



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

SISTEMA DE CONTROL DE PRUEBAS DE ENFERMEDADES DE
TRANSMISIÓN SEXUAL (VIH, VPH, SIFILIS)

T E S I S

QUE PARA OBTENER EL TÍTULO DE

INGENIEROS EN COMPUTACIÓN

PRESENTAN

APARICIO COLÍN ETHEL GIORDANNA
ARGUELLO NANCO HÉCTOR AURELIO
MOHEDANO VALDEZ GABRIEL OLIVER
HERNÁNDEZ SÁNCHEZ GERARDO

ASESOR:

M.I. JUAN CARLOS ROA BEIZA



Ciudad Universitaria, CD. MX., 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

ETHEL GIORDANNA APARICIO COLÍN

A mi familia que me apoyó desde el inicio cuando escogí esta carrera, creyendo en mi, alentándome siempre a seguir adelante.

A mi madre Elva Colín por darme las herramientas tanto psicológicas como financieras a lo largo de mis estudios, aún cuando no tuvieras las posibilidades, siempre exigiendo lo mejor de mi y sobre todo dándome tu amor y comprensión, porque nunca me recriminaste el muchas veces no estar presente porque ... tengo tarea, tengo que estudiar, tengo examen, tengo proyecto, etc., gracias te quiero mucho.

A mi padre Eduardo Aparicio por tu cariño, tus advertencias, tu ayuda, aunque no pudieras estar conmigo.

A mis hermanos Edgar, Erika, Eduardo y Jesús por sus palabras de inspiración, sus bromas, sus cuidados, sus sacrificios a todos gracias por quererme y aceptarme como soy.

A mis sobrinos Ale, Leo, Diego y Ximena, que me sacan canas pero que son los mas cariñosos del mundo y me hacen sonreír con su inocencia y amor.

A mi cuñado Omar por el apoyo que me has dado a mi y a mi familia, gracias por ir por mi a la Prepa cuando ya no había combis.

A mi novio José Luis Tapia por soportarme, apoyarme, comprenderme, por darme siempre la mejor de tus sonrisas y siempre estar para mi a pesar de todo, por todo tu amor y tu ternura que han sido mi motor desde que te conocí.

A mis compañeros y amigos que están, que estuvieron, gracias por compartir las risas, los enojos, el estrés, la casa para hacer el proyecto, la recomendación de profesores, los chistes locales que siguen formando parte de nuestra vida, las anécdotas que ahora tenemos para contar, todo eso que contribuyó para hacer mas livianas las desveladas y mas maravillosa esta experiencia.

Al PAT y al M.I. Juan Carlos Roa Beiza por darnos la oportunidad de cerrar este ciclo con su orientación y apoyo.

AGRADECIMIENTOS

A la UNAM por darme la oportunidad y el privilegio de formar parte de su comunidad desde la secundaria, proporcionándome los conocimientos empíricos y científicos no sólo para obtener el título que ostento sino con ello poder retribuir a la sociedad con mi trabajo y si se diera el caso inclusive transmitirlo a un legado. Gracias también por darme las mejores experiencias y amigos.

A la Facultad de Ingeniería y maestros por enseñarme que sacar dieces ya no era tan fácil como antes y que eso y mas requiere de un esfuerzo constante, de sacrificios, de autodidactismo donde al final uno mismo es el responsable del éxito o fracaso, el único beneficiado o afectado por lo cual la recompensa será aún mayor, gracias por su asesoría, sus conocimientos y experiencias.

HÉCTOR AURELIO ARGUELLO NANCO

A mis padres

Por estar siempre a mi lado, por su amor, comprensión y apoyo que gracias a ello estoy terminando este ciclo de mi vida. Agradezco a mi madre María Félix Nanco por su inmenso amor hasta el último momento de su vida. Agradezco a mi padre Faustino Arguello quien luchó por mi educación y hasta el momento me ha dado todo el cariño y apoyo.

A mis hermanos

A cada uno de mis 9 hermanos de quienes he recibido un gran impulso para salir adelante, gracias por sus palabras de aliento, por sus regaños, por todo lo que han aportado para que pudiera salir adelante y terminar mis estudios.

A mis amigos

Por la ayuda recibida durante la carrera, por todas esas horas compartidas para estudiar, para convivir y por todo ese apoyo recibido en los momentos difíciles de la vida y de la carrera. Agradezco especialmente a Andrés Sales por su ayuda incondicional.

A todos mis Familiares

En especial a mi tío Felipe Nanco quien fue un gran apoyo durante la carrera, agradezco mucho los consejos y enseñanzas aportadas.

A la Universidad Nacional Autónoma de México

Por darme la oportunidad de formarme como profesional en sus aulas, a todos los maestros de los que recibí grandes conocimientos y que me ayudaron a prepararme para enfrentar la vida profesional.

Al M.I. Juan Carlos Roa Beiza y al PAT

Por darme esa confianza en hacer el trabajo de Tesis que estamos presentando, por brindarnos la oportunidad y la guía correcta a mí y a muchos compañeros que hemos salido de la carrera para terminar nuestros estudios profesionales.

A todas y cada una de las personas que me brindaron su ayuda para que esta etapa de vida pueda concluir.

A todos y cada uno de ellos.

¡Muchas Gracias!

GABRIEL OLIVER MOHEDANO VALDEZ

A mis padres que me han apoyado en todo momento, por darme su amor, sus consejos, su compañía y levantarme en los momentos de tristeza y frustración. A mi mamá Raquel Valdez por siempre creer en mi y enseñarme el valor del trabajo y la dedicación. A mi papá Gabriel Mohedano quien me demostró todo su apoyo a lo largo de su vida y quien es y será mi ejemplo a seguir. Éste trabajo es el fruto de sus esfuerzos y sacrificios .

A mi hermano Christian Mohedano, mi mejor amigo, por que siempre me has cuidado, sé que siempre podré contar con tu apoyo, tus consejos y tu amor.

A mi amada Sandra Beltrán quien me ha motivado a cerrar éste ciclo y con quien he compartido muchas alegrías. Me has ayudado a desarrollarme como persona y siempre me demuestras tu amor y comprensión.

A Román Mohedano cuya ternura y amor incondicional no se compara con nada. Los paseos a tu lado me han servido a despejar mi mente y no desistir, agradezco lo mucho que has hecho por mi familia.

A Clory Mohedano y Cesar Novelo mis ángeles, gracias a ustedes he podido consumir éste título, siempre los llevaré en mi corazón, éste título va dedicado a ustedes ¡Los amo!.

A mi familia que siempre ha demostrado ser muy unida en las buenas y en las malas, me siento muy orgulloso y afortunado de ser parte de ustedes.

A mi Universida Nacional Autónoma de México, Facultad de Ingeniería en donde conocí a mi segunda familia y en donde viví una de las etapas mas bonitas de mi vida. Gracias por darme la formación académica para poder enfrentar las cicunstancias del ámbito laboral permitime ser parte de su día a día.

Al PAT y al M.I. Juan Carlos Roa Beiza por su guía en la elaboración de esta tesis y a mis compañeros Ethel Aparicio, Héctor Argüello y Gerardo Hernández ha sido un honor trabajar nuevamente a su lado.

MUCHAS GRACIAS.

AGRADECIMIENTOS

GERARDO HERNANDEZ SANCHEZ

A mi madre María Teresa Hernández, que me brindó el gran regalo de la vida y de la educación, y que me apoyó para llegar hasta este punto, con su cariño y buenos consejos siempre supo como sacarme adelante.

A mi tío Nemorio García que siempre ha estado a mi lado, me ha apoyado y me ha dado consejos y enseñanzas aún hasta el día de hoy, y ha ayudado a poderme brindar una educación de calidad desde el inicio de mi vida académica.

A mi esposa Alicia Paula García y a mis hijas Ángela y Ximena Hernández, que me han dado el aliento para terminar este ciclo, con su apoyo, su amor y su comprensión, han sabido guiarme a través de este proceso.

A mi familia que me ha brindado su apoyo incondicional a lo largo de mi vida, y que al día de hoy continúan alentándome a ser cada día mejor

A la Universidad Nacional Autónoma de México y a la Facultad de Ingeniería, por abrirme las puertas para poder realizar mis estudios, y más que eso, por darme la oportunidad de vivir una de las experiencias más maravillosas de la vida, que es, ser UNIVERSITARIO.

Al PAT y al M.I. Juan Carlos Roa Beiza por su apoyo en la generación de este trabajo de titulación.

A mis compañeros y amigos, Ethel Aparicio, Héctor Argüello y Gabriel Moherano, los cuales siempre estarán en mi mente por su gran dedicación en este logro.

¡MUCHAS GRACIAS!

ÍNDICE

CAPÍTULO 1: ANTECEDENTES.....	1
1.1. Objetivo	1
1.2. Introducción.....	1
1.3. Conceptos generales de las ETS	6
1.3.1. Definiciones	6
1.3.2. Métodos de diagnóstico.....	10
1.4. Historia de las ETS.....	12
1.4.1. Epidemiología de las ETS en el siglo XX.....	12
1.4.2. El descubrimiento del VIH	12
1.4.3. El origen de la epidemia.....	13
1.4.4. Impacto de las ETS en México	14
1.5. Acotamiento del problema a resolver	15
1.5.1. Historial clínico.....	16
1.5.2. Nota de evolución	16
1.5.3. Nota de Referencia/Traslado.....	17
Referencias Capítulo 1.....	18
CAPÍTULO 2.- MARCO TEÓRICO	20
2.1 Características, ventajas y desventajas de las bases de datos relacionales y la metodología UWE.....	20
2.1.1 Bases de datos relacionales	20
2.1.2 Metodología UWE.....	23
2.2 Características, ventajas y desventajas del lenguaje de programación PHP y el servidor WEB Apache Tomcat.....	28
2.2.1 PHP.....	28
2.2.2 Apache.....	31
2.3 Características, ventajas y desventajas del Sistema Gestor de Base de Datos MySQL.....	34
2.3.1 Introducción de MySQL	34
2.3.2 Características de MySQL	36
2.3.3 Ventajas de MySQL	37
2.3.4 Desventajas de MySQL	41
2.4 Características, ventajas y desventajas del ambiente Cliente - Servidor.....	43
Referencias Capítulo 2.....	54

CAPÍTULO 3: PLANTEAMIENTO DEL PROBLEMA.....	57
3.1. Análisis del problema	57
3.2. Requerimientos generales y particulares de la aplicación.	64
3.3. Análisis y posicionamiento de los lenguajes para el <i>back-end</i> y <i>front-end</i>	69
3.4. Diseño a nivel de bloques de la interfaz de usuario.	84
Referencias Capítulo 3.....	87
CAPÍTULO 4: ANÁLISIS, DISEÑO Y CONSTRUCCIÓN.....	89
4.1 Aplicación y separación de los módulos según la metodología UWE.....	89
4.2. Diseño de modelos	93
4.2.1. Diagrama de contenido	93
4.2.2. Diagrama de Navegación	95
4.2.3. Diagrama de presentación.....	97
4.2.4. Diagrama de estructura de procesos	101
4.2.5. Diagrama de procedimientos.....	104
4.2.6. Diagrama de casos de uso	107
4.2.7 Diagrama entidad relación.....	111
4.2.8. Normalización de la base de datos	114
4.3. Desarrollo e implementación de la base de datos	122
4.3.2. Administración de la base de datos	122
4.3.2.1. MySQL Administrator	122
4.3.2.2. MySQL Query Browser.....	125
4.4. Desarrollo e implementación del fron-end	129
4.4.1. Módulo de seguridad	129
4.4.2. Módulo de administración de usuarios.....	131
4.4.3. Módulo de atención de pacientes	133
4.4.4. Módulo de administración de pruebas entre jurisdicciones	135
4.4.5. Módulo de reportes	137
4.5. Pruebas y Reportes.....	138
4.5.1. Pruebas	138
4.5.1.1. Pruebas funcionales	139
4.5.1.2. Pruebas no funcionales.....	141
4.5.1.2. Pruebas estructurales.....	144
4.5.1.3. Pruebas asociadas a cambios: Repetición de pruebas y pruebas de regresión.	144

4.5.2. Reportes.....	145
4.5.2.1. Todos los pacientes	145
4.5.2.2 Por jurisdicción	148
4.5.2.3 Estadística por edad	148
4.5.2.4 Estadística por género	152
4.5.2.6 Reporte general del paciente	155
CONCLUSIONES	158
BIBLIOGRAFÍA.....	160
ANEXOS	166
1. Instalación del SGBD	166
2. Gráficas de Reportes	176
3. Diccionario de datos	177

CAPÍTULO 1: ANTECEDENTES

1.1. Objetivo

Analizar, desarrollar e implementar un sistema *web* para el control de pruebas de enfermedades de transmisión sexual, enfocado en pruebas de VIH, VPH y Sífilis, en donde se podrá registrar el expediente médico de los pacientes, además del control del seguimiento de las pruebas realizadas a cada uno de ellos, con la finalidad de poder obtener reportes y estadísticas así como también tener registro del historial de los pacientes.

1.2. Introducción

Las Enfermedades de Transmisión Sexual (ETS) son: “enfermedades venéreas (contagiosas) que reciben su nombre por la diosa romana Venus, que representaba el amor, la belleza y la fertilidad”; además del principal y más común medio de transmisión que tienen estas enfermedades que es por contacto sexual de cualquier índole. En este trabajo de tesis estaremos hablando principalmente de 3 enfermedades Virus de Inmunodeficiencia Adquirida (VIH), Virus del Papiloma Humano (VPH) y Sífilis. Otras formas de contagio son:

- Jeringas contaminadas.
- Contacto directo con la sangre o fluidos de una persona con ETS.
- De la madre a un hijo en periodo de gestación.

Estas enfermedades son causadas en su mayoría por gérmenes como bacterias, parásitos, protozoos, hongos y virus.

Existen diversos factores socioeconómicos y culturales que pueden influir para el desarrollo de estas enfermedades en una urbe; sin embargo afecta más a los grupos de bajos estratos sociales y grupos jóvenes.

CAPÍTULO 1: ANTECEDENTES

¿Por qué se ven afectados estos grupos principalmente? esto se relaciona con falta de higiene, comercio sexual, falta de información, acceso limitado a servicios de salud, aumento de la población, por estar en etapa reproductiva, abuso de drogas y alcohol, debido a la revolución sexual se desarrollaron comportamientos tales como: iniciar actividad sexual a edades más tempranas, tener más de una pareja sexual, casarse más tardíamente o no hacerlo, uso escaso o nulo de preservativo, etc.

Según cifras de la Organización Mundial de la Salud (OMS): “900 mil personas se infectan a diario, 340 millones de casos nuevos de ETS son registrados por año en el mundo, los casos de ETS se presentan en mayor proporción entre personas de 15 a 49 años, semejante para ambos géneros, con un ligero predominio entre los hombres”.

En lo que respecta a la República Mexicana en el período 1986-2004 el comportamiento que presentan algunas ETS son: “la disminución de las ETS clásicas (sífilis adquirida de 6.3 a 1.4%, sífilis congénita de 0.2 a 0.1%; linfogranuloma venéreo de 0.5 a 0.2; chancro blando de 1.2 a 0.6 y Gonorrea de 19.1 a 12), tricomoniasis urogenital de 31.6 a 95.3 y un incremento de las nuevas ETS, herpes genital de 1.1 a 2.7, exceptuando la infección por hepatitis B que disminuyó de 0.6 a 0.5 tasa por 100,000 habitantes”.

Algunos ejemplos de los síntomas físicos que se pueden llegar a presentar están: brotes de abscesos o protuberancias anómalas, dolor abdominal, lesiones en la piel, olor desagradable, picazón, dolor o ardor al orinar, fiebre, síntomas de gripe; cabe mencionar que estos síntomas no son exclusivos de este tipo de enfermedades.

Una vez contraído una ETS por virus no hay forma de que esta desaparezca puede controlarse con medicamento, más sin embargo siempre estará presente en el organismo. En algunos casos no se desarrolla la enfermedad y sólo se es portador, sin embargo, el hecho de tener una ETS no te exime de contraer algún otro tipo o variedad de esta.

No siempre se presentan síntomas físicos, o se desconocen los síntomas, por ello es recomendable acudir a revisiones periódicas; muchas empresas someten anualmente a sus empleados a diferentes estudios dentro de los que se encuentra la prueba del VIH; lo que deriva además de ser un problema de salud pública, en consecuencias de rechazo social y laboral.

Para evitar el contagio se recomienda el uso de preservativos, tener una sola pareja sexual, no compartir jeringas, mantener una buena salud sexual, informarse acerca de estos padecimientos; en el caso del VPH existen vacunas para prevenir determinadas clases de este virus, y en caso de sospechar el haber contraído una ETS, debe realizarse una prueba y abstenerse del contacto sexual mientras se encuentre en tratamiento.

La detección a tiempo es un factor primordial en el tratamiento de estas enfermedades evitando así que otras personas se encuentren bajo la misma condición, o que haya una complicación por no tratarse oportunamente y los métodos para la cura tiendan a ser más agresivos y para extender la calidad de vida en caso que la enfermedad no tenga cura.

Algunos indicadores para saber si se está contagiado de alguna ETS pueden ser:

- Reconocimiento de síntomas.
- Auscultación.
- Estudios de laboratorio.

El auge de estas enfermedades representa una mayor demanda de un servicio médico que otorgue un diagnóstico y si es el caso tratamiento para el paciente siendo capaz de llevar un control adecuado en el manejo de pruebas de ésta naturaleza.

La toma de muestras para laboratorio junto con la exploración es sin duda la forma más certera de saber se ha contraído una ETS, lamentablemente muchas veces estos

servicios no se llevan a cabo de la manera más adecuada ya sea en sector público o privado, esto porque existen deficiencias tales como:

- El control de los pacientes que solicitan los servicios se registran en hojas de cálculo.
- La historia clínica se registra en libros que no siempre son almacenados para un seguimiento.
- Fuga y pérdida de información.
- Confusión de resultados.
- Extravío de expedientes.

Se propone entonces un sistema web para llevar el control de pacientes de una manera más eficiente; enfocado al sector privado.

El sistema contará con un front-end que le permitirá al usuario registrar la historia clínica del paciente y poder almacenarla en una base de datos para un futuro seguimiento del mismo y así poder consultar los resultados de las pruebas de laboratorio.

Una vez obtenido el diagnóstico se le notificará al paciente que puede recoger los resultados, dicho documento contará con la validez para el organismo que así lo requiera; así mismo se le indicarán los cuidados recomendados y se canalizará para atención médica o a una ONG según sea el caso.

Con los datos recabados se obtendrán estadísticas para controles internos y de la evolución del paciente.

Las herramientas con las que se realizará el sistema será software libre en su totalidad, para no consumir demasiados recursos y evitar dependencia de licencia de terceros pero capaces para cumplir con las necesidades que exige el requerimiento.

La tesis estará conformada por cuatro capítulos:

En el primer capítulo abordaremos el panorama actual de las ETS: ¿Qué personas son candidatas potenciales para el desarrollo de estas enfermedades?, medidas de prevención, tipos de pruebas para las principales ETS, conceptos relacionados a las principales ETS. Nos enfocaremos en explicar de manera general 3 de las principales ETS.

En el segundo capítulo se explicará el porqué de las herramientas/lenguajes: MySQL, PHP y Apache y el porqué de la metodología a usar UML-based Web Engineering (UWE) - Ingeniería Web Basada en UML, además de sus principales características concernientes a cada una, las ventajas y desventajas que representan

En el tercer capítulo se analizará el problema que hemos planteado, completando así el requerimiento, una vez hecho esto se expone la solución técnica que proponemos para el back-end y front-end.

En el cuarto capítulo mostramos el diseño para cada componente del sistema de acuerdo a la metodología adoptada y la construcción de estos diseños; el resultado de este conjunto de actividades nos lleva a la implementación del sistema viéndose reflejado en pruebas de usuario y reportes que se generarán a partir de las entradas propuestas en dichas pruebas.

Al final reportaremos nuestras conclusiones de lo que hemos construido empezando por el cumplimiento de los objetivos, el nivel de eficiencia y control de información que se logró obtener con la automatización de los procedimientos que se llevan a cabo para la atención y pruebas de los pacientes con ETS mediante un sistema específico.

1.3. Conceptos generales de las ETS

1.3.1. Definiciones

- Virus de la inmunodeficiencia humana (VIH): “Es un lentivirus (de la familia Retroviridae), causante del síndrome de inmunodeficiencia adquirida (sida). Fue descubierto y considerado como el agente de la naciente epidemia de sida por el equipo de Luc Montagnier en Francia en 1983. El virión es esférico, dotado de una envoltura y con una cápside proteica. Su genoma es una cadena de ARN monocatenario que debe copiarse provisionalmente al ADN para poder multiplicarse e integrarse en el genoma de la célula que infecta. Los antígenos proteicos de la envoltura exterior se acoplan de forma específica con proteínas de la membrana de las células infectables, especialmente de los linfocitos T CD4”.
- Virus del Papiloma Humano (VPH o HPV): “Son grupos diversos de virus ADN pertenecientes a la familia de los Papillomaviridae y representa una de las enfermedades de transmisión sexual más comunes, y se conocen más de 100 tipos virales que en relación a su patogenia oncológica, se clasifican en tipos de alto y de bajo riesgo oncológico”.
- Sífilis: “es una infección de transmisión sexual crónica producida por la bacteria espiroqueta *Treponema pallidum*, subespecie *pallidum* (pronunciado pá lidum)”. Se le llama “la gran imitadora” porque tiene muchísimos síntomas posibles y muchos de estos se parecen a los síntomas de otras enfermedades.
- Virus: (del latín virus, «toxina» o «veneno») es un agente infeccioso microscópico acelular que solo puede multiplicarse dentro de las células de otros

organismos. Los virus infectan todos los tipos de organismos, desde animales y plantas, hasta bacterias y arqueas

- **Bacteria:** Microorganismos unicelulares que presentan un tamaño de algunos micrómetros de largo (entre 0,5 y 5 μm , por lo general) y diversas formas incluyendo esferas, barras y hélices. Las bacterias son procariotas y, por lo tanto, a diferencia de las células eucariotas de los animales y las plantas no tienen núcleo ni orgánulos internos.
- **Treponema pallidum:** es una especie del género *Treponema*, compuesta por entre ocho y veinte espiras enrolladas, lo que le da un movimiento de rotación similar a un sacacorchos. Mide de 5 a 20 micras de largo y 0,5 de diámetro, y es una espiroqueta altamente contagiosa. Es causante de varias enfermedades del ser humano, principalmente la sífilis. Su estructura básica consiste en un filamento axial incluido en un cilindro helicoidal de citoplasma. El filamento es morfológicamente similar al flagelo bacteriano, y le otorga movilidad.
- **Retroviridae** es una familia de virus que comprende los retrovirus. Son virus con envoltura que presentan un genoma de ARN monocatenario de polaridad + y se replican de manera inusual a través de una forma intermedia de ADN bicatenario.
- **Síndrome de la Inmunodeficiencia Adquirida (SIDA):** Es el estado de la infección por el VIH caracterizado por bajos niveles de defensas y la aparición de infecciones oportunistas.
- **Sistema Inmunitario:** Es el conjunto de tejidos, células y moléculas responsables de la inmunidad (la protección del cuerpo frente a agentes extraños y perjudiciales), y su respuesta colectiva y coordinada frente a la entrada en el organismo de ciertas sustancias extrañas se denomina respuesta inmunitaria. Para poder llevar a cabo esta respuesta inmunitaria nuestro cuerpo

cuenta con las células del sistema inmunitario, entre las que destacan los linfocitos o glóbulos blancos.

- Seropositivo: Dentro del cuadro clínico del VIH se considera que una persona es seropositiva cuando está infectada por el VIH. La seropositividad se determina mediante la realización de un análisis de sangre que demuestra la presencia de anticuerpos específicos contra el VIH en un organismo infectado.
- Infecciones oportunistas: Son infecciones producidas por otros agentes que aparecen cuando las defensas inmunitarias de una persona infectada por el VIH son insuficientes. Se trata de infecciones que un sistema inmunitario sano podría combatir pero contra las cuales un organismo infectado es incapaz de protegerse.
- Linfocitos CD4: Son un tipo de células que forman parte del SI y que se encargan de la fabricación de anticuerpos para combatir las infecciones. Son la diana preferente del VIH. Al destruirlos el VIH acaba con la capacidad defensiva del cuerpo.
- Lentivirus: Son virus cuyo periodo de incubación es muy largo. Su nombre contiene el prefijo latino lenti-, aludiendo a la demora con que aparecen o la lentitud con que se desarrollan los signos de las infecciones que producen.
- Parásitos: “Organismo que vive y se alimenta de otro ser vivo dentro de él o sobre su piel, sin aportar beneficios a su huésped”
- Hongos: “Son organismos eucarióticos, heterótrofos, unicelulares o pluricelulares con estructura de talo, Poseen pared celular, parecida a la de los vegetales, pero no tienen celulosa”

Existen alrededor de 30 ETS dentro de las que se destacan:

- Virus de Inmunodeficiencia Humana (VIH) –Origen Virus.

- Hepatitis B-Origen Virus.
- Ladillas- Origen Parásito.

Secretoras

- Gonorrea- Origen Bacteria Neisseria gonorrhoeae.
- Clamidia- Origen Bacteria Chlamydia trachomatis.
- Enfermedad Pélvica Inflamatoria (EPI)- Origen Ascenso de Bacterias del tracto genital.
- Tricomoniasis-Origen Parásito Trichomonas vaginalis.
- Vaginosis Bacteriana – Origen Bacterias G. Vaginalis y Mycoplasma hominis, Mobilunco sp. y anaerobios, etc.
- Candidosis Vaginal- Origen Hongo Cándida albicans.

Ulcerativas

- Sífilis-Origen Bacteria Treponema pallidum.
- Virus de Herpes Simple (VHS)-Origen Virus herpes simplex tipo 1 y 2.
- Chancroide-Origen Bacteria Haemophilus ducreyi.
- Linfogranuloma Venéreo (LGV)- Origen Bacteria Chlamydia trachomatis tipo L1, L2, L3.
- Granuloma inguinal- Origen Bacteria Calymmatobacterium granulomatis.

Tumorales

- Condilomas Acuminados (CA)- Origen Virus de Papiloma Humano.

1.3.2. Métodos de diagnóstico

Para realizar un diagnóstico de una ETS puede realizarse mediante los siguientes tres métodos:

- Manejo etiológico: Se requiere del apoyo de un laboratorio debido a que los síntomas entre algunas ETS son similares, se utiliza para confirmar agentes etiológicos cuando el manejo sindromático no resulte exitoso y permite un diagnóstico preciso del problema.
- Manejo Clínico: Se basa en la experiencia clínica del tratante con ayuda de una historia clínica y un examen físico, éste diagnóstico no permite identificar si se tiene más de un tipo de ETS simultáneamente.
- Manejo Sindromático: El examinador trata todos los síntomas identificados por el paciente y por la exploración física.

El alcance de la tesis se encuentra enfocado únicamente a tres ETS: VIH por ser la principal ETS causante de muerte en países en desarrollo, VPH debido a que es de los más comunes y es una causa principal de cáncer cervicouterino y muchas veces silencioso, y Sífilis por contar con los índices más altos sobre todo en hombres de preferencias homosexuales. La **Figura 1.3.1** muestra el tipo de prueba que suele realizarse para la detección de las ETS antes mencionadas.

ETS	PRUEBA
VIH	<p>Análisis de sangre.</p> <p>Prueba bucal con hisopo; se usa un instrumento especial para analizar las células del interior de la boca.</p> <p>Análisis de orina (poco frecuente).</p>
VPH	<p>No hay una prueba de virus del papiloma humano para los hombres.</p> <p>Para la mujer existen análisis de muestras de células del cuello del útero; posiblemente se use un instrumento especial llamado coloscopio para detectar verrugas que son demasiado pequeñas que no puedan ser detectadas a simple vista.</p>
Sífilis	<p>Análisis de sangre.</p> <p>Análisis de fluido tomado de una úlcera de sífilis.</p>

Figura 1.3.1: Pruebas Realizadas de acuerdo a la ETS

1.4. Historia de las ETS

1.4.1. Epidemiología de las ETS en el siglo XX

A comienzos del siglo XX las ETS planteaban un grave problema epidemiológico. En 1902, en Francia había un millón de sifilíticos y dos millones de blenorragicos. Las ETS se extienden todavía más tras la I Guerra Mundial: en 1929, los luéticos franceses son ya 8.000.000.

En 1935, Hitler, preocupado por la sífilis congénita y la pureza de la raza aria, promulga la ley por la que se hace obligatorio el examen prenupcial, prohibiendo el matrimonio a los afectos de ETS al tiempo que se les imponía la esterilización por castración.

Tras la II Guerra Mundial, con la introducción de la penicilina, se consigue una rápida disminución en el número de casos, alcanzando a mediados de la década de 1950 sus cotas más bajas.

Sin embargo, la ilegalización de la prostitución, en muchos países, por esta época, redujo la desaparición de los controles sanitarios obligatorios, conlleva pronto un nuevo incremento. A lo largo de los años 1960 algunos factores refuerzan esta tendencia: la aparición de los anticonceptivos orales disminuye el miedo al embarazo y facilita las relaciones sexuales; los viajes y el turismo posibilitan una rápida propagación; y finalmente la "revolución sexual", que a partir de 1968 crea una "nueva moral".

De 20 millones de casos de sífilis declarados en el mundo se pasa a 70 millones en 30 años, y se considera que esto es sólo la punta del iceberg. A ello hay que añadir el aumento notable del resto de ETS, especialmente la gran epidemia de herpes genital de EE.UU. y la aparición reciente de nuevas enfermedades.

1.4.2. El descubrimiento del VIH

El modo de contagio del SIDA estaba claro: se trataba de una enfermedad transmisible que podía adquirirse por contacto sexual, transmisión materna y transfusión de sangre y hemoderivados. Además los pacientes presentaban una afectación muy selectiva de los linfocitos T, con inversión del cociente T4/T8. Una enfermedad de los gatos, la

leucemia felina, presentaba caracteres análogos. Esta infección estaba causada por un retrovirus. Robert Gallo había detectado (1980-82) que dos retrovirus, HTLV-I y HTLV-II, podían afectar al hombre.

En Francia, el equipo del Instituto Pasteur, encabezado por Luc Montagnier, en 1983, tomó una de las adenopatías generalizadas de un homosexual en fase preliminar. Tras un cultivo celular pudieron detectar la retrotranscriptasa del virus. Lo bautizaron como Lymphadenopathy Associated Virus (LAV).

Mientras tanto se habían desarrollado diversas pruebas de laboratorio para la detección de anticuerpos, entre las que sobresalen el ELISA y el Western blot.

1.4.3. El origen de la epidemia

La aparición brusca del Sida, una nueva enfermedad infecciosa, en un momento en que la medicina occidental se jactaba de haber controlado la mayor parte de enfermedades infecciosas, planteó importantes incógnitas. ¿Dónde, cómo y porqué se había originado la misteriosa enfermedad?

El VIH pertenece a un subgrupo particular de retrovirus similar al de los virus de la inmunodeficiencia simia (SIV), que pueden afectar al mandril, al mangabey y al mono verde. Los monos afectados por estos virus los soportan muy bien y no padecen depresión inmunitaria, tal vez porque se han adaptado ya a estos tipos de infección.

La localización geográfica del mangabey coincide bastante con el área afectada por el virus humano VIH. Pudiera ser que algunos de estos monos, que habitualmente son capturados y usados como animales de compañía en las aldeas de la zona, pudiera haber transmitido el virus, tal vez a través de mordeduras. Sin embargo nadie ha podido comprobar estas hipótesis.

Además, la enfermedad parece anterior a la epidemia actual. El caso más antiguo descrito, el de un americano, se remonta a 1952. Se han efectuado pruebas retrospectivas detectando ADN vírico en suero congelado en un marino de Missouri fallecido en 1959, y en un adolescente de Missouri de 1968.

Tal vez el origen de la epidemia deba buscarse en las transformaciones de nuestras sociedades. Quizá, más importante que los cambios en el virus, sean los cambios humanos en el sentido de una menor resistencia natural a la infección los que han favorecido la evolución de la enfermedad. Las grandes migraciones y el turismo de masas probablemente no serían ajenos a la producción de este fenómeno. El cambio en la moral sexual, que siguió a la anticoncepción hormonal, tampoco.

Desgraciadamente el SIDA todavía no es historia. Se trata de un problema vivo y acuciante, que en el instante de escribir estas líneas afecta a millones de seres humanos, la mitad de ellos en África. Las previsiones de la OMS para el año 2000 es de unos 20 millones de infectados en todo el mundo. Poco sentido tiene, pues, seguir tratándolo en un capítulo de historia de las ETS.

La Medicina aprendió de la sífilis no pocas nociones acerca de patogenia, etiología y quimioterapia, y también hoy los nuevos conceptos que está aportando la investigación sobre el SIDA están siendo altamente revolucionarios.

Aunque la historia jamás se repite, ya que varían las circunstancias, sí pueden establecerse ciertos paralelismos. Debemos aprender del pasado para aplicarlo al presente y ayudarnos a vislumbrar, quizá, un posible futuro.

1.4.4. Impacto de las ETS en México

Hasta hace 50 años las ETS constituían una parte importante en la práctica médica; tanto las manifestaciones agudas como sus complicaciones eran comunes. La aparición de nuevos antibióticos y un nuevo escenario hizo pensar que serían erradicadas. Sin embargo, tres acontecimientos modificaron la dinámica de las ETS en las últimas tres décadas provocando su resurgimiento; por un lado los cambios del comportamiento originados por la revolución sexual que aumentaron la probabilidad de exposición, la aparición de cepas de "*Neisseria gonorrhoeae*" y "*Haemophilus ducreyi*", resistentes a antibióticos tradicionales y finalmente la aparición del Virus de la Inmunodeficiencia Humana (VIH), causante del Síndrome de la Inmunodeficiencia Adquirida (SIDA). En la última década, la incidencia de estos padecimientos se ha relacionado tanto con el movimiento poblacional, el crecimiento de las zonas urbanas, y

el desarrollo de zonas turísticas, además del incremento en el uso de drogas y el alcoholismo.

En México, las enfermedades de transmisión sexual ocupan uno de los cinco primeros lugares de demanda de consulta en el primer nivel de atención médica.

Las enfermedades venéreas las podemos ubicar entre las diez primeras causas de mortalidad general en el grupo de hombres y mujeres en edades comprendidas entre los 15 y los 44 años de edad en nuestro país.

No se hace necesaria la penetración vaginal o anal para contagiarse de una enfermedad de transmisión sexual, vale con solo el roce o con un simple tocamiento de los genitales para infectarse, si existe alguna herida en la persona sana.

Antes de que el hombre eyacule existe la salida de líquido pre seminal que es también potencialmente infectante. Una mujer o un hombre pueden tener dos o más enfermedades venéreas en un mismo momento de la vida.

1.5. Acotamiento del problema a resolver

El sistema generará un expediente clínico del paciente de validez oficial, para ello es necesario cubrir con la NORMA Oficial Mexicana NOM-004-SSA3-2012, Del expediente clínico. La cual indica que dicho expediente debe cumplir con:

1.5.1. Historial clínico

La historia clínica es la herramienta más importante en el manejo sindromático de las ETS ya que permiten obtener información de los signos y síntomas presentes. Por lo que el personal de salud siempre debe realizarla:

- Preguntas Generales
 - Nombre
 - Edad
 - Sexo
 - Lugar de origen y residencia
 - Ocupación, escolaridad y estado civil
- Antecedentes Médicos
 - Historia de ETS (secreción vaginal/uretral, úlceras y/o verrugas genitales)
 - Uso de drogas (inyectables u otro tipo)
 - Transfusiones de sangre o hemoderivados
 - Antecedente de alergia a antibióticos, como penicilina o sulfas u otros
 - Historia sexual: edad de inicio de vida sexual, número de parejas sexuales, circuncisión, uso del condón, menarca, fecha de última menstruación, gestas, partos, abortos, cesáreas y uso de métodos anticonceptivos.
- Exploración física: Deberá tener como mínimo: signos vitales (temperatura, tensión arterial, frecuencia cardiaca y respiratoria), peso y talla, así como, datos de la cabeza, cuello, tórax, abdomen, miembros y genitales.
- Resultados previos y actuales de estudios de laboratorio.
- Diagnósticos o problemas clínicos.
- Pronóstico.
- Indicación terapéutica.

1.5.2. Nota de evolución

Deberá elaborarla el médico cada vez que proporcione atención al paciente ambulatorio, de acuerdo con el estado clínico del paciente, esta describirá lo siguiente:

- Evolución y actualización del cuadro clínico.
- Signos vitales, según se considere necesario.

- Resultados relevantes de los estudios de los servicios auxiliares de diagnóstico y tratamiento que haya sido solicitados previamente.
- Diagnósticos o problemas clínicos.
- Pronóstico.

1.5.3. Nota de Referencia/Traslado

En caso de que las muestras de VIH sean positivas, el paciente será canalizado a una ONG para la ayuda de su tratamiento, por lo que se elaborará una solicitud de referencia con los siguientes datos:

- Establecimiento que envía.
- Establecimiento receptor.
- Resumen clínico: el cual contiene el motivo por el cual se envía, impresión diagnóstica (incluido abuso y dependencia del tabaco, del alcohol y de otras sustancias psicoactivas).
- Terapéutica empleada (si la hubo).

Es necesario llevar un control de muestras realizadas por instituciones de salud por lo cual el sistema mostrará la cantidad de muestras que han sido enviadas, recibidas y administradas a los diferentes centros de salud.

Se contará con una base de datos con las pruebas aplicadas a los pacientes y sus resultados, así como un historial de pruebas aplicadas por cada instituto de salud.

Dentro de las actividades a mejorar para el acceso al servicio de salud en ETS están el acceso a búsqueda de casos así como la investigación, es por esto que es necesario realizar el seguimiento de los pacientes así como estadísticas por sexo, edad y ETS.

El sistema contará con un módulo de administración en el cual se podrán dar de alta usuarios con los siguientes perfiles:

- Administrador Sistema.
- Administrador de laboratorios.
- Médico.
- Personal de laboratorio.
- Recepción.

Referencias Capítulo 1

1 Accessed December 2015. [Online]. Available <http://etimologias.dechile.net/?vene.reo> (Ver página: **¡Error! Marcador no definido.**).

2 Accessed December 2015. [Online]. Available <http://www.salud.gob.mx/unidades/cdi/documentos/DOCSAL7609.pdf> (Ver página: **¡Error! Marcador no definido.**).

3 Accessed December 2015. [Online]. Available https://es.wikipedia.org/wiki/Virus_de_la_inmunodeficiencia_humana (Ver página: **¡Error! Marcador no definido.**).

4 Accessed December 2015. [Online]. Available https://es.wikipedia.org/wiki/Virus_del_papiloma_humano (Ver página: **¡Error! Marcador no definido.**).

5 Accessed December 2015. [Online]. Available <https://es.wikipedia.org/wiki/Sífilis> (Ver página: **¡Error! Marcador no definido.**).

6 Accessed December 2015. [Online]. Available <http://www.encyclopediasalud.com/definiciones/parasito> (Ver página: **¡Error! Marcador no definido.**).

