



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO  
FACULTAD DE ESTUDIOS SUPERIORES ARAGÓN  
INGENIERÍA EN COMPUTACIÓN

Título del informe:

“Experiencia profesional en tecnologías  
de la información en la iniciativa  
privada”



Informe de Ejercicio Profesional para obtener el  
título de  
Ingeniero en Computación

Presenta:

Miguel Angel De León Trejo

Asesor del Informe:

M en I. Arcelia Bernal Díaz

México D.F., Abril 2013



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 todas las personas que de uno u otro modo han colaborado en el logro de mis objetivos profesionales.

Jefe de Carrera de Ingeniería en Computación:

M. en C. Felipe de Jesús Gutiérrez López

A mis asesores de este informe profesional:

M. en C. Ricardo Arturo Gutiérrez Orozco

M. en C. Graciela Lourdes Prado Becerra

Ing. Oscar Estrada García

M. en C. Jesús Hernández Cabrera

Especialmente M. en I. Arcelia Bernal Díaz, por su invaluable apoyo.

A la Universidad Nacional Autónoma de México.

## DEDICATORIA

A mis amigos.

A Erika, quién ha estado ahí, conmigo; a pesar de todo.

A mis hermanos, de quienes espero ser un ejemplo a seguir y juntos lograr las metas que nos propongamos.

A mi padre, quien estando cerca o lejos siempre ha sido un soporte y ejemplo para mí y me ha brindado su apoyo incondicional.

Y principalmente a mi madre, qué ha dedicado su vida entera a sus hijos, a cuidarlos y protegerlos, quien nos ha brindado más que amor, educación, comprensión y paciencia; una razón de ser y existir.

<b>INTRODUCCIÓN.....</b>	<b>5</b>
<b>CAPÍTULO -- 1 COMPONENTES DE HARDWARE, CATALOGACIÓN, FUNCIONAMIENTO Y ENSAMBLE .....</b>	<b>7</b>
1.1 MEMORIAS RAM.....	7
1.2 PROCESADORES .....	11
1.3 DISCOS DUROS .....	16
1.4.2 Tipos de conexión .....	19
1.4.3 Estructura lógica.....	20
<b>CAPÍTULO 2 -- ANÁLISIS, DISEÑO Y EJECUCIÓN DE LA ESTRATEGIA DE PRUEBAS PARA SISTEMAS DE SOFTWARE .....</b>	<b>22</b>
2.1 ANÁLISIS.....	23
2.1.1 Levantamiento de requerimientos.....	23
2.1.2 Casos de uso.....	25
2.2 PRUEBAS .....	25
2.2.1 Elaboración de casos y matrices de prueba.....	26
2.2.1.1 Estructura de los casos de prueba.....	26
2.2.1.2 Registro de defectos.....	28
2.3 Tipos de pruebas .....	28
2.3.1 Pruebas unitarias.....	28
2.3.2 Pruebas de Integración.....	30
2.3.3 Caja blanca.....	30
2.3.4 Caja negra.....	30
2.3.5 Pruebas de regresión.....	31
2.3.6 Pruebas funcionales.....	31
2.3.7 Metodologías de pruebas.....	32
<b>CAPÍTULO 3 IMPLEMENTACIÓN (SOFTWARE E INFRAESTRUCTURA) DEL SISTEMA DE VERIFICACIÓN Y MONITOREO DEL IFE.....</b>	<b>34</b>
3.1 CONFIGURACIONES GENERALES .....	35
3.1.1 Nombre del equipo.....	35
3.1.2 Interfaces y Direcciones IP .....	36
3.1.3 Zona Horaria y Servidor NTP.....	37
3.1.4 Archivo Hosts.....	38
3.2 SERVIDOR 1 .....	38
3.2.1 Recursos compartidos.....	39
3.2.2 Configuración de encoders .....	39
3.2.3 Configuración de Front End.....	42
3.2.4 Sección Mantenimiento.....	42
3.3 SERVIDOR 2.....	43
3.3.1 Partición de disco.....	44
3.3.2 Servicios.....	44
3.3.3 Archivos .DAT, .FMT y .IDX.....	45
3.4 SERVIDOR 3.....	45
3.4.1 Archivos de configuración .....	45
3.4.2 Servicios del Servidor 3.....	45
3.5 SERVIDOR 4 .....	46
<b>CONCLUSIONES.....</b>	<b>47</b>
<b>GLOSARIO Y ACRÓNIMOS.....</b>	<b>49</b>
<b>BIBLIOGRAFÍA Y REFERENCIAS .....</b>	<b>51</b>

## Introducción

El presente informe de ejercicio profesional tiene como objetivo dar a conocer las actividades desarrolladas en las diferentes áreas de trabajo en que he participado; para obtener el título de Ingeniero en Computación, mi experiencia ha sido adquirida y fortalecida en cada una de las instituciones en las que laboré y se enlistan en la siguiente tabla, así como los diferentes roles que he desempeñado, iniciando en el área de soporte técnico, y conociendo las diferentes áreas de conocimiento de acuerdo al ciclo de vida de los proyectos, tales como son Análisis, Diseño, Construcción, Pruebas y Capacitación.

En cada capítulo se detallarán los conocimientos adquiridos y como fueron aplicados en los proyectos en los que he participado.

<b>Rol desempeñado</b>	<b>Fecha de duración</b>	<b>Institución</b>
Líder de proyecto	Actualmente	Intellego-Banorte
Asesor de carrera - Arquitecto QA	Actualmente	Intellego- Varios Proyectos
Líder técnico de testing	03/2010 - 09/2010	Certum
Líder técnico	08/2009 - 02/2010	Hildebrando-Multipack
Analista de sistemas	04/2008 - 08/2009	IFE-Dirección de Prerrogativas y Partidos Políticos
Informático Administrativo	01/2008 - 04/2008	IFE-Subdirección de Capacitación
Instructor	08/2007 - 12/2007	IFE-Subdirección de Capacitación
Supervisor de mantenimiento	01/2006 - 06/2007	Microhard Systems
Prácticas Profesionales	03/2005 - 07/2005	Consejo de la Judicatura Federal
Servicio Social	01/2005 - 04/2005	Consejo de la Judicatura Federal
Técnico en mantenimiento de equipo de cómputo	01/2001 - 12/2004	Particular

El primer capítulo del presente informe menciona los componentes del hardware, su funcionamiento y ensamble así como las actividades que como técnico y supervisor de mantenimiento realicé en la empresa que laboraba.

En el capítulo número 2 se describe el análisis, el diseño y la estrategia de las pruebas para sistemas de software y la forma como diseñaba actividades para la capacitación de las mismas. Asimismo se profundiza en la etapa de pruebas de software, explicando tipos de pruebas, metodologías y técnicas de diseño y ejecución.

En el último capítulo se aborda la operación y configuración de la infraestructura y componentes que comprenden la solución del Sistema de Verificación y Monitoreo.

## Capítulo -- 1 Componentes de hardware, catalogación, funcionamiento y ensamble

Como técnico y supervisor de mantenimiento, una de mis principales actividades era la Identificación y catalogación de componentes de hardware para el ensamble de equipo, esto debido al volumen de computadoras que se recibían como equipo discontinuado y del cual se obtenían todas las piezas que se pudieran reutilizar para el ensamblado de nuevos equipos.

Era de gran importancia el catalogar correctamente los componentes, puesto que no todos son compatibles entre sí, los componentes físicamente son diferentes por lo que no causa dificultad su instalación, más no sus características internas (del mismo componente). Es decir, una mother board a la que se le instalan memorias rimm, éstas tendrán que ser de la misma capacidad y correr a la misma velocidad, de lo contrario, el equipo no funcionará.

A continuación se presentará una breve descripción de los componentes de hardware que eran catalogados y sus características.

### 1.1 Memorias RAM

SIMM (siglas de Single In-line Memory Module), es un tipo de memoria RAM que consisten en placas de circuito impreso sobre las que se montan los integrados de memoria DRAM. Estos módulos se insertan en zócalos sobre la placa base. Los contactos en ambas caras están interconectados. Cuentan con capacidades de 256 kB, 1 MB, 4 MB, 16 MB. En algunos sistemas debían usarse a pares ya que cada banco de memoria estaba integrado por dos módulos. La aparición del Intel 80486 trae también el paso al nuevo formato de 108 mm (4,25 pulgadas) y 72 pines. Esto era debido a que en un 386/486 era necesario instalar 4 SIMMs de 30 pines para completar una bancada de memoria. Dicha bancada podía sustituirse por un sólo DIMM de 72 pines (dos en los equipos Pentium).

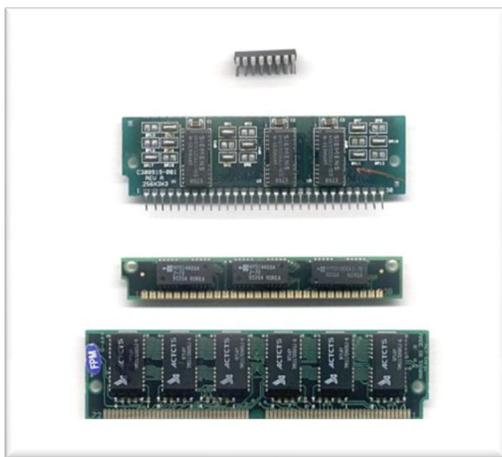


Imagen 1.1.1 Memorias DRAM, SIMM de 30 y 72 pines

DIMM son las siglas de «Dual In-line Memory Module» y se traduce como Módulo de Memoria en línea doble. Al igual que la memoria SIMM, contiene chips de memoria y se conecta directamente en ranuras de la placa base. Los módulos DIMM son reconocibles externamente por poseer sus contactos (o pines) separados en ambos lados, a diferencia de los SIMM que poseen los contactos de modo que los de un lado están unidos con los del otro.

Las memorias DIMM comenzaron a reemplazar a las SIMM como el tipo predominante de memoria cuando los microprocesadores Intel Pentium dominaron el mercado. Un DIMM puede comunicarse con el PC a 64 bits (y algunos a 72 bits) en vez de los 32 bits de los SIMMs. Funcionan a una frecuencia de 100 y 133 MHz cada una.

Los módulos en formato DIMM, al ser memorias de 64 bits, no necesitan emparejamiento, poseen chips de memoria en ambos lados de la placa de circuito impresa, y poseen a la vez, 84 conectores de cada lado, lo cual suma un total de 168 clavijas. Además de ser de mayores dimensiones que los módulos SIMM (130x25mm), estos módulos poseen una segunda muesca que evita confusiones.

Cabe mencionar que los conectores DIMM han sido mejorados para facilitar su inserción, gracias a las palancas ubicadas a ambos lados de cada conector.

También existen módulos más pequeños, conocidos como SO DIMM (DIMM de contorno pequeño), diseñados para ordenadores portátiles. Los módulos SO DIMM sólo cuentan con 144 clavijas en el caso de las memorias de 64 bits, y con 77 clavijas en el caso de las memorias de 32 bits.



Imagen 1.1.2 Memorias DIMM PC100 y PC133

RIMM de Rambus Inline Memory Module (Módulo de Memoria en Línea Rambus), designa a los módulos de memoria RAM que utilizan una tecnología denominada RDRAM, desarrollada por Rambus Inc. a mediados de los años 1990 con el fin de introducir un módulo de memoria con niveles de rendimiento muy superiores a los módulos de memoria SDRAM de 100 Mhz y 133 Mhz.

Los módulos RIMM RDRAM cuentan con 184 pines y debido a sus altas frecuencias de trabajo requieren de difusores de calor consistentes en una placa metálica que recubre los chips del módulo. Se basan en un bus de datos de 16 bits y están disponibles en velocidades de 300MHz (PC-600), 356 Mhz (PC-700), 400 Mhz (PC-800) y 533 Mhz (PC-1066) que por su bus de 16 bits tenía un rendimiento 4 veces menor que la DDR. La RIMM de 533MHz tiene un rendimiento similar al de un módulo DDR133.

Inicialmente los módulos RIMM fueron introducidos para su uso en servidores basados en Intel Pentium 4. Rambus no manufactura módulos RIMM si no que tiene un sistema de licencias para que estos sean manufacturados por terceros siendo Samsung el principal fabricante de éstos.

A pesar de tener la tecnología RDRAM niveles de rendimiento muy superiores a la tecnología SDRAM y las primeras generaciones de DDR RAM, debido al alto costo de esta tecnología no han tenido gran aceptación en el mercado de PC. Su momento clave tuvo lugar durante el periodo de introducción del Pentium 4 para el cual se diseñaron las primeras placas base, pero Intel ante la necesidad de lanzar equipos más económicos decidió lanzar placas base con soporte para SDRAM y más adelante para DDR RAM desplazando esta última tecnología a los módulos RIMM del mercado.

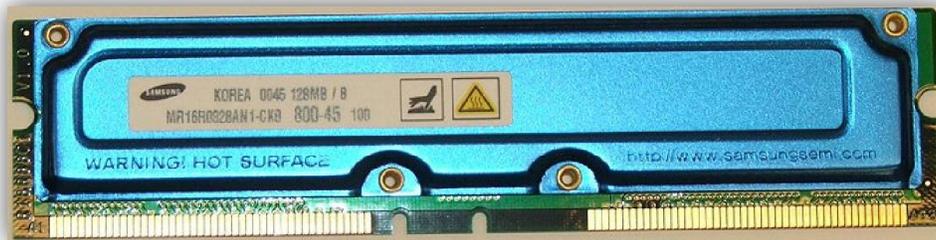


Imagen 1.1.3 Memoria RIMM

DDR (Double Data Rate) significa doble tasa de transferencia de datos. Son módulos de memoria RAM compuestos por memorias síncronas (SDRAM), disponibles en encapsulado DIMM, que permite la transferencia de datos por dos canales distintos simultáneamente en un mismo ciclo de reloj.

