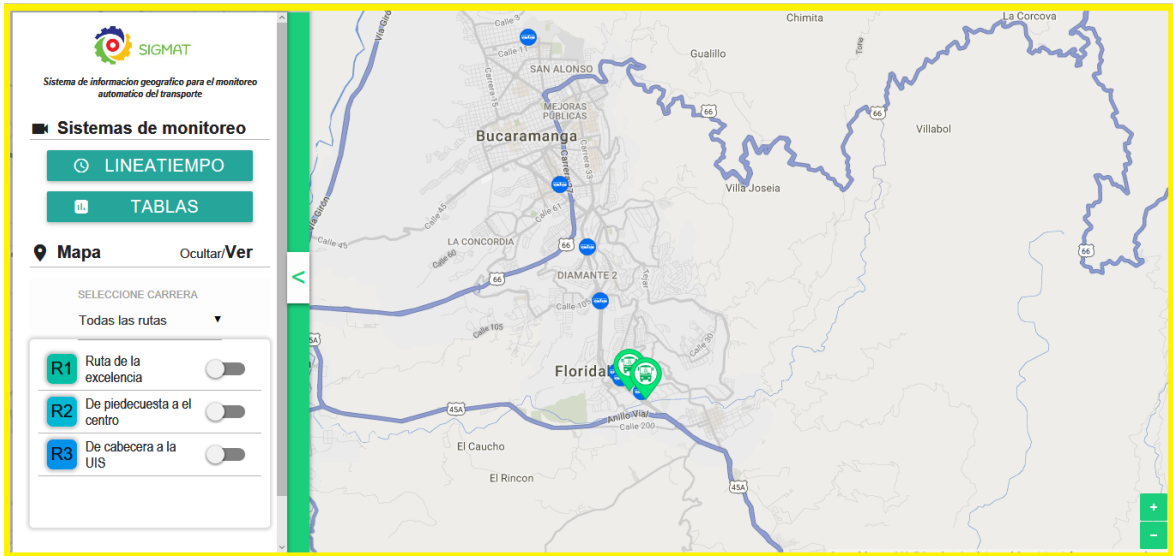
En esta nueva interfaz principal - segunda iteración, cambia el orden del nivel de importación de las funcionalidades respecto a la primera iteración, en esta primera interfaz se muestra en primera instancia una ventana modal que sobre sale de la interfaz principal ofreciéndole al usuario-operador dirigirse en primer momento al monitoreo en líneas de tiempo, como se muestra en la figura 8.

Figura 8. Prototipo inicial - segunda iteración - interfaz pantalla principal – Modal



Por otra parte, si el usuario-operador no ingresa a el tipo de monitoreo en líneas de tiempo priorizado en la pantalla modal por SIGMAT, puede cerrar dicha ventana y SIGMAT mostrar su interfaz principal, figura 9.

Figura 9. Prototipo inicial - segunda iteración - interfaz pantalla principal



- **Visualización de la información en líneas de tiempo.**

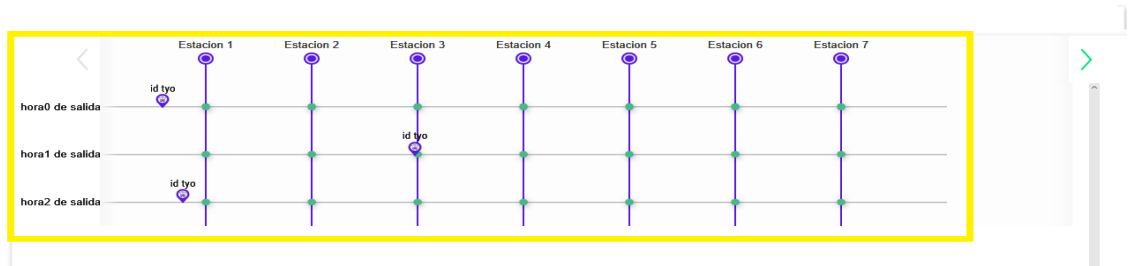
Esta segunda funcionalidad es el reemplazo de la visualización en gráfica, se hace este tipo de modificación para garantizar un aspecto de monitoreo más entendible y confiable para el usuario-operador, además es una manera óptima de representar la información en líneas rectas, extendiendo la capacidad de visualización de la información en la interfaz gráfica.

El nivel de importancia para la segunda iteración del prototipo inicial en líneas de tiempo es alto, se desea llevar al usuario-operador a tener de preferencia este sistema de monitoreo y que este sea el punto de partida hacia los otros sistemas de monitoreo.

Se agrega en el diseño factores innovadores en las líneas y en sus marcadores. Esta funcionalidad de visualización en línea de tiempo está dividida en dos (2):

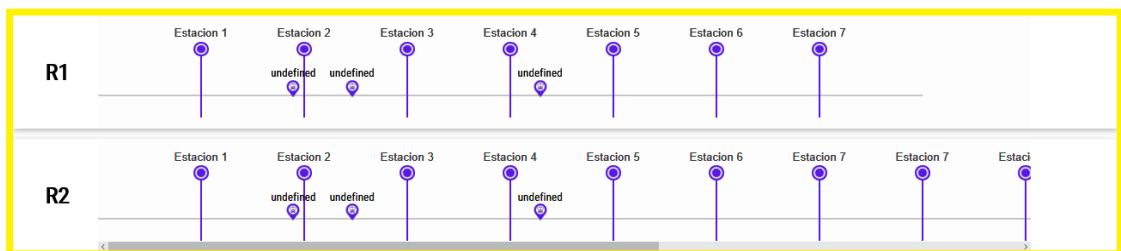
- El usuario operador podrá ver la interacción de cada bus de una ruta en diferentes líneas de tiempo en una misma interfaz gráfica.

Figura 10. Prototipo inicial -segunda iteración -interfaz línea de tiempo - única ruta



- El usuario operador podrá ver la interacción de todos los buses de una o múltiples rutas individualmente en una sola línea de tiempo y en una misma interfaz.

Figura 11. Prototipo inicial-segunda iteración-interfaz línea de tiempo - múltiple ruta



- **Visualización de la información en tablas.**

En esta segunda iteración la funcionalidad en la visualización de la información en tablas ha tomado un valor agregado bastante importante e influyente en la toma de decisiones por parte del usuario.

Además de mostrar la información correspondiente al itinerario de una ruta.

Se agregaron identificadores de movilidad capaces de informarle al usuario cualquier irregularidad del bus en su recorrido entre cada una de sus estaciones.

Por otra parte, le permite al usuario ocultar, mostrar las columnas y filas que el crea convenientes para la visualización de su información.

Figura 12. Prototipo inicial - segunda iteración - interfaz tablas

Bus	Conductor...	Ruta	Estado	Salida	Llegada E1	Llegada E2	Llegada E3	Llegada E4	Llegada E5*
ZOE101	1000000			13:00:00	13:09:00	13:23:47	13:29:52	13:35:25	13:45:25
ZOE202	1098468424			13:15:00	13:25:00	14:35:48	13:44:32	13:52:32	13:58:32
ZOE203	1096544654			13:32:00	13:41:00	13:51:48	13:57:55	14:06:26	
ZOE204	1096484894	R1-4		13:44:00	13:54:00	14:04:48	14:09:55	14:20:26	
ZOE205	1099999999	R1-5		14:00:00	14:11:00	14:20:48	14:26:55	14:34:26	
ZOE206	1333333333	R1-6		14:16:00	14:25:00	14:35:48	14:39:55		
ZOE207	1777777777	R1-7		14:30:00	14:39:00	14:51:48	14:55:55		
ZOE208	1888888888	R1-8		14:45:00	14:56:00	15:05:48	15:11:55		

Artículos Totales: 8 (Artículos Seleccionados: 1)

5.6 PRUEBA DE USABILIDAD EN SIGMAT POR PARTE DE LOS USUARIOS

DISEÑO DEL EXPERIMENTO

La prueba se realizara en un ambiente de oficina, con buena iluminación y con mucho ruido, siendo el lugar escogido la biblioteca de la UIS, lo que se busca es emular las condiciones bajo las cuales se hará uso cotidiano del software, dado que el lugar donde se realiza el monitoreo es compartido con los radio operadores del sistema, por lo cual es crucial, que el operador encargado de realizar el monitoreo, pueda interpretar la información de forma clara, rápida y sencilla, de lo contrario los factores externos (como el ruido) pueden hacer que el operador

interprete la información de forma errada y por ende tomar decisiones erróneas, en base a información inconsistente o ambigua.