7 Accessed December 2015. [Online]. Available http://fresno.pntic.mec.es/msap0005/1eso/T09-virus-bacteria-otros/Tema_9.htm (Ver página: **¡Error! Marcador no definido.**).

8 Accessed January 2015. [Online]. Available <http://www.salud.gob.mx/unidades/cdi/documentos/DOCSAL7609.pdf> (Ver página: **¡Error! Marcador no definido.**).

9 Accessed January 2015. [Online]. Available <http://www.salud.gob.mx/unidades/cdi/documentos/DOCSAL7609.pdf> (Ver página: **¡Error! Marcador no definido.**).

10 Accessed January 2015. [Online]. Available <http://www.salud.gob.mx/unidades/cdi/documentos/DOCSAL7609.pdf> (Ver página: **¡Error! Marcador no definido.**).

11 Accessed January 2015. [Online]. Available <https://www.plannedparenthood.org/esp/temas-de-salud/enfermedades-de-transmision-sexual/pruebas-para-detectar-enfermedades-de-transmision-sexual> (Ver página: **¡Error! Marcador no definido.**).

12 Accessed December 2015. [Online]. Available: http://www.censida.salud.gob.mx/descargas/normatividad/guia_prevenccion_diagnostico_ITS-FEB13CS4.pdf (Ver página: **¡Error! Marcador no definido.**)

CAPÍTULO 2.- MARCO TEÓRICO

2.1 Características, ventajas y desventajas de las bases de datos relacionales y la metodología UWE

2.1.1 Bases de datos relacionales

Introducción

De manera práctica, una base de datos se define como un conjunto de información relacionada que se encuentra agrupada o estructurada, en términos informáticos se entiende también como un sistema formado por una colección de datos almacenados que permiten el acceso directo a ellos, y el grupo de programas que manipulan dichos datos.

El modelo relacional se basa en dos ramas de las matemáticas la teoría de conjuntos y la lógica de predicados de primer orden. El hecho de que el modelo relacional esté basado en la teoría de las matemáticas es lo que lo hace tan seguro y robusto. Al mismo tiempo, estas ramas de las matemáticas proporcionan los elementos básicos necesarios para crear una base de datos relacional con una buena estructura y proporcionan las líneas que se utilizan para formular buenas metodologías de diseño. Las matemáticas son básicas en el modelo relacional, por fortuna, no hay que aprender teoría de conjuntos o lógica de predicados de primer orden para utilizar el modelo relacional.

La teoría matemática proporciona la base para el modelo relacional y, por lo tanto, hace que el modelo sea predecible, fiable y seguro. La teoría describe los elementos básicos que se utilizan para crear una base de datos relacional y proporciona las líneas a seguir para construirla. El organizar estos elementos para conseguir el resultado deseado es lo que se denomina diseño.

El modelo relacional representa la segunda generación de los sistemas gestores de base de datos (SGBD). En él, todos los datos están estructurados a nivel lógico como tablas formadas por filas y columnas, aunque a nivel físico pueden tener una estructura completamente distinta. Un punto fuerte del modelo relacional es la sencillez de su

estructura lógica. Pero detrás de esa simple estructura hay un fundamento teórico importante del que carecen los SGBD de la primera generación, lo que constituye otro punto a su favor.

Dada la popularidad del modelo relacional, muchos sistemas de la primera generación se han modificado para proporcionar una interfaz de usuario relacional, con independencia del modelo lógico que soportan (de red o jerárquico). En los últimos años, se han propuesto algunas extensiones al modelo relacional para capturar mejor el significado de los datos, para disponer de los conceptos de la orientación a objetos y para disponer de capacidad deductiva.

Características del modelo de base de datos relacional

A continuación se enuncian las características más relevantes de las bases de datos relacionales:

- Se componen por tablas, las cuales son matrices rectangulares que almacena líneas con una estructura concreta. La primera línea de una tabla, es una cabecera que indica el nombre de cada columna. O sea, cada columna tiene asignado un nombre único, e indica los valores almacenados en esa.
- Las tablas se componen a su vez de tuplas, las cuales son línea de la tabla que almacena valores concretos para cada columna
- Cada una de las tuplas está formada por elementos atómicos llamados atributos.
- El grado de una tabla es el número de campos (columnas) que posee, el cual regularmente es invariable a lo largo del tiempo.
- El orden de las filas y de las columnas carece de importancia a efectos del SGBD. Este hecho es el que verdaderamente diferencia las tablas relacionales del concepto matemático de relación, en el que el orden de las columnas es fundamental.
- Los atributos solo pueden tomar valores de acuerdo al dominio previamente establecido, por ejemplo, texto, numero entero, numero decimal, carácter, etc.

- Las tuplas poseen claves, las cuales son atributos o conjunto de atributos cuyo valor es único y diferente para cada tupla, de estas se define una de ellas como clave primaria.
- Existen las claves foráneas, la cual es una clave primaria, que es copiada o migrada desde una tupla existente desde otra tabla, y nos permite generar la relación entre 2 tuplas en diferentes tablas.
- En ninguna tupla de una tabla, ninguno de los atributos que formen parte de la clave primaria de una relación podrá tomar un valor nulo. El valor de la clave será único para cada tupla de la tabla
- Si una tupla de una tabla A posee atributos (a1..an) que hacen referencia a la clave primaria de otra tupla de una tabla B, dichos atributos poseen, o bien valores nulos, o bien valores (v1..vn) que se corresponden con la clave de una tupla concreta de B

Ventajas

- Provee herramientas que garantizan evitar la duplicidad de registros.
- Garantiza la integridad referencial, así, al eliminar un registro elimina todos los registros relacionados dependientes.
- Favorece la normalización por ser más comprensible y aplicable.
- Permite la accesibilidad de la información a través de llaves
- Da la posibilidad de integración de manera adecuada con varios sistemas
- Garantiza la independencia de datos
- Se tiene mucha información de cómo realizar su construcción, lo que permite un mayor éxito al momento de su implementación
- La indexación de información permite la búsqueda rápida y ágil de la misma

Desventajas

- Presentan deficiencias con datos gráficos, multimedia, CAD y sistemas de información geográfica.
- No se manipulan de forma manejable los bloques de texto como tipo de dato.
- Dificultad para el manejo de datos no atómicos
- Tiene restricciones en el almacenamiento de datos muy extensos que no pueden ser identificados por llaves

2.1.2 Metodología UWE

Introducción

UWE es un enfoque de ingeniería de software para el dominio Web con el objetivo de cubrir todo el ciclo de vida de desarrollo de aplicaciones Web. El aspecto clave que distingue UWE es la base en diferentes estándares.

El foco principal del enfoque UWE es proporcionar:

- Dominio basado en el Lenguaje de Modelado Unificado (UML por sus siglas en inglés)
- Herramienta de apoyo para el diseño sistemático
- Herramienta de apoyo para la (semi) generación automática de aplicaciones Web.

La notación de UWE se define como una extensión "ligera" de UML, proporcionando algo llamado Perfil UML para el dominio Web.

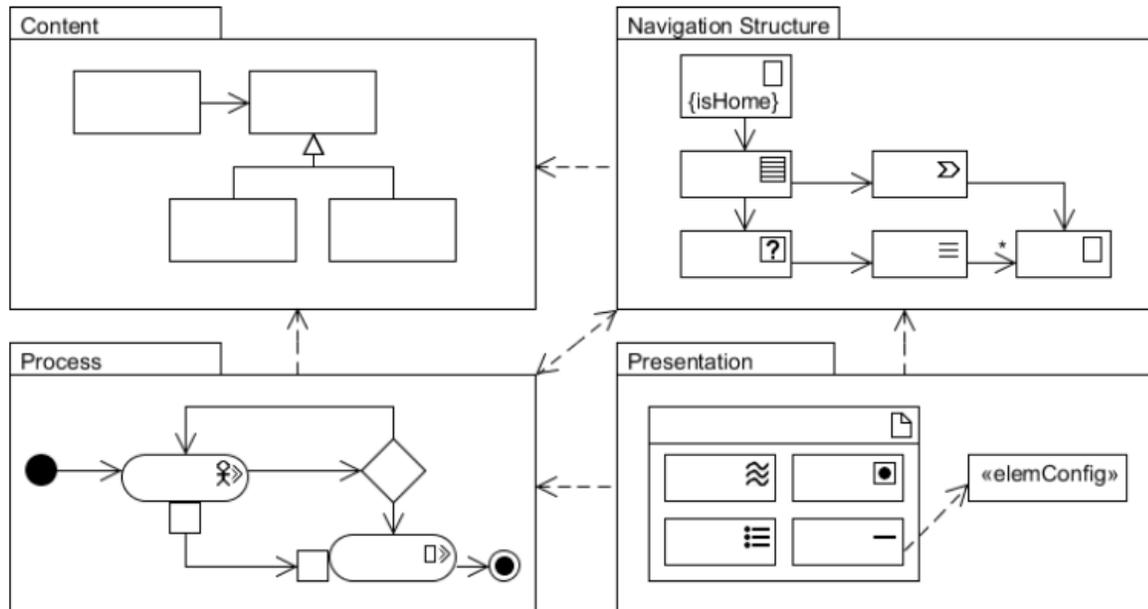
El enfoque actual es en la integración de las funciones de seguridad web en UWE. Además, UWE proporciona soporte de la herramienta para el diseño de modelos, modelo de revisiones de consistencia, y la generación semi-automática de sistemas Web. ArgoUWE (ArgoUML) y MagicUWE (MagicDraw) son plugins que apoyan la notación del perfil UWE y transformaciones para ayudar el trabajo del diseñador.

Para la generación semi-automática de las aplicaciones Web se implementaron diferentes enfoques y actualmente están en proceso de evaluación. Estos enfoques se basan en ATL (ATLAS Transformation Language) y transformaciones gráficas.

El enfoque UWE proporciona una notación de dominio específico, un proceso de desarrollo dirigido por modelos, y el apoyo de herramientas para la ingeniería de aplicaciones Web. Una característica de UWE es el hecho de que no se limita al uso de UML solamente, sino que también utiliza XMI (Metadata Interchange – Intercambio de metadatos) como un formato de intercambio de modelos, MOF (Meta Object Facility - meta-modelado de Objetos) para meta-modelado, los principios basados en modelos del enfoque MDA (Model Driven Architecture – Arquitecturas Dirigidas por Modelos), el lenguaje de transformación de modelos QVT (Query/View/Transformation – Consulta/Vista/Transformación), y XML (Extensible Markup Language – Lenguaje de Marcas Extensible). Las principales razones para el uso de los mecanismos de extensión de UML en lugar de una de las técnicas de modelado de propiedad es la aceptación del UML en el desarrollo de sistemas de software, la flexibilidad para la definición de un lenguaje de modelado específico de dominio Web: el llamado perfil UML, y amplio apoyo de modelado visual por herramientas CASE (Computer Aided Software Engineering – Ingeniería de Software Asistida por Computadora) UML existentes. UWE utiliza notación UML “pura” y tipos de diagramas UML siempre que sea posible para el análisis y diseño de aplicaciones Web, es decir, sin las extensiones de cualquier tipo. Por las características Web, como nodos y enlaces de la estructura de hipertexto, el perfil UWE incluye estereotipos, valores etiquetados y restricciones definidas para los elementos de modelado. La extensión UWE cubre la navegación, la presentación, los procesos de negocio y los aspectos de adaptación. La notación UWE se define como una extensión "ligera" del UML.

Características

En la **Figura 2.1.2.1** se muestra la descripción y las características de la metodología UWE



2.1.2.1 Figura Descripción general de los modelos UWE

El enfoque de diseño UWE para los procesos de negocio en la Web consiste en la introducción de clases específicas de procesos que forman parte de un modelo de proceso independiente con una interfaz definida en el modelo de navegación. Para modelar características adaptativas de aplicaciones web de una manera no invasiva, UWE utiliza técnicas de modelización orientada a aspectos (OMA). Siguiendo el principio “separation of concerns” UWE propone la construcción de un modelo de adaptación de los sistemas personalizados o dependientes del contexto y tejer los modelos después.

UWE divide el desarrollo de los sistemas web en las siguientes capas:

- **Contenido:** El objetivo del modelo de contenido es proporcionar una representación visual del dominio de información relevante para el sistema Web que incluye principalmente el contenido de la aplicación web

- Estructura de navegación: En las aplicaciones Web que contienen la lógica de negocio de los procesos de negocio deben integrarse en la estructura de navegación. Los puntos de entrada y salida de los procesos de negocio son modelados por las clases de procesos en el modelo de navegación, el vínculo entre sí y para las clases de navegación se modela mediante enlaces de proceso. Cada clase de proceso está asociado con un caso que los modelos de un proceso de negocio.
- Procesos: Cada clase de procesos incluidos en el modelo de navegación es refinado en un modelo de proceso que consiste en un modelo de flujo de proceso y opcionalmente de un modelo de estructura de procesos. El control y los datos de flujo se modela en el proceso de flujo de modelo en la forma de un diagrama de actividades UML. Es el resultado de un proceso de refinamiento que se inicia desde el flujo de trabajo en el modelo de requisitos.
- Presentación: El modelo de presentación ofrece una visión abstracta de la interfaz de usuario (UI) de una aplicación web. Se basa en el modelo de navegación y los resúmenes de los aspectos concretos de la interfaz de usuario, como el uso de colores, fuentes, y donde los elementos de la interfaz se colocan en la página web.

Ventajas

- Apoyado en diferentes estándares y modelos
- El pilar de la metodología UWE se encuentra basado en UML, la cual es un estándar de modelado utilizado ampliamente por los desarrolladores de sistemas
- Metodología modular
- Permite presentar prototipos visuales que permiten una conceptualización del producto final
- Específicamente enfocada al desarrollo de soluciones web
- Provee un alineamiento con los objetivos requeridos por el negocio
- Metodología de características prácticas y fácil de entender
- Genera una estructura muy entendible, lo que puede ayudar en la posterior implementación, sobre todo en sistemas con un alto nivel de complejidad

Desventajas

- Al ser modular, si se realiza un cambio en alguna fase, esta afecta las otras 3
- Solo es aplicable a sistemas web
- La metodología puede ser poco específica
- Al estar compuesta de varias etapas, puede generar tiempos altos de mantenimiento, sobre todo en sistemas complejos

2.2 Características, ventajas y desventajas del lenguaje de programación PHP y el servidor WEB Apache Tomcat.

2.2.1 PHP

Es un lenguaje de programación de interpretación por parte del servidor de propósito general y de código abierto, orientado para el desarrollo web y que puede ser embebido en páginas HTML. Sus siglas corresponden al acrónimo (Hypertext Preprocessor – Procesador de Hipertexto). Su logo corresponde a las siglas PHP dentro de un ovalo morado como se observa en la **Figura 2.2.1.1**.



Fig. 2.2.1.1 Logo del lenguaje PHP

PHP es una evolución de un producto llamado PHP/FI. Creado en 1994 por Rasmus Lerdorf, la primera encarnación de PHP era un conjunto simple de archivos escritos en el lenguaje de programación C utilizados para rastrear las visitas al Currículum en línea de Rasmus por lo que las siglas PHP en ese momento significaban Personal Home Page Tools. Con el paso del tiempo se reescribió PHP Tools, permitiendo interactuar con bases de datos, entre otras funcionalidades, dichas herramientas permitieron desarrollar aplicaciones web dinámicas sencillas tales como libros de visitas.

Características de PHP

Un lenguaje interpretado del lado del servidor como lo es PHP es aquel que se ejecuta en el servidor web, justo antes de que se envíe la página a través de Internet al cliente. El cliente solamente recibe una página con el código HTML resultante de la ejecución de PHP aunque no se sabrá el código que formó dicho HTML. El servidor web puede ser configurado incluso para que procese todos los ficheros HTML con PHP, por lo que no hay manera de que los usuarios identifiquen un archivo HTML o una página creada en HTML por PHP como se puede ver en la **Figura 2.2.1.2**.

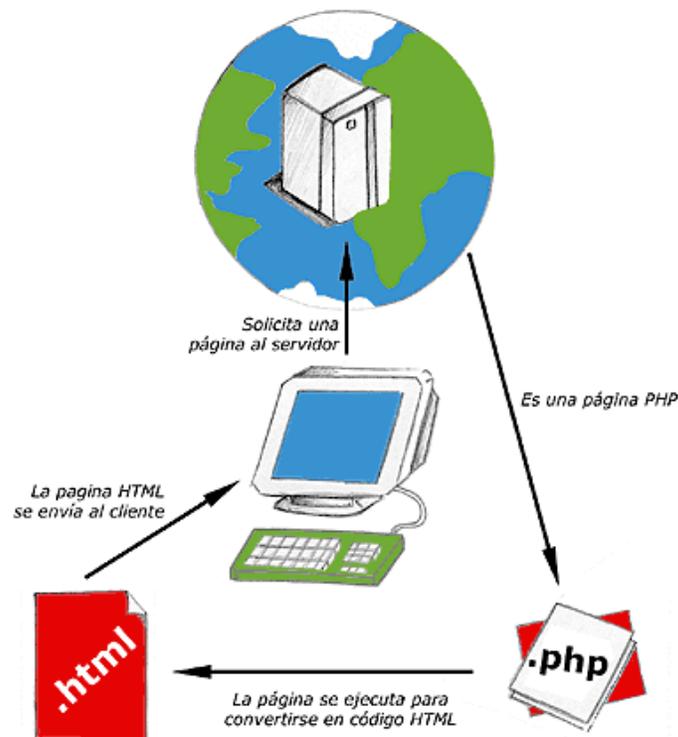


Figura 2.2.1.2 Esquema del funcionamiento de páginas PHP

Debido a la orientación de scripts ejecutados por parte del servidor, PHP es un lenguaje que permite realizar cualquier actividad que otro programa CGI (Interfaz Común de Puerta de Enlace), esto es, crear contenido dinámico en páginas web, recabar datos de formularios, o manejo de cookies.

Ventajas

- Es un lenguaje de acceso libre por lo que no es necesario el pago de licencias.
- Es un lenguaje con una curva de aprendizaje suave y que contiene sintaxis de lenguajes populares como Java o C.
- Los entornos de desarrollo son de fácil configuración.
- Existen paquetes totalmente autoinstalables que integran PHP y el servidor Web.
- PHP puede emplearse en todos los sistemas operativos principales, incluyendo Linux, Unix, Microsoft Windows, Mac OS X, RISC OS.
- Admite la mayoría de servidores web de hoy en día, incluyendo Apache, IIS, y muchos otros.
- Tiene soporte para un amplio abanico de bases de datos como MySQL, PostgreSQL, Oracle, MS SQL Server entre otros.
- Permite conectarse a la base de datos bajo el estándar de Conexión Abierta a Bases de Datos por medio de la extensión ODBC.
- Tiene soporte para comunicarse con otros servicios usando protocolos tales como LDAP, IMAP, SNMP, NNTP, POP3, HTTP, COM (en Windows).
- Permite aplicar conceptos de programación orientada a objetos.
- Existe una gran comunidad de desarrolladores permitiendo encontrar una amplia documentación en la red, así como mantener discusiones en foros.

Desventajas

- Dada la inmensa cantidad de opciones que se tienen para resolver un problema, el no usar buenas prácticas o estándares de programación repercuten en una lógica compleja e incomprensible difícil de dar mantenimiento.

- Debido a que desarrollar una página web dinámica en PHP es relativamente sencillo muchas personas que no son desarrolladores pueden crear aplicaciones sin tener en cuenta factores de seguridad.
- Al ser un lenguaje interpretado, suele ser más lento que un lenguaje de bajo nivel.
- No cuenta con un ambiente de desarrollo que facilite la identificación y corrección de errores.

2.2.2 Apache

El servidor Apache HTTP es un servidor web desarrollado por la Apache Software Foundation dentro del proyecto HTTP/Server con el objetivo de suministrar un servidor seguro, eficiente, y extensible que proporcione servicios HTTP en sincronía con los estándares HTTP actuales. Véase su logotipo en la **Figura 2.2.2.1**.

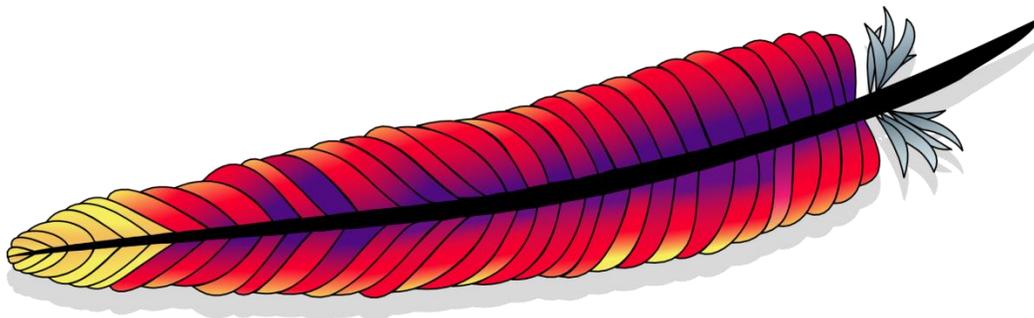


Figura 2.2.2.1 Logotipo del Servidor Apache HTTP

Su nombre proviene de la frase inglesa “a patchy server” debido a que nació como un conjunto de parches para el servidor NCSA HTTPd. Una de las ventajas más grandes de Apache, es que es un servidor web multiplataforma, es decir, puede trabajar con diferentes sistemas operativos y mantener su excelente rendimiento aparte de ser completamente libre.

El servidor Apache es un software que está estructurado en módulos. La configuración de cada módulo se hace mediante la configuración de las directivas que están

contenidas dentro del módulo. Los módulos del Apache se pueden clasificar en tres categorías:

- Módulos Base: funciones básicas del Apache. Las funcionalidades más elementales de Apache se encuentran en este modulo
- Módulos Multiproceso: son los responsables de la unión con los puertos de la máquina, aceptando las peticiones y enviando a los otros procesos a atender a las peticiones. Se han diseñado varios módulos multiproceso para cada uno de los sistemas operativos sobre los que se ejecuta Apache, optimizando el rendimiento y rapidez del código.
- Módulos Adicionales: Cualquier otro módulo que le añada una nueva funcionalidad al servidor. El resto de las funcionalidades del servidor se consiguen por medio de módulos adicionales que se pueden cargar. Para añadir un conjunto de utilidades al servidor, simplemente hay que añadirle un módulo de forma que no es necesario volver a instalar el software.

La arquitectura del Servidor Apache se puede ver en la **Figura 2.2.2.2**

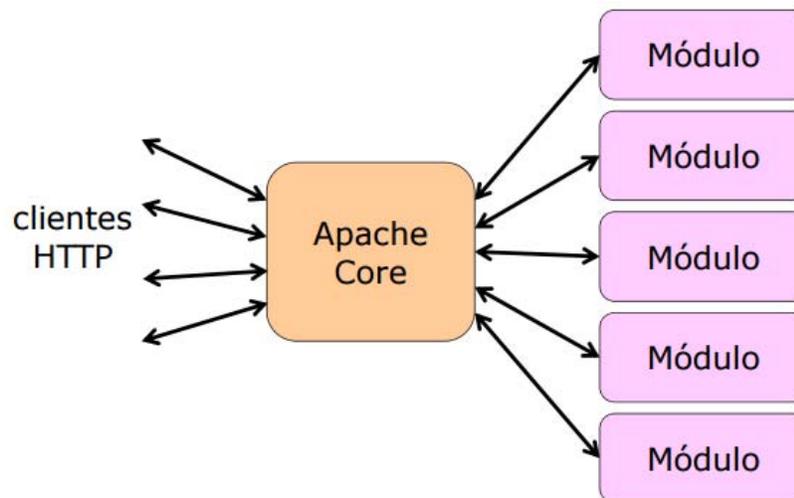


Figura 2.2.2.2 Arquitectura del Servidor Apache

Ventajas

- Apache Web Server es un software de código abierto y completamente gratuito.
- Permite a cualquier programador crear una solución personalizada basada en el programa núcleo de Apache, o ampliar las funciones del software.
- Muchas de estas extensiones personalizadas se han escrito y están disponibles de forma gratuita.
- Al ser de código abierto, Apache se actualiza constantemente.
- El enfoque de código abierto significa que los usuarios no tienen que esperar a que versiones completas del software sean liberadas.
- Apache ha incorporado en su soporte a una amplia gama de lenguajes de programación web, como Perl, PHP y Python.
- Incluye soporte SSL y TLS, protocolos para enviar datos encriptados a través de internet.
- Cuenta con una gran comunidad de usuarios de soporte.
- Al ser de código abierto, Apache está conectado a muchos usuarios que son capaces de crear parches y correcciones de errores técnicos muy rápidamente.
- Se puede instalar en una amplia variedad de servidores y sistemas operativos.
- Es capaz de ejecutarse en todas las versiones del sistema operativos UNIX. Linux es compatible, así como los sistemas operativos Windows y MacOS.
- Es uno de los sistemas más adaptables disponibles en la actualidad, y se ejecuta en una amplia gama de entornos técnicos.
- Es el más usado a nivel mundial.

Desventajas

- No cuenta con formatos de configuración estándar.
- No cuenta con una configuración gráfica oficial.
- Cualquier error en los archivos de configuración, puede ocasionar un mal funcionamiento del servidor.

2.3 Características, ventajas y desventajas del Sistema Gestor de Base de Datos MySQL

2.3.1 Introducción de MySQL

MySQL Es el SGBD más popular de código abierto, Oracle Corporation lo desarrolla, distribuye y le da soporte, nombrado MySQL por la hija del co-fundador Monty Widenius, My. Véase logotipo de MySQL en la **Figura 2.3.1.1**.



Figura 2.3.1.1 Logotipo de MySQL.

Cualquier persona puede utilizar y modificar el software de MySQL sin que esto requiera de un pago, usando el General Public License (GPL) – Licencia Pública General GNU, Acrónimo recursivo de GNU No es Unix en donde se define que se debe o no hacer con el software; si es utilizado de una manera no descrita en la licencia o

para formar parte de una aplicación comercial sería necesario realizar un pago de licencia comercial.

Las bases de datos My SQL son relacionales, lo que nos indica que la información es almacenada en tablas separadas en lugar de almacenar toda la información en un enorme repositorio, el modelo lógico de almacenamiento que utiliza My SQL brinda la posibilidad de establecer las reglas para relacionar los campos de las tablas y los índices entre las diferentes tablas.

Para tener acceso a las Bases de Datos MySQL utiliza Structured Query Language(SQL) - Lenguaje de Consulta Estructurado, este lenguaje se adapta perfectamente a este manejador que basa toda su operatividad en dicho lenguaje. SQL se introduce dependiendo del ambiente de programación utilizado: directamente en el código, embebido en otro lenguaje, por medio de una Application Programming Interface(API) - Interfaz de Programación de Aplicaciones.

El motor de almacenamiento que utiliza por defecto MySQL, es Indexed Sequential Access Method(MyISAM) - Método de Acceso Secuencial Indexado, en este motor las tablas pueden ser estáticas, dinámicas o comprimidas; utiliza un archivo de índices y otro de datos, diseñado para trabajar con muchas operaciones de lectura no así de escritura.

My SQL está enfocado para sistemas cliente/servidor, se maneja con el concepto de multi-hilos y puede brindar el soporte a diferentes aplicaciones informáticas y librerías, herramientas administrativas y una variedad de APIs. Sin embargo My SQL también provee software para sistemas embebidos.

A continuación se muestra un cuadro **Figura 2.3.1.2** de las plataformas para las que descargan MySQL.

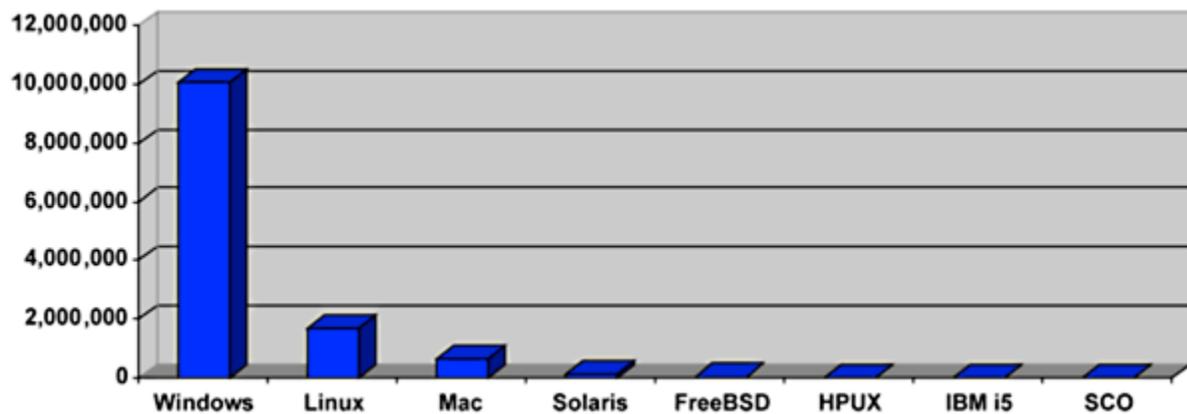


Figura 2.3.1.2 Cuadro de estadísticas de descarga de MySQL entre las principales plataformas

2.3.2 Características de MySQL

- MySQL está escrito en C y C++.
- Se encuentra en constante desarrollo por lo que ha alcanzado una madurez tecnológica al día de hoy ofreciendo niveles confiables de velocidad y seguridad.
- Permite relacionar datos.
- Es de código abierto.
- Permite trabajar con multitud de lenguajes de programación: C, C++, Eiffel, Java, Perl, PHP, Python, Ruby, etc; plataformas tipo: Unix, Linux, Windows, Macintosh, motores: InnoDB, Merge, Memory, Example, Federated, Archive y CSV.

- Las aplicaciones pueden ser usadas de manera aislada o en ambientes donde no existe una red disponible.
- Los índices que maneja pueden ser: Clave Primaria (Valor exclusivo sin nulos), Índice Exclusivo (Permite duplicados), Índice de Texto Completo (Para búsqueda de palabras clave), Índice Ordinario (Provoca bajo rendimiento).
- Se puede trabajar con MySQL desde una ventana de comandos, o través de una interfaz gráfica (phpMyAdmin), desarrollada en PHP la cual permitirá realizar funciones en la base de datos.

2.3.3 Ventajas de MySQL

- MySQL tiene la capacidad de manejar aplicaciones embebidas usando solo 1MB para correr almacenamientos de datos masivos de terabytes de información.
- Puede adaptarse a muchas plataformas: Linux, UNIX, Windows y la naturaleza de código abierto de MySQL permite una completa configuración para cuando sean necesarios requerimientos personalizados para el servidor de base de datos.
- Posee un motor de almacenamiento único permite a bases de datos profesionales configurar el servidor de base de datos para aplicaciones particulares teniendo como resultado una gran eficiencia, si la aplicación es un sistema con transacciones de alta velocidad o un sitio web de gran volumen que requiere de billones de consultas al día, My SQL puede cubrir esas y las más elevadas demandas de cualquier sistema.
- MySQL ofrece mecanismos de alta eficiencia: herramientas de carga de alta velocidad, memoria caché distintiva, índices completos de textos, todo lo que se requiere para los sistemas más críticos de hoy en día.

- Posee una gran disponibilidad de opciones para las configuraciones de réplica maestro/esclavo hasta especializados servidores *cluster* (conjunto de computadoras en red).
- MySQL ofrece uno de los más poderosos motores de base de datos transaccionales en el mercado.
- Todas las cualidades de MySQL aplican al igual para ambientes de almacenamiento donde MySQL escala hacia un rango de terabytes en cada servidor o arquitectura.
- MySQL posee características excepcionales de seguridad que aseguran la protección de datos.
- En cuanto a la autenticación de base de datos MySQL asegura que sólo los usuarios autorizados tendrán acceso al servidor de base de datos con la capacidad de bloquear a los usuarios hasta el nivel de máquina del cliente.
- El password se mantiene cifrado en todo el trayecto mientras se conecta al servidor.
- Provee soporte para Secure Shell (SSH) - Protocolo de transmisión segura y Secure Socket Layer (SSL) - Protocolo de transmisión segura) para asegurar conexiones seguras y protegidas.
- Los usuarios solo ven la información que deben, un poderoso cifrado de datos y funciones de des encriptación aseguran que la información sensible está protegida de vistas no autorizadas.
- Las utilidades para respaldo y recuperación con las que cuenta MySQL permiten la protección de los datos físicos y lógicos.
- Provee soporte para cada necesidad de desarrollo de aplicaciones.
- Dentro de la base de datos se puede encontrar soporte para: procedimientos almacenados, triggers, funciones, vistas, cursores, etc.
- En caso de aplicaciones embebidas bibliotecas de **plug-in**-inserción (añade un servicio específico a un sistema) están disponibles para integrar el apoyo de base de datos MySQL.

- MySQL también provee controladores que permiten a todas las formas de aplicaciones hacer uso de MySQL como SGBD preferido sin importar el lenguaje que se esté usando MySQL provee todo para permitir la construcción del sistema de información.
- MySQL posee la capacidad de un rápido arranque con un tiempo promedio desde la descarga del producto hasta completar la instalación de 15 minutos sobre las plataformas Windows, Linux, Macintosh o UNIX.
- MySQL provee un ambiente de administración gráfica y herramientas de migración que permite al administrador de la base gestionar problemas y controlar el funcionamiento de muchos servidores MySQL desde una sola estación de trabajo.
- Existen herramientas de terceros que pueden adaptarse a MySQL, van desde manejar tareas desde el diseño de datos hasta completar la administración de base de datos y supervisión de rendimiento.
- Existe una versión empresarial en la que se puede obtener el soporte necesario y el servicio profesional, incluso cuenta con indemnización todo por un costo único.
- El coste por migrar una base de datos actual a MySQL o por utilizarla en un proyecto nuevo es menos costoso que los que ofrecen los vendedores de software de propietario.
- Capacidad de transacción distribuida, soporte de transacciones multi versión donde las lecturas no bloquean escrituras y viceversa.
- Una completa integridad de datos, transacciones especializadas en niveles aislados y detecciones inmediatas de puntos muertos.
- Probado con un amplio rango de compiladores.
- Utiliza un rápido sistema de almacenamiento en memoria basado en hilos.
- Provee motores de almacenamiento transaccional y no transaccional.
- Ejecuta el comando de unión muy rápido utilizando un ciclo de uniones.
- Implementa en memoria tablas de estructuras de datos para almacenar un número elevado de información (*hash*) sobre la cual se requiere realizar algún tipo de operación) que son utilizadas como tablas temporales.

- Soporta hasta 50 millones de registros, 200,000 tablas y aproximadamente 5,000,000,000 renglones.
- Se puede conectar a MySQL usando gran variedad de protocolos.
- Los programas en las aplicaciones pueden estar escritos en muchos lenguajes.
- Completo soporte para establecer diferentes caracteres.
- El servidor puede dotar de mensajes de error a los clientes en muchos lenguajes
- Gran cantidad de API's se encuentran disponibles para los distintos lenguajes de las aplicaciones.
- Puede instalarse cómodamente en una computadora tipo laptop o de escritorio.
- No requiere de mucho mantenimiento.
- Se puede optimizar su potencial ajustando memoria, CPU y la capacidad de los puertos disponibles de la máquina en donde se encuentre instalado.
- En la **Figura 2.3.3.1** se muestran los **queries-consultas** por segundo que alcanzan las últimas versiones de MySQL.

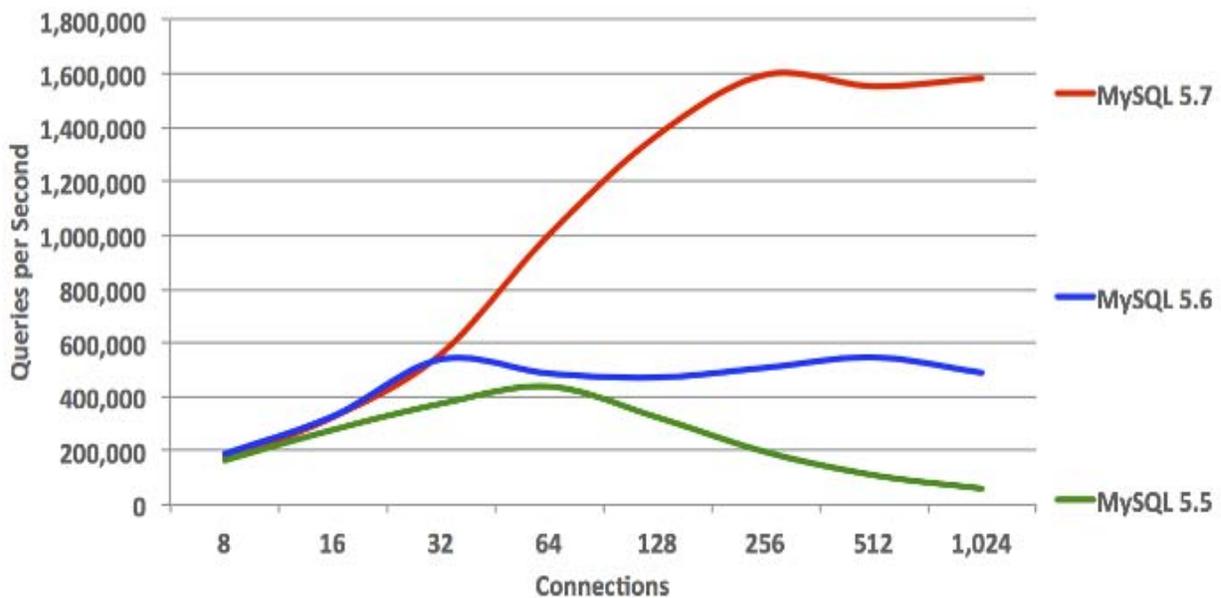


Figura 2.3.3.1 Queries por segundo en las 3 últimas versiones de MySQL.

2.3.4 Desventajas de MySQL

- Se Necesita realizar un buen diseño en la base de datos para no obtener datos inconsistentes, duplicados, etc.
- No todos los tipos de índices están disponibles para todos los tipos de tablas.
- Cada que se realiza una operación de actualización o alta es necesario volver a organizar los índices lo que acarrea una saturación de trabajo para el sistema.
- No llega a ser tan intuitivo como lo son otros gestores de base de datos como Oracle o PostgreSQL.
- No cumple del todo el conjunto de propiedades atomicidad, consistencia, aislamiento, durabilidad (ACID) ya que dependiendo del motor de almacenamiento se acoplara o no a este concepto.
- En cuestión de indexado, se ve superado por las opciones alternas pero dependerá del motor de almacenamiento que se elija ya que ofrecen distintas técnicas de indexación, aunque hay ciertos motores que no la soportan.
- Por lo general, todos los motores de almacenamiento soportan los tipos de datos básicos (números, cadenas de caracteres, etc.) Sin embargo hay otros tipos de datos más complejos a los que no todos los motores dan soporte, por ejemplo BLOB (imágenes, sonido u otros archivos multimedia).
- Dependiendo del motor de almacenamiento puede variar el límite de datos o la anchura de caracteres.
- Es necesaria comprar una licencia GPL de MySQL si desea modificar MySQL y redistribuir el resultado como software no libre.
- Es necesaria una licencia comercial si desea incorporar MySQL dentro de un programa no-Libre.
- El soporte de transacciones o la integridad referencial (la gestión de claves foráneas) en MySQL está condicionado a un esquema de almacenamiento de tabla concreto que puede ser el esquema de almacenamiento “tradicional” MyISAM o el esquema InnoDB.