Fueron primero adoptadas en sistemas equipados con procesadores AMD Athlon. Intel con el Pentium 4 en un principio utilizó únicamente memorias RAMBUS, más costosas. Ante el avance en ventas y buen rendimiento de los sistemas AMD basados en DDR SDRAM, Intel se vio obligado a cambiar su estrategia y utilizar memoria DDR, lo que le permitió competir en precio. También se utiliza la nomenclatura PC1600 a PC4800

Muchas placas base permiten utilizar estas memorias en dos modos de trabajo distintos:

- Single Memory Channel: Todos los módulos de memoria intercambian información con el bus a través de un sólo canal, para ello sólo es necesario introducir todos los módulos DIMM en el mismo banco de slots.

- Dual Memory Channel: Se reparten los módulos de memoria entre los dos bancos de slots diferenciados en la placa base, y pueden intercambiar datos con el bus a través de dos canales simultáneos, uno para cada banco.

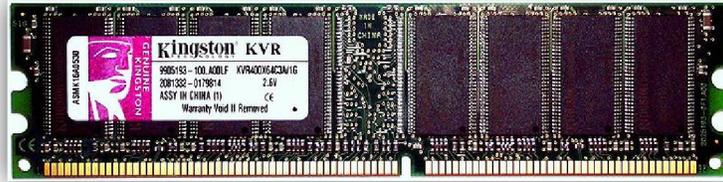


Imagen 1.2.4 Memoria DDR

Los módulos DDR2 son capaces de trabajar con 4 bits por ciclo, es decir 2 de ida y 2 de vuelta en un mismo ciclo mejorando sustancialmente el ancho de banda potencial bajo la misma frecuencia de una DDR SDRAM tradicional (si una DDR a 200 MHz reales entregaba 400 MHz nominales, la DDR2 por esos mismos 200 MHz reales entrega 800 MHz nominales). Este sistema funciona debido a que dentro de las memorias hay un pequeño buffer que es el que guarda la información para luego transmitirla fuera del modulo de memoria, este buffer en el caso de la DDR convencional trabajaba tomando los 2 bits para transmitirlos en 1 sólo ciclo, lo que aumenta la frecuencia final. En las DDR2, el buffer almacena 4 bits para luego enviarlos, lo que a su vez redobla la frecuencia nominal sin necesidad de aumentar la frecuencia real de los módulos de memoria.



Imagen 1.1.5 Memoria DDR2

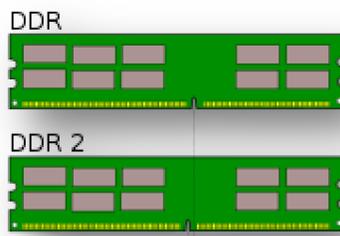


Imagen 1.1.6 Comparación gráfica de una memoria DDR y DDR2

## 1.2 Procesadores

El Intel 80286 es un microprocesador de 16 bits de la familia x86, que fue lanzado al mercado por Intel. Las versiones iniciales del 286 funcionaban a 6 y 8 MHz, pero acabó alcanzando una velocidad de hasta 25 MHz.

Tras las versiones iniciales a 6 y 8 MHz, Intel lanzó un modelo a 12,5 MHz. AMD y Harris ampliaron esa velocidad a 20 MHz y 25 MHz, respectivamente. En promedio, el 80286 tenía una velocidad de unas 0,21 instrucciones por ciclo de reloj. El modelo de 6 MHz operaba a 0,9 MIPS, el de 10 MHz a 1,5 MIPS, y el de 12 MHz a 2,66 MIPS.

El rendimiento del 80286 es más del doble que el de sus predecesores (los Intel 8086 e Intel 8088) por ciclo de reloj. De hecho, el aumento de rendimiento por ciclo de reloj puede ser el mayor entre las diferentes generaciones de procesadores x86. El cálculo de los modos de direccionamiento más complejos (como base + índice) utilizaban menos ciclos de reloj porque eran realizados por un circuito especial en el 286; el 8086, ha de realizar el cálculo de la dirección efectiva en la ULA general, lo que consume muchos ciclos. Además, operaciones matemáticas complejas (como MUL/DIV) ocupan menos ciclos que en el 8086.

Al tener un bus de direcciones de 24 bits, es capaz de direccionar hasta 16 MB (16 MiB) de memoria RAM, mientras que el 8086 sólo puede 1 MB. Aunque MS-DOS puede utilizar la RAM adicional ( memoria extendida ) mediante una llamada a la BIOS INT, como disco RAM o mediante emulación de memoria expandida previamente habilitada mediante software la memoria extendida, pocos ordenadores basados en el 286 tuvieron más de 1 MB de memoria.

El 286 fue diseñado para ejecutar aplicaciones multitarea, control de procesos en tiempo real y sistemas multiusuario. Para ello se le añadió un modo protegido, en el cual existían cuatro anillos de ejecución y división de memoria mediante tablas de segmentos. En este modo trabajaban las versiones de 16 bits del sistema operativo OS/2. En este modo protegido se permitía el uso de toda la memoria directamente, y que además ofrecía protección entre aplicaciones para evitar la escritura de datos accidental (o malintencionada) fuera de la zona de memoria asignada. Sin embargo, una vez que el procesador entraba en el modo protegido, no podía volver al modo real sin resetear el procesador. El microcontrolador Intel 8042, utilizado para controlar el teclado en los AT, tiene una función que resetea sólo la CPU (la famosa combinación CTRL - ALT - DEL).

El sucesor del 286 fue el Intel 80386, de 32 bits.

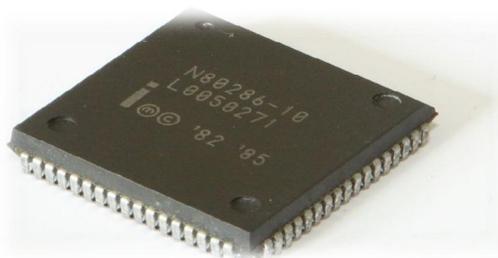


Imagen 1.2.7 Microprocesador 286

El Intel 80386 fue empleado como la unidad central de proceso de muchos ordenadores personales desde mediados de los años 80 hasta principios de los 90.

En mayo de 2006 Intel anunció que la fabricación del 386 finalizaría a finales de septiembre de 2007. Aunque ha quedado obsoleto como CPU de ordenador personal, Intel ha seguido fabricando el chip para sistemas embebidos y tecnología aeroespacial.

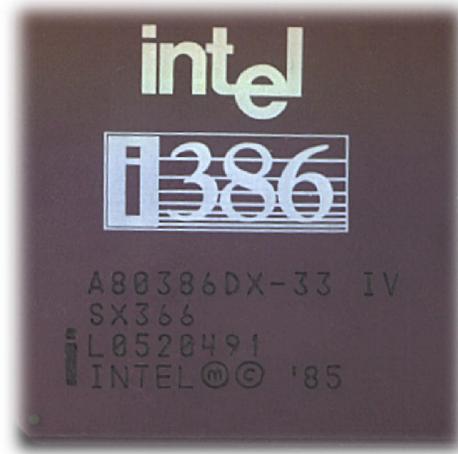


Imagen 1.2.8 Microprocesador 386

Los Intel 80486 son una familia de microprocesadores de 32 bits con arquitectura x86. Son muy similares a sus predecesores, los Intel 80386. Las diferencias principales son que los 486 tienen un conjunto de instrucciones optimizado, una unidad de coma flotante y un caché unificado integrados en el propio circuito integrado del microprocesador y una unidad de interfaz de bus mejorada. Estas mejoras hacen que los 486 sean el doble de rápidos que un 386. De todos modos, algunos 486 de gama baja son más lentos que los 386 más rápidos.

Las velocidades de reloj típicas para los i486 eran 16 MHz (no muy frecuente), 20 MHz (tampoco frecuente), 25 MHz, 33 MHz, 40 MHz, 50 MHz (típicamente con duplicación del reloj), 66 MHz (con duplicación del reloj), 75 MHz (con triplicación del reloj), 100 MHz (también con triplicación del reloj) y 120 MHz (con cuatriplicación de reloj en una variante de AMD, el Am486-DX5).

El sucesor del microprocesador i486 es el Intel Pentium.



Imagen 1.2.9 Microprocesador 486

Intel Pentium es una gama de microprocesadores de quinta generación. El primer Pentium se lanzó al mercado con velocidades iniciales de 60 y 66 MHz, cache interno de 8 KB para datos y 8 KB para instrucciones; sucediendo al procesador Intel 80486. Las versiones que incluían instrucciones MMX no sólo brindaban al usuario un mejor manejo de aplicaciones multimedia, como por ejemplo, la lectura de películas en DVD sino que se ofrecían en velocidades de hasta 233 MHz, incluyendo una versión de 200 MHz y la más básica proporcionaba unos 166 MHz de reloj.

La aparición de este procesador se llevó a cabo con un movimiento económico impresionante, acabando con la competencia, que hasta entonces producía procesadores equivalentes, como es el 80386, el 80486 y sus variaciones.

Las siguientes empresas fueron afectadas por la aparición del Pentium:

- Advanced Micro Devices, Mejor conocida como AMD. Tuvo que crear sus procesadores desde cero. Este es el K5 y el K6 Cyrix, que producía 486, luego fue adquirida por VIA
- Harris
- LU-MATH

Estas últimas dos no fueron muy conocidas aunque sus versiones de procesadores de alto rendimiento (como el Harris 80386) no eran del todo compatibles.



Imagen 1.2.10 Microprocesador Pentium

El Pentium II se comercializó en versiones que funcionaban a una frecuencia de reloj de entre 166 y 450 MHz. Poseía 32 KB de memoria caché de primer nivel repartida en 16 KB para datos y otros 16 KB para instrucciones. Como novedad respecto al resto de procesadores de la época, el Pentium II se presentaba en un encapsulado SECC, con forma de cartucho. El cambio de formato de encapsulado se hizo para mejorar la disipación de calor. Este cartucho se conecta a las placas base de los equipos mediante una ranura Slot 1.



Imagen 1.2.11 Microprocesador Pentium II

El Pentium III era muy similar al Pentium II, siendo la diferencia más importante la introducción de las instrucciones SSE. Al igual que con el Pentium II, existía una versión Celeron de bajo presupuesto y una versión Xeon para quienes necesitaban de gran poder de cómputo. Esta línea ha sido finalmente reemplazada por el Pentium 4, aunque la línea Pentium M, para equipos portátiles, está basada en el Pentium III. Existen tres versiones de Pentium III: Katmai, Coppermine y Tualatin.

#### Katmai

La primera versión era muy similar al Pentium II, los primeros modelos tenían velocidades de 450 y 500 MHz.

### Coppermine

Esta versión tenía memoria caché L2 de 256 KB integrada, lo cual mejoró significativamente el rendimiento en comparación con Katmai. Tenían velocidades desde los 500 hasta 1GHz. Una versión de 1,13 GHz fue introducida al mercado poco después, pero debió ser cancelada por ser excesivamente inestable. El problema residía en que la memoria caché integrada tenía problemas para trabajar a más de 1 GHz.

La primera generación de la consola Xbox usa este procesador en una versión más estable de 733 Mhz, con 128KB de cache L2.

### Tualatin

La tercera y última versión (llamados Pentium III-S) estaba enfocada al mercado de servidores con velocidades desde 1.3 hasta 1.4 MHz.

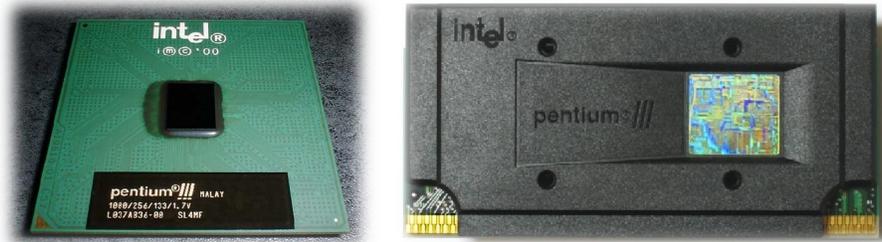


Imagen 1.2.12 Microprocesadores Pentium III

El Pentium 4 es un microprocesador de séptima generación basado en la arquitectura x86. Trabajaba a 1.4 y 1.5 GHz. El Pentium 4 no mejoró el viejo diseño de la microarquitectura Intel P6 según las dos tradicionales formas para medir el rendimiento: velocidad en el proceso de enteros u operaciones de punto flotante. La estrategia de Intel fue sacrificar el rendimiento de cada ciclo para obtener a cambio mayor cantidad de ciclos por segundo y una mejora en las instrucciones SSE. En 2004, se agregó el conjunto de instrucciones x86-64 de 64 bits al tradicional set x86 de 32 bits. Al igual que los Pentium II y Pentium III, el Pentium 4 se comercializa en una versión para equipos de bajo presupuesto (Celeron), y una orientada a servidores de gama alta (Xeon).

Los nombres en código, a partir de la evolución de las distintas versiones, son: Willamette (180 nanómetros), Northwood (130 nm), Gallatin (*Extreme Edition*, también 130 nm), Prescott (90 nm) y Cedar Mill (65 nm).



Imagen 1.2.13 Microprocesador Pentium IV

DualCore y QuadCore, procesadores orientados a negocios, diseño y videojuegos, con procesadores de dos y cuatro núcleos y velocidades desde 1.7 hasta 3.2 GHz. Aunque los precios del procesador de 4 núcleos (QuadCore) aún eran muy altos, acabaron demostrando que la tendencia de los procesadores era aumentar el número de núcleos para optimizar su rendimiento multimedia.



Imagen 1.2.14 Microprocesador DualCore, Core 2 Duo y Core 2 Quad

Intel Xeon es la actual generación de microprocesadores Intel para servidores PC. El primer procesador Xeon apareció en 1998 como Pentium II Xeon.

