



**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**

---

**FACULTAD DE INGENIERÍA**

# **Mejora y Automatización de Procesos en Telefónica México**

**INFORME DE ACTIVIDADES PROFESIONALES**

Que para obtener el título de  
**Ingeniero en Computación**

**P R E S E N T A**

Carlos Alberto Chávez Hernández

**ASESOR DE INFORME**

M. I. Aurelio Sánchez Vaca



Ciudad Universitaria, Cd. Mx., 2017



*A mis hermanas, porque quiero transmitirles que con dedicación, empeño y esfuerzo  
las posibilidades se transforman en realidades.*  
**Carlos Alberto Chávez Hernández**



# Agradecimientos

---

A mis padres, por su apoyo incondicional y su credibilidad en mí, por educarme de la mejor manera y por cada uno de sus sacrificios que llevaban por delante el soporte de mis metas.

A mis amigos, por acompañarme en el camino no sólo del conocimiento sino de la vida misma, por compartir momentos inolvidables que hicieron el trayecto más ameno y que constituyen un pilar por demás importante en este logro, en especial a Vero, gracias por tanto.

A la Facultad de Ingeniería y a la Universidad, por todo lo que he aprendido.

**Carlos Alberto Chávez Hernández**



# Índice general

---

<b>Índice de figuras</b>	<b>VII</b>
<b>Índice de tablas</b>	<b>IX</b>
<b>Introducción</b>	<b>1</b>
<b>1. La Empresa</b>	<b>3</b>
1.1. Historia de Telefónica en México . . . . .	3
1.2. Descripción del puesto de trabajo . . . . .	5
<b>2. Antecedentes</b>	<b>7</b>
2.1. La importancia del análisis de información . . . . .	7
2.2. Indicadores de rendimiento . . . . .	8
2.3. Sistemas virtuales . . . . .	9
2.4. Definición del problema . . . . .	10
<b>3. Análisis de los requerimientos</b>	<b>11</b>
3.1. Requerimientos del sistema . . . . .	11
3.2. Requerimientos de hardware . . . . .	11
3.2.1. Cliente . . . . .	11
3.2.2. Servidor . . . . .	12
3.3. Requerimientos de software . . . . .	12
3.3.1. Cliente . . . . .	12
3.3.2. Servidor . . . . .	12
3.4. Entorno de desarrollo . . . . .	13
3.4.1. Servicios de internet . . . . .	13
<b>4. Tecnologías a utilizar</b>	<b>15</b>
4.1. Tecnologías WEB . . . . .	15
4.1.1. PHP . . . . .	17
4.1.2. Bootstrap . . . . .	17
4.1.3. MySQL Workbench . . . . .	17
4.1.4. DataTables . . . . .	18

## ÍNDICE GENERAL

---

4.1.5. AWK . . . . .	19
4.1.6. Highcharts . . . . .	19
<b>5. Proceso de automatización de la información</b>	<b>21</b>
5.1. Scripts necesarios . . . . .	21
5.2. Codificación . . . . .	23
5.3. Generación reportes . . . . .	23
5.4. Diseño de la interfaz de usuario . . . . .	24
5.4.1. Flujo de pantallas . . . . .	24
5.5. Análisis de información gráfica . . . . .	32
<b>6. Resultados</b>	<b>33</b>
<b>Conclusiones</b>	<b>35</b>
<b>Bibliografía</b>	<b>37</b>
<b>A. Código</b>	<b>39</b>
A.1. PHP . . . . .	39
A.1.1. header.php . . . . .	39
A.1.2. index.php . . . . .	41
A.1.3. footer.php . . . . .	43
A.1.4. capacidadMPS.php . . . . .	43
A.1.5. ocupacionDSR.php . . . . .	45
A.1.6. detalleOcupacion.php . . . . .	46
A.2. CSS . . . . .	48
A.2.1. estilos.php . . . . .	48
A.3. AWK . . . . .	49
A.3.1. process_data.js . . . . .	49
A.4. JavaScript . . . . .	50
A.4.1. graficas.js . . . . .	50
A.4.2. tablas.js . . . . .	54
<b>B. Glosario</b>	<b>55</b>



# Índice de figuras

---

1.1. Ingresos y accesos de Telefónica México . . . . .	3
1.2. Marcas comerciales de Telefónica . . . . .	4
4.1. Tecnologías WEB . . . . .	16
4.2. DataTables . . . . .	18
4.3. Highcharts . . . . .	19
5.1. Diagrama de flujo . . . . .	22
5.2. Ejemplo de tabla . . . . .	23
5.3. Home . . . . .	25
5.4. Cuadro de Mando del DSR . . . . .	26
5.5. Ocupacion de interfaces . . . . .	27
5.6. Capacidad de MPS . . . . .	28
5.7. RAM . . . . .	29
5.8. Carga de CPUs . . . . .	30
5.9. Answer Timeout . . . . .	31
5.10. Balanceo de servidores . . . . .	32



# Índice de tablas

---

2.1. Ejemplos de KPIs . . . . .	8
2.2. Pros y contras de un sistema virtual. . . . .	9



# Introducción

---

Este trabajo tiene como finalidad ofrecer una descripción general de las actividades realizadas durante un periodo de 6 meses en la empresa Telefónica México, mis actividades comprendían desde el desarrollo de scripts mediante acceso remoto a equipos distantes, creación de reportes y monitoreo de la red, hasta desarrollos web para facilitar la interpretación de datos y la toma de decisiones, en el contenido de este reporte nos centraremos específicamente en el último punto, los sistemas web desarrollados, así como las experiencias profesionales que me proporcionó la empresa al estar trabajando en la posición de Ingeniero de Desempeño de Red en un proyecto de mejora y automatización en los procesos del área de Calidad y Desempeño.

Los datos aquí presentados son una simulación para ejemplificar el funcionamiento del sistema, y no corresponden a información real de la empresa.

## Objetivo

La función principal del área de Calidad y Desempeño es analizar e informar sobre el comportamiento de la red a partir de reportes estadísticos, los cuales se obtienen de los equipos que la conforman.

Uno de los problemas a los que se enfrenta el área es la interpretación de los estadísticos, ya que estos suelen ser archivos de log con mucha información, pero en formato de texto plano. El proceso utilizado para darle sentido a los datos es conocido como ETL, siglas en inglés que se refieren a Extracción, Transformación y Carga. Dicho proceso es aplicado tanto en forma manual como automática, siendo esta última, la manera óptima para realizar el procesado de los datos, ya que se realiza de manera prácticamente instantánea y transparente, lo que permite un análisis y toma de decisiones en un periodo de tiempo mas breve y casi sin intervención humana. Esto se traduce en la solución más ágil de incidencias y la previsión a corto plazo del comportamiento que tendrán los usuarios, la presentación de la información se realiza a través de tableros o cuadros de mando, los cuales son accesibles por las áreas de operaciones y calidad.

## INTRODUCCIÓN

---

La mejora en el proceso ETL planteada consiste en la recolección de los estadísticos directamente del servidor en que son generados y la posterior limpieza y transformación de manera automática para su carga a bases de datos, desde las cuales se alimentan los dashboards para representar la información de manera gráfica, al automatizar el flujo de la información también se mejora la distribución de la misma, ya que los reportes cambian de hojas de calculo en Excel a una página web dentro de la intranet empresarial, eliminando el tiempo de creación y envío de los reportes.

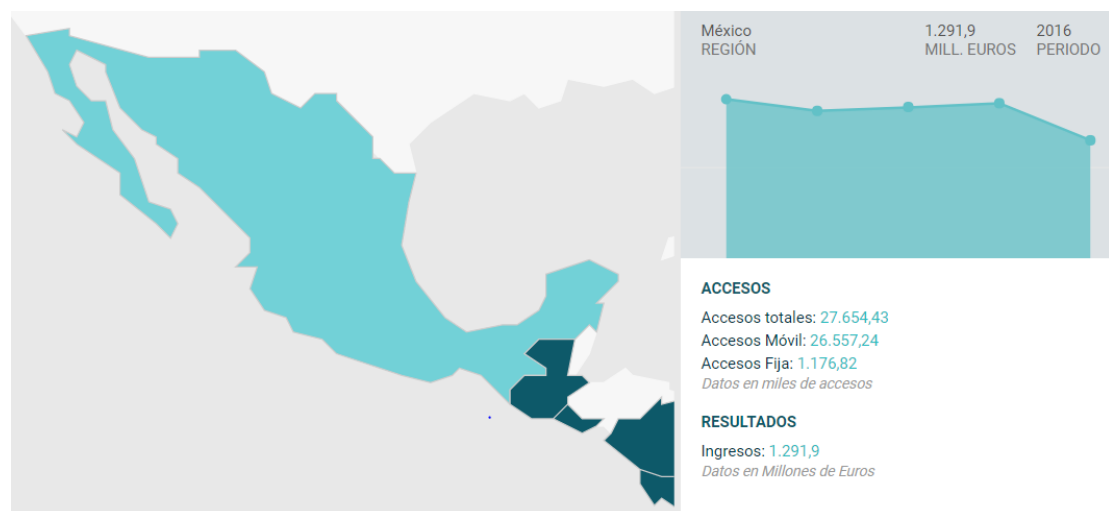
La periodicidad de actualización depende de la información a ser monitoreada y varia el tiempo de refresco de los tableros, pero en general se cuenta con información reciente con un atraso de máximo una hora.

# La Empresa

---

## 1.1. Historia de Telefónica en México

Telefónica México, es conocida legalmente como Pegaso PCS, S.A. de C.V., es una operadora de telefonía móvil en México, subsidiaria del Grupo Telefónica. En México, Movistar es la segunda compañía en cuanto a número de de usuarios y cobertura, superada únicamente por Telcel, es también la tercera en ingresos, después de Telcel y AT&T. Estadísticamente hasta el 30 de Junio de 2017, se tienen registrados más de 27.6 millones de usuarios activos en el país, de los cuales: 26.5 millones de los accesos son móviles y más de 1.1 millones de accesos de telefonía fija [Telefónica, 2017b].