- No soporta procedimientos almacenados dependiendo de la versión que se utilicé, estas características se incluyen ya en la versión 5.0.
- No incluye el uso de disparadores (Triggers) hasta la versión 5.0.2. por lo que no se podían establecer reglas de integridad y consistencia a nivel de servidor.
- No incluye vistas en versiones previas a la 5.
- Versiones de MySQL previas a la versión 5 no incluía muchas funcionalidades que le permitieran colocarse como un motor de bases de datos mucho más avanzado a la par de sus competidores directos como PostgreSQL o SQL-Server.
- No incluye características de objetos como tipos de datos estructurados definidos por el usuario, herencia, etc.
- El hecho de que no maneje integridad referencial hace de este gestor una solución pobre sobre todo para aquellos programadores que provienen de otros gestores que si poseen esta característica.

Desventaja del motor MyISAM

- No tiene características de atomicidad por lo que puede llegar a dar problemas en la recuperación de datos ya que requiere una exploración, reparación y reconstrucción de índices de datos de las tablas que aún no habían sido volcadas a disco.
- Cuando se realiza una tarea de inserción o actualización de un registro, la tabla se queda bloqueada hasta que finalice esa acción, por lo que mientras dura esa acción, no se podrá realizar ninguna otra.
- No soporta claves foráneas.

Desventaja del motor InnoDB

- El diseño de su estructura es mucho más complejo que el diseño de una tabla MyISAM.

- No permite índices **full-text**-texto completo, que permiten manejar campos con grandes cantidades de texto.
- Es un motor lento en las consultas sencillas (SELECT) pero en los últimos años ha aumentado bastante su velocidad reduciendo mucho la diferencia.
- No dispone de compresión de datos como MyISAM por lo que el espacio en disco y en memoria RAM y su uso de CPU es mayor.
- Las tablas “solo” pueden almacenar 64TB y además cambiar su ubicación es complicado.
- No permite sentencias **TRUNCATE**

Dicho lo anterior si realizamos la comparación entre los dos motores, podemos llegar a una serie de conclusiones: - MyISAM por norma general será más rápido en acciones de inserción, actualizaciones y selecciones de datos. - InnoDB se recupera de errores o reinicios no esperados del sistema a partir de sus logs, mientras que MyISAM requiere una exploración, reparación y reconstrucción de índices de los datos de las tablas que aún no habían sido volcadas a disco.

2.4 Características, ventajas y desventajas del ambiente Cliente - Servidor.

Desde el punto de vista funcional, se puede definir al modelo Cliente/Servidor como una arquitectura distribuida que permite a los usuarios finales obtener acceso a la información en forma transparente aún en entornos multiplataforma.

El funcionamiento básico de este modelo consiste en que un programa cliente realiza peticiones a un programa servidor y espera hasta que el servidor devuelva una respuesta Ver **Figura 2.4.1**.

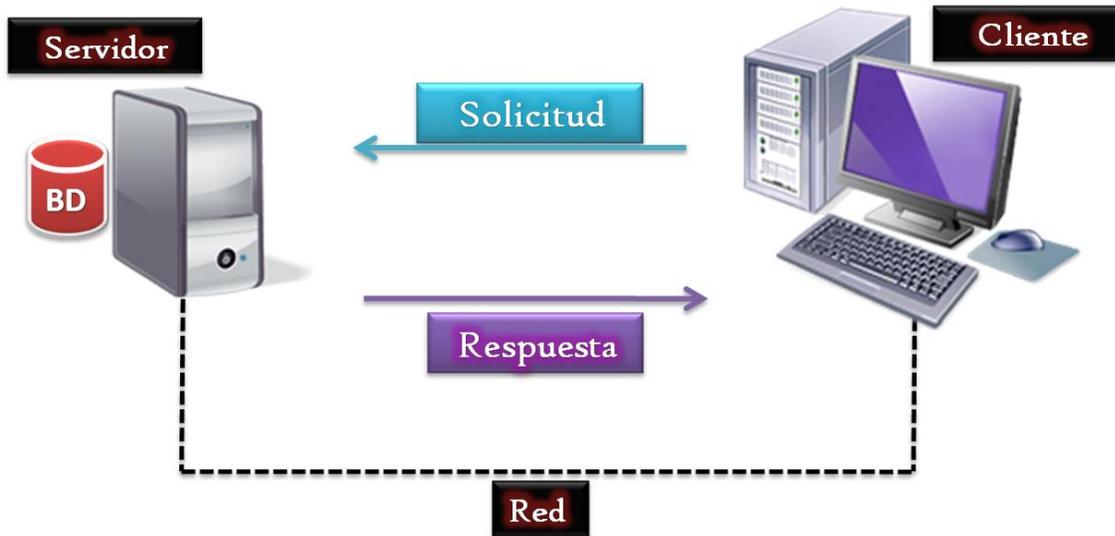


Figura 2.4.1 Modelo Cliente-Servidor.

En su forma más básica debe existir por lo menos dos componentes: el proceso servidor que puede ser ejecutado en las diversas plataformas existentes en el mercado y el proceso cliente, estos procesos clientes se comunican en la red usando uno o varios protocolos de LAN o WAN.

Características del Cliente. El remitente de una solicitud es conocido como cliente, sus principales características son:

- Es quien inicia solicitudes o peticiones, tienen por tanto un papel activo en la comunicación (dispositivo maestro o amo).
- Espera y recibe las respuestas del servidor.
- Por lo general, puede conectarse a varios servidores a la vez.
- Normalmente en el cliente se interactúa directamente mediante una interfaz gráfica de usuario.
- Frecuentemente se comunican con procesos auxiliares que se encargan de establecer conexión con el servidor, El cliente es quien inicia las solicitudes o peticiones, se encarga de recibir la respuesta, manejar las fallas y realizar actividades de sincronización y de seguridad. Por lo que tienen un papel activo en la comunicación

Características del Servidor. El receptor de la solicitud enviada por el cliente se conoce como servidor, sus principales características son:

- Al iniciarse esperan a que lleguen las solicitudes de los clientes, desempeñan entonces un papel pasivo en la comunicación (dispositivo esclavo).
- Tras la recepción de una solicitud, la procesan y luego envían la respuesta al cliente.
- Por lo general, aceptan conexiones desde un gran número de clientes (en ciertos casos el número máximo de peticiones puede estar limitado).
- No es frecuente que interactúen directamente con los usuarios finales.

Los Servidores desempeñan un papel pasivo en la comunicación debido a que al iniciarse esperan a que lleguen las solicitudes de los clientes. En algunos casos existen procesos auxiliares que se encargan de verificar la protección, activar un proceso servidor para satisfacer el pedido, recibir su respuesta y enviarla al cliente devolviendo los resultados. Además deben manejar los bloqueos, la recuperación ante fallas, y otros aspectos afines.

Por las razones anteriores la plataforma computacional asociada con los servidores es más poderosa que la de los clientes. Por esta razón se utilizan PCs poderosas, estaciones de trabajo o sistemas de cómputo robustos. Además deben manejar servicios como administración de la red, mensajes, control y administración de la entrada al sistema ("*login*"), auditoría, recuperación y contabilidad. Usualmente en los servidores existe algún tipo de servicio de bases de datos.

Para que los clientes y los servidores puedan comunicarse se requiere una infraestructura de comunicaciones, la cual proporciona los mecanismos básicos de direccionamiento y transporte. La mayoría de los sistemas Cliente-Servidor actuales se basan en redes locales y por lo tanto utilizan protocolos no orientados a conexión, lo cual implica que las aplicaciones deben hacer las verificaciones.

La red debe tener características adecuadas de desempeño, confiabilidad, transparencia y administración.

Ventajas.

- Uno de los aspectos que más ha promovido el uso de sistemas con esquema Cliente-Servidor es la existencia de plataformas de hardware cada vez más baratas. Esta constituye a su vez una de las más palpables ventajas de este esquema, la posibilidad de utilizar máquinas considerablemente más baratas que las requeridas por una solución centralizada, basada en sistemas grandes. Se pueden utilizar componentes, tanto de hardware como de software, de varios fabricantes, lo cual contribuye considerablemente a la reducción de costos y favorece la flexibilidad en la implantación y actualización de soluciones.
- El esquema Cliente-Servidor facilita la integración entre sistemas diferentes y comparte información, permitiendo, por ejemplo que las máquinas ya existentes puedan ser utilizadas pero utilizando interfaces más amigables al usuario. De esta manera podemos, por ejemplo, integrar PCs con sistemas medianos y grandes, sin que todas las máquinas tengan que utilizar el mismo sistema operativo.
- Al favorecer el uso de interfaces gráficas interactivas, los sistemas construidos con este esquema tienen una interacción más intuitiva con el usuario. Si se utilizan interfaces gráficas para interactuar con el usuario, el esquema Cliente-Servidor presenta la ventaja, con respecto a uno centralizado, de que no es siempre necesario transmitir información gráfica por la red pues esta puede residir en el cliente, lo cual permite aprovechar mejor el ancho de banda de la red.
- Es más rápido el mantenimiento y el desarrollo de aplicaciones pues se pueden emplear las herramientas existentes.
- La estructura inherentemente modular facilita además la integración de nuevas tecnologías y el crecimiento de la infraestructura computacional, favoreciendo así la escalabilidad de las soluciones.

Desventajas.

- El mantenimiento de los sistemas es más difícil pues implica la interacción de diferentes partes de hardware y de software, distribuidas por distintos proveedores, lo cual dificulta el diagnóstico de fallas.
- Se cuenta con muy escasas herramientas para la administración y ajuste del desempeño de los sistemas.

En el desarrollo de aplicaciones Cliente-Servidor se deben tener en cuenta diferentes aspectos, que se mencionan a continuación:

- Es importante que los clientes y los servidores utilicen el mismo mecanismo de comunicación (por ejemplo *sockets* o RPC), lo cual implica que se deben tener mecanismos generales que existan en diferentes plataformas.
- Además hay que tener estrategias para el manejo de errores y para mantener la consistencia de los datos.
- La seguridad de un esquema Cliente-Servidor es otra preocupación importante. En este caso los mecanismos son distintos que en el caso de los sistemas centralizados. Se deben hacer verificaciones en el cliente y en el servidor. También se puede recurrir a otras técnicas como el cifrado de información.
- El desempeño es otro de los aspectos que se deben tener en cuenta en el esquema Cliente-Servidor. Problemas de este estilo pueden presentarse por congestión en la red, dificultad de predecir el tráfico, etc.
- Cómo distribuir los datos en la red. En el caso de una empresa, este puede ser hecho por departamentos, geográficamente o de otras maneras. Además hay que tener en cuenta que en algunos casos, por razones de confiabilidad o eficiencia se pueden tener datos replicados y que puede haber actualizaciones simultáneas.

Clasificaciones cliente-servidor.

Esquemas de una capa.

- Aplicaciones mono-capa. Tanto los datos de aplicación, la interfaz, la lógica del modelo residen en un mismo equipo de cómputo.

Esquemas de 2 capas.

Las aplicaciones Cliente-Servidor clásicas o de 2 capas como su nombre lo indica agrupan la lógica de presentación (interfaz) y la lógica de aplicación en la máquina cliente y acceden a fuentes de datos compartidos a través de una conexión de red que se encuentran en el servidor de datos.

Estas aplicaciones de 2 capas trabajan bien en aplicaciones con un modesto número de usuarios, una base de datos sencilla y una red segura y rápida. Ver **Figura 2.4.2**

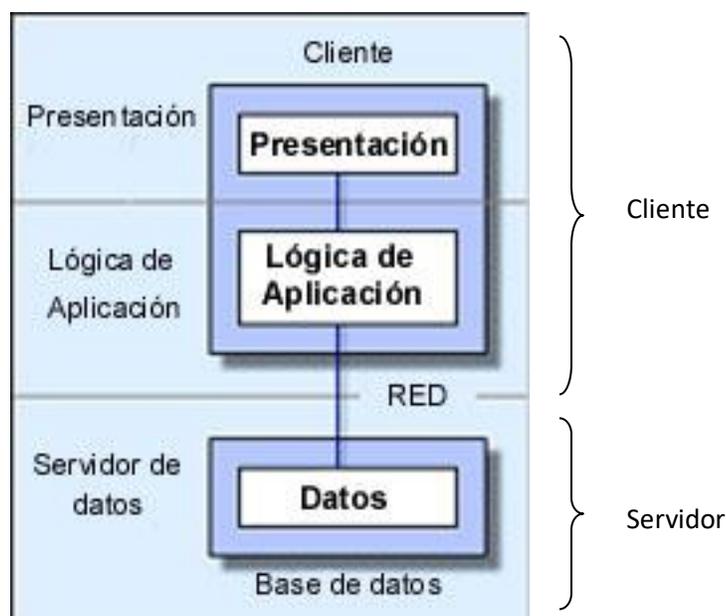


Figura 2.4.2 Esquema Cliente-Servidor de 2 Capas.

Ventajas de Cliente-Servidor de 2 Capas.

- Los datos están centralizados. Esta centralización beneficia a la empresa pues es más fácil compartir los datos, se simplifica la generación de reportes y se proporciona consistencia en el acceso a los datos.
- Desde el punto de vista de procedimientos, es como se localizan en la propia base de datos, pueden procesar la información en su misma fuente, sin tener que transferir a través de la red.

Desventajas de Cliente-Servidor de 2 Capas.

- Difíciles de mantener. Esto viene dado por el hecho de que son difíciles de mantener las reglas de negocio de la lógica de aplicación ya que estas están programadas en cada cliente y esto implica que cualquier cambio tiene que ser redistribuido en todos los clientes.
- Se compromete la confidencialidad. Al tener programada la lógica de aplicación en el cliente este tiene a su disposición todas las reglas de negocio de la empresa.
- Están estrechamente limitadas a una fuente de datos. Los clientes casi siempre están configurados para acceder a una base de datos en particular por lo que mover los datos a una base de datos diferente se hace realmente complicado.
- Hablando de procedimientos almacenados, tienen que ejecutarse en la misma máquina en la que estén los datos lo cual limita la escalabilidad de la aplicación.

El esquema de 3 capas.

En esquema cliente-servidor de 3 capas la presentación, la lógica de aplicación y los elementos de datos están conceptualmente separados. Los componentes de la capa de presentación manejan la interacción con el usuario y realizan las peticiones del cliente a los componentes de la capa intermedia. Los componentes de la capa intermedia, manipulan la lógica de negocio y hacen las peticiones a la base de datos.

Ver **Figura 2.4.3**

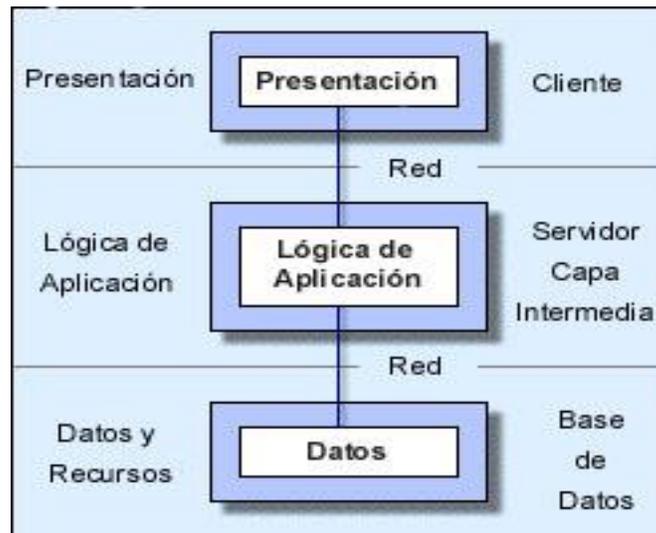


Figura 2.4.3 Cliente-Servidor de 3 Capas.

Una de las diferencias claves entre el modelo tradicional cliente-servidor de dos capas y el de tres capas es el uso de lo que es conocido como “middleware”. En la industria de la computación, “middleware” es un término general que define a cualquier aplicación que sirve para “unir” o mediar entre 2 aplicaciones que usualmente trabajan independientes. Algunos de los más populares tipos de “*middleware*” son los *Transactions Monitors* - Monitores de Transacciones y los *Object Request Brokers* (ORB) - Objeto de los Módulos de Solicitud.

Monitores de Transacciones. Trabajan prácticamente como policías del tráfico ya que estos se colocan entre los clientes y el servidor de los datos y controlan el paso de las consultas o actualizaciones desde los clientes a la base de datos adecuada. Por otro lado dichos monitores mantienen abiertas solo aquellas conexiones a la base de datos que son necesarias lo que trae consigo un mejor rendimiento para el trabajo de la base de datos.

Objeto de los Módulos de Solicitud. Los ORB caen en una categoría diferente a los Monitores de Transacciones. Estos proporcionan transparencia en la ubicación de los módulos y servicios de la aplicación. Esto significa que cuando una aplicación necesita interactuar con un proceso servidor no tiene que saber en qué lugar de la red se encuentra dicho proceso. La única ubicación que conoce la aplicación es la de su ORB “client stub” que se encuentra en la misma máquina que la aplicación.

El ORB “client stub” conoce donde están corriendo todos los procesos servidores o en que máquina tienen que correr. Si un proceso en particular no está corriendo cuando una aplicación cliente lo solicita, el ORB inicia dicho proceso en una de las máquinas para las cuales el proceso está configurado para correr. La aplicación cliente no conoce en que servidor el proceso solicitado está corriendo o si aún lo está. El ORB mantiene una pista de todos los procesos servidores para las aplicaciones clientes.

Ventajas.

- Los componentes de la aplicación pueden ser desarrollados en cualquier lenguaje general lo que posibilita que el grupo de desarrolladores no se centre en el uso de un solo lenguaje.
- Los componentes están centralizados lo que posibilita su fácil desarrollo, mantenimiento y uso.
- Los componentes de la aplicación pueden estar esparcidos en múltiples servidores permitiendo una mayor escalabilidad.
- Los problemas de limitación para las conexiones a las bases de datos se minimizan ya que la base de datos solo es vista desde la capa intermedia y no

desde todos los clientes. Además que las conexiones y los drivers de las bases de datos no tienen que estar en los clientes.

- Los componentes de aplicación de la capa intermedia pueden ser asegurados centralmente usando una infraestructura común. Se pueden conceder o denegar los permisos componente a componente simplificando la administración.

Clasificación de modelos Cliente/Servidor

Se tiene una clasificación de acuerdo a los conceptos de Cliente/Servidor ligero y pesado. En esta se toman en cuenta los grados de libertad que brinda el modelo cliente/servidor para balancear los procesos entre los niveles de presentación, aplicación y base de datos.

Cliente Pesado/Servidor Ligero

- La mayor parte de la funcionalidad de la aplicación corre en el lado del cliente
- El servidor exporta los datos en bruto (sin formato)
- Los servidores son mecanismo de acceso a recursos compartidos.
- Los clientes saben la organización de los datos en el servidor
- Mayor flexibilidad para aplicaciones que implementan nuevas funcionalidades.

En esta podemos encontrar a los servidores de archivos o servidores de bases de datos

Cliente Ligero/Servidor Pesado

- La mayor parte de la funcionalidad se implementa del lado del servidor

- El servidor exporta métodos que operan sobre los datos
- El cliente no consume mayores recursos que la interfaz gráfica
- Incrementar la reusabilidad del código.
- Son más fáciles de desplegar y administrar.
- Se basan en servidores más abstractos que reducen el flujo por la red.
- En vez de proporcionar datos, exportan procedimientos.

En esta última clasificación podemos encontrar al servidor Web, los servidores de transacciones, servidores de correo, etc. Vea **Figura 2.4.4**

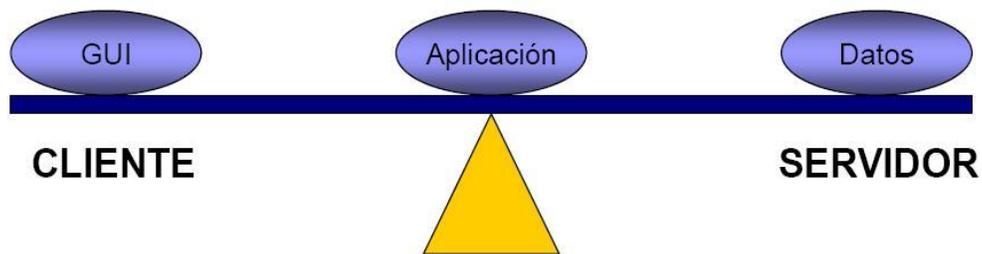


Figura. 2.4.4 Cliente Ligero/Servidor Pesado

Referencias Capítulo 2

13 (2006), Accessed January 2016. [Online]. Available <http://www.uazuay.edu.ec/analisis/Modelo%20Entidad%20Relacion.pdf> (Ver página **¡Error! Marcador no definido.**).

14 N. Koch, A.Knapp and G. Zhang, Accessed January 2016. [Online]. Available <http://www.pst.ifi.lmu.de/veroeffentlichungen/uwe.pdf> Ver página **¡Error! Marcador no definido.**).

15 Accessed December 2015. [Online]. Available <http://php.net/manual/es/history.php.php> Ver página **¡Error! Marcador no definido.**).

16 Accessed December 2015. [Online]. Available: http://dev.mysql.com/tech-resources/articles/move_from_microsoft_SQL_Server.html Ver página **¡Error! Marcador no definido.**).

17 http://ocw.uoc.edu/computer-science-technology-and-multimedia/bases-de-datos/bases-de-datos/P06_M2109_02151.pdf Ver página **¡Error! Marcador no definido.**).

18 Accessed December 2015. [Online]. Available: <http://dev.mysql.com/doc/refman/5.7/en/what-is-mysql.html> Ver página **¡Error! Marcador no definido.**).

19 Accessed December 2015. [Online]. Available: <http://www.mysql.com/why-mysql/benchmarks/> Ver página **¡Error! Marcador no definido.**).

20 Accessed December 2015. [Online]. Available <http://www.tufuncion.com/mysql-innodb-myisam> Ver página **¡Error! Marcador no definido.**).

21 Accessed December 2015. [Online]. Available: http://docente.uco.mx/rcarlos/public_html/cliente-servidor.htm Ver página **¡Error! Marcador no definido.**).

22 Accessed December 2015. [Online]. Available:
http://www.alegsa.com.ar/Respuesta/ventajas_y_desventajas_del_modelo_clienteservidor.htm Ver página **¡Error! Marcador no definido.**)

23 Accessed January 2016. [Online]. Available
<https://programacionwebisc.wordpress.com/2-1-arquitectura-de-las-aplicaciones-web>
Ver página **¡Error! Marcador no definido.**)

CAPÍTULO 3: PLANTEAMIENTO DEL PROBLEMA

3.1. Análisis del problema

Las limitaciones que enfrenta el sistema de salud para atender la ETS plantean muchos retos. En primer lugar, es fundamental que el diagnóstico de las enfermedades se realice de manera oportuna y no en una etapa avanzada, lo que implicaría un inicio tardío del tratamiento y un pronóstico poco alentador para los pacientes.

Las principales razones por las cuales no se acude a un servicio clínico de ETS son:

- Naturaleza asintomática de alguna ETS
- Desconocimiento de los síntomas más comunes
- No se reconoce el riesgo personal
- No se reconoce el riesgo de la pareja
- Rechazo de los servicios por medio al estigma y/o discriminación
- Servicios inaccesibles, inaceptables o de baja calidad.

Es por eso que las consultas médicas son de suma importancia para un oportuno diagnóstico y tratamiento así como para promover el sexo seguro y uso sistemático del preservativo. Brindar información sobre educación sexual y consejería.

Si bien los diagnósticos basados en la exploración física o en el reconocimiento del conjunto de síntomas permiten administrar un tratamiento desde la primera visita ofreciendo al paciente mayores oportunidades de curarse no siempre se obtiene un diagnóstico adecuado ya que no permite identificar ETS mixtas. Es por esto que el diagnóstico basado en pruebas de laboratorio es indicado como la forma adecuada de ofrecer el servicio médico a un paciente con ETS.

CAPÍTULO 3: PLANTEAMIENTO DEL PROBLEMA

Los signos y síntomas de diferentes ETS son similares y para brindar al paciente atención individualizada y resolver su problema, se requiere de laboratorios que realicen pruebas rápidas para el diagnóstico de estas infecciones.

Para el funcionamiento de laboratorios se requiere de personal capacitado, material y equipo que en algunas ocasiones no están disponibles en las unidades de atención, por lo que este servicio se ubica generalmente en aquellas unidades de segundo y tercer nivel de atención.

El proceso para el diagnóstico por pruebas de laboratorio actualmente tarda varios días ya que el médico le brinda al paciente una orden de laboratorio para la cual se debe agendar una cita, dicha cita normalmente es agendada con una diferencia de meses respecto al día en el que el paciente fue a consulta.

Una vez que el paciente ha acudido al laboratorio debe esperar un periodo de tiempo para obtener sus resultados y una vez obtenidos dichos resultados debe de agendar una nueva cita con el médico que lo trata para que interprete los resultados y le brinde un tratamiento adecuado para las ETS encontradas. Dicho proceso se muestra en la

Figura 3.1.1

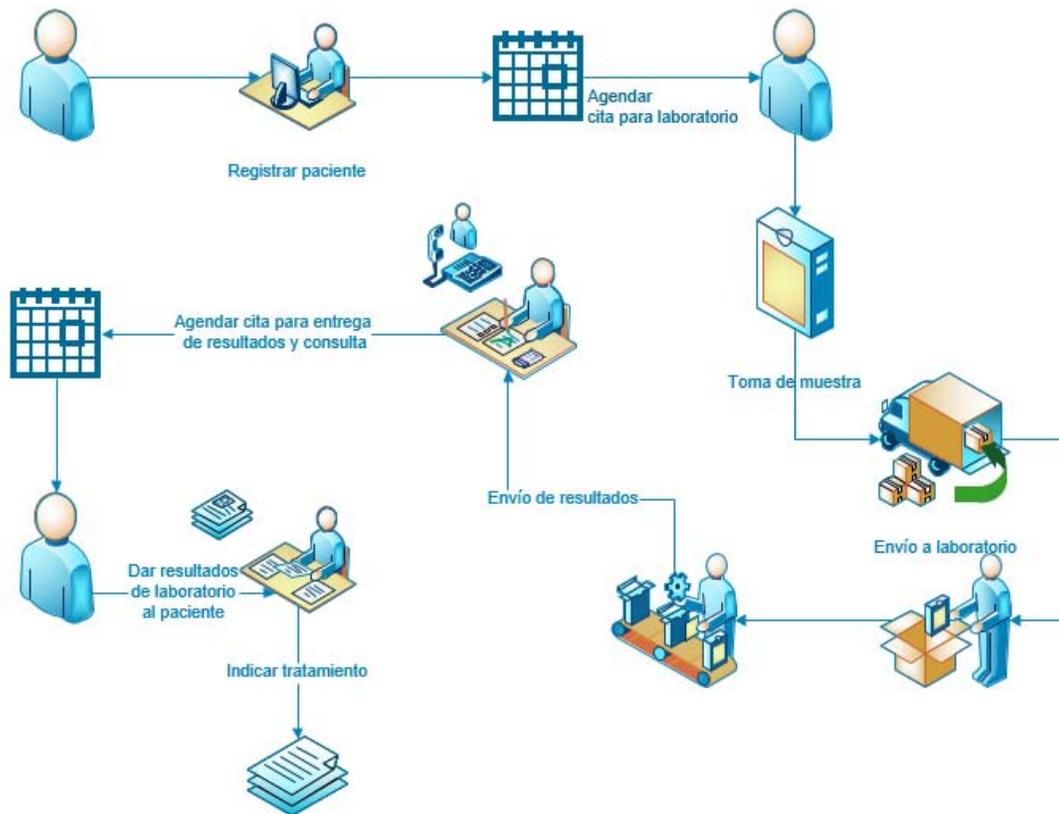


Figura. 3.1.1 Descripción general del diagnóstico de una ETS por medio de pruebas de un laboratorio promedio.

CAPÍTULO 3: PLANTEAMIENTO DEL PROBLEMA

Adicionalmente se indica el manejo general de una persona con sospecha de tener una ETS en la **Figura 3.1.2**

Pasos a seguir para el manejo adecuado de personas con ETS	
Inicio de la consulta	<ul style="list-style-type: none"> • Preguntar el motivo de la consulta. • Escuchar las respuestas y tomar nota de los datos relevantes.
Proveer de información básica sobre ETS	<ul style="list-style-type: none"> • Explicar qué son las ETS y cómo se adquieren. • Subrayar la importancia de recibir tratamiento para las ETS.
Detección de ETS	<ul style="list-style-type: none"> • Elaborar la historia clínica de la persona. • Hacer la revisión física, con énfasis en los órganos sexuales, para detectar los síndromes de ETS más comunes. • Realizar pruebas rápidas o solicitar otros estudios de laboratorio si se consideran necesarios y están disponibles. • Invitar a la persona a volver por sus resultados y a obtener el tratamiento que requiere. • Señalar los riesgos de la automedicación.
Tratamiento	<ul style="list-style-type: none"> • Ofrecer a la persona el tratamiento adecuado a partir de los síndromes detectados, o los resultados de laboratorio obtenidos. • Explicar la importancia de la adherencia al tratamiento para el éxito del mismo.
Prevención	<ul style="list-style-type: none"> • En todos los casos, ofrecer tratamiento para la pareja y los contactos, así como señalar los riesgos de una reinfección. • En todos los casos positivos de ETS, ofrecer la prueba de VIH. • Promover la detección de otras ETS disponibles, como el Virus Papiloma Humano (VPH) y hepatitis B. • Ofrecer las vacunas para el VPH y hepatitis B cuando se consideren necesarias dependiendo de cada caso.
Poblaciones Especiales	<ul style="list-style-type: none"> • En Hombres que tienen sexo con hombres: ofrecer la vacunación para hepatitis B, la detección del VPH, VIH y otras ETS. Canalizar a los Centros Ambulatorios de Prevención y Atención en SIDA y ETS (CAPASITS). • Mujeres embarazadas: ofrecer la prueba de VIH y de sífilis sistemáticamente. Canalizar a los servicios de atención prenatal y planificación familiar. • Personas usuarias de drogas inyectables y otras: ofrecer pruebas de hepatitis B y vacunación, promover la detección de otras infecciones de transmisión sanguínea y canalizar a los Centros “Nueva Vida” o Centros de Integración Juvenil (CIJ) locales • Jóvenes: ofrecer pruebas de ITS y VIH, si se da el caso, y reforzar la necesidad de prevenir la reinfección y el uso de métodos de prevención. Canalizar al Centro Ambulatorio para la Prevención y Atención en SIDA e Infecciones de Transmisión Sexual (CAPASITS) más cercano y a los servicios de salud locales gubernamentales o de la sociedad civil.

Figura 3.1.2 Pasos a seguir para el manejo adecuado de personas con ETS

El proceso resultante con el sistema que se propone busca reducir los tiempos para identificar las ETS contraídas por el paciente y así poder iniciar un tratamiento lo más pronto posible, permitiendo incrementar las probabilidades de un mejor control de los síntomas de dicha enfermedad o bien para la completa erradicación de la enfermedad, el nuevo proceso contempla la aplicación de pruebas rápidas para que el mismo día de la prueba se tengan los resultados, de igual forma una notificación para recoger los resultados de las pruebas es enviado por correo electrónico al paciente y registrado dentro de un sistema para un mejor seguimiento del paciente. El proceso resultante de implementar dicho sistema se ve en la **Figura 3.1.3**.

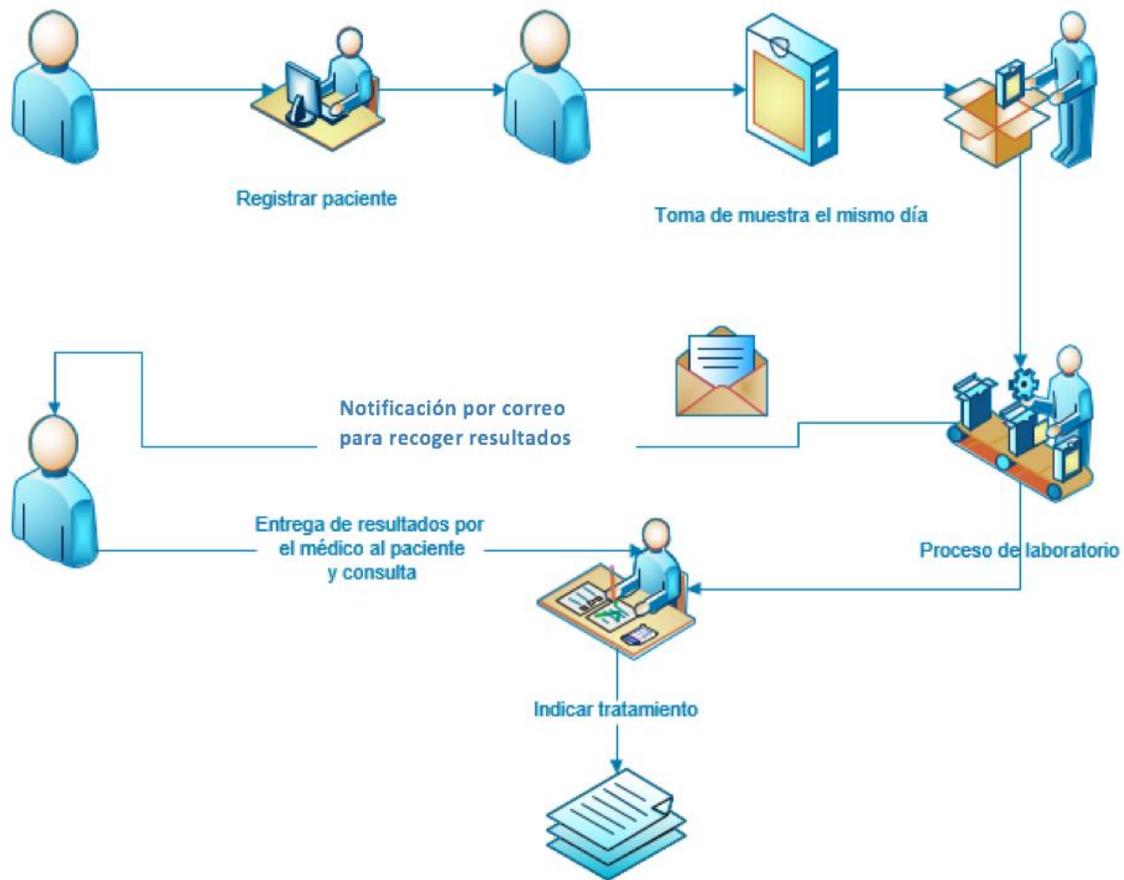


Figura. 3.1.3 Proceso del diagnóstico de una ETS por medio de pruebas de laboratorio resultante de la implantación de un sistema de control de muestras y seguimiento

CAPÍTULO 3: PLANTEAMIENTO DEL PROBLEMA

En cuanto al apego al tratamiento, existen diversas estrategias que pueden realizarse para explicar su importancia a los pacientes e incrementar el apego de cada uno de los pacientes a su tratamiento ya que se ha estimado en distintos tipos de enfermedades y en diversos grupos de poblacionales que la población de pacientes que no toman adecuadamente su medicamentos es de entre el 20 y 80%, con 50% como promedio. También es importante consolidar a los grupos, talleres informativos o vivenciales relacionados con el tratamiento, lo que se puede lograr fortaleciendo a los grupos ya existentes (mejorando la infraestructura, educando al personal).

En lo que respecta a las instituciones de salud, se deben integrar y reorganizar, siguiendo guías de tratamiento claramente establecidas y creando bases de datos que reflejen los distintos aspectos de la epidemia en el país. Debe haber una mejor interacción entre personal médico, de enfermería y trabajadores sociales, así como mayor participación de la sociedad civil y las ONG.

Otro aspecto es limitar el surgimiento de cepas resistentes y limitar el número de cambios que se le deben hacer a un esquema. Es importante tomar en cuenta que el número de pacientes que necesiten antirretrovirales en el caso del VIH/SIDA irá aumentando en los próximos años; siempre y cuando su apego haya sido mayor al 90% para dicha enfermedad, debido a que habrá nuevas personas que cumplan con criterios de inicio de tratamiento. Es por esto que se debe hacer un pronóstico lo más preciso posible de la cantidad de medicamentos que se necesitarán en el futuro próximo. Así, diferentes instituciones como el gobierno y las compañías farmacéuticas podrán planear y movilizar los recursos para continuar con un adecuado abasto de medicamentos.

Diversos factores influyen en el apego a los distintos tipos de tratamiento. Éstos se pueden agrupar en cuatro categorías principales:

CAPÍTULO 3: PLANTEAMIENTO DEL PROBLEMA

- Las características propias del paciente
- El régimen de tratamiento
- La relación médico-paciente
- El sistema de salud

En relación con las características de los pacientes, se han encontrado resultados contradictorios en variables como edad, sexo, escolaridad, estado laboral y estado socioeconómico. Se ha reportado que las personas más jóvenes, de sexo masculino, desempleados y con escolaridad más baja podrían tener un mayor riesgo de presentar un apego inadecuado, pero existen otros estudios en los que estos factores no tienen influencia en la adherencia al tratamiento.

Por otro lado, las características del régimen de tratamiento constituyen un factor que afecta el nivel de apego. La posibilidad de tener un buen apego disminuye con el aumento en el número de pastillas, la frecuencia de las dosis, la frecuencia y severidad de los efectos adversos, y la complejidad y duración anticipada del régimen. Entre más simple sea el régimen antirretroviral, el cumplimiento será mayor.

Los últimos factores que se deben tomar en cuenta son la relación médico-paciente y el sistema de salud en general. Los pacientes que se sienten satisfechos con el estilo de su médico tienen una mayor probabilidad de apegarse al tratamiento. Asimismo, es igualmente importante que el médico crea realmente en el éxito del tratamiento y le transmita esta certeza al paciente. Para que la relación entre ambos sea óptima, debe haber confianza, continuidad e, idealmente, disponibilidad mutua. Se ha visto que si los médicos demuestran que son competentes, tienen una buena comunicación con sus pacientes y los incluyen en las decisiones con respecto a su tratamiento, la relación entre ambos mejora de manera importante.

El personal de salud puede estereotipar a los pacientes o presentar actitudes enjuiciadoras y homofobia, lo que puede ocasionar que las personas infectadas por ETS eviten el sistema de salud. Se ha demostrado lo difícil que es para el personal de salud predecir qué pacientes presentarán buena adherencia y cuáles no, por lo que no

se deben hacer juicios subjetivos sobre el apego a su tratamiento. Otro aspecto importante en relación al sistema de salud es la no disponibilidad y el desabasto de medicamentos, lo que puede interferir de manera importante con la adherencia al mismo.

3.2. Requerimientos generales y particulares de la aplicación.

Los requerimientos son aquellas necesidades del usuario final que definen las cualidades o características que el producto o servicio debe otorgar para cumplir las necesidades del cliente.

El realizar adecuadamente el levantamiento y la definición de los requerimientos es un factor muy importante para llevar a cabo un proyecto exitoso.