El test tiene como objetivo medir la usabilidad del software SIGMAT, por medio de las siguientes métricas: eficacia (se medirá la task completion rate), eficiencia (se medirá la task time completion rate) y satisfacción (se usará la encuesta SUS). El test se ha realizado con la técnica think aloud, en la que los participantes piensan en voz alta mientras realizan la prueba y los evaluadores anotan tanto las dificultades con las que se encuentran los usuarios como los comentarios que consideran más interesantes.

La prueba se repetirá de forma iterativa a grupos diferentes de 4 personas hasta que se logre un diseño de GUI con una valoración superior a 4.0, teniendo en cuenta que en cada iteración se le realizará al software los ajustes que resulten necesarios, surgidos a partir de los comentarios de los usuarios y los resultados obtenidos en las pruebas, en las métricas antes expuestas.

PARTICIPANTES

Se les hará la prueba a grupos de 4 estudiantes, tres hombres una mujer, los tres hombres una mujer, esta proporción se estimó de acuerdo a la visita realizada a las instalaciones de metrolínea, se excluirán estudiantes de Ingeniería de sistemas o personas que no hayan tenido contacto con equipos de cómputo.

DISPOSITIVOS

Para la prueba se usará un portátil con pantalla de 15 pulgadas y relación de aspecto 16:9 con sistema operativo Windows 10, y se llevará a cabo en el navegador Google Chrome.

TAREAS

Los usuarios realizarán 4 tareas, siempre en el mismo orden:

- **Tarea 1:** Localice y habilite la visualización de la Ruta R2 en el Mapa.
- **Tarea 2:** Localice el monitoreo de Líneas de tiempo, seleccione la ruta R2, R3 y monitóreelas.
- **Tarea 3:** Localice el monitoreo de líneas de tiempo y habilite la visualización independiente de la ruta R3. (Esta tarea es similar a la tarea 2, con la diferencia de que al ser la segunda vez que el usuario monitorea una Ruta, y se espera que el usuario la realice con mayor solvencia).
- **Tarea 4:** Localice el monitoreo en Tablas, seleccione la ruta R2 y monitóreela.

PROCEDIMIENTO

La prueba se realizará de a un usuario por vez a un grupos de cuatro usuarios, primero se le realiza una corta encuesta para medir que tan familiarizado está con el sistema operativo y el programa sobre el cual este va a ejecutarse, conocer su nivel de estudio y definir si es un candidato apto.

Se les pedirá a los usuarios que comenten en voz alta lo que hagan en cada momento durante el test. No se les limitará el tiempo para realizar las tareas, y se les debe indicar que en cualquier momento podrán abandonarlas si no ven la forma de hacerlas. Mientras ellos hacían las tareas, un observador cronometrará el tiempo que tardaba el usuario en realizar cada una de las tareas, anotará si la

finalizaba con éxito y las dificultades que el usuario tenga durante la realización de la misma.

Esas anotaciones, son habituales en estudios de usabilidad (Frøkjær, Hertzum, Hornbæk; 2000) (ISO; 1998) se usaran para obtener las siguientes métricas:

EFFECTIVIDAD:

La efectividad se refiere a la capacidad del sistema para ofrecer las funcionalidades para las que se ha diseñado.

Durante el test se midió el tiempo que cada usuario tardaba en realizar cada tarea.

EFICIENCIA:

La eficacia al esfuerzo necesario para conseguir realizar estas funcionalidades.

Durante el test se midió el tiempo que cada usuario tardaba en realizar cada tarea.

SATISFACCIÓN:

Medida en la que el usuario se sintió al realizar la tarea (nada satisfecho o molesto, ni satisfecho-ni insatisfecho, muy satisfecho). **Tiempo:** Cantidad de tiempo que le toma al usuario realizar la tarea. **# Errores:** Cantidad de intentos fallidos realizando una tarea. **Sugerencias:** Opinión del usuario respecto a la tarea (Obligatoria).

Durante el test se anotaran todos los comentarios de los usuarios y, una vez finalizado, se les entregará un cuestionario SUS (Scale Usability System) (Brooke;

1996) para conocer su satisfacción con la plataforma y obtener la puntuación promedio de cada numeral.

Tabla 1. Prueba de usabilidad - datos

Nombre:	
Edad:	*****
Ocupación:	*****
Experiencia:	*****

EFFECTIVIDAD

Tabla 2. Prueba de usabilidad – efectividad

Tarea	Tiempo esperado (seg)	Tiempo Empleado (seg)	N° de función aprendidas	N° de errores	Sugerencias
TAREA #1	5	1- 12 2- 14 3- 13 4- 12	1- 2 2- 2 3- 2 4- 3	1- 3 2- 3 3- 4 4- 3	Consistencia en la navegación
TAREA #2	4	1- 8 2- 8 3- 9 4- 9	1- 3 2- 2 3- 2 4- 2	1- 3 2- 2 3- 4 4- 3	Mejorar la sintaxis
TAREA #3	4	1- 10 2- 9 3- 11 4- 10	1- 2 2- 2 3- 2 4- 2	1- 4 2- 6 3- 6 4- 5	Agregar más información en los sistemas de monitoreo
TAREA #4	7	1- 12 2- 14 3- 11 4- 17	1- 3 2- 3 3- 2 4- 2	1- 4 2- 3 3- 5 4- 3	Diseño

EFICIENCIA

Tabla 3. Prueba de usabilidad - eficiencia

Tarea	Tiempo empleado	N° de clicks realizados	Tiempo en cada pantalla	Sugerencias
TAREA #1	5	1- 7 2- 9 3- 8 4- 8	1- 14 2- 13 3- 11 4- 12	Consistencia en la navegación
TAREA #2	5	1- 7 2- 6 3- 7 4- 6	1- 9 2- 8 3- 7 4- 8	Agregar más información en los sistemas de monitoreo
TAREA #3	5	1- 4 2- 3 3- 5 4- 3	1- 11 2- 14 3- 12 4- 10	Mejorar la sintaxis
TAREA #4	5	1- 10 2- 12 3- 10 4- 9	1- 18 2- 19 3- 19 4- 17	Diseño

SATISFACCIÓN

De acuerdo a los pasos realizados para llevar a cabo cada tarea, ¿Cómo considera fue la respuesta de la aplicación en cada tarea realizada?

Tabla 4. Prueba de usabilidad - satisfacción

Tarea	Insatisfecho	Ni satisfecho, ni insatisfecho	Satisfecho
Localice y habilite la visualización de la Ruta R2 en el Mapa .	1- 2- X 3- 4- X	1- X 2- 3- X 4-	1- 2- 3- 4-

Tarea	Insatisfecho	Ni satisfecho, ni insatisfecho	Satisfecho
Localice el monitoreo de Líneas de tiempo , seleccione la ruta R2, R3 y monitoréelas.	1- 2- 3- 4-	1- 2- 3- X 4-	1- X 2- X 3- 4- X
Localice el monitoreo de líneas de tiempo y habilite la visualización independiente de la ruta R3	1- X 2- X 3- 4- X	1- 2- 3- X 4-	1- 2- 3- 4-
Localice el monitoreo en Tablas , seleccione la ruta R2 y monitoréela.	1- 2- 3- 4-	1- 2- 3- 4- X	1- X 2- X 3- X 4-

Instrucciones: Conteste por favor las siguientes preguntas, de acuerdo a su percepción como usuario.

1. ¿La información que se le presenta en pantalla le da los suficientes indicios para completar adecuadamente las tareas propuestas?