**Figura 1.1:** Ingresos y accesos de Telefónica México

Telefónica es una de las Operadoras Móviles mas grandes en Europa y Latino América contando con 15 años de prescencia en México.

## 1. LA EMPRESA

---

Grupo Telefónica Móviles adquirió presencia en México, en el año 2000 al adquirir las cuatro operadoras ubicadas en el Norte del país: Movitel, BajaCel, Cedetel y Norcel. Para el año 2002 Telefónica se consolida a nivel nacional después de adquirir a Pegaso PCS. Durante 2002 y 2003 se efectúa la fusión de dichas empresas, consolidándose como Telefónica México.

Misión:

”Queremos crear, proteger e impulsar las conexiones de la vida para que las personas elijan un mundo de posibilidades infinitas. En Telefónica creemos que la tecnología forma parte de la vida de todos. Hoy, la conectividad no sirve sólo para relacionarnos, es esencial para nuestra vida personal y laboral. Nuestro papel es facilitarte el disfrute de la conexión, salvaguardando el uso de tus datos y dándote el control de tu vida digital. Queremos ser una ONLIFE TELCO, una telco para las personas, una telco para tu vida.”

Compromisos:

- Una conexión que pone a las personas por delante de la tecnología e impulsa la capacidad de elección en un mundo lleno de opciones.
- Una conexión que ofrece protección y confianza para que las personas se sientan soberanas de su vida digital.
- La conexión como un ecosistema de vida: piensa en las personas y optimiza la vida.
- Una conexión que ayuda a disfrutar de un mundo de posibilidades.

Actualmente Grupo Telefónica cuenta con presencia en 21 países y un promedio de 127.000 empleados, las marcas con las que es conocida la empresa a nivel internacional son: Telefónica, Movistar, Vivo y O2 [Telefónica, 2017a].



**Figura 1.2:** Marcas comerciales de Telefónica



## 1.2. Descripción del puesto de trabajo

La misión del Ingeniero de Desempeño de Red dentro de Telefónica México es obtener KPIs del comportamiento del Core de Voz en la red de Telefónica, de los deferentes nodos y OSS, para detectar desviaciones en su comportamiento de acuerdo a umbrales establecidos y de esta forma, notificar a las áreas correspondientes para que realicen acciones correctivas y evitar una afectación al servicio o una posible afectación. Las funciones implícitas en el desempeño del puesto son:

- Analizar e interpretar datos obtenidos para determinar su posible afectación al servicio.
- Almacenar los datos para examinar su comportamiento a lo largo del tiempo.
- Comunicar las desviaciones encontradas para su análisis y corrección por las diferentes áreas (Ingeniería, Soporte, Operaciones).
- Obtener KPIs para los diferentes nodos que se integran a la red de Telefónica.
- Verificar que los resultados de los KPIs sean consistentes con el comportamiento de los servicios en los nodos.
- Seguir las acciones correctivas realizadas por las áreas y verificar si estas fueron eficaces enviando nuevamente el análisis del KPI.
- Estimar de acuerdo a comportamiento histórico posible degradación del servicio con gráficas y tablas.

La principal naturaleza de los desafíos es establecer y agrupar KPIs que muestren de una manera más oportuna desviaciones en el comportamiento de la red. Para efectuar la toma de decisiones se debe notificar la afectación de servicio a las áreas correspondientes, o se deben escalar en jerarquía en caso de no obtener respuesta a una afectación.

Las relaciones que establece el Ingeniero de Desempeño de Red son:

- Ingeniería, establecer y validar las formulas para los indicadores de desempeño.
- Operaciones, validar indicadores de desempeño y seguimiento a desviaciones de comportamiento.
- Soporte, validar indicadores de desempeño y seguimiento a desviaciones de comportamiento.
- Proveedores, apoyo para documentación y nuevos indicadores de desempeño.



# Antecedentes

---

## 2.1. La importancia del análisis de información

La interpretación de información es esencial hoy en día haciendo énfasis en la cantidad tan abrumadora de datos que recibimos. Tratándose de una empresa que tiene como uno de sus principales objetivos generar ganancias resulta de suma importancia la correcta interpretación de tales datos, como pueden ser el comportamiento de sus clientes, la relación de inversiones contra ganancias o el seguimiento de sus operaciones para poder evaluar el desempeño de sus trabajadores, [Telefónica, 2011].

Al hablar de datos producidos en un ambiente real y de manera digital por una empresa que tiene alcance en gran parte de un país, la cantidad de información se vuelve complicada de interpretar de manera correcta por lo que se han desarrollado procesos que transforman y simplifican las grandes tablas producidas por las computadoras en información más concreta.

El proceso ETL es bastante utilizado gracias a las tecnologías de generación de gráficas que se alimentan de bases de datos. Consiste en la extracción de los datos desde un archivo de log generado directamente en el servidor que se quiere monitorear, seguido de la transformación de esos datos de manera que resulte posible su carga en una base de datos la cual se mantendrá actualizada cada cierto intervalo de tiempo para alimentar las gráficas en un portal web, [Kimball, 2004].

Un análisis realizado de manera correcta beneficia a la empresa al proporcionar información que indica los puntos a mejorar o reforzar. Si bien este proceso ya existía dentro del área, se realizaba de manera poco eficiente, al requerir de la participación humana en un proceso que debe ser automatizado, ya se contaba con algunos desarrollos funcionales aunque realizados con prácticas no recomendadas hoy en día, como distribuir los elementos de una página con tablas, el uso de macros para transformar la información de varios archivos en uno solo, la repetición constante de código en ca-

da uno de los archivos existentes o la inclusión de funciones, estilos y estructura en el mismo archivo, dificultando su lectura, el desarrollo de este proyecto incorpora al ambiente general del área nuevas y mejores practicas a seguir para futuros desarrollos, esto mediante el uso de plantillas para modularizar los desarrollos, manteniendo los elementos encapsulados, facilitando así el mantenimiento y la escalabilidad del sistema, al mismo tiempo se separa la funcionalidad, la conexión a la base de datos, el estilo visual y la estructura.

### 2.2. Indicadores de rendimiento

Para convertir los datos recibidos en información útil se requiere crear medidores de desempeño, los conocidos como KPIs por sus siglas en inglés (Key Performance Indicator), los cuales se relacionan con un objetivo a medir. Un indicador correctamente definido debe ser útil, de fácil comprensión y debe de contar con un periodo de actualización constante, esto con la finalidad de mostrar los cambios de comportamiento en modelos gráficos.

Para poder definir correctamente los KPIs es necesario tener predefinido un proceso de negocio y delimitar el rango de respuestas con umbrales que faciliten el seguimiento y detección de anomalías y al mismo tiempo que permitan verificar si el KPI tiene el comportamiento esperado, es decir, si el elemento a medir se desempeña de manera correcta.

La tabla (2.1) muestra distintos ejemplos de KPIs.

<b>Indicadores de rendimiento</b>	
<b>Categoría</b>	<b>Ejemplo</b>
De calidad	Eficiencia, equipos con más fallos.
De servicio	Tiempo en responder llamadas, cantidad de tráfico en la red.
De cliente	Índice de satisfacción, horas pico de uso de la red.
Económicos	Ingresos, egresos.
Financieros	Rentabilidad, solvencia, liquidez.

**Tabla 2.1:** Ejemplos de KPIs

## 2.3. Sistemas virtuales

Las TIC han ayudado a la rápida evolución de los sistemas económicos y empresariales, al facilitar el manejo de grandes cantidades de información, esto mismo ha causado que la competencia sea más agresiva que hace unas décadas, los sistemas virtuales apoyados en estas TICs son ampliamente buscados por las empresas con la finalidad de mantenerse a la vanguardia y tomar ventaja de sus competidores, el fin de un proceso ETL es presentar los resultados en un sistema virtual [Kenneth, 2005], estos sistemas cuentan con ventajas y desventajas que son listadas a continuación:

La tabla (2.2) muestra los pros y contras de un sistema virtual.

Sistemas virtuales	
Pros	Contras
Ahorro de tiempo en la interpretación de información.	Dependen del correcto funcionamiento de bases de datos y servidores.
Generación de reportes estadísticos más ágil.	La información puede sufrir desfases y ser malinterpretada.
Mayor alcance de la información.	Se requiere una conexión para acceder al sistema.
Posibilidad de acceder a la información de manera remota.	Es necesario un dispositivo con la capacidad de desplegar el sistema.
Menor tiempo de actualización.	Posible pérdida o robo de información.

**Tabla 2.2:** Pros y contras de un sistema virtual.

La correcta denominación de un sistema de monitoreo es DSS, y se encuentran orientados a ejecutivos de escala intermedia en una empresa, su principal objetivo es poder realizar un análisis rápido sin recurrir a herramientas de escritorio como pueden ser hojas de cálculo, [Carlos, 2010].

La fuente de alimentación de datos para un sistema DSS es el histórico de información que posee la empresa, con el cual se generan estimaciones a corto y mediano plazo, generalmente los resultados se presentan en un panel de datos que contiene la información en forma gráfica con la posibilidad de una vista a detalle, la agrupación de múltiples paneles de datos es conocida como cuadro de mando.

El Cuadro de Mando es una herramienta que contiene una agrupación de KPIs, diseñados específicamente para tomar decisiones rápidamente por los responsables de una empresa o departamento, por lo tanto es indispensable que la información se actualice de manera constante, para asegurar que se cuenta con estadísticas recientes.

