



UNIVERSIDAD AMERICANA DE ACAPULCO
"EXCELENCIA PARA EL DESARROLLO"

FACULTAD DE INGENIERÍA EN COMPUTACIÓN
INCORPORADA A LA UNIVERSIDAD NACIONAL
AUTÓNOMA DE MÉXICO
CLAVE DE INCORPORACIÓN 8852-16

**"SISTEMA MONITOR DE LA RED ELÉCTRICA DE
POTENCIA DEL ÁREA CENTRAL DE MÉXICO"**

T E S I S

QUE PARA OBTENER EL TÍTULO DE
INGENIERO EN COMPUTACIÓN

PRESENTA
IVAN GARAVITO GARCIA

DIRECTOR DE TESIS
DR. RENE EDMUNDO CUEVAS VALENCIA



ACAPULCO, GUERRERO, AGOSTO 2016.



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

AGRADECIMIENTOS

A la Fuerza Creadora y mis Guías Espirituales, que me han ayudado a inspirarme aún en los momentos más difíciles.

A mis Padres, por darme toda esa libertad académica desde muy temprana edad, y por todo su impulso a lo largo de mi desarrollo.

A mis Hermanos, por su compañía y apoyo incondicional.

A Yamilka, por todo su apoyo y palabras en momentos críticos de mi vida.

A la Universidad Americana de Acapulco, que por cinco años fue mi segunda casa y que aún después de egresado me recibe con las puertas abiertas.

A la Ing. Eloisa Mercedes Vivas Villasana, por darme todas esas oportunidades para culminar un ciclo tan importante en mi vida.

Al M. en C. José Mario Martínez Castro, por sus valiosos consejos y guiarme en el rumbo correcto para terminar ésta tesis.

DEDICATORIA

A mis Padres, por su amor y esfuerzo para que saliera adelante.
Por sus enseñanzas y consejos que me hicieron hallar mi camino.

A mi gran amiga Citlalli Duarte[†], por ampliar mi visión sobre la vida y su sinfín de posibilidades.

A mi Familia y Amigos, por su compañía y apoyo incondicional.

ÍNDICE GENERAL

| | |
|---|----|
| INTRODUCCIÓN | 7 |
| PLANTEAMIENTO DEL PROBLEMA | 9 |
| JUSTIFICACIÓN | 11 |
| HIPÓTESIS | 13 |
| OBJETIVO GENERAL | 15 |
| OBJETIVOS ESPECÍFICOS | 17 |
| CAPÍTULO 1. ESTADO DEL ARTE | 21 |
| 1.1. Subdirección de Generación | 24 |
| 1.2. Subdirección de Transmisión | 27 |
| 1.3. Subdirección de Distribución | 30 |
| 1.4. Subdirección del CENACE | 33 |
| 1.5. Unidad Terminal Maestra | 34 |
| 1.6. Enviando la Información al CENACE | 34 |
| 1.7. Supervisory Control And Data Acquisition (SCADA) | 39 |
| 1.8. Una Maestra para el Proceso de Transmisión | 43 |
| 1.9. Área de Trabajo | 45 |
| CAPÍTULO 2. CASO DE ESTUDIO | 51 |
| 2.1. Sistemas Actuales | 51 |
| 2.2. Cómo Funcionan | 51 |
| 2.3. Cliente SCADAWeb | 52 |

| | |
|--|-----------|
| 2.4. Cliente WebSICLE | 54 |
| 2.5. Problemática Actual | 54 |
| CAPÍTULO 3. CONCEPTOS Y DEFINICIONES BÁSICAS | 59 |
| 3.1. Bases de Datos | 59 |
| 3.1.1. DBMS – Database Management System | 59 |
| 3.1.2. RDBMS – Relational Database Management System | 59 |
| 3.1.3. SQL - Structured Query Language..... | 60 |
| 3.2. Servidor Web..... | 61 |
| 3.2.1. HTTP – Hypertext Transfer Protocol..... | 62 |
| 3.3. Diseño Web y Aplicaciones | 65 |
| 3.3.1. HTML – Hypertext Markup Language..... | 66 |
| 3.3.2. CSS – Cascade Style Sheet..... | 66 |
| 3.3.3. SVG – Scalable Vector Graphics..... | 67 |
| 3.3.4. XML - Extensible Markup Language | 69 |
| 3.3.5. Web Service – Servicio Web | 70 |
| 3.3.6. JSON – Javascript Object Notation..... | 70 |
| 3.3.7. AJAX..... | 72 |
| 3.3.8. The Dojo Toolkit..... | 72 |
| 3.4. Lenguajes de Programación..... | 73 |
| 3.4.1. PHP – PHP: Hypertext Preprocessor..... | 74 |
| 3.4.2. Javascript..... | 75 |
| 3.5. Expresiones Regulares..... | 76 |

| | |
|--|------------|
| 3.6. Integración de Tecnologías | 77 |
| 3.7. Metodología de Desarrollo de Software | 81 |
| 3.8. Metodología a Utilizar | 84 |
| 3.9. Diagrama Unifilar | 84 |
| 3.9.1. Nomenclatura | 86 |
| CAPÍTULO 4. DESARROLLO E IMPLEMENTACIÓN | 91 |
| 4.1. Estableciendo las Bases | 91 |
| 4.1.1. Concepto de Monitor | 91 |
| 4.1.2. Sistema Monitor y sus Subsistemas | 91 |
| 4.1.3. Metodología de Desarrollo de Software | 92 |
| 4.2. Monitor :: UTMu | 92 |
| 4.2.1. Análisis | 93 |
| 4.2.2. Diseño | 102 |
| 4.2.3. Desarrollo | 108 |
| 4.2.4. Pruebas | 113 |
| 4.3. Monitor :: Web, Cliente | 115 |
| 4.3.1. Análisis | 115 |
| 4.3.2. Diseño | 120 |
| 4.3.3. Desarrollo | 123 |
| 4.3.4. Pruebas | 130 |
| CAPÍTULO 5. RESULTADOS Y TRABAJO A FUTURO | 135 |
| 5.1. Registros de Actividad de Monitor :: UTMu | 135 |

| | |
|---|-----|
| 5.2. Página Principal en Monitor :: Web | 137 |
| 5.3. Monitor del Doble Anillo de 400 kV | 138 |
| 5.4. Monitor de la Red Troncal de 400 kV | 139 |
| 5.5. Monitor de la Red Eléctrica de Potencia de 230 kV | 139 |
| 5.6. Compatibilidad de Navegadores Web | 140 |
| 5.7. Conjuntando Resultados | 141 |
| 5.8. Trabajo a Futuro | 141 |
| CAPÍTULO 6. CONCLUSIONES | 145 |
| BIBLIOGRAFÍA | 147 |
| ÍNDICE DE FIGURAS | 155 |
| ÍNDICE DE TABLAS | 161 |
| ANEXO A. GLOSARIO | 165 |
| ANEXO B. SUBSISTEMA MONITOR :: UTMU » CLASES IMPLEMENTADAS | 183 |
| Anexo B.1. Núcleo del Subsistema Monitor :: UTMu | 183 |
| Anexo B.2. Motores del Subsistema de Adquisición y Reconocimiento de Datos | 184 |
| ANEXO C. SUBSISTEMA MONITOR :: WEB » CLASES IMPLEMENTADAS | 187 |
| Anexo C.1. Núcleo de la Aplicación Servidor del Subsistema Monitor :: Web | 187 |
| Anexo C.2. Jerarquía de Monitores de la Aplicación Servidor | 188 |
| Anexo C.3. Motores Implementados por la Aplicación Servidor | 189 |

| | |
|---|------------|
| Anexo C.4. Plantilla HTML Usada por los Monitores..... | 189 |
| Anexo C.5. Jerarquía de Clases Lado Cliente..... | 191 |
| ANEXO D. REGISTROS DE ACTIVIDAD SIMPLIFICADOS DEL SUBSISTEMA MONITOR :: UTMU | 195 |

INTRODUCCIÓN

Dada la necesidad de realizar un monitoreo integral a corto plazo de los flujos de energía y estado de los interruptores de la Red Troncal Eléctrica de 400 kilovolts (kV) y la Red Eléctrica de Potencia de 230 kV del área central del país, en este proyecto se realizará un sistema basado en los datos de los sistemas Supervisory Control And Data Acquisition (SCADA) de la Comisión Federal de Electricidad (CFE) para permitir la toma efectiva de decisiones en condiciones normales, durante la ocurrencia de fallas y en las acciones de restablecimiento ante los diversos sucesos que pueden acontecer.

A continuación se lista un breve resumen de lo que se puede encontrar en cada capítulo de esta obra:

Capítulo 1. Aquí conoceremos todos los antecedentes del sistema Monitor implementado en el área central en CFE, como se manejaba con anterioridad utilizando los sistemas existentes.

Capítulo 2. Se analizará el funcionamiento de los sistemas existentes, para determinar los puntos de oportunidad y la forma de utilizar su información.

Capítulo 3. Se da a conocer la teoría básica detrás de un sistema Web dinámico.

Capítulo 4. Se muestra todo el desarrollo llevado a cabo para poder crear un sistema de visualización de datos.

Capítulo 5. Registros e impresiones del funcionamiento del sistema, así como las acciones que se pueden llevar a cabo para mejorarlo.

Capítulo 6. Aquí se da a conocer en base a los resultados obtenidos, la conclusión de la hipótesis planteada en este documento.

PLANTEAMIENTO DEL PROBLEMA

La CFE cuenta con diversos sistemas de información en tiempo real con los datos de las subestaciones eléctricas del país. Dichos sistemas son independientes y ajenos a los demás, por lo que llevar un seguimiento de como fluye la energía eléctrica de instalación en instalación es una ardua tarea que lleva tiempo y conlleva a errores por los cambios que se presentan de un momento a otro. Sobre este tenor, conjuntar la información de algún disturbio en la red eléctrica es una tarea difícil de realizar ya que en ocasiones implica consultar más de un sistema.

Se han llevado a cabo esfuerzos por crear sistemas que conjuntan la información por grupos de instalaciones. Sin embargo, todos están enfocados a los datos por instalación, lo cual es una limitación derivado de que no permiten visualizar trabajos de mantenimiento y el restablecimiento de fallas en situaciones que involucran más de una instalación simultánea.

Al estar enfocados en una sola instalación, no se puede apreciar la cargabilidad (véase Anexo A. Glosario) de líneas de transmisión (véase Anexo A. Glosario). Esto conlleva a que es difícil observar el panorama general de los disturbios en el Sistema Eléctrico Nacional (SEN), así como tampoco se puede identificar las problemáticas en la calidad de medición de energía en las líneas de transmisión.

Al no tener un sistema en tiempo real que permita los puntos anteriores, se dificulta la toma de decisiones del personal directivo y de

mantenimiento de forma oportuna, ya que al no tener la información actualizada, no se logra un sólido análisis de eventos en el SEN ni la reasignación de prioridades de mantenimiento preventivo y correctivo.

Se cuenta con un servidor disponible para poder implantar un sistema que permita abatir dichas limitaciones. Sin embargo, su capacidad de almacenamiento es de 80 gigabytes (GB), por lo cual no se puede mantener localmente los históricos. Así mismo, no cuenta con licencias como Windows, Office ni de algún software para adquisición de datos SCADA.

Algunos sistemas SCADA existentes en las instalaciones tienen su capacidad de comunicación al límite. Existen otros que únicamente cuentan con puertos de comunicación serial, por lo cual se requeriría la compra de convertidores de medios para adquirir sus datos vía remota.

Cabe señalar que configurar un solo sistema SCADA implica la utilización de muchos recursos: tiempo, personal, de equipo y administrativo. Así mismo, un sistema SCADA requiere dejarlo fuera de servicio para cargar una nueva configuración, por lo que se necesita la autorización del Centro Nacional de Control de Energía (CENACE) quien determina si los trabajos son factibles o se deben reprogramar derivado de las condiciones del SEN.

JUSTIFICACIÓN

Este proyecto permitirá visualizar los flujos de energía eléctrica en tiempo real del área central de país, permitiendo la oportuna toma de decisiones del personal directivo y de mantenimiento ante la operación normal del SEN y bajo situaciones de contingencia; también habilitará el acceso a información específica de cada instalación útil ante algún disturbio eléctrico. Así mismo, se tendría la referencia de todos los sistemas existentes para consulta de información.

HIPÓTESIS

Se desarrollará un sistema de monitoreo de datos en tiempo real de las subestaciones eléctricas del área central del país, que permitirá dotar al personal directivo y de mantenimiento de la información necesaria para la toma de decisiones en condiciones normales, durante una falla en el Sistema Eléctrico de Potencia y en las acciones de mitigación y restablecimiento de dichas fallas, a través de la visualización de estado de interruptores y mediciones de los flujos de energía eléctrica.

OBJETIVO GENERAL

Desarrollar un sistema que concentre los datos del estado operativo de las subestaciones monitoreadas por los sistemas SCADAWeb (véase Anexo A. Glosario) y WebSICLE (véase Anexo A. Glosario).

OBJETIVOS ESPECÍFICOS

1. Utilizar una metodología para el desarrollo del sistema.
2. Separar el sistema en dos subsistemas que permitan la adquisición y visualización de datos de manera independiente.
3. Definir una plataforma libre que no requiera de licencias de software.
4. Definir tecnologías estándar para el subsistema de visualización que permitirá al usuario desplegar la información en cualquier dispositivo.
5. Establecer un patrón de ejecución en el subsistema de adquisición que permita la tolerancia a fallas.
6. Agrupar en bloques independientes la adquisición de datos de las instalaciones.
7. Habilitar el acceso a la información específica de cada instalación a través del subsistema de visualización.
8. Almacenar los datos adquiridos del estado operativo de las subestaciones en una base de datos.

CAPÍTULO 1

ESTADO DEL ARTE

CAPÍTULO 1. ESTADO DEL ARTE

En México, el sector eléctrico es de propiedad federal por la CFE (véase la Figura 1.1), la cual es una empresa paraestatal que fue creada el 14 de agosto de 1937, y es la encargada de controlar, generar, transmitir y comercializar energía eléctrica en todo el territorio mexicano.



Figura 1.1. Comisión Federal de Electricidad

La CFE está constituida por una Dirección General, de la que dependen la Dirección de Operación, otras cuatro Direcciones y otras siete estructuras de la organización que ven desde finanzas, administración,

planeación de proyectos de modernización, entre otros, tal como se muestra en la Figura 1.2.

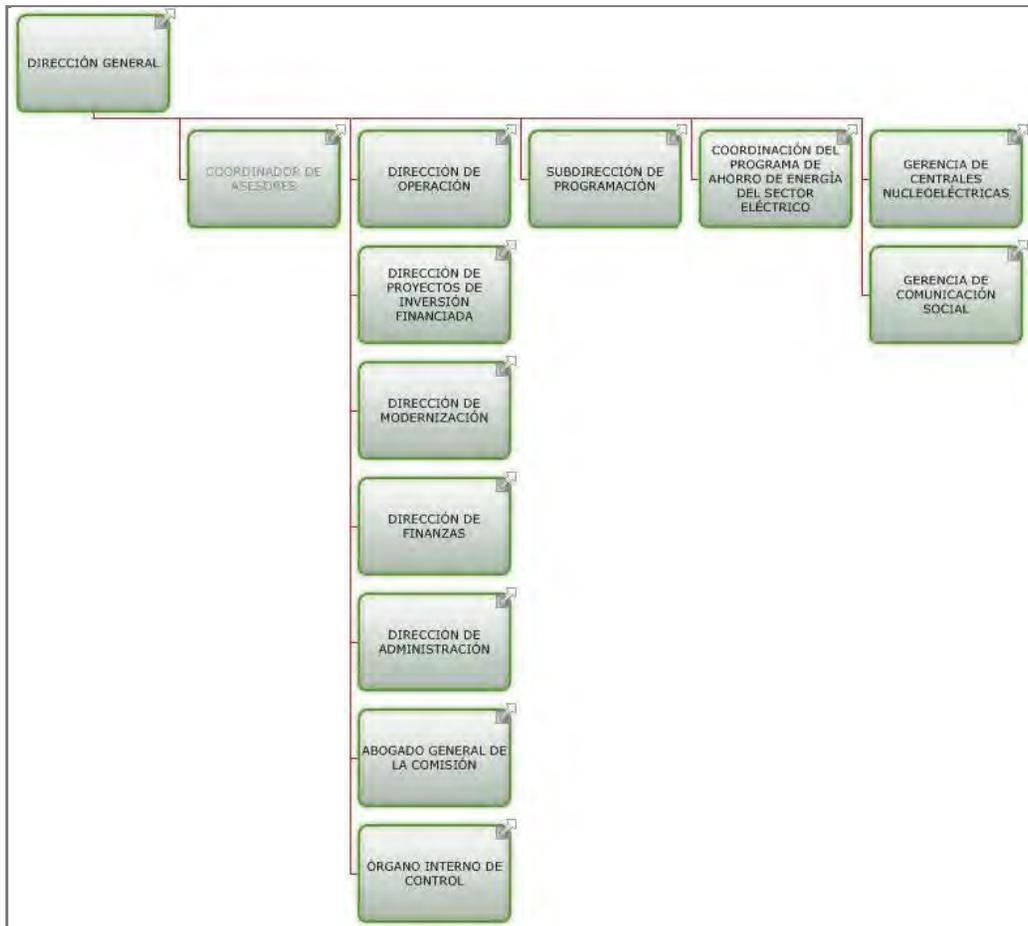


Figura 1.2. Organigrama de la Dirección General de CFE

Dentro de la Dirección de Operación, tal como se muestra en la Figura 1.3, están la Subdirección de Generación, la Subdirección de Transmisión, la Subdirección de Distribución y la Subdirección del CENACE, las cuales son las responsables de los diferentes procesos técnicos operativos encargados de cumplir el objetivo de CFE.

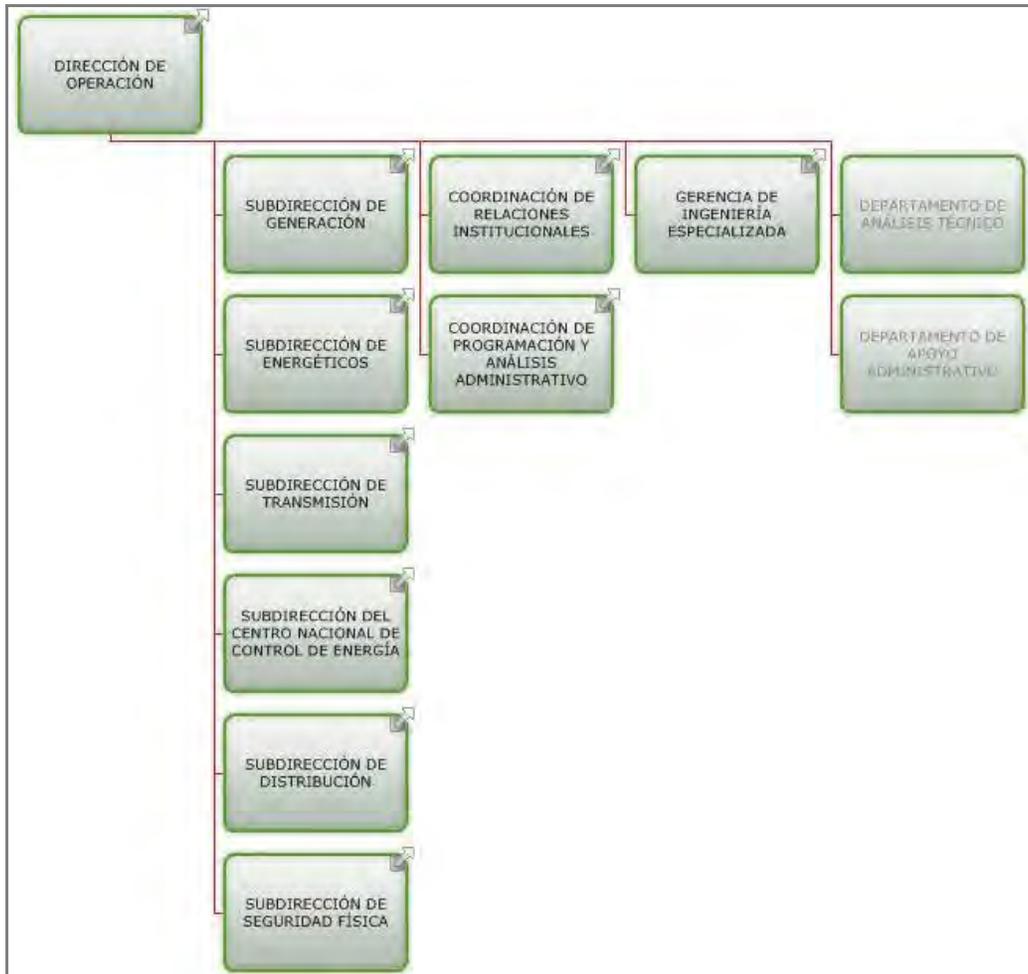


Figura 1.3. Organigrama de la Dirección de Operación de CFE

Para entender cómo es que la CFE cumple con su objetivo de suministrar energía eléctrica, en la Figura 1.4 se da a conocer las funciones de dichas Subdirecciones mostrándolas de forma resumida de izquierda a derecha como: (1) la energía generada en las diferentes centrales generadoras, tales como parques eólicos, hidroeléctricas y termoeléctricas, (2) viaja a través de la red de transporte hasta las subestaciones de transmisión, (3) entonces pasa a la red de distribución hasta las subestaciones de distribución, de donde entonces es entregada al consumo doméstico e

industrial, considerando que (4) el centro de control realiza todas las acciones necesarias para que la energía fluya del punto 1 al punto 3.

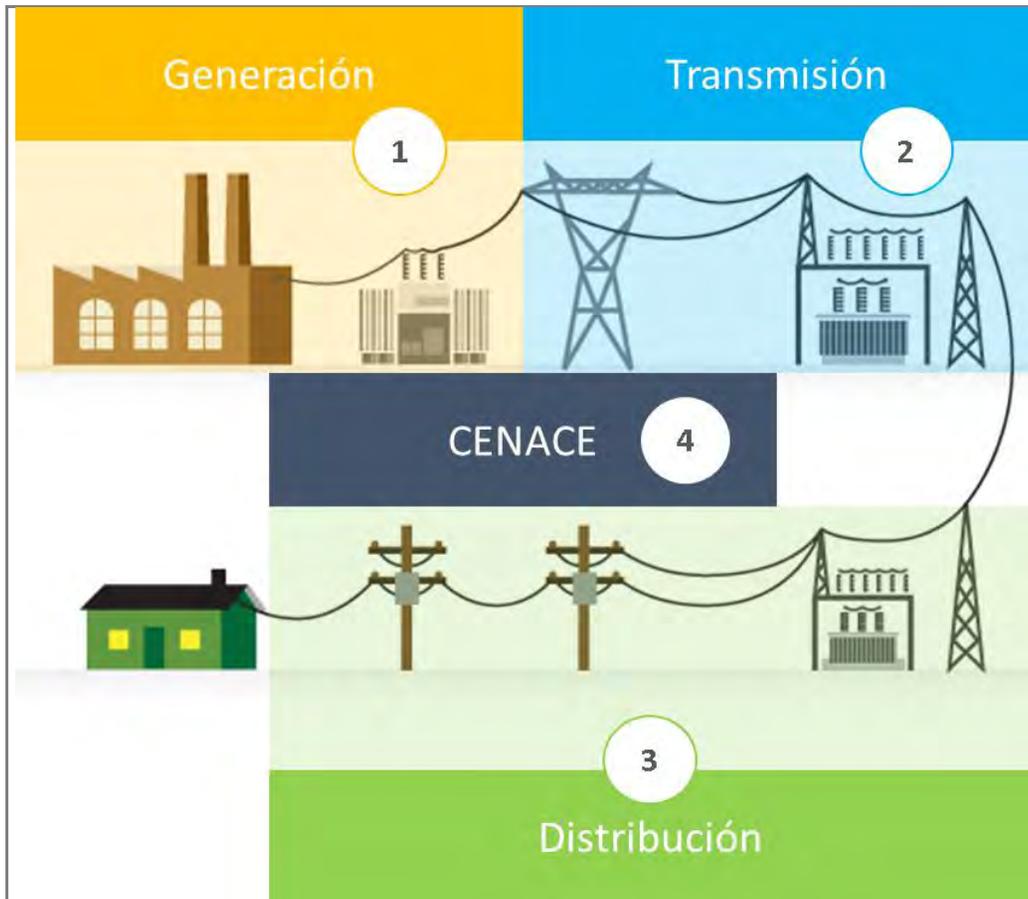


Figura 1.4. Proceso de Generación, Transmisión, Distribución y Control de Energía

1.1. Subdirección de Generación

La Subdirección de Generación es la encargada de las funciones de generación (véase Anexo A. Glosario) de energía eléctrica. Esto lo logra mediante las diversas centrales generadoras distribuidas a todo lo largo y ancho del país. En la Figura 1.5 se observa una central generadora.



Figura 1.5. Central generadora

Para producir energía eléctrica, se utilizan generadores eléctricos en las centrales geotérmicas, hidroeléctricas, termoeléctricas, eólicas y nucleares. En la Figura 1.6 se muestra un ejemplo de generadores eléctricos.

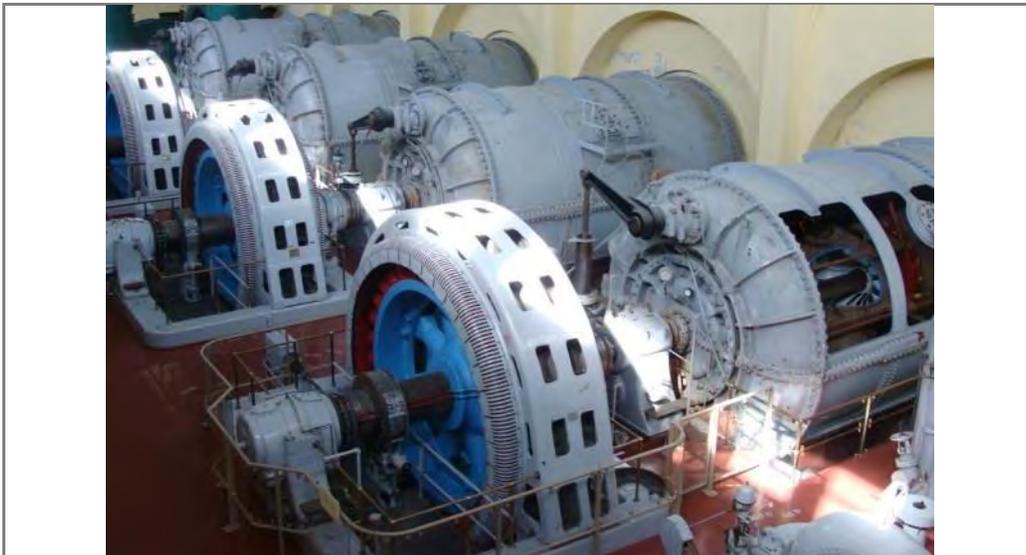


Figura 1.6. Generadores de una central hidroeléctrica

Tal como se muestra en la Figura 1.7, un generador es un dispositivo compuesto por una turbina y un alternador, el cual es capaz de mantener una diferencia de potencial eléctrico transformando la energía mecánica en eléctrica, la cual es proporcionada por la turbina. Esta transformación se consigue por la acción de un campo magnético sobre los conductores eléctricos dispuestos sobre el estator. Basándose en la Ley de Faraday, cuando el rotor a través de la turbina produce mecánicamente un movimiento relativo entre los conductores y el campo, se generará una fuerza electromotriz. (Maxwell, 1881).

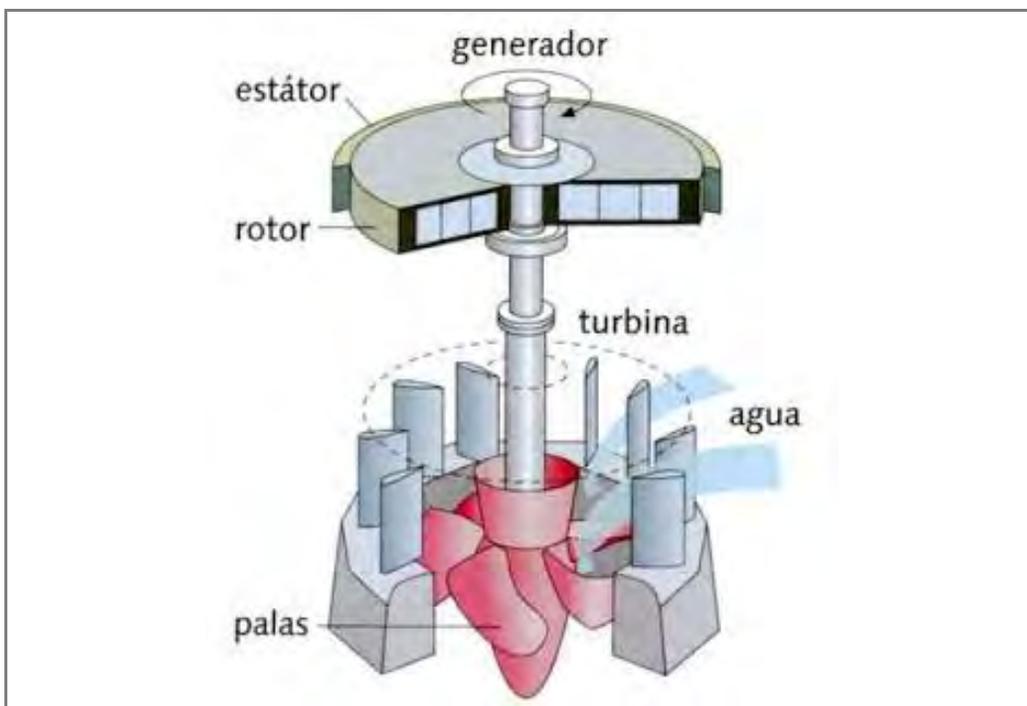


Figura 1.7. Generador eléctrico

1.2. Subdirección de Transmisión

Derivado de que las centrales generadoras están retiradas a cientos o miles de kilómetros de los grupos de consumo, la energía eléctrica producida se transforma en niveles de tensión mayores a 69 kV, aunque generalmente se utiliza voltajes de 400 ó 230 kV (véase Anexo A. Glosario), reduciendo así las pérdidas durante el proceso de transmisión de la electricidad.

Para la transmisión de la energía eléctrica se utilizan las líneas de transmisión (véase Anexo A. Glosario), las cuales son un conjunto de conductores que transportan la electricidad y están sostenidos por una serie de estructuras, tal como se observa en la Figura 1.8.



Figura 1.8. Líneas de transmisión eléctrica

Éstas líneas de transmisión se interconectan por medio de subestaciones eléctricas (véase Anexo A. Glosario), las cuales permiten controlar los flujos de energía mediante la operación de interruptores de potencia.

La función de los interruptores (véase Anexo A. Glosario) es permitir o interrumpir el paso de la energía eléctrica, de manera similar a los apagadores de una casa para encender o apagar las luces.

En la Figura 1.9, se muestra una subestación, a la cual la energía eléctrica que llega es conducida a unos buses, los cuales concentran la electricidad que se mandará a otras líneas de transmisión y a los bancos de transformación (véase Anexo A. Glosario).



Figura 1.9. Subestación eléctrica

Un transformador de potencia como el que se muestra en la Figura 1.10 en color amarillo, permite bajar el nivel de voltaje para que la electricidad viaje por redes eléctricas de menor extensión.



Figura 1.10. Transformador de potencia

Ya que los niveles usados para la transmisión de la energía eléctrica son muy elevados para disminuir las pérdidas, es necesario disminuir dichos niveles para acercar la electricidad a los clientes.

En la zona del Valle de México y área metropolitana, por ejemplo, que comprende las entidades federativas de Distrito Federal, Estado de México, Morelos, Hidalgo y Tlaxcala, se puede representar el proceso de transmisión de la energía eléctrica en sus diferentes niveles de tensión antes de entregar la electricidad para su distribución (véase Anexo A. Glosario) a los clientes y usuarios finales. En la Figura 1.11 se muestra con colores los diferentes niveles de tensión, transformados a niveles más bajos hasta llegar a la Red y Circuitos de Distribución.