En ocasiones se presentan casos en los cuales el sistema opera de manera exitosa, sin embargo, la funcionalidad prestada no cumple con los objetivos que el cliente esperaba, esto es sin lugar a dudas el resultado de un entendimiento inadecuado de las necesidades del cliente, lo cual puede tener impactos importantes a nivel operativo, económico, deterioro de la reputación, entre otros.

Algunas de las dificultades que se pueden presentar al momento de realizar el proceso de administración de los requerimientos pueden ser:

- No se tiene claridad de la necesidad por parte del usuario
- Los requerimientos son muy generales y no se tienen correcta o adecuadamente acotados
- No se tiene documentación de procesos del negocio que sustenten y ayuden a la identificación correcta de los requerimientos
- Existen requerimientos de algunas áreas que pueden ser contrapuestos a otros generados por otras áreas
- Las fuentes de los requerimientos pueden ser muy diversas
- Los requerimientos pueden ser demasiados

- Se puede omitir información importante por parte del usuario final

Por otra parte, algunas de las principales ventajas que se tienen de una administración adecuada de requerimientos son:

- Se gestiona de manera dirigida y estructurada la administración del proyecto
- Se tienen objetivos específicos y bien definidos a cubrir
- Se reducen tiempos ya que se evita el re-trabajo por requerimientos mal definidos u omitidos
- Mejoran la calidad del sistema desarrollado
- Se adecua mejor a la experiencia del usuario, y por ende mejora el nivel de satisfacción del cliente
- Disminuye costos y ayuda a administrar mejor el tiempo durante las diferentes fases del proyecto

Para realizar un proceso adecuado de administración de los requerimientos, se recomienda seguir los siguientes pasos:

- Levantamiento de requerimiento
- Análisis de requerimiento
- Especificación de requerimiento
- Validación de requerimiento

Véase la **Figura 3.2.1**:



Figura 3.2.1: Flujo de la administración de requerimientos

Levantamiento del requerimiento: consiste en entrevistas con los usuarios clave que realizan el proceso actual o utilizan el sistema actual, al cual se le realizan preguntas para con ello identificar los requerimientos que este necesita para el nuevo sistema.

Análisis del requerimiento: Se revisa la información recopilada para validar la viabilidad del requerimiento, y con ello poder acotar el alcance del mismo y sus restricciones.

Especificación del requerimiento: en este paso se busca acotar de manera adecuada el requerimiento, buscando que este cumpla con lo siguiente:

- Conciso: fácil de leer y entender
- Completo: no requiere ampliar detalles en su definición
- Consistente: no es contradictorio con otros requerimientos
- Sin ambigüedad: debe tener una interpretación clara
- Verificable: permite ser comprobable mediante métodos de análisis, inspección, demostración o pruebas

Validación de requerimiento: se revisa con el usuario el requerimiento y se confirma que sea adecuado a la necesidad del mismo, y se adecua en caso de no serlo.

A continuación se describen los requerimientos generales y particulares del sistema, estos se agrupan con base en el nivel en que estos apliquen, ya sea de manera global (como pueden ser requerimientos técnicos), o requerimientos específicos generados regularmente por las áreas de negocio.

3.3.1. Requerimientos Generales

A continuación se desarrollan los requerimientos generales que debe cubrir el sistema, dichos requerimientos como su nombre lo dice, deben de cumplirse para la plataforma en general, indistintamente de las personas u áreas que hagan uso de ella.

Los requerimientos son los siguientes:

- Funcionar sobre plataformas Windows 7, Windows 8.1 y Windows 10

- Aplicación *web*
- Funcionalidad completa independiente del navegador *web*
- Soportar múltiples usuarios simultáneamente
- Sin necesidad de *software* adicional para su funcionamiento

Funcionar sobre plataformas Microsoft Windows: Debe de soportar las plataformas actuales del sistema operativo Windows de Microsoft que son Windows 7, Windows 8.1 y Windows 10.

Aplicación *Web*: La plataforma debe ser *web*, por lo que no requiere el uso de *tick client*, la carga de procesamiento debe generarse principalmente en el servidor *web* de la aplicación y en el servidor de base de datos.

Funcionalidad completa independiente del navegador *web*: la aplicación *web* debe funcionar sobre versiones recientes de los navegadores más utilizados en el mercado, los cuales son, internet explorer, edge (spartan), mozilla firefox, google chrome.

Soportar múltiples usuarios simultáneamente: La plataforma debe poder ser accedida por múltiples usuarios a la vez, se considerará un número de 10 conexiones simultáneas como máximo.

Sin necesidad de *software* adicional para su funcionamiento: El sistema solo debe requerir que el equipo de cómputo del usuario final solo necesite la instalación de alguno de los navegadores *web* mencionados anteriormente, no debe ser necesario la instalación de *software* complementario para la funcionalidad del mismo.

Acceso mediante usuario y contraseña: El sistema requerirá el uso de un usuario y contraseña para poder ingresar, sin esto el sistema no permitirá el acceso.

3.3.2. Requerimientos Particulares

CAPÍTULO 3: PLANTEAMIENTO DEL PROBLEMA

A continuación se describen los requerimientos particulares del cliente(Organización con fines para la detección y tratamiento de ETS), los cuales se generaron con base en la información particular de las áreas que intervienen en los procesos que interactúan con el sistema.

- Captura y almacenamiento de datos generales del paciente
- Administración de inventario de pruebas nuevas
- Registro de resultados de las pruebas realizadas
- Historial de pruebas realizadas
- Seguimiento del paciente de manera ágil
- Almacenamiento de datos de usuario a los que se les proporcione medicamento
- Estadísticas de pacientes hombres y mujeres
- Estadísticas de pacientes con infecciones de ETS por rango de edad

Captura y almacenamiento de datos generales del paciente: El sistema tendrá la capacidad de capturar datos generales del paciente, se deben tomar como base los siguientes:

- Datos personales, como nombre completo, fecha de nacimiento, estado civil, entre otros
- Datos de domicilio donde habita
- Datos como talla: peso: presión arterial y frecuencia cardiaca

Administración de inventario de pruebas nuevas: El sistema tendrá la capacidad de administrar el inventario de las distintas localidades, para con ello determinar qué área requiere se provea de más pruebas reactivas y con ello evitar la falta de material para la operación de estas.

Registro de resultados de las pruebas realizadas: El sistema debe permitir la captura de los resultados de las pruebas realizadas a cada paciente para con ello tener acceso de manera electrónica.

Historial de pruebas realizadas: El sistema debe permitir llevar un histórico de las pruebas realizadas a cada paciente, para con esto comparar los resultados anteriores y llevar el historial clínico del paciente de manera adecuada.

Seguimiento del paciente de manera ágil: El sistema deberá permitir una rápida búsqueda de la información del paciente, para que la atención que se le proporcione sea más ágil en comparación con las búsquedas manuales que se realizan en el proceso actual.

Estadísticas de pacientes hombres y mujeres: el sistema proporcionará la opción de obtener datos estadísticos de las pruebas realizadas a pacientes hombres y mujeres, para con esto evaluar tendencias de infección por género.

Estadísticas de pacientes con infecciones de ETS por rango de edad: el sistema proporcionará la opción de obtener datos estadísticos de las pruebas realizadas a personas de diferente edad, para con esto evaluar tendencias de infección por rango de edades.

3.3. Análisis y posicionamiento de los lenguajes para el *back-end* y *front-end*.

Al día de hoy tenemos a nuestro alcance una variedad de tecnologías de las que podemos hacer uso dependiendo de nuestras necesidades; de acuerdo a los requerimientos que demanda nuestro sistema es necesario contar con el siguiente tipo de *software* que nos sirva como herramienta para el desarrollo del producto:

- Sistema Gestor de Base de Datos.
- Lenguaje de programación, para una arquitectura cliente-servidor.
- Un navegador de *internet*.

El sistema planteado es una aplicación *web* que se ejecutará en *internet*, los datos son almacenados y procesados en la *web*, se puede acceder a este servicio desde cualquier lugar y momento mientras exista una conexión a *internet*, enviando a nuestro

dispositivo el contenido de la solicitud que hayamos realizado, algunos ejemplos de servicios que utilizan aplicaciones *web* son: Gmail, Yahoo! , Google Drive, Facebook, etc.

Algunas ventajas de utilizar aplicaciones *web* son:

- Muchas de las aplicaciones *web* que existen son gratuitas.
- Se puede tener acceso a la información en cualquier momento y desde cualquier lugar.
- No influye el dispositivo de acceso ya que la información está contenida en la red.
- Permiten que varios usuarios estén realizando peticiones al mismo tiempo.
- La información no se pierde o se elimina a menos que así lo solicite el usuario.

Las aplicaciones *web* se basan en la arquitectura cliente- servidor que administran servidores *web* utilizando como interfaz páginas *web*.

El cliente lanza una petición en una interfaz mediante un navegador que solicita páginas con información al servidor *web*, el cual las recibe mediante una conexión de red utilizando el protocolo *Hypertext Transfer Protocol* (HTTP) – Protocolo de Transferencia de Hipertextos), el servidor localiza la página y la envía de vuelta al navegador que realizó la solicitud, como se muestra gráficamente en la **Figura 3.3.1**.

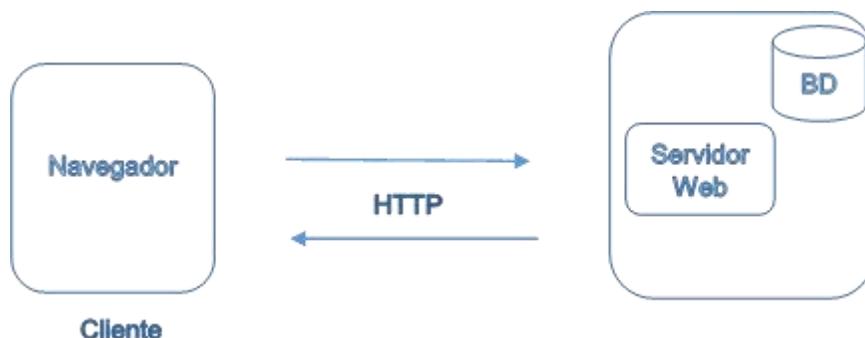


Figura 3.3.1 Aplicación *web* sobre un modelo Cliente Servidor.

Para el SGBD, hacer una elección incorrecta de este elemento puede implicar altos costos tanto de *hardware* como de *software*, en este componente se encontrarán alojados toda la información del sistema y es crucial darle la importancia que merece.

Para elegir el mejor SGBD nos podemos basar en diversos parámetros entre los principales se encuentran:

- Soporte que brinda el SGBD.
- Carga de transacciones que soporta.
- Sistema operativo en el que se implementará.
- Para el caso de las aplicaciones *web* se considera la disponibilidad y carga de trabajo que tendrá el servidor.
- Experiencias previas propias o externas.
- Entorno con el que interactuará.
- Costos que implicará en licencias, soporte, etc.
- Recuperación de información en caso de fallo.
- *Hardware* y *Software* necesario.

Para decir que un SGBD cuenta con un buen desempeño intervienen 3 elementos para dicha evaluación: Potencia del procesador del *Central Processing Unit (CPU)* - Unidad Central de Proceso), memoria *Random Access Memory (RAM)*- Memoria de Acceso Aleatorio disponible, eficiencia total real tanto de la red como del disco duro, adicional a esto es importante considerar otro elemento llamado diseño, si contamos con los recursos más eficientes pero hacemos un mal diseño de la base de datos estaríamos poniendo en riesgo el hecho de garantizar la integridad y una eficaz operatividad de la base; y por otro lado si no contamos con los recursos optimizados por factores internos y externos que intervengan limitando el potencial del sistema, un buen diseño (consultas ágiles del lado del cliente) y configuración(ambiente del SGBD correctamente configurado del lado del servidor para una respuesta pronta a las solicitudes del cliente) de la base nos darán un mejor rendimiento con las herramientas existentes que se provean.

A continuación se describirán algunos SGBD para encontrar cual se adapta mejor a los requerimientos del sistema, y la mejor integración que pueda realizarse con este, el servidor *web* y el lenguaje de programación.

- MySQL: Distribuido por Oracle, es el sistema más popular de código abierto, aunque con la opción de adquirir una licencia para una aplicación comercial; las bases de datos de MySQL son relacionales, enfocado a sistemas cliente/servidor, soporta gran variedad de lenguajes de programación y plataformas; el lenguaje que utiliza para obtener acceso a la base de datos es SQL, ofrece niveles confiables de velocidad y seguridad, puede utilizarse desde sistemas simples hasta sistemas complejos como clúster en red, ofrece un ambiente tanto por línea de comandos o mediante una interfaz gráfica, algunos de los casos de éxito para este sistema son: Ticketmaster, Wikipedia, Pay Pal, Juegos Big Fish, etc.
- SQL Server Express: Distribuido por Microsoft, es la edición gratuita de SQL Server, utilizada para aplicaciones *web* y pequeños servidores, utiliza el mismo motor de base de datos que SQL Server pero en una versión adaptada, cuenta con herramientas para la administración, desarrollo, etc., cuenta con un almacenamiento de base de datos de 10GB, su lenguaje de consulta se basa en T-SQL y ANSI SQL, cuenta con un ambiente gráfico, y con una amplia comunidad en línea.
- PostgreSQL: Es un sistema de bases de datos relacionales de código abierto con más de 15 años de desarrollo activo, muy popular en la comunidad UNIX/LINUX, soporta una variedad de arquitecturas, cuenta con interfaces nativas de programación tales como C/C++, Java, .NET, Perl, Python, Ruby, etc., cuenta con amplia documentación, el tamaño máximo para la base de datos es ilimitado e incluye la mayoría de los tipos de datos de SQL:2008 algunos de

CAPÍTULO 3: PLANTEAMIENTO DEL PROBLEMA

los casos de éxito son: Radio Paradise, Shannon Medical Center, Spiros Louis Stadium, etc.

Teniendo en cuenta las características de cada SGDBD en la **Figura 3.3.2** se muestra una tabla comparativa entre los sistemas anteriormente descritos.

Parámetro	SGBD		
	My SQL	SQLServer Express	PostgreSQL
Costo	Libre y de Pago	Libre	Libre
Open Source	Si	No	Si
Plataformas	Unix, Linux, Windows, Macintosh, SuSE, Fedora, Solaris, HPUS,AIX, SCO, FreeBSD	Windows	Unix, Linux(AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64) y Windows
Tamaño límite de la base	Limitado por el Sistema Operativo	10 Gb	Ilimitado
ACID	Depende del motor que se utilice	Si	Si
Transacciones	Si	Si	Si
Vistas	Si	Si	Si
Procedimientos Almacenados	Si	Si	si
Triggers	Si	Si	Si
Subconsultas	Si	Si	Si
Escalable	Si	No	Si
Veloz	Si	No	Si
Consumo de recursos	Bajo	Bajo	Alto

Figura 3.3.2 Tabla comparativa con las principales características de MySQL, SQLServer y PostgreSQL.

La plataforma de desarrollo para la aplicación *web* es lo que le va permitir al sistema crear páginas dinámicas que son recibidas e interpretadas por el servidor; así mismo

CAPÍTULO 3: PLANTEAMIENTO DEL PROBLEMA

este envía una página después de ejecutar los *scripts* correspondientes en respuesta a una solicitud del cliente; para escribir dichas páginas existen diversas tecnologías.

Para realizar la elección de la tecnología adecuada tenemos los siguientes parámetros:

- Nivel de madurez del lenguaje.
- Comunidad en línea.
- Disponibilidad y variedad de librerías y aplicaciones.
- Curva de aprendizaje del lenguaje.
- Compatibilidad con el *software* en su entorno.
- Rendimiento
- Escalabilidad
- Seguridad
- Costo.

Entre algunas tecnologías se encuentran:

- PHP: Es un lenguaje libre, diseñado para ser incorporado en documentos HTML, cuenta con una amplia documentación online, tiene soporte para una variedad de manejadores tales como: MySQL, PostgreSQL, Oracle, Sybase; su curva de aprendizaje es suave, tiene integración con librerías externas, permite la generación de documentos PDF, de fácil mantenimiento, archivos XML los cuales puede autogenerar y almacenar en el sistema de archivos en lugar de presentarlos en la pantalla, soporta la mayoría de servidores *web*: Apache, Microsoft Internet Information Server, Personal Web Server, Netscape e iPlanet, O'Reilly Website Pro server, Caudium, Xitami, OmniHTTPd, etc; algunos casos de éxito de este lenguaje son: Wikipedia, Yahoo! , Facebook, etc.
- Java: Es una tecnología popular, con gran variedad de herramientas eficaces como los IDE: Eclipse, Netbeans, JCreator, es un lenguaje complicado de comprender, en parte por la gran variedad librerías, herramientas de terceros y

CAPÍTULO 3: PLANTEAMIENTO DEL PROBLEMA

técnicas que hay que dominar, es una tecnología rápida, escalable; permite generar dinámicamente HTML, XML, en respuesta a una solicitud del lado del cliente, puede correr en diferentes plataformas y servidores de páginas *web*.

- HTML5: Es un *software* libre, desarrollado por *World Wide Web Consortium (W3C)* – Consorcio WWW y *Web Hyertext Application Technology Working Group (WHATWG)* - Grupo Web de Hipertexto y Aplicación de Tecnología de trabajo), es la última versión de HTML, ahora HTML5 posee más tecnologías como la integración de contenidos de video y audio permitiendo a las aplicaciones *web* tener un mayor alcance; HTML5 puede ser editado mediante un editor de textos básico; no es considerado como un lenguaje de programación estrictamente debido a que no crea aplicaciones sino documentos, para agregar estilos se combina con lenguajes como *Cascading Style Sheet (CSS)* – Hoja de estilo en cascada.

En la **Figura 3.3.3** se muestra una comparación entre los lenguajes: PHP, JAVA, HTML5

CAPÍTULO 3: PLANTEAMIENTO DEL PROBLEMA

Parámetro	Lenguaje de Programación		
	PHP	Java	HTML5
Costo	Libre	Libre	Libre
Soporte	Soporte gratis en la comunidad en línea Opciones disponibles de pago	Soporte gratis en la comunidad en línea Opciones disponibles de pago	Soporte gratis en la comunidad en línea
Plataformas	Multiplataforma	Multiplataforma	Multiplataforma
Costo del Hardware	Bajo	Alto	Bajo
Seguridad	Muy buena	Buena	Buena
Rendimiento	Muy bueno	Requiere de configuración extra para un buen rendimiento	Muy bueno en ordenadores, bajo en móviles.
Escalabilidad	Buena	Buena con una configuración apropiada	Buena
Administración	Regular	Moderada	Moderada
Configuración	Regular	Moderada	Moderada
Frameworks	Una gran variedad	Uno estandarizado	Una gran variedad
Compatibilidad entre versiones	Muy Buena	Mala	Buena
Fácil de aprender	Si	Moderada	Si
Compatibilidad con navegadores	Muy buena	En general buena solamente algunas incompatibilidades con chrome	En general bien soportado por Firefox, Chrome, Opera, Deficiente para Safari e Internet Explorer 10

Figura 3.3.3 Cuadro comparativo basado en distintos parámetros de los lenguajes PHP, JAVA y HTML5.

En la **Figura 3.3.4** se muestra un cuadro con los sitios más populares y las plataformas y lenguajes que utilizan.

Site	Up Since	Server Platform	Programming Language
Google.com	November 1998	Linux	C, Java, C++, PHP & MySQL
Facebook.com	February 2004	Linux	PHP, MySQL and C++
YouTube.com	February 2005	Linux	C, Java and MySQL
Yahoo.com	August 1995	Linux	C++, C, Java, PHP & MySQL
MSN.com (owned by Microsoft)	August 1995	Windows	ASP.net
Live.com (owned by Microsoft)	August 2008	Windows	ASP.net
Wikipedia	January 2001	Linux	PHP & MySQL
Amazon.com	October 1995	Linux & Solaris	C++, Java, J2EE
WordPress.com	November 2005	Linux	PHP & MySQL

Figura 3.3.4 Sitios de *internet* más populares y las tecnologías que utilizan.

CAPÍTULO 3: PLANTEAMIENTO DEL PROBLEMA

Un servidor *web* es un programa que sirve para atender y responder a las diferentes peticiones de los navegadores, proporcionando los recursos que soliciten usando el protocolo HTTP o el protocolo HTTPS (la versión cifrada y autenticada).

Un servidor *web* adiciona las siguientes ventajas:

- Se puede acceder y realizar transacciones desde cualquier punto de la red.
- La información que maneja es de distintos tipos tales como: datos, documentos, multimedia, etc. Por lo general la información es de solo lectura por lo que el usuario no puede modificar o alterar las fuentes originales.
- El servidor *web* es de fácil acceso, por lo que permite propagar información de forma instantánea mediante su almacenamiento en el servidor.

El esquema básico de un servidor *web* consta del siguiente algoritmo que se ejecuta n veces.

- Espera las peticiones en el puerto TCP indicado el 80 por defecto.
- Recibe la petición
- Busca el recurso
- Envía el recurso utilizando la misma conexión por la que recibió petición
- Vuelve al segundo punto

Para elegir un servidor *web* se definen los siguientes criterios:

- Necesidades de la aplicación: En ocasiones basta con los requisitos definidos para el sistema, pero en otras ocasiones la aplicación va unida al servidor tal ejemplo es PHP con Apache.
- Naturaleza de la aplicación: Debido a un requerimiento específico puede resultar necesario la utilización de un componente particular que viene incluido en un servidor determinado.
- Experiencia propia o de terceros:

CAPÍTULO 3: PLANTEAMIENTO DEL PROBLEMA

- Base de servidores: Tener muchos servidores o todas sus versiones, puede llegar a implicar un crecimiento desmedido y como consecuencia complicar el mantenimiento y la actualización de los mismos, así como retrasar la detección oportuna de errores.
- Experiencia de quien administra el sistema: Mantener los servidores funcionando depende del administrador, un mal administrador puede tener funcionando una buena aplicación y le resultará complicado a un buen administrador mantener una aplicación deficiente.
- Rendimiento.

Tomando en cuenta lo descrito tenemos las siguientes opciones como servidores de aplicaciones:

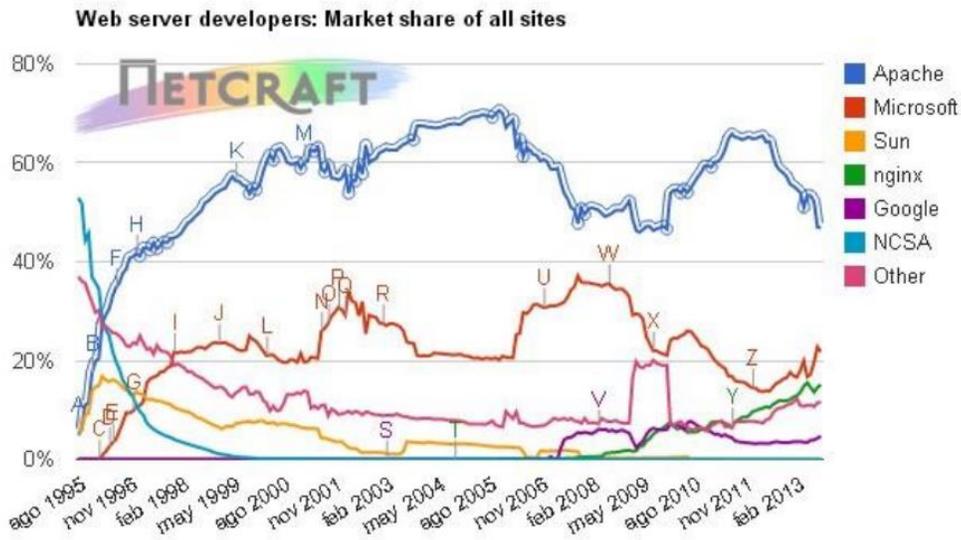
- **Apache:** Es un servidor *web* de código abierto multiplataforma que implementa el protocolo HTTP, su objetivo es ofrecer un servidor *web* confiable, eficiente extensible y de código fuente abierto. Es el servidor *web* con mayor uso de la red desde hace mucho tiempo (según la lista de la Netcraft el 60% de los servidores *web* de *internet* son Apache).
- **Cherokee:** De código abierto, inició en 2001 por Álvaro López Ortega, servidor multiplataforma, escrito en C, soporta complementos para aumentar funcionalidad, su objetivo es ser rápido sin ser robusto, toda la configuración se realiza mediante Cherokee-Admin una interfaz amigable y de gran alcance; el principal inconveniente de este servidor es la falta de servicios dinámicos.
- **NGINX:** De código abierto, servidor *web* y *proxy* inverso (correo electrónico o *e-mail* del tipo IMAP/POP3), ligero y de alto rendimiento, su funcionamiento está basado en eventos y se ejecuta desde el mismo proceso de los servidores más usados.

En la **Figura 3.3.5** se muestra un cuadro comparativo entre los principales servidores *web* libres.

Parámetro	Servidor Web		
	Apache	Cherokee	NGINX
Costo	Libre	Libre	Libre
Rendimiento	Bueno	Muy bueno	Bueno
Plataformas	Multiplataforma	Multiplataforma	Multiplataforma
Tecnologías que soporta	PHP, Perl, CGI,SCGI, Java, SSL, TLS, HTTP, etc	FastCGI, SCGI, PHP, uWSGI, LDAP, CGI, SSI, TLS/SSL, HTTP, proxy, streaming, video, etc.	TSL, SSL, FastCGI, SCGI o uWSGI, entre otros.
Configuración	Fácil	Regular	Complicada
Soporte	Extenso soporte gratis en la comunidad en línea (Apache Group)	Soporte gratis en la comunidad en línea	Soporte gratis en la comunidad en línea
Escalabilidad	Regular	Buena con los complementos adecuados	Buena
Administración	Fácil	Regular	Regular
Uso de Recursos	Robusto	Ligero	Ligero
Velocidad	Medio	Alto	Alto
Contenido dinámico	Si	No	No(Solo por medio de herramientas externas)
Seguridad	Buena con la configuración adecuada	Buena	Alta

Figura 3.3.5 Cuadro comparativo entre Apache, Cherokee y NGINX.

En la **Figura 3.3.6** se muestra la cuota de mercado en 2013 para los servidores *web*



Developer	August 2013	Percent	September 2013	Percent	Change
Apache	336,622,050	46.96%	346,288,706	46.86%	-0.10
Microsoft	163,098,703	22.75%	160,691,763	21.74%	-1.01
nginx	104,311,568	14.55%	111,680,078	15.11%	0.56
Google	30,550,914	4.26%	34,806,502	4.71%	0.45

Figura 3.3.6 Estudio de mercado de Netcraft Septiembre 2013.

Al momento de definir la arquitectura que se utilizará sin duda se eligió una aplicación *web* como solución, siendo que se basa en el modelo Cliente-Servidor en donde el usuario podrá hacer uso de la aplicación desde un navegador a través de *internet* y mandar peticiones al servidor *web*.

Por parte del cliente su acceso será por medio de una interfaz dinámica en un navegador, dicha interfaz será desarrollada de acuerdo al alcance del sistema y las necesidades que se desean cubrir; cabe mencionar que para lo que deseamos cubrir el lenguaje PHP nos brinda sin duda una solución bastante adecuada a lo que se desea, por su sencillez su compatibilidad con los navegadores y con la mayoría de los servidores pero en especial Apache y su alto rendimiento, se descarta Java porque su

CAPÍTULO 3: PLANTEAMIENTO DEL PROBLEMA

configuración es más complicada y se descarta HTML5 porque la presentación de información no se puede procesar ni almacenar ya que no permite la manipulación por un programa debido a su anarquía.

Por otro lado por la parte del servidor necesitaremos de un servidor *web* en este caso utilizaremos Apache ya que nos permite el desarrollo y ejecución de aplicaciones *web* dinámicas dando más fuerza a elegir PHP como el lenguaje de nuestra interfaz, ya que se llevan muy bien Apache y PHP, además que Apache es el único que permite contenido dinámico sobre los otros servidores analizados.

El servidor de aplicaciones *web* aparte de alojar nuestra aplicación incluyendo toda la lógica de negocio (Validaciones, seguridad, transacciones a BD) también tendrá interacción con un manejador de Base de Datos existen opciones de *software* libre como PostgreSQL que descartamos por su alto consumo de recursos y su limitado tamaño de la base de datos, SQLServer Express también se descartó por su baja velocidad y su tamaño limitado de la base de datos; sin duda el que mejor se apega es MySQL porque brinda agilidad, fácil manejo de la información y administración, bajo consumo de recursos, velocidad y es bien conocida la compatibilidad PHP-Apache-MySQL.

Por lo anterior mencionado la mejor solución que se apega para cubrir nuestra necesidad es desarrollar una aplicación *web* que estará atendiendo solicitudes de usuarios en distintas sesiones a través de una interfaz (desarrollado en PHP) desde un navegador que residirá en un servidor de aplicaciones (Apache) que aparte de dar respuesta a cada solicitud también tendrá comunicación con una base de datos (MySQL).

3.4. Diseño a nivel de bloques de la interfaz de usuario.

La estructura base de la página se muestra a continuación en un diseño a nivel de bloques (**Figura 3.4.1**) en el cual se definen las partes principales a considerar

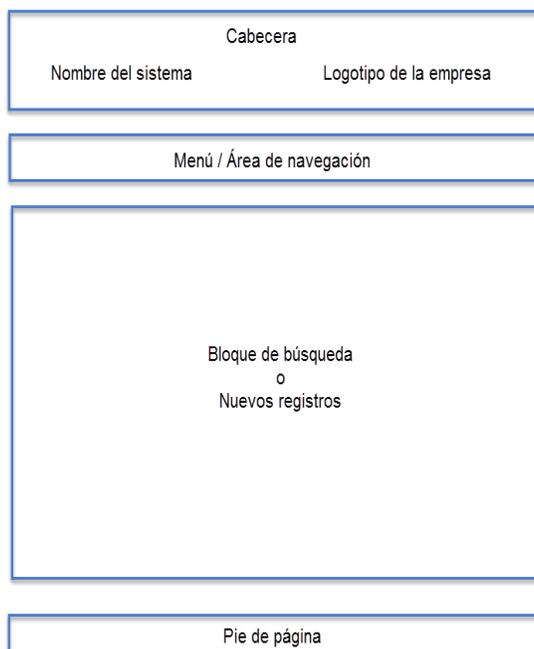


Figura 3.4.1 Diseño a nivel bloques

Cabecera

Bloque en el que se definen el nombre del sistema el cual será Sistema de control de pruebas de enfermedades de transmisión sexual (VIH, VPH y Sífilis) y en la parte izquierda se muestra el logotipo de la empresa.

Menú o área de navegación

Bloque en el cual se van a mostrar las opciones a realizar dentro del sistema, se tienen consideradas las siguientes opciones (Ver **Figura 3.4.2**):

CAPÍTULO 3: PLANTEAMIENTO DEL PROBLEMA

- Buscar paciente, al seleccionar esta opción el sistema va a mostrar los campos de búsqueda autorizados tales como apellido paterno, apellido materno, nombre o RFC
- Alta paciente, al seleccionar esta opción el usuario podrá registrar a un paciente con todos los datos necesarios para llevar el control correspondiente, estará separado por secciones las cuales serán:
 - Datos Generales
 - Domicilio Completo
 - Antecedentes
 - Laboratorio
 - Tipo de prueba a realizar
- Control de muestras, esta opción permitirá registrar las muestras que han sido enviadas al laboratorio para su evaluación, así como también se podrá llevar un inventario de las muestras que llegan, que salen o que se consumen en el laboratorio.
- Reportes se podrán visualizar los reportes configurados de acuerdo a las necesidades de la empresa con la finalidad de tener información estadística de las pruebas realizadas a los pacientes.
- Salir, opción que permite al usuario salir del sistema.



Figura 3.4.2 Menú

Bloque de búsqueda o nuevos registros

De acuerdo a la opción seleccionada en el menú (Bloque anterior), en éste bloque se podrán ver los campos ya sea para una búsqueda de algún registro o para el ingreso de un nuevo registro, ya sea paciente o resultados de las muestras realizadas.

Contorno de la página: se refiere a la plantilla que tiene toda página *web*, imprescindible para distribuir los contenidos de la página. La anchura puede ser expandible y abarcar toda la ventana del navegador, o fija donde el contenido siempre tiene el mismo ancho, independientemente del tamaño de la ventana.

Cabecera: se refiere a los logotipos, marcas y colores que se utilizan en las diferentes formas de *marketing* de una compañía, como pueden ser tarjetas de visita y folletos. Debe contener el nombre y el logotipo de la compañía y ubicarse en la parte superior de cada página. Este bloque permite a los usuarios asegurarse que las páginas por las que navegan son parte de un mismo sitio.

Menú o área de navegación: el sistema de navegación del sitio debe ser sencillo de utilizar y estar situado en la mitad superior de la página. La ubicación de esta mitad de la página dependerá de las dimensiones del navegador y la resolución en pantalla del usuario. Con una resolución de 800×600 *píxeles*, contando con la barra de dirección y la barra de estado, la mitad está situada aproximadamente a unos 200 *píxeles* a partir de la parte superior.

Bloque de búsqueda o registros nuevos: el contenido es lo esencial en Diseño *web*. Cuando un visitante no puede encontrar lo que busca se va a otro sitio. Es importante mantener este bloque en el centro de atención del diseño para que el usuario acceda al sistema y encuentre la información que necesita.

Pie de página: situado en la parte inferior de la página, normalmente contiene el *copyright*, el contacto y la información legal, así como algunos vínculos a las secciones principales del sitio. El pie de página debe indicar a los usuarios que se encuentran en la parte inferior de la página.

Referencias Capítulo 3

24 Accessed December 2015. [Online]. Available: <http://www.comentum.com/php-vs-asp.net-comparison.html> (Ver página **¡Error! Marcador no definido.7**).

25 Accessed December 2015. [Online]. Available: <http://www.redeszone.net/2015/03/29/nginx-el-servidor-de-paginas-web-comparado-con-apache-http-server/> (Ver página **¡Error! Marcador no definido.1**)

CAPÍTULO 4: ANÁLISIS, DISEÑO Y CONSTRUCCIÓN

4.1 Aplicación y separación de los módulos según la metodología UWE

Las actividades base de modelado de UWE son el análisis de requerimientos al cuál se hizo referencia en el capítulo 3, el modelo conceptual, el modelo de navegación y el modelo de presentación los cuales se presentaran más adelante. A estos modelos se pueden sumar otros modelos como lo son el modelo de interacción y la visualización de Escenarios Web.

Los diagramas usados por UWE, son diagramas UML puro, en la metodología UWE se hace distinción de casos de uso que son solamente consultas y presentación de datos al usuario, conocidos como “*browsing*” y aquellos casos en los que se modela un cambio en la información “*processing*”.

Para realizar el diagrama de casos de uso es necesario establecer quienes serán los actores del sistema.

- **Recepcionista:** El actor recepcionista es aquel encargado de verificar si un paciente se encuentra registrado en el sistema y en caso contrario deberá llenar la forma dentro del sistema con los datos generales del paciente. Dicho actor también es el encargado de tomar las muestras de análisis y enviarlos al laboratorio. La **Figura 4.1.1** muestra los casos de uso para dicho actor.

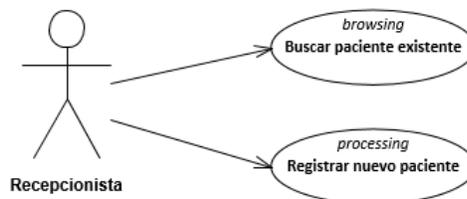


Figura 4.1.1 Diagrama de casos para el actor Recepcionista

- Personal de laboratorio: Una vez obtenido los resultados de los estudios de laboratorio se deberán ingresar al sistema. Los resultados también se enviarán físicamente al médico. El diagrama de casos de uso para éste actor se muestra en la **Figura 4.1.2**

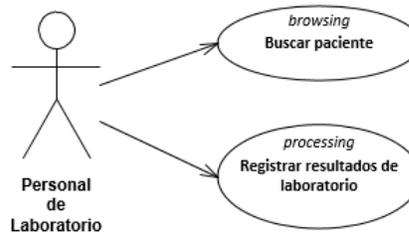


Figura 4.1.2 Diagrama de casos para el actor Personal de laboratorio

- Médico: El médico constatará que los resultados de las pruebas de laboratorio hayan sido ingresados al sistema, informa personalmente al paciente sobre los resultados y proporcionará un tratamiento de ser necesario. Los casos de uso para el médico se observan en la **Figura 4.1.3**

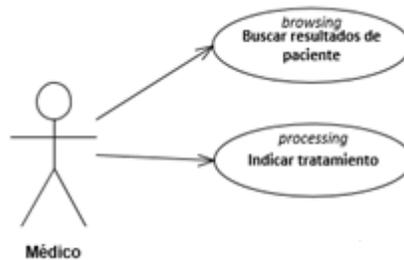


Figura 4.1.3 Diagrama de casos para el actor Médico

- Administrador de laboratorio: Es el encargado de registrar los kits de muestras que se envían a las distintas jurisdicciones, confirmar la recepción de los kits, registrar cuántas muestras han sido usada por cada una de las jurisdicciones, así como generar los reportes del uso de dichas muestras y estadísticas sobre los pacientes. Su diagrama de caso de uso se muestra en la **Figura 4.1.4**

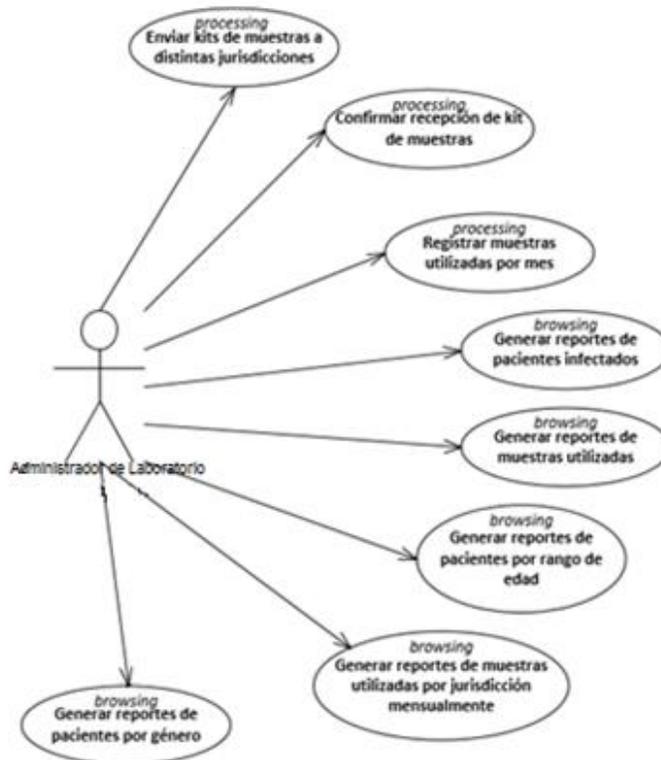


Figura 4.1.4 Diagrama de casos para el actor Administrador de Laboratorio

- Administrador del sistema: Su función será creación de usuarios dentro del sistema o reestablecer las contraseñas de los mismos, como se muestra en la **Figura 4.1.5**

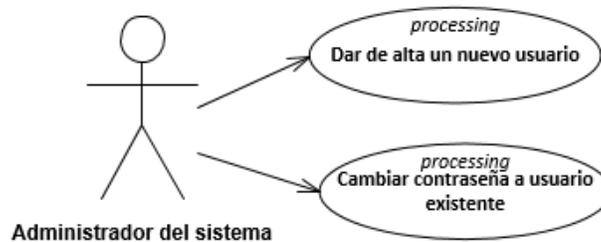


Figura 4.1.5 Diagrama de casos para el actor Administrador del sistema

Definición de procesos

CAPÍTULO 4: ANÁLISIS, DISEÑO Y CONSTRUCCIÓN

- Registro de pacientes: Cuando llega un paciente a consulta el recepcionista debe verificar si se encuentra dado de alta en el sistema. En caso de que se trate de un paciente nuevo se registrarán los datos generales del paciente. En esta etapa del proceso se aplicará la muestra y se enviará al laboratorio.
- Registro de resultados de laboratorio: Se obtendrán los resultados el mismo día que se realizaron las pruebas de laboratorio, el personal de laboratorio registrará en el sistema dichos resultados e informará al médico tratante del paciente.
- Interpretación de resultados y tratamiento: El médico recibe los resultados físicamente y los comparará con los capturados en el sistema. El médico indicará el tratamiento e indicaciones a seguir.
- Conclusión del expediente: El médico notifica al paciente por correo electrónico que sus resultados están listos para que acuda a su interpretación
- Control de muestras
- Reportes

4.2. Diseño de modelos

4.2.1. Diagrama de contenido

El objetivo del modelo de contenido es proporcionar una especificación visual de la información en el dominio relevante para la aplicación Web. Este es un diagrama UML normal de clases, por ello se debe pensar en las clases que son necesarias para el caso de estudio presentado. En la **Figura 4.2.1.1** se presenta el diagrama de clases para el modelo de contenido.