### 2.4. Definición del problema

Al llegar a la empresa se me asignó como ocupación el monitoreo de un elemento de la red LTE manejada actualmente, dicho elemento es conocido como DSR, el cual es un intermediario entre el resto de dispositivos de red [O'reilly, 1994], su principal función es conectar subredes y encaminar los paquetes de datos correspondientes a cada subred.

El monitoreo se realizaba con un proceso por demás largo y lento, El primer paso necesario consistía en la recolección de los archivos de log directamente de los servidores de manera manual a través de programas como Filezilla, el cual permite la transferencia de archivos de manera remota mediante el protocolo FTP. Seguido de la limpieza de dichos archivos en una tabla de Excel, nuevamente de manera manual, la limpieza consistía en la separación de columnas, selección de los datos útiles, cambio de formato de datos en columnas, sumatorias y promedios. Una vez con que la información se tenía en un estado más puro, se procedía a la creación de gráficas con Excel, y por último su análisis. Dependiendo del resultado obtenido en la etapa de análisis se procedía a el envío de las gráficas, tablas y conclusiones a las áreas correspondientes.

El proceso se realiza de manera correcta aunque poco práctica, en situaciones de emergencia, resulta lento para encontrar la incidencia en cuestión.

La propuesta de mejora tiene como principales argumentos la agilización del proceso, la automatización del mismo y la facilidad de difusión de la información obtenida. Se plantea además la creación de una estructura homogénea para adaptar el resto de desarrollos existentes mediante el uso de plantillas de fácil implementación, y un home de llegada como página principal ya que a la fecha solo se contaba con los enlaces a las distintas páginas y dependía de cada persona el tenerlos a la mano.

Todo esto con el uso de tecnologías accesibles y de libre uso para evitar costos en el desarrollo, siguiendo las metodologías y buenas prácticas aprendidas durante mis años de estudio en la Facultad.

# Análisis de los requerimientos

---

## 3.1. Requerimientos del sistema

Los principales requerimientos planteados para el sistema se listan a continuación:

- Acceso solo desde equipos con conexión a la intranet empresarial.
- Generación de reportes en formato de Excel.
- Muestreo de información acumulada por hora con actualización cada 5 minutos.
- Página inicial con la información más importante a la vista y páginas secundarias con información más detallada.
- Utilización de tecnologías que no impliquen un costo por parte de la empresa.

## 3.2. Requerimientos de hardware

### 3.2.1. Cliente

Los requisitos mínimos necesarios para el correcto funcionamiento del sistema son:

- RAM 1 GB.
- Procesador Dual Core.
- Tarjeta de Red.

### 3. ANÁLISIS DE LOS REQUERIMIENTOS

---

#### 3.2.2. Servidor

Del lado del servidor los requisitos mínimos son:

- RAM 1 GB.
- Procesador Dual Core.
- Tarjeta de Red.

Necesarios para mantener el sistema alojado y disponible.

### 3.3. Requerimientos de software

#### 3.3.1. Cliente

En un equipo cliente bastara con tener instalado:

- Navegador de internet.

Las mejores opciones para el correcto funcionamiento son Mozilla Firefox o Google Chrome, en caso de Internet Explorer se recomienda la versión más actualizada aunque no es necesario.

#### 3.3.2. Servidor

En el lado del servidor se requiere de:

- Servidor Apache.
- PHP 5.4 ó superior.
- MySQL.
- AWK.
- CRON.

Para mantener el sistema alojado y disponible.



## 3.4. Entorno de desarrollo

Para desarrollo del sistema se utilizo el siguiente ambiente de trabajo, los programas utilizados fueron:

- Servidor Apache.
- PHP 5.4.
- MySQL Workbench.
- Sublime Text.
- Goggle Chrome.

### 3.4.1. Servicios de internet

Algunos de los servicios disponibles de internet son:

- Correo electrónico.
- FTP.
- Mensajería instantanea.

El principal servicio a utilizar en este desarrollo es la transferencia de archivos por FTP.



# Tecnologías a utilizar

---

## 4.1. Tecnologías WEB

Los desarrollos web modernos suelen simplificarse al separar sus componentes en partes más fundamentales, las cuales cimientan sus bases en tres lenguajes, HTML, CSS y Javascript.

HTML tiene como propósito establecer la estructura general de la página haciendo uso de etiquetas y atributos, las cuales especifican los elementos que contendrá la pantalla así como la distribución que estos tendrán.

CSS es un lenguaje complementario a HTML para reducir sus limitaciones, la principal característica de CSS es la capacidad que le brinda a un proyecto para separar el contenido del formato, en versiones anteriores de HTML cada componente se estiliza dentro de su etiqueta, haciendo el mantenimiento o actualización un proceso largo y repetitivo, con la evolución de la tecnología las capacidades de HTML se volvieron insuficientes para desarrolladores y diseñadores por igual, la respuesta a esta necesidad es CSS, permitiendo dar formato al estilo de múltiples elementos con un esfuerzo mucho menor ya que permite agruparlos en clases, además no se requiere modificar la estructura ya establecida en los elementos.

Por otro lado Javascript provee de funcionalidad al desarrollo, es un lenguaje interpretado que también se considera complementario a HTML. Una de las ventajas de Javascript es su velocidad de ejecución, ya que resulta similar a la que se puede encontrar en aplicaciones de escritorio. Al ser un lenguaje amplio, potente y estandarizado facilita la creación de aplicaciones web muy útiles, sin mencionar la gran cantidad de librerías desarrolladas por terceros.

La posibilidad de mantener separados tanto la estructura, el estilo y la funcionalidad es lo que convierte a estas tres tecnologías en las predilectas para el desarrollo de proyectos web de carácter más dinámico.

#### 4. TECNOLOGÍAS A UTILIZAR

---



**Figura 4.1:** Tecnologías WEB

Para este proyecto se empleó la versión más reciente de HTML, HTML5, que representa un cambio importante respecto a sus versiones previas al difuminar la barrera existente entre aplicaciones de escritorio y aplicaciones móviles, dando versatilidad a un mismo desarrollo [Diego, 2012].

Algunas de las ventajas de utilizar HTML5 son:

- Agilizar el desarrollo y el mantenimiento del proyecto al simplificar y reducir las etiquetas del lenguaje.
- Evitar múltiples desarrollos al poder adaptar un desarrollo móvil a uno de escritorio o viceversa.
- Permite crear elementos estáticos y dinámicos en la estructura.

En cuanto al formato se utilizó CSS3, versión con características nuevas que mantienen el proyecto actualizado y escalable por la facilidad que representa el modificar los elementos de manera conjunta o individual.

Algunas de las ventajas de utilizar CSS3 son:

- Capacidad de definir un estilo diferente o adaptativo para cada tipo de pantalla.
- Páginas con mayor atractivo y dinamismo.
- Elimina la necesidad de recurrir a tecnologías externas.

La funcionalidad dinámica del sistema se realizó con Javascript, que mediante programación permite interactuar con los elementos y manipular datos según las necesidades.

Algunas de las ventajas de utilizar Javascript son:

- Rapidez al ejecutar funciones, brindando agilidad al sistema.
- Compatibilidad con la mayoría de los navegadores actuales más utilizados.
- Muy útil para funciones sencillas, ya que resultan fáciles de programar.

#### 4.1.1. PHP

El núcleo central del desarrollo se realizó con PHP por su agilidad para la generación de paginas dinámicas, es un lenguaje con la capacidad de ser embebido en páginas HTML que resulta sencillo de trabajar pero igualmente tiene la potencia suficiente para aplicaciones con mayores exigencias, entre sus características permite la consulta a bases de datos del lado del servidor desde un cliente [PHP, 2017].

Las ventajas de utilizar PHP son:

- Es de código abierto, lo que significa que no hay que pagar una licencia.
- El acceso a bases de datos es muy sencillo.
- Su curva de aprendizaje no es elevada.
- Es multiplataforma, por lo que opera en varios sistemas operativos.

#### 4.1.2. Bootstrap

Para el apartado visual, se utilizo Bootstrap, que es un framework que permite realizar interfaces web basadas en CSS y JavaScript, siendo su principal atractivo el ajustarse al dispositivo final del usuario.

Bootstrap fue creado por Twitter, es de código abierto y ayuda a mantener los estilos de un sitio web de manera más organizada.

Las ventajas de utilizar Bootstrap son:

- Agilizar el desarrollo de interfaces al contar con estilos propios ya establecidos.
- Adaptabilidad del sitio a distintos tipos de dispositivos, desde smartphones hasta PC's de escritorio.
- Soporte por parte de la comunidad, por ser uno de los frameworks más utilizados de los últimos años.

#### 4.1.3. MySQL Workbench

El Sistema Gestor de Bases de Datos elegido es MySQL Workbench. El cual fue desarrollado por Oracle Corporation, y se encuentra bajo licencia Open Source, es un gestor ligero y ágil [Rafael, 2005].

MySQL Workbench nos aporta las siguientes características:

## 4. TECNOLOGÍAS A UTILIZAR

---

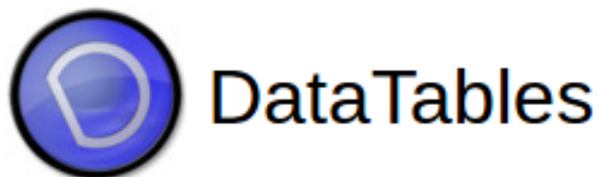
- Conectividad segura.
- Amplia disponibilidad en sistemas compatibles.
- Documentación muy extendida, por ser uno de los SGBD más populares y utilizados.
- Es uno de los gestores con mayor rendimiento.
- Está optimizado para equipos con varios procesadores.