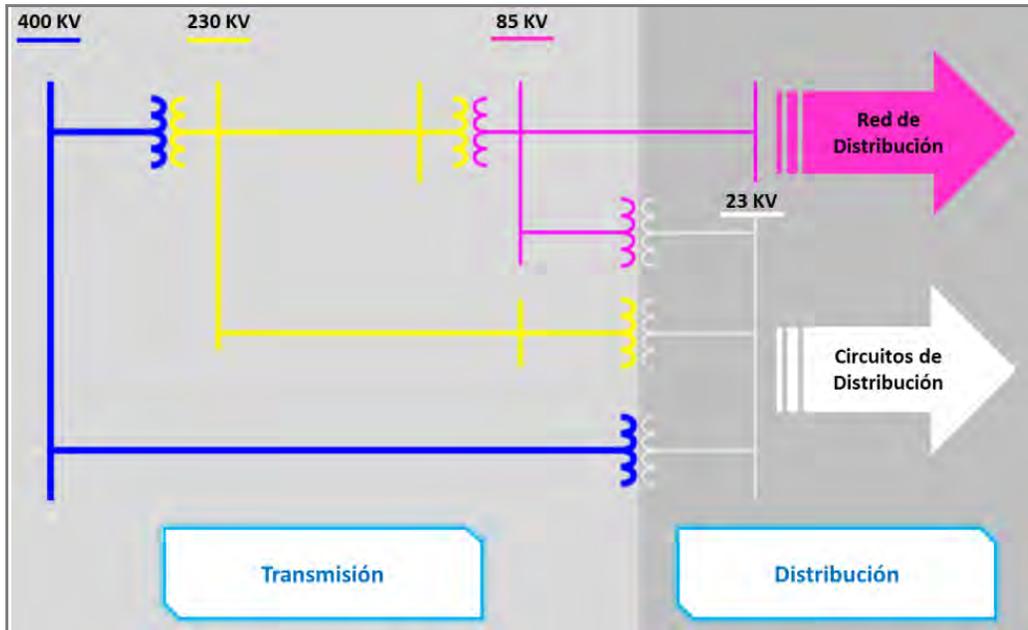


Figura 1.11. Proceso de Transmisión y Distribución en el Valle de México y área metropolitana

1.3. Subdirección de Distribución

Una vez que la energía eléctrica pasa a niveles de tensión menor, es enviada por las redes de distribución menores a 69 kV, las cuales son de áreas geográficas pequeñas. Las líneas de distribución, también conocidas como de subtransmisión, llegan a subestaciones eléctricas las cuales transforman nuevamente el voltaje a 23 ó 13.8 kV, siendo en dichos niveles más seguro llevar la electricidad a los clientes.

En la Figura 1.12 se muestra una subestación de 23 kV, mucho más pequeña que la mostrada en la Figura 1.9.



Figura 1.12. Subestación eléctrica de 23 kV

De las subestaciones de distribución, salen los circuitos de 23 kV ó 13.8 kV que viajan por las calles a través de postes o por cable subterráneo de aislamiento seco (Secretaría de Energía, 2012), para que pueda mandarse la energía eléctrica de forma segura para el usuario. Una vez por la calle, se distinguen porque se utilizan tres cables horizontales como se ve en la Figura 1.13.



Figura 1.13. Circuito de 13.8 kV por postes

De estos circuitos, la energía eléctrica se transforma nuevamente mediante transformadores de poste en los puntos de entrega final a los clientes. De éstos salen cuatro cables verticales, los cuales se usan como las fases A, B y C en 127 volts, voltaje utilizado en México, y el cuarto cable es usado como neutro, tal como se muestra en la Figura 1.14.

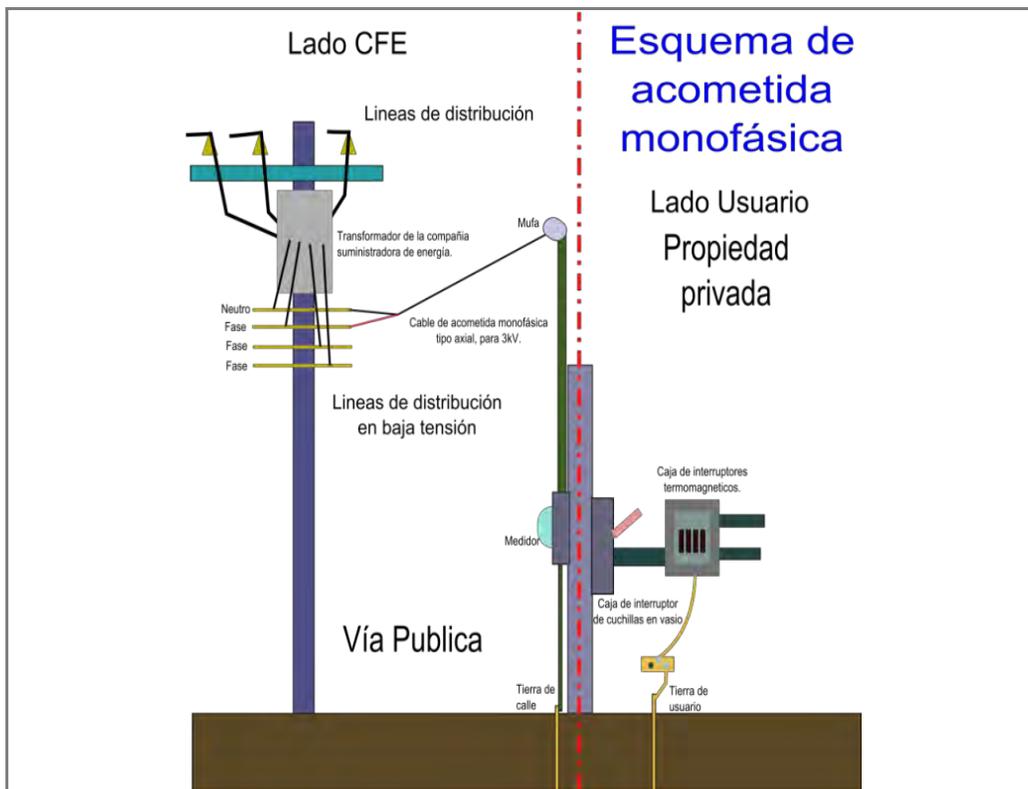


Figura 1.14. Entrega de energía de Distribución al Usuario

Para llegar al usuario, se tiende un cable de una de estas fases y otro del neutro, los cuales llegan a un medidor de consumo para la cobranza, después a un centro de carga con fusibles o interruptores termomagnéticos, y hasta entonces, a los aparatos eléctricos e iluminación que todos tenemos.

1.4. Subdirección del CENACE

Ahora bien, para garantizar que la electricidad se suministre a todos los usuarios, es necesario monitorear permanentemente todas las variables del SEN, es decir, vigilar el correcto funcionamiento de cada una de las unidades generadoras en las centrales, la cantidad de energía que sale de las mismas y seguir su flujo por las líneas de transmisión teniendo en cuenta su cargabilidad, así como todas las subestaciones eléctricas a las que llega, considerando en éstas el estado de los interruptores, cuchillas (véase Anexo A. Glosario), bancos de transformación y demás equipo que la componen. Una vez saliendo de las subestaciones eléctricas, se continúa vigilando como se distribuye la energía a todos los centros de consumo, pequeños como pueblos o grandes como las principales ciudades del país, hasta llegar a los circuitos que viajan por las calles en postes.

La función del CENACE es precisamente la de vigilar que toda la demanda de energía (distribución) sea satisfecha produciendo la cantidad de energía exacta en cada instante (generación) y que sea transmitida entre ambos puntos (transmisión). Esto debe suceder a pesar de las contingencias que se puedan presentar derivadas de mantenimiento a equipo que se requiere desenergizado, fallas provocadas por obsolescencia, robo de estructuras y cable, así como por fenómenos ambientales, por ejemplo, descargas atmosféricas, nevadas, ciclones, etc.

1.5. Unidad Terminal Maestra

Para que el CENACE pueda vigilar todo el Sistema Eléctrico Nacional, utiliza equipos llamados Unidad Terminal Maestra (véase Anexo A. Glosario), también conocida como UTM o simplemente Maestra, la cual contiene una base de datos de todas las variables eléctricas y estados operativos de las subestaciones, los cuales son mostrados en videomuros y consolas a los operadores, quienes son los encargados de vigilar y avisar cualquier anomalía que visualicen en sus sistemas.

En la Figura 1.15 se muestra las pantallas de visualización de un Centro de Control con las redes troncales de 400, 230, 115 y 85 kV del área central del país.



Figura 1.15. Fotografía del Centro de Control del Área de Control Central

1.6. Enviando la Información al CENACE

Para que el CENACE pueda monitorear en tiempo real como fluye la energía a través de las líneas de transmisión y de las subestaciones, es necesario situarse en una instalación y describir de qué equipos está conformada. A continuación se explica cómo se lleva a cabo en el Proceso de Transmisión.

En primera instancia está el Equipo Primario, que generalmente está a la intemperie y ocupa grandes cantidades de espacio, ya que se debe respetar la distancia de seguridad para evitar que se drene a tierra la energía y dañe al propio equipo, estructuras cercanas y personal. Dicha distancia de seguridad es el espacio entre el elemento energizado y una persona u objeto y que éste último puede ser alcanzado por una descarga eléctrica o arco eléctrico, y se basa en que por cada kV que tiene la energía, idealmente se debe guardar 1 cm. Es decir, que para una línea de transmisión de 400 kV, a los 400 cm o 4 m de distancia se produce un arco eléctrico, variando dicha distancia por las condiciones de humedad, polvo y contaminación del entorno. Para mayor información véase (CFE, Reglamento de Seguridad e Higiene en el trabajo, Capítulo 800, 2011).



Figura 1.16. Arco eléctrico producido al cerrar cuchillas de 230 kV con potencial durante maniobra normal

Como se visualiza en la Figura 1.16, el equipo primario es lo que vemos en una subestación eléctrica cuando pasamos cerca, y se enlista a continuación en la Tabla 1.1.

Tabla 1.1. Listado de Equipo Primario de una subestación

| | |
|-------------------------|---------------------------------|
| Interruptor | Transformador de Corriente (TC) |
| Cuchilla | Transformador de Potencial (TP) |
| Banco de transformación | Apartarrayos |

Existe otra clase de equipo denominado Equipo Auxiliar y descrito más adelante en la Tabla 1.2, el cual también es necesario monitorear en tiempo real. Dicho equipo está instalado en la Caseta de Control, la cual tiene canalización desde todo el equipo primario para que por medio de cable de control se conecten todas las señales de alarma, indicación, mandos o medición.

Tabla 1.2. Listado de Equipo Auxiliar de una subestación

| | | |
|----------------------|---|---------------------------------|
| Equipo de protección | Unidad Terminal Remota (UTR) | Consola de Control |
| Medidor multifunción | Modulo de Control y Adquisición de Datos (MCAD) | Global Positioning System (GPS) |
| Servidores SCADA | Switch | Ruteador |

Adicionalmente a las alarmas e indicaciones del equipo primario y equipo auxiliar, también es necesario explotar las variables eléctricas descritas en la Tabla 1.3 que se calculan para ser enviadas en tiempo real a los diferentes despachos de energía del CENACE, ya que permiten determinar las condiciones de operación del SEN, en cuanto a generación, suministro y calidad de la energía.

Tabla 1.3. Variables eléctricas monitoreadas por el CENACE

| Variable | Uso en |
|-------------------|--|
| Voltaje | Unidades generadoras, líneas de transmisión, etc. |
| Corriente | Circuitos de distribución. |
| Potencia aparente | Capacidad en bancos de transformación. |
| Potencia efectiva | Unidades generadoras, líneas de transmisión y bancos de transformación. |
| Potencia reactiva | Compensador Estático de VARs (CEV), bancos de capacitores y reactores |
| Frecuencia | Unidades generadoras, líneas de transmisión, buses de subestaciones y circuitos. |
| Posición de TAP | Bancos de transformación. |

Una vez que conocemos por cuales equipos está conformada una subestación eléctrica, llegó el momento de explicar cómo se realiza su monitoreo.

Los interruptores, cuchillas y transformadores cuentan con gabinetes de alarmas, como los mostrados en la Figura 1.17, los cuales reciben alimentación de 127 VCA para funcionamiento de motores y resistencias calefactoras que evitan la condensación de la humedad en el interior, así como 125 VCD para alimentación de los sensores, relés de contacto seco (véase Anexo A. Glosario), circuitos de control, etc.

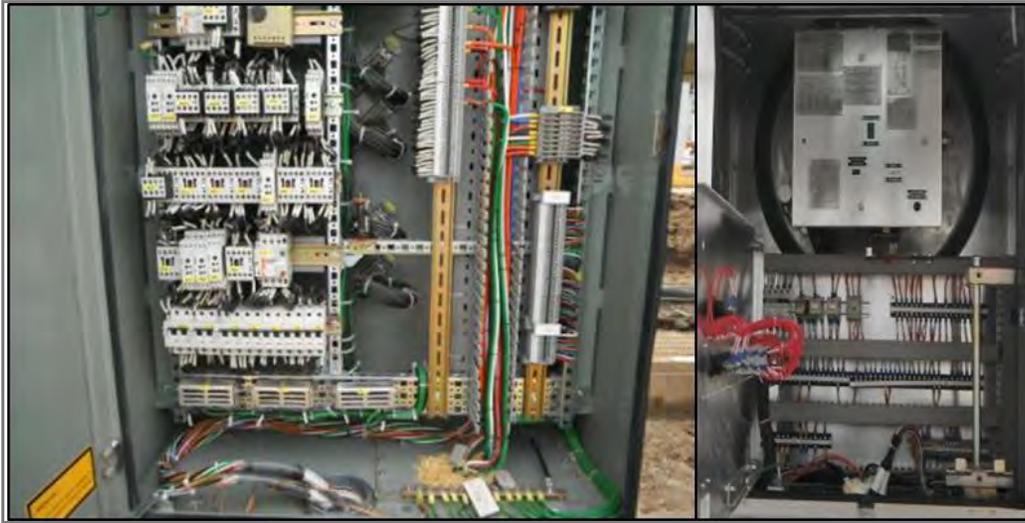


Figura 1.17. Gabinetes de alarmas de un interruptor de potencia

Estos gabinetes también cuentan con tablillas de conexión como las mostradas la Figura 1.18, las cuales permiten conectar el cable de control que se utiliza para integrar las alarmas e indicaciones a los equipos SCADA.



Figura 1.18. Tablilla de conexiones

1.7. Supervisory Control And Data Adquisition (SCADA)

Un equipo SCADA es un dispositivo o conjunto de dispositivos encargados de monitorear y controlar procesos industriales. El Proceso de Transmisión cuenta con dos arquitecturas SCADA en las subestaciones:

1. Unidad Terminal Remota (UTR)

En la Figura 1.19 se muestra la arquitectura de una UTR.

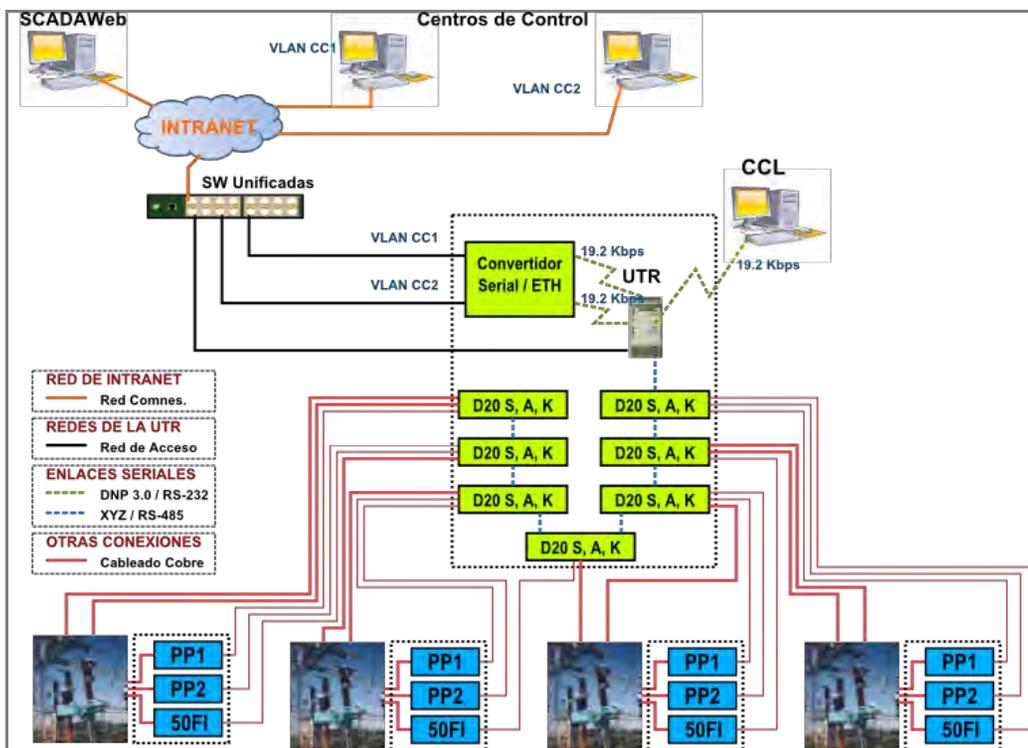


Figura 1.19. Diagrama de una UTR

Una UTR está constituida por una unidad central de procesamiento encargada de comunicarse con sus módulos de

adquisición y control, los cuales digitalizan las señales físicas, entonces la unidad central de procesamiento conjunta los datos adquiridos y los almacena en diferentes tipos de bases de datos, disponibles para la Consola de Control de la subestación, así como los diferentes despachos de energía que se establezca para solicitar información y telecontrolar la instalación.

2. Sistema de Información y Control Local de Estación (SICLE)

Está constituido por un conjunto de equipos MCAD, los cuales realizan el control y la adquisición de alarmas, indicaciones y medición del equipo primario en una zona específica de la subestación. Dichos MCAD están conectados a una Red de Datos aislada por medio de la cual las Consolas de Control y Servidores SCADA adquieren de manera independiente los datos de los MCAD, de forma tal que en caso de que alguno de estos equipos falle, los demás continúan teniendo el control de la subestación. Así mismo, los Servidores SCADA mantienen en diferentes tipos de bases de datos la información de la instalación, y la tienen disponible para los diferentes despachos de energía a los cuales se establezca que solicitarán datos para el monitoreo y telecontrol de la subestación. Un diagrama simplificado de un SICLE se muestra en la Figura 1.20.

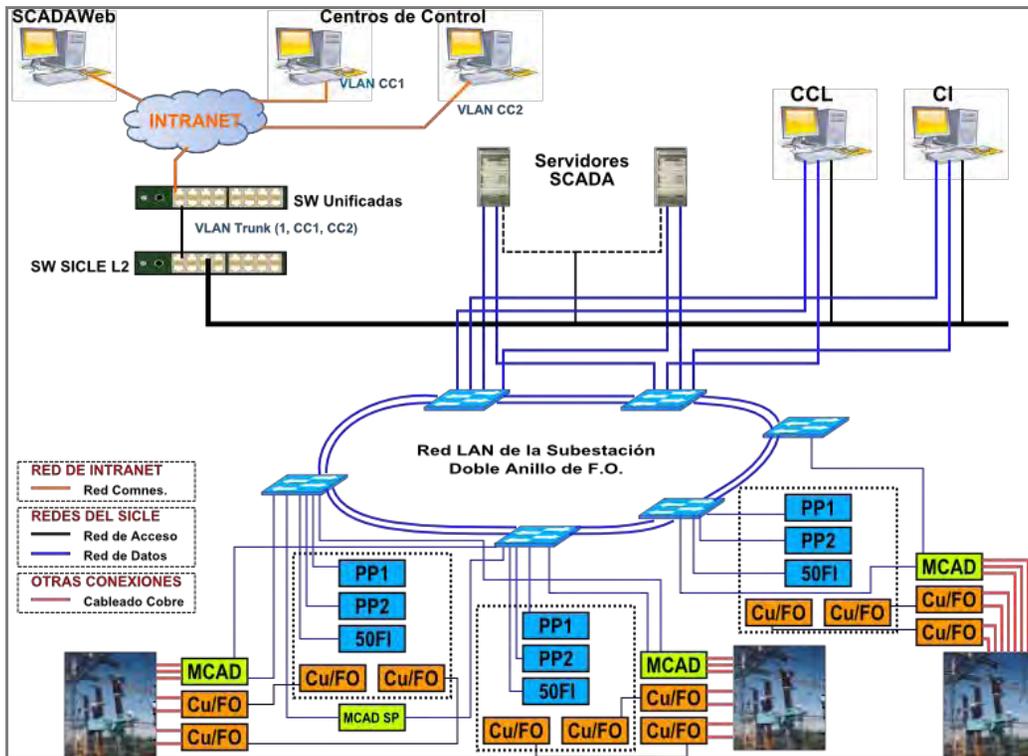


Figura 1.20. Diagrama de un SICLE

Dependiendo de la tecnología y capacidad utilizada de las Maestras de los Centros de Control o despachos establecidos, se utilizan canales seriales por modem (punto a punto) o canales por las redes unificadas dentro de la Intranet de CFE.

Independientemente del tipo de canal utilizado, la Maestra realiza el monitoreo de las subestaciones preguntando a cada sistema SCADA por las bases de datos de entradas digitales y de entradas analógicas (véase Anexo A. Glosario), llevándose de esta forma una fotografía del estado operativo de toda la instalación. Esto es conocido como poleo o pregunta de integridad, y se lleva a cabo de forma periódica para garantizar que la información que tiene la Maestra está actualizada. Dependiendo de la

tecnología de la Maestra por parte del CENACE, así como el tamaño de las bases de datos, de la capacidad de los canales de comunicación y la tecnología del sistema SCADA en la subestación, el administrador de la Maestra configura el mejor tiempo en que éste proceso se llevará a cabo, por lo que esta pregunta de integridad puede llevarse desde cada 30 segundos a cada hora.

Para optimizar los recursos utilizados durante el monitoreo de una subestación, existen otras dos bases de datos de tamaño variable, las cuales contienen únicamente los cambios de valor en las entradas digitales y en las entradas analógicas, conocidas también como eventos, históricos o secuencia de eventos (SOE, del inglés sequence of events) para los cambios de digitales, y cambios de medición para las analógicas. Debido a su naturaleza de contener sólo cambios, los sistemas SCADA en las subestaciones vacían estas bases de datos cuando la Maestra confirma que ha recibido toda la información. Así, cuando la Maestra pregunta nuevamente por cambios, están sólo aquellos generados desde que se eliminaron los eventos anteriores.

De esta forma, al tratarse de una menor cantidad de datos y para que se lleve a cabo un monitoreo en tiempo real, los administradores de la Maestra la configuran para realizar preguntas por cambios cada 50 ms (muestreo programado en la Maestra del Área de Control Central para equipo SCADA comunicado por redes unificadas).

Para garantizar que los datos enviados por los equipos SCADA de las subestaciones son recibidos correctamente por las Maestras, se utiliza principalmente el protocolo de comunicación SCADA llamado Distributed

Network Protocol versión 3 (DNP3) el cual puede ser utilizado en canales de comunicación serial o por canales de redes unificadas, dependiendo de la tecnología de la UTR o SICLE de la instalación, la tecnología y capacidad utilizada de la Maestra, y los requerimientos del Centro de Control.

Este protocolo es el preferido en el Proceso de Transmisión, ya que también puede ser usado de forma confiable para la comunicación entre los MCAD, Servidores SCADA y Consolas de Control, ya que al tratarse de subestaciones eléctricas, la contaminación electromagnética inducida en los cables de cobre puede provocar alteraciones en los datos digitales transmitidos. Esto se logra porque el protocolo separa los datos enviados en bloques, y agrega un código Comprobación de Redundancia Cíclica (CRC) que permite verificar que cada bloque haya sido recibido sin alteraciones, y en caso de alguna discrepancia, la información se retransmite. Adicionalmente, al ser un estándar internacional, permite la interoperabilidad de los equipos de diferentes marcas y modelos.

1.8. Una Maestra para el Proceso de Transmisión

Con la finalidad de que el Proceso de Transmisión pudiera monitorear también el estado operativo de sus subestaciones, se iniciaron dos proyectos independientes para desarrollar una Maestra propia de la CFE que realizará la adquisición de datos de los sistemas SCADA de las subestaciones utilizando el protocolo DNP3, y la información concentrada se ofreciera por servicios Web para ser visualizada en navegadores Web, tales como Internet Explorer, Firefox, Chrome, etc. Dichos servicios,

como se muestra en la Figura 1.21, son diagramas unifilares, tendencias y registro de eventos en línea.



Figura 1.21. Servicios Web de los sistemas de monitoreo en línea

Estos dos proyectos fueron nombrados SCADAWeb y WebSICLE (véase Anexo A. Glosario) siendo el primero de ellos el más usado en diferentes regiones del país. Su utilidad se vio reflejada casi de inmediato al permitir al personal de mantenimiento en las subestaciones poder contar con información del estado de cualquier subestación integrada al sistema, así como poder consultar las secuencias de eventos en caso de falla, ahorrando de esta forma mucho tiempo ya que anteriormente se tenía que trasladar personal a la instalación para obtener el histórico de los eventos, y solo hasta entonces poder empezar a realizar un análisis de lo sucedido para poder tomar decisiones correctivas y de restablecimiento.

1.9. Área de Trabajo

Hasta este punto se ha descrito como se estructura todo el proceso de la energía eléctrica, desde su generación hasta su consumo doméstico o industrial.



Figura 1.22. Sector Sur de la Zona de Transmisión Metropolitana

Actualmente el Sector Sur, mismo que se muestra en la Figura 1.22 el área geográfica que abarca, pertenece a la Zona de Transmisión Metropolitana, la cual está encargada de todas las subestaciones con voltajes de 230 kV y 400 kV de lo que fue Luz y Fuerza del Centro. Ésta Zona depende administrativamente de la Gerencia Regional de Transmisión Central (GRTC), la cual cuenta con un total de ocho Zonas. Ésta Gerencia es una de nueve Gerencias Regionales de Transmisión, tal como se muestra en la Figura 1.23, las cuales dependen de la Subdirección de Transmisión que se describió en el capítulo 1.2.



Figura 1.23. Mapa de las nueve Gerencias Regionales de Transmisión

El Sector Sur conformado por siete subestaciones, es un área operativa dedicada a la modernización, mantenimiento y atención a fallas de las líneas de transmisión y de las subestaciones dentro del área geográfica asignada, así como de todo el equipo primario y auxiliar que éstas contienen.

Debido a la complejidad de los equipos que se atienden, el Sector Sur cuenta con cinco oficinas de especialidades técnicas, mismas que se describen en la Tabla 1.4.

Tabla 1.4. Especialidades técnicas en Transmisión

| Especialidad | Responsabilidad |
|---------------|--|
| Líneas | Líneas de transmisión y derechos de vía. |
| Subestaciones | Predio, estructuras y equipo primario. |

| | |
|-----------------------|---|
| Protección y Medición | Equipo de protección, cableado de medición, alarmas, señalización y mandos de equipo primario, medidores multifunción y pruebas a los TC y TP. |
| Control | Regulación de voltaje de bancos de transformación, sistemas de monitoreo, control y de enfriamiento de los CEV, cargadores de baterías de 48 y 125 VCD, servidores de gestión de activos, concentradores de información, UTR, GPS, consolas de control, inversores. En un SICLE: equipos MCAD, switches, distribuidores ópticos y la fibra óptica multimodo de la red de datos interna de la subestación (de acuerdo a la especificación CFE-G0000-34 Sistema de Información y Control Local de Estación del año 2010), transductores; así como la comunicación a las Maestras de los Centros de Control. |
| Comunicaciones | Switches, ruteadores y sistemas de transporte analógicos, digitales y de fibra óptica monomodo para servicios de redes unificadas. |

El cargo actual establecido dentro de la empresa es de Jefe de Oficina de Control del Sector Sur, lo que significa que de acuerdo a la Tabla 1.4 se es él directamente responsable de todo el equipo citado en el punto de Control, del personal sindicalizado y no sindicalizado, vehículos, bodegas, equipos de prueba, herramienta y material asignados a la especialidad en dicho sector. La responsabilidad administrativa va desde la planeación, coordinación y ejecución de proyectos de modernización, hasta las actividades de mantenimiento y apoyos multidisciplinarios, así como la atención a fallas de equipo propio de la especialidad y apoyo en fallas de equipo de las demás especialidades.

CAPÍTULO 2

CASO DE ESTUDIO

CAPÍTULO 2. CASO DE ESTUDIO

2.1. Sistemas Actuales

La información SCADA de las subestaciones eléctricas actualmente se concentra en los sistemas distribuidos SCADAWeb y WebSICLE, los cuales están instalados en servidores independientes distribuidos en diferentes áreas de responsabilidad.

El sistema WebSICLE se encuentra en las oficinas de la Gerencia Regional de Transmisión Oriente en el estado de Veracruz.

El sistema SCADAWeb es usado en tres servidores independientes por la Gerencia Regional de Transmisión Occidente en Zapopán, Jalisco, y en ocho sistemas distribuidos en la Gerencia Regional de Transmisión Central en las ciudades de México, Colorines, Acapulco, Cuernavaca, Puebla, Texcoco, Querétaro y Tula.

2.2. Cómo Funcionan

Ambos sistemas funcionan de forma similar, ya que ambos son una Maestra que explota la información de los equipos SCADA de las subestaciones eléctricas de un área geográfica determinada por medio del protocolo DNP3 a través de la red interna de CFE.

La información de las subestaciones es procesada, validada y almacenada en bases de datos relacionales que mantienen los datos

actualizados en tiempo real, básicamente en tres tablas de la base de datos, como se muestra a continuación en la Tabla 2.1.

Tabla 2.1. Principales tablas en bases de datos del SCADAWeb y WebSICLE

| Tipo de tabla | Uso |
|----------------------|---|
| Estados digitales | Se utiliza para estado de interruptores de potencia, cuchillas, alarmas e indicaciones de equipo primario o auxiliar. |
| Estados analógicos | Mediciones de las diversas variables eléctricas, descritas en la Tabla 1.3. |
| Secuencia de eventos | Usada para el análisis de algún evento, tal como falla de equipo, accidentes, atentados, problemáticas, maniobras, etc. |

Cada sistema cuenta con su propio cliente, el cual es una aplicación Web hospedada en un servidor HTTP, la cual al ser accedida muestra una interfaz de usuario, y de forma organizada visualiza los datos en diagramas unifilares (véase Anexo A. Glosario), tabuladores, gráficas de tendencia, etc. Esta información es extraída mediante consultas SQL (véase Anexo A. Glosario), predeterminadas y estructurada en formato XML (véase Anexo A. Glosario) desde el servidor, y enviada de forma asíncrona al cliente.

2.3. Cliente SCADAWeb

El sistema SCADAWeb requiere de un inicio de sesión que se envía por método POST (véase Anexo A. Glosario) del protocolo HTTP, para entonces dar acceso a las diferentes secciones de información con que cuenta: Diagramas unifilares, Eventos, Estados y Mediciones

principalmente. En el caso muy particular de los diagramas unifilares, se envía al cliente una página Web, mostrada en la Figura 2.1, en la cual viene un objeto de Adobe Flash con el diagrama unifilar, y código Javascript embebido el cual se encarga de obtener de manera asíncrona y de forma separada los estados y las mediciones en formato XML para el cual el sistema no requiere de inicio de sesión, y posteriormente actualiza cada variable del objeto Flash.

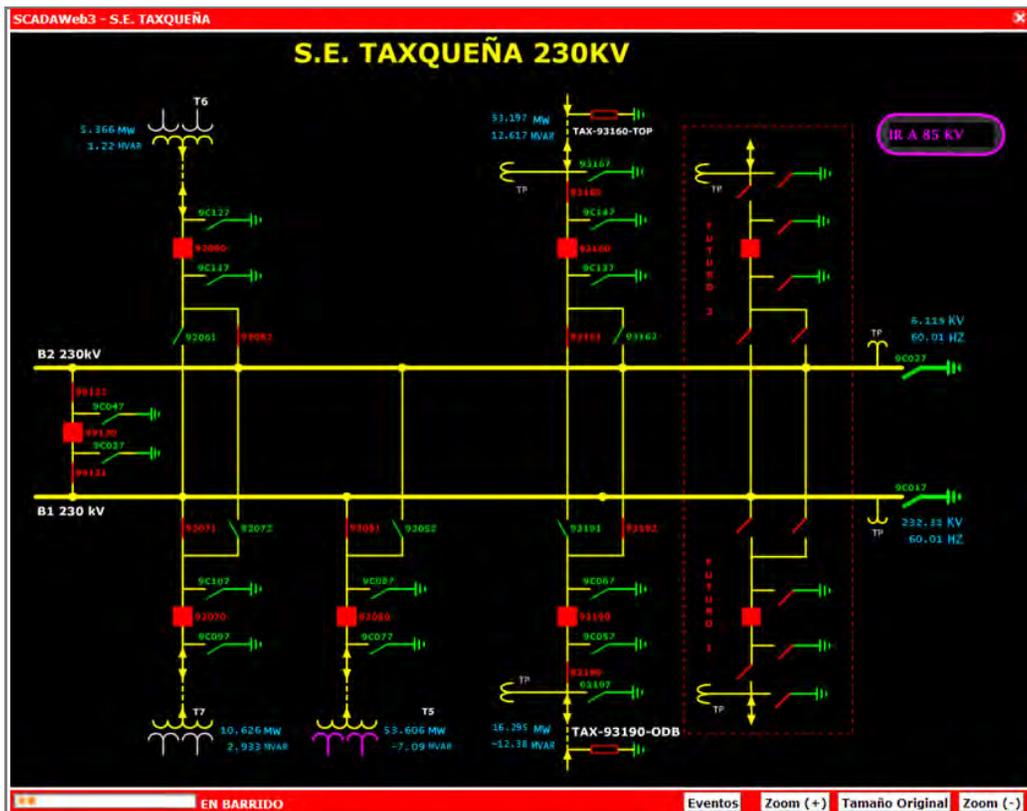


Figura 2.1. Cliente del sistema SCADAWeb

2.4. Cliente WebSICLE

El sistema WebSICLE, no requiere de inicio de sesión. Tiene un menú de selección de los diagramas unifilares disponibles, los cuales son enviados como una página Web que contiene imágenes ubicadas en posiciones relativas en pantalla y la referencia a un archivo Javascript específico para dicho diagrama, el cual contiene la lógica de estado y nombres de las partes dinámicas, así como también se encarga de la obtención asíncrona del archivo XML que contiene toda la información de la instalación.