En 2001, el Pentium III Xeon se reemplazó por el procesador Intel Xeon. El Xeon está basado en la arquitectura NetBurst de Intel y es similar al CPU Pentium 4.



Imagen 1.2.15 Microprocesador Xeon

## 1.3 Discos duros

### 1.3.1 Estructura física

Disco duro (en inglés Hard Disk Drive, HDD) es un dispositivo de almacenamiento de datos no volátil que emplea un sistema de grabación magnética para almacenar datos digitales. Se compone de uno o más platos o discos rígidos, unidos por un mismo eje que gira a gran velocidad dentro de una caja metálica sellada. Sobre cada plato se sitúa un cabezal de lectura/escritura que flota sobre una delgada lámina de aire generada por la rotación de los discos.

A lo largo de los años, los discos duros han disminuido su precio al mismo tiempo que han multiplicado su capacidad, siendo la principal opción de almacenamiento para PC desde su aparición en los años 60. Los discos duros han mantenido su posición dominante gracias a los constantes incrementos en la densidad de grabación, que se ha mantenido a la par de las necesidades de almacenamiento.

Los tamaños también han variado mucho, desde los primeros discos IBM hasta los formatos estandarizados actualmente: 3.5" los modelos para PCs y servidores, 2.5" los modelos para dispositivos portátiles. Todos se comunican con la computadora a través del controlador de disco, empleando un interfaz estandarizado. Los más comunes hoy día son IDE (también llamado ATA o PATA), SCSI (generalmente usado en servidores y estaciones de trabajo), Serial ATA y FC (empleado exclusivamente en servidores).

Para poder utilizar un disco duro, un sistema operativo debe aplicar un formato de bajo nivel que defina una o más particiones. La operación de formateo requiere el uso de una fracción del espacio disponible en el disco, que dependerá del formato empleado. Además, los fabricantes de discos duros, SSD y tarjetas flash miden la capacidad de los mismos usando prefijos SI, que emplean múltiplos de potencias de 1000 según la normativa IEC, en lugar de los prefijos binarios clásicos de la IEEE, que emplean múltiplos de potencias de 1024, y son los usados mayoritariamente por los sistemas operativos. Esto provoca que en algunos sistemas operativos sea representado como múltiplos 1024 o como 1000, y por tanto existan ligeros errores, por ejemplo un Disco duro de 500 GB, en algunos sistemas operativos sea representado como 465 GB (Según la IEC o Gigabyte binario, que son 1024 Mb).

Existe otro tipo de almacenamiento que recibe el nombre de *Unidades de estado sólido*; aunque tienen el mismo uso y emplean los mismos interfaces, no están formadas por discos mecánicos, sino por memorias de circuitos integrados para almacenar la información. El uso de esta clase de dispositivos anteriormente se limitaba a las supercomputadoras, por su elevado precio, aunque hoy en día ya son muchísimo más asequibles para el mercado doméstico.



Imagen 1.3.1.1 Estructura física de un HDD. De izquierda a derecha, fila superior: tapa, carcasa, plato, eje; fila inferior: espuma aislante, circuito impreso de control, cabezal de lectura / escritura, actuador e imán, tornillos.

Dentro de un disco duro hay uno o varios platos (entre 2 y 4 normalmente, aunque hay hasta de 6 ó 7 platos), que son discos (de aluminio o cristal) concéntricos y que giran todos a la vez. El cabezal (dispositivo de lectura y escritura) es un conjunto de brazos alineados verticalmente que se mueven hacia dentro o fuera según convenga, todos a la vez. En la punta de dichos brazos están las cabezas de lectura/escritura, que gracias al movimiento del cabezal pueden leer tanto zonas interiores como exteriores del disco.

Cada plato posee dos caras, y es necesaria una cabeza de lectura/escritura para cada cara. Cada uno de los brazos es doble, y contiene 2 cabezas: una para leer la cara superior del plato, y otra para leer la cara inferior. Por tanto, hay 8 cabezas para leer 4 platos, aunque por cuestiones comerciales, no siempre se usan todas las caras de los discos y existen discos duros con un número impar de cabezas, o con cabezas deshabilitadas. Las cabezas de lectura/escritura nunca tocan el disco, sino que pasan muy cerca (hasta a 3 nanómetros), debido a una finísima película de aire que se forma entre éstas y los platos cuando éstos giran (algunos discos incluyen un sistema que impide que los cabezales pasen por encima de los platos hasta que alcancen una velocidad de giro que garantiza la formación de esta película). Si alguna de las cabezas llega a tocar una superficie de un plato, causaría muchos daños en él, rayándolo gravemente, debido a lo rápido que giran los platos (uno de 7.200 revoluciones por minuto se mueve a 129 km/h en el borde de un disco de 3.5 pulgadas).



Imagen 1.3.1.2 Superficie de un plato y el cabezal de lectura/escritura.

Hay varios conceptos para referirse a zonas del disco:

- Plato: Cada uno de los discos que hay dentro del disco duro.
- Cara: Cada uno de los dos lados de un plato.
- Cabeza: Número de cabezales.
- Pista: Una circunferencia dentro de una cara; la pista 0 está en el borde exterior.
- Cilindro: Conjunto de varias pistas; son todas las circunferencias que están alineadas verticalmente (una de cada cara).
- Sector: Cada una de las divisiones de una pista. El tamaño del sector no es fijo, siendo el estándar actual 512 bytes, aunque próximamente serán 4 KB. Antiguamente el número de sectores por pista era fijo, lo cual desaprovechaba el espacio significativamente, ya que en las pistas exteriores pueden almacenarse más sectores que en las interiores. Así, apareció la tecnología ZBR (grabación de bits por zonas) que aumenta el número de sectores en las pistas exteriores, y utiliza más eficientemente el disco duro.

El primer sistema de direccionamiento que se usó fue el CHS (cilindro-cabeza-sector), ya que con estos tres valores se puede situar un dato cualquiera del disco. Más adelante se creó otro sistema más sencillo: LBA (direccionamiento lógico de bloques), que consiste en dividir el disco entero en sectores y asignar a cada uno un único número. Éste es el que actualmente se usa.



Imagen 1.3.1.3 Cilindro, Cabeza y Sector

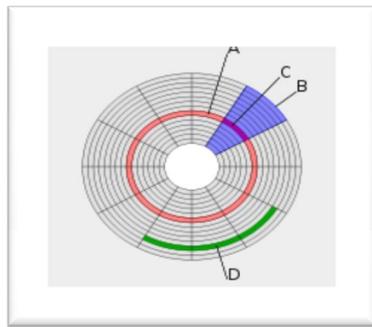


Imagen 1.3.1.4 Pista (A), Sector (B), Sector de una pista (C), Clúster (D)

## 1.4.2 Tipos de conexión

Existen distintos tipos de conexión con la placa base, los cuales son: SATA, IDE, SCSI o SAS:

- IDE: Integrated Device Electronics (Dispositivo con electrónica integrada) o ATA (Advanced Technology Attachment), controla los dispositivos de almacenamiento masivo de datos, como los discos duros y ATAPI (Advanced Technology Attachment Packet Interface) Hasta aproximadamente el 2004, el estándar principal por su versatilidad.
- SCSI: Son interfaces preparadas para discos duros de gran capacidad de almacenamiento y velocidad de rotación. Se presentan bajo tres especificaciones: SCSI Estándar (Standard SCSI), SCSI Rápido (Fast SCSI) y SCSI Ancho-Rápido (Fast-Wide SCSI). Su tiempo medio de acceso puede llegar a 7 milisegundos y su

velocidad de transmisión secuencial de información puede alcanzar teóricamente los 5 Mbps en los discos SCSI Estándares, los 10 Mbps en los discos SCSI Rápidos y los 20 Mbps en los discos SCSI Anchos-Rápidos (SCSI-2). Un controlador SCSI puede manejar hasta 7 discos duros SCSI (o 7 periféricos SCSI) con conexión tipo margarita (daisy-chain). A diferencia de los discos IDE, pueden trabajar asincrónicamente con relación al microprocesador, lo que posibilita una mayor velocidad de transferencia.

- SATA (Serial ATA): El más novedoso de los estándares de conexión, utiliza un bus serie para la transmisión de datos. Notablemente más rápido y eficiente que IDE. Existen tres versiones, SATA 1 con velocidad de transferencia de hasta 150 MB/s (hoy día descatalogado), SATA 2 de hasta 300 MB/s, el más extendido en la actualidad; y por último SATA 3 de hasta 600 MB/s el cual se está empezando a hacer hueco en el mercado. Físicamente es mucho más pequeño y cómodo que los IDE, además de permitir conexión en caliente.
- SAS (*Serial Attached SCSI*): Interfaz de transferencia de datos en serie, sucesor del SCSI paralelo, aunque sigue utilizando comandos SCSI para interactuar con los dispositivos SAS. Aumenta la velocidad y permite la conexión y desconexión en caliente. Una de las principales características es que aumenta la velocidad de transferencia al aumentar el número de dispositivos conectados, es decir, puede gestionar una tasa de transferencia constante para cada dispositivo conectado, además de terminar con la limitación de 16 dispositivos existente en SCSI, es por ello que se pronostica que la tecnología SAS irá reemplazando a su predecesora SCSI. Además, el conector es el mismo que en la interfaz SATA y permite utilizar estos discos duros, para aplicaciones con menos necesidad de velocidad, ahorrando costos. Por lo tanto, las unidades SATA pueden ser utilizadas por controladoras SAS pero no a la inversa, una controladora SATA no reconoce discos SAS.

### 1.4.3 Estructura lógica

Dentro del disco se encuentran:

- El Master Boot Record (en el sector de arranque), que contiene la tabla de particiones.
- Las particiones, necesarias para poder colocar los sistemas de archivos.

Integridad

Debido a la distancia extremadamente pequeña entre los cabezales y la superficie del disco, cualquier contaminación de los cabezales de lectura/escritura o las fuentes puede dar lugar a un accidente en los cabezales, un fallo del disco en el que el cabezal raya la superficie de la fuente, a menudo moliendo la fina película magnética y causando la pérdida de datos. Estos accidentes pueden ser causados por un fallo electrónico, un repentino corte en el suministro eléctrico, golpes físicos, el desgaste, la corrosión o debido a que los cabezales o las fuentes sean de pobre fabricación.

Características de un disco duro

Las características que se deben tener en cuenta en un disco duro son:

- Tiempo medio de acceso: Tiempo medio que tarda la aguja en situarse en la pista y el sector deseado; es la suma del Tiempo medio de búsqueda (situarse en la pista), Tiempo de lectura/escritura y la Latencia media (situarse en el sector).
- Tiempo medio de búsqueda: Tiempo medio que tarda la aguja en situarse en la pista deseada; es la mitad del tiempo empleado por la aguja en ir desde la pista más periférica hasta la más central del disco.
- Tiempo de lectura/escritura: Tiempo medio que tarda el disco en leer o escribir nueva información: Depende de la cantidad de información que se quiere leer o escribir, el tamaño de bloque, el número de cabezales, el tiempo por vuelta y la cantidad de sectores por pista.
- Latencia media: Tiempo medio que tarda la aguja en situarse en el sector deseado; es la mitad del tiempo empleado en una rotación completa del disco.
- Velocidad de rotación: Revoluciones por minuto de los platos. A mayor velocidad de rotación, menor latencia media.
- Tasa de transferencia: Velocidad a la que puede transferir la información a la computadora una vez la aguja está situada en la pista y sector correctos. Puede ser velocidad sostenida o de pico.

## Capítulo 2 -- Análisis, diseño y ejecución de la estrategia de pruebas para sistemas de software

Durante el ciclo de vida de todo proyecto de desarrollo de software, una parte importante de este ciclo es la de las pruebas al sistema. Desafortunadamente aún no hay una cultura que fortalezca esta práctica de probar los sistemas, puesto que muchas veces los tiempos de entrega no permiten realizar estas actividades, sin embargo cada vez hay más técnicas que han robustecido esta área, hasta llegar incluso a la automatización de las pruebas, permitiendo así economizar tiempo, recursos materiales y humanos.

En este capítulo se dará una breve descripción de los tipos de pruebas que se pueden aplicar a los sistemas y que están reconocidas dentro de la etapa de testing de un proyecto de desarrollo de software.

El campus virtual es una plataforma tecnológica que la Subdirección de medios Educativos en Informática desarrolló con la finalidad de satisfacer la demanda de capacitación al personal del IFE, los altos costos del envío de material didáctico y la planeación y seguimiento a los más de 3000 empleados del instituto en sus diferentes Juntas locales y Distritales así como en oficinas centrales, permitieron la implementación de este proyecto.

Una de mis principales actividades como informático administrativo era la capacitación a cierto número de participantes, de acuerdo a la planeación de la subdirección. Esta capacitación incluía la ingesta de insumos al sistema, así como la creación de los mismos para el caso de materiales didácticos.

- Analista de sistemas (Analista de calidad Auditoría del sistema)
  - o Descripción de actividades Auditoría de calidad para el sistema de Campus virtual
    - Levantamiento de requerimientos
    - Elaboración de casos de uso
    - Elaboración de casos de prueba y matrices de prueba
    - Descripción de tipos de prueba (caja negra, caja blanca, contrato, formato, usabilidad, seguridad, estrés, etc.)
    - Elaboración de documentos de liberación

En la Subdirección de Medios Educativos en Informática, además del Campus Virtual, se implementaban otros desarrollos que surgían de la necesidad de comunicación en las diferentes áreas del instituto, así como el incremento en la demanda de cursos de capacitación y programas alternos para su solución.

Una de mis principales actividades era la documentación de los nuevos sistemas, el análisis y levantamiento de requerimientos, así como la etapa de pruebas y capacitación

de los mismos, por lo que para brindar una visión más global, se enlistarán y detallaran las etapas del ciclo de vida del software.