### 4.1.4. DataTables

En cuanto al tratamiento y presentación de las tablas para la visualización y manipulación de información, se hará uso del plug-in para jQuery, DataTables, que otorga dinamismo y añade características de gran utilidad como lo son la búsqueda y el ordenamiento por columnas, entre otras.

Al utilizar DataTables obtenemos las siguientes ventajas:

- Dinamismo y facilidad de navegación en las tablas del sitio.
- Compatibilidad con Bootstrap para homogeneizar el estilo del sistema.
- Búsqueda y paginación de información para agilizar la consulta de datos por parte del usuario final.
- Capacidad de exportar los datos como archivos de Excel.
- Integración con el resto de la funcionalidad al basarse en Javascript.



**Figura 4.2:** DataTables

#### 4.1.5. AWK

Para la limpieza y transformación de la información se empleó AWK que es un lenguaje de programación diseñado para procesar datos en archivos de texto haciendo más sencilla su interpretación.

AWK ofrece las siguientes ventajas:

- Facilidad de uso al ser parecido a lenguajes como C.
- Capacidad de autoejecutar los scripts en periodos de tiempo establecidos.
- Programas sencillos y ágiles en su ejecución.

#### 4.1.6. Highcharts

La necesidad de representar la información en forma gráfica requiere del uso de un framework adecuado, como lo es Highcharts, que se caracteriza por su sencillez y alta compatibilidad al tener soporte con Internet Explorer 6, una versión bastante desactualizada del navegador.

Al utilizar Highcharts obtenemos las siguientes ventajas:

- Compatibilidad con todos los navegadores, tanto de escritorio como en sus versiones móviles.
- Gran cantidad de gráficos diferentes para utilizar según convenga.
- Basado en Javascript, por lo que la funcionalidad del sistema continua homologada en la misma tecnología.
- Facilidad de uso, su implementación resulta muy sencilla.



**Figura 4.3:** Highcharts





# Proceso de automatización de la información

---

## 5.1. Scripts necesarios

Los scripts necesarios para la correcta automatización del sistema son los siguientes:

1. Scripts para la recolección de archivos de log en servidores remotos.
2. Scripts para la limpieza de la información en los archivos de log.
3. Scripts para la carga en base de datos.
4. Scripts para la ejecución periódica y el borrado de archivos anteriores.
5. Scripts de consulta a base de datos para alimentar las gráficas del sistema.

Dichos scripts tienen características variadas y se realizaron en distintos lenguajes como son AWK, BASH o SQL en su totalidad o bien, embebidos en scripts más largos.

El proceso ETL al ser una actividad cíclica permite el uso de scripts de esta naturaleza para automatizar tareas, evitar errores humanos y agilizar la actualización de almacenes de datos.

A continuación se muestra el flujo de la información a través de equipos cliente y servidor, desde la generación de los datos en el servidor, hasta el despliegue de los mismos en un navegador web del lado del cliente.

## 5. PROCESO DE AUTOMATIZACIÓN DE LA INFORMACIÓN

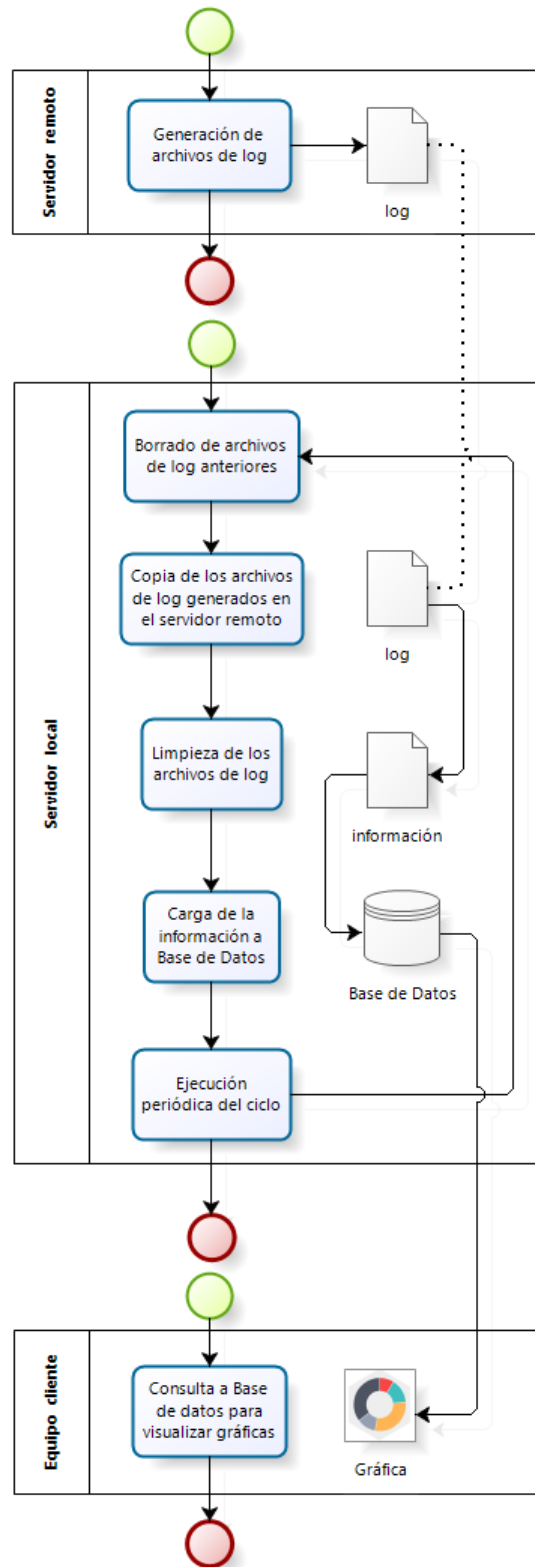


Figura 5.1: Diagrama de flujo

## 5.2. Codificación

Una vez se tiene claros los requerimientos y el plan de trabajo la codificación resulta más sencilla, al modularizar las páginas en plantillas también se favorece la reutilización de código ahorrando tiempo de desarrollo.

El Home fue la primera página en ser elaborada con el objetivo de contar con un sitio general de enlace al resto de los desarrollos, seguido del cuadro de mando del DSR. Teniendo la base del sitio se añadieron el resto de páginas en estructura solamente para realizar el apartado referente a la navegación. Una vez que la navegación por el sistema se finalizó, el siguiente paso fue la adición de gráficas, para este momento las consultas que alimentan las gráficas ya se encontraban realizadas por lo que el principal objetivo fue la personalización y colocación dentro de las páginas, tamaño, ancho de los elementos, color e interacción con el usuario.

Por último la carga del proyecto en el servidor y su despliegue en los paneles de monitoreo del área de soporte.

## 5.3. Generación reportes

Sin duda alguna la mejor presentación para los datos es en forma de gráficas, sin embargo tener los datos en un reporte resulta muy práctico si es que se requieren análisis de distinto tipo, para este propósito el sistema permite la creación de reportes estadísticos, mediante la exportación de los datos que alimentan a las gráficas, se emplea el formato de salida de valores separados por comas CSV.

Fecha	Nodo	Server	Msg_rx_peak	Msg_rx_avg
2016-12-11	DSR_SOAM01	ALL	4768	3817.87
2016-12-11	DSR_SOAM02	SERVIDOR 1	0	0
2016-12-11	DSR_SOAM01	SERVIDOR 2	0	0
2016-12-11	DSR_SOAM02	SERVIDOR 4	4725	4328.77
2016-12-11	DSR_SOAM01	SERVIDOR 6	4768	4352.9
2016-12-11	DSR_SOAM02	SERVIDOR 3	3065	2771.93
2016-12-11	DSR_SOAM01	SERVIDOR 7	0	0
2016-12-11	DSR_SOAM02	SERVIDOR 5	0	0
2016-12-11	DSR_SOAM01	ALL	5201	4004.83
2016-12-11	DSR_SOAM02	SERVIDOR 8	0	0

Figura 5.2: Ejemplo de tabla

### 5.4. Diseño de la interfaz de usuario

El diseño de la interfaz de usuario, es sencillo, apelando al minimalismo para facilitar el uso del sistema, se optó por estandarizar las vistas del sistema, con colores representativos de la empresa y sin sobrecargarlas de elementos para evitar distraer la atención de los usuarios, se cuenta con un menú superior que permite el salto de una estadística a otra sin mayor complicación, de igual manera se puede volver al Home del sistema mediante un clic en “Desempeño de Red” en el header de cualquier página, la información a mostrar y es la que más protagonismo y énfasis tiene, grandes gráficas que abarcan casi toda la pantalla y colores llamativos, existen gráficas con un sistema de umbrales basados en semáforos para el correcto monitoreo en todo momento.

La interfaz también es adaptable para dispositivos con pantallas pequeñas, adaptándose a tabletas y teléfonos celulares sin ninguna complicación.

#### 5.4.1. Flujo de pantallas

El flujo a través de pantallas en la navegación del sistema se planteó de la siguiente manera:

1. Home — Es la página inicial desde la que se puede acceder al resto de desarrollos para el completo monitoreo de la red, el desarrollo realizado en este proyecto es el primero en esta página, DSR.
2. DSR — Cuadro de mando de comportamiento general del DSR, para un rápido vistazo a las estadísticas más útiles.
3. Ocupacion de interfaces — Detalle del aprovechamiento y utilización de las interfaces del DSR, permitiendo seleccionar por día y hora para análisis más profundos de incidencias.
4. Capacidad de MPS — Se observa la actividad de los usuarios y se hacen estimaciones de comportamiento futuro, también se muestra el balanceo de los MPS.
5. RAM — Gráficas de la actividad pico y promedio de las memorias RAM según los MPS del DSR, con intervalos tipo semáforo para su correcto monitoreo.
6. Carga de CPUs — Muestra la distribución de cargas de trabajo en los distintos núcleos de cada elemento del DSR.
7. Answer Timeout — Registro de los retardos que experimenta la red al responder a una solicitud de acceso.