2.5. Problemática Actual

➤ Falta de licenciamiento

El área de trabajo en donde se desarrollará el sistema, no cuenta con licencias disponibles de ningún tipo, como de algún sistema operativo para servidores ni para sistemas de administración de bases de datos.

➤ Almacenamiento local

En promedio la información de históricos consume 10 GB de almacenamiento mensuales teniendo únicamente 7 subestaciones monitoreadas (estadística de almacenamiento en el sistema SCADAWeb de la Subárea de Transmisión Puebla), y los servidores SCADAWeb de la Gerencia Regional de Transmisión Central, por ejemplo, supervisan 132 instalaciones. Por lo cual la capacidad de 80 GB del servidor disponible puede y será rebasada en menos de un mes.

➤ Actualización de información SCADA

Para mantener la información SCADA actualizada en tiempo real, se tiene que adquirir periódicamente cada archivo XML disponible en los sistemas SCADAWeb y WebSICLE, y ser procesado para actualizar y almacenar de forma continua los datos de las subestaciones. Estos datos son los que utilizará el cliente para actualizar su interfaz.

➤ Cliente multiplataforma

Debido al avance tecnológico, los usuarios no solo pueden acceder al sistema propuesto desde una computadora de escritorio o laptop, sino también desde su teléfono móvil. Esto conlleva a una variedad de sistemas operativos y navegadores Web con diferentes motores Javascript y tecnologías de gráficos vectoriales. Afortunadamente, dicho avance en los dispositivos conlleva también mayor poder de procesamiento, acceso a recursos de red y de visualización de gráficos.

Los sistemas SCADAWeb (mostrado en la Figura 2.2) usan tecnología Adobe Flash para los diagramas, la cuál no es compatible con dispositivos móviles.

El sistema WebSICLE utiliza una página HTML estática, cuyos elementos no se adaptan a las condiciones del cliente. Asimismo dicho sistema WebSICLE está diseñado para que sus usuarios solo puedan visualizar y tener datos actualizados en navegadores Web de Microsoft, Internet Explorer.



Figura 2.2. SCADAWeb como diagrama unifilar y secuencia de eventos

➤ Visualización de las redes troncales

Los sistemas actuales, SCADAWeb y WebSICLE, no cuentan con diagramas que muestren las redes troncales de 400 y 230 kV del área central de México, en que se visualice como se transporta la energía desde las grandes generadoras del occidente, oriente y sureste del país, hasta los diversos estados y ciudades responsabilidad de la GRTC.

CAPÍTULO 3

CONCEPTOS Y DEFINICIONES BÁSICAS

CAPÍTULO 3. CONCEPTOS Y DEFINICIONES BÁSICAS

Para entender cómo es qué se puede desarrollar un sistema para visualizar los flujos de energía del área central del país, es necesario asimilar algunos conceptos previos.

3.1. Bases de Datos

Una base de datos es una colección de datos organizados. Los datos son normalmente organizados para modelar aspectos relevantes de la realidad de forma que soporten procesos que requieran esta información.

3.1.1. DBMS – Database Management System

Es una aplicación de software diseñada especialmente para interactuar con el usuario, otras aplicaciones y la base de datos misma, para capturar y analizar datos.

Existen DMBS de propósito general el cual es un software diseñado para permitir la definición, creación, consulta, actualización y administración de bases de datos.

3.1.2. RDBMS – Relational Database Management System

Es un DBMS que se basa en el modelo relacional (Edgar F. Codd, en 1970 definió el término en su publicación "A Relational Model of Data for Large Shared Data Banks" (Codd, 1970)), en el cual se define básicamente que los datos: (1) son presentados como relaciones; y que

(2) para su manipulación se debe proveer operadores relacionales en forma tabular, es decir, como tablas.

Un ejemplo de RDBMS es el software MySQL, el cual es la opción más popular de bases de datos para uso en aplicaciones Web. Generalmente se encuentra en entornos integrados LAMP (Linux, Apache, MySQL and PHP (Gerner, Naramore, Owens, & Warden, 2006)) y WAMP (Windows, Apache, MySQL and PHP), los cuales son soluciones compuestas de software libre y open source, adecuado para construir sitios Web dinámicos de alta disponibilidad y uso intenso.

3.1.3. SQL - Structured Query Language



Es un lenguaje de programación de propósito especial diseñado para administrar datos contenidos en un RDBMS. Está definido bajo el estándar ISO/IEC 9075-1:2011 (ISO & IEC, ISO/IEC 9075-1:2011 SQL - - Part 1: Framework (SQL/Framework), 2011).

El uso más común en SQL es la consulta, la cual permite al usuario describir los datos deseados. Sin embargo, incluye un subconjunto denominado Data Manipulation Language (DML: (ISO & IEC, ISO/IEC 9075-1:2011 SQL -- Part 1: Framework (SQL/Framework), 2011) usado para agregar, eliminar y actualizar datos.

Existe otro subconjunto de SQL llamado Data Definition Language (DDL: (ISO & IEC, ISO/IEC 9075-1:2011 SQL -- Part 1: Framework (SQL/Framework), 2011), el cual consiste en una colección de verbos

imperativos cuyo efecto es modificar el esquema de una base de datos al agregar, modificar y eliminar tablas y otros objetos.

3.2. Servidor Web

Servidor Web (López-Quijado, 2007), es aquel equipo (hardware) o aplicación (software) cuya función principal es almacenar, procesar y entregar páginas Web a los clientes.

La comunicación toma lugar usando el protocolo HTTP, con el cual un agente de usuario, comúnmente conocido como navegador Web, inicia la comunicación haciendo una petición para un recurso específico, y el servidor responde con el contenido de ese recurso o un mensaje de error si no puede entregarlo.

Un recurso en un servidor Web se refiere a contenido estático como archivos HTML, imágenes, archivos de audio, videos, archivos CSS o Javascript, etc.

Aunque la función principal de un servidor Web es entregar contenido, gracias al protocolo HTTP también soporta recibir archivos de los clientes. Esta característica es usada a través de formularios que incluyen la opción de subir archivos.

Muchos servidores Web genéricos también soportan ejecución de aplicaciones en el servidor (server-side scripting), lo que permite separar el comportamiento del servidor en archivos separados, mientras que el software del servidor permanece intacto. Usualmente esta función se

utiliza para generar contenido dinámico “al momento”, ideal cuando se consulta y/o modifica información de bases de datos.

La ventaja principal de los servidores Web, es que el cliente no requiere de aplicaciones adicionales para poder acceder a un sitio Web, ya que todo equipo de cómputo y teléfonos inteligentes cuentan con uno.

3.2.1. HTTP – Hypertext Transfer Protocol

A small blue square icon containing the text "http://" in white.

El Hypertext Transfer Protocol, es un protocolo de capa de aplicación para sistemas de información hipermedia (medio de información no linear, el cual incluye gráficos, audio, video, texto plano e hipervínculos), distribuidos y colaborativos. HTTP es usado para toda la comunicación de datos de la World Wide Web (WWW o W3, comúnmente conocida como la Web).

Fue desarrollado por la Internet Engineering Task Force (IETF) y la World Wide Web Consortium (W3C), culminando en la publicación del estándar (RFC N. W., RFC-2616 Hypertext Transfer Protocol -- HTTP/1.1, 1999), siendo la versión de uso más común en la actualidad.

Los recursos son identificados y localizados en la red por las URI (RFC N. W., RFC-3986 Uniform Resource Identifier (URI) Generic Syntax, 2005). Estas se clasifican de acuerdo a la Tabla 3.1 que a continuación se muestra.

Tabla 3.1. Tipos de URI

| Tipo URI | Uso |
|----------|--|
| URL | Uniform Resource Locator. Referida frecuentemente como dirección Web, es usada para identificar la dirección en que un recurso se encuentra disponible. Una analogía sería la dirección donde vive una persona. |
| URN | Uniform Resource Name. Se utiliza para identificar un recurso por nombre en un particular espacio de nombres. Siguiendo la analogía anterior, equivale al nombre de una persona. |

El International Standard Book Number (ISBN, sistema para la identificación única de libros) es un típico ejemplo del uso de una URN. ISBN 1-57189-557-4 identifica sin ambigüedad una edición del libro Kabbalah on love. La URN de dicha edición sería urn:isbn:1-57189-557-4. Sin embargo, para tener acceso a este objeto y leer el libro, se necesita su ubicación, para lo cual una URL debe ser proveída.

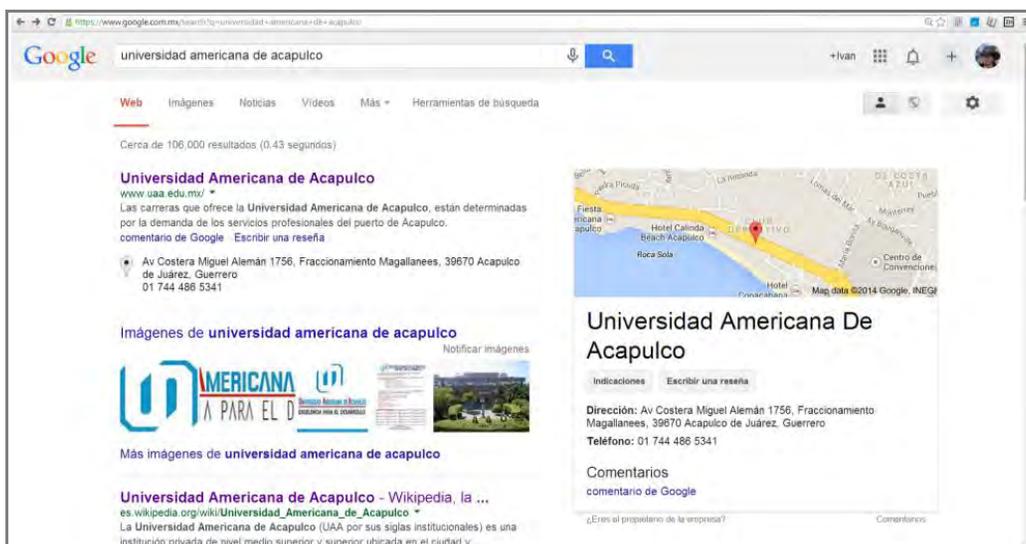


Figura 3.1. Ejemplo de una URL

Un ejemplo de una URL con todos sus elementos es <http://www.google.com.mx/search?q=universidad+americana+de+acapulco>, mostrada en la Figura 3.1, la cual define que: (1) por protocolo «http», (2) en el servidor «www.google.com.mx», (3) hay un recurso llamado «/search», (4) al cual le pasamos un Query (consúltese la sección 3.4 del (RFC N. W., RFC-3986 Uniform Resource Identifier (URI) Generic Syntax, 2005)) con el valor «universidad+americana+de+acapulco» asignado al parámetro «q». En un navegador Web al poner esta URL, se enviará una petición GET (véase Tabla 3.2) del protocolo HTTP, la cual indicará al servidor que el cliente desea obtener un recurso llamado «/search». El servidor Web verifica que dicho recurso se encuentra disponible, y como se trata de un script o aplicación del servidor, lo ejecutará con el parámetro «q=universidad+americana+de+acapulco». Dicho script realiza una consulta a su base de datos buscando referencias a sitios Web que coincidan con la búsqueda solicitada: «universidad americana de acapulco». Al obtener los resultados de su base de datos, genera código HTML con todas las URL que concuerdan, agrega referencias a imágenes y diseños CSS usados para la interfaz de usuario. Entonces es enviado al cliente como respuesta a la petición solicitada y visualizada como se muestra en la Figura 3.1.

El estándar HTTP define métodos para indicar la acción a ser realizada en el recurso identificado. Lo que éste recurso representa, haya datos preexistentes o los datos sean generados dinámicamente, depende de la implementación del servidor. Comúnmente el recurso corresponde a un

archivo o al resultado de un ejecutable que reside en el servidor. Los tres métodos más usados se especifican a continuación en la Tabla 3.2.

Tabla 3.2. Métodos HTTP más comunes

| Método | Descripción |
|--------|--|
| GET | Hace una petición de la representación del recurso especificado, y es usado para únicamente obtener datos y no debe tener otros efectos. |
| HEAD | Hace una petición idéntica al método GET, pero no espera el cuerpo de la respuesta. Útil para obtener los encabezados de la respuesta sin tener que transportar todo el contenido. |
| POST | Es una petición usada para que el servidor acepte la «entidad» contenida en la petición para el recurso especificado, es decir, que el servidor procese los datos enviados, los cuales por ejemplo pueden ser un comentario, un inicio de sesión o un nuevo elemento para una base de datos. |

3.3. Diseño Web y Aplicaciones

Diseño Web y Aplicaciones involucra los estándares para construir y desplegar páginas Web, incluidos HTML, CSS, SVG, Application Programming Interface (API, conjunto de funciones o métodos que ofrece una biblioteca para ser usada por otro software como una capa de abstracción) de dispositivos y otras tecnologías para aplicaciones Web (W3C, Web Design and Applications).



3.3.1. HTML – Hypertext Markup Language

Es el lenguaje usado para describir la estructura de páginas Web. HTML permite a los autores:

- Publicar documentos en línea con encabezados, texto, tablas, listas, fotos, etc.
- Obtener en línea documentos con enlaces de hipertexto.
- Diseñar formularios para llevar a cabo transacciones con servicios remotos, para uso en búsqueda de información, realizar reservaciones, ordenar productos, etc.
- Incluir hojas de cálculo, videoclips, clips de sonido y otras aplicaciones directamente en sus documentos.

Con HTML, los autores definen la estructura de las páginas usando marcas, las cuales son las etiquetas del lenguaje usadas para las piezas de contenido tales como párrafos, listas, tablas, etc. (W3C, Web Design and Applications » HTML && CSS, 2012).



3.3.2. CSS – Cascade Style Sheet

Es el lenguaje para describir la presentación de las páginas Web, incluyendo colores, estructura y fuentes. Permite adaptar la presentación a diferentes tipos de dispositivos, tales como pantallas grandes, pantallas pequeñas e impresoras. CSS es independiente de HTML, y puede ser usado con cualquier lenguaje de marcas basado en XML (véase XML -

Extensible Markup Language (XML)). La separación de HTML de CSS facilita el mantenimiento de sitios, compartir hojas de estilo entre páginas y ajustar las páginas a diferentes entornos. Esto es referido como la separación de la estructura (contenido) de la presentación (W3C, Web Design and Applications » HTML & CSS, 2012).



3.3.3. SVG – Scalable Vector Graphics

La Web es más que solo texto e información, es también un medio de expresión artística, visualización de datos y optimización de la presentación de información para diferentes audiencias con diferentes necesidades y expectativas. El uso de gráficos en sitios Web mejora la experiencia para los usuarios.

SVG (W3C, Scalable Vector Graphics (SVG) 1.1 (Second Edition), 2011) es como HTML pero para gráficos. Es un lenguaje de marcas para describir todos los aspectos de una imagen o aplicación Web, desde la geometría de las formas hasta la animación o presentaciones multimedia como videos y sonido. Es totalmente interactivo e incluye acceso por programación a un DOM (W3C, Document Object Model (DOM) Level 3 Core Specification, 2004) , así como animación declarativa, a través de la especificación Synchronized Multimedia Integration Language (SMIL) (W3C, Synchronized Multimedia Integration Language (SMIL 3.0), 2008). Soporta un amplio rango de características visuales, tales como gradientes, opacidad, filtros, enmarcado y aplicación de máscaras.

Como se muestra en la Figura 3.2, a diferencia de las imágenes raster que usan una estructura de matriz de puntos que representa una imagen rectangular, el uso de SVG permite gráficos escalables, definidos y reusables, desde simples dibujos hasta páginas Web mejoradas, gráficas totalmente interactivas de visualización de datos, juegos e imágenes independientes de alta calidad. SVG es soportado nativamente en la mayoría de los navegadores Web modernos (y con extensiones para permitir su uso en todos), y es ampliamente disponible en dispositivos móviles y receptores/decodificadores de televisión. Las principales herramientas de dibujo de gráficos vectoriales importan y exportan SVG, así como también pueden ser generados a través de lenguajes de programación de lado del cliente o del servidor (W3C, Web Design and Applications » Graphics).

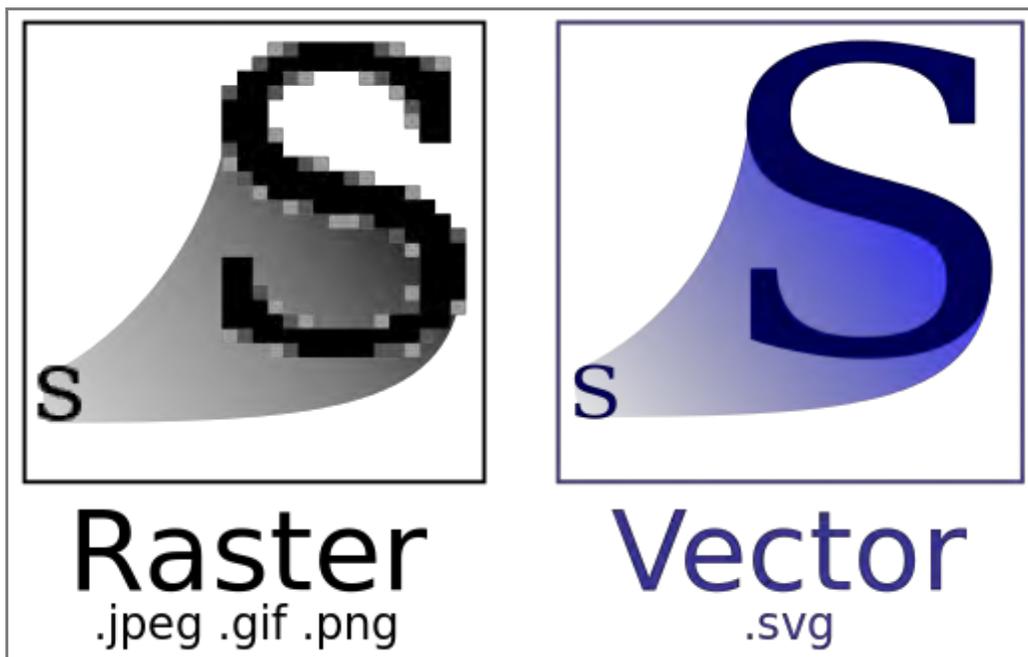


Figura 3.2. Diferencia de imágenes Raster y Vectoriales

3.3.4. XML - Extensible Markup Language



Formato simple basado en texto para la representación estructurada de información: documentos, datos, configuración, libros, transacciones, comunicados, etc. Es derivado de un viejo estándar llamado SGML (ISO, ISO 8879:1986 Standard Generalized Markup Language (SGML), 1986), con el fin de ser más adecuado para uso en la Web.

Es muy similar a un documento HTML. Sin embargo, las reglas sintácticas de XML son estrictas, es decir, que las herramientas XML no procesarán archivos que contengan errores. Esto significa que casi todos los documentos XML pueden ser procesados de forma segura por el software de computadoras.

XML es ampliamente usado en la actualidad. Es la base de muchos grandes estándares tales como Universal Business Language (UBL), Universal Plug and Play (UPnP) usado para aparatos electrónicos del hogar, formatos de procesamiento de texto como Open Document Format (ODF) y Open Office XML (OOXML), formatos de gráficos como SVG, también es usado para comunicación con XML Remote Procedure Call (XMLRPC, Remote Procedure Call que usa XML para codificar sus llamadas y usa HTTP como mecanismo de transporte) y Servicios Web, este es soportado directamente por lenguajes de programación y bases de datos, desde servidores gigantes hasta teléfonos móviles.

«Si llevas tu carro a reparación, la computadora del vehículo envía XML al sistema de diagnóstico del mecánico. En esta era, XML está en todas partes.» (W3C, XML Technology » XML Essentials).

3.3.5. Web Service – Servicio Web

Un servicio Web es un sistema software diseñado para soportar interacción máquina-máquina interoperable sobre una red. Éste tiene una interfaz descrita en un formato procesable por una máquina (especialmente Web Services Description Language, WSDL). Otros sistemas interactúan con el servicio Web en la forma prescrita por su descripción usando mensajes Simple Object Access Protocol (SOAP), transportados típicamente usando HTTP con una serialización XML en conjunto con otros estándares relativos a la Web (W3C, Web Services Glossary, 2004).



3.3.6. JSON – Javascript Object Notation

JSON es un formato de texto que facilita el intercambio de datos estructurados entre todos los lenguajes de programación. JSON tiene sintaxis de llaves «{» y «}», corchetes «[» y «]», dos puntos «:» y comas «,» que son usados en muchos contextos, perfiles, aplicaciones. JSON fue inspirado en la representación de objetos de Javascript, también conocido como ECMAScript en su definición de la especificación (ECMA, ECMA-404 The JSON Data Interchange Format, 2013). No se intenta imponer las representaciones internas de ECMAScript en otros lenguajes de programación. En lugar de ello, este comparte un pequeño subconjunto de representaciones textuales de ECMAScript con todos los otros lenguajes de programación.

JSON es agnóstico acerca de los números. En cualquier lenguaje de programación, puede haber una variedad de tipos de números de varias capacidades y complementos, fijos o flotantes, binarios o decimales. Eso puede hacer difícil el intercambio entre diferentes lenguajes de programación. En lugar de eso, JSON ofrece únicamente la representación de números usada por los humanos: una secuencia de dígitos. Todos los lenguajes de programación saben cómo tratar secuencias de dígitos aunque no correspondan con su representación interna. Eso es suficiente para permitir el intercambio.

El texto en JSON es una secuencia de puntos de código Unicode (Estándar ISO/IEC 10646 definido por el Unicode Consortium: Sitio oficial en la URL: <http://www.unicode.org/>). JSON también permite la notación con números hexadecimales de Unicode «\u».

Los lenguajes de programación varían ampliamente en si soportan objetos, y si lo hacen, también varían en las características y limitantes que ofrecen los objetos. Los modelos de sistemas de objetos pueden ser muy divergentes y continuamente están evolucionando. JSON, en lugar de ello, provee de una notación simple para expresar colecciones de pares «nombre/valor». La mayoría de los lenguajes de programación tendrán alguna implementación para representar tales colecciones, las cuales pueden ser nombradas como: «record, struct, dict, map, hash u object».

JSON también tiene soporte para listas ordenadas de valores. Todos los lenguajes de programación tienen alguna implementación para representar tales listas, como: «array, vector o list». Debido a que objetos

y arreglos se pueden anidar, árboles y otras estructuras complejas de datos pueden ser representados. Aceptando la sencilla notación de JSON, estructuras complejas de datos pueden ser fácilmente intercambiadas entre lenguajes de programación incompatibles.

JSON no soporta grafos cíclicos, al menos no directamente. JSON no es el indicado para aplicaciones que requieran datos binarios.



3.3.7. AJAX

AJAX se trata de actualizar partes de una página Web, sin necesidad de recargar todo el contenido, y proviene de la contracción Asynchronous JavaScript and XML.

AJAX puede ser usado como técnica para crear páginas Web rápidas y dinámicas, al solo actualizar pequeñas cantidades de datos con el servidor detrás de escena.



3.3.8. The Dojo Toolkit

Es un framework en Javascript que provee una interfaz de programación de aplicación, así como de controles para facilitar el desarrollo de aplicaciones Web. Se encuentra disponible en <https://dojotoolkit.org/>.

3.4. Lenguajes de Programación

«... Procesos computacionales son seres abstractos que habitan las computadoras. A medida que evolucionan, los procesos manipulan otras cosas abstractas llamadas *datos*. La evolución de un proceso está dirigida por un patrón de reglas llamadas *programa*. Las personas crean programas para dirigir procesos. En efecto, nosotros conjuramos los espíritus de las computadoras con nuestros hechizos.

Un proceso computacional es de hecho muy parecido a la idea de espíritu de un hechicero. Éste no puede ser visto ni tocado. Éste no está compuesto por materia. Sin embargo, éste es muy real. Puede desempeñar trabajo intelectual. Puede resolver preguntas. Puede afectar el mundo al desembolsar el dinero de un banco o al controlar un brazo robótico en una fábrica. Los procesos que usamos para conjurar los procesos son como los hechizos del hechicero. Ellos están cuidadosamente compuestos de expresiones simbólicas en lenguajes de programación arcanos y esotéricos que prescriben las tareas que queremos que nuestros procesos realicen.» (Abelson, Sussman, & Sussman, 1996).

Dicho de forma más técnica, los lenguajes de programación son lenguajes formales contruidos diseñados para comunicar instrucciones a una máquina, en particular una computadora. Pueden ser usados para controlar el comportamiento de una máquina o para expresar algoritmos (Hopcroft, Motwani, & Ullman, 2002).

3.4.1. PHP – PHP: Hypertext Preprocessor



PHP (The PHP Group, 2014) es un lenguaje interpretado ampliamente usado, open source (Programa cuyo código está disponible al público en general para uso y/o modificación de su diseño original) y de propósito general, adecuado especialmente para desarrollo Web y puede ser incrustado dentro del código HTML. Su sintaxis es parecida a los lenguajes C, Java y Perl. Aunque el objetivo principal del lenguaje es permitir que los desarrolladores Web puedan escribir páginas Web dinámicamente generadas, tiene el potencial para hacer tareas de otras áreas.

Hay tres áreas principales donde los scripts PHP son usados:

- Server-side scripting. El código se ejecuta en el servidor Web, generando el código HTML que es enviado al cliente. El cliente recibe los resultados de ejecutar ese script, pero no conoce el código que hay detrás.
- Programas de línea de comandos. Se puede ejecutar un script PHP sin servidor o cliente Web. Sólo se requiere el interprete PHP para usarlo en esta forma.
- Aplicaciones de escritorio. PHP probablemente no es el mejor lenguaje de programación para crear aplicaciones de escritorio con una interface de usuario, pero se puede usar algunas características avanzadas de PHP en las aplicaciones de lado del cliente como PHP-GTK, con la posibilidad de que sean aplicaciones multiplataforma.

Una de las mayores y más significantes fortalezas de PHP es su soporte para un amplio rango de bases de datos.



3.4.2. Javascript

Es un lenguaje de programación dinámico. Es comúnmente usado como parte de los navegadores Web, cuya implementación permite scripts de lado del cliente para interactuar con el usuario, controlar el navegador, comunicarse asíncronamente con algún servidor y alterar el documento que se está mostrando. También es usado en programación de redes de lado del servidor (ejemplo Node.js, plataforma software para aplicaciones de red y de servicios, diseñada para maximizar el flujo de datos y eficiencia, usando E/S sin bloqueo y eventos asíncronos. Está disponible en la URL: <http://nodejs.org>), desarrollo de juegos y la creación de aplicaciones de escritorio y para móviles.

Javascript es un lenguaje interpretado basado en prototipos con tipos de datos dinámicos y funciones de primera clase (funciones pueden ser usadas como valores dentro del código). Su sintaxis esta influenciada por C, y copia muchos nombres y convenciones de nombramiento de Java, pero no está relacionado con ninguno de los dos lenguajes y tienen semántica muy diferente. Los principios de diseño claves en Javascript han sido tomados de los lenguajes de programación Self y Scheme. Este es un lenguaje multiparadigma, con soporte de los estilos: orientado a objetos, imperativo y funcional.

Fue inventado por Brendan Eich para ser usado en el navegador Web Netscape en 1995, y fue adoptado por la asociación de estándares ECMA en 1997 bajo la especificación (ECMA, ECMA-262 ECMAScript Language Specification, 2011) y el estándar (ISO & IEC, ISO/IEC 16262:2011 ECMAScript scripting language, 2011).



3.5. Expresiones Regulares

En ciencia computacional teórica y teoría de lenguajes formales, una expresión regular denominada regex o regexp, referirse a (López-Quijado, 2007), es una secuencia de caracteres que forma un patrón de búsqueda, usado principalmente para operaciones de búsqueda y reemplazo de patrones de texto. El concepto inicio en 1950, cuando el matemático Stephen Kleene formalizó la descripción de un lenguaje regular, y llegó a ser de uso común en utilidades de procesamiento de texto en Unix, como «ed» y «grep» (global regular expression print).

Cada carácter en una expresión regular es entendido como un metacarácter con su significado especial, o un carácter regular con su significado literal. Juntos, pueden ser usados para identificar material textual de cierto patrón, o procesar un número de instancias de éste que puede variar de una igualdad muy precisa a ser muy similares al patrón. La secuencia del patrón en sí misma es una expresión la cual es una sentencia en un lenguaje específicamente diseñado para representar objetivos prescritos en la forma más concisa y flexible para dirigir la

automatización del procesamiento de archivos de texto generales, específicas formas de texto o cadenas de texto aleatorias.

Un procesador de expresiones regulares traduce una expresión regular en un autómata finito no determinístico, el cual es convertido en determinístico y ejecutado en la cadena de texto objetivo para reconocer subcadenas que concuerdan con la expresión regular. (Hopcroft, Motwani, & Ullman, 2002).

Las expresiones regulares están especificadas por el estándar *POSIX* (ISO, IEC, & IEEE, ISO/IEC/IEEE 9945:2009 Portable Operating System Interface (POSIX) Base Specifications, Issue 7, 2009) bajo los conjuntos *BRE (Basic Regular Expression)* y *ERE (Extended Regular Expression)*, así como definidas por la especificación *Perl 5* (Perl, 2014).

3.6. Integración de Tecnologías

Ya que el Sistema Monitor deberá almacenar, actualizar y consultar datos, se deberá utilizar un RDBMS en el cual se tenga toda la información organizada, y para ello se ha escogido el software MySQL por las siguientes razones:

1. Los datos se almacenan en archivos que tienen un tamaño menor que otras bases de datos.
2. El motor de datos de MySQL es mucho más rápido, tanto grabando datos como localizándolos y recuperándolos, que el de otras bases de datos.

3. Es gratuito, por lo cual no se requiere comprar licencias. Aunque existen licencias comerciales.

Para la programación del subsistema de adquisición de datos, se escogerá el lenguaje PHP, derivado de que uno de sus usos es como aplicación en modo línea de comandos. Con esto tendremos acceso a recursos de red y conexión a bases de datos, por lo que se puede programar perfectamente el subsistema planteado.

El subsistema de visualización de datos utilizará el Modelo de Estándares Web definido como The Open Web Platform (consúltese en la URL: https://www.w3.org/wiki/Open_Web_Platform) descrito en la Figura 3.3, el cual define el uso de HTML, CSS y Javascript. Esto permitirá que cualquier usuario desde cualquier dispositivo pueda acceder al Sistema Monitor.

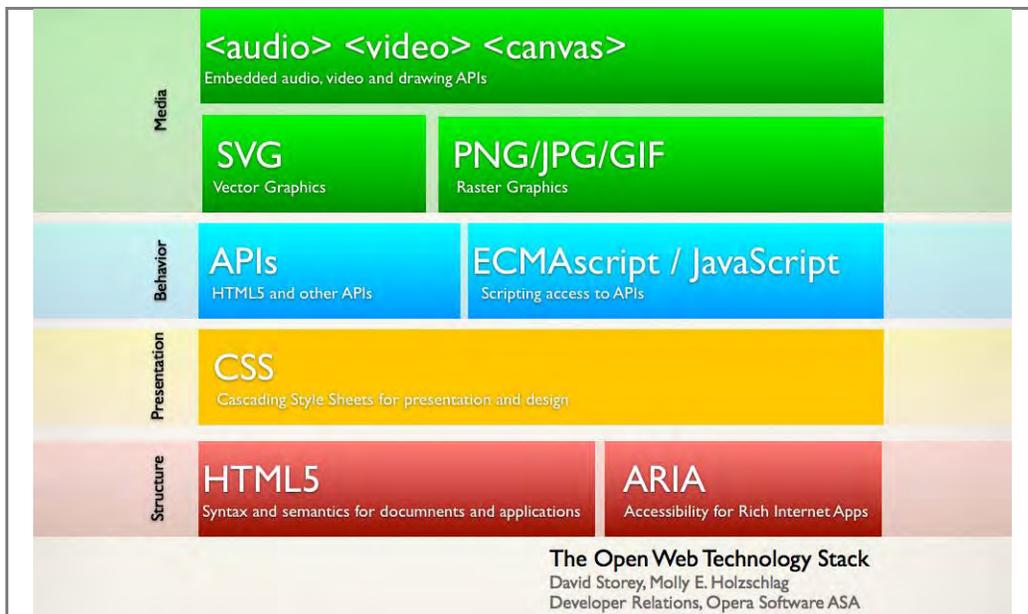


Figura 3.3. Modelo de Estándares Web, por The Open Web

Ya que los datos de tiempo real estarán almacenados en una base de datos de MySQL, se deberá realizar la programación dinámica de lado del servidor en PHP para la generación del código HTML y para el envío de los datos de los monitores. Así mismo, se deberá mantener disponibles todos los recursos CSS, Javascript e imágenes que utilice el subsistema. Una representación simplificada se encuentra en la Figura 3.4.



Figura 3.4. Arquitectura de Aplicaciones Web

Para lograrlo, se utilizará un servidor Web llamado Apache Web Server (comúnmente nombrado como Apache), el cual permite utilizar el lenguaje PHP para generar de forma dinámica el código HTML, así como también es el más usado a nivel mundial de acuerdo a estadísticas de

W³Techs (véase Figura 3.5), mismas que pueden ser consultadas en la URL: http://w3techs.com/technologies/overview/Web_server/all.