## **2.1 Análisis**

### **2.1.1 Levantamiento de requerimientos**

En esta etapa se hacía una reunión Analista - Usuario (un representante institucional, departamental o cliente particular) para identificar las metas globales, se analizaban las perspectivas del cliente, sus necesidades y requerimientos, sobre la planificación temporal y presupuestal y otros puntos que puedan ayudar a la identificación y desarrollo del proyecto, como:

- Reconocimiento del problema.
- Evaluación y Síntesis.
- Modelado.
- Especificación.
- Revisión

Antes de cada reunión analista- usuario, se preparaba un documento conceptual del proyecto, donde a manera de minuta se establecen los requerimientos y son firmados de conformidad con el usuario.

A continuación se detalla brevemente algunas de las actividades y de las reglas de negocio del sistema que se debían tomar en cuenta para la realización de las mismas.

- Alta de insumos del sistema (grupos, foros, chats, materiales y alumnos)

Ingresar a la administración del campus virtual del IFE, Dar de alta el curso que se impartirá, para cada curso se dan de alta los grupos.

- Alumnos, instructores y coordinadores de grupo.

Por medio de la interfaz, para cada grupo que se dio de alta, se presentan los coordinadores de grupo y instructores a los alumnos que realizaran el curso, tomando en cuenta el estatus de cada participante.

Los estatus son:

En grupo: El participante está integrado al grupo. Sin embargo; aún no puede ver los materiales asignados al grupo.

Inscrito: El participante está asociado al grupo y puede ver todos los materiales del grupo siempre y cuando estos estén vigentes y activos.

Terminó Curso: el participante sólo puede ingresar a los foros del grupo y verificar su calificación.

Ya no labora en el IFE: El participante ya no puede ingresar al sistema por que su cuenta se ha deshabilitado.

- Foros

Existen 2 tipos de foros, los de grupo y los de curso.

Foros de curso: Todos los alumnos pueden leer y escribir sobre los foros sin importar a qué grupo pertenezcan.

Foros de grupo: Los alumnos solamente pueden leer y escribir sobre los foros del grupo en los que estén dados de alta.

Para cada grupo que se dio de alta; se vinculan los foros en los que podrán participar los alumnos. Las propiedades del foro son: Nombre, pertenece a un grupo o un curso, Fecha inicio-fin de vigencia y activo.

- Materiales

Existen 2 tipos principales de materiales: los de grupo y los de curso; que a su vez se pueden catalogar en lecturas, procedimientos, guías de usuario y cápsulas demostrativas.

Materiales de curso: Todos los alumnos pueden descargar los materiales sin importar a qué grupo pertenezcan.

Para cada grupo que se dio de alta, se vinculan los materiales que podrán descargar los alumnos, las propiedades de los materiales son: Nombre, pertenece a: grupo o curso, Fecha inicio y fin de vigencia, tipo de material y activo.

- Chats

Existen 2 tipos de chats: los de grupo y los de curso.

Chat de curso: Todos los alumnos pueden participar sobre los chats sin importar a qué grupo pertenezcan.

Chat de grupo: Los alumnos solamente pueden participar sobre los chats del grupo al cual están inscritos.

Para cada grupo que se dio de alta, se vinculan los chats en los que podrán participar los alumnos, las propiedades del chat son: Nombre, pertenece a: grupo o curso, Fecha inicio y fin de vigencia y activo.

- Implementación de materiales didácticos (Multimedia)

Por medio de la herramienta de diseño de cápsulas y simulaciones, se realiza un recorrido del sistema en el explorador (flujo básico) y se toman en cuenta todas las secciones de la pantalla que se deseen explicar. Una vez grabado el flujo básico, se ingresa nuevamente a la herramienta, que se encarga de separar la grabación en diapositivas y asignar

tiempos a los movimientos del mouse y a las capturas de texto que se realizaron. Para cada acción que se realizó, se pueden agregar dos elementos más, un indicador de luz que permite resaltar la sección o elemento de la pantalla y un globo de texto donde se capturará el texto explicativo para el usuario. Adicionalmente, para el caso de las simulaciones, se insertan validaciones sobre campos que simulan las acciones del usuario en el sistema real y se muestra la respuesta del sistema.

### **2.1.2 Casos de uso**

Un caso de uso es una técnica para la captura de requisitos potenciales de un nuevo sistema o una actualización de software. Cada caso de uso proporciona uno o más escenarios que indican cómo debería interactuar el sistema con el usuario o con otro sistema para conseguir un objetivo específico. Normalmente, en los casos de usos se evita el empleo de jergas técnicas, prefiriendo en su lugar un lenguaje más cercano al usuario final. En ocasiones, se utiliza a usuarios sin experiencia junto a los analistas para el desarrollo de casos de uso. Estos casos de uso, tuve oportunidad de aplicarlos en todos y cada uno de los proyectos que se realizaron en el Instituto Federal Electoral, durante mi desempeño laboral.

En otras palabras, un caso de uso es una secuencia de interacciones que se desarrollarán entre un sistema y sus actores en respuesta a un evento que inicia un actor principal sobre el propio sistema. Los diagramas de casos de uso sirven para especificar la comunicación y el comportamiento de un sistema mediante su interacción con los usuarios y/u otros sistemas. Los diagramas de casos de uso se utilizan para ilustrar los requerimientos del sistema al mostrar cómo reacciona a eventos que se producen en su ámbito o en él mismo.

### **2.2 Pruebas**

Las pruebas de software, en inglés testing, son los procesos que permiten verificar y revelar la calidad de un producto software. Son utilizadas para identificar posibles fallos de implementación, calidad, o usabilidad de un programa de ordenador o videojuego.

Dicho de otro modo, se ejecuta un programa y mediante técnicas experimentales se trata de descubrir qué errores tiene.

Es necesario testear los nuevos programas en un entorno de pruebas separado físicamente del de producción. Para crear un entorno de pruebas en una máquina independiente, se requiere crear las mismas condiciones o lo más parecido posible que en la máquina de producción para prevenir comportamientos inesperados por la diferencia de ambientes.

## **2.2.1 Elaboración de casos y matrices de prueba**

Casos de prueba o Test Case, son un conjunto de condiciones o variables bajo las cuáles el analista determinará si el requisito de una aplicación es parcial o completamente satisfactorio.

Se pueden realizar muchos casos de prueba para determinar que un requisito es completamente satisfactorio. Con el propósito de comprobar que todos los requisitos de una aplicación son revisados, debe haber al menos un caso de prueba para cada requisito, a menos que un requisito tenga requisitos secundarios. En ese caso, cada requisito secundario deberá tener por lo menos un caso de prueba. Algunas metodologías como RUP recomiendan el crear por lo menos dos casos de prueba para cada requisito. Uno de ellos debe realizar la prueba positiva de los requisitos y el otro debe realizar la prueba negativa.

Si la aplicación es creada sin requisitos formales, entonces los casos de prueba se escriben basados en la operación normal de programas de una clase similar.

Lo que caracteriza un escrito formal de caso de prueba es que hay una entrada conocida y una salida esperada, los cuales son formulados antes de que se ejecute la prueba. La entrada conocida debe probar una precondition y la salida esperada debe probar una postcondición.

Bajo circunstancias especiales, podría haber la necesidad de ejecutar la prueba, producir resultados, y luego un equipo de expertos evaluaría si los resultados se pueden considerar como "Correctos". Esto sucede a menudo en la determinación del número del rendimiento de productos nuevos. La primera prueba se toma como línea base para los subsecuentes ciclos de pruebas/lanzamiento del producto.

Los casos de prueba escritos, incluyen una descripción de la funcionalidad que se probará, la cuál es tomada ya sea de los requisitos o de los casos de uso, y la preparación requerida para asegurarse de que la prueba pueda ser dirigida.

Los casos de prueba escritos se recogen generalmente en una suite de pruebas.

### **2.2.1.1 Estructura de los casos de prueba**

Formalmente, los casos de prueba escritos en los diferentes proyectos que he trabajado, y basados en las metodologías que explicaré más adelante, consisten principalmente en tres partes:

Visión general:

Introducción: Contiene información general acerca de los Casos de Prueba.

Identificador: Es un identificador único para futuras referencias, por ejemplo, mientras se describe un defecto encontrado.

Dueño/creador: Es el nombre del analista o diseñador de pruebas, quien ha desarrollado pruebas o es responsable de su desarrollo.

Versión: La actual definición del caso de prueba.

Nombre el caso de prueba: Debe ser un título entendible para la fácil comprensión del propósito del caso de prueba y su campo de aplicación.

Identificador de requerimientos: El cual está incluido por el caso de prueba. También aquí puede ser identificador de casos de uso o especificación funcional.

Propósito: Contiene una breve descripción del propósito de la prueba, y la funcionalidad que revisa.

Dependencias.

Actividades de los casos de prueba

Ambiente de prueba/configuración: contiene información acerca de la configuración del hardware o software en el cuál se ejecutará el caso de prueba.

Inicialización/Precondiciones: Describe acciones, que deben ser ejecutadas antes de que los casos de prueba se hayan inicializado. Por ejemplo, debemos abrir algún archivo.

Finalización/ Postcondiciones: Describe acciones, que deben ser ejecutadas después de realizado el caso de prueba. Por ejemplo si el caso de prueba borra la base de datos, el analista debe restaurarla antes de que otro caso de prueba sea ejecutado.

Acciones: Pasos a realizar para completar la prueba.

Descripción de los datos de entrada.

Resultados

Resultados esperados: Contiene una descripción de lo que el analista debería ver tras haber completado todos los pasos de la prueba.

Resultados obtenidos: Contienen una breve descripción de lo que el analista encuentra después de que los pasos de prueba se hayan completado. Esto se sustituye a menudo con un Correcto/Fallido. Si un caso de prueba falla, frecuentemente la referencia al defecto implicado se debe enumerar en esta columna.

### **2.2.1.2 Registro de defectos**

En general, un **bug** es cuando el programa no realiza lo que el usuario espera, o bien, puede describirse como un error de programación o un fallo en la semántica de un programa de ordenador.

Una práctica común, y que implementamos en todos los proyectos, es que el proceso de pruebas de un programa sea realizaba por un grupo independiente de "testers" al finalizar su desarrollo y antes de sacarlo a producción.

Otra práctica, era distribuir una versión no final del producto para que sean los propios usuarios los que la prueben. En ambos casos, a la versión del producto en pruebas y que es anterior a la versión final, se le denomina beta, y a dicha fase de pruebas, beta testing.

Puede además existir una versión anterior en el proceso de desarrollo llamada alpha, en la que el programa, aunque incompleto, dispone de funcionalidad básica y puede ser testeado.

Finalmente y antes de salir al mercado o producción, es cada vez más habitual que se realice una fase de RTM testing (Release To Market), dónde se comprueba cada funcionalidad del programa completo en entornos de producción.

La importancia de la detección oportuna

En la cadena de valor del desarrollo de un software específico, el proceso de prueba es clave a la hora de detectar errores o fallas. Considerábamos conceptos como estabilidad, escalabilidad, eficiencia y seguridad, que se relacionan a la calidad de un producto bien desarrollado. Las aplicaciones de software crecían en complejidad y tamaño, y por consiguiente también en costos. Era crucial verificar y evaluar la calidad de lo construido de modo de minimizar el costo de su reparación. Mientras antes se detecte una falla, más barata es su corrección.

## **2.3 Tipos de pruebas**

### **2.3.1 Pruebas unitarias**

Una prueba unitaria es una forma de probar el correcto funcionamiento de un módulo de código. Esto sirve para asegurar que cada uno de los módulos funcione correctamente por separado. Luego, con las Pruebas de Integración, se aseguraba el correcto funcionamiento del sistema o subsistema bajo pruebas.

La idea era escribir casos de prueba para cada función o método, de forma que cada caso sea independiente del resto.

Características:

Para que una prueba unitaria sea buena se deben cumplir los siguientes requisitos:

- Automatizable: No debería requerirse una intervención manual. Esto es especialmente útil para integración continua.
- Completas: Deben cubrir la mayor cantidad de código.
- Repetibles o Reutilizables: No se deben crear pruebas que sólo puedan ser ejecutadas una sola vez. También es útil para integración continua.
- Independientes: La ejecución de una prueba no debe afectar a la ejecución de otra.
- Profesionales: Las pruebas deben ser consideradas igual que el código, con la misma profesionalidad, documentación, etc.
- Aunque estos requisitos no tienen que ser cumplidos al pie de la letra, se recomienda seguirlos o de lo contrario las pruebas pierden parte de su función.

Ventajas:

- El objetivo de las pruebas unitarias es aislar cada parte del programa y mostrar que las partes individuales son correctas. Estas pruebas aisladas proporcionan cinco ventajas básicas:
- Fomentan el cambio: Las pruebas unitarias facilitan que el programador cambie el código para mejorar su estructura (lo que se ha dado en llamar refactorización), puesto que permiten hacer pruebas sobre los cambios y así asegurarse de que los nuevos cambios no han introducido errores.
- Simplifica la integración: Puesto que permiten llegar a la fase de integración con un grado alto de seguridad de que el código está funcionando correctamente. De esta manera se facilitan las pruebas de integración.
- Documenta el código: Las propias pruebas son documentación del código puesto que ahí se puede ver cómo utilizarlo.
- Separación de la interfaz y la implementación: Dado que la única interacción entre los casos de prueba y las unidades bajo prueba son las interfaces de estas últimas, se puede cambiar cualquiera de los dos sin afectar al otro.
- Los errores están más acotados y son más fáciles de localizar

Limitaciones:

Las pruebas unitarias no descubrirán todos los errores del código. Por definición, sólo prueban las unidades por sí solas. Por lo tanto, no descubrirán errores de integración, problemas de rendimiento y otros problemas que afectan a todo el sistema en su conjunto. Las pruebas unitarias sólo eran efectivas si se usan en conjunto con otras pruebas de software.