Nunca: ___ En Ocasiones: **_X_** Generalmente: ___ Casi siempre: ___
Siempre: ___

2. En relación a la información proporcionada por el observador, ¿Cómo la considera usted?

Nada de información: ___ Confusa: ___ Regular: **_X_** Clara: ___ Muy clara: ___

3. ¿La información que se le presenta en pantalla le da los suficientes indicios para completar adecuadamente las tareas propuestas?

Nada de información: ___ Confusa: ___ Regular: **_X_** Clara: ___ Muy clara: ___

4. ¿Cómo calificaría la interface gráfica de la plataforma web?

Pésima: ___ Mala: ___ Regular: ___ Buena: X Excelente: ___

5. ¿Cómo considera que están distribuidos los elementos dentro de la plataforma web?

Pésima: ___ Mala: ___ Regular: X Buena: ___ Excelente: ___

6. En general, ¿Cómo califica la plataforma web?

Pésima: ___ Mala: ___ Regular: X Buena: ___ Excelente: ___

Sugerencias y comentarios finales

- Poca consistencia en la navegación entre interfaces.
- Mejorar el texto posteadado para brindar accesibilidad en las interfaces.
- Mejorar la manera en que se muestran las ventanas modales.
- Mejorar el diseño en las interfaces de navegación.

CONCLUSIONES

En la prueba realizada anteriormente, se generó un promedio en las tablas de efectividad y eficacia del tiempo que les tomo a todos los grupos de usuarios realizar la tarea respecto al tiempo esperado por los desarrolladores. Se establece de esta manera por el grado de similitud en las respuestas de los usuarios de un grupo correspondiente que realizaron la prueba y para reducir la cantidad de hojas en el presente documento.

Según los resultados obtenidos en esta prueba y siguiendo las sugerencias y comentarios de cada uno de los participantes se comprueba que el nivel de satisfacción no es el adecuado y por lo tanto se procede a realizar cambios en el prototipo inicial – segunda iteración. Los cambios son agregados al prototipo final.

5.7 PROTOTIPO FINAL

Una vez concluida la segunda iteración del prototipo inicial y siguiendo cada uno de los resultados obtenidos en la prueba de usabilidad de la plataforma SIGMAT, se hace uso de la metodología adecuada aplicada a este proyecto y se realiza los últimos cambios pertinentes al prototipo inicial, una vez concluida esta etapa se realizara una evaluación heurística de la plataforma y según sus resultados se aplicara o no la quinta (5) y última fase de la metodología.

- **INTERFAZ PRINCIPAL - PROTOTIPO FINAL**

El total de cambios realizados en cada una de las fases planteadas en la metodología, se hizo con el objetivo de cumplir con la mayoría de las necesidades que puede tener un usuario-operador al monitorear un sistema de transporte público en el momento de tomar decisiones acertadas que afecten o impacten el servicio.

Una vez ya establecidas las funcionalidades requeridas para la plataforma web SIGMAT, se dividen en cuatro (4) tipos de monitoreo diferentes:

- Monitoreo en mapa una y múltiples rutas
- Monitoreo en líneas de tiempo - múltiples rutas
- Monitoreo en líneas de tiempo - única ruta
- Monitoreo en tablas - única ruta

Ahora, SIGMAT busca bríndale un aspecto agradable y eficiente en cada una de sus interfaces implementadas y probadas para cumplir y ofrecer un producto de alta calidad.

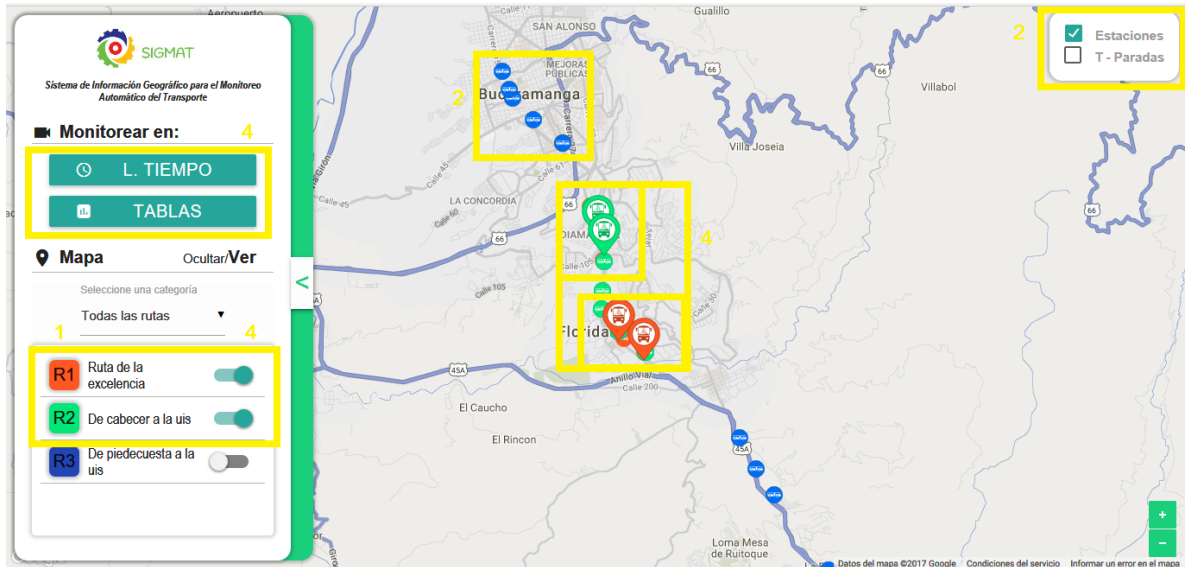
Es por eso que la página principal del sitio web SIGMAT, contara con un panel principal capaz de orientar al usuario-operador a seleccionar algún tipo de

monitoreo de los cuatro (4) que ofrece la plataforma, además contara con un mapa de fondo indicándole intuitivamente al usuario-operador realizar específicamente el tipo de monitoreo en mapa.

Se agregan los siguientes cambios delimitados por color amarillo en sus respectivas figuras:

1. Se agregan diferenciadores en modo **color** en el nombre de cada una de las rutas disponibles en el sistema.
2. Contará con un mini panel ubicado en la parte superior-derecha de la plataforma web SIGMAT, ofreciendo una nueva funcionalidad en la visualización de la información sobre el mapa, permitiendo ver en marcadores diferenciados todas las estaciones y paradas del sistema de transporte que existan en la actualidad.
3. Se elimina la ventana modal la cual sobre salía de la pantalla principal siendo esta una característica principal del prototipo inicial – segunda iteración.
4. El panel contara con botones referenciados a los tipos de monitores restantes: Líneas de tiempo y tablas. Además checkbox de tipo switch para la operación de monitoreo en mapa en una o múltiples rutas seleccionadas.

Figura 13. Prototipo final - interfaz principal

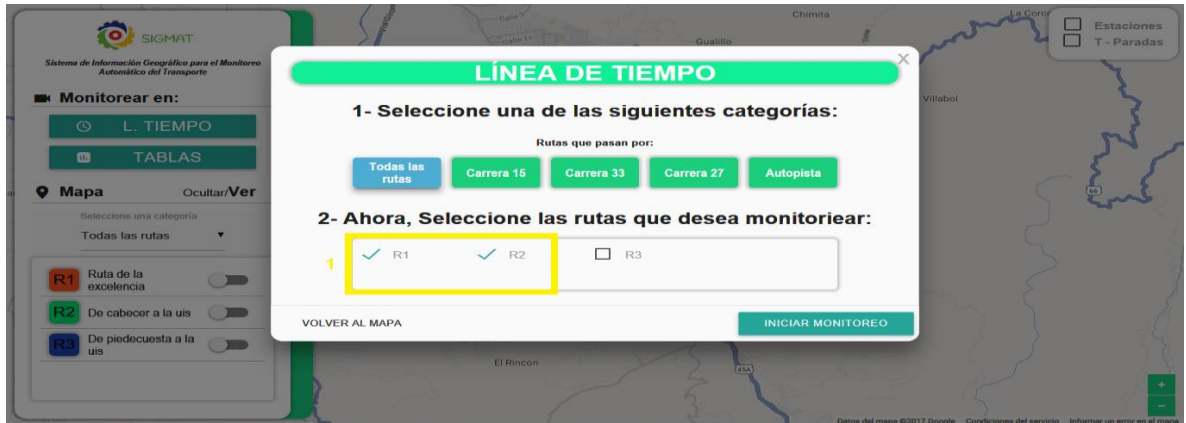


- **VENTANAS DE NAVEGACIÓN - PROTOTIPO FINAL**

Se agregan dos ventanas modal cada una correspondiente a un tipo de monitoreo, línea de tiempo ó tablas, se establecen párrafos generales para representar una idea de la acción a ejecutar.