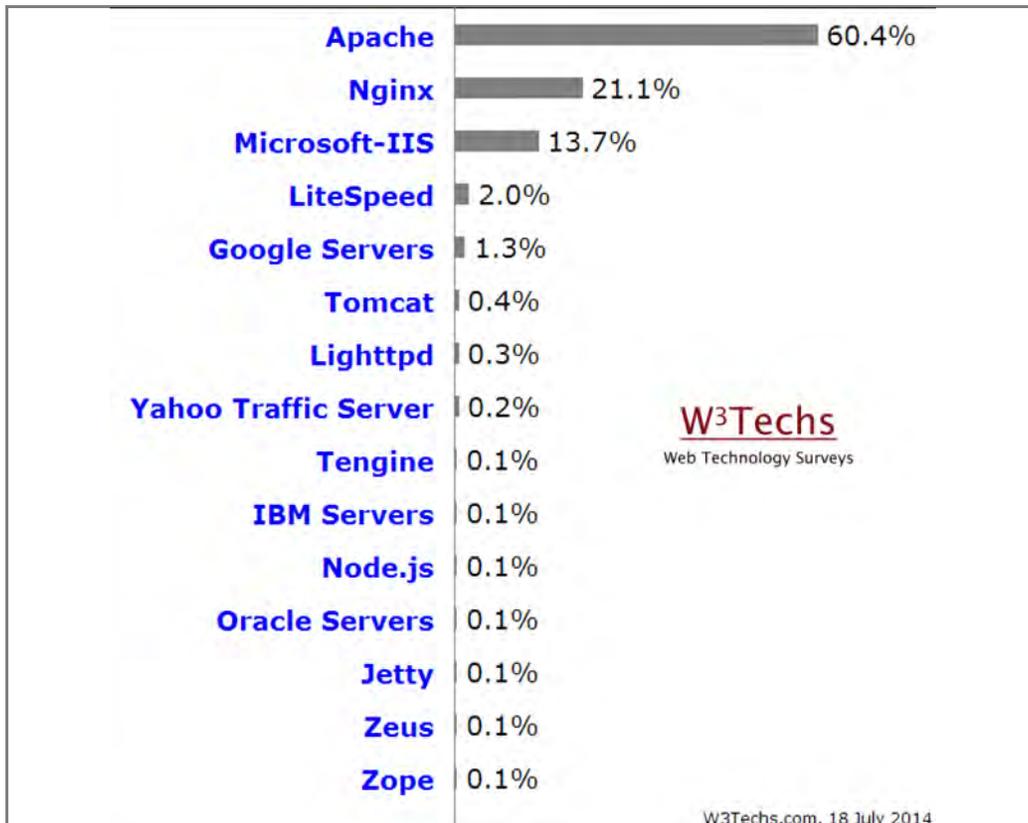


Figura 3.5. Porcentajes de uso de servidores Web

Ya que se utilizará programación dinámica de lado del servidor por el subsistema de visualización de datos, se podrá conseguir generar el código HTML para la visualización del subsistema, y de forma eficiente adquirir datos de la base de datos y enviarlos al cliente en formato JSON.

También se deberá crear una base común de clases y librerías de objetos que permitan unificar la forma de acceso a bases de datos, configuración,

registro de actividad, librerías y motores de los sistemas remotos de datos de tiempo real.

Dichos sistemas remotos, los cuales son el SCADAWeb y el WebSICLE, utilizan archivos XML generados dinámicamente por sus servidores los cuales cuentan con los datos de tiempo real que se requieren para el Sistema Monitor, por lo cual para extraer de forma eficiente la información se utilizará expresiones regulares, la cual una vez extraída, se almacenará en la base de datos de MySQL.

3.7. Metodología de Desarrollo de Software

«En la década de 1970 los programas fueron creciendo en complejidad, por lo que la antigua técnica *code & fix* (codificar y corregir) terminó quedando obsoleta» (Cantone, 2006).

«La metodología para el desarrollo de software es un modo sistemático de realizar, gestionar y administrar un proyecto para llevarlo a cabo con *altas posibilidades de éxito*. Esta sistematización nos indica como dividiremos un gran proyecto en módulos más pequeños llamados etapas, y las acciones que corresponden con cada una de ellas, nos ayuda a definir las entradas y salidas para cada una de las etapas y, sobre todo, normaliza el modo en que administraremos el proyecto» (Cantone, 2006).

Los métodos de desarrollo se refieren a la separación en distintas etapas de actividades destinadas a la planeación y administración del proyecto. Dentro de los principales tenemos los siguientes:

- **Lineal o en cascada**, es el más sencillo de todos los métodos. Consiste en descomponer la actividad global del proyecto en etapas separadas que son realizadas de manera lineal, tal como se muestra en la Figura 3.6. Algunas adaptaciones sugieren retroceder a la etapa anterior para corregir problemas encontrados en la etapa actual.

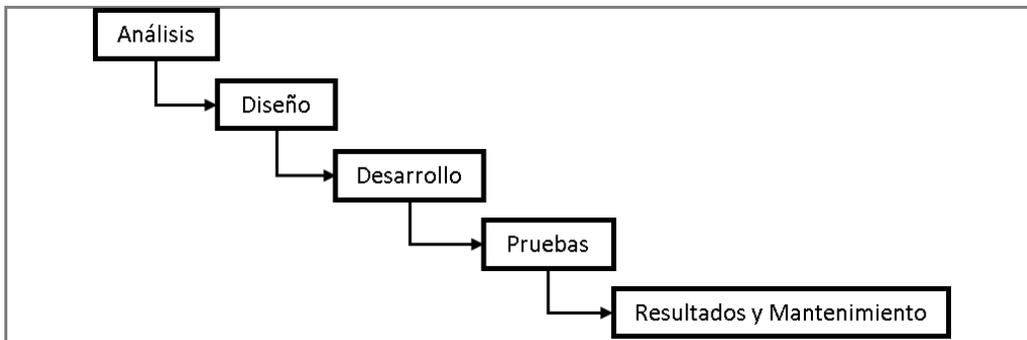


Figura 3.6. Ciclo de vida lineal

- **Por prototipos**, usado cuando no se conocen las especificaciones de forma precisa, logrando un producto parcial o intermedio, antes de realizar el producto final. Véase Figura 3.7.

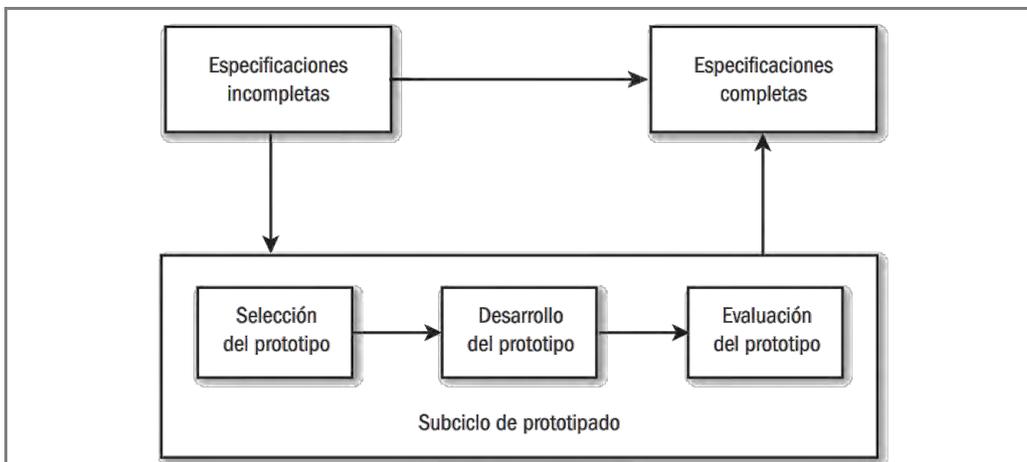


Figura 3.7. Ciclo de vida por prototipos

- **En espiral**, se basa en una serie de ciclos repetitivos para ir ganando madurez en el producto final. Toma en cuenta el concepto de riesgo que aparece debido a las incertidumbres e ignorancias de los requerimientos proporcionados al principio del proyecto o que surgirán durante el desarrollo. En la Figura 3.8 se puede apreciar gráficamente.

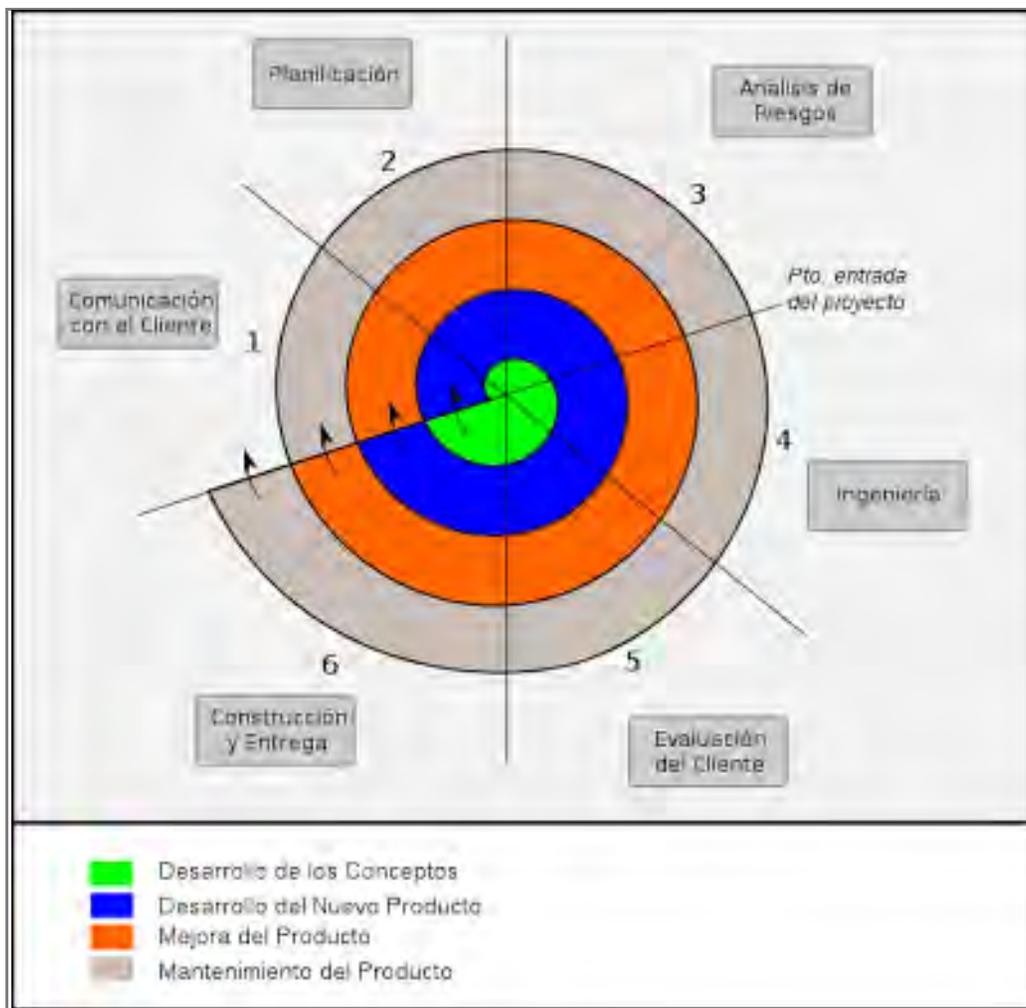


Figura 3.8. Ciclo de vida en espiral

3.8. Metodología a Utilizar

Conforme a las metodologías expuestas en el capítulo Metodología de Desarrollo de Software, y derivado de que no se cuenta con especificaciones claras acerca del sistema final, se procederá a utilizar la metodología por prototipos para garantizar que el sistema vaya acercándose a la versión definitiva conforme se vaya definiendo las características.

3.9. Diagrama Unifilar

Es representación gráfica de una red o un arreglo eléctrico en la que los conductores y/o fases se representan mediante una única línea. Dentro de la CFE se utilizan para simplificar los arreglos y visualizar fácilmente el estado operativo de las subestaciones. También incluyen diversos elementos como se muestra en la Figura 3.9.

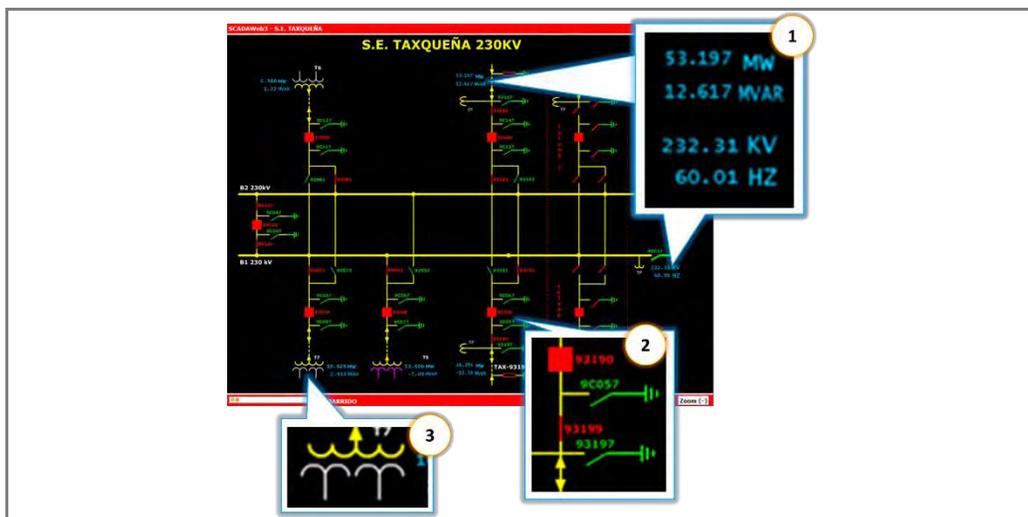


Figura 3.9. Componentes de un Diagrama Unifilar

Estos elementos numerados hacen referencia a las mediciones y estados de interruptores y cuchillas, así también se muestran elementos dentro de la subestación. A continuación se describen:

- El número 1 son las mediciones de potencia activa, reactiva, voltaje y frecuencia.
- El número 2 son los interruptores de potencia representados en forma de cuadro, y las cuchillas se presentan como líneas en forma vertical, horizontal o en diagonal, y se colorean de acuerdo a la Tabla 3.3. Colores de elementos dinámicos en diagramas unifilares.

Tabla 3.3. Colores de elementos dinámicos en diagramas unifilares

| Color | Descripción |
|--------------|--|
| Rojo | Indica que el elemento se encuentra cerrado, es decir, está conduciendo corriente. Las cuchillas adicionalmente al color, se representan horizontales o verticales de acuerdo a las líneas de interconexión con los demás elementos. |
| Verde | Indica que el elemento se encuentra abierto, es decir, no permite el flujo de corriente. Las cuchillas adicionalmente al color, se representan en diagonal para indicar que no hay interconexión con los demás elementos. |

- El número 3 son los bancos de transformación o transformadores de potencia, independientemente de si son monofásicos o trifásicos, su representación es como dos bobinas encontradas y apenas separadas.

3.9.1. Nomenclatura

La nomenclatura es utilizada para hacer referencia de forma única a los elementos que intervienen en la operación del Sistema Eléctrico Nacional. Dichas nomenclaturas están divididas para subestaciones y equipo eléctrico primario.

Subestaciones

Su nomenclatura está compuesta por tres letras mayúsculas propias de su nombre. Las reglas para determinar la nomenclatura de una subestación están definidas en el REDOSEN (véase Anexo A. Glosario). Como ejemplo:

- La nomenclatura de la subestación Taxqueña es TAX.
- La nomenclatura de la subestación Contreras es CRS.

Equipo Eléctrico Primario

El equipo eléctrico primario debe tener una nomenclatura para identificar la bahía eléctrica a la que pertenece y que papel juega dentro de la misma. Está compuesta por 5 caracteres que representan:

- Primero, tensión de operación de acuerdo a la Tabla 3.4.
- Segundo, tipo de equipo.
- Tercero y cuarto, número de identificación asignado al equipo.
- Quinto, tipo de dispositivo.

Tabla 3.4. Caracteres de identificación de los niveles de tensión

| kV desde | kV hasta | Carácter |
|---------------------|---------------------|-----------------|
| 0 | 2.4 | 1 |
| 2.41 | 4.16 | 2 |
| 4.17 | 6.99 | 3 |
| 7 | 16.5 | 4 |
| 16.6 | 44 | 5 |
| 44.1 | 70 | 6 |
| 70.1 | 115 | 7 |
| 115.1 | 161 | 8 |
| 161.1 | 230 | 9 |
| 230.1 | 500 | A |
| 500.1 | 700 | B |

CAPÍTULO 4

DESARROLLO E IMPLEMENTACIÓN

CAPÍTULO 4. DESARROLLO E IMPLEMENTACIÓN

4.1. Estableciendo las Bases

El sistema será referido como Monitor, y estará compuesto por dos subsistemas, uno para la adquisición de los datos de flujos de energía y estado operativo de las subestaciones de los sistemas SCADAWeb y WebSICLE, y otro para la representación gráfica de los flujos mediante la visualización de diagramas unifilares de las redes troncales.

Cabe señalar que el único punto en común de ambos subsistemas será la base de datos, garantizando así la independencia de su funcionalidad.

4.1.1. Concepto de Monitor

Uno de los conceptos clave que utiliza el Sistema Monitor, es su propio nombre. «**Monitor**» surge como la abstracción requerida para visualizar de forma general y simplificada los flujos de energía de las redes eléctricas troncales del área central del país.

4.1.2. Sistema Monitor y sus Subsistemas

El sistema Monitor estará dividido en dos subsistemas independientes con la finalidad de garantizar su independencia funcional y forma de implementación, tal como se muestra en la Figura 4.1. Esto tiene la finalidad de realizar el desarrollo como dos subproyectos de desarrollo de software:

- **Subsistema «Monitor :: UTMu»**, encargado del proceso de adquisición de datos de tiempo real de las subestaciones.
- **Subsistema «Monitor :: Web»**, encargado del proceso de visualización y representación gráfica ante el usuario de los datos de tiempo real de las subestaciones.

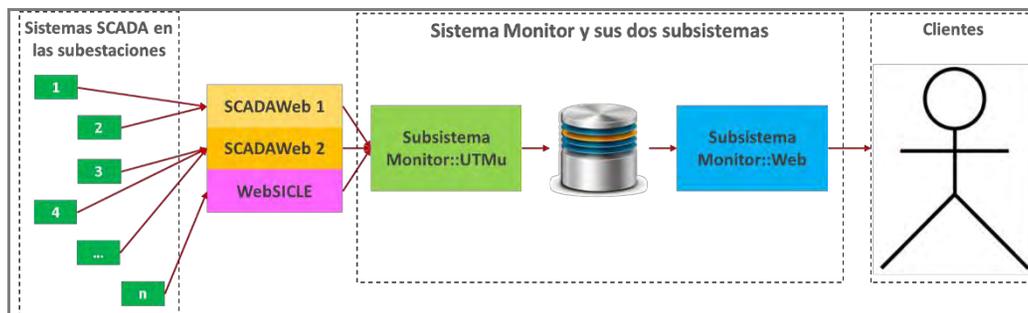


Figura 4.1. Diagrama simplificado del sistema Monitor

4.1.3. Metodología de Desarrollo de Software

En el capítulo anterior se definió las metodologías más comunes para el desarrollo de proyectos de software, y éste capítulo se desarrollará los dos subsistemas de forma independiente en base a la metodología por prototipos.

4.2. Monitor :: UTMu

A continuación se documenta el prototipo final en forma de versión “as-is” (como está) del subsistema Monitor :: UTMu.

4.2.1. Análisis

➤ Levantamiento de datos

Los datos a adquirir se encuentran alojados en servidores distribuidos en diferentes localidades del área central de país.

Al momento de realizar éste análisis, existen dos tipos de sistemas diferentes de los cuales se extraerá los datos de tiempo real: SCADAWeb y WebSICLE.

Para el SCADAWeb los datos de tiempo real de los diagramas unifilares se adquieren de forma asíncrona por método GET del protocolo HTTP. Existen dos scripts programados en lenguaje ASP ubicados normalmente en la ubicación **«http://<ip_servidor>/scadaWeb/diagramas/»** que sirven la información de acuerdo al tipo de datos:

- **«traeestados.asp»**, utilizado para valores de estados.
- **«traemedsicle.asp»**, utilizado para valores de mediciones.

Para identificar de qué subestación se requieren los datos, ambos scripts esperan en la sección Query de la URL, la variable CveUnifilar con el nombre del unifilar dentro de dicho sistema SCADAWeb, el cual normalmente coincide con la nomenclatura de la subestación.

Sin requerir ningún tipo de autenticación de usuario, el sistema SCADAWeb responde con los datos solicitados en formato XML usando como estructura una etiqueta general y etiquetas internas:

- **Para estados**, como se muestra en la Figura 4.2, el nombre de la etiqueta lleva las tres letras nomenclatura de la subestación seguidas por la nomenclatura del equipo primario o nombre de la alarma. El valor de la etiqueta es el valor digital.

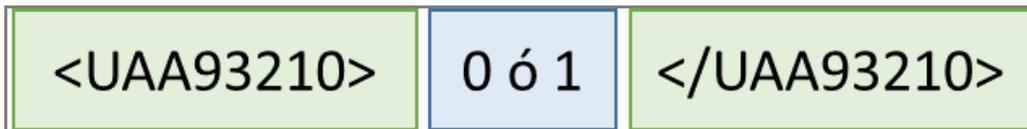


Figura 4.2. Formato de estados del Sistema SCADAWeb

- **Para mediciones**, como se muestra en la Figura 4.3, el nombre de la etiqueta lleva las tres letras nomenclatura de la subestación seguidas por la nomenclatura de la bahía eléctrica y un sufijo que es la unidad de ingeniería usada para la medición. El valor de la etiqueta es el valor analógico.

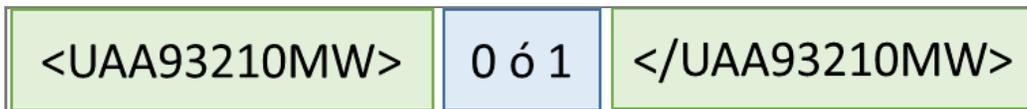


Figura 4.3. Formato de mediciones del Sistema SCADAWeb

Para el sistema WebSICLE, los datos de tiempo real de los diagramas unifilares del sistema WebSICLE se adquieren de forma asíncrona por método GET del protocolo HTTP. Existen para cada diagrama un script llamado «**BuildXml.asp**» y está programado en lenguaje ASP ubicado en la ubicación «**http://<ip_servidor>/<ruta_base>/<dir_diagrama>/**». En algunas ocasiones el nombre del script puede tener un prefijo.

Sin requerir ningún tipo de autenticación de usuario, el sistema WebSICLE responde con los datos solicitados en formato XML iniciando con la etiqueta «<?xml», los valores de los atributos de las etiquetas pueden estar encerrados por comillas dobles «"» o apóstrofes «'». Asimismo, dicho sistema utiliza dos formatos posibles:

- **Formato 1**, mostrado como ejemplo en la Figura 4.4, inicia con una etiqueta general, cuyo nombre son las tres letras nomenclatura de la subestación seguidas sufijos variables en cantidad de letras y números separados por los caracteres «_» y «-». A continuación hay una colección de etiquetas «**DEVICE**» con una serie de etiquetas internas:
 - El nombre de la bahía eléctrica está contenido en una etiqueta «**NAME**». Puede existir espacios antes o después del nombre.
 - Dependiendo del tipo de bahía eléctrica, puede contener mediciones. Si la bahía en cuestión contiene mediciones, éstas se encuentran como valor de la etiqueta «**MWATTS**».
 - El estado del interruptor de dicha bahía es el valor de la etiqueta «**BkrSt**». El nombre de la etiqueta puede encontrarse seguido de cualquier cantidad de espacios.

```

<?xml version='1.0' encoding='ISO-8859-1'?>
<UAA_NH_PRI-400 TIME='19:31:16' DATE='01/06/2014'>
  <DEVICE ID='A3580 '>
    <NAME>A3580 </NAME>
    <HWATTS>553.161</HWATTS>
    <TOTVAR>8.20713</TOTVAR>
    <BkrSt >></BkrSt >>
    <BlkTxGAT >></BlkTxGAT >>
    <DcProt >></DcProt >>
    <OutRngVDP >></OutRngVDP >>
    <RecOn >></RecOn >>
    <SwitB2 >></SwitB2 >>
    <SwitGn >>1</SwitGn >>
    <SwitHBkr >></SwitHBkr >>
    <SwitLn >></SwitLn >>
    <TxDTT >></TxDTT >>
  </DEVICE>
  <DEVICE ID='A3790 '>
    <NAME>A3790 </NAME>
    <HWATTS>-348.803</HWATTS>
    <TOTVAR>-12.3107</TOTVAR>
    <BkrSt >></BkrSt >>
    <BlkRec1P >></BlkRec1P >>
    <ChgrFail >></ChgrFail >>
    <ClsBlkBkr >></ClsBlkBkr >>
    <LoPresAir >></LoPresAir >>
    <LoPresOil >></LoPresOil >>
    <LoPresSF6 >></LoPresSF6 >>
    <OutRngVDP >></OutRngVDP >>
    <PoleDisc >></PoleDisc >>
    <PsBkr >></PsBkr >>
    <PsProt >></PsProt >>
    <PsSwit >></PsSwit >>
    <RxDTT >></RxDTT >>
    <SwitB2 >></SwitB2 >>
    <SwitGn >>1</SwitGn >>
    <SwitHBkr >></SwitHBkr >>
    <SwitLn >></SwitLn >>
  </DEVICE>
</UAA_NH_PRI-400>

```

Figura 4.4. Formato XML número 1 del Sistema WebSICLE

- **Formato 2**, mostrado como ejemplo en la Figura 4.5, inicia con una etiqueta general, cuyo nombre son las tres letras nomenclatura de la subestación. A continuación hay una colección de etiquetas «**DEVICE**» cuyo valor del atributo ID corresponde con las tres letras nomenclatura de la subestación seguidas del nombre de la bahía eléctrica. Cada etiqueta «**DEVICE**» contiene una serie de etiquetas internas:
 - El nombre de la bahía eléctrica está contenido en una etiqueta «**NAME**» y es exactamente igual al atributo «**ID**» de la etiqueta padre «**DEVICE**».
 - Dependiendo del tipo de bahía eléctrica, puede contener mediciones. Si la bahía en cuestión contiene mediciones, éstas se encuentran como valor de la etiqueta «**TotW**».
 - El estado del interruptor de dicha bahía es el valor de la etiqueta «**XCBR**». El nombre de la etiqueta en algunas

ocasiones esta precedido por el prefijo «B» y opcionalmente un dígito, ó bien con por el prefijo «RE» y opcionalmente un dígito. Asimismo, dicha etiqueta puede tener como sufijo un dígito.

```
<?xml version='1.0' encoding='ISO-8859-1'?>
<UAA TIME='19:29:14' DATE='01/10/2008'>
  <DEVICE ID='UAAA3240'>
    <NAME>UAAA3240</NAME>
    <TotW>193.50</TotW>
    <TotVar>-45.11</TotVar>
    <SWIT0>0</SWIT0>
    <XCBR0>1</XCBR0>
    <B2SDRV0>1</B2SDRV0>
    <SDRV3>1</SDRV3>
    <SDRV2>1</SDRV2>
  </DEVICE>
  <DEVICE ID='UAAA3U30'>
    <NAME>UAAA3U30</NAME>
    <TotW>154.20</TotW>
    <TotVar>-32.01</TotVar>
    <SDRV2>1</SDRV2>
    <SDRV3>1</SDRV3>
    <XCBR0>1</XCBR0>
    <SWIT0>0</SWIT0>
    <B2SDRV0>1</B2SDRV0>
  </DEVICE>
</UAA>
```

Figura 4.5. Formato XML número 2 del Sistema WebSICLE

En cualquiera de los formatos, los valores de estado pueden estar configurados para enviarse como lógica inversa, es decir, mientras que normalmente el estado 0 indica interruptor abierto y el estado 1 cerrado, con lógica inversa el estado 0 indica cerrado y el estado 1 indica interruptor abierto.

➤ Definición de requerimientos

A continuación en la Tabla 4.1, se definen los requerimientos funcionales del Subsistema Monitor :: UTMu.

Tabla 4.1. Tabla de requerimientos funcionales de Monitor :: UTMu

| Identificador del Requerimiento | Descripción |
|--|--|
| SMU_RF1 | El subsistema se ejecutará como una aplicación CLI o como un daemon Linux. |
| SMU_RF2 | El subsistema deberá generar un archivo por día con el registro de actividad. |
| SMU_RF3 | El subsistema deberá mostrar, si el usuario lo indica al momento de iniciar la ejecución, los mensajes de depuración emitidos para detección de problemas o situaciones no previstos. |
| SMU_RF4 | Se utilizará el patrón de ejecución Supervisor Worker. |
| SMU_RF5 | El Supervisor realizará el monitoreo de la base de datos por cambios en los repositorios configurados, creando o eliminando Workers. |
| SMU_RF6 | El Supervisor monitoreará la ejecución de los Worker, y en caso de que un Worker haya terminado su ejecución inesperadamente, el Supervisor deberá crear un nuevo Worker para continuar los trabajos del anterior. |
| SMU_RF7 | El Worker verificará el estado de conexión a la base de datos antes de cada ciclo de interrogación, en caso de haberse perdido la conexión, éste terminará su ejecución. |
| SMU_RF8 | El Worker deberá monitorear al Supervisor, en caso de detectar su ausencia, éste deberá terminar su ejecución aunque el Supervisor no lo haya indicado. |
| SMU_RF9 | El Worker será un subsistema de Adquisición, Reconocimiento y Almacenamiento (ARA) de los datos de tiempo real de un grupo de subestaciones. |

En la Tabla 4.2, se especifican los requerimientos no funcionales del subsistema Monitor :: UTMu.

Tabla 4.2. Tabla de requerimientos no funcionales de Monitor :: UTMu

| Clasificación | Identificador del Requerimiento | Descripción |
|----------------------|--|---|
| Interfaz | SMU_RNF1 | El subsistema deberá tener una interfaz en línea de comandos. |
| | SMU_RNF2 | El subsistema deberá ofrecer un parámetro para ejecutarse en modo daemon. |
| | SMU_RNF3 | El subsistema permitirá habilitar los mensajes de depuración del Supervisor. |
| | SMU_RNF4 | El subsistema permitirá habilitar los mensajes de depuración del Worker. |
| Codificación | SMU_RNF5 | Utilizará un conjunto de clases que actúen como el núcleo del sistema. |
| | SMU_RNF6 | El núcleo del sistema implementará un mecanismo para que el Supervisor y los Worker registren su actividad. |
| | SMU_RNF7 | Utilizará una jerarquía de clases que permita encapsular en cada nivel características particulares de motores. |
| | SMU_RNF8 | Los motores implementarán los métodos de adquisición y reconocimiento para los sistemas SCADAWeb y WebSICLE. |
| | SMU_RNF9 | Se utilizará RegEx tipo PERL5 para la extracción y reconocimiento de los |

| | | |
|----------------|-----------|--|
| | | datos obtenidos en formato XML de los motores destinados a los sistemas SCADAWeb y WebSICLE. |
| Configuración | SMU_RNF10 | Existirá un archivo de configuración del sistema, en el que se especifique acceso a base de datos, tiempo máximo de respuesta de solicitudes y tiempo de ciclo de interrogación. |
| | SMU_RNF11 | La base de datos contendrá una tabla de los repositorios, con la configuración de acceso y propiedades de los datos. |
| Almacenamiento | SMU_RNF12 | La tabla de los repositorios deberá incluir un campo con el nombre del Worker que será el responsable de mismo. |
| | SMU_RNF13 | La tabla de los repositorios deberá incluir un campo con el nombre de la subestación a la que pertenecen. Esto derivado de que el motor implementado requiera más de un repositorio, o bien, haya más de un sistema SCADA en la subestación. |
| | SMU_RNF14 | Los datos de tiempo real se almacenarán en pares Tag / Value (Etiqueta / Valor) en una tabla de la base de datos. |
| Hardware | SMU_RNF15 | Se utilizará el servidor destinado por el área en que surge este proyecto. |

| | | |
|------------------------------|-----------|---|
| Límitantes de recursos | SMU_RNF16 | Capacidad de almacenamiento de 80 GB - el consumido el stack LAMP (Linux, Apache, MySQL y PHP). |
| | SMU_RNF17 | Capacidad de 4 GB de RAM. |
| | SMU_RNF18 | Procesador de doble núcleo. |
| Software | SMU_RNF19 | El sistema operativo del servidor utilizará el stack LAMP (Linux, Apache, MySQL y PHP). |
| Compatibilidad de plataforma | SMU_RNF20 | El Worker se ejecutará de forma independiente como un subproceso POSIX del Supervisor. |

➤ Modelado de negocio

En la Figura 4.6 se muestra los casos de uso de las situaciones generales del subsistema Monitor :: UTMu. Cada escenario describe una secuencia de eventos y cada secuencia a su vez se ejecuta mediante la interacción del actor administrador del sistema con los repositorios.