A continuación las pantallas correspondientes.

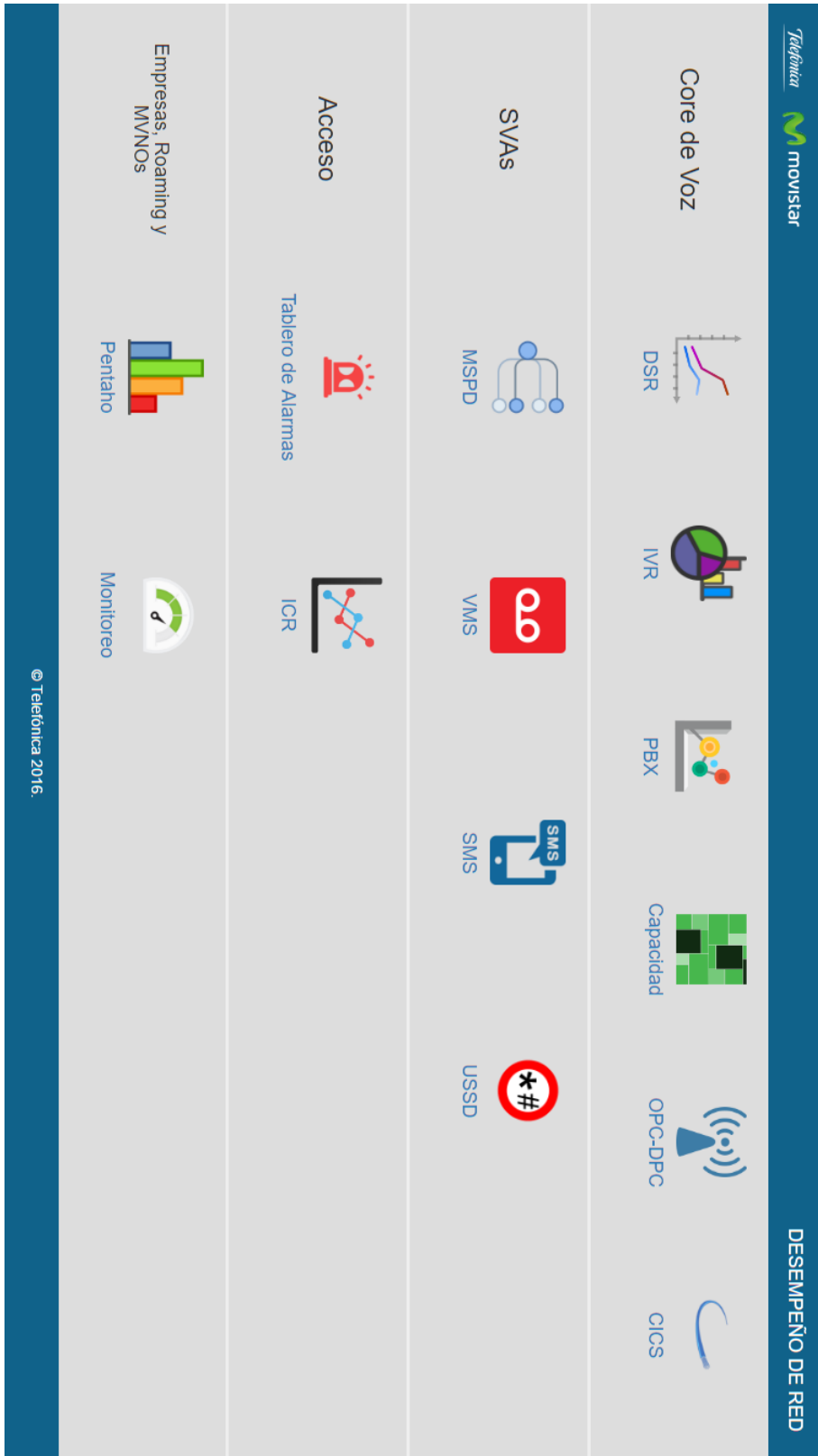


Figura 5.3: Home

## 5. PROCESO DE AUTOMATIZACIÓN DE LA INFORMACIÓN

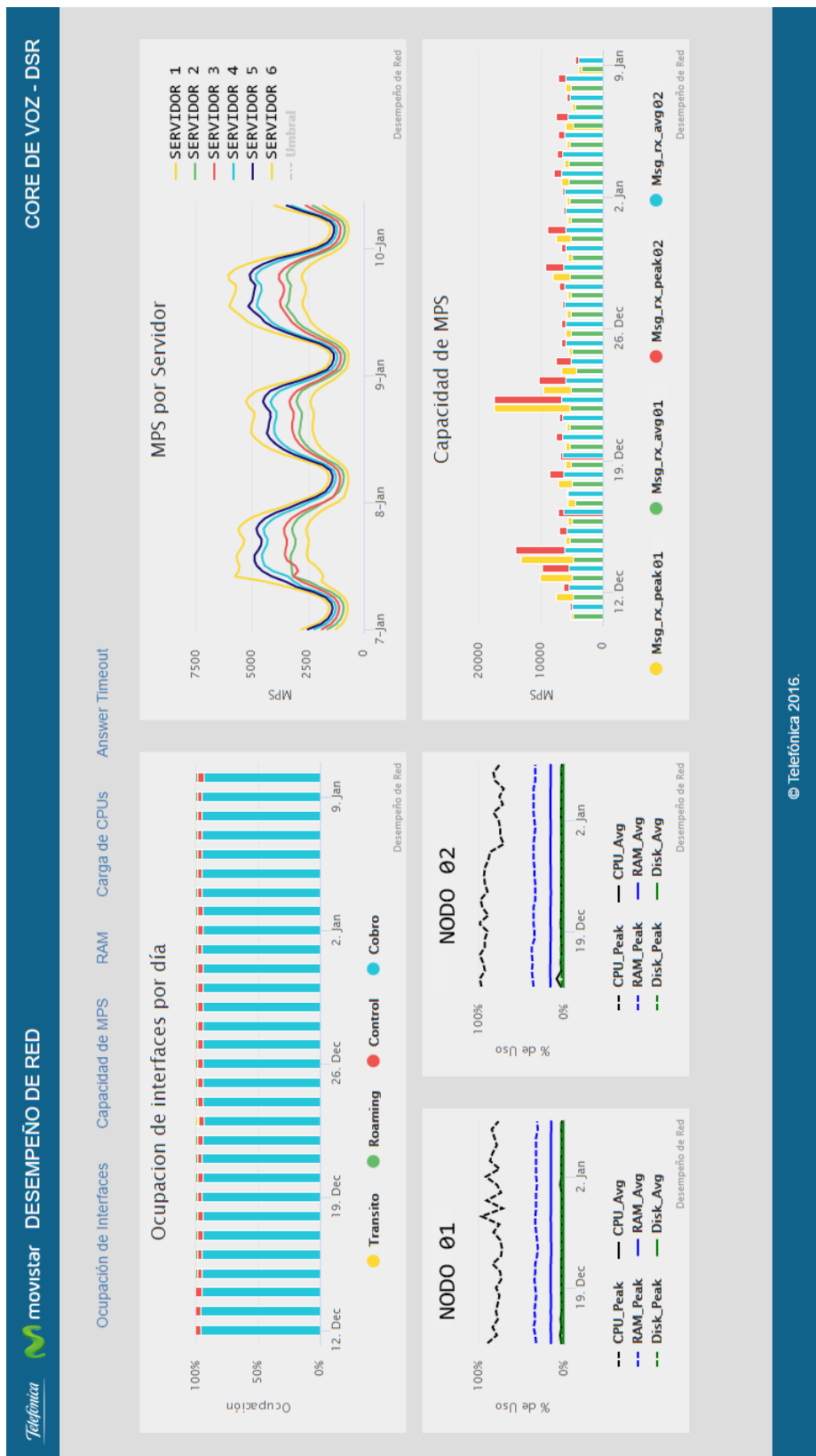


Figura 5.4: Cuadro de Mando del DSR



Figura 5.5: Ocupacion de interfaces

## 5. PROCESO DE AUTOMATIZACIÓN DE LA INFORMACIÓN



Figura 5.6: Capacidad de MPS



## 5.4 Diseño de la interfaz de usuario

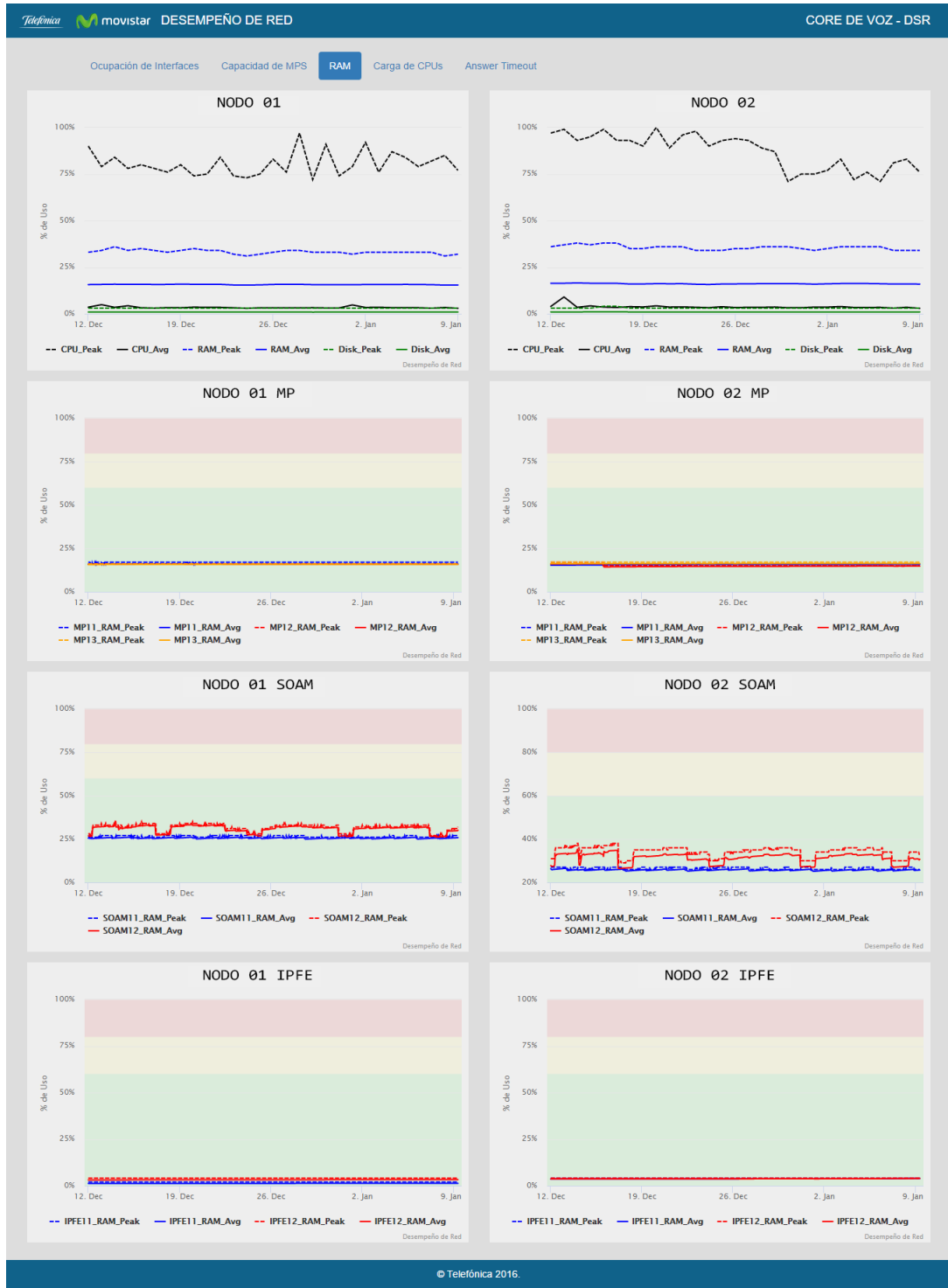


Figura 5.7: RAM

## 5. PROCESO DE AUTOMATIZACIÓN DE LA INFORMACIÓN



Figura 5.8: Carga de CPUs



Figura 5.9: Answer Timeout

### 5.5. Análisis de información gráfica

El último paso del análisis consiste en la correcta interpretación de la información obtenida, como ejemplo en la siguiente gráfica se muestra un comportamiento no esperado.

Es una gráfica de balanceo de servidores, en la primera imagen se observa que la carga de trabajo en no se encuentra distribuida correctamente, mostrando con mayor o menos cantidad de MPS.

En la segunda imagen se ven líneas que se sobrepone unas a otras, denotando un balanceo más equilibrado.

Con solo dar un vistazo a la representación de los datos, se puede detectar un fallo y comunicar al área correspondiente para ser atendido a la brevedad, está es la finalidad del desarrollo aquí mostrado, mejorar el tiempo de toma de decisiones.

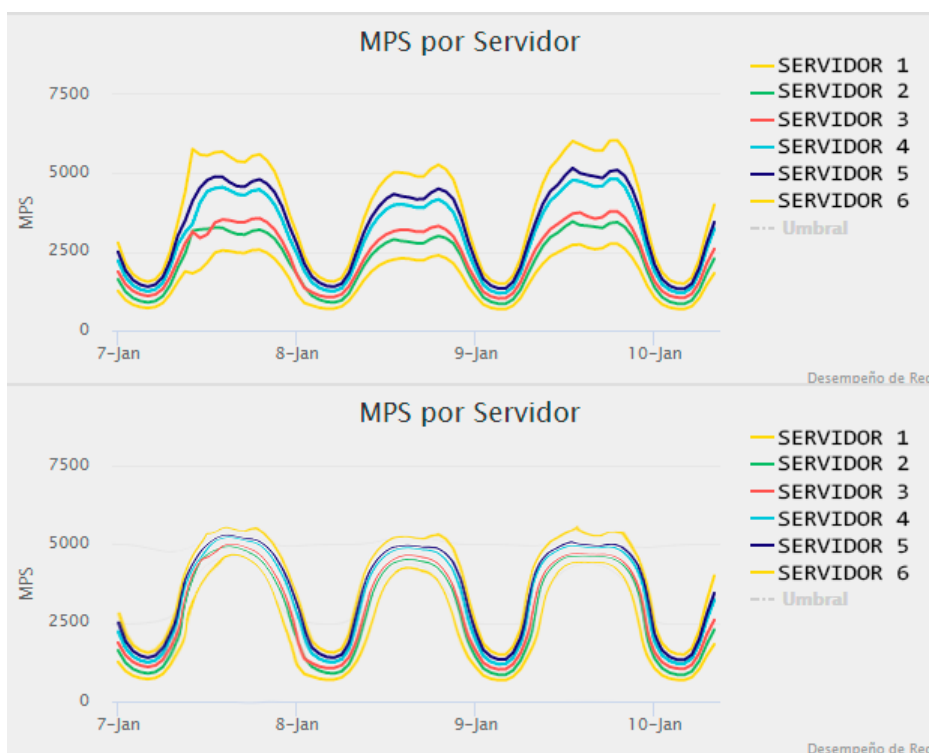


Figura 5.10: Balanceo de servidores

# Resultados

---

Las pruebas para verificar el correcto funcionamiento del sistema se realizaron llevando a cabo el procedimiento manual y haciendo una comparación entre los resultados obtenidos en Excel con los que arroja el sistema.

La accesibilidad al sistema es posible desde cualquier equipo con acceso a la internet empresarial como se requería

La línea de tendencia a futuro es bastante precisa en la mayoría de los casos en los que el comportamiento de la red es usual.

La adaptabilidad del sistema a pantallas más pequeñas es transparente al usuario.

Las tablas de Excel que se pueden exportar funcionan correctamente y contienen la información esperada.