1- En la modal correspondiente a las líneas de tiempo se establece un tipo de check box de tipo selección, se le otorga la posibilidad al usuario realizar una acción con una o múltiples rutas según su elección.

Figura 14. Prototipo final - interfaz de navegacion - L.Tiempo



2- Por otra parte en la modal de tablas se establece un tipo de check box de tipo radio buttons, llevando a el usuario-operador a seleccónar una unica opcion por ende permitiendo en SIGMAT monitorear una unica ruta.

Figura 15. Prototipo final - interfaz de navegacion - tablas

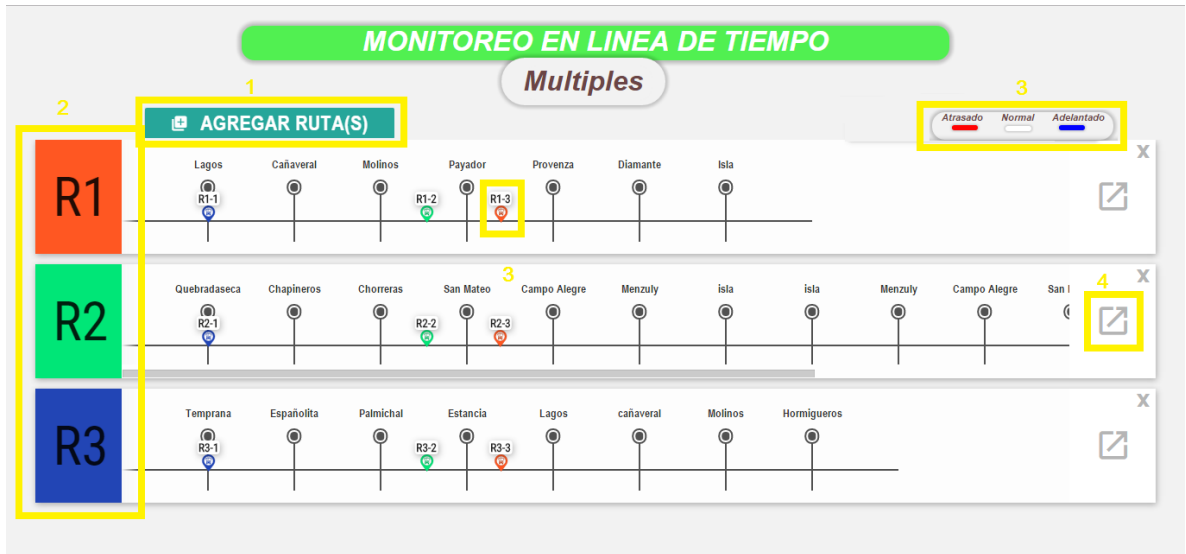


• INTERFAZ LÍNEAS DE TIEMPO - PROTOTIPO FINAL

En esta ocasión y respecto a la interfaz en línea de tiempo del prototipo inicial – segunda iteración se realizan algunos cambios significativos según las recomendaciones y resultados de la prueba de usabilidad realizada en la fase #4.

- Cambios realizados en la interfaz de monitoreo en múltiples rutas:
 1. Inclusión de un botón en el monitoreo de múltiples rutas para agregar una o varias rutas adicionales.
 2. Diferenciadores representados en colores en cada uno de los nombres de las rutas seleccionadas.
 3. Se agregan identificadores de movilidad diferenciados por colores en cada uno de los marcadores que representan los buses totales de cada ruta, estos identificadores se van alterando según el estado total de su recorrido en cada una de las rutas seleccionadas.
 4. Inclusión de un botón, para el re direccionamiento al monitoreo individual de una única ruta seleccionada, garantizando consistencia en la navegación.

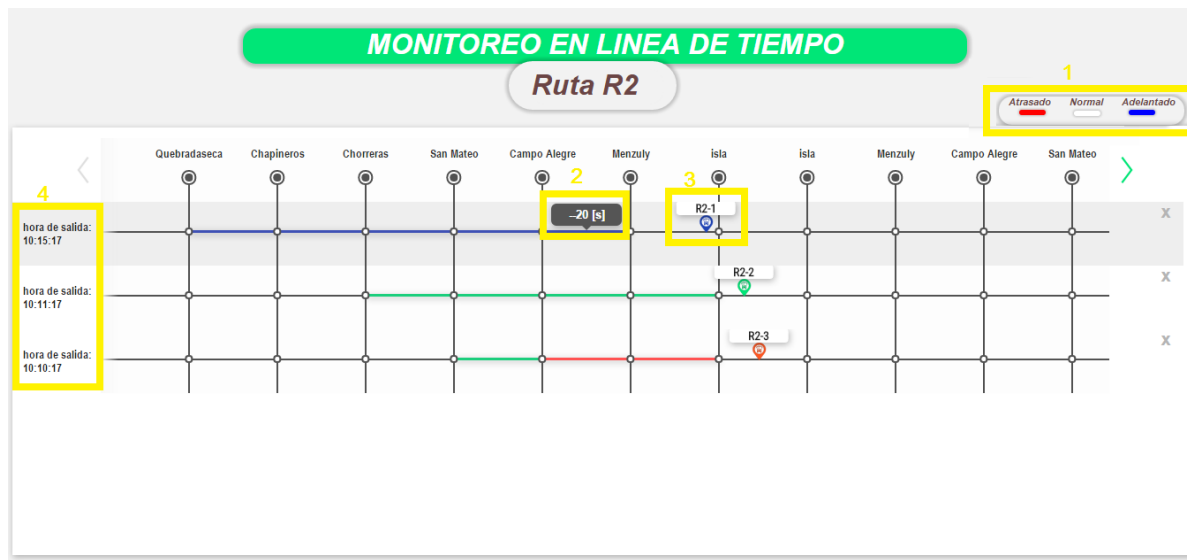
Figura 16. Prototipo final - interfaz multiples rutas - L.Tiempo



- Cambios realizados en la interfaz de monitoreo en única ruta:
 - 1- Se agregan identificadores de movilidad diferenciados por colores en las líneas que conectan a una estación con la otra, estos identificadores se van alterando según el tiempo de recorrido entre estaciones.

- 2- Se agrega información que valida la diferencia de tiempo entre las horas de llegada reales y estimadas entre dos estaciones.
- 3- Se agregan identificadores de movilidad diferenciados por colores en cada uno de los marcadores que representan los buses totales de cada ruta, estos identificadores se van alterando según el estado que obtengan al finalizar su estadía en una estación.
- 4- Se agrega individualmente las horas de salida de cada bus.

Figura 17. Prototipo final - interfaz unica ruta - L.Tiempo



- **INTERFAZ TABLAS - PROTOTIPO FINAL**

Finalizando este último sistema de monitoreo en SIGMAT, se adicionan características que generan un valor agregado especial a este tipo de monitoreo en la visualización de la información en tablas.

SIGMAT busca mostrarle al usuario operador información concreta y pertinente para la toma de decisiones, es por eso que en este último prototipo final se reduce la cantidad de información mostrada en las columnas de cada tabla.