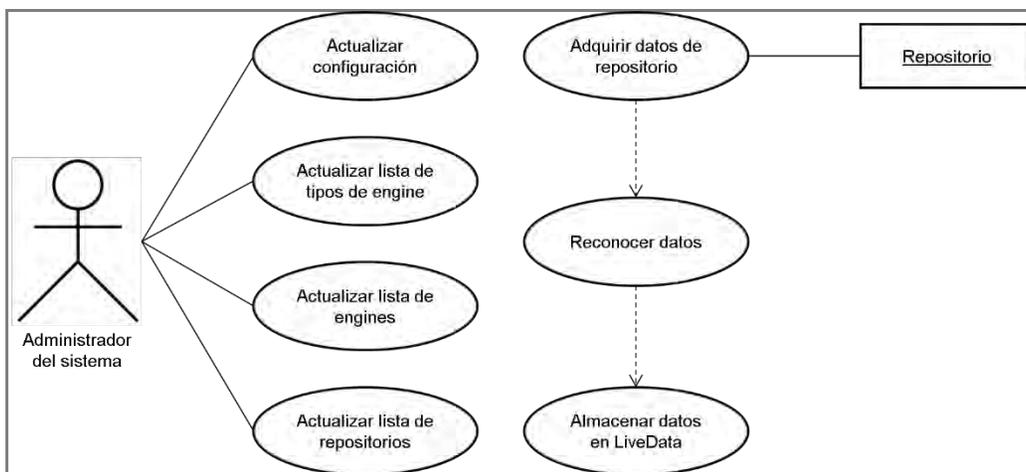


Figura 4.6. Diagrama de caso de uso general del subsistema Monitor :: UTMu

4.2.2. Diseño

➤ Modelo de datos

A continuación en la Figura 4.7, se aprecia el modelo entidad-relación de la base de datos del subsistema Monitor :: UTMu.

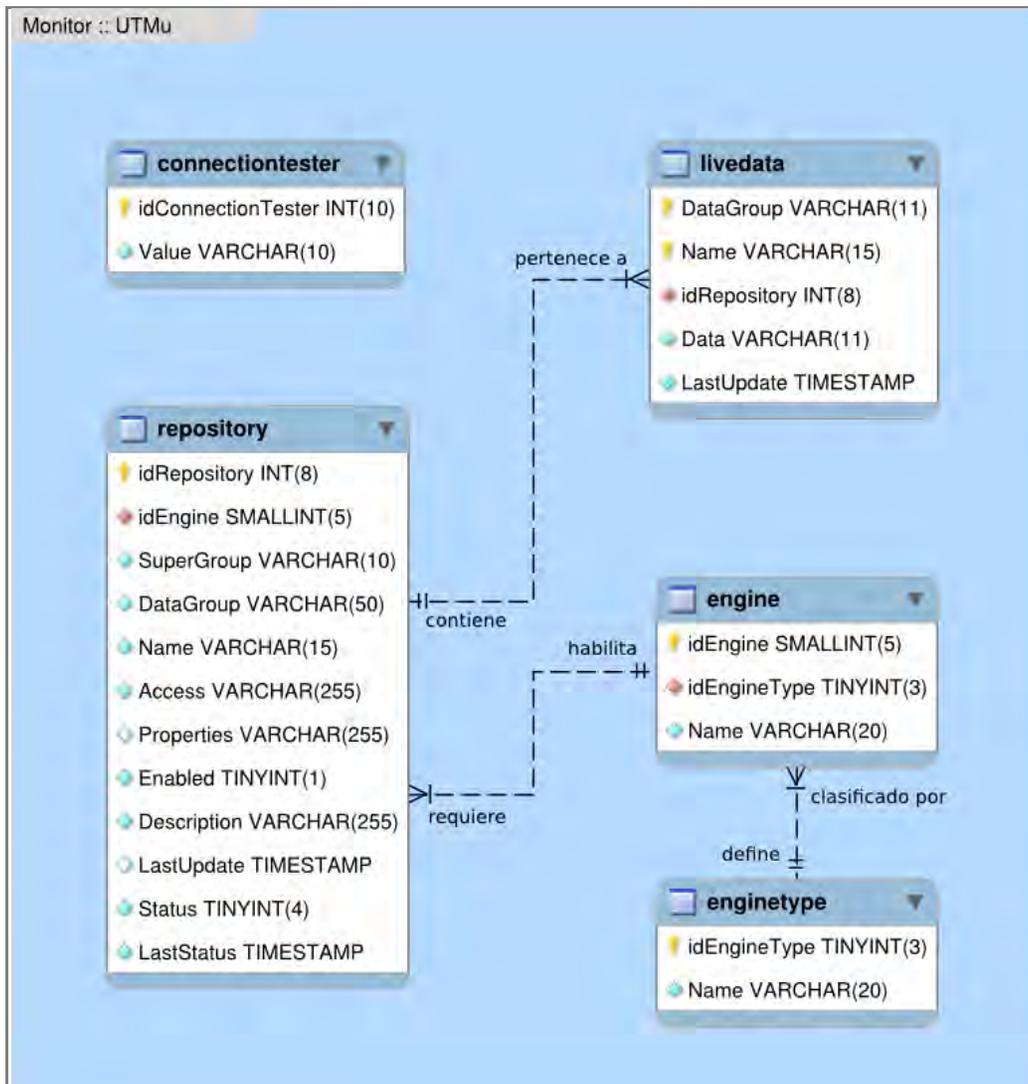


Figura 4.7. Modelo entidad relación del Subsistema Monitor :: UTMu

➤ Diccionario de datos

Tabla connectiontester, es usada para pruebas de conectividad de base de datos del Supervisor y los Worker. Véase la Tabla 4.3.

Tabla 4.3. Diccionario de datos: Tabla connectiontester

| Columna | Tipo | Nulo | Pred. | Comentarios |
|----------------------------|-------------|------|-------|--------------------|
| idConnectionTester (PK) | INT(10) | | | m |
| Value | VARCHAR(10) | | | Valor del registro |

Tabla enginetype, es usada como catálogo de los tipos de motores implementados. Véase la Tabla 4.4.

Tabla 4.4. Diccionario de datos: Tabla enginetype

| Columna | Tipo | Nulo | Pred. | Comentarios |
|----------------------|-------------|------|-------|--------------------------|
| idEngineType (PK) | TINYINT(3) | | | Id del tipo de motor |
| Name | VARCHAR(20) | | | Nombre del tipo de motor |

Tabla engine, contiene el catálogo de motores implementados para la Adquisición y Reconocimiento de datos de tiempo real. Véase a continuación la Tabla 4.5.

Tabla 4.5. Diccionario de datos: Tabla engine

| Columna | Tipo | Nulo | Pred. | Comentarios |
|------------------|-------------|------|-------|--------------|
| idEngine (PK) | SMALLINT(5) | | | Id del motor |

| | | | | |
|--------------|-------------|--|--|----------------------|
| idEngineType | TINYINT(3) | | | Id del tipo de motor |
| Name | VARCHAR(20) | | | Nombre del motor |

Tabla repository, enlista todos los repositorios configurados en el sistema y de los cuales se adquiere los datos de tiempo real de las subestaciones. Véase en la Tabla 4.6.

Tabla 4.6. Diccionario de datos: Tabla repository

| Columna | Tipo | Nulo | Pred. | Comentarios |
|----------------------|--------------|------|-------|---|
| idRepository (PK) | INT(8) | | | Id del repositorio |
| idEngine | SMALLINT(5) | | | Id del motor requerido |
| SuperGroup | VARCHAR(10) | | | Nombre del Worker a cargo |
| DataGroup | VARCHAR(11) | | | Nombre de la Subestación |
| Name | VARCHAR(15) | | | Nombre del repositorio |
| Access | VARCHAR(255) | | | Query con la configuración de acceso |
| Properties | VARCHAR(255) | | | Query con la definición específica de los datos |

| | | | | |
|-------------|--------------|----|---------------------|------------------------------------|
| Enabled | TINYINT(1) | | | Repositorio activo |
| Description | VARCHAR(255) | | | Descripción del repositorio |
| LastUpdate | TIMESTAMP | Si | 0000-00-00 00:00:00 | Última actualización de datos |
| Status | TINYINT(4) | | | Estado del repositorio |
| LastStatus | TIMESTAMP | | CURRENT_TIMESTAMP | Estampa de tiempo de último estado |

Tabla livedata, contiene todos los datos de tiempo real obtenidos de las subestaciones, almacenados en pares Tag / Value, con información de origen y última actualización. Véase a continuación en la Tabla 4.7.

Tabla 4.7. Diccionario de datos: Tabla livedata

| Columna | Tipo | Nulo | Pred. | Comentarios |
|-------------------|-------------|------|-------|---------------------------|
| DataGroup (PK) | VARCHAR(11) | | | Subestación de origen |
| Name (PK) | VARCHAR(15) | | | Nombre del Tag |
| idRepository | INT(8) | | | Id del repositorio origen |
| Data | VARCHAR(11) | | | Valor del Tag |

| | | | | |
|------------|-----------|----|-------------------|-------------------------------|
| LastUpdate | TIMESTAMP | Si | CURRENT_TIMESTAMP | Última actualización de datos |
|------------|-----------|----|-------------------|-------------------------------|

➤ Diagramas de flujo

El proceso inicial del subsistema Monitor :: UTMu mostrado en la Figura 4.8, es el encargado de la inicialización del núcleo del sistema, de la carga de librerías comunes, así como de iniciar el Supervisor.

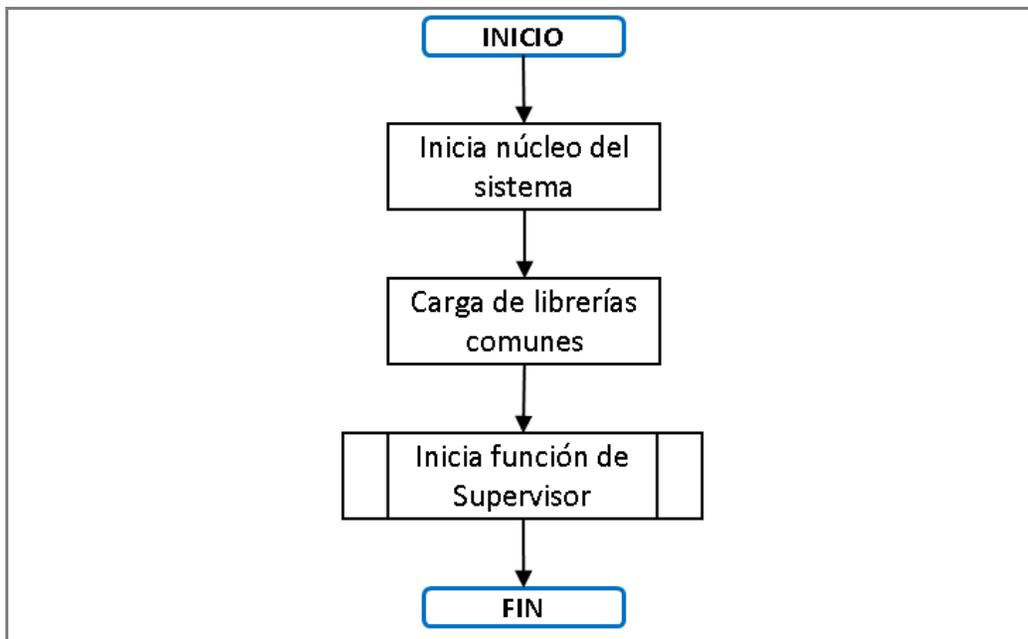


Figura 4.8. Diagrama de flujo del proceso principal de Monitor :: UTMu

Véase en la Figura 4.9 el proceso de ejecución del Supervisor, el cual es el encargado de asegurar la conexión a la base de datos, verificar la adición o eliminación de los SuperGroup, así como de supervisar, ejecutar y terminar los Worker necesarios.

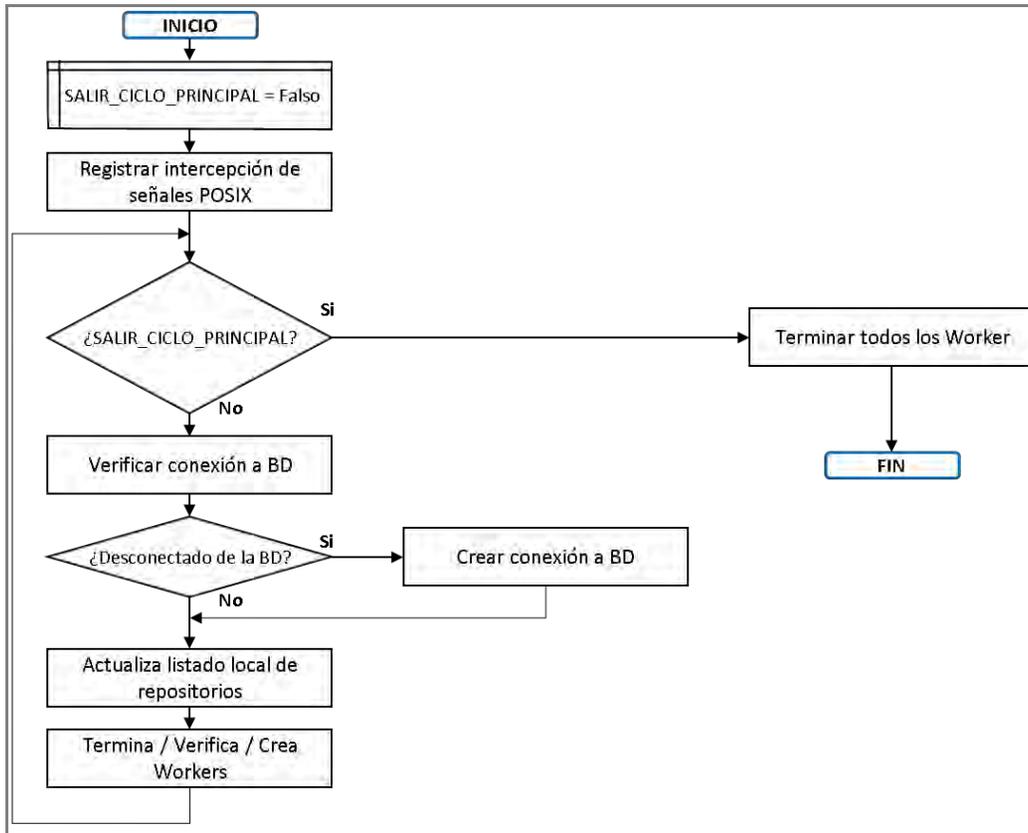


Figura 4.9. Diagrama de flujo del proceso del Supervisor

A continuación, el proceso de ejecución del Worker, visto en la Figura 4.10, el cual es el encargado de la adquisición, reconocimiento y almacenamiento de los datos obtenidos de los repositorios asignados. Para lograr su objetivo, está permanentemente atento al Supervisor para autoterminarse y evitar ser un proceso zombie [Véase Herber, R. (13 de 12 de 1997). *Zombies(5) Defunct, zombie and immortal processes*]. Así mismo administra su propia conexión de base de datos, actualiza la lista de repositorios de su SuperGroup asignado, y realiza la adquisición y reconocimiento de los datos de cada repositorio, para su posterior almacenamiento en la base de datos.

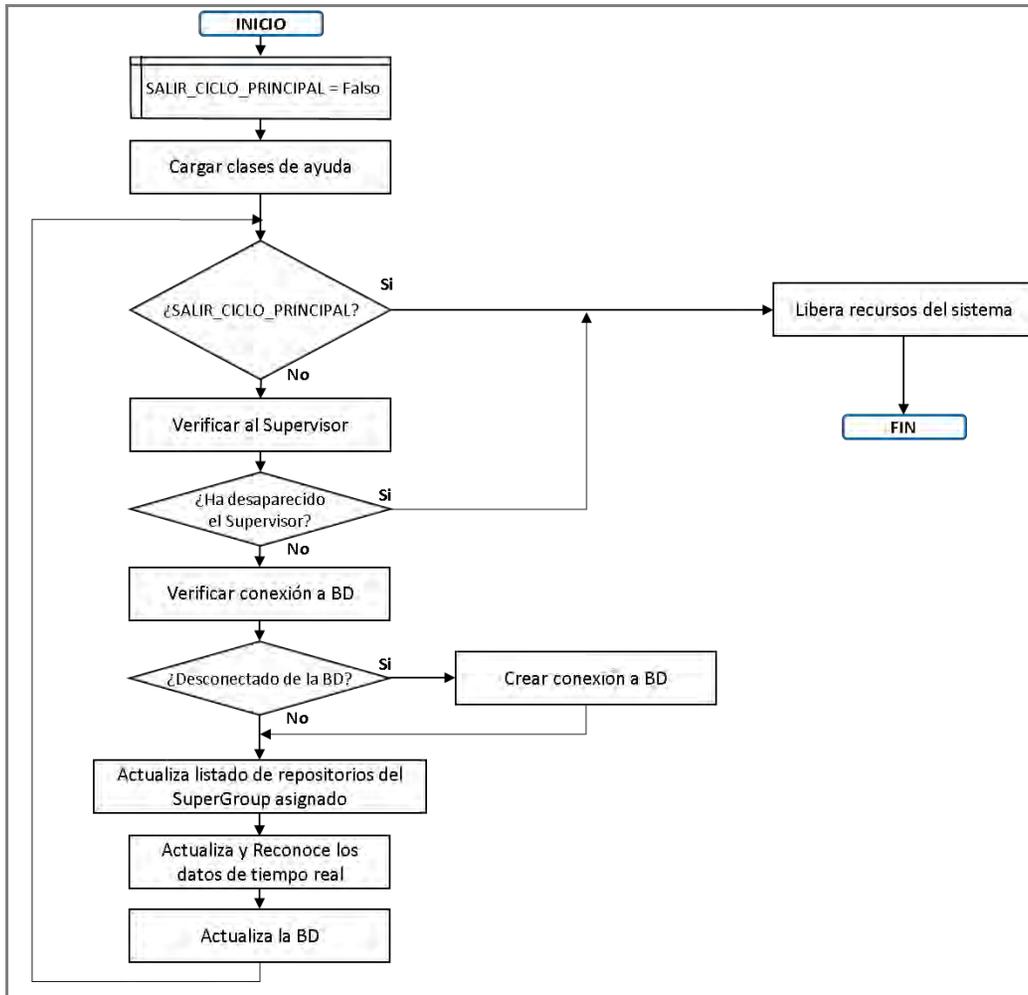


Figura 4.10. Diagrama de flujo del proceso del Worker

4.2.3. Desarrollo

➤ Base de datos

La base de datos de todo sistema es uno de los aspectos más importantes a considerar. Para el Sistema Monitor se utilizó MySQL como sistema administrador de base de datos debido a (1) que es gratuito, y

(2) por su desempeño con el motor de almacenamiento MyISAM (MySQL, 2011) el cual se ve favorecido al no realizar las verificaciones de cumplimiento ACID (Haerder & Reuter, 1983).

➤ System, el Núcleo del Sistema

Se ha desarrollado para el subsistema Monitor :: UTMu un núcleo que contiene la información y las funciones básicas del sistema. Este es un objeto global llamado «System», el cual cuenta con un método «boot» que es invocado al iniciar el sistema. Su función es asignar a las propiedades de acceso común los objetos e instancias que serán utilizados más adelante.

Véase el Anexo B.1. Núcleo del Subsistema Monitor :: UTMu, en el cual se muestra de forma simplificada los bloques que componen dicho núcleo.

➤ Supervisor y su Ciclo Principal de Ejecución

Una de las secciones críticas dentro de la programación de todo proceso, incluido el subsistema Monitor :: UTMu, es la lógica del ciclo principal de ejecución mediante la cual el sistema realiza todas las funciones para las que fue creado.

En el subsistema Monitor :: UTMu, su ciclo principal de ejecución está dividido en tres secciones que están dedicadas a la supervisión de los elementos críticos del sistema:

- Conexión a la base de datos del sistema.
- Configuración de súper grupos.
- Terminación, monitoreo y creación de los Worker.

➤ Worker y su Ciclo Principal de Ejecución

Los Worker del sistema llevan a cabo un serie de actividades destinadas principalmente a ejecutar con éxito el subsistema ARA (Adquisición, Reconocimiento y Almacenamiento), con el objetivo de tener siempre actualizada la información de tiempo real obtenida de los sistemas SCADAWeb y WebSICLE.

Básicamente, dichas actividades se reducen a verificar las siguientes condiciones:

1. Si el Supervisor instruye terminar la ejecución
2. Si el Supervisor ha muerto (su ejecución terminó inesperadamente y no alcanzó a instruir el cese de actividades a los Worker)
3. Si la conexión a la base de datos sigue activa

El punto número 2 citado anteriormente es de suma relevancia, ya que si el Supervisor muere, es necesario terminar la ejecución de todos los procesos de los Worker. Esto es un control que permite asegurarse que el sistema no tenga procesos Worker sin la dirección de un Supervisor, ya que se convertirían en una especie de procesos inmortales (Herber, 1997) que consumen recursos innecesarios.

➤ Subsistema ARA

Uno de los puntos principales respecto al Subsistema ARA que se puede observar en la Figura 4.11, es que la Adquisición y el Reconocimiento de los datos de tiempo real se logra gracias a los motores, dejando el proceso de Almacenamiento de información a un grupo de administradores de datos que realizan la interacción con la base de datos. De esta forma la funcionalidad queda separada en bloques independientes, dejando la puerta abierta al sistema para ampliar el soporte a nuevos motores y sistemas de bases de datos.

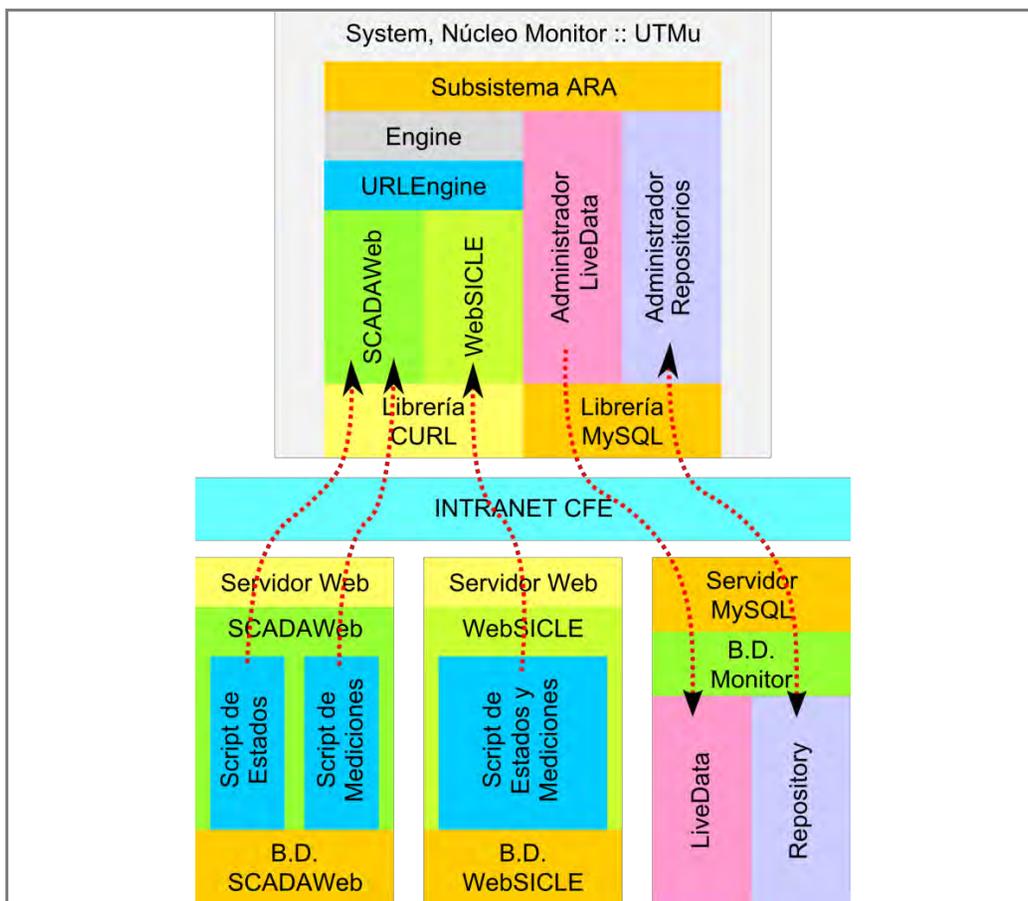


Figura 4.11. Diagrama simplificado del Subsistema ARA y su interacción

➤ Motores de Adquisición y Reconocimiento de Datos

La parte de Adquisición y Reconocimiento se abstrae en bloques de funcionalidad encapsulada, denominados motores, los cuales están diseñados específicamente para cada tipo de sistema (SCADAWeb y WebSICLE), derivado de que las URL en que están disponibles los datos de tiempo real son diferentes. Además, los formatos XML en que entregan la información no tienen comparación entre ambos sistemas.

Tomando lo anterior en cuenta, se creó una clase llamada Engine, la cual es la base para todo tipo de motores que utilizará el sistema. Define el método «getRepositoryData» para obtener los datos de los sistemas remotos, el cual a su vez llama a tres métodos internos que el motor debe implementar:

- `_requestData`, dedicada a adquirir los datos brutos del repositorio indicado.
- `_parseData`, dedicada a realizar el reconocimiento de los datos y ordenarlos en pares tag/value.
- `_translateError`, el cual realiza la traducción de los errores generados por el motor.

En el Anexo B.2. Motores del Subsistema se muestra de forma simplificada la lógica de las clases y la funcionalidad implementada de los dos motores actuales.

4.2.4. Pruebas

Después de la implementación del subsistema Monitor :: UTMu, y se configuró 75 repositorios, se realizaron las siguientes pruebas:

- Pruebas de usabilidad
- Pruebas de funcionalidad
- Pruebas de tolerancia a fallas
- Prueba de desempeño

➤ Pruebas de usabilidad

Estas pruebas consistieron en utilizar los parámetros por línea de comandos obteniendo los siguientes resultados:

- Ejecución de aplicación normal por línea de comandos
- Ejecución de aplicación en modo daemon de Linux
- Habilitar mensajes de depuración del Supervisor
- Habilitar mensajes de depuración del Worker

➤ Pruebas de funcionalidad

Estas pruebas consistieron en iniciar el subsistema Monitor :: UTMu y...

- Creación de nuevos Worker por configuración de un nuevo repositorio con un nuevo SuperGroup.
- Terminación de Worker existentes por no existir más el SuperGroup asignado.

➤ Pruebas de tolerancia a fallas

Estas pruebas consistieron en forzar situaciones de falla tal como se lista a continuación:

- ✓ Ejecución con el servidor de BD fuera de servicio.
- ✓ Ejecución con un servidor de BD en condiciones normales, y después de un tiempo dejar fuera de servicio dicho servidor de BD.
- ✓ Creación de un nuevo Worker al matar alguno de los Worker existentes.
- ✓ Terminación de ejecución de los Worker ante una terminación no esperada del Supervisor.

➤ Pruebas de desempeño

Las pruebas de desempeño del subsistema se obtuvieron al revisar los registros de actividad. A continuación en la Tabla 4.8 se muestran los resultados.

Tabla 4.8. Resultados de desempeño del subsistema Monitor :: UTMu

| Tiempo | Descripción |
|---------------|---|
| 5 s | Tiempo entre la ejecución del subsistema y la actualización completa en base de datos de los datos de tiempo real de las subestaciones. |
| 1 - 4 s | Tiempo entre el inicio de la Adquisición, Reconocimiento y Almacenamiento de los datos de tiempo real. |

4.3. Monitor :: Web, Cliente

A continuación se documenta el prototipo final en forma de versión “as-is” (como está) del subsistema Monitor :: Web.

4.3.1. Análisis

➤ Levantamiento de datos

Los datos de tiempo real a adquirir se encuentran alojados en la base de datos del Sistema Monitor, en la tabla livedata.

La diversidad de navegadores Web que utiliza el usuario debe considerarse como un factor importante, para de esta forma asegurar al máximo la compatibilidad cross-browser (característica de un sistema Web que soporta múltiples navegadores Web).

De acuerdo a una publicación de la Universidad Cardenal Herrera en su blog de internet (consúltese <https://blog.uchceu.es/informatica/ranking-de-navegadores-de-internet-mas-usados-para-2015/>), existen cinco navegadores Web que principalmente usan los usuarios de PC de escritorio (Figura 4.12) y de dispositivos móviles (Figura 4.13):

- Internet Explorer
- Chrome
- Safari
- Firefox
- Opera

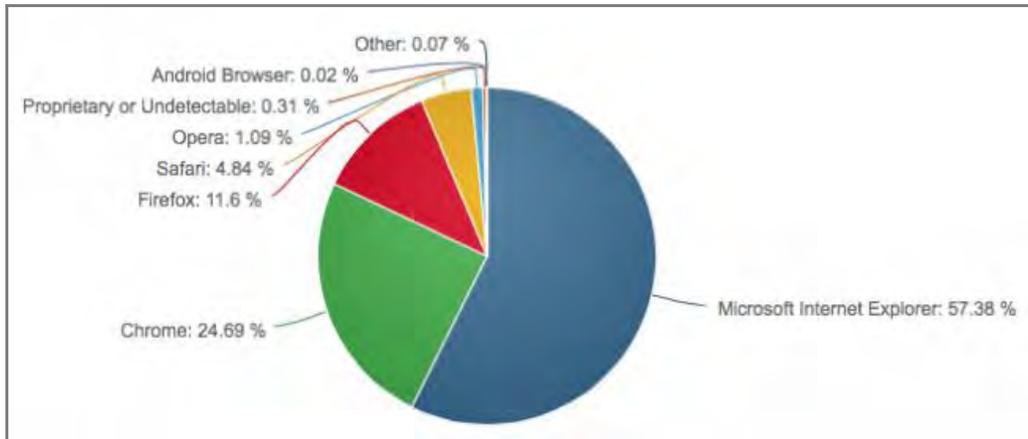


Figura 4.12. Estadísticas de uso de navegadores Web en PC de escritorio

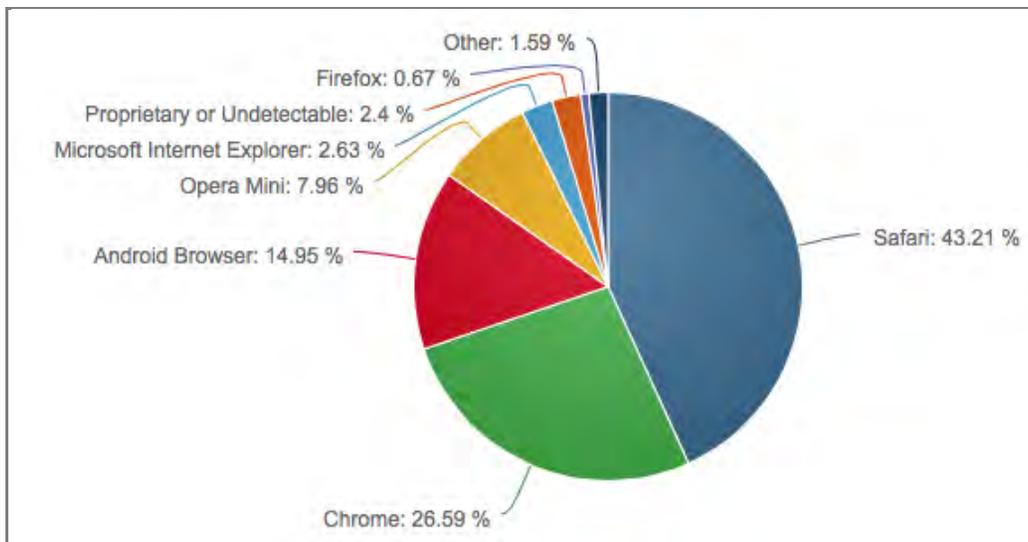


Figura 4.13. Estadísticas de uso de navegadores Web en dispositivos móviles

➤ Definición de requerimientos

En la Tabla 4.9 se muestran los requerimientos funcionales del subsistema Monitor :: Web.

Tabla 4.9. Tabla de requerimientos funcionales de Monitor :: Web

| Identificador del Requerimiento | Descripción |
|--|--|
| SMW_RF1 | El subsistema visualizará la información como una aplicación Web a través de un navegador Web. |
| SMW_RF2 | La interfaz deberá ser fácil de usar por el usuario. |
| SMW_RF3 | El subsistema debe separar el diseño de la lógica, permitiendo cambiar la apariencia sin necesidad de reprogramar la lógica. |
| SMW_RF4 | El subsistema usará los datos de tiempo real de la tabla livedata, misma que siempre está actualizando el subsistema Monitor :: UTMu. |
| SMW_RF5 | El subsistema deberá mostrar al usuario una interfaz principal con la lista de monitores disponibles. |
| SMW_RF6 | Al seleccionar un monitor, el subsistema deberá generar una plantilla HTML mínima que cargue de forma asíncrona toda la información y archivos que requiera. |
| SMW_RF7 | Los diagramas de los monitores deberán estar en formato SVG, y el subsistema deberá desplegarlo usando la tecnología de gráficos vectoriales del navegador. |

Los requerimientos no funcionales del subsistema Monitor :: Web se muestran a continuación en la Tabla 4.10.