### **2.3.2 Pruebas de Integración**

Pruebas integrales o pruebas de integración son aquellas que se realizan en el ámbito del desarrollo de software una vez que se han aprobado las pruebas unitarias. Únicamente se refieren a la prueba o pruebas de todos los elementos unitarios que componen un proceso, hecha en conjunto, de una sola vez. Consistía en realizar pruebas para verificar que un gran conjunto de partes de software funcionan juntos.

### **2.3.3 Caja blanca**

Se denomina cajas blancas a un tipo de pruebas de software que se realiza sobre las funciones internas de un módulo. Así como las pruebas de caja negra ejercitan los requisitos funcionales desde el exterior del módulo, las de caja blanca están dirigidas a las funciones internas. Entre las técnicas usadas se encuentran; la cobertura de caminos (pruebas que hagan que se recorran todos los posibles caminos de ejecución), pruebas sobre las expresiones lógico-aritméticas, pruebas de camino de datos (definición-uso de variables), comprobación de bucles (se verifican los bucles para 0,1 y n iteraciones, y luego para las iteraciones máximas, máximas menos uno y más uno. Las pruebas de caja blanca se llevan a cabo en primer lugar, sobre un módulo concreto, para luego realizar las de caja negra sobre varios subsistemas (integración). En los sistemas orientados a objetos, las pruebas de caja blanca pueden aplicarse a los métodos de la clase, pero según varias opiniones, ese esfuerzo debería dedicarse a otro tipo de pruebas más especializadas (un argumento podría ser que los métodos de una clase suelen ser menos complejos que los de una función de programación estructurada).

### **2.3.4 Caja negra**

Se denomina caja negra a aquel elemento que es estudiado desde el punto de vista de las entradas que recibe y las salidas o respuestas que produce, sin tener en cuenta su funcionamiento interno. En otras palabras, de una caja negra nos interesaba su forma de interactuar con el medio que le rodea (en ocasiones, otros elementos que también podrían ser cajas negras) entendiendo qué es lo que hace, pero sin dar importancia a cómo lo hace. Por tanto, de una caja negra debían estar muy bien definidas sus entradas y salidas, es decir, su interfaz; en cambio, no se precisa definir ni conocer los detalles internos de su funcionamiento.

En pruebas de software, conociendo una función específica para la que fue diseñado el producto, se pueden diseñar pruebas que demuestren que dicha función está bien realizada. Dichas pruebas eran llevadas a cabo sobre la interfaz del software, es decir, de la función, actuando sobre ella como una caja negra, proporcionando unas entradas y estudiando las salidas para ver si son o no las esperadas.

### 2.3.5 Pruebas de regresión

Se denominan Pruebas de regresión a cualquier tipo de pruebas de software que intentan descubrir las causas de nuevos errores (bugs), carencias de funcionalidad, o divergencias funcionales con respecto al comportamiento esperado del software, inducidos por cambios recientemente realizados en partes de la aplicación que anteriormente al citado cambio no eran propensas a este tipo de error. Esto implica que el error tratado se reproduce como consecuencia inesperada del citado cambio en el programa.

Este tipo de cambio podía ser debido a prácticas no adecuadas de control de versiones, falta de consideración acerca del ámbito o contexto de producción final y extensibilidad del error que fue corregido (fragilidad de la corrección), o simplemente una consecuencia del rediseño de la aplicación.

Por lo tanto, en la mayoría de los proyectos de software se considera una buena práctica que cuando se localiza y corrige un bug, se grabe una prueba que exponga el bug y se vuelvan a probar regularmente después de los cambios subsiguientes que experimente el programa.

Existen herramientas de software que permiten detectar este tipo de errores de manera parcial o totalmente automatizada, la práctica habitual es que este tipo de pruebas se ejecutaban en cada uno de los pasos del ciclo de vida del desarrollo del software.

### 2.3.6 Pruebas funcionales

Una prueba funcional es una prueba basada en la ejecución, revisión y retroalimentación de las funcionalidades previamente diseñadas para el software. Las pruebas funcionales se hacen mediante el diseño de modelos de prueba que buscan evaluar cada una de las opciones con las que cuenta el paquete informático.

Las pruebas de validación en la ingeniería de software son el proceso de revisión que el sistema de software producido cumple con las especificaciones y que cumple su cometido. Era normalmente una parte del proceso de pruebas de software de los proyectos, el también utilizar técnicas tales como evaluaciones, inspecciones, y tutoriales. La validación es el proceso de comprobar lo que se ha especificado es lo que el usuario realmente quería.

Se trata de evaluar el sistema o parte de este, durante o al final del desarrollo, para determinar si satisface los requisitos iniciales. La pregunta a realizarse es: ¿Es esto lo que el usuario quiere?

Características

- Comprobar que se satisfacen los requisitos:

- Se usan la mismas técnicas, pero con otro objetivo.
- No hay programas de prueba, sino sólo el código final de la aplicación.
- Se prueba el programa completo.
- Uno o varios casos de prueba por cada requisito o caso de uso especificado.
- Se prueba también rendimiento, capacidad, etc. (y no sólo resultados correctos).
- Pruebas alfa (desarrolladores) y beta (usuarios).

### 2.3.7 Metodologías de pruebas

#### Modelo V

Es el modelo de desarrollo de software más utilizado en mi experiencia, donde el desarrollo y el testing son dos ramas que apuntan a los mismos niveles, ya que para cada nivel de desarrollo existe su correspondiente nivel de Testing.

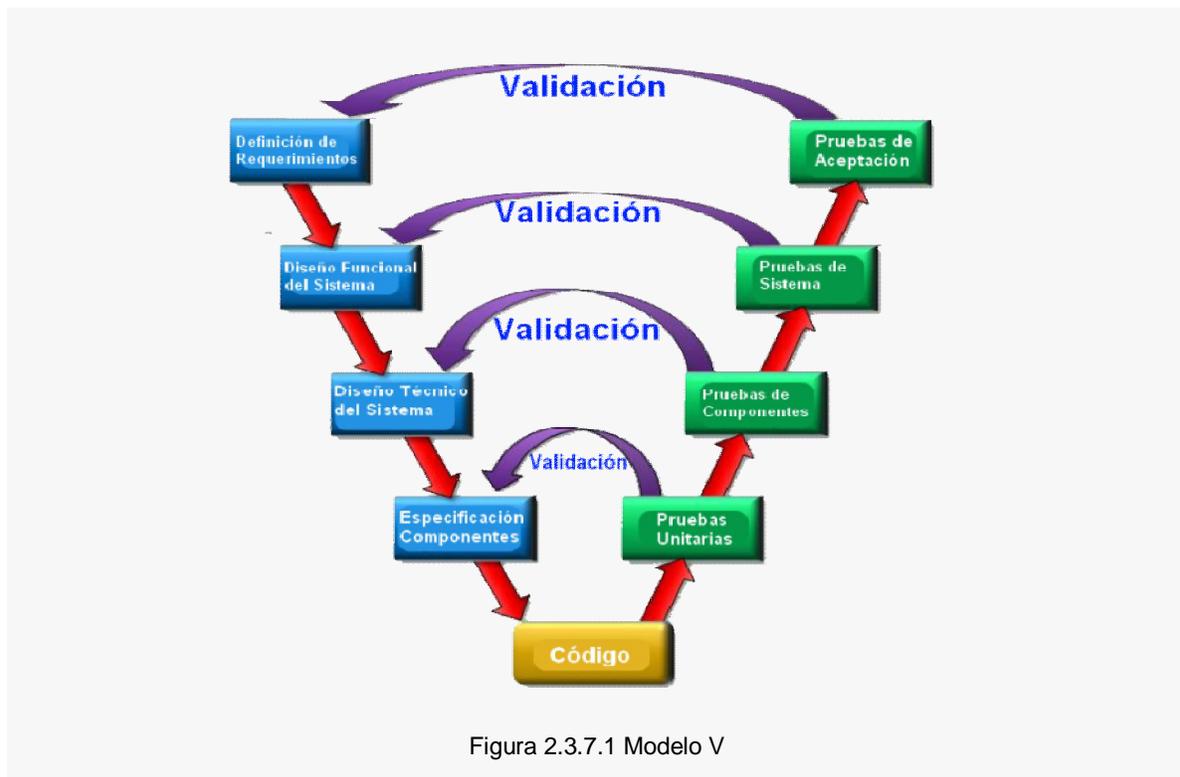


Figura 2.3.7.1 Modelo V

En la práctica, este modelo puede tener más o menos niveles dependiendo del proyecto y el producto que se desarrolló. Se creó como respuesta a los distintos problemas que ocurrían en el desarrollo en cascada (Waterfall Model). El Modelo V, propone realizar testing desde el comienzo del proyecto, realizando Testing Estático en cada una de las fases de desarrollo, preparando las pruebas para esos niveles y luego ejecutando las mismas en forma ascendente se tiene siempre un control sobre cada una de las etapas.

## Modelo Iterativo

Es una alternativa a los modelos secuenciales donde se propone una cantidad N de ciclos de desarrollo pequeños donde se van agregando características al producto, verificándolas y lanzadas a Producción al terminar el mismo.

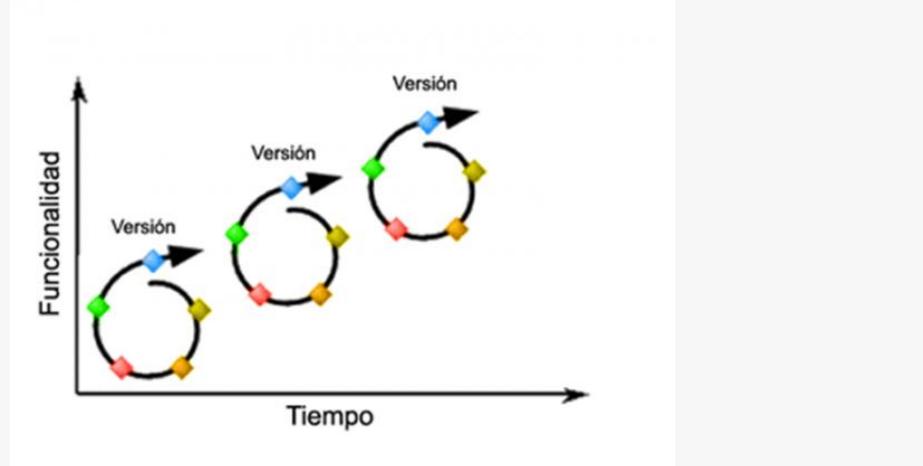


Figura 2.3.7.2 Modelo Iterativo

El testing en cada iteración va a ir siendo mayor ya que debíamos correr pruebas de regresión para asegurar que las nuevas funcionalidades no afecten a las ya desarrolladas, por lo que las pruebas automatizadas son un factor de relevancia. Este modelo provee de una forma de trabajo tal que el cliente puede tener salidas a producción de una manera más rápida, ya que las funcionalidades críticas pueden ser desarrolladas en las primeras iteraciones y dejar las funcionalidades menores para posteriores entregas. Al mismo tiempo, el costo de desarrollo se ve dividido en pequeños proyectos por lo que la inversión inicial puede ser menor.

Los modelos iterativos más relevantes son RUP y XP

Principios de todos los modelos

- Cada actividad de desarrollo debe ser probada
- Ninguna porción del software puede quedar sin ser probada, si ha sido desarrollada tanto de forma (una única fase) o iterativa
- Cada nivel de prueba debería ser probado de forma específica
- Cada nivel de pruebas cuenta con sus objetivos de prueba propios
- Las pruebas llevadas a cabo en cada nivel deben reflejar estos objetivos
- El proceso de pruebas comienza con mucha antelación a la ejecución de pruebas
- Tan pronto como el desarrollo comienza puede comenzar la preparación de las pruebas correspondientes
- También es el caso de las revisiones de documentos comenzando por los conceptos, especificación y el diseño global (en conjunto).

### **Capítulo 3 Implementación (Software e infraestructura) del Sistema de Verificación y Monitoreo del IFE**

Como Informático especializado tuve la oportunidad de participar en uno de los proyectos más importantes del IFE, participando no sólo en la parte de testing, sino también en el área de infraestructura y configuración de servidores de la solución.

La reforma electoral 2007-2008 modificó profundamente el marco que regula la competencia política en México. La piedra angular de esta renovación institucional la constituye un nuevo modelo de acceso a la radio y la televisión para los partidos políticos las autoridades electorales, animado por una concepción distinta de la posición que los medios de comunicación social deben ocupar en la vida democrática.

El sello distintivo del nuevo modelo es que es la única vía para difundir mensajes electorales en la radio y la televisión a través de los tiempos oficiales del Estado, administrados por el IFE, conforme a criterios claramente establecidos en la Constitución y la ley. Por lo tanto, queda prohibido contratar o adquirir publicidad en los medios electrónicos. Así, todo promocional que los partidos políticos y candidatos deseen transmitir debe ser entregado al IFE para que se difunda en los tiempos del Estado.

La solución del sistema se conformó de cuatro bloques generales: Digitalización (Servidor 1), Almacenamiento (Servidor 2), Detección (Servidor 3) e Integración (Servidor 4). Los cuatro módulos operan entre sí de manera dinámica a través de la interconexión de tres redes virtuales: RED 1, RED 2, e Intranet.

La solución permite el cruce de dos vías de información:

1. Digitalizada: Información recibida (material) por los sintonizadores y procesada por los equipos digitalizadores. El material generado se almacena temporalmente en los equipos digitalizadores y posteriormente en los equipos de almacenamiento.
2. Información IFE: Material digital que entregan los Partidos Políticos, Coaliciones y Autoridades Electorales al IFE y que fueron entregados a los medios de comunicación.