Se evidencio en las pruebas de usabilidad realizadas el exceso de información que contiene el itinerario de una ruta es por eso que se realizan los siguientes cambios, delimitados de color amarillo en la figura 18:

1. Se agregan identificadores de movilidad.
2. Se ocultan las horas de llegada estimadas y reales de cada bus en su respectiva estación.
3. Se agrega la diferencia de tiempo en minutos entre la hora estimada de llegada y hora llegada real de un bus al llegar a una estación, sobre coloreada según su condición.
4. Se agrega funcionalidad dentro de la visualización de la información en tablas capaz de ocultar, mostrar, seleccionar y exportar en pdf, las columnas y filas con la información que el crea pertinente a la hora de tomar decisiones eficaces que influyan en el sistema de transporte.

Figura 18. Prototipo final - interfaz - tablas

MONITOREO EN TABLAS
RUTA R1

1 Atrasado Normal Adelantado

4	Conductor...	Ruta	Estado	Salida...	Est 1	Est 2	Est 3	Est 4	Est 5
✓	Pedro Roa	R1-1	true	0 Min	1 Min	-3 Min	-4 Min	0 Min	0 Min
✓	Andrea Roa	R1-2	true	0 Min	0 Min	0 Min	-3 Min	-2 Min	0 Min
✓	Elkin Roa	R1-3		-2 Min	1 Min	1 Min	0 Min	1 Min	17 Min
✓	Brayan Roa	R1-4		1 Min	1 Min	11 Min	1 Min	0 Min	30 Min
✓	Carlos Roa	R1-5		0 Min	-1 Min	0 Min	-1 Min	1 Min	45 Min
✓	Julian Roa	R1-6		-1 Min	0 Min	0 Min	1 Min	50 Min	0 Min
✓	Camilo Roa	R1-7		0 Min	1 Min	-1 Min	0 Min	5 Min	15 Min
✓	Nelson Roa	R1-8		0 Min	-1 Min	0 Min	-1 Min	20 Min	30 Min

4

Exportar todo como pdf

Exportar vista como pdf

Exportar selección como pdf

Columnas:

- Id Bus
- Conductor
- Ruta
- Estado
- H.S.E
- Salida
- Salida

Artículos Totales: 8 (Artículos Seleccionados: 3)

Figura 19. Prototipo final - interfaz - tablas – pdf

bd0f0562-1eb5-4c25-8030-fed433573405

Ruta:

Bus	Conductor	Ruta	Estado	Salida	Llegada E1	Llegada E2	Llegada E3	Llegada E4	Llegada E5
ZOE101	1000000	R1-1	TRUE	13:00:00	13:09:00	13:23:47	13:29:52	13:35:25	13:45:25
ZOE204	1096484894	R1-4	FALSE	13:44:00	13:54:00	14:04:48	14:09:55	14:20:26	
ZOE207	17777777	R1-7	FALSE	14:30:00	14:39:00	14:51:48	14:55:59		

+

+

-

6. PRUEBAS DE VALIDACIÓN

A continuación se realizarán todas las pruebas necesarias para verificar que el sistema de software realizado cumple con las especificaciones y que logra su cometido final y el buen funcionamiento.

6.1 PRUEBA DEL SISTEMA DE MONITOREO WEB POR MEDIO DE UNA SIMULACIÓN DE UN SISTEMA BRT DE PEQUEÑA ESCALA.

Para esta prueba y en sí para todo el desarrollo y diseño web de la plataforma, se utilizó una herramienta denominada **Prepros** para el desarrollo front-end, esta herramienta permite grandes ayudas en el proceso de creación de proyectos, procesando y compilando lo que se necesite y optimizando y testeando el sitio web.

Prepros fue elegida por su gran variedad de funcionalidades que abarcan desde el desarrollo con lenguajes como JavaScript y CSS, a la optimización de datos. Aligerando el flujo de trabajo.

Funcionalidades utilizadas:

- Un servidor HTTP construido en la propia app, para testeado de archivos que requieran conexiones http sin tener que depender de otros servidores web más complejos.
- Concatenar todos los archivos JS, ahorrando tiempo y facilitando que el sitio tenga menos solicitudes HTTP al servidor.

- Refresco de página automático, algo parecido al conocido "live reload" que recarga sin tener que hacer nada en la página del navegador.

Además se utilizó un editor de texto llamado **sublime text** para poder manipular y editar cada uno de los archivos pertinentes para el correcto funcionamiento de la prueba.

La prueba consistió en probar el funcionamiento de cada uno de las funcionalidades implementadas en este proyecto.

A continuación se muestran los pasos a seguir para validar el funcionamiento en algunas de las interfaces implementadas en SIGMAT.

- Se hicieron peticiones locales, según url estructuradas por el programador.

Figura 20. Estructuración de las peticiones y el archivo json

```
var pnt = function(){
    $http.get("http://localhost:8000/data/Datos a consultar.json")
        .success(function(data) {
            Acciones a ejecutar |
        });
};
//Almacenamos las referencias al elemento, para que así no tengamos que buscarlo en el DOM
function initPnt( pnt, time ){
    return setInterval(pnt, time);
};
//Ejecutamos la función pnt
pnt();
// activamos el intiptnt para realizar peticiones cada 5 segundos
initPnt(pnt, 5000);
```

Se debe tener en cuenta que en este tipo de prueba las url son creadas según la ubicación del archivo.json que se desea consultar dentro de la carpeta establecida por el servidor local.

En una situación real, estas url son creadas por el web server y llevan una estructura previamente planificada.

Para poder visualizar la información extraída del Json en tiempo cuasi real se realizan peticiones cada cinco (5) segundos en esta prueba para ver los resultados casi al instante de esta manera se verifica si se ejecuta la acción como se desea. Se almacenan las referencias al elemento para que no se tengan que buscar en el DOM cada que se ejecute el intervalo y la función así no se carga cada vez en memoria.

En una situación real estas peticiones se hacen en un intervalo de tiempo mucho mayor, este tiempo es establecido previamente con algunos estudios de movilidad.

Para el intercambio de datos se estructuraron archivos .json con datos simulados de posibles itinerarios, recorridos, en si toda la información correspondiente a los buses de un sistema de transporte público.

Para validar que la plataforma web está obteniendo y procesando esos datos en la figura 21 se muestra un ejemplo con la información pertinente consultada.

funcionamiento normal en todos los navegadores o si es compatible con todos los sistemas operativos necesarios.

Las pruebas realizadas a continuación se hicieron con diferentes tipos de navegadores:

- Google Chrome
- Microsoft Edge – Windows 10
- Firefox

En los siguientes sistemas operativos:

- Windows 7,10
- Os X

Y con resoluciones de pantalla diferentes:

- 800*600
- 1024*768
- 1280*800

Cuando se trata de pruebas de compatibilidad del navegador estos son los tres (3) factores principales que pueden afectar la forma como se ve el sitio web en diferentes equipos.

La prueba consistió en evaluar cualitativamente cada una de la interfaces disponibles en la plataforma web SIGMAT.

Si el nivel de compatibilidad es ACEPTABLE, SIGMAT está disponible para la ejecución en ese navegador correspondiente a un sistema operativo y según una resolución.

Si por el contrario el nivel de compatibilidad es INACEPTABLE, SIGMAT no está disponible para la ejecución en ese navegador correspondiente a un sistema operativo y según una resolución.

Nivel de compatibilidad:

- **ACEPTABLE**
- **INACEPTABLE**

Tabla 5. Pruebas de compatibilidad con diferentes navegadores

<u>Navegadores/</u> <u>OS/</u> <u>Resolución</u>	Principal	L.Tiempo Única-Ruta	L.Tiempo Múltiples- Rutas	Tablas
Google Chrome/ Windows 7,10/ 800*600-1024*768- 1280*800	ACEPTABLE	ACEPTABLE	ACEPTABLE	ACEPTABLE
Google Chrome/ Os X/ 800*600-1024*768- 1280*800	ACEPTABLE	ACEPTABLE	ACEPTABLE	ACEPTABLE
Microsoft Edge/ Windows 10/ 800*600-1024*768- 1280*800	ACEPTABLE	ACEPTABLE	ACEPTABLE	ACEPTABLE
Firefox/ Windows 7,10/ 800*600-1024*768- 1280*800	ACEPTABLE	ACEPTABLE	ACEPTABLE	ACEPTABLE
Firefox/ Os X/ 800*600-1024*768- 1280*800	ACEPTABLE	ACEPTABLE	ACEPTABLE	ACEPTABLE

Conclusiones:

Según las pruebas realizadas para determinar la compatibilidad se deduce que la visualización de la información es adecuada y se muestra correctamente en todos los navegadores planteados en la prueba.