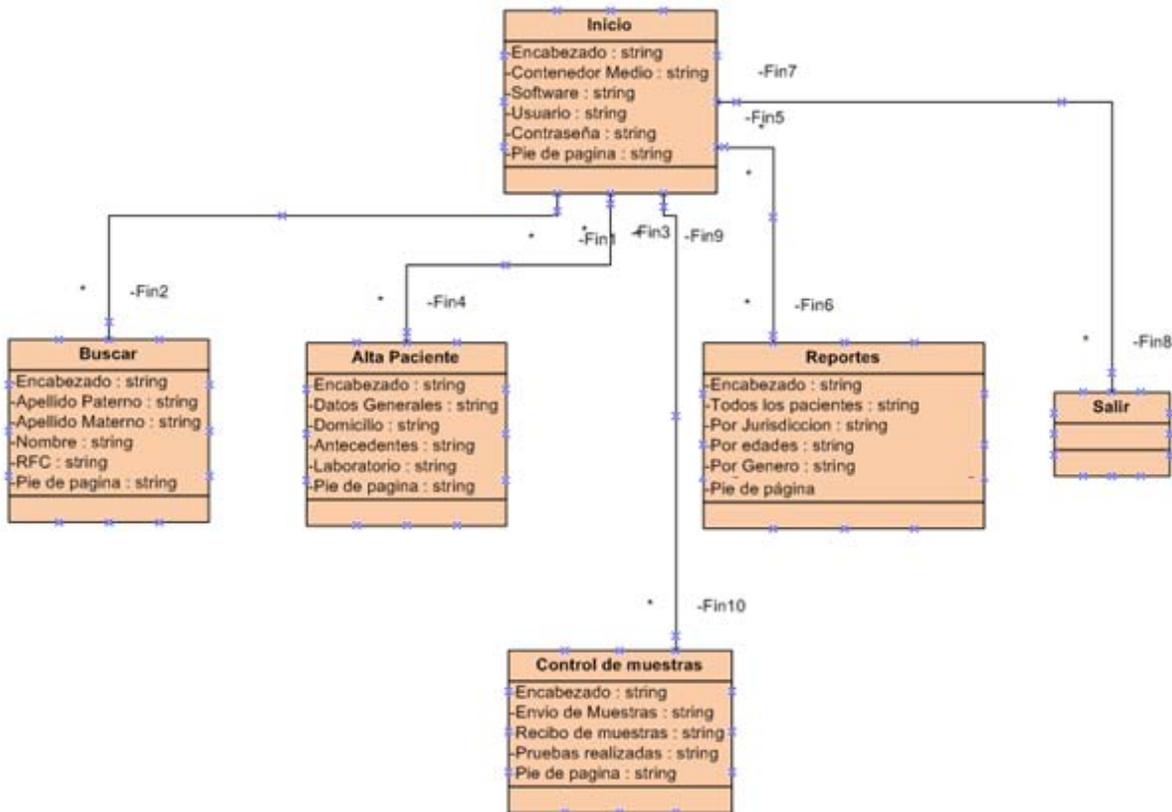


Figura 4.2.1.1 Diagrama de contenido

En la clase “Inicio” se modela el inicio de la aplicación web, se almacenan las credenciales y propiedades que sirven para identificar al usuario que quiere iniciar

CAPÍTULO 4: ANÁLISIS, DISEÑO Y CONSTRUCCIÓN

sesión, aquí se validan que los datos proporcionados por el usuario sean correctos para que inicie sesión en el sistema.

En la clase “Buscar” se modela la información que el usuario proporciona para realizar una consulta y los métodos se usan para generar un listado de resultados obtenidos de acuerdo a la información que haya proporcionado el usuario, la búsqueda puede ser por datos específicos del paciente como apellido paterno, apellido materno, nombre o RFC.

En la clase “Alta paciente” se registran los datos del paciente para ser guardados en tablas de la base de datos, estos datos van desde información general del paciente, domicilio, antecedentes, registrar primera consulta en el laboratorio y el tipo de muestra o toma que se va a realizar el paciente.

En la clase “Control de muestras” se registra un inventario de las muestras que llegan al laboratorio, que salen del laboratorio o que son ocupadas por los pacientes que se registran en el laboratorio correspondiente.

En la clase “Reportes” se enlistan una serie de reportes que muestran estadísticas de las enfermedades contraídas por los pacientes, así como la clasificación de acuerdo a la zona o jurisdicción correspondiente.

La clase “Salir” termina la sesión del usuario llevándolo a la página de inicio, listo para la próxima vez que el usuario desee ingresar al sistema.

4.2.2. Diagrama de Navegación

En una aplicación para la Web es útil saber cómo están enlazadas las páginas. Ello significa que se requiere un diagrama de navegación con nodos y enlaces. Este diagrama se modela con base en el análisis de los requisitos y el modelo de contenido. UWE provee diferentes estereotipos para el modelado de navegación, en la **Figura 4.2.2.1** se presentan los usados en este caso de estudio y seguidamente se da una descripción de cada uno de ellos.

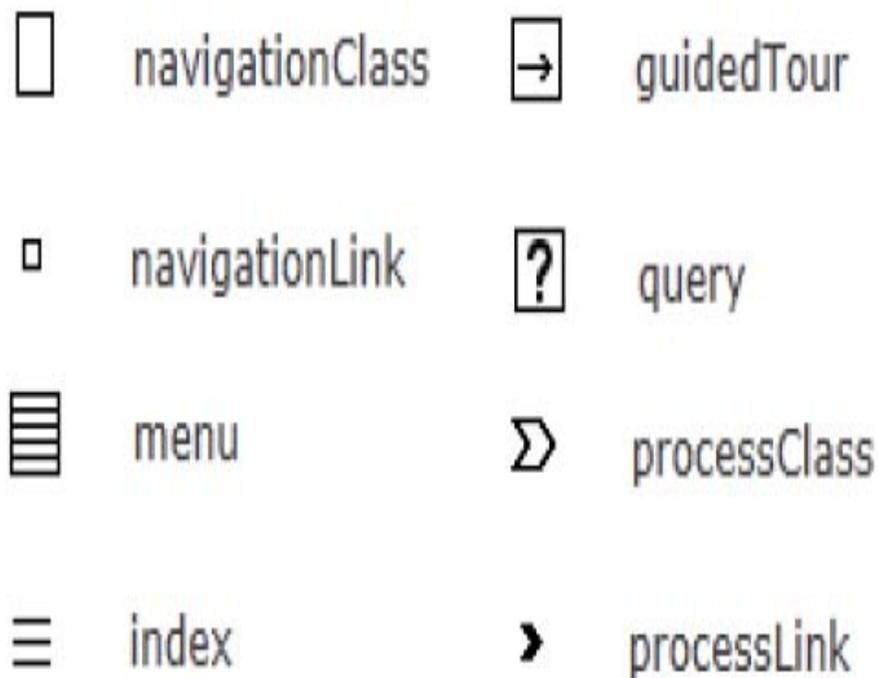


Figura 4.2.2.1 Estereotipos del diagrama de navegación

Las clases de navegación (□ «*navigationClass*») representan nodos navegables de la estructura de hipertexto; Los enlaces de navegación (□ «*navigationLink*») muestran vínculos directos entre las clases de navegación; las rutas alternativas de navegación son manejadas por menú (☰ «*menu*»). Los accesos se utilizan para llegar a múltiples instancias de una clase de navegación (≡ «*index*» o «☞ «*guidedTour*») o para seleccionar los elementos (❓ «*query*»). Las clases de procesos (Σ «*processClass*») forman los puntos de entrada y salida de los procesos de negocio en este modelado y la vinculación entre sí y a las clases de navegación se modela por enlaces de procesos (➤ «*processLink*»).

En la **Figura 4.2.2.2** se presenta el diagrama de navegación del sistema

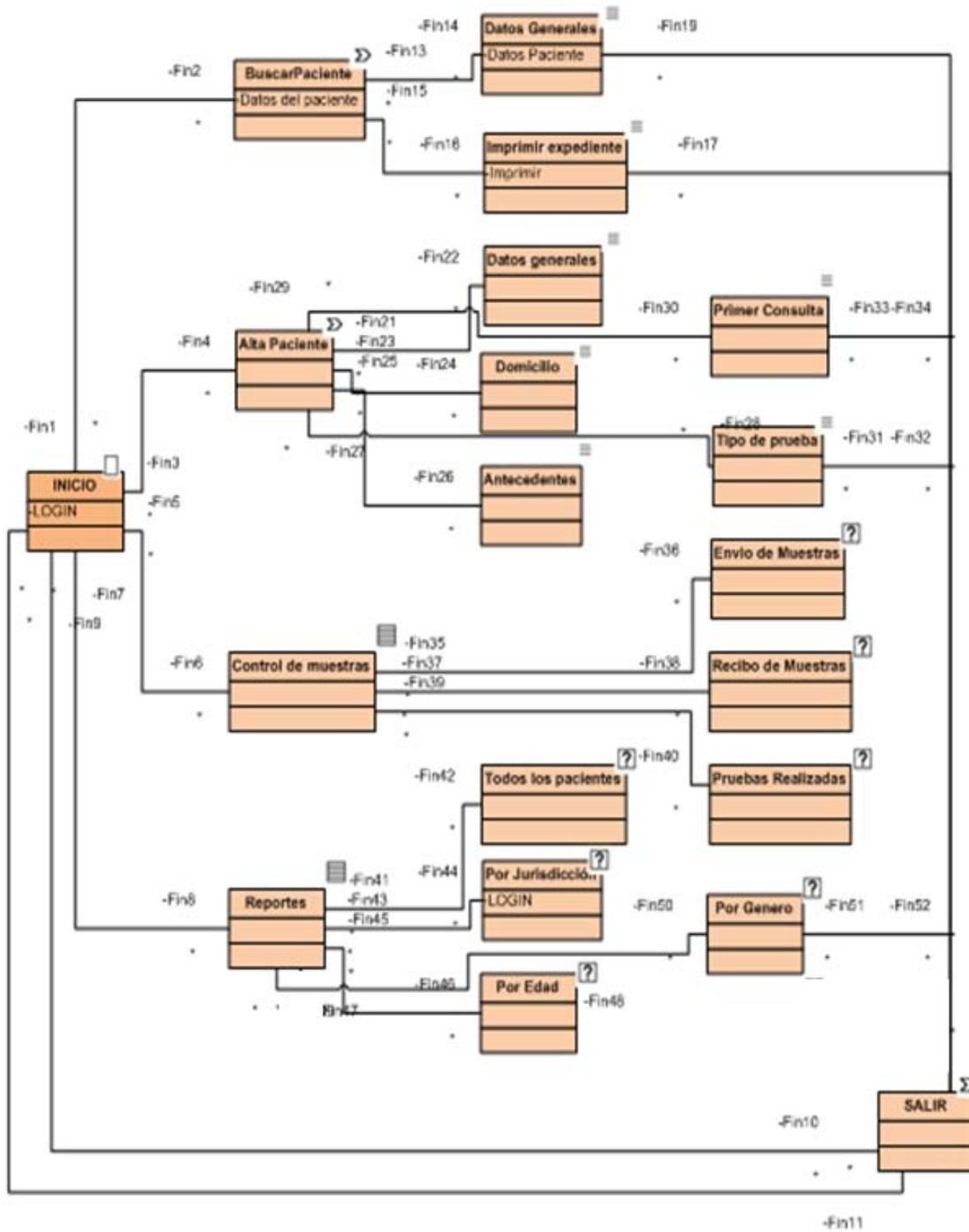


Figura 4.2.2.2 Diagrama de Navegación

4.2.3. Diagrama de presentación

CAPÍTULO 4: ANÁLISIS, DISEÑO Y CONSTRUCCIÓN

El modelo de presentación ofrece una visión abstracta de la interfaz de usuario de una aplicación Web. Se basa en el modelo de navegación y en los aspectos concretos de la interfaz de usuario (IU). Describe la estructura básica de la IU, es decir, ¿qué elementos de interfaz de usuario (por ejemplo, texto, imágenes, enlaces, formularios) se utilizan para presentar los nodos de navegación? Su ventaja es que es independiente de las técnicas actuales que se utilizan para implementar un sitio Web, lo que permite a las partes interesadas discutir la conveniencia de la presentación antes de que realmente se aplique.

Una clase de presentación está compuesta de elementos de IU como texto ( «*text*»), enlaces ( «*anchor*»), botones ( «*button*»), imágenes ( «*image*»), formularios ( «*form*») y colecciones de enlaces ( «*anchored collection*»).

En la **Figura 4.2.3.1** se detalla la página de presentación de inicio

En la **Figura 4.2.3.2** se detalla la página de Menú principal

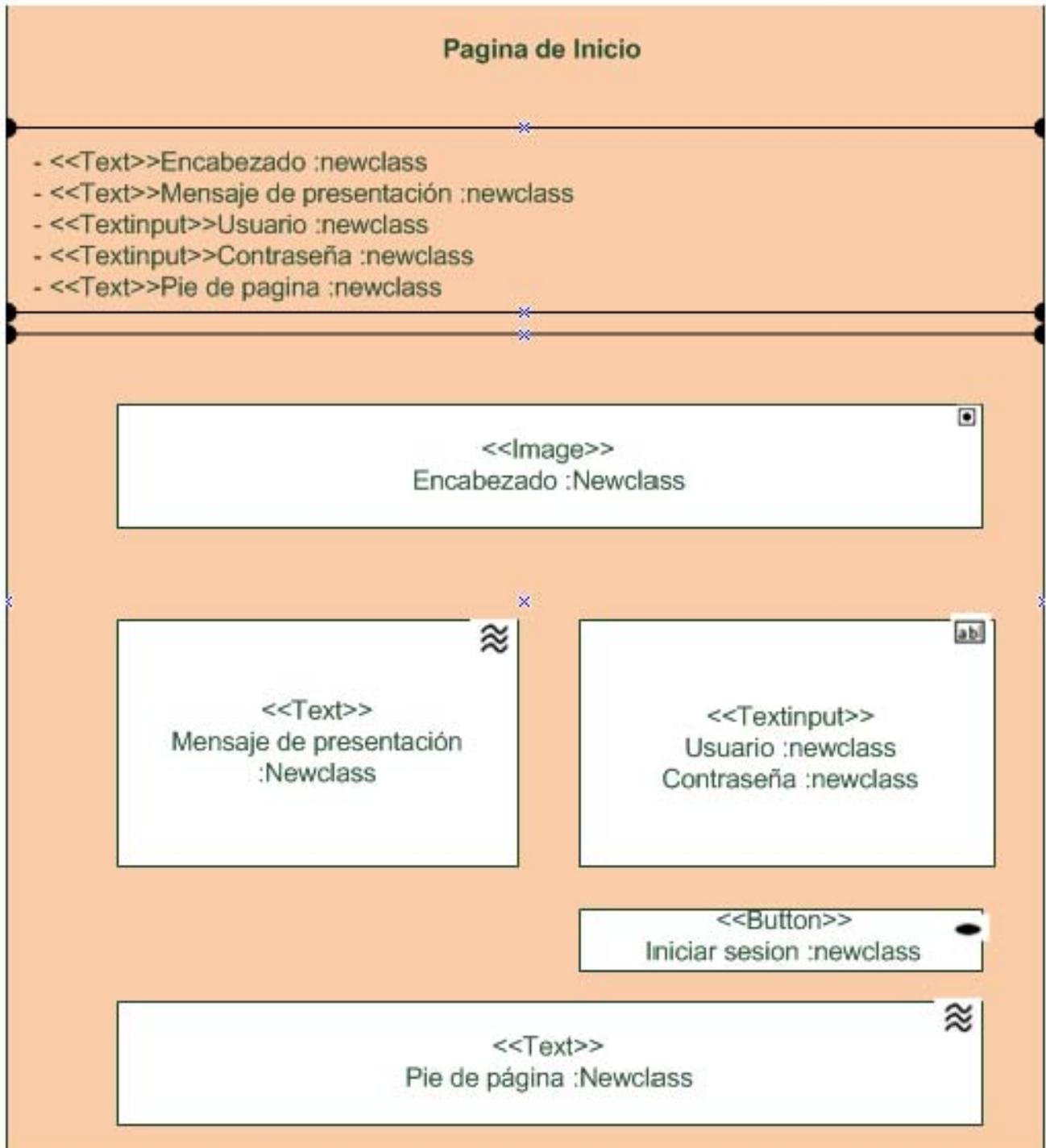


Figura 4.2.3.1 Página de inicio

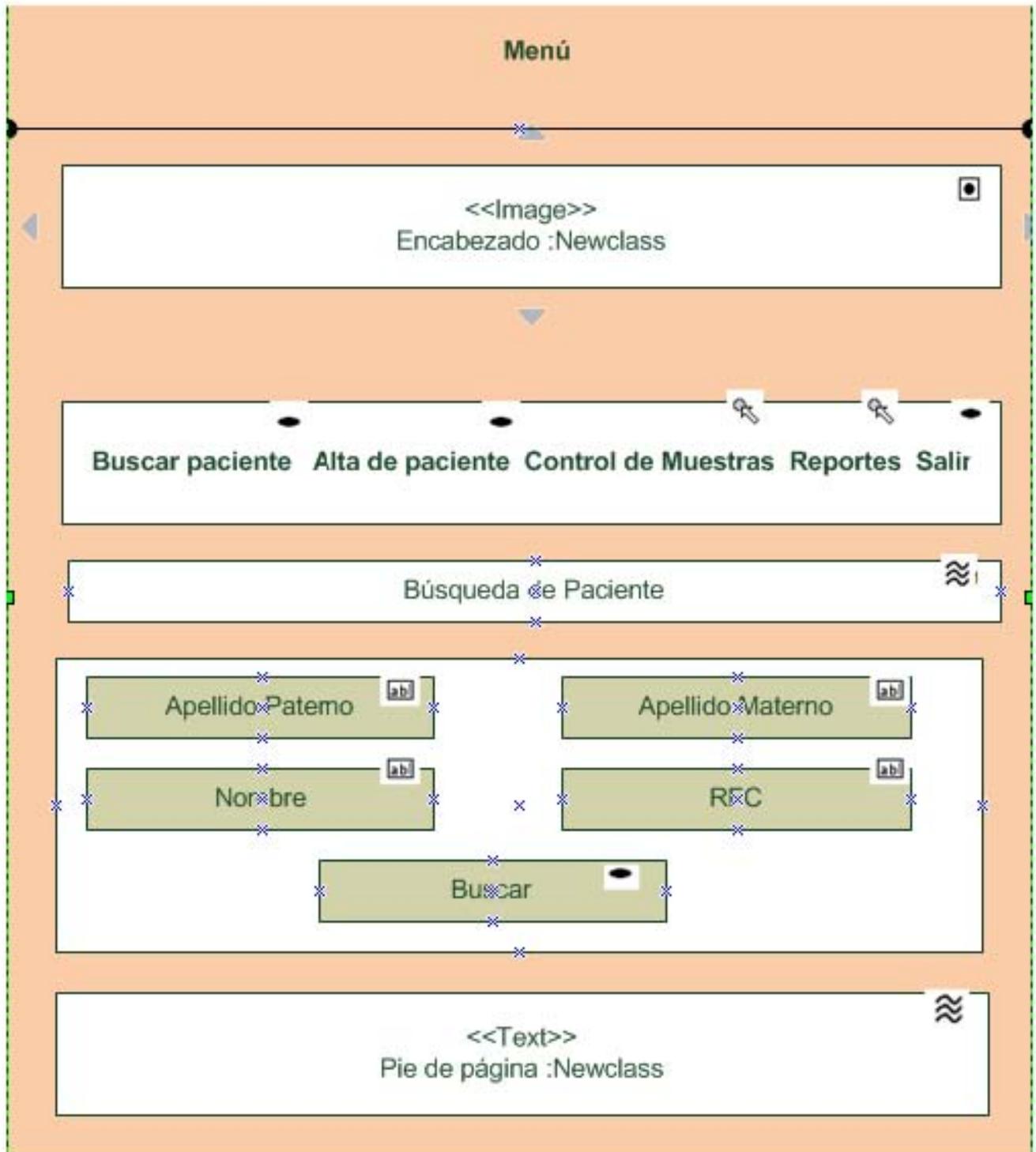


Figura 4.2.3.2 Página de Menú principal

4.2.4. Diagrama de estructura de procesos

La estructura de navegación puede ser extendida mediante clases de procesos que representan la entrada y la salida de procesos de negocio. El diagrama del proceso representa el aspecto que tienen las acciones de las clases de proceso. En este diagrama se consideran dos tipos:

- Diagrama de estructura del proceso, que describe las relaciones entre las diferentes clases de proceso.
- Diagrama de procedimientos, que especifica las actividades conectadas con cada `processClass`.

El diagrama de estructura del proceso. Es representado por un mapa de clases donde se describen las relaciones entre las diferentes clases de proceso. A continuación se podrán ver la **Figura 4.2.4.1** Estructura de proceso paciente y **Figura 4.2.4.2** Estructura de proceso Muestras.

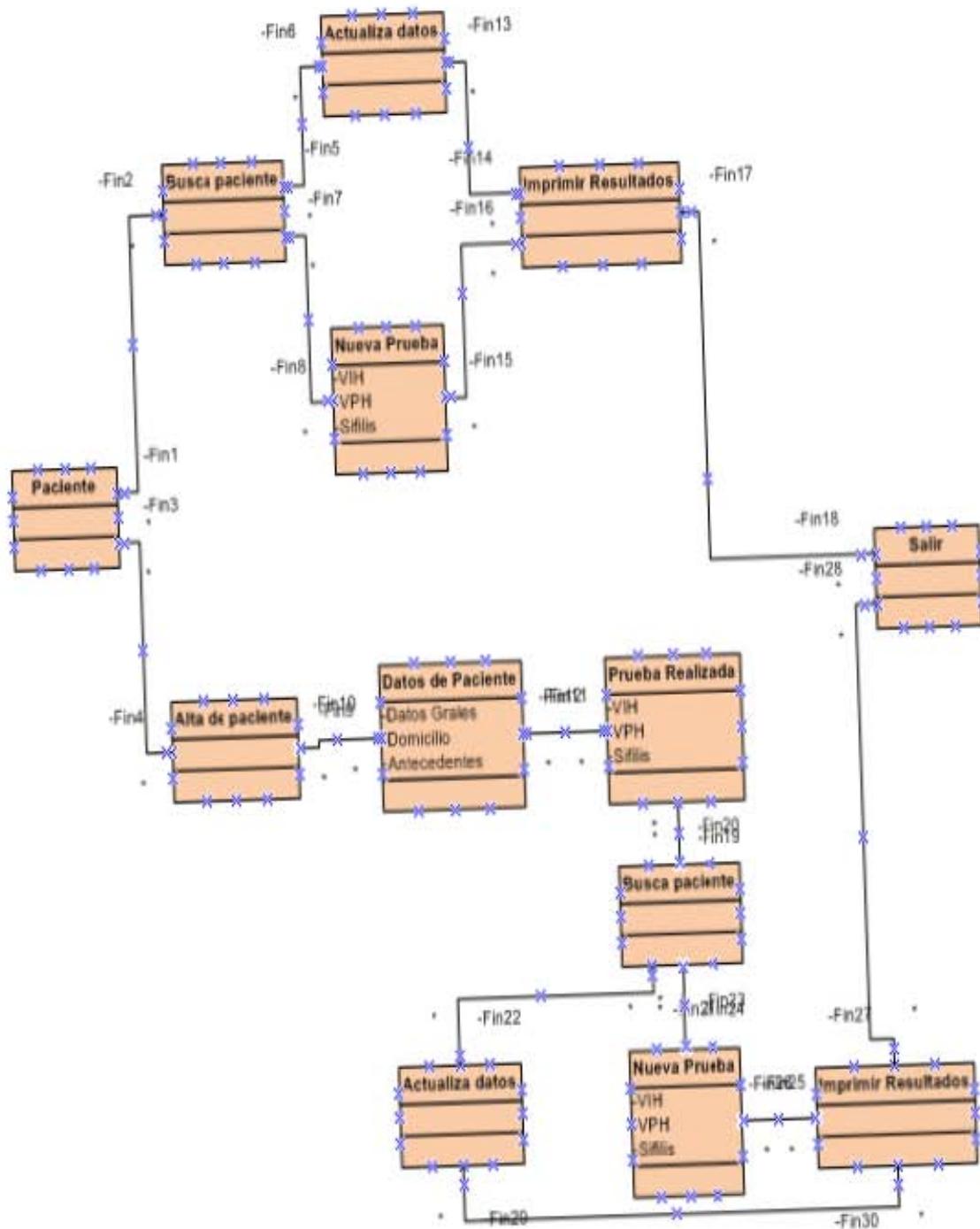


Figura 4.2.4.1 Estructura de proceso paciente

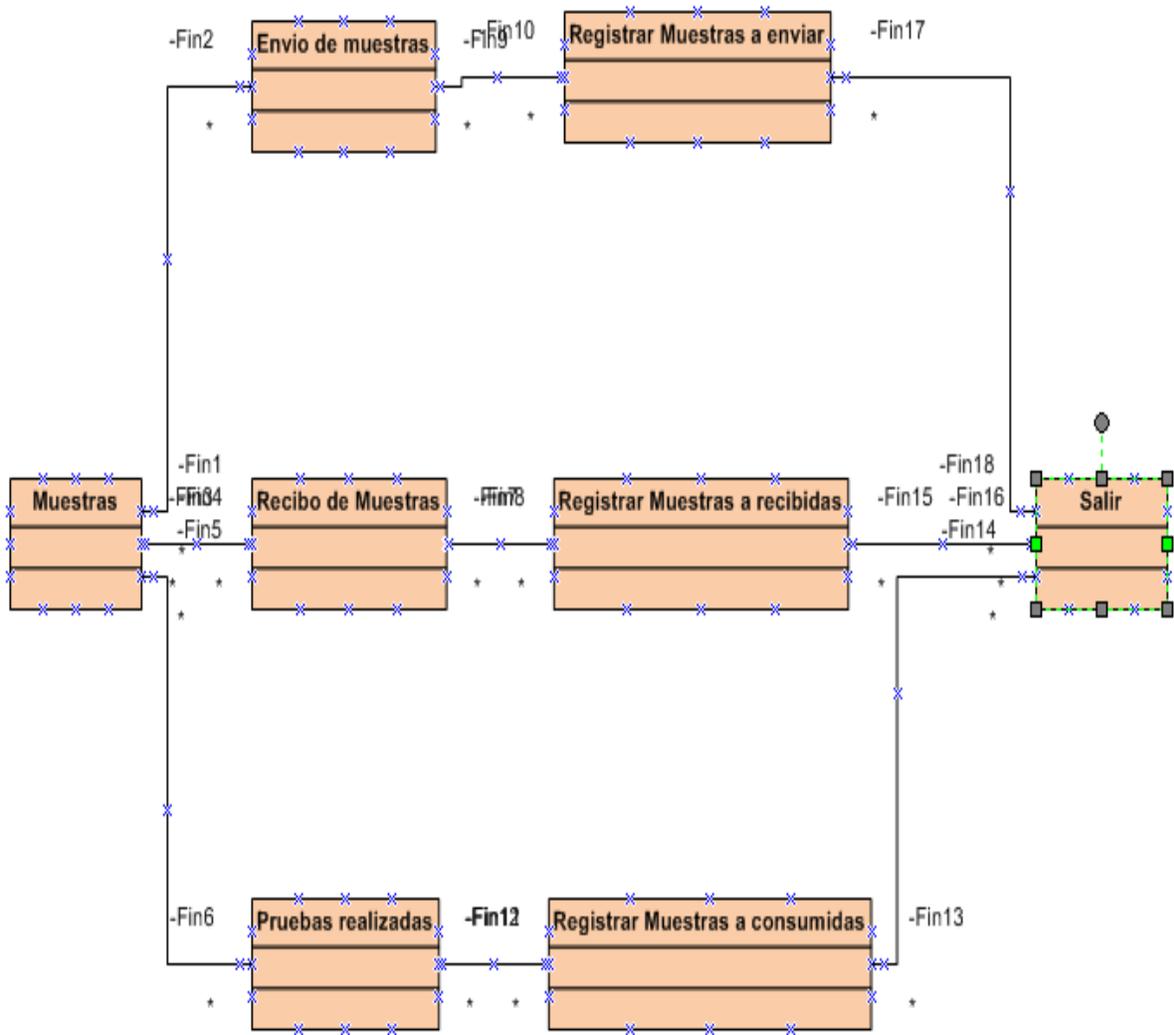


Figura 4.2.4.2 Estructura de proceso de muestras

4.2.5. Diagrama de procedimientos

Siguiendo el principio de la utilización de UML se han refinado los requisitos con los diagramas de procedimiento UML. Los diagramas de procedimiento incluyen actividades, actores responsables de estas actividades (opcional) y elementos de flujo de control. Ellos pueden ser enriquecidos con flujos de objetos que muestran objetos relevantes para la entrada o salida de esas actividades. Estos diagramas representan el flujo del proceso, describiendo el comportamiento de una clase de proceso. A continuación se presentan los diagramas de procedimiento. En la **Figura 4.2.5.1** podemos ver el procedimiento para buscar un paciente.

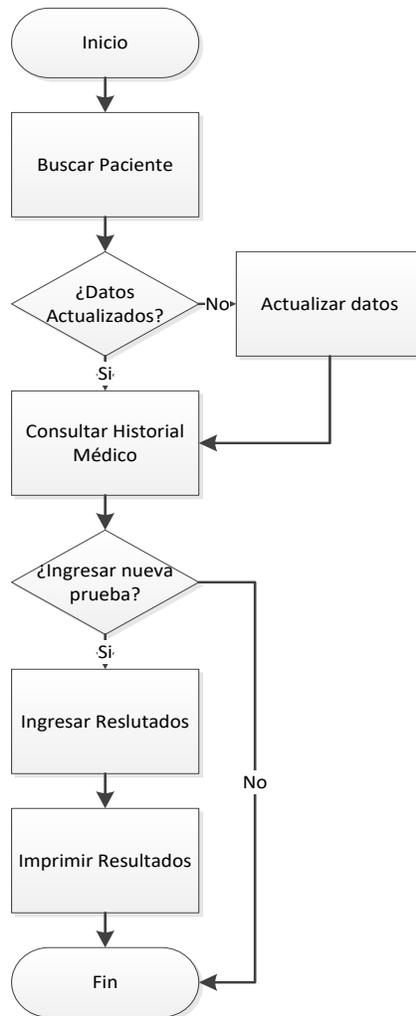


Figura 4.2.5.1 Procedimiento Buscar Paciente

En la **Figura 4.2.5.2** podemos ver el procedimiento para ingresar un nuevo paciente.

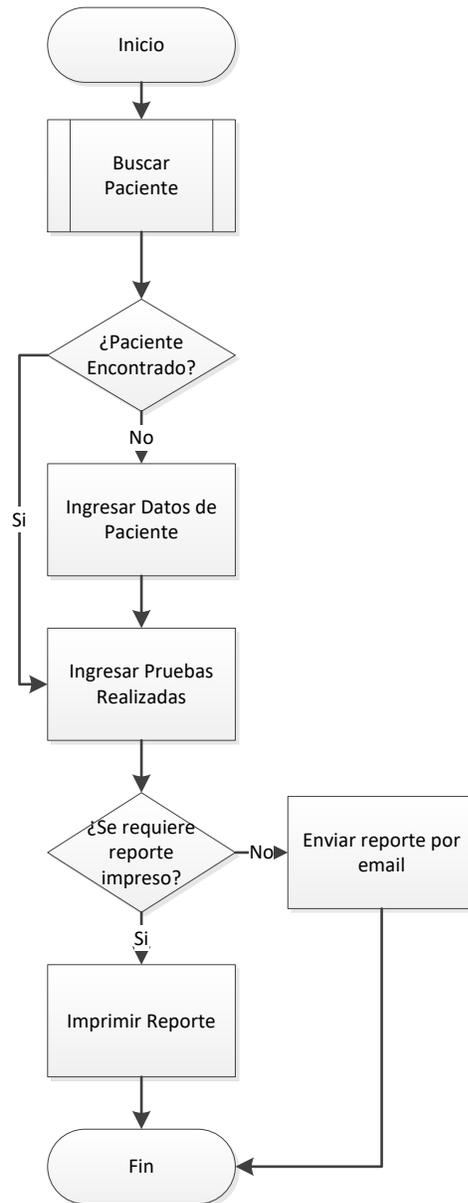


Figura 4.2.5.2 Procedimiento Ingresar Paciente

En la **Figura 4.2.5.3** podemos ver el procedimiento para el control que se lleva el inventario de las muestras.

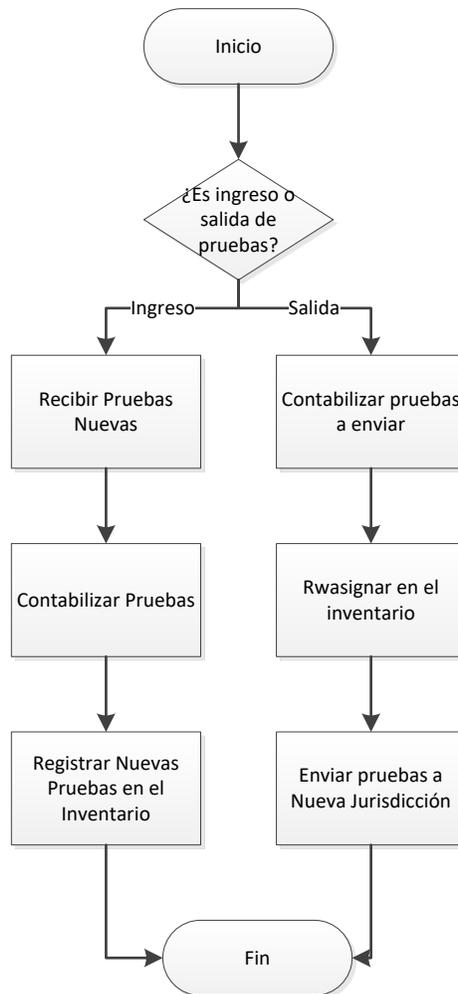


Figura 4.2.5.3 Procedimiento Control de Muestras

4.2.6. Diagrama de casos de uso

A continuación se muestran los diagramas de secuencia de los casos de uso de la sección 4.1. Ver **Figura 4.2.6.1** Caso de uso “Buscar Paciente”, **Figura 4.2.6.2** Caso de uso “Registro Paciente”, **Figura 4.2.6.3** Caso de uso “Registro Resultados de Pruebas”, **Figura 4.2.6.4** Caso de uso “Indicar Tratamiento”, **Figura 4.2.6.5** Caso de uso “Generar Reportes”, **Figura 4.2.6.6** Caso de uso “Control Recepción de Muestras”, **Figura 4.2.6.7** Caso de uso “Control Envío de Muestras”, **Figura 4.2.6.8** Caso de uso “Alta de Usuarios” y **Figura 4.2.6.9** Caso de uso Actualizar Contraseña.