Tabla 4.10. Tabla de requerimientos no funcionales de Monitor :: Web

| Clasificación | Identificador del Requerimiento | Descripción |
|----------------------|--|--|
| Interfaz | SMW_RNF1 | El subsistema deberá tener una interfaz sencilla y amigable. |

| | | |
|----------------|-----------|--|
| | SMW_RNF2 | El subsistema deberá ofrecer un grupo de enlaces a los monitores disponibles. |
| | SMW_RNF3 | El subsistema mostrará en los monitores un diagrama unifilar. |
| | SMW_RNF4 | El subsistema ofrecerá enlaces a los sistemas remotos (SCADAWeb y WebSICLE) fuente de los datos de tiempo real. |
| Codificación | SMW_RNF5 | Utilizará un conjunto de clases que actúen como el núcleo del sistema en el servidor. |
| | SMW_RNF6 | Utilizará un conjunto de clases que actúen como el núcleo del sistema en el cliente. |
| Configuración | SMW_RNF7 | Existirá un archivo de configuración del sistema, en el que se especifique acceso a base de datos, ubicación de la librería The Dojo Toolkit, enlaces adicionales y plugins. |
| | SMW_RNF8 | La base de datos contendrá una tabla de los monitores, con la configuración específica. |
| | SMW_RNF9 | La base de datos contendrá una tabla con todos los elementos dinámicos que compongan los monitores, con configuración específica para cada uno. |
| Almacenamiento | SMW_RNF10 | La tabla de los elementos dinámicos deberá incluir un campo con el nombre de la clase que instanciará en el navegador Web. |

| | | |
|------------------------|-----------|--|
| | SMW_RNF11 | La tabla de los elementos dinámicos incluirá un campo con el nombre del Tag que representará de la tabla livedata. |
| Hardware | SMW_RNF12 | Se utilizará el servidor destinado por el área en que surge este proyecto. |
| Límitantes de recursos | SMW_RNF13 | Capacidad de almacenamiento de 80 GB - el consumido el stack LAMP (Linux, Apache, MySQL y PHP). |
| | SMW_RNF14 | Capacidad de 4 GB de RAM. |
| | SMW_RNF15 | Procesador de doble núcleo. |
| Software | SMW_RNF16 | El sistema operativo del servidor utilizará el stack LAMP (Linux, Apache, MySQL y PHP). |
| | SMW_RNF17 | De lado del cliente se utilizará tecnología HTML, CSS, Javascript para la estructura, presentación y comportamiento. |
| | SMW_RNF18 | De lado del cliente se utilizará tecnología vectorial propia del navegador Web, acotada a SVG, Silverlight y VML. |

➤ Modelado de negocio

A continuación en la Figura 4.14 se describe los casos de uso que representan las situaciones generales del subsistema Monitor :: Web. Cada escenario describe una secuencia de eventos y cada secuencia a su vez se ejecuta mediante la interacción de los actores administrador del sistema y los usuarios.

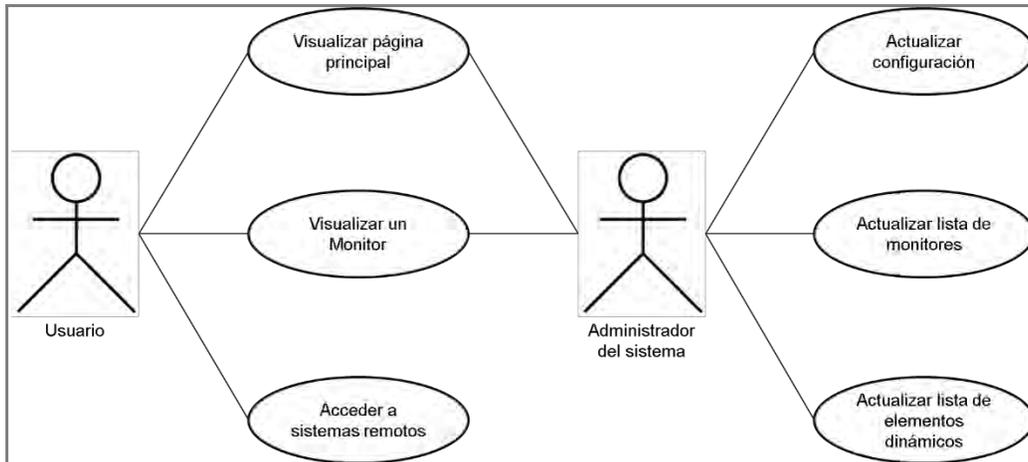


Figura 4.14. Diagrama de caso de uso general del subsistema Monitor :: Web

4.3.2. Diseño

➤ Modelo de datos

A continuación en la Figura 4.15 se muestra el modelo entidad-relación de la base de datos del subsistema Monitor :: Web.

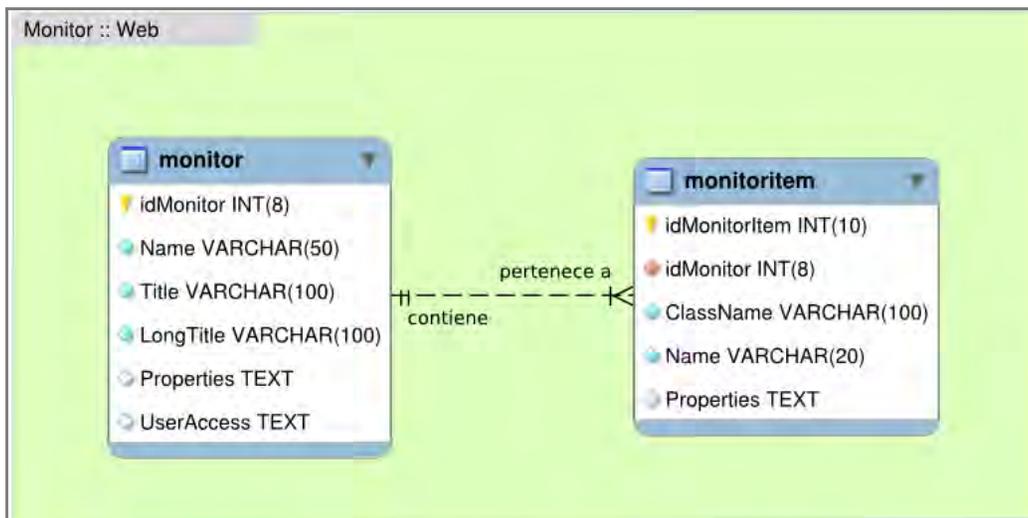


Figura 4.15. Modelo entidad relación del Subsistema Monitor :: Web

➤ Diccionario de datos

La tabla monitor, enlista todos los monitores configurados en el sistema y en los cuales se representará los datos de tiempo real de las subestaciones. Véase a continuación la Tabla 4.11.

Tabla 4.11. Diccionario de datos: Tabla monitor

| Columna | Tipo | Nulo | Pred. | Comentarios |
|-------------------|--------------|-------------|--------------|---|
| idMonitor (PK) | INT(8) | | | Id del monitor |
| Name | VARCHAR(50) | | | Nombre del monitor |
| Title | VARCHAR(100) | | | Título corto del monitor |
| LongTitle | VARCHAR(100) | | | Título completo del monitor |
| Properties | TEXT | | | Configuración de cada Monitor de lado del cliente, en formato JSON |
| UserAccess | TEXT | | | Configuración de cada Monitor de lado del servidor, en formato JSON |

➤ Diagramas de flujo

En la Figura 4.16 se observa el proceso principal del subsistema Monitor :: Web de lado del servidor, encargado de responder con la información requerida por el usuario mediante su interacción con el proceso de lado del cliente.

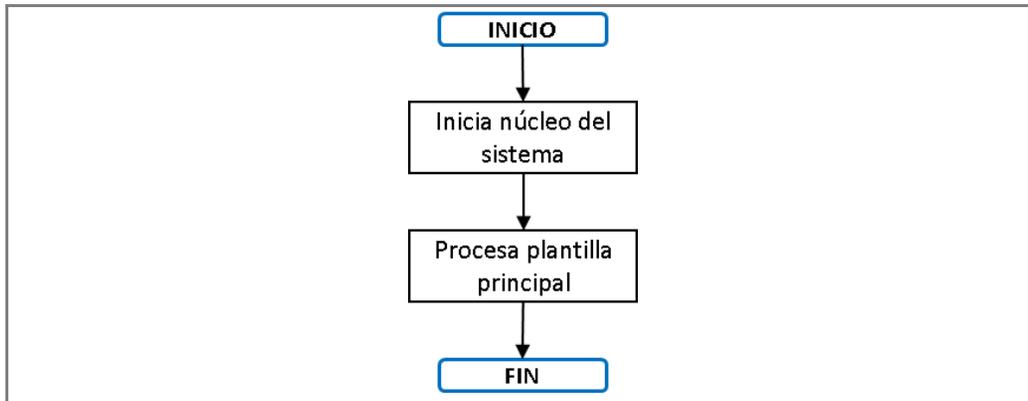


Figura 4.16. Diagrama de flujo del script principal de Monitor :: Web

En la Figura 4.17, se observa el proceso de visualización y datos de los monitores de lado del servidor, encargado de responder con la información, archivos o accesos requeridos por el usuario mediante su interacción con el proceso de lado del cliente.

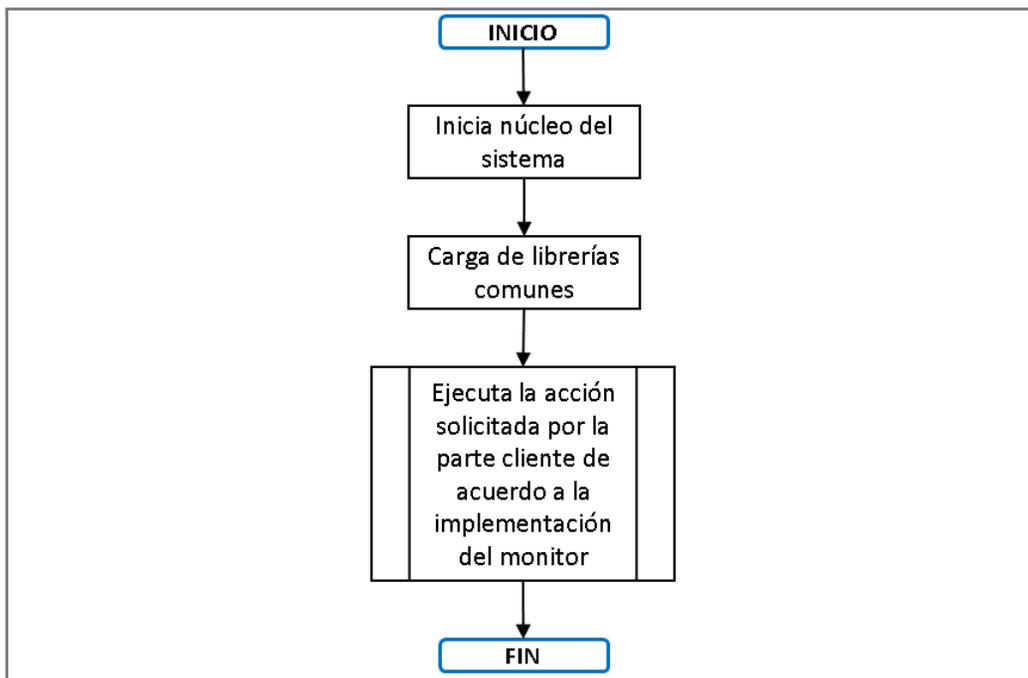


Figura 4.17. Diagrama de flujo del script de visualización y datos de los monitores

4.3.3. Desarrollo

➤ Monitores

Como se observa en la Figura 4.18, son los diagramas unifilares que se configuran en el subsistema y estos contienen una serie de objetos que actualizan apariencia conforme a su programación ante cambios en los datos de tiempo real de las subestaciones.

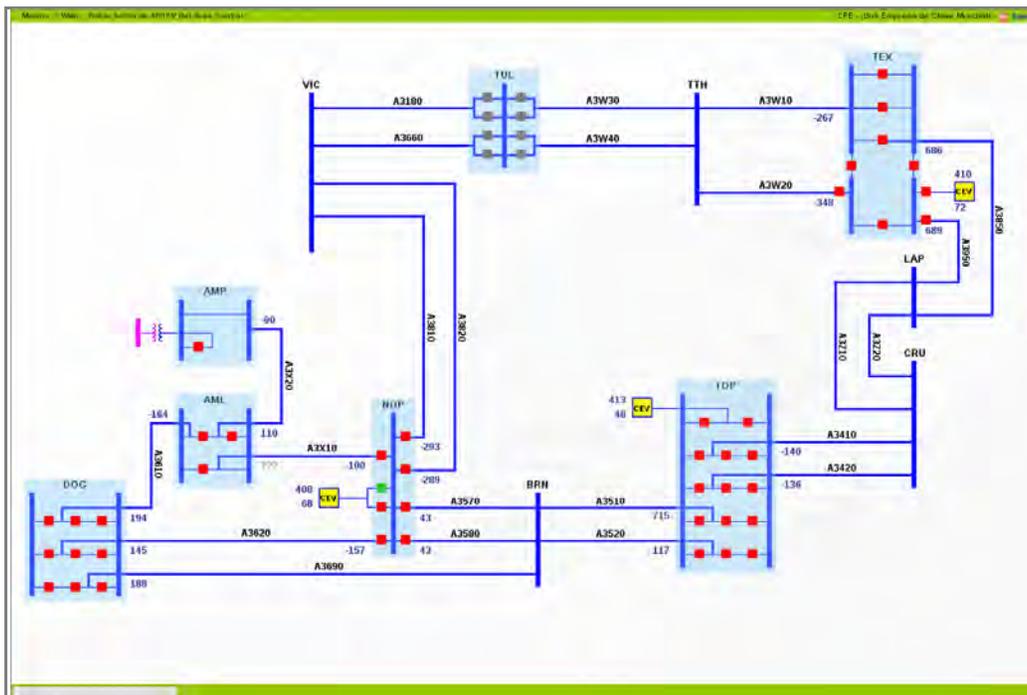


Figura 4.18. Monitor del Doble Anillo de 400 kV del Valle de México

➤ Aplicación lado servidor

La *aplicación servidor* programada en lenguaje PHP se ejecuta «de lado del servidor Web», y es la encargada de generar la estructura HTML del cliente, servir archivos estáticos, actuar como proxy para el correcto funcionamiento del sistema WebSICLE en navegadores Web diferentes

a Internet Explorer, así como adquirir y enviar a la aplicación cliente la gran diversidad de información contenida en la base de datos para su procesamiento.

Como se muestra en la Figura 4.19, el código del script principal es bastante reducido por mantener una base común de código con el subsistema Monitor :: UTM, así como por la alta abstracción de objetos y librerías que se ha conseguido.

```
// Ruta predeterminada de System
define('_BOOT_', 'boot/', true);

// Carga el núcleo del sistema
require _BOOT_.'System.php';
System::$BasePath = dirname(__FILE__).'/';

// Inicia el núcleo del sistema
System::boot();

// Carga la clase de abstracción del Monitor
System::$Core->loadClass('DBMonitor');

// Crea una instancia del Monitor
$monitor =& new DBMonitor;

// Ejecuta la acción solicitada por la aplicación cliente
$monitor->exec(System::$Options['action']);

// Si la ejecución llega aquí es por un error no previsto
System::abort();
```

Figura 4.19. Script principal del Subsistema Monitor :: UTMu

La abstracción utilizada por la aplicación servidor se representa en la Figura 4.20.

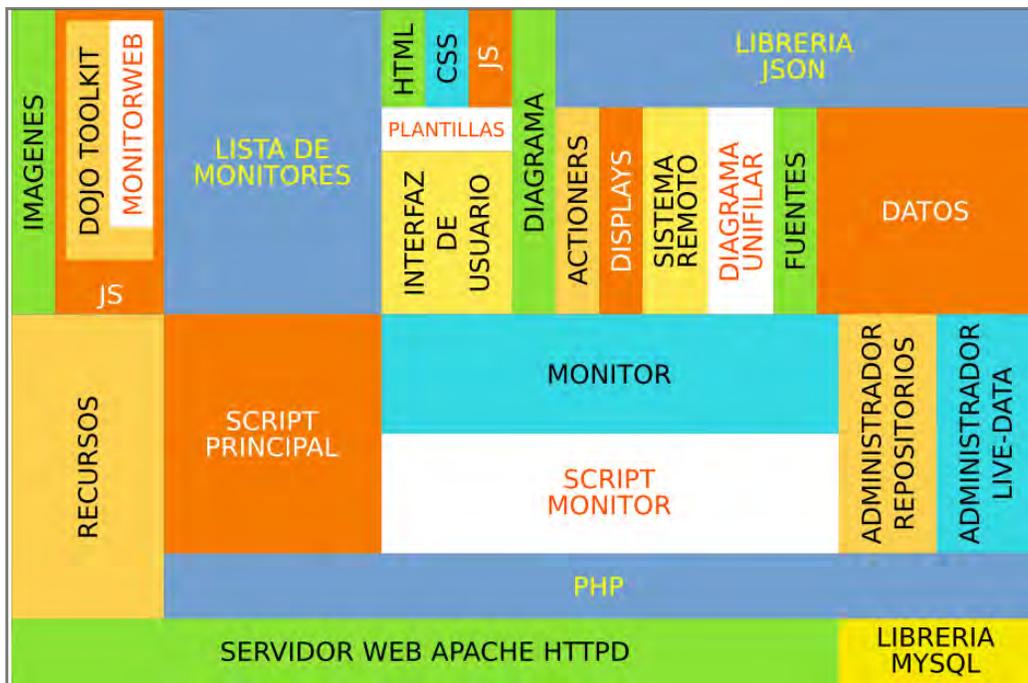


Figura 4.20. Diagrama simplificado de la aplicación servidor de Monitor :: Web

➤ El Núcleo de la Aplicación Servidor

Basado en el núcleo del subsistema Monitor :: UTMu, el núcleo de la aplicación servidor tiene la función de brindar la funcionalidad básica requerida por el sistema, así como ofrecer un punto único de acceso a toda la información y objetos a utilizar.

En el Anexo C.1. Núcleo de la Aplicación Servidor del Subsistema Monitor :: Web se muestra de forma simplificada la lógica del núcleo.

➤ La Aplicación Servidor y la Jerarquía de Monitores

Cuando un Monitor es invocado, el servidor ejecuta un Monitor predeterminado «DBMonitor» que sigue un modelo basado en base de datos con las acciones especificadas por la URL que se deben llevar a cabo, accediendo a información de la base de datos, utilizando plantillas para la interfaz de usuario y diagramas unifilares en formato SVG.

En el Anexo C.2. Jerarquía de Monitores de la Aplicación Servidor se muestra de forma simplificada la lógica usada en las jerarquías de monitores que se ha implementado para la aplicación servidor.

➤ La Aplicación Servidor y los Motores Implementados

Ésta es la aplicación servidor del subsistema Monitor :: Web, el cual es un cliente para el acceso y visualización de los flujos de energía con datos de tiempo real directo en el sistema remoto, y que hace uso de una implementación especial de los Motores definidos por el subsistema Monitor :: UTMu.

La lógica utilizada por las jerarquías de monitores se puede consultar en el Anexo C.3. Motores Implementados por la Aplicación Servidor.

Notas del Motor WebSICLE:

- El sistema WebSICLE sólo funciona en Internet Explorer
- El método «rpc_fixHtml» incluye código HTML para definir la función «getXMLHTTP», la cual obtiene el objeto indicado para

adquisición asíncrona de datos (Gerner, Naramore, Owens, & Warden, 2006)

- El método «rpc_fixJs» se redujo por simplicidad de la lógica y por incluir detalles internos de funcionamiento del sistema WebSICLE

➤ Aplicación Cliente

La aplicación cliente, es la encargada de la interfaz gráfica para el usuario del sistema. Así que utiliza un conjunto de tecnologías Web para poder brindar una gran experiencia a través de una interfaz sencilla y con una visualización única por la nitidez de los gráficos vectoriales usados para los Monitores.

Las tecnologías Web utilizadas son HTML para la estructura, CSS para la presentación y Javascript para el comportamiento o funcionalidad «de lado del cliente», tal como se muestra en la Figura 4.21. Está diseñada para maximizar el uso de llamadas asíncronas de petición de información a la aplicación servidor, generando una experiencia única al ser utilizada de forma fluida por el usuario.



Figura 4.21. Diagrama simplificado de la aplicación cliente de Monitor :: Web

Derivado de la utilización de la librería The Dojo Toolkit (consúltese <http://dojotoolkit.org/>), gran parte de la complejidad de la programación en el navegador Web se ha reducido, y ha permitido soportar la característica cross-browser, de forma que el sistema se puede visualizar en casi cualquier dispositivo.

➤ Plantilla usada para los Monitores

La implementación «TemplateMonitor» especifica que el Monitor a visualizar utiliza una plantilla HTML la cual es usada como la interface inicial del Monitor. Dicha plantilla usa variables que son sustituidas durante la generación. Estas variables se identifican por estar encerradas por tres pares de llaves «{{{ }}}» dentro del código HTML.

Para conocer el código HTML de la plantilla utilizada referirse al Anexo C.4. Plantilla HTML Usada por los Monitores.

Una vez que el navegador Web recibe el código HTML generado, éste inicia la petición de los recursos de imágenes, archivos CSS y Javascript.

➤ La Aplicación Cliente y su Jerarquía de Clases

La aplicación cliente, implementa una jerarquía de clases que permiten al sistema abstraer toda la complejidad. Se pueden consultar en el Anexo C.5. Jerarquía de Clases Lado Cliente.

monitorWeb.userinterface.SvgUserInterface, lleva a cabo la creación del Monitor en el navegador, genera un sistema de mensajes para el usuario, define una serie de canales para publicar / suscribirse a

notificaciones usadas para cambios de estado, propiedades del Monitor, etc.

monitorWeb.fetcher.JsonFetcher, encapsula la comunicación asíncrona única o por intervalos con la aplicación servidor, utilizando el formato implementado para intercambio de datos definido en la clase `JsonMonitor`.

monitorWeb.actioner.SvgActionBox, utiliza un cuadro en el fondo del diagrama, justo debajo de la representación de una subestación, y es utilizado para visualizar de forma gráfica el estado del repositorio. También provee un menú contextual con las opciones definidas por su Motor utilizado.

monitorWeb.display.SvgInterruptor, utiliza el cuadro pequeño asignado dentro del diagrama unifilar, el cual se utiliza para representar el estado de un interruptor de una línea de transmisión en una subestación.

monitorWeb.display.SvgMeter, utiliza un elemento de texto asignado en el diagrama unifilar, el cual se utiliza para desplegar una medición asignada a algún flujo de potencia, nivel de voltaje en una subestación, frecuencia, etc.

MonitorWeb.monitor.SvgMonitor, la cual actúa como una interfaz de control del diagrama unifilar. Realiza la instanciación del «espacio de trabajo donde es cargado el diagrama unifilar» (`workspace`) de clase base «`monitorWeb.svg.ContentPane`», y de todos los elementos dinámicos

«....SvgActionBar», «....SvgInterruptor» y «....SvgMeter» utilizados en el diagrama. En base a los datos de tiempo real que llegan a la aplicación cliente, verifica los criterios para invalidación de datos y publica los cambios de valor y/o estado. Establece y controla las propiedades interactivas de «ajuste automático del zum», «zum manual», «desplazamiento horizontal/vertical» y «control de la adquisición de datos». También es el encargado de mostrar la compatibilidad de navegadores Web de dicha implementación del Monitor.

MonitorWeb.svg.ContentPane, utiliza la clase «dijit.layout.ContentPane» proveida por la librería «The Dojo Toolkit» usada originalmente como un contenedor de visualización HTML, para implementar contenido de gráficos vectoriales en base a las opciones («gfxRenderer: 'svg,silverlight,vml'», véase Anexo C.4. Plantilla HTML Usada por los Monitores), de las cuales SVG es la de mayor prioridad.

La adaptación de dicha clase carga el archivo SVG para procesarlo mediante DOM, y en base a su estructura crea en el navegador los elementos vectoriales con sus propiedades y transformaciones necesarias para la correcta visualización del diagrama unifilar.

4.3.4. Pruebas

Después de la implementación del subsistema Monitor :: Web, se realizarán el ingresar al sistema utilizando diferentes navegadores Web en diferentes dispositivos, logrando su visualización y actualización de los datos de tiempo real.

En la Figura 4.22 se aprecia el subsistema Monitor :: Web, mostrando el resultado del script principal desde en navegador Firefox bajo un ambiente Linux.



Figura 4.22. Prueba cross-browser: Firefox bajo ambiente Linux

A continuación en la Figura 4.23, se observa el script de visualización y datos de los monitores del subsistema Monitor :: Web, el cual actualiza la representación de los interruptores de acuerdo a su estado, y las mediciones varían de acuerdo a los valores obtenidos por el subsistema Monitor :: UTMu.

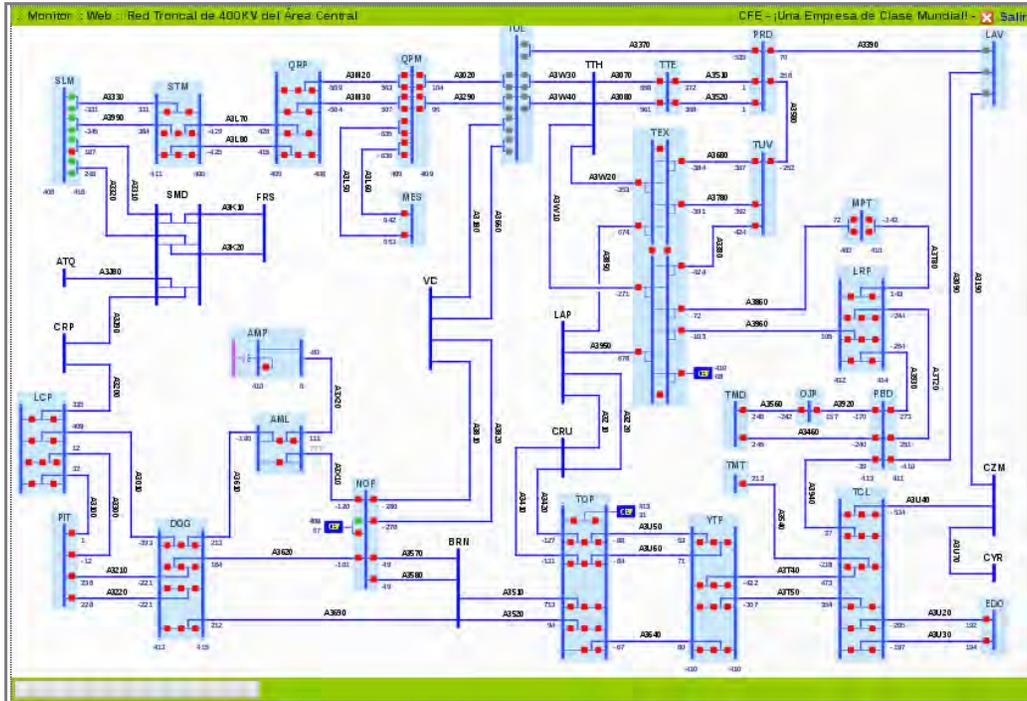


Figura 4.23. Prueba cross-browser: Chrome bajo ambiente Windows

CAPÍTULO 5

RESULTADOS Y TRABAJO

A FUTURO

CAPÍTULO 5. RESULTADOS Y TRABAJO A FUTURO

Como se mostrará a continuación, funciona la adquisición, reconocimiento y almacenamiento de la información de tiempo real de las subestaciones eléctricas por parte del Subsistema Monitor :: UTMu, así como la visualización en navegadores Web en forma de diagramas unifilares de la Red Troncal de 400 kV y la Red Eléctrica de Potencia de 230 kV.

5.1. Registros de Actividad de Monitor :: UTMu

Debido a que este subsistema no tiene una interfaz gráfica de usuario, en el ANEXO D. REGISTROS DE ACTIVIDAD SIMPLIFICADOS DEL SUBSISTEMA MONITOR :: UTMU se muestra 58 registros de actividad que representan los primeros 5 segundos de operación del subsistema Monitor :: UTMu ejecutado con el argumento «--debug» (*Nota: Por seguridad de los sistemas se ha modificado las direcciones reales de acceso. También se ha eliminado la fecha para optimizar el espacio en esta publicación y se dejó únicamente dos SuperGroups*).

En esos 5 segundos de operación, el subsistema realiza la inicialización y la adquisición de 75 repositorios configurados. Esto significa que la información para visualizar los flujos de energía de forma óptima puede tener un retraso aproximado de dicho tiempo en actualizarse en la base de datos, ya que depende de diversos factores que se consiga dicho tiempo.

A continuación en la Tabla 5.1 se muestran los factores antes mencionados.

Tabla 5.1. Factores para un óptimo funcionamiento del Subsistema Monitor :: UTMu

| Factor | Descripción |
|--------|--|
| 1 | Disponibilidad de los enlaces de comunicación. |
| 2 | Latencia de la red derivado al ancho de banda de los canales, tráfico existente y número de saltos para llegar al destino. |
| 3 | Capacidad de procesamiento del servidor del sistema remoto donde se hospeda el sistema SCADAWeb y el WebSICLE. |
| 4 | Capacidad de procesamiento del servidor local, es decir, en el servidor en que se instale el Sistema Monitor. |
| 5 | Saturación de procesamiento por procesos y servicios existentes en los servidores. |
| 6 | Velocidad de lectura y escritura del motor de base de datos empleado del RDBMS. |
| 7 | Parámetros de configuración sobre tiempos de espera e intervalos de actualización. |
| 8 | Cantidad de repositorios configurados de los Super Grupos para tener un Worker destinado al subsistema ARA. |
| 9 | Velocidad de ejecución de los lenguajes de programación y tecnologías utilizados. |

5.3. Monitor del Doble Anillo de 400 kV

En la Figura 5.2 se muestra el monitor del Doble Anillo de 400 kV con estado de los interruptores y los flujos de energía de las subestaciones que se tienen monitoreadas por los sistemas existentes.

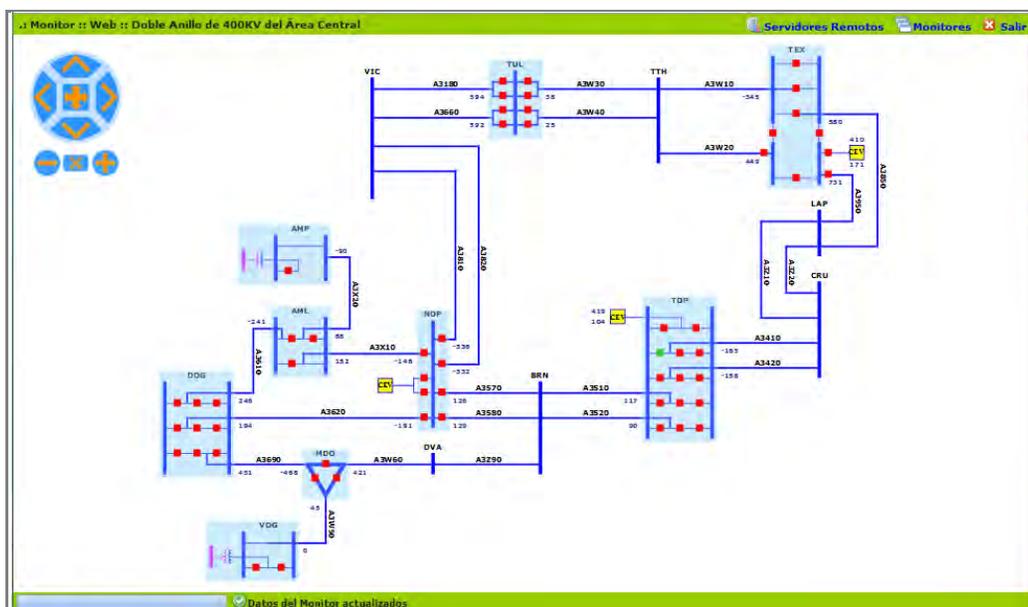


Figura 5.2. Monitor del Doble Anillo de 400 kV

5.4. Monitor de la Red Troncal de 400 kV

En la Figura 5.3 se muestra el monitor de la Red Troncal de 400 kV con estado de los interruptores y los flujos de energía de las subestaciones que se tienen monitoreadas por los sistemas existentes.