Se recomienda utilizar el navegador google Chrome en cualquier tipo de OS y la resolución pertinente por el ordenador, la fluidez en este navegador sobre sale por encima de los demás haciendo de este el más rápido en cada interacción con algún elemento del sitio web.

6.3 EVALUACIÓN HEURÍSTICA

Esta evaluación es realizada por un experto en el área y el objetivo es verificar si las interfaces de usuario siguen los principios de usabilidad establecidos. Según los resultados se define la viabilidad de la plataforma web desarrollada.

A continuación se presenta el informe del análisis heurístico realizado sobre el sitio <http://localhost:8080/docs/api/SIGMAT/index.html> con fecha **27/01/2017**

DATOS DEL ANÁLISIS

Tabla 6. Prueba de usabilidad - datos de análisis

Fecha	27/01/2017	
Tipo de conexión	HTTP	
Plataforma	SIGMAT	
Agente de usuario	Mg. Luis Eduardo Bautista Rojas	Magister en Ing. Sistemas Informática - HCI Diseñador – industrial
Tareas		Mencionadas en el siguiente ítem
Pantallas		Mencionadas en el siguiente ítem

OBJETIVOS

Tabla 7. Prueba de usabilidad - objetivos

Objetivo	Tarea	Pantalla
<p>Verificar si las interfaces de usuario siguen los principios de usabilidad establecidos.</p> <p>Efectividad Eficiencia Satisfacción</p>	<ol style="list-style-type: none"> 1. Localice y habilite la visualización de la Ruta R2 en el Mapa. 2. Localice y habilite la visualización de todas las Estaciones del sistema en el Mapa. 3. Localice el monitoreo de Líneas de tiempo, seleccione la ruta R2, R3 y monitoréelas. 4. Agregue la ruta R1 estando en el paso Anterior 5. Localice y habilite la visualización independiente de la ruta R3, estando en el paso Anterior y monitoréela. 6. Localice el monitoreo en Tablas, seleccione la ruta R2 y monitoréela. 	<ol style="list-style-type: none"> 1. Interfaz principal – tarea 1,2 y 6 2. Navegación entre interfaces – tarea 3 3. Interfaz Línea de tiempo múltiples rutas – tarea 4 4. Navegación interfaz línea de tiempo múltiples rutas e interfaz línea de tiempo-única ruta – tarea 5 5. Interfaz tablas única ruta – tarea 6

MEDICIONES

Las mediciones que dan valor a los heurísticos siguen el siguiente patrón ⁶

Tabla 8. Prueba de usabilidad - mediciones

Valor	Observaciones
1	Se da la mínima expresión del heurístico en las páginas evaluadas
2	Se da una expresión baja del heurístico en las páginas evaluadas
3	Se da una expresión media del heurístico en las páginas evaluadas
4	Se da una expresión alta del heurístico en las páginas evaluadas
5	Se da la máxima expresión del heurístico en las páginas evaluadas

HEURÍSTICOS GENERALES ⁷

Tabla 9. Prueba de usabilidad - generales

Generales	Valor
¿Cuáles son los objetivos del sitio web? ¿Son concretos y bien definidos?	5
¿Los contenidos y servicios que ofrece se corresponden con esos objetivos?	5
¿Tiene una URL correcta, clara y fácil de recordar? ¿Y las URL de sus páginas internas? ¿Son claras y permanentes?	N/A
¿Muestra de forma precisa y completa qué contenidos o servicios ofrece realmente el sitio web?	5
¿La estructura general del sitio web está orientada al usuario?	4
¿El look & feel general se corresponde con los objetivos, características, contenidos y servicios del sitio web?	5
¿Es coherente el diseño general del sitio web?	5

⁶ Cuando el heurístico no sea de aplicación se notará con un espacio en blanco, computando como nulo su valor de tal modo que no afecte al promedio

⁷ Basados en Guía de Evaluación Heurística de Sitios Web (<http://www.nosolousabilidad.com/articulos/heuristica.htm>) de Yusef Hassan Montero

Generales	Valor
¿Es reconocible el diseño general del sitio web?	4
¿El sitio web se actualiza periódicamente?	5
¿Es amigable, familiar y cercano?	4
¿1 párrafo = 1 idea?	N/A
TOTAL	4,6

Tabla 10. Prueba de usabilidad – identidad e información

Identidad e información	Puntos
El logotipo, ¿es significativo, identificable y suficientemente visible?	3
El eslogan o tagline, ¿expresa realmente qué es la empresa y qué servicios ofrece?	4
¿Se ofrece algún enlace con información sobre la empresa, sitio web, 'webmaster'?	2
¿Se proporciona mecanismos para ponerse en contacto con la empresa?	2
TOTAL	2,7

Tabla 11. Prueba de usabilidad – lenguaje y redacción

Lenguaje y redacción	Puntos
¿El sitio web habla el mismo lenguaje que sus usuarios?	4
¿Empieza un lenguaje claro y conciso?	4
TOTAL	4

Tabla 12. Prueba de usabilidad - rotulado

Rotulado	Puntos
Los rótulos, ¿son significativos?	4
¿Usa rótulos estándar?	4
¿Usa un único sistema de organización, bien definido y claro?	5
¿Utiliza un sistema de rotulado controlado y preciso?	4
El título de las páginas, ¿es correcto? ¿ha sido planificado?	4
TOTAL	4,2

Tabla 13. Prueba de usabilidad – estructura y navegación

Estructura y navegación	Puntos
La estructura de organización y navegación, ¿es la más adecuada?	3
En el caso de estructura jerárquica, ¿mantiene un equilibrio entre profundidad y anchura?	5
En el caso de ser puramente hipertextual, ¿están todos los nodos comunicados?	5
¿Los enlaces son fácilmente reconocibles como tales? ¿Su caracterización indica su estado (visitados, activos,...)?	5
En menús de navegación, ¿se ha controlado el número de elementos y de términos por elemento para no producir sobrecarga memorística?	4
¿Es predecible la respuesta del sistema antes de hacer clic sobre el enlace?	5
¿Se ha controlado que no haya enlaces que no lleven a ningún sitio?	5
¿Existen elementos de navegación que orienten al usuario acerca de dónde está y cómo deshacer su navegación?	4
¿Se ha evitado la redundancia de enlaces?	5
¿Se ha controlado que no haya páginas "huérfanas"?	5
¿Es fácilmente reconocible como tal?	5
¿Permite la búsqueda avanzada?	4
¿Muestra los resultados de la búsqueda de forma comprensible para el usuario?	5
¿La caja de texto es lo suficientemente ancha?	N/A
¿Asiste al usuario en caso de no poder ofrecer resultados para una consultada dada?	2
¿Puede el usuario disfrutar de todos los contenidos del sitio web sin necesidad de tener que descargar e instalar plug-ins adicionales?	5
¿Se ha controlado el peso de la página?	5
¿Se puede imprimir la página sin problemas?	N/A
TOTAL	4,5

Tabla 14. Prueba de usabilidad – layout de la página

Layout de la página	Puntos
¿Se aprovechan las zonas de alta jerarquía informativa de la página para contenidos de mayor relevancia?	4
¿Se ha evitado la sobrecarga informativa?	5
¿Es una interfaz limpia, sin ruido visual?	5
¿Existen zonas en "blanco" entre los objetos informativos de la página para poder descansar la vista?	5
¿Se hace un uso correcto del espacio visual de la página?	5
¿Se utiliza correctamente la jerarquía visual para expresar las relaciones del tipo "parte de" entre los elementos de la página?	4
¿Se ha controlado la longitud de página?	5
TOTAL	4,7