El intervalo de actualización de los datos se refleja correctamente por lo que se tiene confianza en que los datos representados hoy en día en el nuevo sistema de monitoreo son correctos y actualizados.

De lo anterior se concluye que el objetivo planteado en un inicio, se cumplió al agilizar el análisis del comportamiento de la red y la automatización del procesamiento de los datos, resultando así en la detección más rápida de incidencias.

La toma de decisiones también se vio mejorada al poder prevenir incidencias gracias al correcto monitoreo de los datos generados en la red.

El optar por una interfaz con gráficas y sustituir los archivos de log resultó en una decisión acertada, la selección de los KPIs fue otro acierto ya que contando solo con las gráficas necesarias se tiene un panorama amplio del funcionamiento de la red.



## Conclusiones

---

La capacidad de análisis desarrollada durante la carrera de Ingeniería en Computación es necesaria en el desarrollo de cualquier sistema, para comprender el negocio del mismo y dar propuestas eficientes para su desarrollo o mejora, mi experiencia en Telefónica no es la excepción ya que para poder iniciar es necesario comprender el flujo de la información, el aprender nuevas tecnologías y adaptar los procesos ya existentes a dichas tecnologías es el siguiente paso y el más interesante.

Al tener un amplio abanico de materias impartidas en la facultad muchos de los términos referentes a la telefonía y las redes no me resultaron ajenos, la naturaleza de la empresa más enfocada en las telecomunicaciones me dejó también nuevo conocimiento y más curiosidad por este ramo de la tecnología al mismo tiempo que comprendí que la mayoría de las profesiones se relacionan entre si.

La experiencia profesional que brinda la industria es muy satisfactoria desde mi punto de vista, ya que en solo seis meses obtuve tanto aprendizaje como experiencia invaluable, confianza en mi trabajo y en mi mismo ya que las labores que desempeñe se reflejaban en la mejora no solo del área de calidad y desempeño, también a nivel empresa.