Ambos materiales se comparan con el servidor 3. Verifica que el contenido de Información IFE coincida con el contenido de la información digitalizada. Envía la información detectada al servidor de Servidor 4.

El servidor Servidor 4 integra toda la información para generar reportes y gráficas que le permitan al IFE informar los resultados obtenidos conforme a pauta.

La administración del proyecto consiste en la configuración y el correcto funcionamiento de los equipos y garantizar la comunicación entre ellos para generar los reportes solicitados.

### 3.1 Configuraciones generales

#### 3.1.1 Nombre del equipo

Para asignar nombre a cada equipo, se debe seguir la nomenclatura dispuesta por el área de sistemas para identificar a cada equipo del Centro de Verificación y Monitoreo. En la Tabla 3.1.1.1 se ejemplifican los dos casos: junta local y junta distrital:

CEVEM	Domino	Ejemplo
Junta Local	jl-Estado.XXX.org.mx	jl-chih.XXX.org.mx
Junta Distrital	jdDD- Estado.XXX.org.mx	jd0X-ags.XXX.org.mx

Tabla3.1.1.1 Definición de juntas.

Equipo	CEVEM en Junta Local	CEVEM en Junta Distrital	Ejemplo
SERVIDOR 1	dgN-cevem.jl- Estado.XXX.org.mx	dgN-cevem.jdDD- Estado.XXX.org.mx	dg1-cevem.jl-chis.XXX.org.mx dg2-cevem.jd05-chis.XXX.org.mx
SERVIDOR 2	xd-cevem.jl- Estado.XXX.org.mx	Servidor 2.jdDD- Estado.XXX.org.mx	xd-cevem.jl-mex.XXX.org.mx xd-cevem.jd38-mex.XXX.org.mx
SERVIDOR 3	Servidor 3N-cevem.jl- Estado.XXX.org.mx	Servidor 3N- cevem.jdDD- Estado.XXX.org.mx	Servidor 32-cevem.jl-qro.XXX.org.mx Servidor 31-cevem.jd02-qro.XXX.org.mx
SERVIDOR 4	Servidor 4.jl- Estado.XXX.org.mx	Servidor 4.jdDD- Estado.XXX.org.mx	Servidor 4.jl-ags.XXX.org.mx Servidor 4.jd03-ags.XXX.org.mx
PC	pc1-cevem. . .pcN-cevem	pc1-cevem. . .pcN-cevem	pc1-cevem.jl-hgo.XXX.org.mx

Tabla 3.1.1.2 Nombres para cada equipo.

#### NOTAS:

**Estado**=Abreviatura utilizada por el Instituto para los estados.

**DD**=Dígito(s) del número de Distrito de la Junta Ejecutiva.

En la Tabla 3.1.1.2 se muestra la configuración de nombres para cada equipo.

### 3.1.2 Interfaces y Direcciones IP

Se asignaron direcciones IP con base en los requerimientos del IFE de acuerdo a los siguientes nombres para cada interfaz de conexión (NIC . Network Interface Connection):

Parámetros	Intranet (Red verde)	Red 1 (Red Roja)	Red 2 (Red Azul)	Red de Máquinas Virtuales
Intranet	10.EDO+xx.DD+100.0/xx	192.168.xx.0/xx	192.168.xx.0/xx	192.168.xx.0/xx
Máscara de Red	255.255.xxx.0	255.255.xxx.0	255.255.xxx.0	255.255.xxx.0
Gateway	10.EDO+xx.DD+100.2xx	192.168.xx.2xx	192.168.xx.2xx	192.168.xx.2xx
Dirección IP Switch	10.EDO+xx.DD+100.250	N/A	N/A	N/A

Tabla 3.1.2.1. Parámetros de configuración para VLAN's.

Cada equipo tiene dos conexiones para dos de las tres redes posibles. En la Tabla 3.1.2.2 se muestra la VLAN y el rango de direcciones IP correspondiente a cada servidor:

Equipo	Dirección IP Intranet	Dirección IP Red 1	Dirección IP red 2
SERVIDOR 4 (IBM 3550)	10.EDO+xx.DD+100.xx	N/A	192.168.xx.xx
SERVIDOR 2 (IBM 3650)	N/A	192.168.xx.11	192.168.xx.xx
Servidor 1	10.EDO+xx.DD+100.xx a 10.EDO+xx.DD+100.xx	192.168.xx.xx a  192.168.xx.xx	N/A
SERVIDOR 3	10.EDO+xx.DD+100.xx a 10.EDO+xx.DD+100.xx	N/A	192.168.xx.xx. a  192.168.xx.xx
PC	10.EDO+xx.DD+100.xxx a 10.EDO+xx.DD+100.xxx	N/A	N/A
UPS	10.EDO+xx.DD+100.x a 10.EDO+xx.DD+100.xx	N/A	N/A

Tabla 3.1.2.2 VLAN y Direcciones IP.

En la Figura 3.1.2.3 se puede ver la interconexión de las tres redes en las que opera la solución:

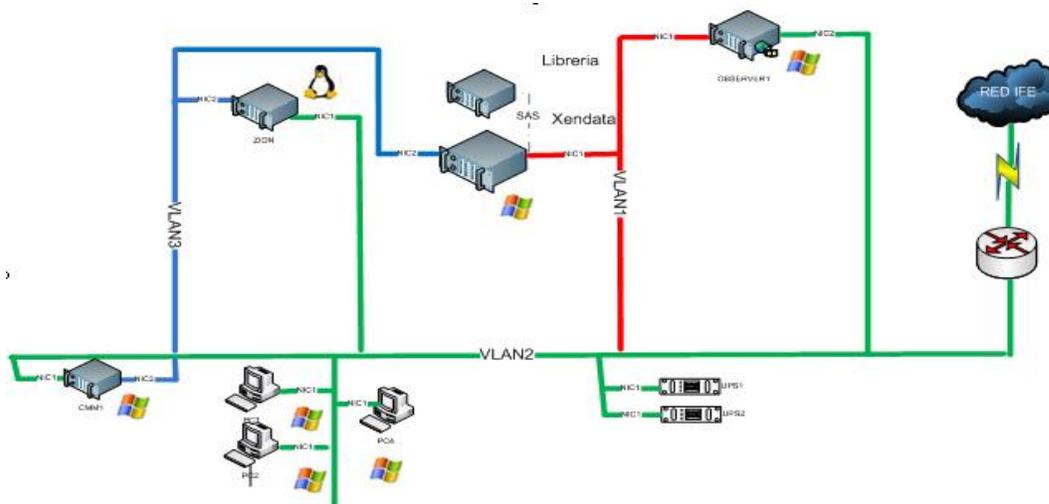


Figura 3.1.2.3 Conexión de VLANs.

### 3.1.3 Zona Horaria y Servidor NTP

Hay tres tipos de zona horaria dependiendo el Estado. En la Figura 3.2.3.1 se muestran las tres zonas horarias:



Figura 3.1.3.1 Horarios de la República Mexicana.

El horario internacional para la República Mexicana es GMT-6:00 . Mexico City, Guadalajara, Monterrey New. Los husos horarios y Estados para cada zona son los siguientes:

- Tiempo del Noreste (Montaña):-1 hora . Baja California Sur, Chihuahua, Sinaloa y Sonora.
- Tiempo del Pacífico: -2 horas . Baja California
- Tiempo del centro: hora Actual . Estados Restantes

### 3.1.4 Archivo Hosts

En el archivo host se aclaran nombres o alias a los diferentes equipos pertenecientes al mismo segmento de red. Permite establecer comunicación entre los equipos usando el nombre o alias en lugar de la dirección IP.

La asignación de nombres para direcciones IP para cada servidor es distinta. Por ejemplo, la configuración del archivo Hosts para el Centro de verificación 00X es:

Servidor 1		Servidor 2		Servidor3	
# Almacenamiento		# Almacenamiento		# Almacenamiento	
192.168.xxx.xx	xd-cevem	192.168.xxx.xx	xd-cevem	192.168.xxx.xx	xd-cevem
# Digitalizacion		# Digitalización		# Digitalización	
10.41.xxx.xx	observer1	192.168.xxx.xx	observer1	10.41.xxx.xx	observer1
10.41.xxx.xx	observer2	192.168.xxx.xx	observer2	10.41.xxx.xx	observer2
# Procesamiento		# Procesamiento		# Procesamiento	
10.41.xxx.xx	zion	192.168.xxx.xx	zion-cevem	192.168.xxx.xx	zion
10.41.xxx.xx	cmml	192.168.xxx.xx	cmml-cevem	10.41.xxx.xx	cmml
10.41.xxx.xx	cmml2	192.168.xxx.xx	cmml-cevem	10.41.xxx.xx	cmml2
# Monitoreo				# Monitoreo	
10.41.xxx.xx	pc1			10.41.xxx.xx	pc1
10.41.xxx.xx	pc2			10.41.xxx.xx	pc2
10.41.xxx.xx	pc3			10.41.xxx.xx	pc3
10.41.xxx.xx	pc4			10.41.xxx.xx	pc4
# Energia				# Energia	
10.41.xxx.xx	ups1			10.41.xxx.xx	ups1
10.41.xxx.xx	ups2			10.41.xxx.xx	ups2
10.41.xxx.xx	ups3			10.41.xxx.xx	ups3

Figura 3.1.4.1 Archivo Host.

### 3.2 Servidor 1

Servidor que recibe la señal analógica y la convierte en una señal digitalizada para poder ser procesada.

### 3.2.1 Recursos compartidos

La información digitalizada se denomina media. Está compuesta por tres archivos: \*.dat, \*.fmt y \*.idx.

Se crean dos carpetas: Archive e índices.

- La carpeta Archive debe ser compartida. Se crea para el almacenamiento de media temporal (Servidor 1) y almacenamiento en disco duro (Servidor 2). Se construye con la palabra Archive y el número de CEVEM a tres dígitos. Por ejemplo, la carpeta compartida para el CEVEM 00X es: Archive00X.
- La carpeta índices no es compartida. Se crea para guardar los archivos tipo idx.

Ambas carpetas deben ser creadas en dentro de la ruta: X:\Video\.



Figura 3.2.1 Carpetas del Sistema

### 3.2.2 Configuración de encoders

Cada CEVEM tiene una configuración específica. Para este ejemplo, utilizaremos el CEVEM 001.

- Abrir el archivo tipo xls donde se encuentran las configuraciones de los encoders. El documento muestra la asignación de carpetas, canales de radio y TV, número de señales de radio y TV, frecuencia sintonizada y la cantidad de tarjetas utilizadas. El archivo tiene la siguiente estructura:
  - Servidor 1 (1)
  - 2 Encoders
    - A1 con cuatro audios asociados (R1, R2, R3, R4).
    - A2 con cinco audios asociados (R1, R2, R3, R4, R5).

ESTACION	OBSERVER 01		ENCODER
	TARJETA 01		
	CABLES DE VIDEO		
	IN 1A	V1	A1
	IN 1B	NC	
	IN 1C	NC	
	IN 1D	NC	

ESTACION	OBSERVER 01		ENCODER
	TARJETA 02		
	CABLES DE VIDEO		
	IN 2A	V2	A2
	IN 2B	NC	
	IN 2C	NC	
	IN 2D	NC	

ESTACION	CABLES DE AUDIO	PROFILE	ENCODER
6	LEFT 01	TV1	
1400	RIGHT 01	AM1	R1,R2
1200	LEFT 02	AM2	
790	RIGHT 02	AM3	R3,R4
	LEFT 03	NC	
	RIGHT 03	NC	
	LEFT 04	NC	
	RIGHT 04	NC	

ESTACION	CABLES DE AUDIO	PROFILE	ENCODER
13	LEFT 01	TV2	
	RIGHT 01	NC	
950	LEFT 02	AM4	R1
1050	RIGHT 02	AM5	R2,R3
	LEFT 03	NC	
	RIGHT 03	NC	R4,R5
93.7	LEFT 04	FM1	
	RIGHT 04	NC	R6

OBSERVACIONES	
TIPO	2
No. DE TARJETAS	1
encoders	2
AM	5
FM	0
TV	2

Figura 3.2.2.1 Distribución de encoders

- Servidor 1 (2)
- 4 Encoders
  - B1 con cinco audios asociados (R1, R2, R3, R4, R5).
  - B2 con cuatro audios asociados (R1, R2, R3, R4).
  - B3 con cuatro audios asociados (R1, R2, R3, R4).
  - B4 con siete audios asociados (R1, R2, R3, R4, R5, R6, R7).

ESTACION	OBSERVER 02 TARJETA 01	ENCODER	
	CABLES DE VIDEO		
	IN 1A V3	B1	
	IN 1B NC		
	IN 1C NC		
	IN 1D NC		
ESTACION	CABLES DE AUDIO	PROFILE	ENCODER
2	LEFT 01 TV3		
740	RIGHT 01 AM6	R1,R2	
860	LEFT 02 AM7		
1240	RIGHT 02 AM8	R3,R4	
1170	LEFT 03 AM9		
	RIGHT 03 NC	R5	B1
	LEFT 04 NC		
	RIGHT 04 NC		

ESTACION	OBSERVER 02 TARJETA 02	ENCODER	
	CABLES DE VIDEO		
	IN 2A V4	B2	
	IN 2B NC		
	IN 2C V5	B3	
	IN 2D NC		
ESTACION	CABLES DE AUDIO	PROFILE	ENCODER
11	LEFT 01 TV4		
1130	RIGHT 01 AM10	R1,R2	
660	LEFT 02 AM11		
1320	RIGHT 02 AM12	R3,R4	B2
4	LEFT 03 TV5		
97.3	RIGHT 03 FM2	R1,R2	
	LEFT 04 NC		
100.1	RIGHT 04 FM3	R3,R4	B3

ESTACION	OBSERVER 02 TARJETA 03	ENCODER	
	CABLES DE VIDEO		
	IN 3A V6	B4	
	IN 3B NC		
	IN 3C NC		
	IN 3D NC		
ESTACION	CABLES DE AUDIO	PROFILE	ENCODER
10	LEFT 01 TV6		
	RIGHT 01 NC	R1	
103.7	LEFT 02 FM4		
92.7	RIGHT 02 FM5	R2,R3	
101.7	LEFT 03 FM6		
105.3	RIGHT 03 FM7	R4,R5	
94.5	LEFT 04 FM8		
107.7	RIGHT 04 FM9	R6,R7	B4

OBSERVACIONES	
TIPO	4+2
No. DE TARJETAS	3
encoders	4
AM	7
FM	9
TV	4

- Total de frecuencias: 10 AM, 9 FM y 5 TV.