Tabla 15. Prueba de usabilidad - búsqueda

Búsqueda (en caso de ser necesaria)	Puntos
¿Se encuentra fácilmente accesible?	5

Tabla 16. Prueba de usabilidad – elementos multimedia

Elementos multimedia	Puntos
¿Las fotografías están bien recortadas? ¿son comprensibles? ¿se ha cuidado su resolución?	N/A
¿Las metáforas visuales son reconocibles y comprensibles por cualquier usuario?	3
¿El uso de imágenes o animaciones proporciona algún tipo de valor añadido?	5
¿Se ha evitado el uso de animaciones cíclicas?	5
TOTAL	4

Tabla 17. Prueba de usabilidad - accesibilidad

Accesibilidad	Puntos
¿El tamaño de fuente se ha definido de forma relativa, o por lo menos, la fuente es lo suficientemente grande como para no dificultar la legibilidad del texto?	4
¿El tipo de fuente, efectos tipográficos, ancho de línea y alineación empleados facilitan la lectura?	5

Accesibilidad	Puntos
¿Existe un alto contraste entre el color de fuente y el fondo?	3
¿Incluyen las imágenes atributos 'alt' que describan su contenido?	4
¿Es compatible el sitio web con los diferentes navegadores? ¿se visualiza correctamente con diferentes resoluciones de pantalla?	4
TOTAL	4

Tabla 18. Prueba de usabilidad – control y retroalimentación

Control y retroalimentación	Puntos
¿Tiene el usuario todo el control sobre el interfaz?	5
¿Se informa constantemente al usuario acerca de lo que está pasando?	4
¿Se informa al usuario de lo que ha pasado?	3
Cuando se produce un error, ¿se informa de forma clara y no alarmista al usuario de lo ocurrido y de cómo solucionar el problema?	3
¿Posee el usuario libertad para actuar?	4
¿Se ha controlado el tiempo de respuesta?	5
TOTAL	4

Conclusiones

A modo de conclusiones podemos ver un mapa de los heurísticos controlados en esta evaluación, agrupados por categorías:

Tabla 19. Prueba de usabilidad – conclusiones

Heurísticos	Resultado	Observaciones
Generales	4,6	Teniendo en cuenta que el contenido mostrado en la web es especializado, se considera que la información suministrada permite la realización de la labor del usuario de manera ordenada. Igualmente, el contenido suministrado al usuario se ha mostrado de manera gráfica, permitiéndole una mayor y más

Heurísticos	Resultado	Observaciones
Identidad e información	2,7	<p>rápida comprensión de la información.</p> <p>Se evidencia que la imagen de marca del producto de software no tiene un uso relevante, identificable y significativo. La imagen por sí sola no permite identificar el tipo de producto y su funcionalidad general.</p> <p>Así mismo no se ofrece una sección de información de ayuda, información con respecto al producto, empresa etc.</p> <p>No se evidencia la existencia de un modo de contacto con la empresa (o proveedor del servicio) para la solicitud de servicio técnico o información adicional.</p>
Lenguaje y redacción	4.0	El producto presente un lenguaje sencillo y claro, sin perder el contexto técnico y el lenguaje conocido por los usuarios finales.
Rotulado	4.2	En la totalidad de la web, se manejan rótulos estandarizados con un tamaño adecuado para su correcta lectura y usando los nombres de las secciones de manera coherente con su funcionalidad.
Estructura y navegación	4.5	<p>En el index se observa una estructura a dos áreas, la primera de control y la segunda de visualización. En la primera área, se debe clasificar mejor las secciones para mejorar la lectura.</p> <p>En la mayoría de secciones, se ha reducido el número de elementos y el contenido, sin embargo, aún permanecen secciones que pueden causar sobre carga cognitiva del usuario.</p>

Heurísticos	Resultado	Observaciones
		No se presentan cuadros de error, ni se evidencia cómo el usuario abordará la presencia de errores.
Layout de la página	4.7	Se observa un aprovechamiento correcto del espacio de trabajo, usando espacios libres para evitar la sobrecarga. Igualmente, se evidencia el uso estructurado de los elementos visuales presentados en cada pantalla.
Búsqueda (en caso de ser necesaria)	5.0	Sin comentarios
Elementos multimedia	4.0	Las metáforas usadas no son reconocibles para cualquier usuario, sin embargo se debe aclarar que el usuario tiene formación especializada en el tema.
Accesibilidad	4.0	<p>Se observa un bajo contraste entre el color de fuente y fondo, consecuencia de estos se puede presentar fatiga visual o confusión en los controles en usuarios con defectos de visión cromáticos.</p> <p>No se incluyen atributos de accesibilidad por teclado (ALT+) para los controles.</p>
Control y retroalimentación	4.0	No se observan mensajes aclarativos e ilustrativos de las acciones realizadas (retroalimentaciones). En la presentación de errores tampoco se comunica la información adecuada para reparar o reanudar la tarea.

PROPUESTAS DE SOLUCIÓN

A continuación se muestran las propuestas de solución para mejorar la usabilidad de la aplicación, siguiendo los resultados obtenidos de la evaluación heurística ⁸.

- Se recomienda mejorar el uso de la imagen del producto de software, buscando posicionamiento de la marca.
- Incluir secciones de ayuda para el usuario
- Incluir secciones de contacto con la empresa y el servicio técnico.
- Elevar el contraste de fondo y fuente para evitar fatiga visual y errores de usuario ocasionados por defectos cromáticos.

Conclusiones por parte de los desarrolladores:

- Según el resultado en la tabla 10. Identidad e información se debe aclarar que este factor no se tuvo en cuenta al momento de desarrollar el prototipo web y por tal el motivo del resultado.

Pero no deja de ser un factor a tener en cuenta para este proyecto o para una posible nueva implementación.

- Según la prueba realizada anteriormente la plataforma web SIGMAT cumple en gran parte su totalidad con los principios de usabilidad planteados al inicio de la prueba y este factor es importante porque se concluye que el prototipo web implementado en este documento es totalmente funcional.

⁸ Se marcan con un prefijo clave o con un color destacado en la celda aquellas propuestas que se consideran críticas para mejorar sensiblemente la aplicación, siendo recomendado que su implementación sea prioritaria.

7. CONCLUSIONES

Finalizado el prototipo web funcional SIGMAT se concluye lo siguiente:

1. Se consideró que el funcionamiento de los sistemas BRT se pueden mejorar ostensiblemente a partir de la adopción de SIGMAT, debido a que proporciona las herramientas necesarias para realizar un monitoreo y control efectivo y eficiente de las rutas, dotando a los encargados del monitoreo de información oportuna, condensada y gráfica, suficiente para mejorar la eficiencia y efectividad en la toma de decisiones por parte del operador, esto se debe gracias a realizar el monitoreo en interfaces graficas diseñadas con la supervisión y acompañamiento de expertos en HCI y UX todo esto desarrollado bajo un ecosistema web compatible con los navegadores más populares.
2. SIGMAT Está construido teniendo en cuenta las necesidades del presente sistema y capacitado para soportar las necesidades que en un futuro sea necesario satisfacer, ofreciendo mejorar ostensiblemente entre un corto a mediano plazo la eficacia y eficiencia del sistema dando soluciones a los problemas que más aquejan actualmente a los usuarios. Se convierte entonces en una opción viable para una posible transición hacia un software más moderno y competente.
3. SIGMAT es un sistema idóneo debido a las funcionalidades complementarias y adicionales que tienen respecto al sistema de monitoreo con el que cuentan actualmente los operadores para la toma de decisiones. Dotando a SIGMAT con una gran viabilidad e importancia.