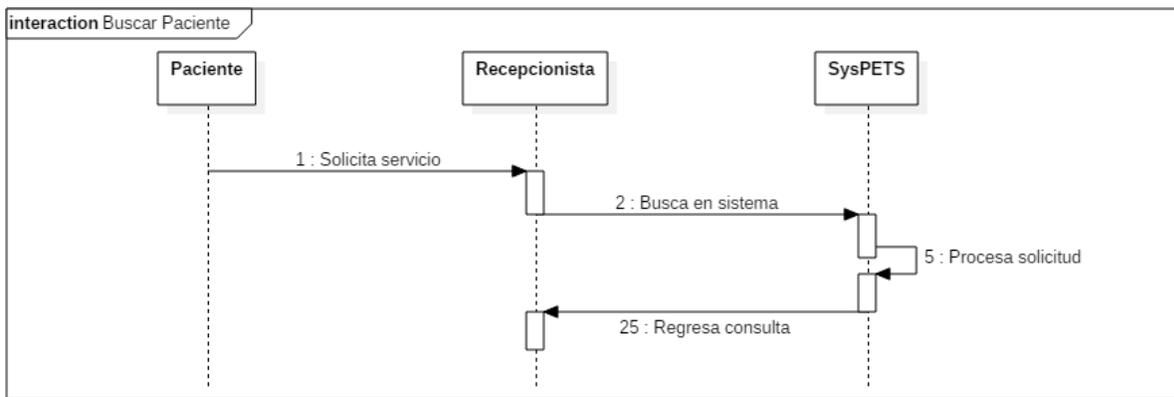


Figura 4.2.6.1 Caso de uso “Buscar Paciente”

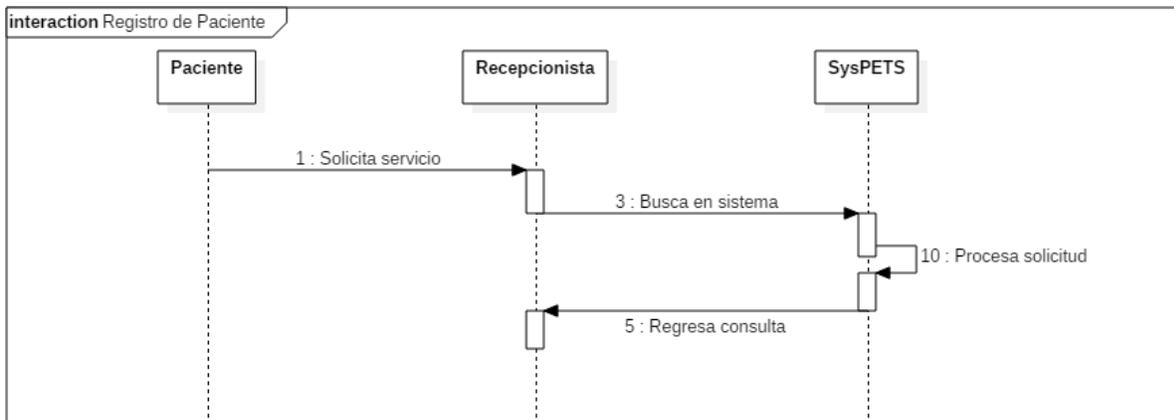


Figura 4.2.6.2: Caso de uso “Registro Paciente”

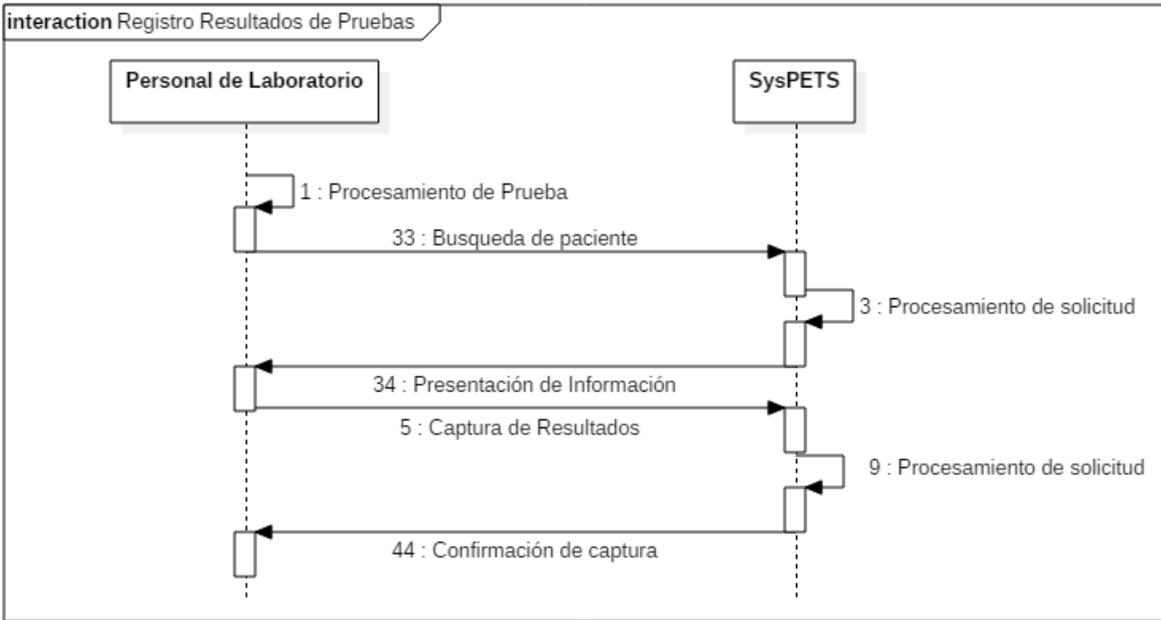


Figura 4.2.6.3: Caso de uso “Registro Resultados de Pruebas”

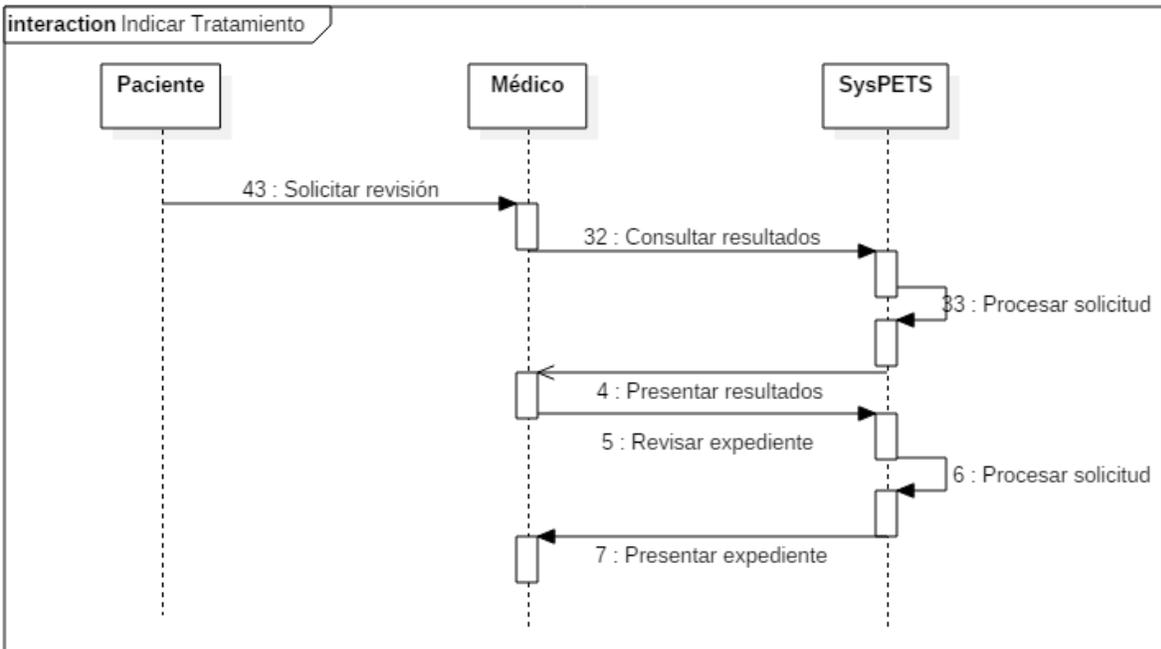


Figura 4.2.6.4: Caso de uso “Indicar Tratamiento”

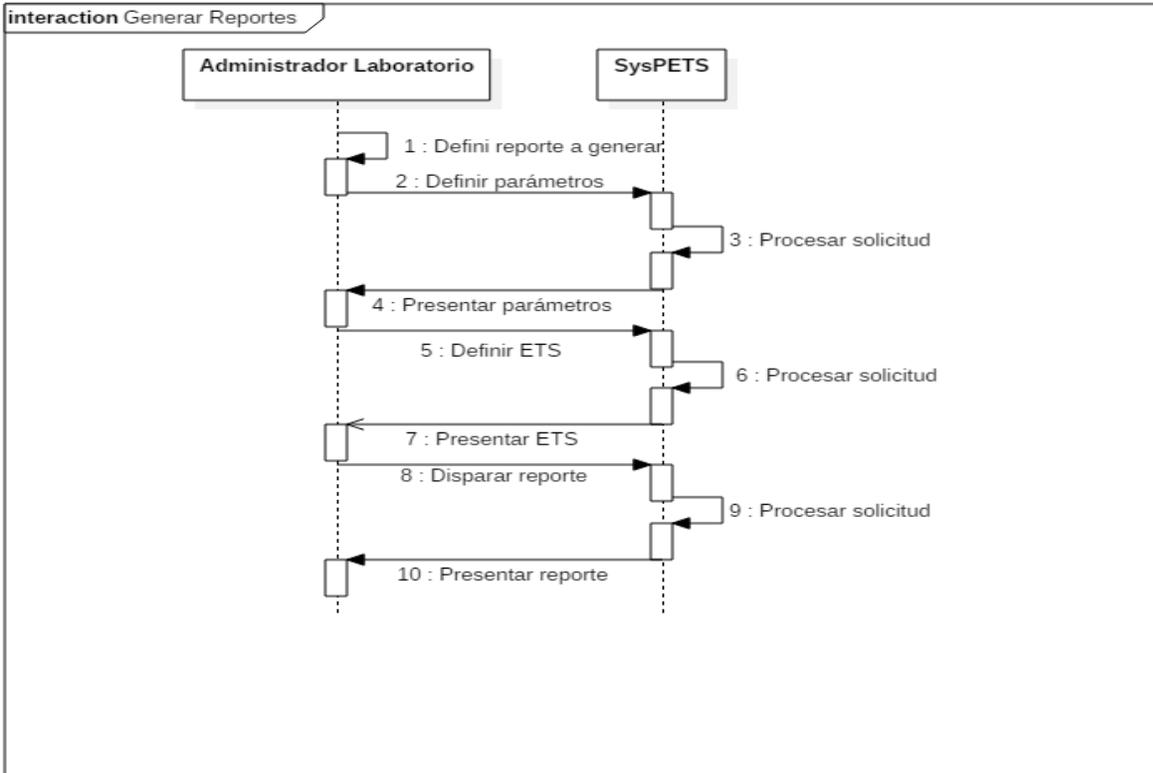


Figura 4.2.6.5: Caso de uso “Generar Reportes”

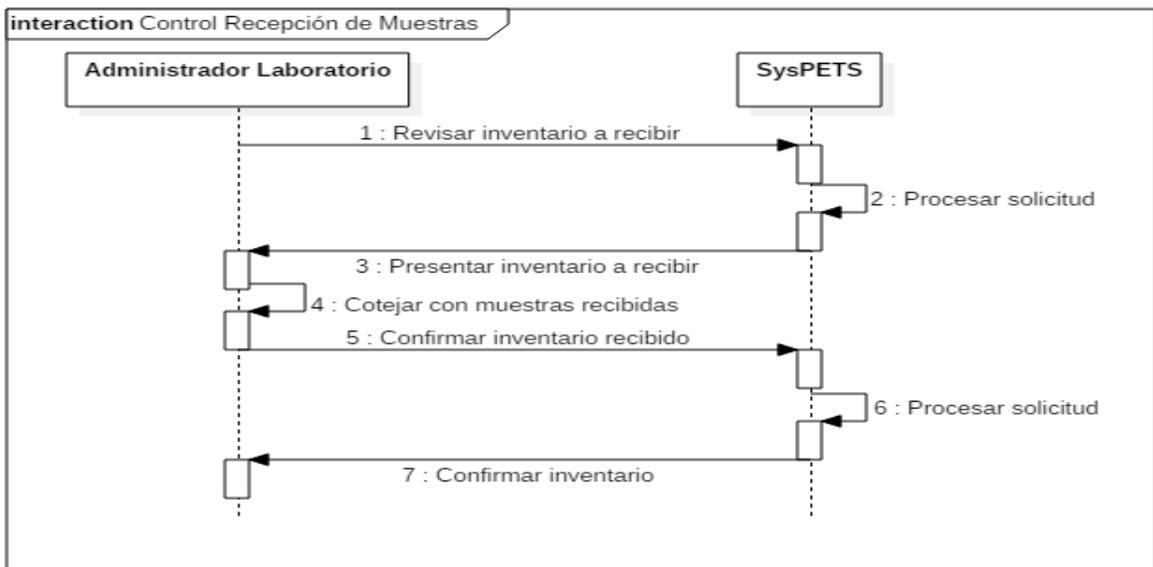


Figura 4.2.6.6: Caso de uso “Control Recepción de Muestras”

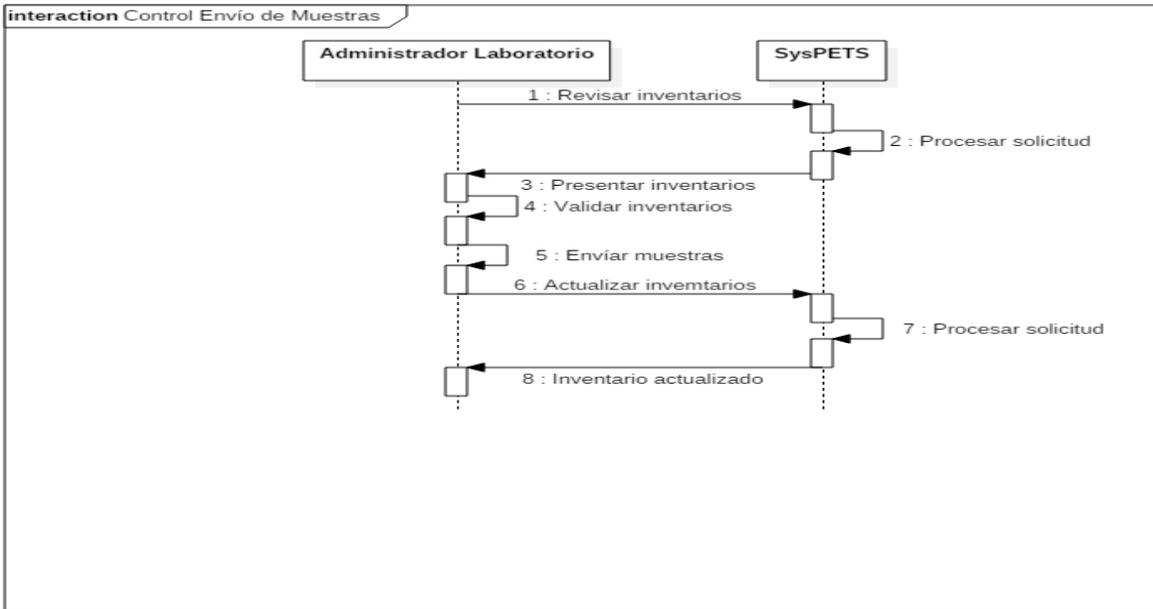


Figura 4.2.6.7: Caso de uso “Control Envío de Muestras”

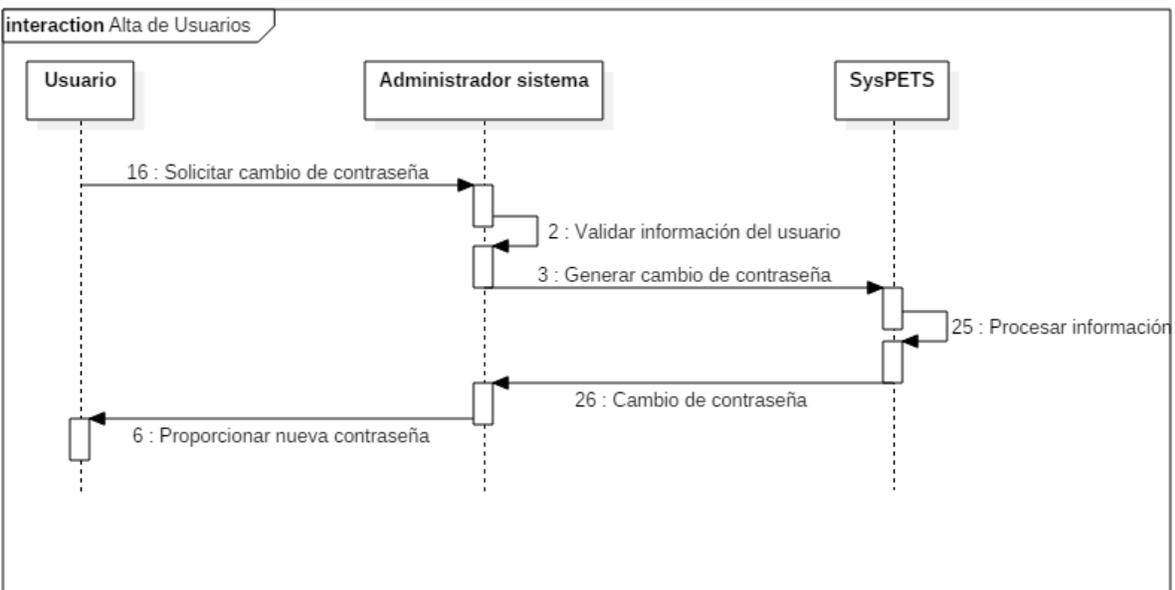


Figura 4.2.6.8: Caso de uso “Alta de Usuarios”

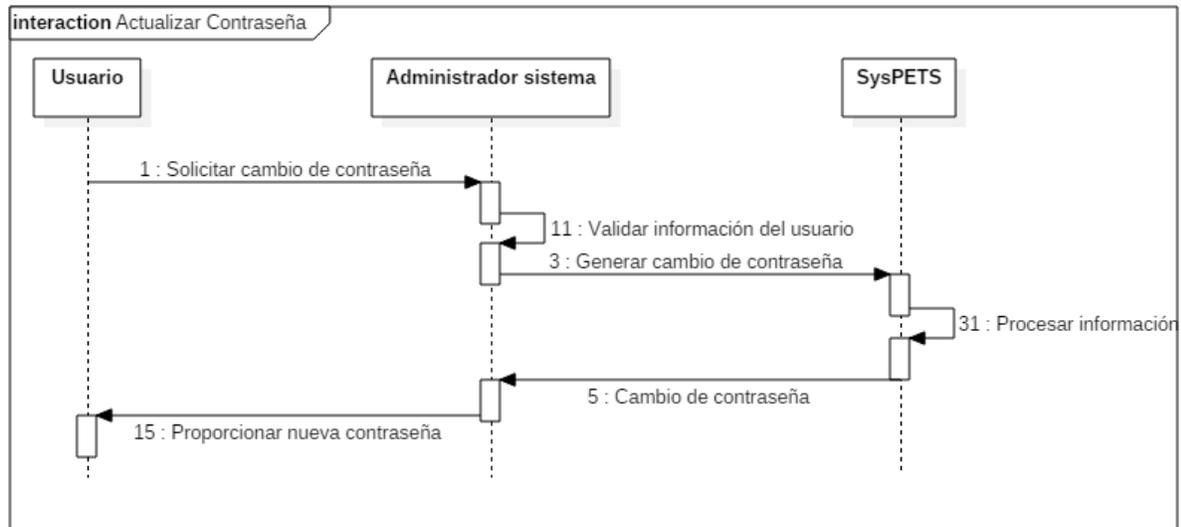


Figura 4.2.6.9 Caso de uso Actualizar Contraseña

4.2.7 Diagrama entidad relación.

El modelo entidad relación (E/R) proporciona a los desarrolladores una herramienta para representar información del mundo real a nivel conceptual.

Este modelo fue desarrollado en 1976 por Peter Chen, el cual permite describir las entidades involucradas en una base de datos, así como las relaciones y restricciones existentes entre dichas entidades y los atributos que estas poseen.

Algunas de las cualidades que posee el modelo entidad relación son:

- Notación gráfica.
- Tiene semántica clara.
- Fácil de entender aún por no especialistas

A continuación se describen los elementos que componen el modelo entidad relación.

Entidad: Se define como un elemento con existencia independiente de otros elementos

Atributo: son las características o propiedades que definen la entidad, estas poseen

valores específicos, por ejemplo: si consideramos perro como una entidad, esta posee atributos como lo son, color de ojos, color de pelo, tamaño, edad, género, etc

Existen diferentes tipos de atributos

- Atributos simples o atómicos: son atributos no divisibles.
- Atributos compuestos: son atributos que se pueden dividir en sus componentes, pudiendo formar jerarquías.
- Atributos monovaluados: son atributos que tienen un solo valor para una entidad en particular.
- Atributos multivaluados: son atributos que tienen límites inferior y superior en el número de valores para una entidad.
- Atributos derivados: atributos generados a partir de otros
- Valores nulos: atributos que pueden o no tener un valor
- Atributos complejos: son atributos compuestos o multivaluados anidados de una manera arbitraria (lista, conjuntos).

Cada uno de los atributos de una entidad posee un dominio, el que corresponde al tipo del atributo.

Para todo conjunto de valores de una entidad, debe existir un atributo o combinación de atributos, que identifique a cada entidad en forma única. Este atributo o combinación de atributos se denomina llave (primaria).

Relación: esta se puede definir como una asociación entre entidades.

Las relaciones pueden como se muestra en la **Figura 4.2.7.1**:

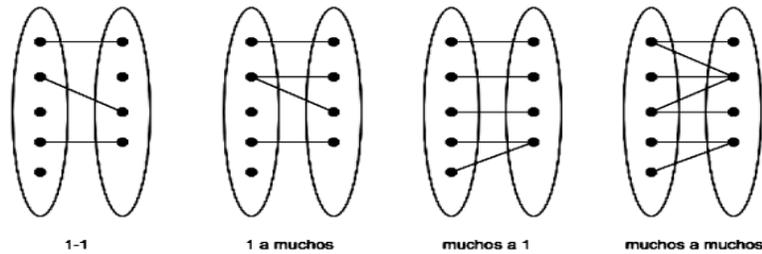


Figura 4.2.7.1- Relación

Relación 1:1: esta relación se tiene cuando un elemento de una entidad se relaciona con uno y solo uno de otra entidad, por ejemplo, la relación que tiene un presidente con el país gobernado.

Relación 1:M: esta relación se tiene cuando un elemento de una entidad se relaciona con uno o más elementos de otra entidad, pero el elemento de la segunda entidad solo tiene relación con un elemento de la primera entidad, por ejemplo, la relación que existe entre un árbol y sus frutos.

Relación M:M: esta relación se tiene cuando pueden existir relaciones múltiples entre 2 entidades dadas, por ejemplo, estudiantes y las escuelas en las que estudian.

A continuación se muestra el diagrama entidad relación del sistema **Figura 4.2.7.2:**

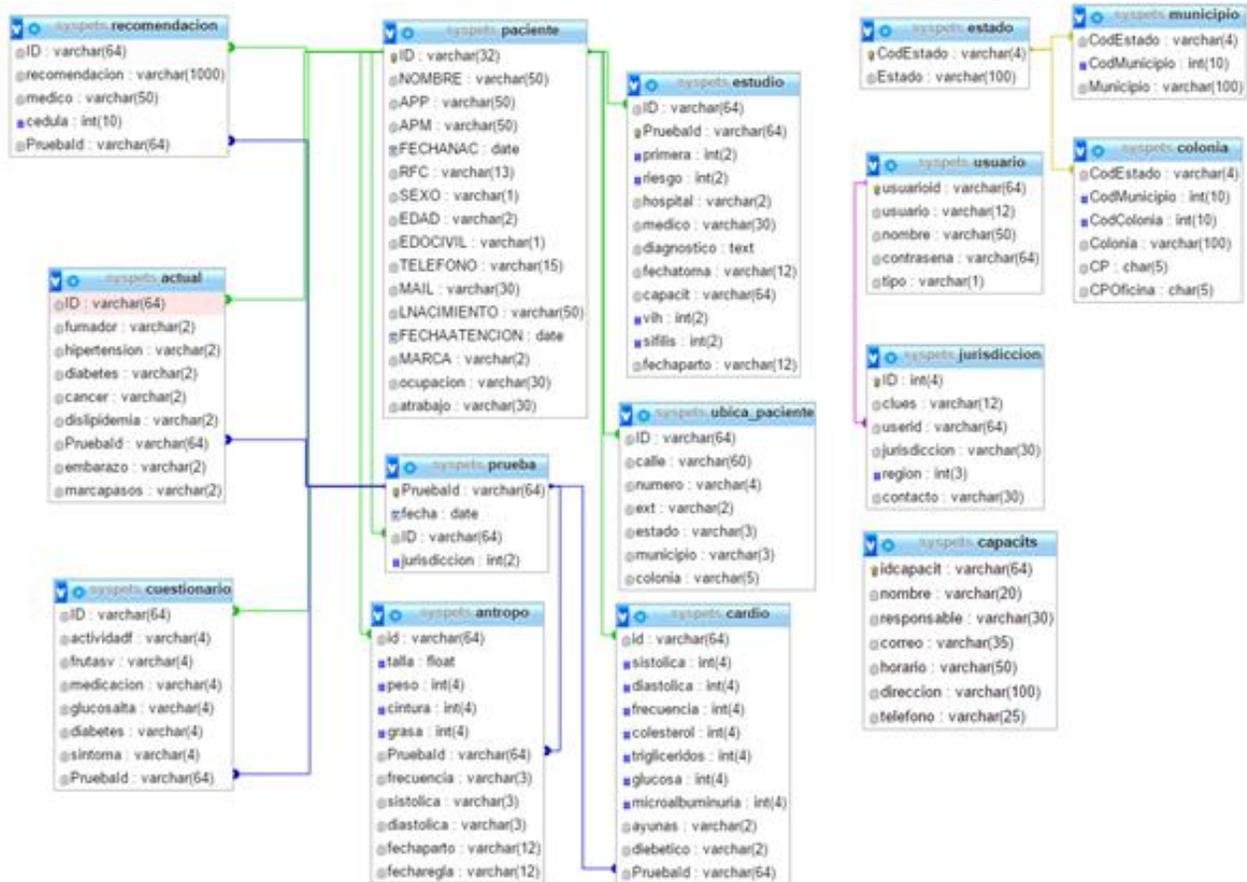


Figura 4.2.7.1 Diagrama Entidad – Relación

4.2.8. Normalización de la base de datos

Se le llama normalización al proceso mediante el cual se transforman datos complejos a estructuras más simples, que adicional a ser más estables, su mantenimiento es más sencillo. Otra manera de definir la normalización es el conjunto de reglas que sirven para ayudar a los diseñadores de bases de datos a desarrollar una estructura que reduzca los problemas de lógica.

La normalización también ayuda a facilitar la comprensión. Los seres humanos tenemos la tendencia de simplificar las cosas al máximo. Las guías que la

CAPÍTULO 4: ANÁLISIS, DISEÑO Y CONSTRUCCIÓN

normalización es el marco de referencia para simplificar una estructura de datos compleja.

Otra mejora de la normalización de base de datos es el consumo de espacio, ya que una base de datos normalizada ocupa menos espacio en disco que la que no lo está, derivado de que hay menos repetición de datos.

Actualmente existen 5 formas normales aplicables a bases de datos, cabe destacar que no es necesario aplicar todas las formas normales, solo las que se requieran por la complejidad de la base de datos a procesar. Para el caso de este proyecto se aplicará hasta la tercer forma normal, sin embargo se describirán las 5 formas normales.

En la **Figura 4.2.8.1** se muestra la tabla Paciente y en la **Figura 4.2.8.2** la tabla Colonia sin normalizar, posteriormente se irá aplicando la normalización.

ID	NOMBRE	APP	APM	FECHANAC	RFC	SEXO	EDAD
44dae411179cd2ee00531d2b77eb1aab	JOSE ANDRES	SALES	SALAZAR	03/10/1985	SASA851003	M	28
707a0aa3787c83d7724d6b6d4644ba8a	HUGO ALBERTA	HORMIGO	SUAREZ	11/11/1979	HOSH791111	M	33
5669f84a3ff6a97463d9ae4b1b85de57	ALFREDO	LOPEZ	GASCA	07/05/2013	LOGA130507	M	25
7e4c9e7197322ede11c48771f4263e15	ALFREDO	ARROYO	ALAZANES	09/05/2013	AOAA130509	M	19
21a18e4993f724ad5136abdacde4ad4c	ERICKA	MARTINEZ	JIMENEZ	18/09/1980	MAJE800918	F	33

EDOCIVIL	TELEFONO	MAIL	LNACIMIENTO	FECHAATENCION	MARCA	ocupacion	atrabajo
0	5586770398	jos.ass@gmail.com		08/10/2013		INGENIERO	UNAM
0	5593285462	halbertos@gmail.com		08/10/2013		CONTADOR	FARMACEUTICA
0	5524278288	alopezg@hotmail.com	Tepic	08/05/2013		COMERCIANTE	COMERCIANTE
0	5508979459	alfrarroyo@gmail.com	Puente de Ixtla	08/05/2013		EMPLEADO	EMPRESA
1	5589942872	eramirezjimenez@yahoo.com		21/10/2013		INGENIERO	IPN

Figura. 4.2.8.1 Tabla Paciente sin normalizar

CAPÍTULO 4: ANÁLISIS, DISEÑO Y CONSTRUCCIÓN

CodEstado	CodMunicipio	CodColonia	Colonia	CP	CPOficina
AGS		1	1 Zona Centro	20000	20001
AGS		1	2 Delegacion de La Secretaria de Comercio y Fomento	20008	20001
AGS		1	3 Palacio de Gobierno Del Estado de Aguascalientes	20009	20001
AGS		1	4 Colinas Del Rio	20010	20001
AGS		1	5 Enrique Olivares Santana	20010	20001
AGS		1	6 Las Brisas	20010	20001
AGS		1	7 Ramon Romo Franco	20010	20001
AGS		1	8 San Cayetano	20010	20001
AGS		1	9 Colinas de San Ignacio	20016	20001
AGS		1	10 Fundicion	20016	20001
AGS		1	11 Fundicion II	20016	20001
AGS		1	12 Los Sauces	20016	20001
AGS		1	13 Linea de Fuego	20018	20001
AGS		1	14 Prado de los Sauces	20019	20001

Figura. 4.2.8.2 Tabla Colonia sin normalizar

Primera Forma Normal

Por definición, una tabla se encuentra en primera forma normal si y solo si todas las columnas de todas las tablas contienen solo valores atómicos. Establece que las columnas repetidas deben eliminarse y colocarse en tablas aparte. Además dicha tabla debe tener una llave primaria.

Una tabla debe tener una llave primaria que identifique a cada uno de los registros de manera única. En la **Figura 4.2.8.3** se observa la tabla Paciente con la asignación a nivel base de datos de la llave primaria.

ID (PRIMARY KEY)	NOMBRE	APP	APM	FECHANAC	RFC	SEXO	EDAD
44dae411179cd2ee00531d2b77eb1aab	JOSE ANDRES	SALES	SALAZAR	03/10/1985	SASA851003	M	28
707a0aa3787c83d7724d6b6d4644ba8a	HUGO ALBERTA	HORMIGO	SUAREZ	11/11/1979	HOSH791111	M	33
5669f84a3ff6a97463d9ae4b1b85de57	ALFREDO	LOPEZ	GASCA	07/05/2013	LOGA130507	M	25
7e4c9e7197322ede11c48771f4263e15	ALFREDO	ARROYO	ALAZANES	09/05/2013	AOAA130509	M	19
21a18e4993f724ad5136abdacde4ad4c	ERICKA	MARTINEZ	JIMENEZ	18/09/1980	MAJE800918	F	33

EDOCIVIL	TELEFONO	MAIL	LNACIMIENTO	FECHAATENCION	MARCA	ocupacion	atrabajo
0	5586770398	jos.ass@gmail.com		08/10/2013		INGENIERO	UNAM
0	5593285462	halbertos@gmail.com		08/10/2013		CONTADOR	FARMACEUTICA
0	5524278288	alopezg@hotmail.com	Tepic	08/05/2013		COMERCIANTE	COMERCIANTE
0	5508979459	alfrarroyo@gmail.com	Puente de Ixtla	08/05/2013		EMPLEADO	EMPRESA
1	5589942872	eramirezjimenez@yahoo.com		21/10/2013		INGENIERO	IPN

Figura. 4.2.8.3 Tabla Paciente con ID asignado como *Primary Key*

En la **Figura 4.2.8.4** se genera la asignación del ID de colonia como llave primaria

CodEstado	CodMunicipio	CodColonia (Primary Key)	Colonia	CP	CPOficina
AGS		1	1 Zona Centro	20000	20001
AGS		1	2 Delegacion de La Secretaria de Comercio y Fomento	20008	20001
AGS		1	3 Palacio de Gobierno Del Estado de Aguascalientes	20009	20001
AGS		1	4 Colinas Del Rio	20010	20001
AGS		1	5 Enrique Olivares Santana	20010	20001
AGS		1	6 Las Brisas	20010	20001
AGS		1	7 Ramon Romo Franco	20010	20001
AGS		1	8 San Cayetano	20010	20001
AGS		1	9 Colinas de San Ignacio	20016	20001
AGS		1	10 Fundicion	20016	20001
AGS		1	11 Fundicion II	20016	20001

Figura. 4.2.8.4 Tabla Colonia en primera forma normal

Segunda Forma Normal

Establece que todas las dependencias parciales se deben eliminar y separar dentro de sus propias tablas. Entiéndase como dependencia parcial aquellos datos que no dependen de la llave primaria de la tabla para identificarlos.

Alcanzada la segunda forma normal, se controlan la mayoría de los problemas de lógica y es posible insertar un registro sin un exceso de datos en la mayoría de las tablas.

La tabla Paciente **Figura 4.2.8.3** ya se encuentra en 2FN, ya que las columnas de la misma son dependientes entre sí, por lo tanto la tabla no se verá modificada.

La tabla Colonia **Figura 4.2.8.4** no se encuentra en segunda forma normal, ya que existe una dependencia parcial con los códigos postales, las cuales se separarán en 2 tablas: Colonia y Códigos Postales (ver **Figura 4.2.8.5**).

CodEstado	CodMunicipio	CodColonia (Primary Key)	Colonia
AGS		1	1 Zona Centro
AGS		1	2 Delegacion de La Secretaria de Comercio y Fomento
AGS		1	3 Palacio de Gobierno Del Estado de Aguascalientes
AGS		1	4 Colinas Del Rio
AGS		1	5 Enrique Olivares Santana
AGS		1	6 Las Brisas
AGS		1	7 Ramon Romo Franco
AGS		1	8 San Cayetano
AGS		1	9 Colinas de San Ignacio
AGS		1	10 Fundicion
AGS		1	11 Fundicion II
AGS		1	12 Los Sauces
AGS		1	13 Linea de Fuego
AGS		1	14 Prado de los Sauces
AGS		1	15 Buenos Aires
AGS		1	16 Circunvalacion 1a Seccion

CodColonia	CP	CPOficina
1	20000	20001
2	20008	20001
3	20009	20001
4	20010	20001
5	20010	20001
6	20010	20001
7	20010	20001
8	20010	20001

Figura. 4.2.8.5 Tabla Colonia en segunda forma normal y tabla de Códigos Postales

Tercera Forma Normal

Una base de datos de encuentra en tercera forma normal si todas las columnas que no son llave son funcionalmente dependientes por completo de la llave primaria y no hay dependencias transitivas. Una dependencia transitiva es aquella en la cual las columnas que no son llave son dependientes de otras columnas que tampoco son llave.

La tercera forma normal previene errores de lógica cuando se insertan o borran registros. Cada columna en una tabla está identificada de manera única por la llave primaria, y no deben existir datos repetidos. Esto provee un esquema limpio y que es fácil de trabajar y expandir.

La tabla se encuentra en esta forma si está 2FN y todas las columnas que no son llave son funcionalmente dependientes por completo de la llave primaria y no hay dependencias transitivas. La tabla Paciente (ver figura 4.2.8.3) se encuentra en 3FN.

La tabla Códigos Postales no se encuentra en 3FN, por lo que se genera una tabla de Catálogos de Códigos Postales asociada a la tabla Códigos Postales (**Figura 4.2.8.6**)

CAPÍTULO 4: ANÁLISIS, DISEÑO Y CONSTRUCCIÓN

CodColonia (Primary Key)	CodCP	CodCPofna
1	1095	1096
2	1097	1096
3	1098	1096
4	1099	1096
5	1099	1096
6	1099	1096
7	1099	1096
8	1099	1096
9	1100	1096

CP	CodCP
16870	1087
16880	1088
16888	1089
16889	1090
16900	1091
16908	1092
16909	1093
16910	1094
20000	1095
20001	1096
20008	1097
20009	1098

Figura. 4.2.8.6 Tabla Código Postal en tercera forma normal y tabla de Catálogo de Códigos Postales

La tabla Paciente no se encuentra en 3FN, por lo que se genera una tabla de Ocupación asociada a la tabla Paciente (**Figura 4.2.8.7**)

ID (PRIMARY KEY)	NOMBRE	APP	APM	FECHANAC	RFC	SEXO	EDAD
44dae411179cd2ee00531d2b77eb1aab	JOSE ANDRES	SALES	SALAZAR	03/10/1985	SASA851003	M	28
707a0aa3787c83d7724d6b6d4644ba8a	HUGO ALBERTA	HORMIGO	SUAREZ	11/11/1979	HOSH791111	M	33
5669f84a3ff6a97463d9ae4b1b85de57	ALFREDO	LOPEZ	GASCA	07/05/2013	LOGA130507	M	25
7e4c9e7197322ede11c48771f4263e15	ALFREDO	ARROYO	ALAZANES	09/05/2013	AOAA130509	M	19
21a18e4993f724ad5136abdacde4ad4c	ERICKA	MARTINEZ	JIMENEZ	18/09/1980	MAJE800918	F	33

EDOCIVIL	TELEFONO	MAIL	LNACIMIENTO	FECHAATENCION	MARCA	ocupacion	atrabajo
0	5586770398	jos.ass@gmail.com		08/10/2013		ING	UNAM
0	5593285462	halbertos@gmail.com		08/10/2013		CON	FARMACEUTICA
0	5524278288	alopezg@hotmail.com	Tepic	08/05/2013		COM	COMERCIANTE
0	5508979459	alfrarroyo@gmail.com	Puente de Ixtla	08/05/2013		EMP	EMPRESA
1	5589942872	eramirezjimenez@yahoo.com		21/10/2013		ING	IPN

ID OCUPACION	ocupacion
ING	INGENIERO
CON	CONTADOR
COM	COMERCIANTE
EMP	EMPLEADO

Figura. 4.2.8.6 Tabla Paciente en tercera forma normal y tabla de Ocupación

Cuarta Forma Normal

La cuarta forma normal es el siguiente nivel de normalización después de la forma normal de Boyce-Codd (BCNF).

Una tabla está en 4NF si y solo si está en Tercera forma normal o en BCNF (Cualquiera de ambas) y no posee dependencias multi-valuadas no triviales. La definición de la 4NF confía en la noción de una dependencia multi-valuada. Una tabla con una dependencia multi-valuada es una donde la existencia de dos o más relaciones independientes muchos a muchos causa redundancia; y es esta redundancia la que es suprimida por la cuarta forma normal.

La forma normal de Boyce-Codd se da si una tabla está en tercera forma normal y además todo determinante es una clave candidata. En base de datos un atributo determinante es un atributo del que depende funcionalmente de manera completa algún otro atributo.

Quinta Forma Normal

Una tabla está en Quinta Forma Normal o Forma Normal de Proyección-Unión si está en cuarta forma normal y las únicas dependencias que existen son las dependencias de unión de una tabla con sus proyecciones relacionándose entre las distintas proyecciones mediante la clave primaria o cualquier clave alternativa. La 5FN se emplea cuando en una misma tabla tenemos mucha información redundante, con pocos atributos o cuando una tabla posee una gran cantidad de atributo

4.3. Desarrollo e implementación de la base de datos

Para empezar a construir la base de datos se necesita realizar la descarga e instalación del manejador de base de datos MySQL sobre una plataforma Microsoft Windows, que es la plataforma con la que cuentan los usuarios y sobre la cual se trabajará.

La preparación del ambiente para la instalación se describe a continuación:

- Una PC con sistema operativo Microsoft Windows XP o posterior.
- Se requiere que se acceda a la PC con un perfil con credenciales de administrador, una vez hecha la instalación no es necesario ejecutar la herramienta como administrador de nuevo.
- Paquete de instalación “MySQL Community Server” Versión 5.7.

4.3.2. Administración de la base de datos

Una vez que se instaló satisfactoriamente el Servidor MySQL en el servidor, se procede a la creación de la Base de Datos del sistema.

4.3.2.1. MySQL Administrator

La instalación del MySQL nos habrá dejado una herramienta que se llama *MySQL Administrator*, la cual tiene la tarea de permitir la administración de los servidores de Base de Datos de MySQL, como se muestra en la **Figura 4.3.2.1.1**.