## Bibliografía

---

- [1] Carlos, T. J. (2010). *Diseño y Explotación de Almacenes de Datos*. Editorial Club Universitario. 9
- [2] Diego, G. J. (2012). *El Gran Libro de HTML5, CSS3 y Javascript*. MARCOMBO. 16
- [3] Kenneth, E. K. (2005). *Análisis y Diseño de Sistemas, Sexta Edición*. Pearson Educación. 9
- [4] Kimball, R. (2004). *The Data Warehouse ETL Toolkit: Practical Techniques for Extracting, Cleanning, Conforming, and Delivering Data*. Wiley. 7
- [5] O'reilly, J. J. (1994). *Principios de Telecomunicaciones, Segunda Edición*. Addison Wesley Iberoamericana. 10
- [6] PHP (Enero de 2017). *¿Qué es PHP?* Recuperado de: <http://php.net/manual/es/intro-what-is.php>. 17
- [7] Rafael, C. P. (2005). *Bases de Datos*. UOC Formación de Posgrado. 17
- [8] Telefónica (2011). *Las Telecomunicaciones de Nueva Generación*. Lerko Print S.A. 7
- [9] Telefónica (Julio de 2017a). *Acerca de Telefónica*. Recuperado de: [https://www.telefonica.com/es/web/about\\_telefonica](https://www.telefonica.com/es/web/about_telefonica). 4
- [10] Telefónica (Julio de 2017b). *¿Dónde estamos?* Recuperado de: [https://www.telefonica.com/es/web/about\\_telefonica/donde\\_estamos/mexico](https://www.telefonica.com/es/web/about_telefonica/donde_estamos/mexico). 3



## A.1. PHP

### A.1.1. header.php

Plantilla que contiene el header de todas la pantallas.

```
1 <!DOCTYPE html>
2 <head>
3   <meta charset="UTF-8">
4   <title>DSR</title>
5   <link rel="shortcut icon" href="img/favicon.png">
6   <link rel="stylesheet" href="css/estilos.css">
7   <link rel="stylesheet" href="http://maxcdn.bootstrapcdn.com/
   bootstrap/3.3.6/css/bootstrap.min.css">
8   <link rel="stylesheet" href="css/dataTables.css">
9   <link rel="stylesheet" href="css/buttons.css">
10  <!-- Funciones propias -->
11  <script src="js/graficas.js"></script>
12  <script src="js/tablas.js"></script>
13  <!-- Librerias externas -->
14  <script src="js/jquery.js"></script>
15  <script src="js/highcharts.js"></script>
16  <script src="js/highcharts-3d.js"></script>
```

## A. CÓDIGO

---

```
17 <script src="js/drilldown.js"></script>
18 <script src="js/dataTables.js"></script>
19 <script src="js/bootstrap.js"></script>
20 <script src="js/buttons.js"></script>
21 <script src="js/html5.js"></script>
22 </head>
23 <body style="background-color: #ddd" onload="Inicio();">
24 <nav class="navbar navbar-default navbar-fixed-top" style="
    background-color: #10628a">
25 <div class="container-fluid">
26 <div class="navbar-header">
27 <a class="navbar-brand" href="http://www.telefonica.com.mx/"
    target="_blank"></a>
28 <a class="navbar-brand" href="http://www.movistar.com.mx/"
    target="_blank"></a>
29 <a class="navbar-brand" href="http://10.225.164.51/desempeno-
    de-red/" style="color: white">DESEMPENO DE RED</a>
30 </div>
31 <div class="navbar-header navbar-right">
32 <a class="navbar-brand" href="index.php" style="color: white
    ">CORE DE VOZ - DSR</a>
33 </div>
34 </div>
35 </nav>
```

### A.1.2. index.php

Cuerpo de la página principal de monitoreo del DSR.

```
1 <?php require ("views/header.php");?>
2
3 <div class="container" style="padding-bottom: 15px;">
4   <ul class="nav nav-pills">
5     <li role="presentation"><a href="ocupacion-interfaces.php">
6       Ocupacion de Interfaces </a></li>
7     <li role="presentation"><a href="mps.php">Capacidad de MPS </a>
8       </li>
9     <li role="presentation"><a href="ram.php">RAM </a></li>
10    <li role="presentation"><a href="#">Carga de CPUs </a></li>
11    <li role="presentation"><a href="diameter-egress-transaction.
12      php">Answer Timeout </a></li>
13  </ul>
14 </div>
15
16 <div class="container-fluid" style="padding-bottom: 15px;">
17   <div class="col-md-6" id="column-chart-dia" style="height: 250
18     px; margin: 0 auto; box-shadow: 0px 0px 0px 0px #cbcbcb">
19   </div>
20   <div class="col-md-6" id="line-chart-mp" style="height: 250px;
21     margin: 0 auto; box-shadow: 0px 0px 0px 0px #cbcbcb">
22   </div>
23 </div>
```

## A. CÓDIGO

---

```
24
25 <div class="col-md-3" id="line-chart-tla" style="height: 250px;
    margin: 0 auto; box-shadow: 0px 0px 0px 0px #cbcbcb">
26 </div>
27
28 <div class="col-md-6" id="capacity-chart-dia" style="height:
    250px; margin: 0 auto; box-shadow: 0px 0px 0px 0px #cbcbcb">
29 </div>
30 </div>
31
32 <script>
33 function Inicio() {
34     Highcharts.setOptions({
35         colors: ['#FDD835', '#66BB6A', '#EF5350', '#26C6DA', '#160C76
            '],
36     });
37
38     OcupacionDSR();
39     CapacidadPorMP();
40     MtyRAM();
41     TlaRAM();
42     CapacidadMPS();
43 }
44 </script>
45
46 <?php require ("views/footer.php");?>
```

### A.1.3. footer.php

Plantilla que contiene el footer de todas la pantallas.

```
1 <footer class="footer">
2 <div class="container">
3 <center>
4 <p style="color: white"> Telef nica 2016.</p>
5 </center>
6 </div>
7 </footer>
8 </body>
9 </html>
```

### A.1.4. capacidadMPS.php

Detalle de la consulta para alimentar la gráfica de capacidad de MPS.

```
1 <?php
2 require ("dbinfo.php");
3 set_time_limit(500);
4 date_default_timezone_set('America/Mexico_City');
5
6 $con = mysql_connect('10.225.164.45:3306', $username, $password)
7     ;
8 if (!$con)
9     die('Not connected : '.mysql_error());
10
11 $db = mysql_select_db($database, $con);
12 if (!$db)
13     die ('Can\'t use db : '.mysql_error());
14
15 $json = array();
```

## A. CÓDIGO

---

```
16 $peakMty['name'] = 'Msg_rx_peak-Mty';
17 $averageMty['name'] = 'Msg_rx_avg-Mty';
18 $peakTla['name'] = 'Msg_rx_peak-Tla';
19 $averageTla['name'] = 'Msg_rx_avg-Tla';
20
21 $query = mysql_query("SELECT Fecha, Nodo, MAX(RxMsgRatePeakMp)
    AS Msg_rx_peak, MAX(RxMsgRateAvgMp) AS Msg_rx_avg FROM
    DSR_CpuLoad_MPS WHERE Nodo = 'DSR_SOAM_MONTERREY' AND Server
    LIKE '%MP%' AND Fecha >= (curdate() - interval 30 day) GROUP
    BY Fecha, Nodo ORDER BY Fecha");
22 while ($row = mysql_fetch_array($query)) {
23     $peakMty['data'][] = $row[2];
24     $averageMty['data'][] = $row[3];
25 }
26
27 $query = mysql_query("SELECT Fecha, Nodo, MAX(RxMsgRatePeakMp)
    AS Msg_rx_peak, MAX(RxMsgRateAvgMp) AS Msg_rx_avg FROM
    DSR_CpuLoad_MPS WHERE Nodo = 'DSR_SOAM_TLALNEPANTLA' AND
    Server LIKE '%MP%' AND Fecha >= (curdate() - interval 30 day)
    GROUP BY Fecha, Nodo ORDER BY Fecha");
28 while ($row = mysql_fetch_array($query)) {
29     $peakTla['data'][] = $row[2];
30     $averageTla['data'][] = $row[3];
31 }
32
33 $peakMty['pointPadding'] = 0.2;
34 $averageMty['pointPadding'] = 0.3;
35 $peakTla['pointPadding'] = 0.2;
36 $averageTla['pointPadding'] = 0.3;
37
38 $peakMty['pointPlacement'] = -0.25;
39 $averageMty['pointPlacement'] = -0.25;
40 $peakTla['pointPlacement'] = 0.25;
```



```
41 $averageTla['pointPlacement'] = 0.25;
42
43 array_push($json,$peakMty);
44 array_push($json,$averageMty);
45 array_push($json,$peakTla);
46 array_push($json,$averageTla);
47
48 mysql_close($con);
49 echo json_encode($json, JSON_NUMERIC_CHECK);
50 ?>
```

#### A.1.5. ocupacionDSR.php

Detalle de la consulta para alimentar la gráfica de ocupación del DSR.

```
1 <?php
2 require ("dbinfo.php");
3 set_time_limit(500);
4 date_default_timezone_set('America/Mexico_City');
5
6 $con = mysql_connect('10.225.164.45:3306', $username, $password)
7     ;
8 if (!$con)
9     die('Not connected : '.mysql_error());
10
11 $db = mysql_select_db($database, $con);
12 if (!$db)
13     die ('Can\'t use db : '.mysql_error());
14
15 $json = array();
16
17 $transito['name'] = 'Transito';
18 $roaming['name'] = 'Roaming';
```

## A. CÓDIGO

---

```
18 $control['name'] = 'Control';
19 $cobro['name'] = 'Cobro';
20
21 $query = mysql_query("SELECT Transito, Roaming, Control, Cobro
    FROM CoreVoz.dsr_ocupacion_interfaces WHERE Fecha >= date_sub
    (now(), interval '31' day) ORDER BY Fecha");
22 while ($row = mysql_fetch_array($query)) {
23     $transito['data'][] = $row[0];
24     $roaming['data'][] = $row[1];
25     $control['data'][] = $row[2];
26     $cobro['data'][] = $row[3];
27 }
28
29 array_push($json,$transito);
30 array_push($json,$roaming);
31 array_push($json,$control);
32 array_push($json,$cobro);
33
34 mysql_close($con);
35 echo json_encode($json, JSON_NUMERIC_CHECK);
36 ?>
```