4. Mediante la investigación, la innovación y consultando con expertos en el área de experiencia de usuario en el funcionamiento del sistema y hablando con un gran número de usuarios, se lograron diseñar soluciones que mejoraran diversos aspectos que se encuentran hoy en día descuidados o ciertas necesidades que no se tuvieron en cuenta en el diseño y posterior construcción del software usado actualmente, esto se hizo con el fin de optimizar y volver más eficaces las frecuencias de los buses dentro del sistema.
5. SIGMAT fue concebido como un software escalable, adaptable y construido de forma modular, pensando siempre en ser un software global que se pueda introducir con facilidad en diferentes ciudades que comparten el mismo sistema de transporte masivo basado en BRT.
6. Las pruebas realizadas fueron un factor importante para validar el buen funcionamiento y el diseño de la plataforma web, después de finalizada la última prueba-evaluación heurística la cual fue realizada por un experto en HCI, se realizaron las últimas modificaciones a SIGMAT siguiendo cada una de las recomendaciones planteadas en el presente documento.

REFERENCIAS BIBLIOGRAFICAS

[1] Estudio EuroTest (Feb/2010). Comparativa de los sistemas de transporte público en 23 ciudades europeas. Recuperado de http://consellsconduir.racc.cat/uploads/download?fichero=/uploads/20120120/comparativa_transporte_publico.pdf

[2] Lincoln Institute of land policy (2010). Sistemas de transporte público masivo tipo BRT (Bus Rapid Transit) y desarrollo urbano en América Latina (Land Lines Article). Recuperado de https://www.lincolninst.edu/pubs/2210_Sistemas-de-transporte-p%C3%BAblico-masivo-tipo-BRT--Bus-Rapid-Transit--y-desarrollo-urbano-en-Am%C3%A9rica-Latina

[3] Blog De Ciudades emergentes(2014). Contribuciones a la Movilidad de Latinoamérica para el Mundo: Sistemas BRT. Recuperado de <http://blogs.iadb.org/ciudadessostenibles/2014/09/09/sistemas-brt/>

[4] Tecnocarreteras(11-abr-2011). Que son los sistemas inteligentes de transporte(ITS). Recuperado de <https://www.tecnocarreteras.es/2011/04/11/que-son-los-sistemas-inteligentes-de-transporte-its/>

[5]Confederación De Empresarios De Andalucía (2010). Sistemas de Información Geográfica tipos y aplicaciones empresariales. Recuperado de <http://sig.cea.es/SIG>

[5.1] wikis paces – ergonómica y cibernética 2011. Recuperado de <https://ergonomia-y-cibernetica-enero-2011-unexpo.wikispaces.com/Cap%C3%ADtulo+2+Sistemas+Hombre-M%C3%A1quina>

[6] Programación Web(Instituto Tecnológico de Matehuala). 2.1 Arquitectura de las aplicaciones web. Recuperado de <https://programacionwebisc.wordpress.com/2-1-arquitectura-de-las-aplicaciones-web/>

[6.1] Programación Web(Instituto Tecnológico de Matehuala). 1.3 introduccion a HTML. Lenguaje de despliegue del web. Recuperado de <https://programacionwebisc.wordpress.com/1-3-introduccion-al-html-lenguaje-de-despliegue-del-web/>

[6.2] Programación Web(Instituto Tecnológico de Matehuala). 4.1 Lenguaje script del cliente.-javascript. Recuperado de <https://programacionwebisc.wordpress.com/4-1-lenguaje-script-del-cliente-javascript/>

[6.3] Desarrollado por W.CSS. Recuperado de http://www.w3schools.com/js/js_json_intro.asp

[7] Mario Saffirio (Tecnologías de Informacion y Gestion de Procesos de Negocios (BPM)). Los Web Services. Recuperado de <https://msaffirio.wordpress.com/2006/02/05/%C2%BFque-son-los-web-services/>

[8] PUSHER (2016). What are Websockets. Recuperado de <https://pusher.com/websockets>

[9] Eumed.net enciclopedia virtual – José Luis Cendejas Valdéz (CV). Recuperado de <http://www.eumed.net/tesis-doctorales/2014/jlcv/software.htm>

[10] Desarrolloweb.com – Ajax con Angular para acceso a API – 03/11/14. Recuperado de <http://www.desarrolloweb.com/articulos/ajax-angularjs-acceso-apirest.html>

[11] JQuery – Introducción – Desarrollado por W3ii.com. Recuperado de http://www.w3ii.com/es/jquery/jquery_intro.html

[12] vBote Blog– Martes 2/12/14. Recuperado de <http://www.vbote.com/vbote-solutions-academy-blog/130-materialize-el-framework-con-principios-de-material-design.html>

BIBLIOGRAFÍA

BLOG DE CIUDADES EMERGENTES Contribuciones a la Movilidad de Latinoamérica para el Mundo: Sistemas BRT. [en línea] disponible en: <http://blogs.iadb.org/ciudadessostenibles/2014/09/09/sistemas-brt/>

CENDEJAS VALDÉZ José Luis Implementación del modelo integral colaborativo (mdsic) como fuente de innovación para el desarrollo ágil de software en las empresas de la zona centro - occidente en México [en línea] disponible en: <http://www.eumed.net/tesis-doctorales/2014/jlcv/software.htm>

CONFEDERACIÓN DE EMPRESARIOS DE ANDALUCÍA Sistemas de Información Geográfica tipos y aplicaciones empresariales. [en línea] disponible en: <http://sig.cea.es/SIG>

Desarrolloweb Ajax con Angular para acceso a API – [en línea] disponible en: <http://www.desarrolloweb.com/articulos/ajax-angularjs-acceso-apirest.html>

ESTUDIO EUROTTEST Comparativa de los sistemas de transporte público en 23 ciudades europeas. [en línea] disponible en: http://consellsconduir.racc.cat/uploads/download?fichero=/uploads/20120120/comparativa_transporte_publico.pdf

JQUERY Introducción Desarrollado. [en línea] disponible en: http://www.w3ii.com/es/jquery/jquery_intro.html

LINCOLN INSTITUTE OF LAND POLICY. Sistemas de transporte público masivo tipo BRT (Bus Rapid Transit) y desarrollo urbano en América Latina (Land Lines

Article). [en línea] disponible en:
https://www.lincolninst.edu/pubs/2210_Sistemas-de-transporte-p%C3%BAblico-masivo-tipo-BRT--Bus-Rapid-Transit--y-desarrollo-urbano-en-Am%C3%A9rica-Latina

PROGRAMACIÓN WEB Instituto Tecnológico de Matehuala. 1.3 introduccion a HTML. Lenguaje de despliegue del web. [en línea] disponible en:
<https://programacionwebisc.wordpress.com/1-3-introduccion-al-html-lenguaje-de-despliegue-del-web/>

PROGRAMACIÓN WEB Instituto Tecnológico de Matehuala. 2.1 Arquitectura de las aplicaciones web. [en línea] disponible en:
<https://programacionwebisc.wordpress.com/2-1-arquitectura-de-las-aplicaciones-web/>

PROGRAMACIÓN WEB Instituto Tecnológico de Matehuala. 4.1 Lenguaje script del cliente.-javascript. [en línea] disponible en:
<https://programacionwebisc.wordpress.com/4-1-lenguaje-script-del-cliente-javascript/>

PUSHER What are Websockets. [en línea] disponible en:
<https://pusher.com/websockets>

SAFFIRIO Mario Tecnologias de Informacion y Gestion de Procesos de Negocios (BPM). [en línea] disponible en:
<https://msaffirio.wordpress.com/2006/02/05/%C2%BFque-son-los-web-services/>

TECNOCARRETERAS Que son los sistemas inteligentes de transporte(ITS). [en línea] disponible en:
<https://www.tecnocarreteras.es/2011/04/11/que-son-los-sistemas-inteligentes-de-transporte-its/>

VBOTE BLOG Materialize: el framework con principios de Material Design [en línea] disponible en: <http://www.vbote.com/vbote-solutions-academy-blog/130-materialize-el-framework-con-principios-de-material-design.html>

W.CSS. JSON – Introduction [en línea] disponible en: http://www.w3schools.com/js/js_json_intro.asp

WIKIS PACES ergonómica y cibernética [en línea] disponible en: <https://ergonomia-y-cibernetica-enero-2011-unexpo.wikispaces.com/Cap%C3%ADtulo+2+Sistemas+Hombre-M%C3%A1quina>