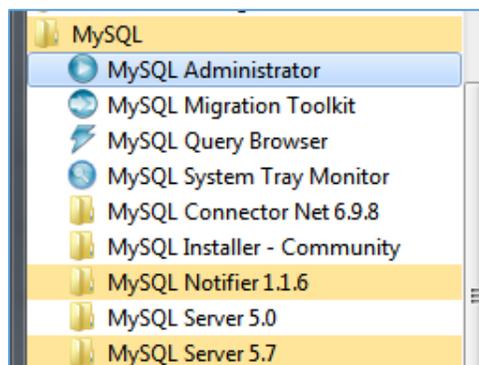


Figura 4.3.2.1.1 Resultado de la instalación de MySQL.

Como se había mencionado con anterioridad, MySQL nos ofrece esta herramienta que es muy intuitiva para su fácil administración y acceso. Esta herramienta nos va a permitir:

- Configuración de inicio de los servidores.
- Detener e iniciar los servidores.
- Monitorear las conexiones que hay hacia los servidores.
- Administración de usuarios de Base de Datos.
- Monitorear el estado de los servidores.
- Gestión de copias de seguridad.
- Gestión de aplicación de puntos de restauración.
- Visualización de Catálogos de información.

Ingresamos a la herramienta “*MySQL Administrator*” ingresando las credenciales del usuario *root* para tener permisos sobre el servidor como lo muestra la **Figura 4.3.2.1.2**.

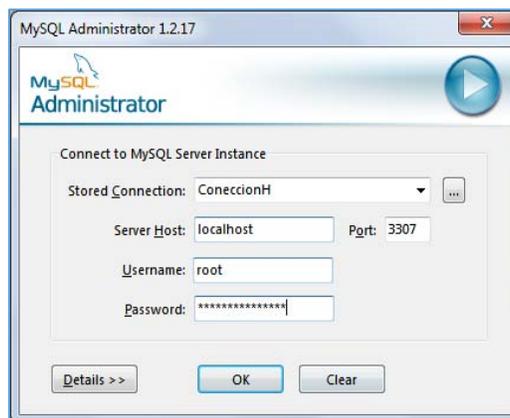


Figura 4.3.2.1.2 Logueo de Conexión al Servidor MySQL.

En la pantalla principal del “*MySQL Administrator*” se muestran todas las actividades que se pueden ejecutar en el servidor de Base de Datos, algunas de ellas son:

- Service control: Inicia y detiene los servidores seleccionados.
- Startup variables: Configuración de las variables de inicio del servidor.
- User Administration: Gestionar usuarios y permisos.
- Server connections: Gestionar todas las conexiones abiertas con el servidor de bases de datos.

- Health: Información sobre el estado del servidor.
- Server Logs: Muestra un historial de logs del servidor.
- Replication Status: Muestra información de los sistemas replicados.
- Backup: Administración de la copia de seguridad de las bases de datos.
- Restore: Permite restaurar las copias de seguridad.
- Catalogs: Permite la administración de las bases de datos y las tablas, índices, vistas y procedimientos, como se muestra en la **Figura 4.3.2.1.3**.

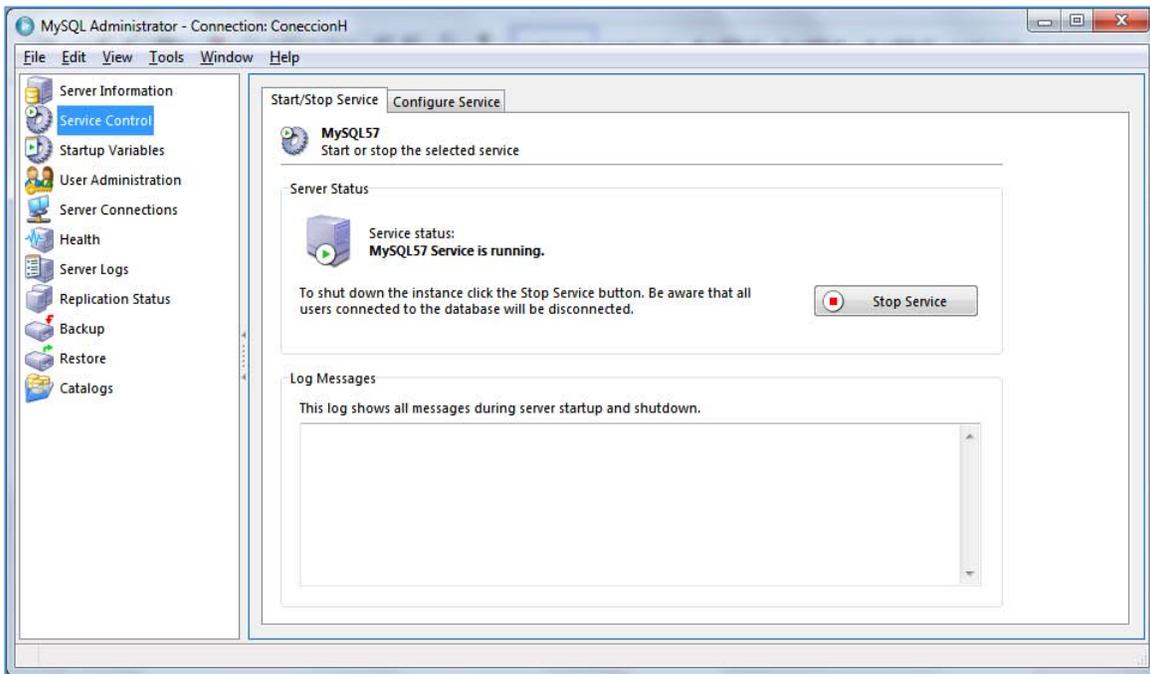


Figura 4.3.2.1.3 Pantalla principal del MySQL Administrator.

Ahora bien, para la creación de la Base de Datos y de las tablas del sistema se utilizará otra potente herramienta que nos ofrece MySQL que es MySQL Query Browser que se incluye en el menú de Tools como lo muestra la **Figura 4.3.2.1.4**.

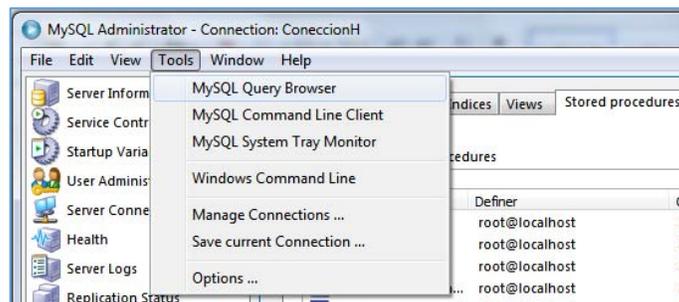


Figura 4.3.2.1.4. MySQL Query Browser.

4.3.2.2. MySQL Query Browser

Esta herramienta cliente nos va a permitir crear, modificar, ejecutar y optimizar sentencias SQL mediante un panel gráfico, como se muestra en la **Figura 4.3.2.2.1.**

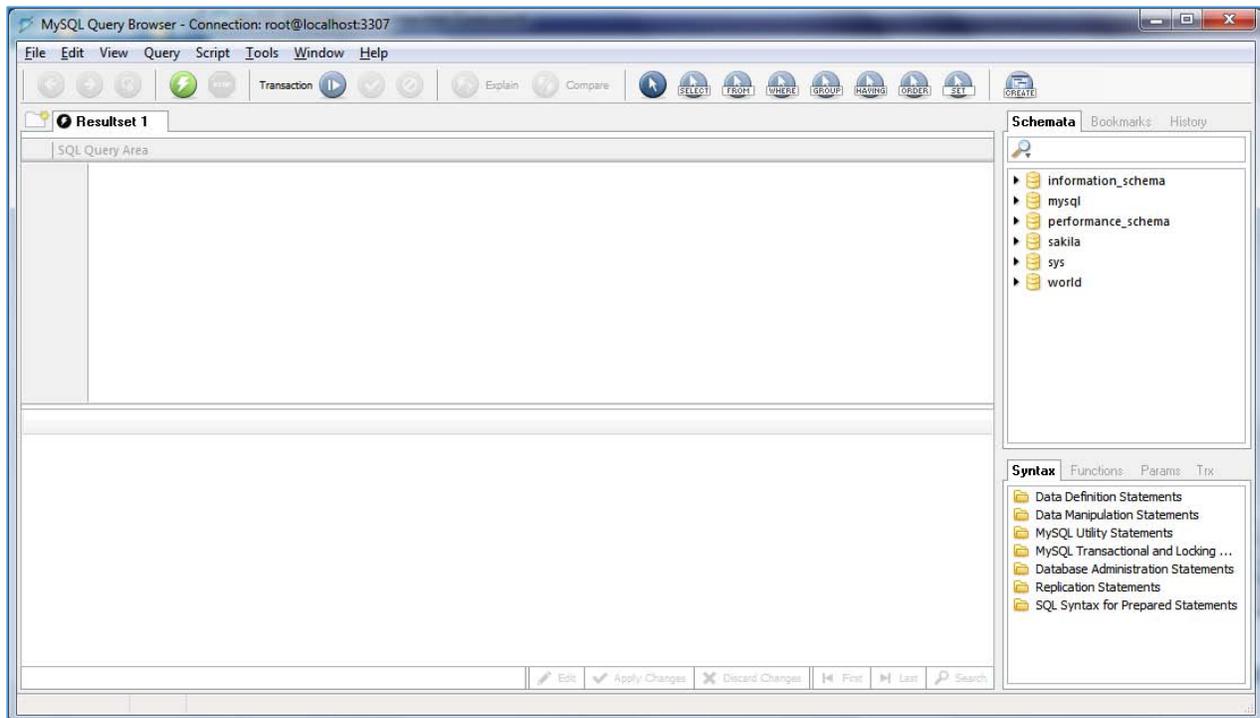


Figura 4.3.2.2.1 Panel principal del MySQL Query Browser.

El panel principal nos ofrece las siguientes características:

- Editor donde se pueden ingresar, editar y ejecutar de manera manual sentencias SQL.
- Muestra un conjunto de resultados en múltiples pestañas para poder navegar entre ellas de manera sencilla.

- Explorador de Base de Datos, tablas y procedimientos disponibles en el servidor.
- Crear y eliminar base de datos y tablas.
- Modificación de la estructura de las tablas.
- Editar perfiles de conexión que pueden conectarse al servidor.
- Historial de sentencias SQL ejecutadas.

A continuación se explicarán algunos de los puntos que se consideran para la instalación de la Base de datos que requiere nuestro sistema en el servidor MySQL.

- **Creación de la Base de Datos:** Existen dos formas de generar la Base de Datos una es utilizando el modo gráfico dando en “Create New Schema”, como se muestra en la **Figura 4.3.2.2.2**.

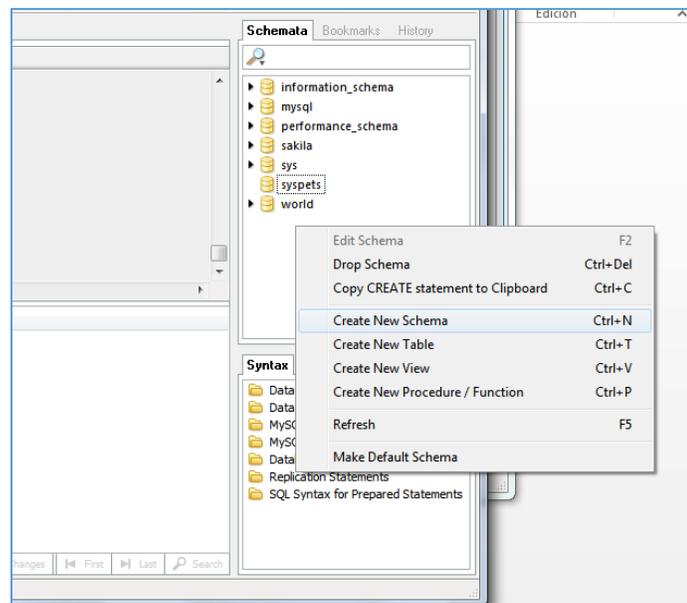


Figura 4.3.2.2.2 forma gráfica de generar una nueva BD.

La otra opción es escribiendo un script SQL en una pestaña de Resultados y ejecutándolo (Rayito) para que nos genere una nueva base de datos como lo muestra la **Figura 4.3.2.2.3**.

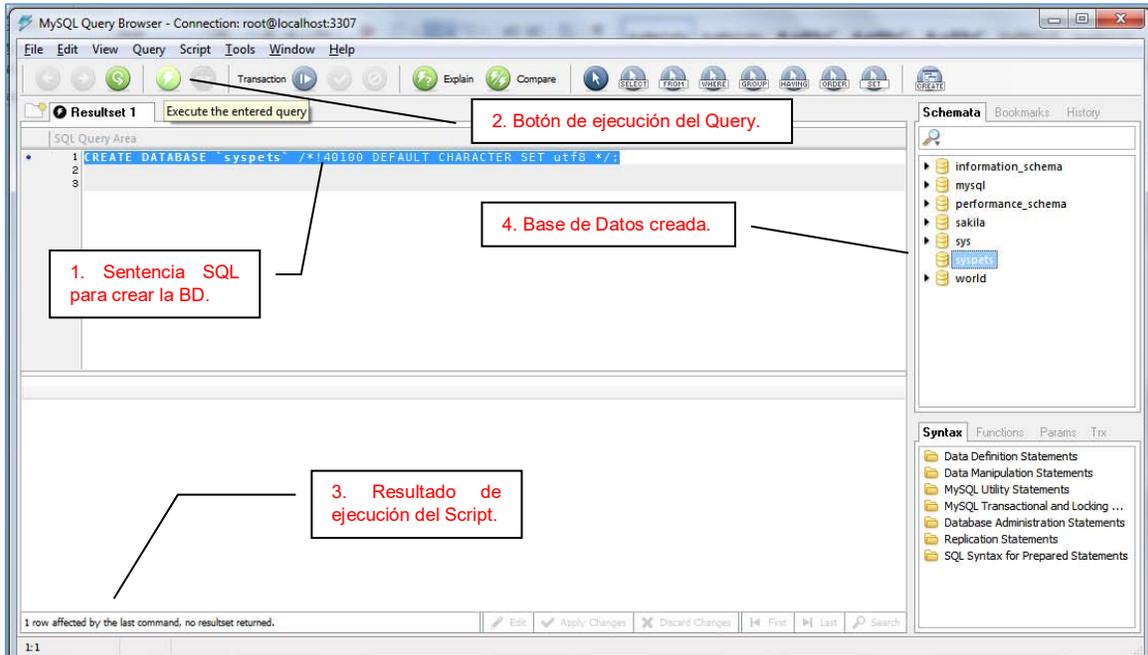


Figura 4.3.2.2.3 Creación de la Base de Datos Syspets (Nombre del sistema que se implementó).

- **Creación de Tablas:** De igual forma existen dos maneras de generar las tablas; La primera es eligiendo en las opciones secundarias de la base de datos que ya hemos generado y dar en "Create New Table", nos mostrará un entorno gráfico para crear las tablas como lo muestra en la **Figura 4.3.2.2.4.**

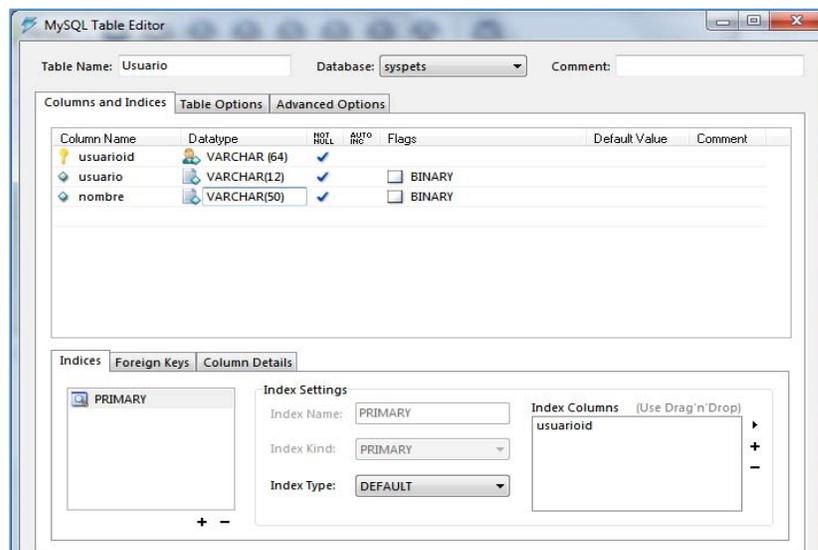


Figura 4.3.2.2.4 Modo gráfico de crear una tabla.

CAPÍTULO 4: ANÁLISIS, DISEÑO Y CONSTRUCCIÓN

La otra forma de generar nuevas tablas es escribiendo la sentencia SQL y ejecutándola (ícono de Rayito) en el editor, como se muestra en la **Figura 4.3.2.2.5.**

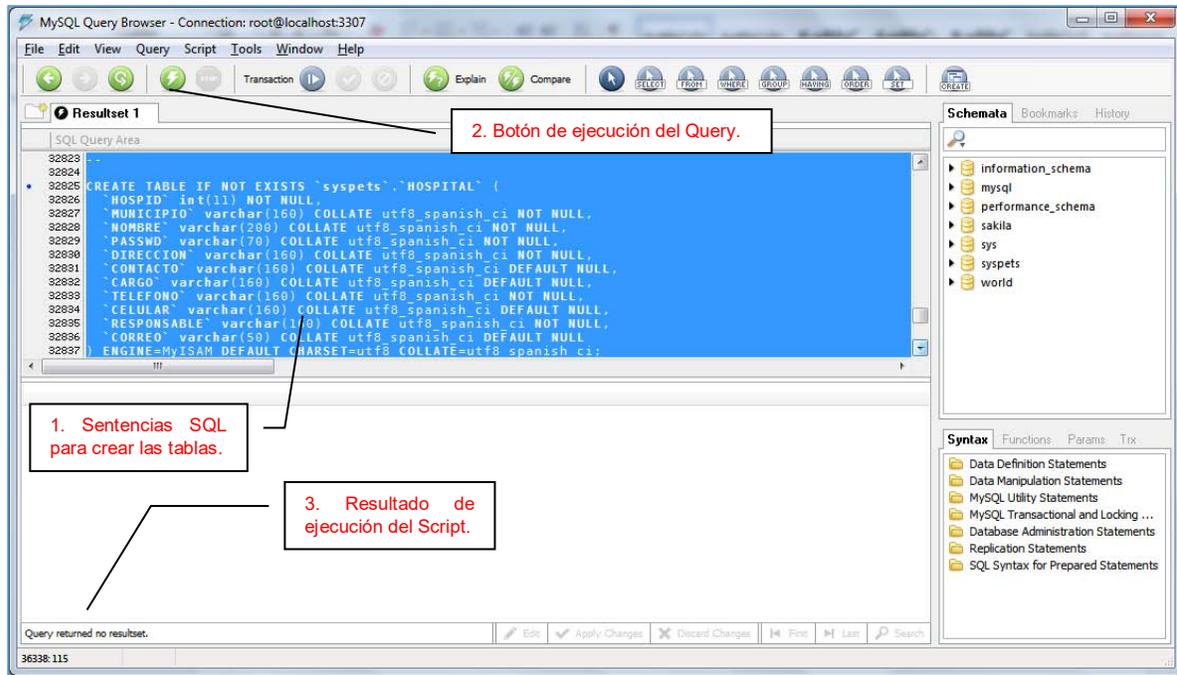


Figura 4.3.2.2.5 Ejecución para creación de Tablas.

Por último se muestra el árbol de tablas que se han creado y los campos que se generaron como se muestra en la **Figura 4.3.2.2.6.**

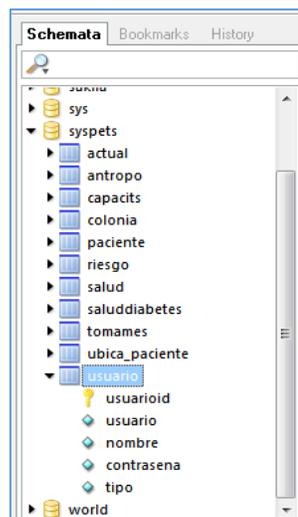


Figura 4.3.2.2.6 Tablas en la BD Syspets.

- **Consulta de información:** Del mismo modo en que se colocó y ejecuto un query para crear las tablas se realizará para obtener información de una tabla por medio de una consulta como se muestra en la **Figura 4.3.2.2.7.**

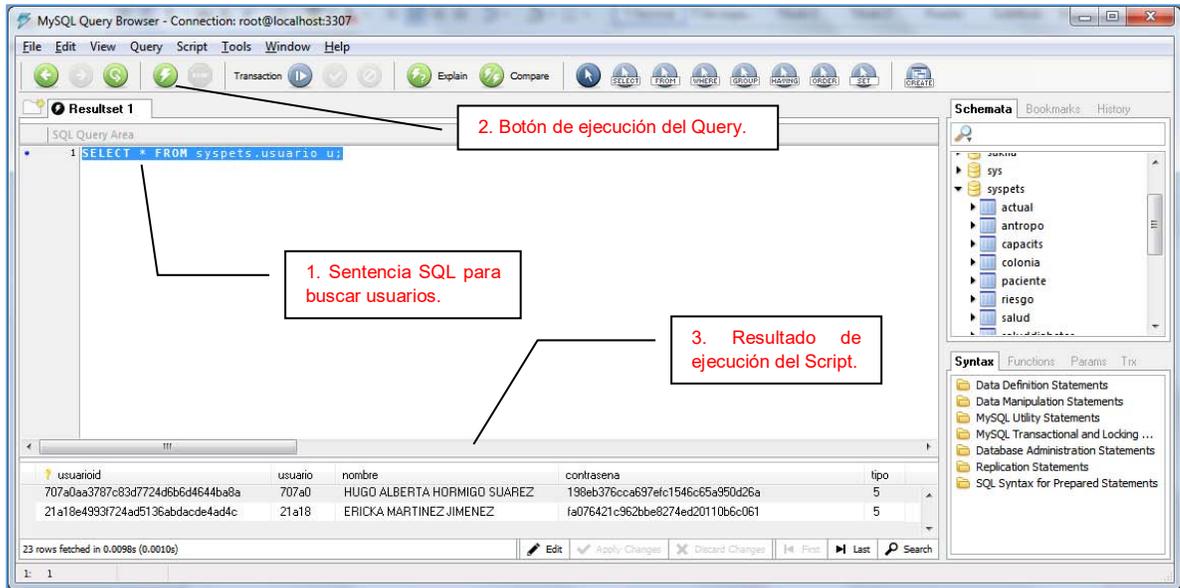


Figura 4.3.2.2.7. Sentencia de Consulta de Usuarios.

Todas las consultas se incluyeron en el código P

4.4. Desarrollo e implementación del fron-end

El sistema de control de pruebas de enfermedades de transmisión sexual (VIH, VPH y Sífilis) (SysPETS) cuenta con diferentes módulos los cuales son:

- Módulo de seguridad
- Módulo de administración de usuarios
- Módulo de atención de pacientes
- Módulo de administración de pruebas entre jurisdicciones
- Módulo de reporte

4.4.1. Módulo de seguridad

La primera pantalla por la que tendrán que pasar todos los usuarios del sistema es la pantalla de autenticación del sistema, la cual solicitará al usuario su clave de usuario así como su contraseña como se muestra en la **Figura 4.4.1.1**



Figura 4.4.1.1 Pantalla de autenticación

En caso de no ingresar correctamente el usuario y/o la contraseña respecto a las registradas en el sistema se mostrará el mensaje de advertencia que se muestra en la **Figura 4.4.1.2**

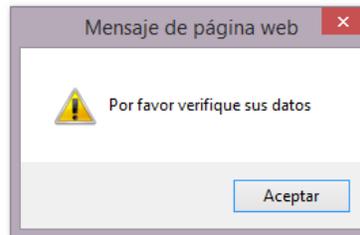


Figura 4.4.1.2 Mensaje de verificar credenciales

Una vez que el usuario ha introducido correctamente sus credenciales, podrá acceder a la pantalla principal, desde donde podrá acceder a las diferentes opciones del sistema. Véase **Figura 4.4.1.3**

Figura 4.4.1.3 Pantalla principal de SysPETS

4.4.2. Módulo de administración de usuarios

Para ingresar al módulo de administración de usuarios, el usuario previamente autenticado deberá dirigirse a la opción del menú “Alta usuario”

El módulo de administración de usuarios consta de dos partes:

Registro de usuario

Para dar de alta un usuario se deberá llenar un formulario con los datos que se muestran en la **Figura 4.4.2.1**

Figura 4.4.2.1 Formulario para dar de alta un usuario dentro del sistema

Dentro del formulario se deberá indicar el rol que tendrá el usuario a registrar.

La segunda parte del módulo de administración consta de una lista con todos los usuarios registrados. Ver **Figura 4.4.2.2**

The image shows a web application interface for user management. At the top, there is a navigation bar with links: "Buscar paciente", "Control de muestras", "Reporte", "Alta usuario", and "Salir". Below this is a header "Usuarios". The main content area is titled "REGISTRO DE USUARIO DATOS GENERALES" and contains a form with the following fields:

- Nombre:** An empty text input field.
- Usuario:** A text input field containing the value "juris".
- Contraseña:** A password input field with masked characters (dots).
- Tipo:** A dropdown menu currently showing "Administrador Medis".

Below the form is a "Registrar" button. Underneath is a table listing existing users:

Usuarios			
Nombre	Usuario	Tipo	Cambiar Contraseña
syspets	syspets	Administrador Medis	
juris	juris	Médico	
MCSSA015560	MCSSA015560	Recepción	
NAUCALPAN	MCSSA015584	Recepción	
TEOTIHUCAN	MCSSA015596	Recepción	
TLALNEPANTLA	MCSSA015613	Recepción	

Figura 4.4.2.2 Lista de usuarios dados de alta dentro del sistema SysPETS

Al dar clic dentro del botón  se mostrará la funcionalidad para modificar la contraseña al usuario seleccionado, tal y como se muestra en la **Figura 4.4.2.3**

REGISTRO DE USUARIO DATOS GENERALES	
Nombre	MCSSA015613
Usuario	TLALNEPANTLA
Contraseña
Tipo	Administrador Medis ▼
Actualizar	

Figura 4.4.2.3 Formulario para modificar la contraseña a un usuario registrado dentro del sistema SysPETS

4.4.3. Módulo de atención de pacientes

Existen dos maneras de acceder al módulo de atención a pacientes. Dentro del menú principal al dar clic en la opción “Buscar paciente” mostrando la pantalla de la **Figura 4.4.3.1**

4.4.3.1

Pacientes

Apellido Paterno: Apellido Materno: Nombre:

RFC:

Buscar

Figura 4.4.3.1 Formulario de búsqueda de pacientes

La cual permite buscar al paciente por:

- Apellido Paterno
- Apellido Materno
- Nombre
- RFC

Una vez que se introducen los criterios de búsqueda, el sistema muestra todos los resultados que cumplen con dichos criterios como se muestra en la **Figura 4.4.3.2**

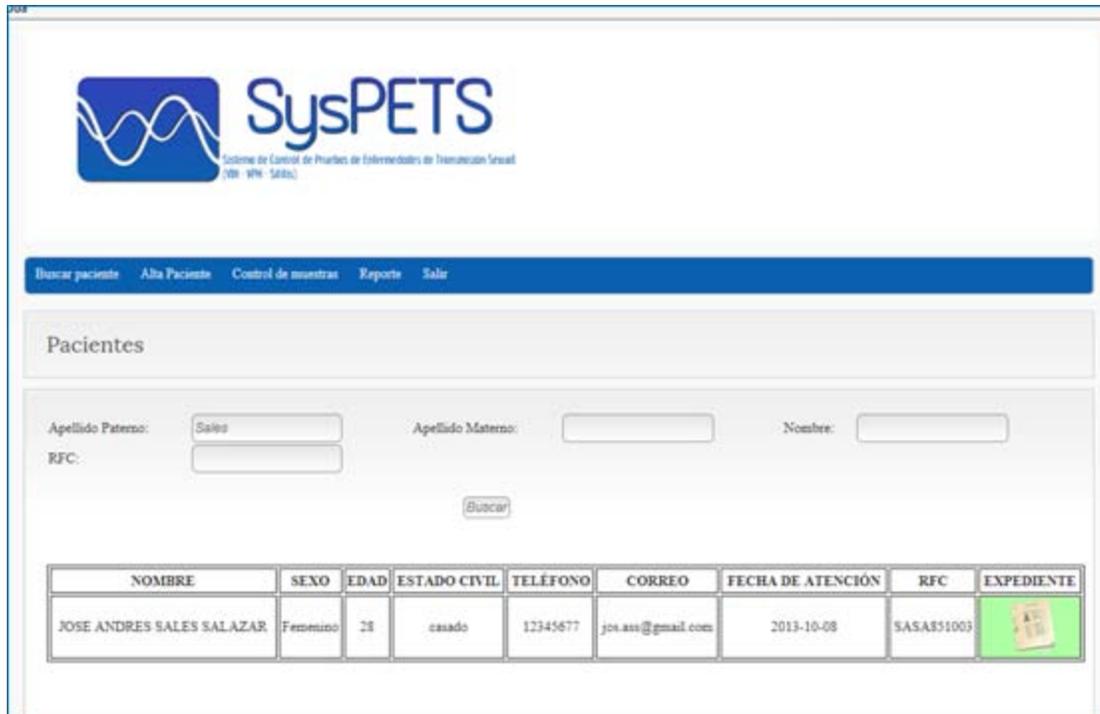


Figura 4.4.3.2 Resultado de realizar la búsqueda de pacientes

Los registros resultantes muestran los datos generales del paciente:

- Nombre
- Sexo
- Edad
- Estado civil
- Teléfono
- Correo electrónico
- Fecha de atención
- RFC

También proporciona un enlace al expediente del paciente, el cual se puede acceder al

hacer clic en el botón

4.4.4. Módulo de administración de pruebas entre jurisdicciones

Dentro del módulo de administración se podrán registrar los envíos de paquetes con pruebas para la detección rápida de enfermedades de transmisión sexual hacia las diferentes jurisdicciones de salud. Para poder registrar dichos envíos es necesario ingresar a la opción del menú Control de muestras y posteriormente en el submenú Envío de muestras, mostrando la pantalla de la **Figura 4.4.4.1**

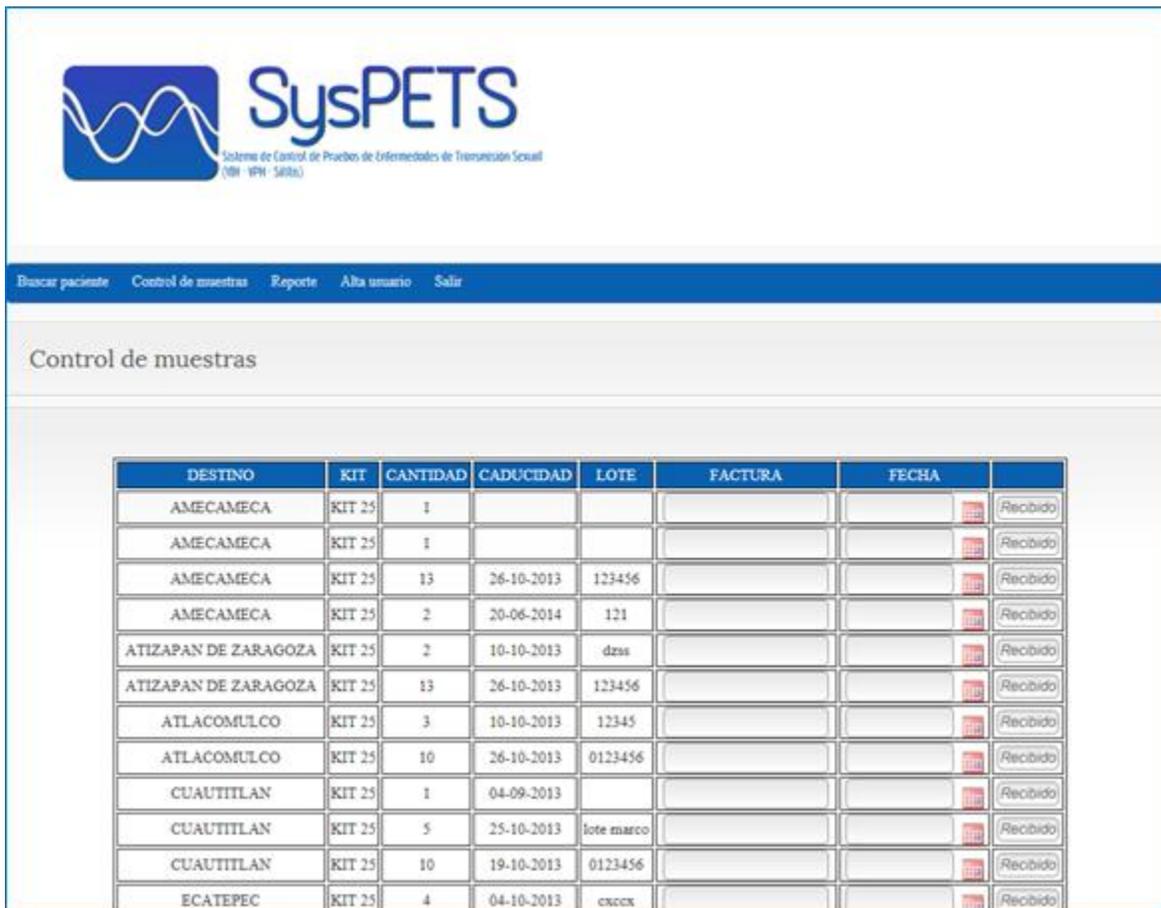
Asignación de muestras por jurisdicción				
JURISDICCIÓN	KIT	CANTIDAD	CADUCIDAD	LOTE
AMECAMECA	KIT 25 MUESTRAS	0	0	0
ATIZAPAN DE ZARAGOZA	KIT 25 MUESTRAS	0	0	0
ATLACOMULCO	KIT 25 MUESTRAS	0	0	0
CUAUTILAN	KIT 25 MUESTRAS	0	0	0
ECATEPEC	KIT 25 MUESTRAS	0	0	0
IXTLAHUACA	KIT 25 MUESTRAS	0	0	0
JILOTEPEC	KIT 25 MUESTRAS	0	0	0
NAUCALPAN	KIT 25 MUESTRAS	0	0	0
NEZAHUALCOYOTL	KIT 25 MUESTRAS	0	0	0
TEJUPILCO	KIT 25 MUESTRAS	0	0	0
TENANCINGO	KIT 25 MUESTRAS	0	0	0

Figura 4.4.4.1 Pantalla para el envío de kits de muestras a las diferentes jurisdicciones

En dicha pantalla se mostrarán todas las jurisdicciones dadas de alta en el sistema, se deberá indicar cuantos kits se le van a asignar a cada una de las jurisdicciones, así como la fecha de caducidad y el lote de las mismas.

De igual forma las jurisdicciones deberán registrar que han recibido dichos kits con muestras, para realizar dicha confirmación es necesario entrar en la Opción Control de muestras y después Recibo de Muestras en dónde se mostrará la lista de los paquetes

de muestras enviados y se tendrá que indicar la fecha en que fueron recibidos e indicar el número de factura expedida en la recepción de dichos kits. La pantalla para la recepción de muestras se muestra en la **Figura 4.4.4.2**



The screenshot shows the SysPETS web application interface. At the top left is the SysPETS logo with the text 'Sistema de Control de Pruebas de Enfermedades de Transmisión Sexual (NH-YPH-SARIS)'. Below the logo is a navigation bar with links: 'Buscar paciente', 'Control de muestras', 'Reporte', 'Alta usuario', and 'Salir'. The main content area is titled 'Control de muestras' and contains a table with the following data:

DESTINO	KIT	CANTIDAD	CADUCIDAD	LOTE	FACTURA	FECHA	
AMECAMECA	KIT 25	1					Recibido
AMECAMECA	KIT 25	1					Recibido
AMECAMECA	KIT 25	13	26-10-2013	123456			Recibido
AMECAMECA	KIT 25	2	20-06-2014	121			Recibido
ATIZAPAN DE ZARAGOZA	KIT 25	2	10-10-2013	dzss			Recibido
ATIZAPAN DE ZARAGOZA	KIT 25	13	26-10-2013	123456			Recibido
ATLACOMULCO	KIT 25	3	10-10-2013	12345			Recibido
ATLACOMULCO	KIT 25	10	26-10-2013	0123456			Recibido
CUAUTILAN	KIT 25	1	04-09-2013				Recibido
CUAUTILAN	KIT 25	5	25-10-2013	lote marco			Recibido
CUAUTILAN	KIT 25	10	19-10-2013	0123456			Recibido
ECATEPEC	KIT 25	4	04-10-2013	cxccx			Recibido

Figura 4.4.4.2 Pantalla de confirmación de recepción de kits de muestras

Para poder visualizar cuantas muestras han sido utilizadas por cada una de las jurisdicciones y cuántas tienen todavía en su inventario es necesario que se registren mensualmente cuántas muestras han sido utilizadas. Es posible acceder a la imagen que se muestra en la **Figura 4.4.4.3** al hacer clic en la opción Control de muestras y posteriormente Pruebas realizadas.

REGISTRO DE MUESTRAS POR MES			
JURISDICCIÓN	MES	AÑO	CANTIDAD
AMECAMECA	FEBRERO	2013	0
ATIZAPAN DE ZARAGOZA		2013	0
ATLACOMULCO		2013	0
CUAUTITLAN		2013	0
ECATEPEC		2013	0
ENTLAHUACA		2013	0
JILOTEPEC		2013	0
NAUCALPAN		2013	0
NEZAHUALCOYOTL		2013	0
TEJUPILCO		2013	0
TENANCINGO		2013	0

Figura 4.4.4.3 Pantalla de registro de pruebas realizadas

4.4.5. Módulo de reportes

Dentro de la opción Reporte – Se pueden generar reportes como:

- Todos los pacientes: Reportes por jurisdicción de los pacientes cuyas pruebas de ETS hayan sido reactivas.
- Por jurisdicción: Reporte de pruebas realizadas y restantes por jurisdicción.
- Por género: Reportes y estadísticas por género de los pacientes cuyas pruebas de ETS hayan sido reactivas en cada jurisdicción.
- Por Rango de edades: Reportes y estadísticas por rango de edades de los pacientes cuyas pruebas de ETS hayan sido reactivas en cada jurisdicción.