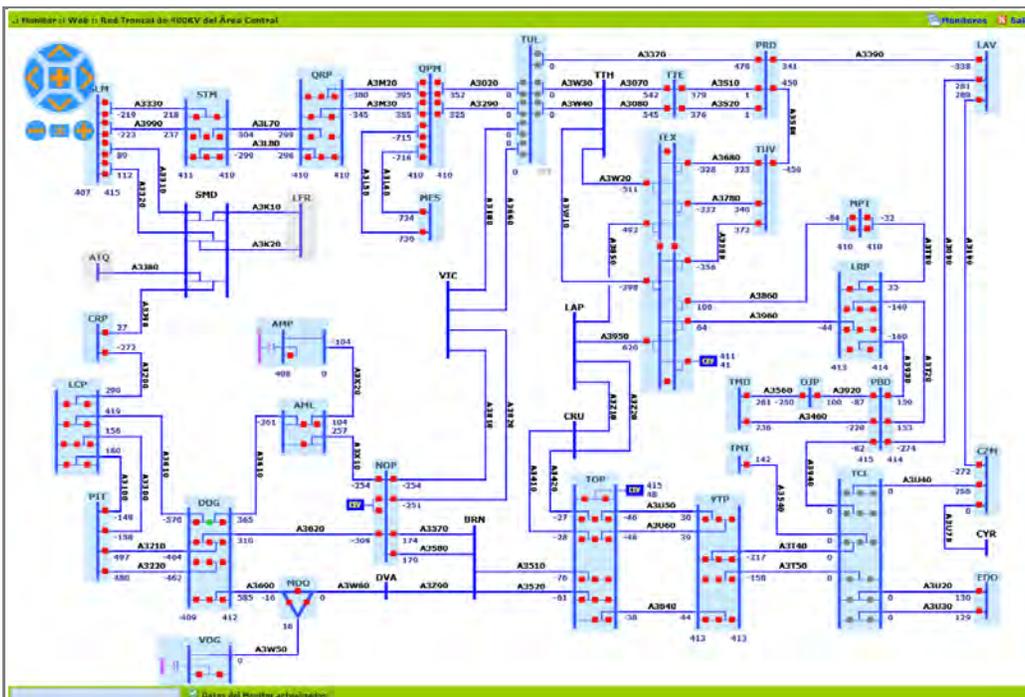


Figura 5.3. Monitor de la Red Troncal de 400 kV

5.5. Monitor de la Red Eléctrica de Potencia de 230 kV

En la Figura 5.4 se muestra el diagrama unifilar de la Red Troncal de 400 kV con estado de los interruptores y los flujos de energía de las subestaciones que se tienen monitoreadas por los sistemas existentes.

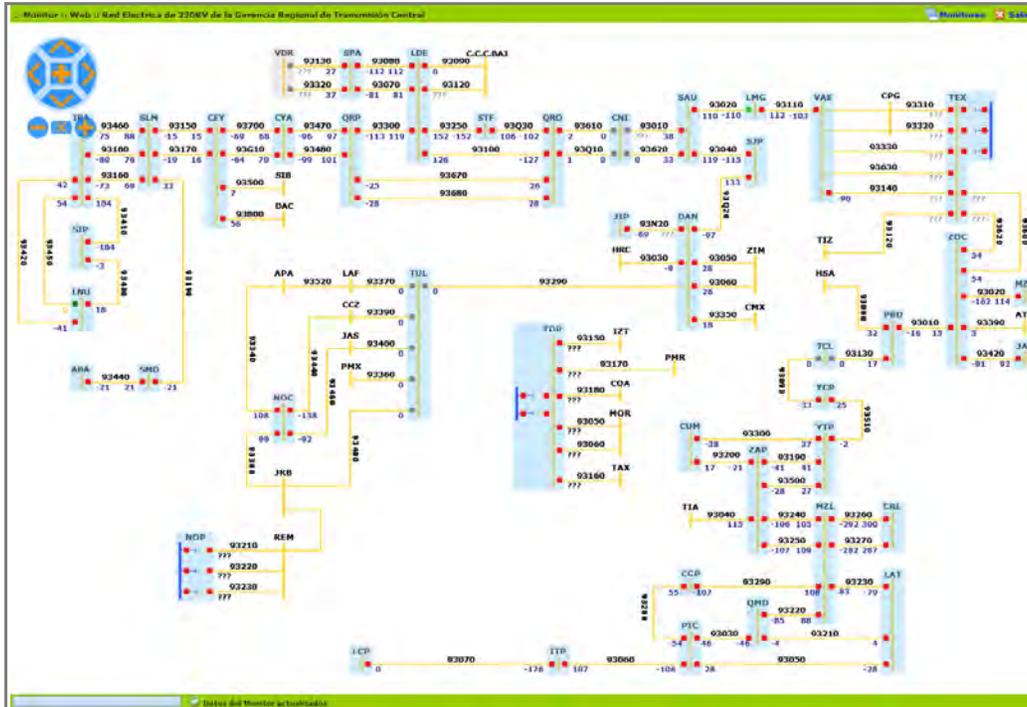


Figura 5.4. Monitor de la Red Eléctrica de Potencia de 230 kV

5.6. Compatibilidad de Navegadores Web

Debido al soporte cross-browser de la librería The Dojo Toolkit, se tiene la siguiente compatibilidad con los navegadores mostrados en la Figura 5.5, en todos ellos adaptándose la tecnología de visualización de gráficos vectoriales.



Figura 5.5. Compatibilidad de Navegadores Web

5.7. Conjuntando Resultados

En los puntos anteriores se mostró evidencia del funcionamiento del Sistema Monitor, el cual como se observa trabaja de acuerdo a lo pactado en los capítulos OBJETIVO GENERAL y OBJETIVOS ESPECÍFICOS.

De igual forma, ante alguna falla el sistema ha permitido al personal directivo reducir la comunicación telefónica con personal de campo y visualizar las maniobras de restablecimiento y la redistribución de los flujos de energía, consiguiendo así generar menos estrés al personal que atiende la falla y con ello que se enfoquen más en sus tareas de análisis y corrección.

Así mismo, de forma rápida y simplificada se observa los flujos de energía en diversas épocas del año, en las cuales, por ejemplo, hay líneas que llegan a transmitir hasta 1200 MW de potencia activa instantánea.

También, se ha podido observar la degradación de los transductores utilizados debido a su nivel de obsolescencia, programando así proyectos anuales para compra y certificación de la calidad en la medición de la energía.

5.8. Trabajo a Futuro

Un sistema de este tipo es de mucha utilidad en la industria eléctrica, ya que la información mostrada te permite tomar decisiones que impactan en el tiempo de respuesta de atención ante fallas, detección de errores de medición, proyección de planes de contingencia por saturación en la

cargabilidad de las líneas de transmisión y acceso unificado a toda la información de tiempo real específica a cada subestación monitoreada.

Sin embargo, la tecnología va avanzando y aparecen nuevas librerías y sistemas que complementan los desarrollos existentes. Para efectos de este sistema, dichos complementos son:

- Utilizar un lenguaje de programación de alto desempeño para el subsistema Monitor :: UTMu.
- Implementar un bus publicista/suscriptor por Web mediante la librería Socket.IO (consúltese la URL: <http://socket.io/>), de forma que no se requiera interrogar periódicamente al servidor mediante AJAX para tener la información actualizada.
- Implementar diagramas vectoriales mediante diseño Web responsivo (Responsive Web Design, consúltese la URL: <http://alistapart.com/article/responsive-Web-design/>), que se adapten en forma a los diferentes tamaños de pantalla desde los que los visualiza el usuario.

CAPÍTULO 6

CONCLUSIONES

CAPÍTULO 6. CONCLUSIONES

En el capítulo anterior se muestra las principales pantallas del sistema en la adquisición y reconocimiento de datos de tiempo real de los sistemas SCADAWeb y WebSICLE, pertenecientes a las subestaciones del área central del país, y su almacenamiento en la base de datos por parte del Subsistema Monitor :: UTMu.

Asimismo, los datos en tiempo real son visualizados en navegadores Web por medio del subsistema Monitor :: Web, el cual de forma dinámica actualiza los estados de interruptores y los flujos de la potencia eléctrica en las líneas de transmisión en cada unos de los diagramas unifilares de la Red Troncal de 400 kV, Doble Anillo del Área Central y la Red Eléctrica de Potencia de 230 kV por los que estén navegando los clientes.

La implantación de este sistema, ha permitido la oportuna toma de decisiones del personal directivo y de mantenimiento de la siguiente forma:

- En condiciones operación normal del SEN, se aprecian de forma rápida los problemas relacionados a la medición que existe en las subestaciones, de tal forma que el personal de mantenimiento puede aplicar las actividades correctivas necesarias, y en caso de deberse a obsolescencia tecnológica o degradación de los elementos que intervienen en la digitalización de la medición, el personal directivo cuenta con un soporte clave en la justificación de compra y modernización de dichos elementos.

- En condiciones de contingencia, el personal de mantenimiento tiene acceso de forma fácil a los servicios ofrecidos por los sistemas fuente (SCADAWeb y WebSICLE), contando con la información requerida para un análisis entre especialistas y toma de decisión de las acciones correctivas necesarias. El personal directivo interviene en menos ocasiones al personal de mantenimiento, favoreciendo un enfoque completo en la contingencia por parte de éstos últimos. Asimismo, el personal directivo también cuenta con la información necesaria para la gestión de recursos y apoyos necesarios para acelerar el restablecimiento de la parte afectada del SEN.

Con todos estos elementos, queda validada la HIPÓTESIS de que el desarrollar un sistema de monitoreo de datos en tiempo real de las subestaciones eléctricas del área central del país, permitirá dotar al personal directivo y de mantenimiento de la información necesaria para la toma de decisiones en condiciones normales, durante una falla en el Sistema Eléctrico de Potencia y en las acciones de mitigación y restablecimiento de dichas fallas.

BIBLIOGRAFÍA

Abelson, H., Sussman, G. J., & Sussman, J. (1996). *Structure and Interpretation of Computer Programs* (2 ed.). Cambridge, Massachusetts: The MIT Press.

ANSI, & IEEE. (28 de Junio de 2014). *ANSI C37.2 Standard for Electrical Power System Device Function Numbers, Acronyms, and Contact Designations*. Obtenido de Standard ANSI C37.2-2008: http://en.wikipedia.org/wiki/ANSI_device_numbers

Braegger, R. (2005). *FACTS e a Estabilidade, Dinâmica e Estabilidade de Sistemas Eléctricos*. FEUP.

Cantone, D. (2006). *IMPLEMENTACIÓN Y DEBUGGING. USERS.CODE*.

CENACE. (2014). *Reglas de Despacho y Operación del Sistema Eléctrico Nacional*. México: Comisión Federal de Electricidad.

CFE. (Marzo de 2010). *CFE G0000-34 Sistema de Información y Control Local de Estación*. Obtenido de Especificación CFE G0000-34: http://lapem.cfe.gob.mx/normas/carga_pagina.asp?pag=G0000-34.pdf

CFE. (2011). *Reglamento de Seguridad e Higiene en el trabajo, Capítulo 800*. México: CFE.

CFE. (12 de Julio de 2012). *NRF-022-CFE Interruptores de potencia de 72.5 kV a 420 kV*. Obtenido de Norma NRF-022-CFE-2010: <http://lapem.cfe.gob.mx/normas/nrf/pdfs/d/NRF-022.pdf>

CFE. (18 de Junio de 2014). *Sitio oficial de la CFE*. Obtenido de Comisión Federal de Electricidad: <http://www.cfe.gob.mx/>

Codd, E. F. (1970). *A Relational Model of Data for Large Shared Data Banks*.

Coronel-Mercado, L. E., Rodríguez-Rivas, J. J., & Huerta-González, P. F. (julio-septiembre de 2013). Simulación del compensador estático de VAR de la subestación Culiacán Tres para el análisis de transitorios. *Ingeniería Investigación y Tecnología*, XIV(3), 423-436.

ECMA. (Junio de 2011). *ECMA-262 ECMAScript Language Specification*. Obtenido de Standard ECMA-262: <http://www.ecma-international.org/ecma-262/5.1/>

ECMA. (Octubre de 2013). *ECMA-404 The JSON Data Interchange Format*. Obtenido de Standard ECMA-404: <http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-404.pdf>

Ericsson Computer Science Laboratory. (2014). *Erlang Programming Language*. Obtenido de Sitio oficial de Erlang: <http://www.erlang.org/>

Gerner, J., Naramore, E., Owens, M. L., & Warden, M. (2006). *Profesional LAMP Desarrollo Web con Linux, Apache, MySQL y PHP 5*. Anaya Multimedia.

Haerder, T., & Reuter, A. (Diciembre de 1983). Principles of transaction-oriented database recovery. *ACM Computing Surveys (CSUR)*, 15(4), 287-317. doi:10.1145/289.291

Herber, R. (13 de 12 de 1997). *Zombies(5) Defunct, zombie and immortal processes*. Obtenido de UNIX System V (Concepts): http://www-cdf.fnal.gov/offline/UNIX_Concepts/concepts.zombies.txt

IEEE. (2008). *IEEE 1003.1-2008 Standard for Information Technology - Portable Operating System Interface (POSIX(R))*. Obtenido de Standard IEEE 1003.1-2008: <http://standards.ieee.org/findstds/standard/1003.1-2008.html>

IEEE. (2012). *IEEE SA - 1815-2012 - IEEE Standard for Electric Power Systems Communications-Distributed Network Protocol (DNP3)*. Obtenido de Standard IEEE 1815-2012: <http://standards.ieee.org/findstds/standard/1815-2012.html>

ISO. (1986). *ISO 8879:1986 Standard Generalized Markup Language (SGML)*. Obtenido de Standard ISO 8879:1986: http://www.iso.org/iso/catalogue_detail.htm?csnumber=16387

ISO, & IEC. (2011). *ISO/IEC 16262:2011 ECMAScript scripting language*. Obtenido de Standard ISO/IEC 16262:

http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=55755

ISO, & IEC. (2011). *ISO/IEC 9075-1:2011 SQL -- Part 1: Framework (SQL/Framework)*. Obtenido de Standard ISO/IEC 9075-1:2011 SQL:

http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=53681

ISO, IEC, & IEEE. (15 de Septiembre de 2009). *ISO/IEC/IEEE 9945:2009 Portable Operating System Interface (POSIX) Base Specifications, Issue 7*. Obtenido de Standard ISO/IEC/IEEE 9945:2009: http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=50516

Marcotte, E. (25 de Mayo de 2010). *Responsive Web Design*. Obtenido de <http://alistapart.com/article/responsive-Web-design/>

Maxwell, J. C. (1881). *A Treatise on Electricity and Magnetism*. Michigan: Clarendon Press.

MySQL. (2011). *MySQL MyISAM El motor de almacenamiento MyISAM*. Obtenido de Reference MySQL MyISAM: <http://dev.mysql.com/doc/refman/5.0/es/myisam-storage-engine.html>

Perl. (7 de Enero de 2014). *Perl 5 perlre (Perl Regular Expressions)*.
Obtenido de Documentation Perl5 perlre:
<http://perldoc.perl.org/perlre.html>

RFC, I. (Marzo de 2014). *RFC-7159 The JavaScript Object Notation (JSON) Data Interchange Format*. Obtenido de Standard RFC-7159: <http://tools.ietf.org/html/rfc7159>

RFC, N. W. (Marzo de 1998). *RFC-2318 The text/css Media Type*.
Obtenido de Standard RFC-2318: <http://tools.ietf.org/html/rfc2318>

RFC, N. W. (Junio de 1999). *RFC-2616 Hypertext Transfer Protocol -- HTTP/1.1*. Obtenido de Standard RFC-2616:
<http://tools.ietf.org/html/rfc2616>

RFC, N. W. (Enero de 2005). *RFC-3986 Uniform Resource Identifier (URI) Generic Syntax*. Obtenido de Standard RFC-3986 URI:
<http://tools.ietf.org/html/rfc3986>

Secretaría de Energía. (29 de noviembre de 2012). *NOM-001-SEDE-2012: Instalaciones Eléctricas (utilización)*. Obtenido de Diario Oficial de la Federación, Tomo DCCX No. 22, Edición Vespertina:
<http://dof.gob.mx/index.php?year=2012&month=11&day=29>

The PHP Group. (2014). *PHP: Hypertext Preprocessor*. Obtenido de Sitio oficial de PHP: <http://www.php.net/>

- W3C. (24 de Diciembre de 1999). *HTML 4.01 Specification*. Obtenido de World Wide Web Consortium: <http://www.w3.org/TR/html4/>
- W3C. (7 de Abril de 2004). *Document Object Model (DOM) Level 3 Core Specification*. Obtenido de Specification DOM 3 1.0: <http://www.w3.org/TR/2004/REC-DOM-Level-3-Core-20040407/>
- W3C. (11 de febrero de 2004). *Web Services Glossary*. Obtenido de World Wide Web Consortium (W3C): <http://www.w3.org/TR/2004/NOTE-ws-gloss-20040211/#Webservice>
- W3C. (29 de Septiembre de 2006). *Extensible Markup Language (XML) 1.1 (Second Edition)*. Obtenido de World Wide Web Consortium (W3C): <http://www.w3.org/TR/xml11/>
- W3C. (5 de Febrero de 2008). *Extensible Markup Language (XML) 1.0 (Fifth Edition)*. Obtenido de World Wide Web Consortium (W3C): <http://www.w3.org/TR/2008/PER-xml-20080205/>
- W3C. (1 de Diciembre de 2008). *Synchronized Multimedia Integration Language (SMIL 3.0)*. Obtenido de Specification SMIL 3.0: <http://www.w3.org/TR/smil/>
- W3C. (16 de Agosto de 2011). *Scalable Vector Graphics (SVG) 1.1 (Second Edition)*. Obtenido de Specification SVG: <http://www.w3.org/TR/SVG/>

W3C. (17 de diciembre de 2012). *Web Design and Applications » HTML && CSS*. Obtenido de World Wide Web Consortium (W3C): <http://www.w3.org/standards/Webdesign/htmlcss>

W3C. (s.f.). *Web Design and Applications*. Obtenido de World Wide Web Consortium (W3C): <http://www.w3.org/standards/Webdesign/>

W3C. (s.f.). *Web Design and Applications » Graphics*. Obtenido de World Wide Web Consortium (W3C): <http://www.w3.org/standards/Webdesign/graphics>

W3C. (s.f.). *XML Technology » XML Essentials*. Obtenido de World Wide Web Consortium (W3C): <http://www.w3.org/standards/xml/core>

ÍNDICE DE FIGURAS

| | |
|---|----|
| Figura 1.1. Comisión Federal de Electricidad | 21 |
| Figura 1.2. Organigrama de la Dirección General de CFE | 22 |
| Figura 1.3. Organigrama de la Dirección de Operación de CFE..... | 23 |
| Figura 1.4. Proceso de Generación, Transmisión, Distribución y Control de Energía | 24 |
| Figura 1.5. Central generadora | 25 |
| Figura 1.6. Generadores de una central hidroeléctrica | 25 |
| Figura 1.7. Generador eléctrico | 26 |
| Figura 1.8. Líneas de transmisión eléctrica | 27 |
| Figura 1.9. Subestación eléctrica..... | 28 |
| Figura 1.10. Transformador de potencia | 29 |
| Figura 1.11. Proceso de Transmisión y Distribución en el Valle de México y área metropolitana | 30 |
| Figura 1.12. Subestación eléctrica de 23 kV..... | 31 |
| Figura 1.13. Circuito de 13.8 kV por postes..... | 31 |

| | |
|--|----|
| Figura 1.14. Entrega de energía de Distribución al Usuario | 32 |
| Figura 1.15. Fotografía del Centro de Control del Área de Control Central | 34 |
| Figura 1.16. Arco eléctrico producido al cerrar cuchillas de 230 kV con potencial durante maniobra normal..... | 35 |
| Figura 1.17. Gabinetes de alarmas de un interruptor de potencia | 38 |
| Figura 1.18. Tablilla de conexiones | 38 |
| Figura 1.19. Diagrama de una UTR..... | 39 |
| Figura 1.20. Diagrama de un SICLE | 41 |
| Figura 1.21. Servicios Web de los sistemas de monitoreo en línea..... | 44 |
| Figura 1.22. Sector Sur de la Zona de Transmisión Metropolitana | 45 |
| Figura 1.23. Mapa de las nueve Gerencias Regionales de Transmisión | 46 |
| Figura 2.1. Cliente del sistema SCADAWeb..... | 53 |
| Figura 2.2. SCADAWeb como diagrama unifilar y secuencia de eventos | 56 |
| Figura 3.1. Ejemplo de una URL..... | 63 |

| | |
|---|-----|
| Figura 3.2. Diferencia de imágenes Raster y Vectoriales | 68 |
| Figura 3.3. Modelo de Estándares Web, por The Open Web | 78 |
| Figura 3.4. Arquitectura de Aplicaciones Web | 79 |
| Figura 3.5. Porcentajes de uso de servidores Web | 80 |
| Figura 3.6. Ciclo de vida lineal | 82 |
| Figura 3.7. Ciclo de vida por prototipos | 82 |
| Figura 3.8. Ciclo de vida en espiral | 83 |
| Figura 3.9. Componentes de un Diagrama Unifilar | 84 |
| Figura 4.1. Diagrama simplificado del sistema Monitor | 92 |
| Figura 4.2. Formato de estados del Sistema SCADAWeb | 94 |
| Figura 4.3. Formato de mediciones del Sistema SCADAWeb | 94 |
| Figura 4.4. Formato XML número 1 del Sistema WebSICLE | 96 |
| Figura 4.5. Formato XML número 2 del Sistema WebSICLE | 97 |
| Figura 4.6. Diagrama de caso de uso general del subsistema Monitor :: UTMu | 101 |

| | |
|--|-----|
| Figura 4.7. Modelo entidad relación del Subsistema Monitor :: UTMu | 102 |
| Figura 4.8. Diagrama de flujo del proceso principal de Monitor :: UTMu | 106 |
| Figura 4.9. Diagrama de flujo del proceso del Supervisor | 107 |
| Figura 4.10. Diagrama de flujo del proceso del Worker | 108 |
| Figura 4.11. Diagrama simplificado del Subsistema ARA y su interacción | 111 |
| Figura 4.12. Estadísticas de uso de navegadores Web en PC de escritorio | 116 |
| Figura 4.13. Estadísticas de uso de navegadores Web en dispositivos móviles..... | 116 |
| Figura 4.14. Diagrama de caso de uso general del subsistema Monitor :: Web | 120 |
| Figura 4.15. Modelo entidad relación del Subsistema Monitor :: Web | 120 |
| Figura 4.16. Diagrama de flujo del script principal de Monitor :: Web . | 122 |
| Figura 4.17. Diagrama de flujo del script de visualización y datos de los monitores | 122 |
| Figura 4.18. Monitor del Doble Anillo de 400 kV del Valle de México . | 123 |

| | |
|---|-----|
| Figura 4.19. Script principal del Subsistema Monitor :: UTMu | 124 |
| Figura 4.20. Diagrama simplificado de la aplicación servidor de Monitor :: Web | 125 |
| Figura 4.21. Diagrama simplificado de la aplicación cliente de Monitor :: Web | 127 |
| Figura 4.22. Prueba cross-browser: Firefox bajo ambiente Linux | 131 |
| Figura 4.23. Prueba cross-browser: Chrome bajo ambiente Windows | 132 |
| Figura 5.1. Página Principal en Monitor :: Web..... | 137 |
| Figura 5.2. Monitor del Doble Anillo de 400 kV | 138 |
| Figura 5.3. Monitor de la Red Troncal de 400 kV..... | 139 |
| Figura 5.4. Monitor de la Red Eléctrica de Potencia de 230 kV | 140 |
| Figura 5.5. Compatibilidad de Navegadores Web..... | 140 |

ÍNDICE DE TABLAS

| | |
|---|-----|
| Tabla 1.1. Listado de Equipo Primario de una subestación | 36 |
| Tabla 1.2. Listado de Equipo Auxiliar de una subestación..... | 36 |
| Tabla 1.3. Variables eléctricas monitoreadas por el CENACE | 37 |
| Tabla 1.4. Especialidades técnicas en Transmisión | 46 |
| Tabla 2.1. Principales tablas en bases de datos del SCADAWeb y WebSICLE | 52 |
| Tabla 3.1. Tipos de URI | 63 |
| Tabla 3.2. Métodos HTTP más comunes..... | 65 |
| Tabla 3.3. Colores de elementos dinámicos en diagramas unifilares ... | 85 |
| Tabla 3.4. Caracteres de identificación de los niveles de tensión..... | 87 |
| Tabla 4.1. Tabla de requerimientos funcionales de Monitor :: UTMu.... | 98 |
| Tabla 4.2. Tabla de requerimientos no funcionales de Monitor :: UTMu | 99 |
| Tabla 4.3. Diccionario de datos: Tabla connectiontester | 103 |
| Tabla 4.4. Diccionario de datos: Tabla enginetype | 103 |

| | |
|---|-----|
| Tabla 4.5. Diccionario de datos: Tabla engine | 103 |
| Tabla 4.6. Diccionario de datos: Tabla repository | 104 |
| Tabla 4.7. Diccionario de datos: Tabla livedata | 105 |
| Tabla 4.8. Resultados de desempeño del subsistema Monitor :: UTMu | 114 |
| Tabla 4.9. Tabla de requerimientos funcionales de Monitor :: Web | 117 |
| Tabla 4.10. Tabla de requerimientos no funcionales de Monitor :: Web | 117 |
| Tabla 4.11. Diccionario de datos: Tabla monitor..... | 121 |
| Tabla 5.1. Factores para un óptimo funcionamiento del Subsistema Monitor :: UTMu | 136 |

ANEXO A

GLOSARIO

ANEXO A. GLOSARIO

Cambiador de TAP

Es un dispositivo que permite al banco cambiar la derivación para ajustar la relación de transformación, es decir, ajustar el voltaje de salida. (Laughton & Warne, 2003).

Cargabilidad

Es la capacidad máxima de potencia eléctrica que se puede transmitir por una línea de transmisión manteniendo sus condiciones normales de operación. (CENACE, 2014).

CSS, Cascading Style Sheets

Lenguaje de hojas de estilo el cual describe la presentación de documentos estructurados, definido por el estándar (RFC N. W., RFC-2318 The text/css Media Type, 1998).

CENACE, Centro Nacional de Control de Energía

Tiene como función principal planear, dirigir y supervisar la operación del sistema eléctrico del país para el logro de los objetivos básicos que son la seguridad, continuidad, calidad y economía del servicio eléctrico. Tiene delegadas además, las funciones relativas a la operación de las instalaciones y equipos, que son administrados conjuntamente con otros recursos como los energéticos, los escurrimientos hidráulicos y el personal de operación. Así mismo, coordina y supervisa la operación de las interconexiones eléctricas con compañías nacionales, extranjeras, así

como con los Generadores o Permisos interconectados con la red de CFE. (CENACE, 2014).

Centro de Control

Es el área constituida para la operación de una parte del Sistema Eléctrico que se le asigna para ese fin. (CENACE, 2014).

CRC, Comprobación de Redundancia Cíclica

Es un código de verificación de datos para detectar errores causados por ruido en canales de transmisión.

CFE, Comisión Federal de Electricidad

La Comisión Federal de Electricidad es un organismo descentralizado de la Administración Pública Federal, con personalidad jurídica y patrimonio propio, que tiene por objeto la planeación del Sistema Eléctrico Nacional, así como la generación, conducción, transformación, distribución y venta de energía eléctrica para la prestación del servicio público y la realización de todas las obras, instalaciones y trabajos que se requieran para el cumplimiento de su objeto, de conformidad con lo dispuesto en la Ley del Servicio Público de Energía Eléctrica, la Ley Orgánica de la Administración Pública Federal, la Ley Federal de las Entidades Paraestatales y demás ordenamientos aplicables. (CFE, 2014).

CEV, Compensador Estático de VAR

Uno de los dispositivos FACTS (flexible AC transmisión systems) más importantes es el CEV o compensador estático de VAR, el cual puede ser utilizado para el control del voltaje y mejorar la estabilidad de los Sistemas Eléctricos de Potencia (SEP). Un CEV puede aportar la

potencia reactiva necesaria para el control dinámico del voltaje, compensando así los desvíos de potencia reactiva de la red, provocados por grandes variaciones de carga, que podrían provocar variaciones inaceptables de voltaje, inestabilidades o hasta colapso del sistema (Braegger, 2005). El CEV también es usado para amortiguar las oscilaciones de potencia, mejorar la estabilidad transitoria y reducir las pérdidas en el sistema por medio del control de la potencia reactiva.

CCL o CI, Consola de Control

Una consola de control es un equipo que explota los datos de una UTR o de los MCAD para su visualización en listados de alarmas y diagramas que permiten realizar la operación de la subestación. A través de dichos diagramas se puede llevar a cabo mandos de apertura y cierre en interruptores y cuchillas. Hay dos tipos de consolas de control:

- *Consola de Control Local*, es la consola que permite la operación de la instalación por aquel personal dedicado a la operación, o bien, por personal de mantenimiento.
- *Consola de Ingeniería*, es una consola de control dedicada al acceso de los dispositivos del SICLE y su configuración.

Contacto seco

Se refiere a aquel «contacto de un relé» al cual uno debe suministrarle un voltaje, también conocido dicho voltaje como «común».

A, Corriente

Unidad de intensidad de corriente eléctrica del Sistema Internacional equivalente a la intensidad de la corriente que, al circular por dos

conductores paralelos, rectilíneos, de longitud infinita, de sección circular despreciable y colocados a la distancia de un metro uno de otro en el vacío, origina entre dichos conductores una fuerza de dos diezmillonésimas de newton por cada metro de conductor. Se mide en ampere o amperio (A).

Cuchillas

Son los dispositivos cuya función consiste en conectar y desconectar un equipo sin carga. (CENACE, 2014).

Diagrama unifilar

Es la gráfica de una red o un arreglo eléctrico en la que los conductores y/o fases se representan mediante una única línea.

Distribución

Es la conducción de energía eléctrica desde los puntos de entrega de la transmisión hasta los puntos de suministro a los usuarios. (CENACE, 2014).

DNP3, Distributed Network Protocol versión 3

Es el estándar (IEEE, IEEE SA - 1815-2012 - IEEE Standard for Electric Power Systems Communications-Distributed Network Protocol (DNP3), 2012) el cual define el protocolo de comunicación abierto orientado a la interoperabilidad entre estaciones SCADA de campo y maestras para la industria eléctrica, agua, gas y transporte.

AI, Entradas analógicas

Es aquella medición que representa el valor de una variable eléctrica.

DI, Entrada digital

Es aquella señal que puede ser representada como un 0 o un 1 lógico. Dentro de los sistemas SCADA se utiliza para señales de alarma, indicaciones, posición, estado, etc.

Equipo de Protección

La función del equipo de protección es aislar la falla de la instalación del sistema iniciando señales de disparo para los interruptores apropiados. Por lo mismo debe diferenciar entre una falla en la instalación y una aportación de corriente para una falla externa. El proceso completo debe ser efectuado en un mínimo tiempo y con el menor disturbio. (Laughton & Warne, 2003). Los valores de las variables eléctricas utilizadas son en base a las mediciones que obtiene de los TC y TP. Se función está determinada por la (ANSI & IEEE, 2014). Existen tres tipos de equipos de protección en base a su tecnología:

- *Electromecánico*, utiliza arreglos de circuitos analógicos.
- *Estático*, utiliza algoritmos en circuitos digitales. Su configuración se reduce a utilización se switches.
- *Digital*, utiliza algoritmos en circuitos digitales. Su configuración es ajustable por software.

Erlang

Lenguaje de programación usado para construir sistemas de tiempo real masivamente escalables con requerimientos de alta disponibilidad. Algunos de sus usos son en telecomunicaciones, banca, comercio electrónico, telefonía y mensajería instantánea. El sistema de tiempo de

ejecución de Erlang tiene soporte interno para concurrencia, distribución y tolerancia a fallos. (Ericsson Computer Science Laboratory, 2014).

Hz, Frecuencia

Es una magnitud que mide en una señal el número de oscilaciones sinusoidales por segundo de forma periódica. Se mide en Hertz (Hz).

Generación

Es la producción de energía eléctrica a partir de fuentes primarias de energía, utilizando los sistemas y equipos correspondientes. (CENACE, 2014).

Gerencia Regional de Transmisión y Transformación

Es el área de CFE que tiene a su cargo el mantenimiento de un conjunto de subestaciones y líneas de transmisión de un área geográfica determinada. (CENACE, 2014).

GB, gigabytes

Unidad de medida de almacenamiento informático en múltiplos de mil millones de bytes.

GPS, Global Positioning System

Equipo utilizado para la sincronización de tiempo en los equipos de la subestación eléctrica, logrando así, que todo equipo que digitaliza las alarmas e indicaciones del equipo primario y equipo auxiliar pueda estampar los cambios con una precisión de ± 1 milisegundo.

HTML, HyperText Markup Language

En un lenguaje basado en SGML usado como lenguaje para la publicación de la World Wide Web, y estandarizado por la especificación (W3C, HTML 4.01 Specification, 1999), el cual define la estructura de un documento Web.

HTTP, Hypertext Transfer Protocol

Es un protocolo a nivel de aplicación para sistemas de información distribuida, colaborativa y de medios, definido por el estándar RFC-2616 (RFC N. W., 1999).

Instalación

Es la infraestructura creada por el Sector Eléctrico, para la generación, transmisión, transformación y distribución de la energía eléctrica, así como la de los Generadores o Permissionarios que se interconectan con el sistema de la CFE. (CENACE, 2014).

Interruptor

Es el equipo para cerrar y abrir circuitos eléctricos con carga. (CENACE, 2014). Para no garantizar que no desestabilice el sistema eléctrico, debe interrumpir el flujo de energía en máximo 50 ms (apertura), o cerrarse en máximo 160 ms (CFE, NRF-022-CFE Interruptores de potencia de 72.5 kV a 420 kV, 2012).

Javascript

Lenguaje dinámico de programación inventado por Brendan Eich para ser usado en el navegador Web Netscape en 1995, y fue adoptado por la asociación de estándares ECMA en 1997 bajo la especificación

(ECMA, ECMA-262 ECMAScript Language Specification, 2011) y el estándar (ISO & IEC, ISO/IEC 16262:2011 ECMAScript scripting language, 2011).