AM	12
FM	9
TV	6

Figura 3.2.2.2 Distribución de encoders

Para realizar cualquier configuración en el Servidor 1, se utiliza el Control Center.

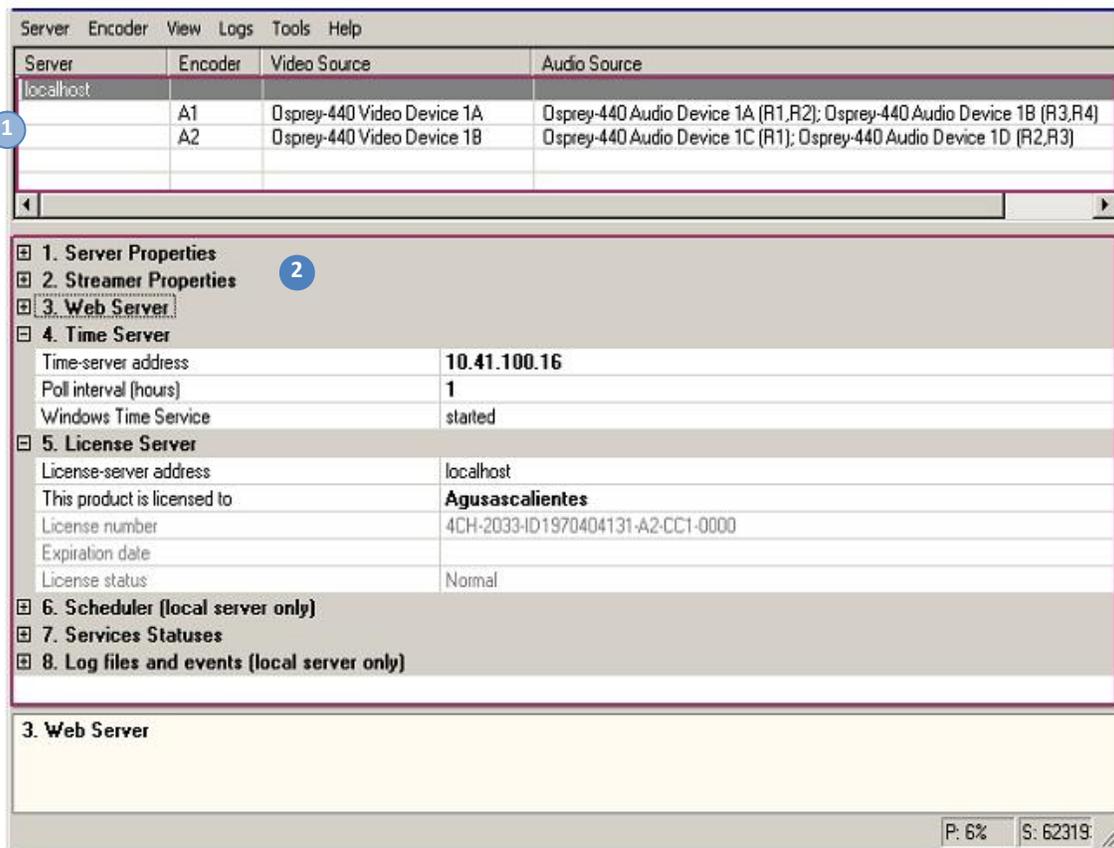


Figura 3.2.2.3 Pantalla configuración de encoders

Referencia	Nombre	Descripción
1	Encoders	En esta sección se muestran los elementos contenidos en el servidor. Enlista los <i>encoders</i> configurados. Muestra el estado en que se encuentran.
2	Parámetros	<p>Muestra las secciones de configuración para el servidor en general y los <i>encoders</i>.</p> <p>Secciones para el servidor:</p> <ul style="list-style-type: none"> <li>• Server Properties</li> <li>• Streamer Properties</li> <li>• Web Server</li> <li>• Time Server</li> <li>• License server</li> <li>• Scheduler (local server only)</li> <li>• Services Statuses</li> <li>• Log files and events (local server only)</li> </ul> <p>Secciones para los <i>encoders</i>:</p> <ul style="list-style-type: none"> <li>• Properties</li> <li>• Input Settings</li> </ul>

		<ul style="list-style-type: none"> <li>• Video Quality</li> <li>• Closed Caption</li> <li>• Storage</li> <li>• Notifications</li> <li>• Capture Sources</li> <li>• RPM Settings</li> <li>• Misc Settings</li> </ul>
--	--	---

Tabla 3.2.4 Secciones Control Center

### 3.2.3 Configuración de Front End

El Front End es la interfaz que permite visualizar las señales de audio o video sin estar logeado en el sistema de verificación y monitoreo, esta consulta de streaming se hace directamente desde el Servidor 1

If this is your first time using the Observer on this computer, you may be asked to ~~install software on this computer~~. Click "Yes" when asked to install software ~~on this computer~~.

~~Observer may not be installed unless the user has administration privileges on the machine where the plug-ins are being installed. If you are unable to proceed with the installation of Observer, you will need to contact your PC administrator.~~

Tabla 3.4.1 Pantalla Front End del sistema

Es necesario que se realice la configuración de canales para el fron end, ya que no es la misma configuración que se realizó en el Control Center.

### 3.2.4 Sección Mantenimiento

En esta sección se configuran los canales que son visibles para ese centro de verificación y monitoreo, se pueden agregar o quitar canales.

**Channel Settings**

Channel Name	System	Color	Hostname	Encoder	Edit	Delete
CANAL 6(+)	NTSC		10.41.100.21	A1		
CANAL 13	NTSC		10.41.100.21	A2		
CANAL 2	NTSC		10.41.100.22	B1		
CANAL 7	NTSC		10.41.100.22	B2		
CANAL 4	NTSC		10.41.100.22	B3		
CANAL 10	NTSC		10.41.100.22	B4		
CopiaB4	NTSC		10.41.100.22	B5		

Tabla 3.2.4.1 Pantalla Front End del sistema

### 3.3 Servidor 2

El servidor 2 es el servidor de almacenamiento. Lo puede hacer tanto en disco como en cintas para poder reemplazar las unidades y poder tener un historial disponible con un volumen considerable.

A continuación se describen brevemente los servicios, componentes y directivas de este servidor.

#### RAID 5

En informática, el acrónimo RAID (del inglés Redundant Array of Independent Disks, «conjunto redundante de discos independientes», anteriormente conocido como Redundant Array of Inexpensive Disks, «conjunto redundante de discos baratos») hace referencia a un sistema de almacenamiento que usa múltiples discos duros entre los que se distribuyen o replican los datos. Dependiendo de su configuración (a la que suele llamarse «nivel»), los beneficios de un RAID respecto a un único disco son uno o varios de los siguientes: mayor integridad, mayor tolerancia a fallos, mayor throughput (rendimiento) y mayor capacidad. En sus implementaciones originales, su ventaja clave era la habilidad de combinar varios dispositivos de bajo costo y tecnología más antigua en un conjunto que ofrecía mayor capacidad, fiabilidad, velocidad o una combinación de éstas que un solo dispositivo de última generación y costo más alto.

En el nivel más simple, un RAID combina varios discos duros en una sola unidad lógica. Así, en lugar de ver varios discos duros diferentes, el sistema operativo ve uno solo. Los RAID suelen usarse en servidores y normalmente (aunque no es necesario) se implementan con unidades de disco de la misma capacidad. Debido al decremento en el precio de los discos duros y la mayor disponibilidad de las opciones RAID incluidas en los chipsets de las placas base, los RAID se encuentran también como opción en las computadoras personales más avanzadas. Esto es especialmente frecuente en las computadoras dedicadas a tareas intensivas y que requiera asegurar la integridad de los

datos en caso de fallo del sistema. Esta característica no está obviamente disponible en los sistemas RAID por software, que suelen presentar por tanto el problema de reconstruir el conjunto de discos cuando el sistema es reiniciado tras un fallo para asegurar la integridad de los datos. Por el contrario, los sistemas basados en software son mucho más flexibles (permitiendo, por ejemplo, construir RAID de particiones en lugar de discos completos y agrupar en un mismo RAID discos conectados en varias controladoras) y los basados en hardware añaden un punto de fallo más al sistema (la controladora RAID).

Todas las implementaciones pueden soportar el uso de uno o más discos de reserva (hot spare), unidades preinstaladas que pueden usarse inmediatamente (y casi siempre automáticamente) tras el fallo de un disco del RAID. Esto reduce el tiempo del período de reparación al acortar el tiempo de reconstrucción del RAID.

### 3.3.1 Partición de disco

Al iniciar el sistema, aparece un arreglo predeterminado. Para poder hacer una partición con preferencias específicas, no podemos modificar el arreglo de fábrica. Es necesario eliminarlo y crear uno nuevo con las especificaciones que se deseen.

### 3.3.2 Servicios

El servicio Removable Storage debe estar levantado para el funcionamiento de la librería.

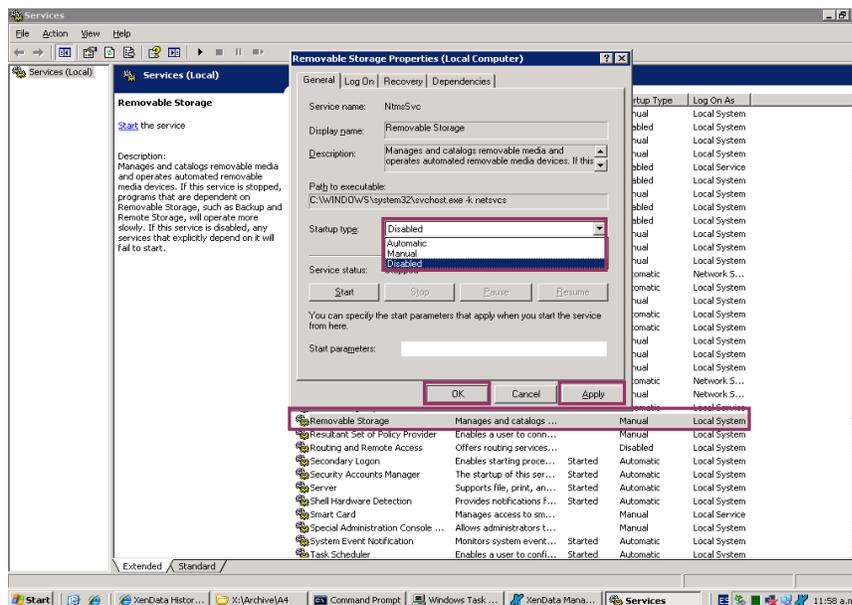


Imagen 3.3.2.1 Pantalla configuración del servicio

### 3.3.3 Archivos .DAT, .FMT y .IDX

Los archivos se guardan en disco y en cinta simultáneamente. El archivo quedará en disco durante 30 días, después será borrado y permanecerá sólo en cinta.

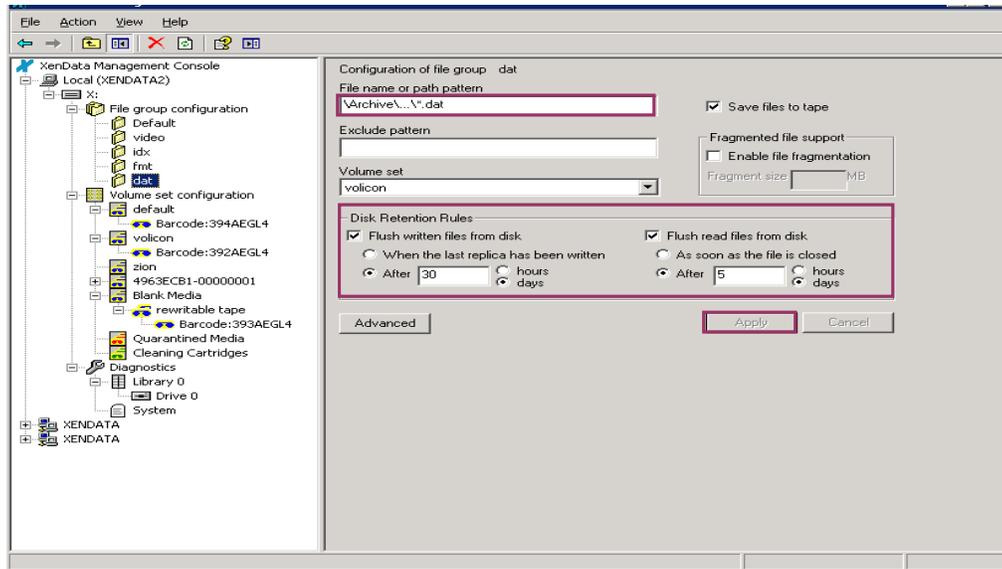


Imagen 3.3.3.1 Pantalla configuración del servidor

## 3.4 Servidor 3

### 3.4.1 Archivos de configuración

Son los archivos donde se dan de alta las sesiones (emisoras) que deben monitorearse, su localización en el Servidor 1 y donde se guardará la información recolectada, en él también se guardan determinados parámetros referentes a manejo de errores y consumo de streaming.