### A.1.6. detalleOcupacion.php

Detalle de la consulta para alimentar la gráfica a detalle de la ocupación de DSR.

```
1 <?php
2 require ("dbinfo.php");
3 set_time_limit(500);
4 date_default_timezone_set('America/Mexico_City');
5
6 $con = mysql_connect('10.225.164.45:3306', $username, $password)
    ;
```

```
7   if (!$con)
8       die('Not connected : '.mysql_error());
9
10  $db = mysql_select_db($database, $con);
11  if (!$db)
12      die ('Can\'t use db : '.mysql_error());
13
14  $json = array();
15
16  $transito['name'] = 'Transito';
17  $roaming['name'] = 'Roaming';
18  $hss['name'] = 'HSS';
19  $mme['name'] = 'MME';
20  $volte['name'] = 'VoLTE';
21  $dpi['name'] = 'DPI';
22  $sdp['name'] = 'SDP';
23  $ngin['name'] = 'NGIN';
24
25  $transito['color'] = '#FDD835';
26  $roaming['color'] = '#66BB6A';
27  $hss['color'] = '#EF5350';
28  $mme['color'] = '#EF5350';
29  $volte['color'] = '#EF5350';
30  $dpi['color'] = '#26C6DA';
31  $sdp['color'] = '#26C6DA';
32  $ngin['color'] = '#26C6DA';
33
34  $query = mysql_query("SELECT transito_por_dia, roaming_por_dia,
        hss_por_dia, mme_por_dia, volte_por_dia, dpi_por_dia,
        sdp_por_dia, ngin_por_dia FROM CoreVoz.
        dsr_ocupacion_interfaces WHERE Fecha = '$_REQUEST[dia]'");
35  while ($row = mysql_fetch_array($query)) {
36      $transito['y'] = $row[0];
```

## A. CÓDIGO

---

```
37     $roaming['y'] = $row[1];
38     $hss['y'] = $row[2];
39     $mme['y'] = $row[3];
40     $volte['y'] = $row[4];
41     $dpi['y'] = $row[5];
42     $sdp['y'] = $row[6];
43     $ngin['y'] = $row[7];
44 }
45
46 array_push($json,$transito);
47 array_push($json,$roaming);
48 array_push($json,$hss);
49 array_push($json,$mme);
50 array_push($json,$volte);
51 array_push($json,$dpi);
52 array_push($json,$sdp);
53 array_push($json,$ngin);
54
55 mysql_close($con);
56 echo json_encode($json, JSON_NUMERIC_CHECK);
57 ?>
```

## A.2. CSS

### A.2.1. estilos.php

Extracto del archivo principal de estilos del sistema.

```
1  html {
2      position: relative;
3      min-height: 100%;
4  }
5
```

```
6 body {
7   padding-top: 70px;
8   padding-bottom: 60px;
9   min-height: 450px;
10 }
11
12 .footer {
13   position: absolute;
14   padding-top: 10px;
15   bottom: 0;
16   width: 100%;
17   height: 50px;
18   background-color: #10628a;
19 }
20
21 .grow {
22   transition: all .2s ease-in-out;
23 }
24 .grow:hover {
25   transform: scale(1.1);
26 }
```

## A.3. AWK

### A.3.1. process\_data.js

Ejemplo de un script para el procesado de los archivos de log generados en los servidores.

```
1 #!/bin/bash
2
3 BEGIN{FS = ","};
4
```

## A. CÓDIGO

---

```
5 $1 ~ /201/ {
6   Fecha=substr($1,1,10);
7   Hora=substr($1,12,2);
8
9   if(FILENAME=="mty.csv")
10    printf "%s\t%d\tDSR_SOAM_MONTERREY\t%s\t%d\t%f\t%d\t%f\n",
11    Fecha,Hora,$4,$6,$5,$10,$7 > "OAMSS.txt";
12   else if(FILENAME=="tla.csv")
13    printf "%s\t%d\tDSR_SOAM_TLALNEPANTLA\t%s\t%d\t%f\t%d\t%f\n",
14    Fecha,Hora,$4,$6,$5,$10,$7 >> "OAMSS.txt";
15 };
```

### A.4. JavaScript

#### A.4.1. graficas.js

Fragmento del archivo principal de funciones, contiene la definición de los elementos de gráficas del sistema.

```
1 var dia_actual = new Date();
2 dia_actual.setDate(dia_actual.getDate());
3
4 var dia_anterior = new Date();
5 dia_anterior.setDate(dia_anterior.getDate()-1);
6
7 var $diaSeleccionado = dia_actual.toISOString().substring(0, 10);
8 var $hora = "00";
9
10 // Grafica de barras
11 function OcupacionDSR() {
12   $.getJSON({
13     url: "php/ocupacionDSR.php",
14     success: function(result) {
```

```
15     chart = new Highcharts.Chart({
16         chart: {
17             renderTo: 'column-chart-dia',
18             type: 'column',
19             zoomType: 'y',
20             resetZoomButton: {
21                 position: {
22                     x: -10,
23                     y: 10
24                 },
25                 relativeTo: 'chart'},
26         },
27         title: {
28             text: 'Ocupacion de interfaces por d a '
29         },
30         xAxis: {
31             type: 'datetime',
32             dateTimeLabelFormats: {
33                 day: '%e-%b'
34             }
35         },
36         yAxis: {
37             title: {
38                 text: 'Ocupacion'
39             },
40             labels: {
41                 format: '{value}%'
42             }
43         },
44         tooltip: {
45             pointFormat: '<span style="color:{series.color}">{series.
46                 name}</span>: <b>{point.percentage:.2f}%</b><br/>',

```

## A. CÓDIGO

---

```
47     crosshairs: true
48   },
49   plotOptions: {
50     column: {
51       stacking: 'percent'
52     },
53     series: {
54       cursor: 'pointer',
55       pointStart: Date.UTC(dia_actual.getFullYear(), dia_actual.
56         getMonth()-1, dia_actual.getDate()+1),
57       pointInterval: 24 * 3600 * 1000,
58       point: {
59         events: {
60           // Gráfica de Pay por día
61           click: function () {
62             $diaSeleccionado = new Date(this.x).toISOString().
63               substring(0, 10);
64             DetalleOcupacion();
65             OcupacionDSRHora();
66             DetalleOcupacionHora();
67           }
68         }
69       },
70       series: result
71     });
72   }
73 });
74 }
75
76 // Grafica de Pay por dia
77 function DetalleOcupacion() {
```



```
78 $.getJSON({
79   url: "php/detalleOcupacion.php",
80   data: {"dia": $diaSeleccionado},
81   success: function(result) {
82     chart = new Highcharts.Chart({
83       chart: {
84         renderTo: 'pie-chart-dia',
85         type: 'pie',
86         options3d: {
87           enabled: true,
88           alpha: 45
89         }
90       },
91       title: {
92         text: $diaSeleccionado
93       },
94       tooltip: {
95         useHTML: true,
96         pointFormat: '{point.name}'
97       },
98       plotOptions: {
99         pie: {
100          dataLabels: {
101            enabled: true,
102            format: '<b>{point.name}</b>: {point.percentage:.2f} %',
103          },
104          allowPointSelect: true,
105          cursor: 'pointer',
106          showInLegend: true,
107          innerSize: 70,
108          depth: 70,
109        }
110      },
```

## A. CÓDIGO

---

```
111     series: [{
112         data: result
113     }]
114 });
115 }
116 });
117 }
```

### A.4.2. tablas.js

Fragmento del archivo de funciones para la generación de tablas en el sistema.

```
1 function DatosMPS() {
2     $.getJSON({
3         url: "php/datosMPS.php",
4         success: function(result) {
5             $('#datos-mps').DataTable({
6                 dom: 'Bfrtp',
7                 lengthMenu: [
8                     [ 10, 25, 50, -1 ],
9                     [ '10 registros', '25 registros', '50 registros', 'Todos los
10                        registros' ]
11                 ],
12                 buttons: [
13                     "csv", 'pageLength'
14                 ],
15                 bProcessing: true,
16                 sAjaxSource: "php/datosMPS.php",
17                 sAjaxDataProp: "data"
18             });
19         }
20     });
21 }
```



## Glosario

---

<b>CSS</b>	<b>Cascading Style Sheets</b> Hojas de Estilo en Cascada que describen como se muestran los elementos de una página web.
<b>DASHBOARD</b>	Representación gráfica y visual de las principales métricas o KPIs de un sistema.
<b>DSR</b>	<b>Diameter Signaling Router</b> Dispositivo que permite la comunicación entre los servidores y el software dentro del núcleo de la red LTE.
<b>DSS</b>	<b>Decision Support System</b> Sistema de Apoyo a la Decisión.
<b>ETL</b>	<b>Extract Transform Load</b> Extraer, Transformar y Cargar, proceso que permite la transformación y presentación de información.
<b>FTP</b>	<b>File Transfer Protocol</b> Permite el intercambio de archivos entre equipos de una red de manera remota.
<b>FRAMEWORK</b>	Marco de trabajo para desarrollar aplicaciones de software.
<b>HTML</b>	<b>HyperText Markup Language</b> Lenguaje de Marcado de HiperTexto para la elaboración de páginas web.
<b>KPI</b>	<b>Key Performance Indicator</b> Indicador Clave del Desempeño, métrica utilizada para medir el desempeño de un sistema.

---

<b>LTE</b>	<b>L</b> ong <b>T</b> erm <b>E</b> volution Evolución a Largo Plazo, en telecomunicaciones en un estándar para la transmisión de datos a alta velocidad.
<b>MPS</b>	<b>M</b> ensajes <b>P</b> or <b>S</b> egundo
<b>OSS</b>	<b>O</b> perations <b>S</b> upport <b>S</b> ystem Sistema de Soporte de Operaciones, aplicación de software que da soporte a una red de telecomunicaciones.
<b>PHP</b>	<b>H</b> yperext <b>P</b> re <b>P</b> rocessor Lenguaje de código abierto para desarrollo web y que puede ser incrustado en HTML.
<b>SGBD</b>	<b>S</b> istema <b>G</b> estor de <b>B</b> ases de <b>D</b> atos Software que permite la manipulación de los datos, creación de tablas, procedimientos almacenados, entre otras acciones.
<b>TIC</b>	<b>T</b> ecnologías de <b>I</b> nformación y <b>C</b> omunicación
<b>TPS</b>	<b>T</b> ransacciones <b>P</b> or <b>S</b> egundo