JSON, JavaScript Object Notation

Es un formato estándar abierto usado para transmitir en texto plano objetos de datos que consisten en pares *atributo valor*. Está definido por dos estándares, *RFC-7159* (RFC, 2014) y (ECMA, ECMA-404 The JSON Data Interchange Format, 2013).

kV, kilovolts

Unidad de medida de voltaje en múltiplos de miles de volts.

Línea de transmisión

Es el elemento de transporte de energía entre dos instalaciones del Sistema Eléctrico. (CENACE, 2014).

Medidor Multifunción

Utiliza las mediciones de TC y TP para calcular diversas variables eléctricas instantáneas y acumuladas por un periodo de tiempo.

ms, milisegundo

Unidad de medida de tiempo en múltiplos de milésimas de segundo.

MCAD, Módulo de Control y Adquisición de Datos

Equipos que adquieren y digitalizan las mediciones, alarmas e indicaciones del equipo primario y equipo auxiliar, para su explotación en algún protocolo de comunicación industrial por los Servidores SCADA.

Así mismo, cuentan con interfaces de salidas digitales que permiten controlar el equipo primario.

PHP, PHP Hypertext Preprocessor

Lenguaje de programación *open source* tipo script que está específicamente adecuado para desarrollo Web. También puede ser usado como script ejecutado desde línea de comandos y para desarrollo de aplicaciones de escritorio. (The PHP Group, 2014).

POSIX

Es una familia de estándares definidos por la especificación (IEEE, IEEE 1003.1-2008 Standard for Information Technology - Portable Operating System Interface (POSIX(R)), 2008) e (ISO, IEC, & IEEE, ISO/IEC/IEEE 9945:2009 Information technology Portable Operating System Interface (POSIX) Base Specifications, Issue 7, 2009) para mantener compatibilidad entre sistemas operativos. Define una interface de programación de aplicaciones, así como shells de línea de comandos e interfaces, para la compatibilidad con variantes UNIX y otros sistemas operativos.

MVA, Potencia aparente

Es la suma vectorial de la potencia que disipa un circuito eléctrico y se transforma en calor o trabajo (W) y la potencia utilizada para la formación de campos eléctricos y magnéticos de sus componentes, que fluctúa entre estos componentes y la fuente de energía (VAR). Se mide en voltamperes (VA). MVA se refiere a millones de VA .

MW, Potencia efectiva, activa o real

Es la potencia capaz de transformar la energía eléctrica en trabajo. Los diferentes dispositivos eléctricos existentes convierten la energía eléctrica en otras formas de energía tales como: mecánica, lumínica, térmica, química, etc. Esta potencia es, por lo tanto, la realmente consumida por los circuitos y, en consecuencia, cuando se habla de demanda eléctrica, es esta potencia la que se utiliza para determinar dicha demanda. Se mide en watt (W). MW se refiere a millones de W.

MVAR, Potencia reactiva

Esta potencia no se consume ni se genera en el sentido estricto y en circuitos lineales solo aparece cuando existen bobinas o condensadores. Se mide en voltamperes reactivos (VAR). MVAR se refiere a millones de VAR.

Red Eléctrica de Potencia de 230 kV

Es la red troncal que usa un voltaje de 230 kilovolts en su conformación de subestaciones eléctricas interconectadas por líneas de transmisión.

Red Troncal

Es el conjunto de centrales generadoras, líneas de transmisión y estaciones eléctricas que debido a su función y/o ubicación, se consideran de importancia vital para el Sistema Eléctrico Nacional. (CENACE, 2014).

Red Troncal Eléctrica de 400 kV

Es la red troncal que usa un voltaje de 400 kilovolts en su conformación de subestaciones eléctricas interconectadas por líneas de transmisión.

RegEx, Regular Expression

Es una secuencia de caracteres que forma un patrón de búsqueda, usado principalmente para operaciones de búsqueda y reemplazo de patrones de texto. Están especificadas por el estándar *POSIX* bajo los conjuntos *BRE (Basic Regular Expression)* y *ERE (Extended Regular Expression)* (ISO, IEC, & IEEE, ISO/IEC/IEEE 9945:2009 Information technology Portable Operating System Interface (POSIX) Base Specifications, Issue 7, 2009), así como definidas por *Perl 5 perlre* (Perl, 2014).

Ruteador

Dispositivo que proporciona conectividad a nivel capa 3 del modelo OSI. Su función consiste en el encaminamiento de paquetes de datos de una red IP a otra red IP.

SCADA, Supervisory Control And Data Acquisition

Es un sistema de control de procesos industriales e infraestructura, que permite el telecontrol y supervisión remota de cada función mediante algún protocolo y canales de comunicación.

SCADAWeb

Sistema de la CFE, desarrollado por la Gerencia Regional de Transmisión Peninsular, que se accede por medio de un navegador Web y que muestra diagramas, tabulares y gráficas de los datos de estado operativo en tiempo real de un conjunto de subestaciones.

SOE, Secuencia de Eventos

Conocida también como eventos, históricos o simplemente SOE, es una base de datos de todos los cambios de valor de las señales digitales. Su

naturaleza es sólo contener cambios, y cada cambio tiene una estampa de tiempo del suceso con una resolución de milisegundos.

Servidores SCADA

Equipos dedicados a la explotación de datos de los MCAD para concentrar la información de la subestación eléctrica y ésta a su vez esté disponible para los diferentes despachos de energía mediante el uso de algún protocolo de comunicación industrial.

SEP, Sistema Eléctrico de Potencia

Es el conjunto de centrales generadoras, estaciones y líneas de transmisión, conectadas eléctricamente entre sí. (CENACE, 2014).

SEN, Sistema Eléctrico Nacional

Es el conjunto de instalaciones destinadas a la generación, transmisión, distribución y venta de energía eléctrica de servicio público en toda la República, estén o no interconectadas. (CENACE, 2014).

SIN, Sistema Interconectado Nacional

Es la porción del Sistema Eléctrico Nacional que permanece unida eléctricamente. (CENACE, 2014).

Subestación

Es la estación que recibe, transforma y/o distribuye energía eléctrica. (CENACE, 2014).

Supervision Tree

Concepto básico en Erlang el cual define un modelo de estructuración de procesos basado en la idea de procesos *worker* que realizan el trabajo y procesos *supervisor* los cuales monitorean el comportamiento de los worker. Erlang (Ericsson Computer Science Laboratory, 2014).

Switch

Equipo de comunicación que permite comunicar dispositivos en una instalación para obtener información sobre variables eléctricas y alarmas que disponen los demás equipos auxiliares.

Transformación

Es la modificación de las características de la tensión y de la corriente eléctrica para adecuarlas a las necesidades de transmisión y distribución de la energía eléctrica. (CENACE, 2014).

TC, Transformador de Corriente

Es un transductor que dará una señal de corriente directamente proporcional en magnitud y fase a la corriente fluyendo en el circuito primario. (Laughton & Warne, 2003).

TP, Transformador de Potencial

Es un transductor de voltaje que da una representación precisa en magnitud y fase del voltaje del conductor primario. (Laughton & Warne, 2003).

Transmisión

Es la conducción de energía eléctrica desde las plantas de generación o puntos de interconexión hasta los puntos de entrega para su distribución. (CENACE, 2014).

UTM, Unidad Terminal Maestra

Es el conjunto de equipos y programas, que procesan información procedente de las unidades terminales remotas, unidades maestras y otros medios, que utilice el Operador para el desempeño de sus funciones y que se encuentran ubicados en los centros de operación de los niveles jerárquicos. (CENACE, 2014).

UTR, Unidad Terminal Remota

Es el conjunto de dispositivos electrónicos que reciben, transmiten y ejecutan los comandos solicitados por las unidades maestras y que se encuentran ubicadas en las instalaciones del SEN. (CENACE, 2014).

Así mismo, adquiere y digitaliza mediante módulos de expansión las mediciones, alarmas e indicaciones del equipo primario y equipo auxiliar, para su explotación en algún protocolo de comunicación industrial a algún despacho de energía. Así mismo, cuentan con módulos de interface de salidas digitales que permiten accionar mecanismos para controlar el equipo primario.

WebSICLE

Sistema de la CFE, desarrollado por la Gerencia Regional de Transmisión Oriente, que se accede por medio de un navegador Web y

que muestra diagramas y tabulares de los datos de estado operativo en tiempo real de un conjunto de subestaciones.

XML, Extensible Markup Language

Formato basado en texto usado para la representación estructurada de información, definido por los estándares (W3C, Extensible Markup Language (XML) 1.0 (Fifth Edition), 2008) y (W3C, Extensible Markup Language (XML) 1.1 (Second Edition), 2006).

ANEXO B

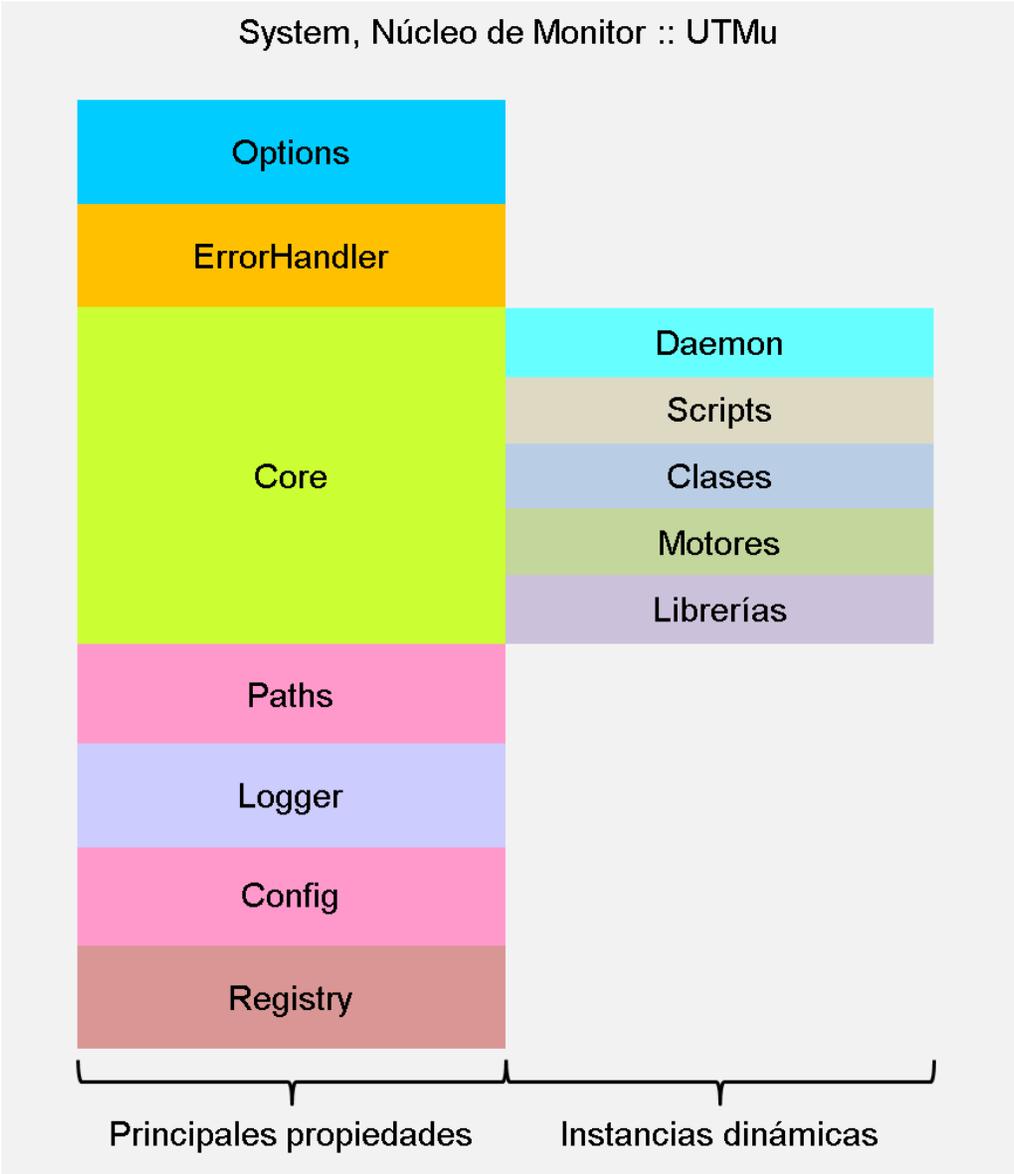
SUBSISTEMA

MONITOR :: UTMU

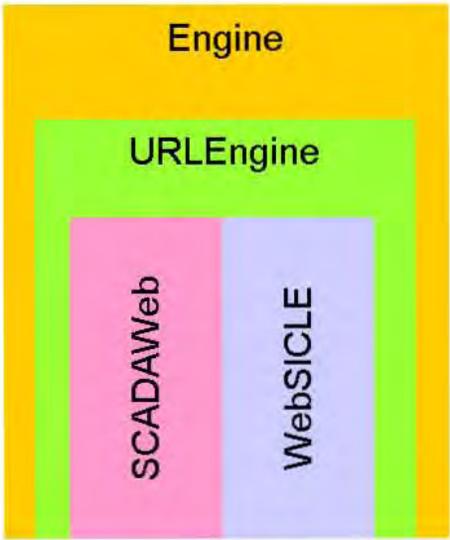
CLASES IMPLEMENTADAS

ANEXO B. SUBSISTEMA MONITOR :: UTMU » CLASES IMPLEMENTADAS

Anexo B.1. Núcleo del Subsistema Monitor :: UTMu



Anexo B.2. Motores del Subsistema de Adquisición y Reconocimiento de Datos



ANEXO C

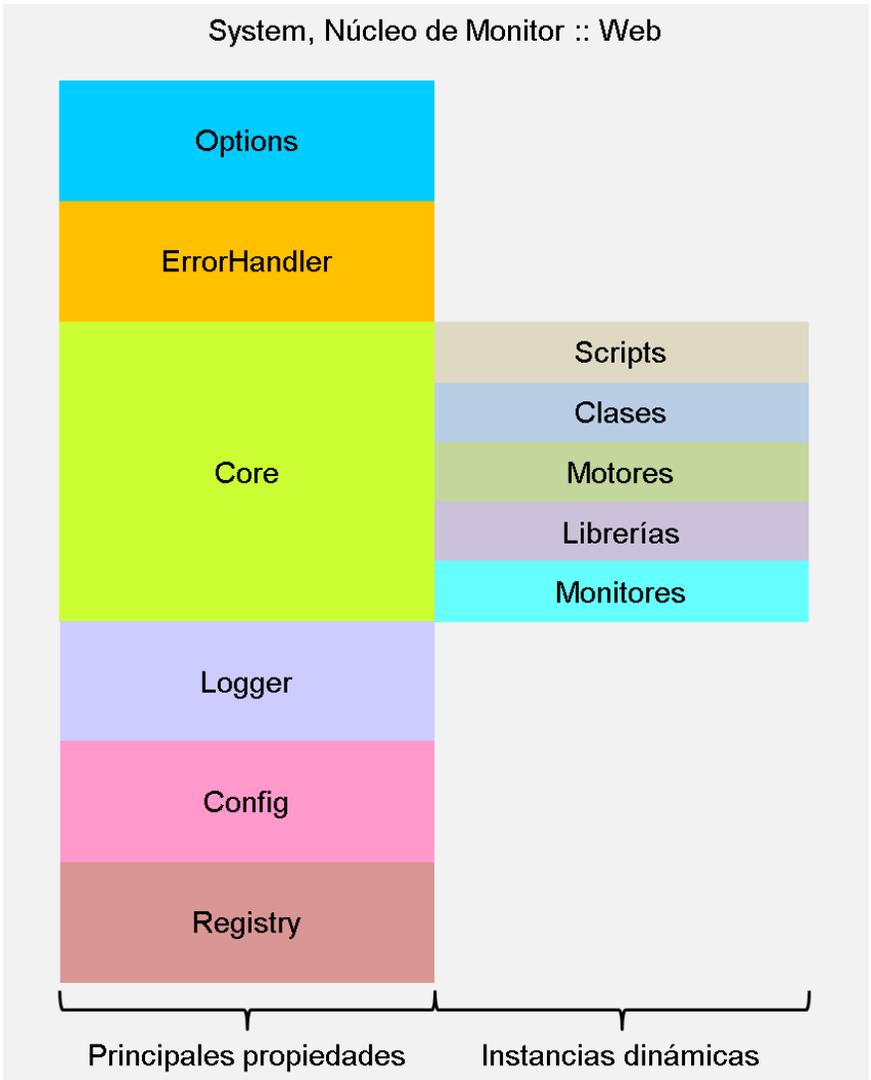
SUBSISTEMA

MONITOR :: WEB

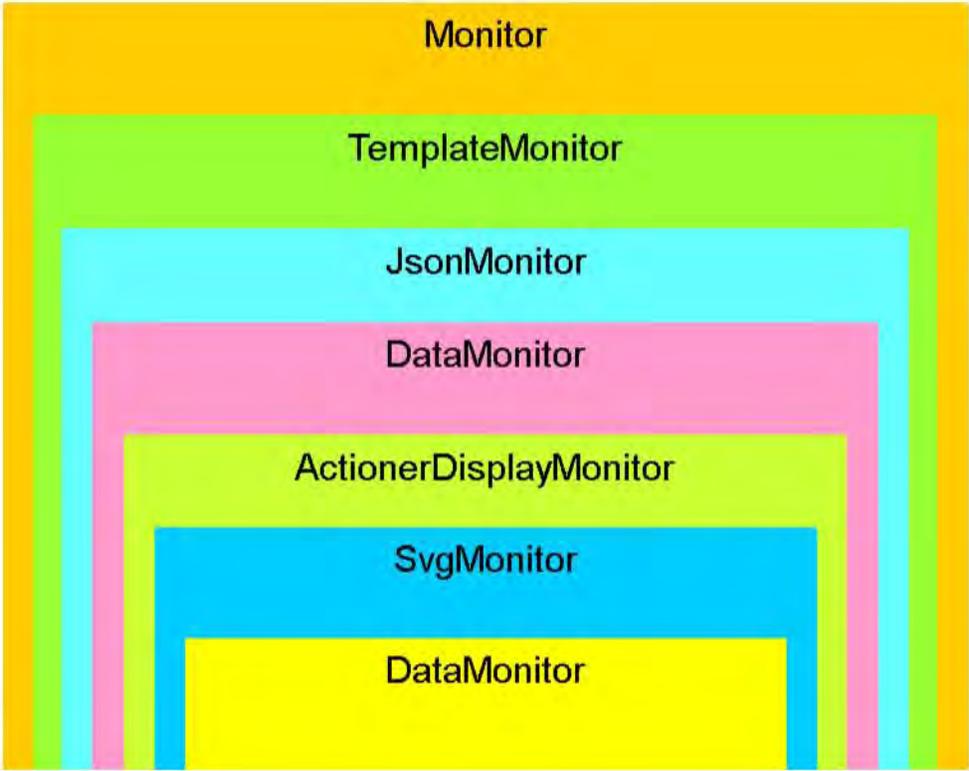
CLASES IMPLEMENTADAS

ANEXO C. SUBSISTEMA MONITOR :: WEB » CLASES IMPLEMENTADAS

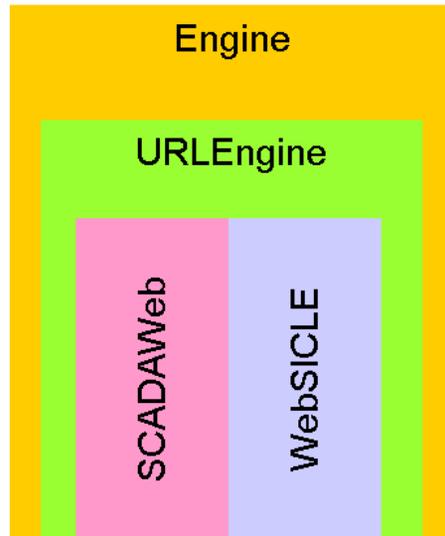
Anexo C.1. Núcleo de la Aplicación Servidor del Subsistema Monitor :: Web



Anexo C.2. Jerarquía de Monitores de la Aplicación Servidor



Anexo C.3. Motores Implementados por la Aplicación Servidor



Anexo C.4. Plantilla HTML Usada por los Monitores

```
<html>
<head>
  <title>.: Monitor :: Web :: {{{TITLE}}}</title>
  <link rel="shortcut icon" href="images/favicon.ico" />
  <!-- CSS usado para la presentación de la interfaz -->
  <style type="text/css">
    @import "js/dojo/resources/dojo.css";
    @import "js/dijit/themes/tundra/tundra.css";
    @import "res/css/JsonMonitor_gui.css";
  </style>
  <!-- Carga el núcleo de la Librería The Dojo Toolkit -->
  <script type="text/javascript" src="js/dojo/dojo.js" djConfig="parseOnLoad:
true, isDebug: {{{DEBUG}}}, gfxRenderer: 'svg,silverlight,vml',
monitorWebConfig: {designMode: {{{DESIGN_MODE}}}"></script>
  <!-- Carga el núcleo de la aplicación cliente -->
```

```

<script type="text/javascript" src="js/monitorWeb.js"></script>
<!-- Script de inicialización de La aplicación cliente -->
<script type="text/javascript" src="res/js/JsonMonitor_gui.js"></script>
<script language="javascript" type="text/javascript">
    // Utiliza una interfaz gráfica implementada para diagramas en formato
SVG
    dojo.require("monitorWeb.userinterface.SvgUserInterface");
    // Inicializa La interfaz gráfica
    var UI = new monitorWeb.userinterface.SvgUserInterface("{{{NAME}}}", {
        diagramId:    "SVGDiaqram",
        interval:    {{{FETCHER_INTERVAL}}},
        timeout:    {{{FETCHER_TIMEOUT}}}
    });
</script>
</head>
<body class="tundra">
    <!-- Begin Layout -->
    <div id="Layout" dojoType="dijit.layout.BorderContainer"
liveSplitters="false" gutters="false">
        <!-- Begin Header -->
        <div id="Header" dojoType="dijit.layout.ContentPane" region="top">
            <div class="headTitle">.: Monitor :: Web</div><div id="TitleMessage"
class="headSubtitle">&nbsp;.: {{{LONG_TITLE}}}</div>
            <div class="headMessage">CFE - &iecl;Una Empresa de Clase Mundial! -
<a class="exit" href="#" alt="">Salir</a></div>
        </div>
        <!-- End Header -->
        <!-- Begin Content -->
        <div id="SVGDiaqram" dojoType="monitorWeb.svg.ContentPane"
fitImage="true" region="center">
            <!--
                Aquí se realizará La carga del diagrama unifilar usando La tecnología
de gráficos vectoriales del navegador
            -->

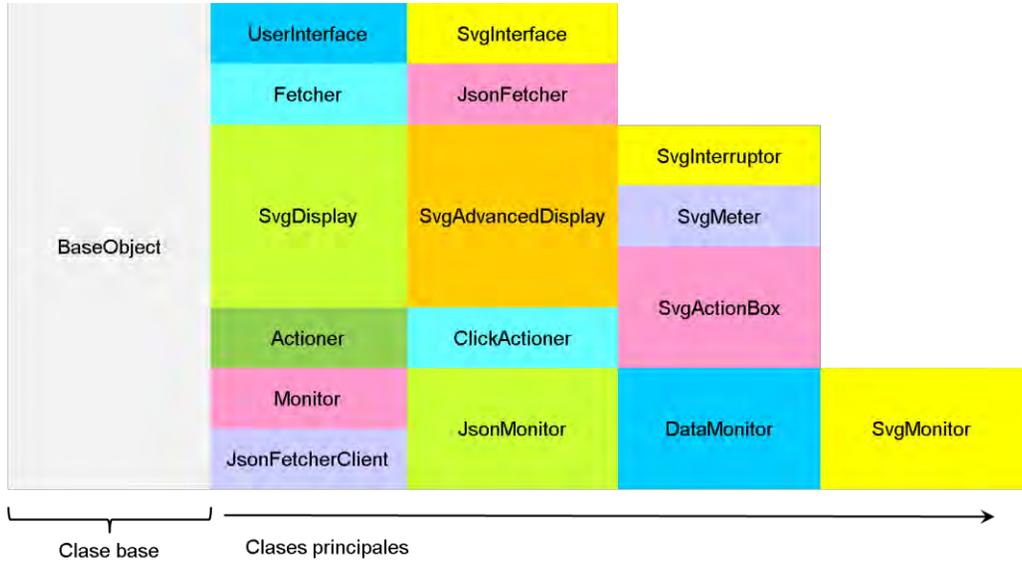
```

```

</div>
<!-- End Content -->
<!-- Begin Footer -->
<div id="Footer" dojoType="dijit.layout.ContentPane" region="bottom">
  <div>&nbsp;</div>
  <div id="ProgressBarContainer"><div id="ProgressBar"
dojoType="dijit.ProgressBar" indeterminate="true"></div></div>
  <div class="message"></div>
</div>
<!-- End Footer -->
</div>
<!-- End Layout -->
</body>
</html>

```

Anexo C.5. Jerarquía de Clases Lado Cliente



ANEXO D

REGISTROS DE ACTIVIDAD SIMPLIFICADOS DEL SUBSISTEMA MONITOR :: UTMU

ANEXO D. REGISTROS DE ACTIVIDAD SIMPLIFICADOS DEL SUBSISTEMA MONITOR :: UTMU

```
14:11:09 UTMU [MESSAGE]: BOOTING SYSTEM
14:11:09 UTMU [MESSAGE]: SYSTEM INITIALIZED
14:11:09 UTMU [MESSAGE]: System Database Connection established
14:11:09 UTMU [MESSAGE]: INIT: Historier - Data table Historier exist
14:11:10 UTMU [MESSAGE]: Subprocess: Creating subprocess GRTC_BAC
14:11:10 GRTC_BAC [MESSAGE]: Subprocess: Subprocess started.
14:11:10 UTMU [MESSAGE]: Subprocess: Subprocess GRTC_BAC created with
PID 8758.
14:11:10 GRTC_BAC [MESSAGE]: ARA SUBPROCESS GRTC_BAC INITIALIZED
14:11:10 GRTC_BAC [MESSAGE]: System Database Connection established
14:11:10 GRTC_BAC [MESSAGE]: INIT: Historier - Data table Historier
exist
14:11:10 GRTC_BAC [MESSAGE]: Collection: Engine has 2 items
14:11:10 GRTC_BAC [MESSAGE]: Collection: Repository has 32 items
14:11:10 GRTC_BAC [DEBUG  ]: <<<STARTING REQUEST DATA>>>
14:11:10 GRTC_BAC [DEBUG  ]: SCADAWeb Engine:
host=scadaWeb.bac.grtc.net unifilar=SPA dataType=digital
14:11:10 GRTC_BAC [DEBUG  ]: SCADAWeb Engine:
host=scadaWeb.bac.grtc.net unifilar=STM dataType=analogic
14:11:11 GRTC_BAC [DEBUG  ]: SCADAWeb Engine:
host=scadaWeb.bac.grtc.net unifilar=MAN dataType=analogic
14:11:11 GRTC_BAC [DEBUG  ]: SCADAWeb Engine:
host=scadaWeb.bac.grtc.net unifilar=CYA dataType=digital
14:11:11 GRTC_BAC [DEBUG  ]: SCADAWeb Engine:
host=scadaWeb.bac.grtc.net unifilar=SPA dataType=analogic
14:11:11 GRTC_BAC [DEBUG  ]: SCADAWeb Engine:
```

```
host=scadaWeb.bac.grtc.net unifilar=SAU dataType=digital
14:11:11 GRTC_GRO [MESSAGE]: Subprocess: Subprocess started.
14:11:11 UTMU [MESSAGE]: Subprocess: Subprocess GRTC_GRO created with
PID 8764.
14:11:11 GRTC_GRO [MESSAGE]: ARA SUBPROCESS GRTC_GRO INITIALIZED
14:11:11 GRTC_GRO [MESSAGE]: System Database Connection established
14:11:11 GRTC_GRO [MESSAGE]: INIT: Historier - Data table Historier
exist
14:11:11 GRTC_GRO [MESSAGE]: Collection: Engine has 2 items
14:11:11 GRTC_GRO [MESSAGE]: Collection: Repository has 10 items
14:11:11 GRTC_GRO [DEBUG ]: <<<STARTING REQUEST DATA>>>
14:11:11 GRTC_GRO [DEBUG ]: SCADAWeb Engine:
host=scadaWeb.gro.grtc.net unifilar=ITP dataType=analogic
14:11:11 GRTC_GRO [DEBUG ]: SCADAWeb Engine:
host=scadaWeb.gro.grtc.net unifilar=LAT dataType=digital
14:11:11 GRTC_GRO [DEBUG ]: SCADAWeb Engine:
host=scadaWeb.gro.grtc.net unifilar=CGP dataType=digital
14:11:11 GRTC_GRO [DEBUG ]: SCADAWeb Engine:
host=scadaWeb.gro.grtc.net unifilar=PIC dataType=analogic
14:11:11 GRTC_BAC [DEBUG ]: SCADAWeb Engine:
host=scadaWeb.bac.grtc.net unifilar=SLM dataType=analogic
14:11:11 GRTC_GRO [DEBUG ]: SCADAWeb Engine:
host=scadaWeb.gro.grtc.net unifilar=ITP dataType=digital
14:11:11 GRTC_BAC [DEBUG ]: SCADAWeb Engine:
host=scadaWeb.bac.grtc.net unifilar=SMD dataType=analogic
14:11:11 GRTC_GRO [DEBUG ]: SCADAWeb Engine:
host=scadaWeb.gro.grtc.net unifilar=QMD dataType=digital
14:11:11 GRTC_BAC [DEBUG ]: SCADAWeb Engine:
host=scadaWeb.bac.grtc.net unifilar=MAN dataType=digital
14:11:11 GRTC_BAC [DEBUG ]: SCADAWeb Engine:
host=scadaWeb.bac.grtc.net unifilar=QRP dataType=analogic
14:11:12 GRTC_GRO [DEBUG ]: SCADAWeb Engine:
```

```
host=scadaWeb.gro.grtc.net unifilar=PIC dataType=digital
14:11:12 GRTC_BAC [DEBUG ]: SCADAWeb Engine:
host=scadaWeb.bac.grtc.net unifilar=QPM dataType=analogic
14:11:12 GRTC_BAC [DEBUG ]: SCADAWeb Engine:
host=scadaWeb.bac.grtc.net unifilar=LDE dataType=analogic
14:11:12 GRTC_GRO [DEBUG ]: SCADAWeb Engine:
host=scadaWeb.gro.grtc.net unifilar=QMD dataType=analogic
14:11:12 GRTC_BAC [DEBUG ]: SCADAWeb Engine:
host=scadaWeb.bac.grtc.net unifilar=CYA dataType=analogic
14:11:12 GRTC_BAC [DEBUG ]: SCADAWeb Engine:
host=scadaWeb.bac.grtc.net unifilar=QPM dataType=digital
14:11:12 GRTC_GRO [DEBUG ]: SCADAWeb Engine:
host=scadaWeb.gro.grtc.net unifilar=LAT dataType=analogic
14:11:12 GRTC_GRO [DEBUG ]: SCADAWeb Engine:
host=scadaWeb.gro.grtc.net unifilar=CGP dataType=analogic
14:11:12 GRTC_BAC [DEBUG ]: SCADAWeb Engine:
host=scadaWeb.bac.grtc.net unifilar=QRO dataType=digital
14:11:12 GRTC_GRO [DEBUG ]: <<<SLEEPING FOR 19 SECONDS>>>
14:11:12 GRTC_BAC [DEBUG ]: SCADAWeb Engine:
host=scadaWeb.bac.grtc.net unifilar=DAN dataType=digital
14:11:12 GRTC_BAC [DEBUG ]: SCADAWeb Engine:
host=scadaWeb.bac.grtc.net unifilar=QRO dataType=analogic
14:11:12 GRTC_BAC [DEBUG ]: SCADAWeb Engine:
host=scadaWeb.bac.grtc.net unifilar=STF dataType=digital
14:11:13 GRTC_BAC [DEBUG ]: SCADAWeb Engine:
host=scadaWeb.bac.grtc.net unifilar=SMD dataType=digital
14:11:13 GRTC_BAC [DEBUG ]: SCADAWeb Engine:
host=scadaWeb.bac.grtc.net unifilar=SAU dataType=analogic
14:11:13 GRTC_BAC [DEBUG ]: SCADAWeb Engine:
host=scadaWeb.bac.grtc.net unifilar=DAN dataType=analogic
14:11:13 GRTC_BAC [DEBUG ]: SCADAWeb Engine:
host=scadaWeb.bac.grtc.net unifilar=SLM dataType=digital
```

```
14:11:13 GRTC_BAC [DEBUG ]: SCADAWeb Engine:  
host=scadaWeb.bac.grtc.net unifilar=LDE dataType=digital  
14:11:13 GRTC_BAC [DEBUG ]: SCADAWeb Engine:  
host=scadaWeb.bac.grtc.net unifilar=QRP dataType=digital  
14:11:14 GRTC_BAC [DEBUG ]: SCADAWeb Engine:  
host=scadaWeb.bac.grtc.net unifilar=STF dataType=analogic  
14:11:14 GRTC_BAC [DEBUG ]: <<<SLEEPING FOR 16 SECONDS>>>
```