### 3.4.2 Servicios del Servidor 3

Se inician los servicios para levantar las sesiones y comenzar con el monitoreo de las señales. Los servicios verifican el archivo de configuración para determinar las sesiones y los parámetros de consumo de streaming.

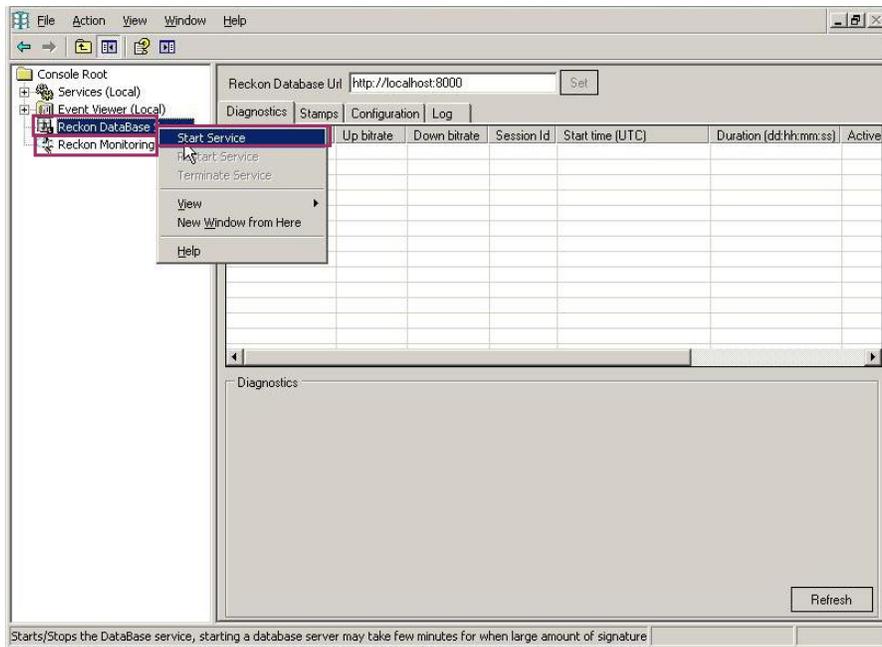


Imagen 3.4.2.1 Pantalla configuración del servidor

### 3.5 Servidor 4

El Servidor 4 es el servidor que integra y muestra toda la solución de la aplicación. Todos los servidores deben estar sincronizados con este equipo que a su vez, estará sincronizado con el Servidor 4 de CENACOM.

## Conclusiones

Entre las actividades que realicé durante el periodo que se aborda en el presente informe están:

- Estimación, planeación y seguimiento de tiempos, costos, recursos y tareas para la ejecución del proyecto.
- Coordinación de grupos de trabajo para el diseño y ejecución de casos de prueba.
- Ejecución y diseño de casos de prueba sobre componentes del sistema, web procesos batch (procesos sin pantalla).
- Identificación y aplicación de criterios para la ejecución de las pruebas funcionales y no funcionales según los requerimientos del área usuaria.
- Creación y ejecución de scripts para la automatización de pruebas de los componentes del sistema.
- Generación de scripts para automatización de pruebas de los procesos batch.
- Generación de scripts para pruebas de carga de procesos batch y ambiente web.
- Capacitación para la ejecución de pruebas a la fábrica de pruebas del cliente.
- Generación de documentación y panel de control para el seguimiento del proyecto, apegado a los estándares CMMi ML 5.
- Participante en el SCAMPI B y A para la obtención de la certificación CMMi ML3 y ML 5, con el rol de líder de proyecto.
- Implantación del plan de desarrollo de carrera para el área de testing.
- Seguimiento al plan de carrera de los testers para que obtengan capacitación y certificaciones.
- Generación de materiales e investigación de nuevas tecnologías para la mejora continua del área.
- Elaboración de propuestas técnicas para soluciones de testing en la empresa.
- Definición de la arquitectura de testing de los proyectos para pruebas de automatización y carga, cloud testing.
- Definición de herramientas para cada tipo de prueba.
- Capacitación a los nuevos integrantes del equipo sobre las reglas de negocio del sistema, así como los procedimientos y metodologías que involucra.
- Garantizar que el análisis y diseño de los sistemas cumplan con la metodología establecida.
- Ejecución de pruebas funcionales y no funcionales como: reglas de negocio, persistencia, formato, usabilidad, performance, estrés y automatización.
- Validar el desarrollo y liberación de los módulos de los sistemas.
- Generar la documentación soporte de las pruebas y el control de calidad de los sistemas. Generación de materiales multimedia (cápsulas y simulaciones), guías de usuario y manual de procedimientos para capacitación.
- Supervisión del proceso de ensamblado de equipo de computo, asesoría y capacitación al personal de la empresa.
- Soporte técnico a nivel software y hardware al personal de las diferentes direcciones que conforman, creación y seguimiento de los reportes solicitados por los usuarios, mantenimiento a equipo de computo e impresoras fuera de garantía.

Por medio de la práctica laboral tuve la oportunidad de enseñar y aprender de otras personas y de dar/recibir retroalimentación de lo aprendido, lo cual es importante porque el campo de la carrera de Ingeniería en Computación es multidisciplinario y con el avance tecnológico en su campo de acción, abarca todas las áreas del conocimiento e influye de manera directa en los diferentes sectores como son el económico, de planeación y servicios en el área científica y de investigación.

El conocimiento adquirido en el transcurso de la carrera de Ingeniería en Computación me permitió realizar las actividades que se explicaron en el presente informe. Durante la trayectoria profesional tengo la posibilidad de cumplir con el perfil del egresado de esta carrera que a la letra dice: %El egresado de la Licenciatura de Ingeniería en Computación cuenta con los conocimientos sólidos en software y hardware, y un manejo fluido de los principios teóricos y de los aspectos prácticos y metodológicos para la configuración, la evaluación, el diseño y el desarrollo de obras y entornos de sistemas computacionales complejos (servicios, telecomunicaciones, arquitecturas y configuración de redes de cómputo y teleproceso, etc.); que le permitan responder a diversas necesidades con soluciones innovadoras (proponiendo metodologías, técnicas y herramientas) en las diferentes áreas con las que interactúa. Puede perfeccionar o reafirmar su orientación y conocimientos mediante especializaciones o posgrado+.

## Glosario y acrónimos

**Almacenamiento:** Referente al servidor de almacenamiento que contiene una librería asociada para reciclar cintas y conservar la mayor cantidad de información posible.

**Bitrate:** tasa de codificación a la que debe ser muestreada la señal. Define la calidad de la señal.

**CENACOM:** Centro Nacional de Control y Monitoreo.

**CEVEM:** Centro de Verificación y Monitoreo.

**CMMI: Capability Maturity Model Integration,** Integración de modelos de madurez de capacidades

**Command Prompt:** Consola del sistema que permite abrir aplicaciones, revisar conectividad con equipos y administrar procesos directamente.

**Detección:** Coincidencia de una huella acústica y el material digitalizado.

**Digitalización:** Proceso de convertir una señal analógica recibida por la antena, en una señal digital para poder ser procesada, almacenada y comparada.

**DNS:** Domain Name Server. Permite la conexión de un servidor con un equipo sin la necesidad de utilizar una dirección IP.

**Encoder:** Señal codificada con parámetros específicos. Puede tener una señal de video asociada y uno o varios audios asociados.

**Filtro:** Criterio de búsqueda que reduce la cantidad de información a consultar.

**Firewall:** Bloquea el acceso de información no autorizada para el sistema como método de protección.

**Gateway:** Puerta de enlace que permite la interconexión de redes y equipos en cualquier protocolo y arquitectura a todos los niveles de comunicación.

**Hosts:** Nombres o alias asociados a las direcciones IP dentro de un segmento de red para identificarlos de manera eficiente.

**Integración:** Acceso a la información de los módulos del sistema para consulta de información requerida por el Instituto.

**Internet Explorer (IE):** Navegador de red que se utiliza para la visualizar las señales configuradas en el Servidor 1.

**Media:** Material digitalizado. Está conformado por 3 archivos: DAT, FMT e IDX.

**Medios:** Concesionarios y permisionarios de radio y televisión.

**Nagios:** Sistema de monitorización de redes. Permite vigilar equipos y servicios de manera remota. Contiene alertas configurables para informar los diferentes eventos ocurridos en una aplicación (SIVM).

**NIC:** Network Interface Card. Tarjeta de interfaz de red. Permite la conexión a un segmento de red o directamente a otro equipo.

**NTP:**Network Time Protocol. Protocolo de Internet que permite sincronizar equipos dentro de un mismo segmento de red.

**Pauta:** Política establecida por el instituto. Contiene horarios, tiempos y duraciones para garantizar la normatividad y cumplimiento de los materiales transmitidos.

**PP:** Partidos Políticos.

**RAID:**Redundant Array of Independent Disks. Arreglo redundante de almacenamiento. Permite mayor integridad, mayor tolerancia, mayor tasa de envío de información y respaldo en caso de fallo de un disco.

**RDP:** Remote Desktop Protocol. Permite la comunicación y control de un equipo a través de un servidor de manera remota dentro de un mismo segmento de red.

**SCAMPI: Standard CMMI Appraisal Method for Process Improvement.** Método de evaluación estándar para la mejora de procesos.

**Sitio:** Lugar donde se instala un CEVEM. Puede ser en una Junta Local o una Junta Distrital.

**Spot:** Anuncio de radio o televisión de los PP.

**Streaming:** Información en tiempo real (en vivo).

**Testigo:** Archivo que contiene información de un spot que ha sido detectado y luego cruzado contra las pautas. Tiene formato \*.asf.

**URL:** Uniform Resource Locator . Dirección de acceso de una aplicación.

**Versión:** Nombre del material transmitido en los medios.

## Bibliografía y referencias

- Cem Kaner, *Scenario-Driven Testing*.+ [SLIDES] (2010) Tutorial at the Conference of the Association for Software Testing. Disponible en: <http://kaner.com/pdfs/CAST2010ScenarioTestingTutorial.pdf> (Consultado el 20 Octubre del 2012)
- Cem Kaner, J. D., Ph. D. *Context driven Testing* Disponible en: <http://context-driven-testing.com> (Consultado el 2 de Noviembre del 2012)
- *Clarus Concept of Operations* (2005) Disponible en: from Federal Highway Administration (FHWA), (en ingles). Publicación No. FHWA-JPO-05-072. ( Consultada el 7 de Noviembre del 2012)
- *Contents pc RAM* (2012) Disponible en: [es.kioskea.net/contents/pc/ram.php3](http://es.kioskea.net/contents/pc/ram.php3) (Consultado el 18 de Noviembre del 2012)
- Forsberg, K., Mooz, H., Cotterman, H. (2005) *Visualizing Project Management* (en ingles), 3ª ed, Nueva York, NY. Editorial John Wiley and Sons.
- *Informática*. Disponible en: <http://es.wikipedia.org> (Consultado el 6 de Noviembre del 2012)
- Lemus. Guillermo, *Tipos de pruebas de software (2012)* Disponible en <http://www.docseek.net/krppti/tipos-de-pruebas-de-software.html> (Consultado el 12 de Noviembre del 2012)
- L, J. Jorge, (2006-20012) *Memorias: La memoria Ram*. Disponible en: <http://www.ladelec.com/teoria/electronica-digital/210-memorias-la-memoria-ram> (Consultado el 12 de Noviembre del 2012)
- *Modelos de desarrollo de software* (2010) Disponible en: <http://ddonofrio.blogspot>. (Consultado el 8 de Noviembre del 2012)
- *Perfil del Egresado*.(2009) Ingeniería en Computación. Disponible en: [http://www.aragon.unam.mx/oferta\\_educativa/licenciaturas/ing\\_computacion/ing\\_computacion.html](http://www.aragon.unam.mx/oferta_educativa/licenciaturas/ing_computacion/ing_computacion.html) (Consultado el 21 de Noviembre del 2012)
- *Procesadores Intel Celeron*. Disponible en: [www.intel.la/content/www/xl/es/processors/.../celeron-processor.html](http://www.intel.la/content/www/xl/es/processors/.../celeron-processor.html) (Consultado el 12 de Noviembre del 2012)
- *Procesador Intel Core 7*. Disponible en: <http://www.intel.la/content/www/xl/es/processors/core/core-i7-processor.html> (Consultado el 12 de Noviembre del 2012)
- *Pruebas de calidad de software* Disponible en: [www.ecured.cu/index.php/](http://www.ecured.cu/index.php/) (Consultado el 6 de Noviembre del 2012)

- *Pruebas de software*. Capacitación y guía para el desarrollo de software. IT MENTOR Disponible en: <http://materias.fi.uba.ar/7548/Pruebas-Intro.pdf> (Consultado el 12 de Noviembre del 2012)
- *¿Qué es la memoria Ram? Definición de memoria Ram. (1999-2013)* Disponible en: <http://www.masadelante.com/faqs/memoria-ram> (Consultado el 12 de Noviembre del 2012)
- *Tipos de memoria RAM*. Disponible en: [http://www.informaticamoderna.com/Memoria\\_RAM.htm](http://www.informaticamoderna.com/Memoria_RAM.htm) (Consultado el 24 de Noviembre del 2012)
- *Tipos de memoria RAM*. Disponible en: [www.areatecnologia.com/](http://www.areatecnologia.com/) (Consultado el 2 de Diciembre del 2012)
- Tuya, Javier. (2009) *Pruebas de Software: ISO/IEC 29119*. Alcalá de Henares. Disponible en: : <http://www.ati.es/IMG/pdf/Tuya.pdf> (Consultado el 3 de Noviembre del 2012)