En la **Figura 4.4.5.1** se puede observar el menú de acceso al módulo de reportes

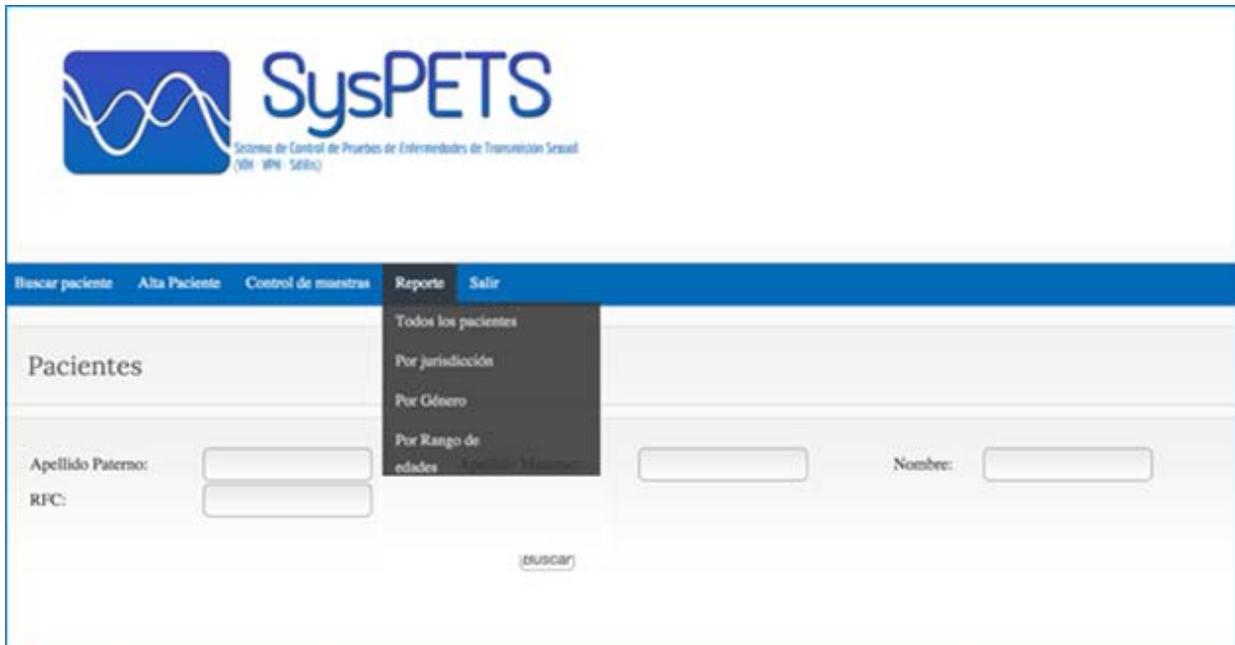


Figura 4.4.5.1 Menú Reportes

4.5. Pruebas y Reportes

4.5.1. Pruebas

Todo sistema esta propenso a tener errores en algunas ocasiones su impacto en el negocio o la funcionalidad del sistema puede llegar a ser muy fuerte al punto de ser considerado un fracaso y esto acarrea consecuencias graves desde ganarse una mala reputación o imagen ante el cliente hasta dañar a la propia empresa en aspectos económicos, sociales, políticos, etc. según sea el ramo de la misma.

Ante esto tanto para los desarrolladores y el cliente es necesario que adquieran la responsabilidad de efectuar pruebas del sistema para minimizar el fallo del sistema contemplando todo tipo de escenarios desde el punto técnico como funcional.

Las pruebas nos permiten asegurar en mayor porcentaje la calidad del sistema dando como resultado beneficios para los desarrolladores con satisfacción de haber entregado un buen producto y al cliente de haber adquirido un producto que cumple con lo que esperaba. También se ve reflejado en reducción de costos de mantenimiento, disminución de tiempos de desarrollo y entrega.

Actualmente existe el International Software Testing Qualifications Board (ISTQB) – Comité Internacional de Cualificación de Pruebas de Software que es una organización de certificación de la calidad del Software y que opera de manera internacional. Esta organización ha ayudado a definir un estándar en las pruebas de Software.

A continuación se mencionan diferentes tipos de pruebas de software que se deben considerar y efectuar en el sistema en función del objetivo en que se centran, estos tipos de pruebas se basan en la ISTQB.

4.5.1.1. Pruebas funcionales

Este tipo de pruebas se entienden como todas las funciones que un sistema, subsistema o componente hacen es decir lo que el sistema hace, estas funcionalidades pueden estar descritas en una especificación de requerimientos, en una especificación de funciones, en los casos de uso o pueden no estar documentadas; Todas estas funciones o características pueden ser usadas como escenario de pruebas.

Algunos tipos de pruebas funcionales son:

- **Pruebas de Caja Negra:** Este tipo de pruebas nos permite mostrar que las funciones del software son operativas y esto se lleva a cabo teniendo valores de entrada ingresados a través de la interfaz del sistema, obviando de alguna forma su codificación interna y produciendo una salida correcta, como se muestra en la **Figura 4.5.1.1**.



Figura 4.5.1.1 Prueba de Caja Negra

Los escenarios que se colocaron a prueba para este tipo fueron:

CAPÍTULO 4: ANÁLISIS, DISEÑO Y CONSTRUCCIÓN

- Búsqueda de paciente.
 - Obtener Expediente de Paciente.
 - Guardar alta de un nuevo paciente.
 - Envío de muestras.
 - Recibo de muestras.
 - Registro de muestras por mes.
 - Reporte todos los pacientes.
 - Reporte por jurisdicción.
 - Reporte por rango de edades.
 - Reporte por género.
- **Pruebas de Seguridad:** Este tipo de pruebas se basa en la seguridad informática que se tiene en el sistema, en estas pruebas se evalúan las entradas de datos en formularios, almacenamiento de información exclusivamente necesaria, tipos de contraseñas, manejo de encriptación de información, colocación de software de seguridad para prevención de ataques accidentales o malintencionados, software para evitar accesos no autorizados.

Los escenarios a los que el sistema fue sometido con estas pruebas son:

- Uso de una cuenta y contraseña para acceder a la aplicación.
 - Solo se solicita información necesaria acorde a la necesidad del sistema.
 - Evaluación de valores correctos en los formularios dentro del sistema.
 - Instalación de antivirus para evitar ataques.
 - Instalación de cortafuegos para evitar accesos no autorizados.
 - Uso de software libre respetando los lineamientos para el desarrollo del sistema.
- **Pruebas de Interoperabilidad:** Pruebas de interoperabilidad es el proceso de pruebas con el objeto de determinar la interoperabilidad de un producto de

software, es decir la capacidad del producto para interactuar con uno o más componentes o sistemas.

Los escenarios de pruebas evaluados usando las pruebas de interoperabilidad en el sistema son:

- El sistema tiene la capacidad de convivir teniendo comunicación con un servicio de envío de correos electrónicos.
- El sistema está desarrollado con tecnologías (MySQL, PHP, Apache) que son multiplataforma brindando la interoperabilidad y también la funcionalidad del producto en distintos sistemas operativos.

4.5.1.2. Pruebas no funcionales

Este tipo de pruebas van enfocadas a cómo trabaja el sistema siendo lo contrario de las pruebas funcionales que se enfocan a lo que hace el sistema. Dentro de éstos tipos de pruebas encontraremos que van más enfocadas en medir las características de los sistemas y software, tales como tiempos de respuesta a solicitudes ejecutadas por el usuario o a la cantidad de información soportada. Las pruebas no funcionales consideran los factores externos del software. A continuación se mencionan tipos pruebas que corresponden a este grupo de pruebas no funcionales.

- **Pruebas de Rendimiento:** El rendimiento del sistema se encargará de determinar si éste cumple con las funciones por las cuales fue realizado considerando tiempo de procesamiento y la tasa de transferencia.

Dentro del sistema efectuamos las siguientes pruebas:

- Se ejecutaron transacciones desde varios clientes a través de un browser sin presentar problemas.

CAPÍTULO 4: ANÁLISIS, DISEÑO Y CONSTRUCCIÓN

- **Pruebas de Estrés:** Van enfocadas en evaluar que el sistema funcione apropiadamente y sin errores, esto se logra al intentar encontrar errores o haciendo que el sistema falle colocando al sistema en condiciones extremas debido a recursos insuficientes, algunos ejemplos; agotamiento de recursos, llenado disco duro, llenado de memoria, accesos a los servidores.

Pruebas ejecutadas en el sistema:

- Debido a la poca cantidad de información manejada a corto plazo en el sistema las pruebas no fueron realizadas con grandes volúmenes pero aun así se generaron pruebas con volúmenes considerables.
- **Pruebas de Carga:** Al igual que las pruebas de estrés buscan encontrar problemas de funcionalidad del sistema pero se apoya en cargas ejemplo; número de usuarios concurrentes, número de transacciones.

Pruebas efectuadas al sistema fueron:

- Se hizo uso del sistema con varios usuarios operando el sistema al mismo momento sin presentar errores.
- Ejecución de varias transacciones (actividades) al sistema por parte de varios usuarios.
- **Pruebas de Usabilidad:** Son pruebas que nos van a apoyar a determinar cuan fácil de aprender a operar, administrar y entender es el sistema.

Las pruebas efectuadas fueron:

- Se le pidió a usuarios que no conocían el sistema a que lo operaran por unos momentos tras una breve explicación de lo que realizaba, siendo esta prueba satisfactoria puesto que los usuarios determinaron que era bastante sencillo utilizarlo y comprenderlo, estos usuarios ya contaban con un conocimiento mínimo de operación en computación.

CAPÍTULO 4: ANÁLISIS, DISEÑO Y CONSTRUCCIÓN

- **Pruebas de Mantenibilidad:** Éste tipo de pruebas nos ayudaran a determinar la facilidad que tiene el sistema para poder ser modificado ante errores, ajustes de operación o funcionalidad del sistema.

Las pruebas efectuadas son:

- Al final del desarrollo del sistema se efectuaron algunos parches de prueba para cambiar colores del sistema, estos parches solo para evaluar la facilidad para la instalación ante un ajuste, la modificación de código del sistema ya final.
- **Pruebas de Fiabilidad:** Su propósito es determinar la fiabilidad del sistema y éste resultado se tardará en definir pues se debe operar el sistema por cierto periodo en condiciones lo más reales posibles para poder de alguna forma generar la confianza en el sistema.

Las pruebas efectuadas fueron:

- El sistema fue puesto a prueba durante un periodo de tiempo (15 días), determinando así la fiabilidad de mismo.
- **Pruebas de Portabilidad:** Esta prueba consiste en determinar cuan portable o transferible es el sistema, es decir su capacidad del sistema para poder ser instalado y configurado en un nuevo hardware o si su entorno de software se ve modificado.

Las pruebas que se efectuaron para poder cumplir este tipo fue:

- Se instaló y operó el sistema en distintos equipos de cómputo para identificar lo portable del mismo, cabe mencionar que estos equipos contaban con características distintas de hardware y sistemas operativos distintos.

4.5.1.2. Pruebas estructurales

Se basan en el cómo está construido el sistema, es decir se encargan de validar el código del sistema, el diseño del sistema, la estructura de la base de datos. Estas pruebas pueden realizarse en todos los niveles de pruebas, se deben diseñar los casos de prueba desde el más alto nivel hasta llegar a efectuar pruebas más detalladas.

- **Pruebas de caja Blanca:** Intentan garantizar que al menos todos las opciones o flujos del sistema sean validados, es decir que se utilicen todas las estructuras internas del sistema.
 - Dentro del sistema se ejecutaron pruebas de los módulos considerando todos los escenarios detectados en el requerimiento, las pruebas se iniciaron desde lo más general hasta contemplar los escenarios a mayor detalle en pruebas unitarias de todos los escenarios posibles que se plantearon, como se muestra en la **Figura 4.5.1.3.**

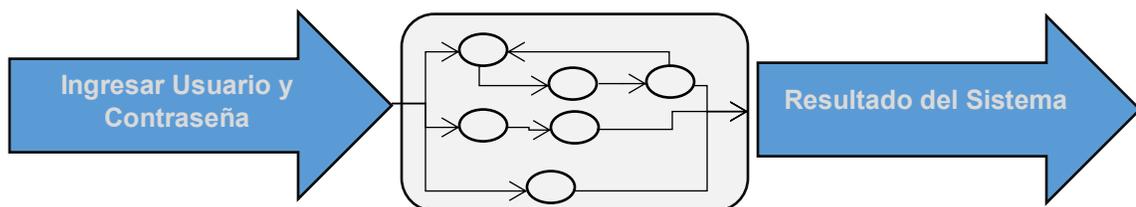


Figura 4.5.1.3 Prueba de Caja Blanca, Autenticación de entrada al sistema.

4.5.1.3. Pruebas asociadas a cambios: Repetición de pruebas y pruebas de regresión.

Estas pruebas son consideradas como una actividad dentro del desarrollo del sistema pues se deben ejecutar cada que se efectúe una corrección o cambio de funcionalidad al sistema. Algunos tipos de pruebas que se efectúan en este género son:

- **Pruebas de Regresión:** Son las pruebas reiteradas sobre el sistema o software ya probado pero que ha sufrido cambios ya sea por una corrección

de algún error o por la mejora de alguna funcionalidad. Las pruebas se efectúan en módulos que estén asociadas al flujo o proceso modificado pero también a módulos que no estén asociados pero que se desea probar que no se vea afectado por los cambios ejecutados.

- **Pruebas de confirmación:** Nos permiten como su nombre lo indica confirmar que el cambio realizado en el sistema funcione correctamente y que resuelven el error o la nueva funcionalidad que se requiere.
 - Durante el desarrollo del sistema se contemplaron las pruebas de regresión y de confirmación para valorar las nuevas funcionalidades introducidas y que no se habían contemplado a detalle en un inicio.
 - Otros escenarios fueron la corrección de errores durante el desarrollo y también durante el periodo de pruebas que se habían establecido una vez que el sistema estaba terminado. Con la ayuda de las demás pruebas se logró detectar errores que no se tenían contemplados pero al hacer los ajustes se tuvieron que utilizar las pruebas de regresión y de confirmación.

4.5.2. Reportes

4.5.2.1. Todos los pacientes

En el menú principal en la pestaña Reporte dar clic en Todos los pacientes como se muestra en la **Figura 4.5.2.1.1**



Figura 4.5.2.1.1 Acceso desde el menú principal para el reporte Todos los pacientes

La siguiente pantalla desplegará un menú donde se podrán introducir los siguientes parámetros que permitirán la visualización del reporte:

Jurisdicción: Permite elegir una jurisdicción sobre la cual hará la búsqueda del total de muestras con las que cuenta, así como el total de pruebas realizadas a los pacientes y pruebas que restarían en esa jurisdicción.

Acción: Dar clic en la lista desplegable – JURISDICCIÓN-- y seleccionar la jurisdicción como se muestra en la **Figura 4.5.2.1.2**.



Figura 4.5.2.1.2 Entrada del parámetro jurisdicción.

VIH: Permite marcar la enfermedad VIH para desplegar el total de pacientes que resultaron positivos a esta enfermedad por jurisdicción.

Acción: Dar clic en el cuadro de selección de la enfermedad VIH para seleccionarlo como filtro de búsqueda o dejarlo en blanco si no se desea como parámetro de entrada.

VPH: Permite marcar la enfermedad VPH para desplegar el total de pacientes que resultaron positivos a esta enfermedad por jurisdicción.

Acción: Dar clic en el cuadro de selección de la enfermedad VPH para seleccionarlo como filtro de búsqueda o dejarlo en blanco si no se desea como parámetro de entrada.

Sífilis: Permite marcar la enfermedad Sífilis para desplegar el total de pacientes

CAPÍTULO 4: ANÁLISIS, DISEÑO Y CONSTRUCCIÓN

que resultaron positivos a esta enfermedad por jurisdicción.

Acción: Dar clic en el cuadro de selección de la enfermedad Sífilis para seleccionarlo como filtro de búsqueda o dejarlo en blanco si no se desea como parámetro de entrada.

Condiciones:

- Ningún parámetro es mandatorio.
- Cuando no se selecciona una jurisdicción el reporte se genera sobre todas las jurisdicciones.
- Cuando no se marca ninguna enfermedad no genera los campos correspondientes.

Una vez que se hayan ingresado todas las entradas en la misma pantalla se desplegara el reporte generado con los siguientes campos:

- NUM: Indica el número consecutivo del registro desplegado.
- JURISDICCIÓN: Indica la Jurisdicción donde se encuentran las muestras de los pacientes.
- MUESTRAS TOTALES: Cantidad de muestras totales con las que cuenta la jurisdicción para realizar pruebas.
- PRUEBAS REALIZADAS: Cantidad total de pruebas realizadas en la jurisdicción.
- PRUEBAS RESTANTES: Cantidad total de pruebas que restan a las realizadas con las que cuenta la jurisdicción.
- VIH REACTIVO: Cantidad total de muestras que dieron como resultado positivo a la enfermedad VIH.
- SÍFILIS REACTIVO: Cantidad total de muestras que dieron como resultado positivo a la enfermedad SÍFILIS.
- VPH REACTIVO: Cantidad total de muestras que dieron como resultado positivo a la enfermedad VPH.

La **Figura 4.5.2.1.3** muestra el reporte generado:

Jurisdicción:	VALLE DE BRAVO
VIH:	Reactivo VIH <input type="checkbox"/>
VPH:	Reactivo VPH <input type="checkbox"/>
Sífilis:	Reactivo Sífilis <input checked="" type="checkbox"/>
<input type="button" value="Buscar"/>	

NUM	JURISDICCIÓN	MUESTRAS TOTALES	PRUEBAS REALIZADAS	PRUEBAS RESTANTES	SIFILIS REACTIVO
1	VALLE DE BRAVO	725	228	497	<input type="button" value="2"/>

Figura 4.5.2.1.3 Reporte Todos los pacientes con Valle de Bravo y Sífilis como entradas.

La pantalla cuenta con las siguientes funciones:

- Acceso al menú principal de SysPETS.
- Acceso al menú de entrada de parámetros para la generación del reporte para realizar un nuevo reporte.
- Mediante la flecha ← “regresar” del navegador permite regresar a la pantalla anterior ya sea de una consulta previa o del menú principal si es la primera consulta.
- Subir y bajar en la pantalla por medio de las teclas de desplazamiento.
- Acceso a los expedientes de los pacientes reactivos en los recuadros con las cantidades en cada ETS.

4.5.2.2 Por jurisdicción

4.5.2.3 Estadística por edad

En el menú principal en la pestaña Reporte dar clic en Por Rango de edades como se muestra en la **Figura 4.5.2.3.1**

Buscar paciente Alta Paciente Control de muestras Reporte Salir					
Pacientes		Todos los pacientes Por jurisdicción Por Género Por Rango de edades			
Apellido Paterno:	<input type="text"/>	Apellido Materno:	<input type="text"/>	Nombre:	<input type="text"/>

Figura 4.5.2.3.1 Acceso desde el menú principal para la Estadística por Edad.

La siguiente pantalla desplegará un menú donde se podrán introducir los siguientes parámetros que permitirán la visualización de la estadística:

Edad: Permite elegir un rango de edad sobre la cual hará la búsqueda de los pacientes que hayan dado positivo a alguna ETS.

Acción: Dar clic en la lista desplegable – EDAD-- y seleccionar el rango de edad (0-20, 21-30, 31-40, 41-60, 61 en adelante) como se muestra en la **Figura 4.5.2.3.2**



Figura 4.5.2.3.2 Entrada del parámetro edad.

VIH: Permite marcar la enfermedad VIH para desplegar el total de pacientes que resultaron positivos a esta enfermedad por edad.

Acción: Dar clic en el cuadro de selección de la enfermedad VIH para seleccionarlo como filtro de búsqueda o dejarlo en blanco si no se desea como parámetro de entrada.

VPH: Permite marcar la enfermedad VPH para desplegar el total de pacientes que resultaron positivos a esta enfermedad por edad.

CAPÍTULO 4: ANÁLISIS, DISEÑO Y CONSTRUCCIÓN

Acción: Dar clic en el cuadro de selección de la enfermedad VPH para seleccionarlo como filtro de búsqueda o dejarlo en blanco si no se desea como parámetro de entrada.

Sífilis: Permite marcar la enfermedad Sífilis para desplegar el total de pacientes que resultaron positivos a esta enfermedad por edad.

Acción: Dar clic en el cuadro de selección de la enfermedad Sífilis para seleccionarlo como filtro de búsqueda o dejarlo en blanco si no se desea como parámetro de entrada.

Condiciones:

- Ningún parámetro es mandatorio.
- Cuando no se selecciona una edad la estadística se genera sobre todas las edades.
- Cuando no se marca ninguna enfermedad el reporte se genera sobre todas las enfermedades.

Una vez que se hayan ingresado todas las entradas en la misma pantalla se desplegara la estadística generada con los siguientes campos:

- NUM: Indica el número consecutivo del registro desplegado.
- EDAD: Indica el rango de edad si se seleccionó un rango o la edad si no se seleccionó un rango.
- JURISDICCIÓN: Indica la Jurisdicción dónde se encuentran las muestras de los pacientes.
- PRUEBAS REALIZADAS: Cantidad total de pruebas realizadas en la jurisdicción.
- VIH REACTIVO: Cantidad total de muestras que dieron como resultado positivo a la enfermedad VIH.

CAPÍTULO 4: ANÁLISIS, DISEÑO Y CONSTRUCCIÓN

- SÍFILIS REACTIVO: Cantidad total de muestras que dieron como resultado positivo a la enfermedad SÍFILIS.
- VPH REACTIVO: Cantidad total de muestras que dieron como resultado positivo a la enfermedad VPH.
- EDAD: Rango de edad seleccionado por el usuario.
- TOTAL PRUEBAS: Total de pruebas realizadas.
- REACTIVOS: Total de reactivos.
- PORCENTAJE: Porcentaje total de reactivos
- %VIH: Indica el porcentaje total de pacientes infectados de VIH por rango de edades.
- %SÍFILIS: Indica el porcentaje total de pacientes infectados de Sífilis por rango de edades.
- %VPH: Indica el porcentaje total de pacientes infectados de VPH por rango de edades.

La **Figura 4.5.2.3.3** muestra la estadística generada:



Figura 4.5.2.3.3 Estadística por Edad con Rango: 31 a 40 y todas las ETS como entrada.

La pantalla cuenta con las siguientes funciones:

- Acceso al menú principal de SysPETS.
- Acceso al menú de entrada de parámetros para la generación de la estadística para realizar una nueva.
- Mediante la flecha ← “regresar” del navegador permite regresar a la pantalla anterior ya sea de una consulta previa o del menú principal si es la primera consulta.
- Subir y bajar en la pantalla por medio de las teclas de desplazamiento.
- Acceso a los expedientes de los pacientes reactivos en los recuadros con las cantidades 1 en cada ETS.

4.5.2.4 Estadística por género

En el menú principal en la pestaña Reporte dar clic en Por Género como se muestra en la **Figura 4.5.2.4.1**

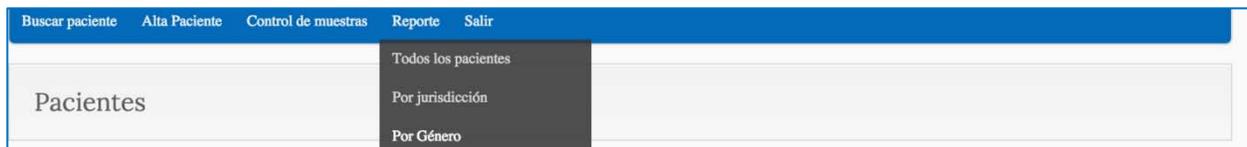


Figura 4.5.2.4.1 Acceso desde el menú principal para la Estadística por Género.

La siguiente pantalla desplegará un menú donde se podrán introducir los siguientes parámetros que permitirán la visualización de la estadística:

Género: Permite marcar un género sobre el cual hará la búsqueda de los pacientes

que hayan dado positivo a alguna ETS.

Acción: Dar clic en el cuadro de selección “F” si la búsqueda se requiere del género Femenino y/o dar clic en el cuadro de selección “M” si la búsqueda se quiere del género Masculino como se muestra en la **Figura 4.5.2.4.2**



Reporte Por Genero

Genero: --GÉNERO--
FEMENINO
✓ MASCULINO

Reactivo VIH

Reactivo VPH

Reactivo Sífilis

Buscar

Figura 4.5.2.4.2 Entrada del parámetro género.

VIH: Permite marcar la enfermedad VIH para desplegar el total de pacientes que resultaron positivos a esta enfermedad por género.

Acción: Dar clic en el cuadro de selección de la enfermedad VIH para seleccionarlo como filtro de búsqueda o dejarlo en blanco si no se desea como parámetro de entrada.

VPH: Permite marcar la enfermedad VPH para desplegar el total de pacientes que resultaron positivos a esta enfermedad por género.

Acción: Dar clic en el cuadro de selección de la enfermedad VPH para seleccionarlo como filtro de búsqueda o dejarlo en blanco si no se desea como parámetro de entrada.

Sífilis: Permite marcar la enfermedad Sífilis para desplegar el total de pacientes que resultaron positivos a esta enfermedad por género.

Acción: Dar clic en el cuadro de selección de la enfermedad Sífilis para seleccionarlo como filtro de búsqueda o dejarlo en blanco si no se desea como parámetro de entrada.

Condiciones:

CAPÍTULO 4: ANÁLISIS, DISEÑO Y CONSTRUCCIÓN

- Ningún parámetro es mandatorio.
- Cuando no se selecciona ningún género la estadística se genera sobre ambos (masculino y femenino).
- Cuando no se marca ninguna enfermedad la estadística se genera sobre todas las enfermedades.

Una vez que se hayan ingresado todas las entradas en la misma pantalla se desplegara la estadística generada con los siguientes campos:

- NUM: Indica el número consecutivo del registro desplegado.
- GÉNERO: Indica el género seleccionado si se marcó alguno en caso contrario muestra la información que exista para ambos géneros.
- JURISDICCIÓN: Indica la Jurisdicción dónde se encuentran las muestras de los pacientes.
- PRUEBAS REALIZADAS: Cantidad total de pruebas realizadas en la jurisdicción.
- VIH REACTIVO: Cantidad total de muestras que dieron como resultado positivo a la enfermedad VIH.
- SÍFILIS REACTIVO: Cantidad total de muestras que dieron como resultado positivo a la enfermedad SÍFILIS.
- VPH REACTIVO: Cantidad total de muestras que dieron como resultado positivo a la enfermedad VPH.
- GÉNERO: Género seleccionado por el usuario.
- TOTAL PRUEBAS: Total de pruebas realizadas.
- REACTIVOS: Total de reactivos.
- PORCENTAJE: Porcentaje total de reactivos.
- %VIH: Indica el porcentaje total de pacientes infectados de VIH por género.
- %SÍFILIS: Indica el porcentaje total de pacientes infectados de Sífilis por género.
- %VPH: Indica el porcentaje total de pacientes infectados de VPH por género.

La **Figura 4.5.2.4.6** muestra la estadística generada:



Figura 4.5.2.4.6 Estadística por Género, con género Femenino, VIH, VPH y Sífilis como entrada.

La pantalla cuenta con las siguientes funciones:

- Acceso al menú principal de SysPETS.
- Acceso al menú de entrada de parámetros para la generación de la estadística para realizar una nueva.
- Mediante la flecha ← “regresar” del navegador permite regresar a la pantalla anterior ya sea de una consulta previa o del menú principal si es la primera consulta.
- Subir y bajar en la pantalla por medio de las teclas de desplazamiento.
- Acceso a los expedientes de los pacientes reactivos en los recuadros con las cantidades 1 en cada ETS.

4.5.2.6 Reporte general del paciente

En la opción Buscar paciente se puede acceder al expediente de cada paciente el cual es un reporte llamado Reporte general del paciente haciendo clic sobre el ícono



, así como se muestra en la **Figura 4.5.2.6.1**

CAPÍTULO 4: ANÁLISIS, DISEÑO Y CONSTRUCCIÓN

Pacientes

Apellido Paterno: Apellido Materno: Nombre:

RFC:

NOMBRE	SEXO	EDAD	ESTADO CIVIL	TELÉFONO	CORREO	FECHA DE ATENCIÓN	RFC	EXPEDIENTE
JOSE ANDRES SALES SALAZAR	Femenino	28	casado	12345677	jos.ass@gmail.com	2013-10-08	SASA851003	
HUGO ALBERTA HORMIGO SUAREZ	Femenino	33	casado	12345678	jos.ass@gmail.com	2013-10-08	HOSH791111	

Figura 4.5.2.6.1 Acceso al Expediente del Paciente por la opción buscar paciente.

En los reportes Todos los pacientes, Por género y Por rango de edades también se puede acceder al Reporte general del paciente haciendo clic en las cantidades de pacientes que arrojan los reportes como se muestra en la **Figura 4.5.2.6.2.**

Genero: MASCULINO ▾

VIH: Reactivo VIH

VPH: Reactivo VPH

Sifilis: Reactivo Sifilis

NUM	GENERO	JURISDICCIÓN	PRUEBAS REALIZADAS	SIFILIS REACTIVO
1	M	VALLE DE BRAVO	2	7

Figura 4.5.2.6.2 Acceso al Reporte general del paciente en reporte por género.

Una vez que abres el reporte se visualiza un formato como el que se muestra en la **Figura 4.5.2.6.3.**

CAPÍTULO 4: ANÁLISIS, DISEÑO Y CONSTRUCCIÓN

Reporte General del Paciente				
DATOS GENERALES				
Nombre	ALBERTO SANDOVAL JUAREZ	Fecha de atención	2016-03-08	
Edad	34 años	Fecha de nacimiento	1982-02-12	
Ocupación	ADMINISTRADOR	Lugar de trabajo	BANCO DE MEXICO	
Teléfono	5557585960	E-mail	baritone2005@gmail.com	
REGISTRO ANTROPOMÉTRICO				
Peso	80 kg	Talla	1.78 m	
IMC	25.25 Kg/m ²	Cintura	90 cm	
RESULTADOS				
Presión Arterial	Sistólica	100 mmHg	Diastólica	80 mmHg
Frecuencia cardiaca	80 Fc'			
Hospital al que se refiere	HG. TEJUPILCO	Médico que refiere	ethel aparicio	
VIH	NEGATIVO			
VPH	REACTIVO			
Sifilis	NEGATIVO			
Fecha de toma	08/03/2016			
CAPACIT al que se refiere	Toluca			
Recomendaciones				

Figura 4.5.2.6.3 Reporte general del paciente.

CONCLUSIONES

- La implementación del sistema SysPETS ha permitido tener un control centralizado de las pruebas realizadas en cada una de las localidades, que con el control físico de pruebas no se tenía.
- La gestión centralizada de inventarios permite al responsable de estos llevar una gestión adecuada de los insumos en cada localidad donde se realizan las pruebas, lo que permite anticipar la posible falta de pruebas y surtirlos nuevamente, previniendo así la falla del servicio.
- El uso de una solución basada en un portal web permite a los usuarios poder ingresar al sistema desde diferentes plataformas, ya que el ambiente utilizado no requiere la instalación de algún software adicional.
- El manejo de histórico de pruebas dentro de la plataforma SysPETS, facilita al médico la revisión del avance del paciente, y hace más ágil el proceso de diagnóstico y atención oportuna.
- El uso del sistema centralizado de información digital que provee SysPETS permite la reducción de costos originados por insumos para impresión de resultados, así como al reducir el número de personas que intervienen en el proceso.
- Las plataformas de BD y de aplicación, poseen un nivel de integración nativo, lo que permite facilitar su interacción y mejora los procesos de control de cambios y actualización sobre la plataforma
- La creación de perfiles de usuario en la plataforma facilita la segregación de funciones, con lo que ayuda a que el personal que interactúa con SysPETS

solamente tendrá acceso a lo que es requerido por sus tareas dentro del proceso.

- La opción de generación de reportes provista por SysPETS provee información que ayuda a las instituciones de salud a obtener información acerca de los sectores más propensos al contagio de Enfermedades de Transmisión Sexual, lo que permite a generar campañas enfocadas a dichos sectores, buscando así la disminución de contagios.
- El sistema SysPETS permite una administración centralizada, lo que permite reducir el número de personas asignadas a esta tarea a uno o dos individuos, independientemente de cuantas localidades se encuentren dadas de alta en el sistema.
- El uso de Apache y MySQL como plataformas de desarrollo, permite tener la opción de escalabilidad en caso de que el sistema requiera un aumento en el desempeño, ya sea creciendo la infraestructura *stand alone*, o con el uso de arquitecturas en *cluster*.
- Al utilizar MySQL, la información puede ser fácilmente respaldada, lo cual permite una recuperación en caso de corrupción o pérdida de la Base de Datos
- El uso de una plataforma Web, permite generar un ambiente de DRP de manera sencilla, lo cual facilita el continuar la operación en caso de un incidente sobre la plataforma productiva.

BIBLIOGRAFÍA

- S. Hawkins, *Apache Web Server Administration & e-commerce Handbook*. Prentice Hall, 2001.
- H. Korth, *et al.*, *Fundamentos de Bases de Datos*, 3rd ed., Madrid, España: McGraw-Hill, 2002.
- O. Pons, *et al.*, *Introducción a las Bases de Datos*, Granada, España: Editorial Paraninfo, 2005.
- J. Sánchez, *Principios sobre Bases de Datos Relacionales*, 5th ed., CreativeCommons, 2005.
- J. Hanke, *PC Cuaderno Los Secretos de PHP y MySQL*, 1st ed., México: Pearson Educación, 2006.
- A Silberschatz, *Fundamentos de Bases de Datos*. 5th ed., Madrid, España: McGraw-Hill, 2006.
- J. Pavona, *Creación de un Portal con PHP y MySQL*. 3rd ed., México: Alfa Omega, 2007.
- P Díaz, *et al.* *Ingeniería de la WEB y patrones de Diseño*, Madrid, España: Prentice Hall, 2005.
- C Pérez, *Dreamweaver 8: Desarrollo de Páginas Web Con PHP y MySQL*. 1st ed., México: Alfa Omega, 2007.
- L. Welling and L. Thompson, *PHP and MySQL Web Development*, 2nd ed., Sams Publishing, 2003.
- A. Pressman., *Ingeniería de Software*. 5th ed. México: McGraw-Hill, 2000.
- P. Velthuis and M. Gerardo, *Tecnología y Diseño de Bases de Datos*, 1st ed. México: Alfa Omega, 2007.

FUENTES EN LÍNEA

- C. Palacios, U. Ramos and A. González, Guía de prevención, diagnóstico y tratamiento de las ITS Dirigida a personal de servicios de salud, (2011), Accesed

- December 2015. [Online]. Available: http://www.censida.salud.gob.mx/descargas/normatividad/guia_preencion_diagnostico_ITS-FEB13CS4.pdf
- (2006) Accessed December 2015. [Online]. Available http://www.censida.salud.gob.mx/descargas/man_org_capasits.pdf 07/12/2015
 - C. Cruz, G. Egremy and U. Ramos, Manual de Capacitadores en el Manejo Sindromático de las Infecciones de Transmisión Sexual (ITS), (2004) Accessed December 2015. [Online]. Available <http://www.salud.gob.mx/unidades/cdi/documentos/DOCSAL7609.pdf>
 - (2014) Respuesta al VIH, Sida e ITS, Accessed December 2015. [Online]. Available http://www.censida.salud.gob.mx/descargas/acerca/PAE_2013_2018_AUTORIZADA.pdf
 - C Magis and H. Barrientos VIH/SIDA y salud pública Manual para personal de salud, (2006), Accessed December 2015. [Online]. Available http://www.censida.salud.gob.mx/descargas/normatividad/manual_personal_salud.pdf
 - A. Wong, Manual Para la Toma, Envío y Recepción de Muestras para Diagnóstico, (2015), Accessed December 2015. [Online]. Available http://www.indre.salud.gob.mx/sites/indre/descargas/pdf/manual_toma_recepcion_muestra_2015.pdf
 - Accessed December 2015. [Online]. Available http://www.censida.salud.gob.mx/descargas/normatividad/manual_usuario_salvar3_5_v2_0.pdf
 - Accessed December 2015. [Online]. Available <http://www.clinicasabortos.mx/enfermedades-de-transmision-sexual/sec12>
 - Accessed December 2015. [Online]. Available http://www.dermocosmos.com/espanol/libros/ETS%20y%20SIDA%201_1.htm
 - Accessed December 2015. [Online]. Available http://www.facmed.unam.mx/sms/seam2k1/2008/ago_01_ponencia.html 08/12/2015
 - Accessed December 2015. [Online]. Available http://vitae.ucv.ve/?module=articulo&n=174_ 14/12/2015

- Accessed December 2015. [Online]. Available: <http://robinclienteservidor.weebly.com/ventajas---desventajas.html>
- Accessed December 2015. [Online]. Available: http://www.alegsa.com.ar/Respuesta/ventajas_y_desventajas_del_modelo_clienteservidor.htm
- P. Ruiz, (2013), Sistemas Operativos en Red. Accessed December 2015. [Online]. Available: <http://somebooks.es/?p=3360>
- Accessed December 2015. [Online]. Available: http://docente.ucol.mx/rcarlos/public_html/cliente-servidor.htm
- Accessed December 2015. [Online]. Available: <http://lctutorial.com/distribucion-bloques-web/>
- (2009), Libro de Estilo del Programa de Guías de Práctica Clínica, Accessed January 2016. [Online]. Available: http://portal.guiasalud.es/libro_estilo/apartado02/arquitectura.html
- J. Lamarca, Diseño de una página Web, Accessed January 2016. [Online]. Available: http://www.hipertexto.info/documentos/diseg_web.htm
- (2016), Accessed January 2016. [Online]. Available: <http://uwe.pst.ifi.lmu.de/teachingTutorialPresentationSpanish.html>
- O. Esgaib and M. Merín, (2009), Accessed January 2016. [Online]. Available: <http://es.slideshare.net/millernegro/aplicando-uwe-un-ejemplo-didactico>
- Accessed December 2015. [Online]. Available: <http://dev.mysql.com/doc/refman/5.7/en/what-is-mysql.html>
- Accessed December 2015. [Online]. Available <http://www.gnu.org/home.es.html>
- Accessed December 2015. [Online]. Available <http://www.revista.unam.mx/vol.4/num2/art3/cluster.htm>
- Accessed December 2015. [Online]. Available http://www.it.uc3m.es/labas/course_notes/dynamic_data_structures_es.html
- L. Casillas, M. Gibert and O. Pérez, Bases de datos en MySQL, Accessed December (2015). [Online]. Available http://ocw.uoc.edu/computer-science-technology-and-multimedia/bases-de-datos/bases-de-datos/P06_M2109_02151.pdf

- Accessed December 2015. [Online]. Available <http://www.tufuncion.com/mysql-innodb-myisam>
- Accessed January 2016. [Online]. Available http://www.gcfaprendelibre.org/tecnologia/curso/informatica_basica/aplicaciones_web_y_todo_acerca_de_la_nube/2.do
- Accessed January 2016. [Online]. Available <https://programacionwebisc.wordpress.com/2-1-arquitectura-de-las-aplicaciones-web>
- Accessed January 2016. [Online]. Available https://books.google.com.mx/books?id=KINC0Gc_RREC&pg=PA446&lpg=PA446&dq=como+escoger+el+mejor+SGBD&source=bl&ots=V2i81AZJOx&sig=cBONjaTViW-X2VQmlU2Za7Jp6cl&hl=es&sa=X&ved=0ahUKEwjA-K7b8qzKAhVN22MKHerB_gQ6AEIQDAG#v=onepage&q&f=false
- (1995) Accessed January 2016. [Online]. Available <http://www.computerworld.es/archive/factores-clave-para-elegir-un-sgbd>
- Accessed January 2016. [Online]. Available <http://www.microsoft.com/es-xl/server-cloud/products/sql-server-editions/overview.aspx>
- Accessed January 2016. [Online]. Available <http://www.postgresql.org/about/>
- Accessed January 2016. [Online]. Available http://dev.mysql.com/tech-resources/articles/move_from_microsoft_SQL_Server.html
- S. Montoro, (2013), Cómo seleccionar una plataforma de desarrollo para un proyecto web Accessed January 2016. [Online]. Available <http://lapastillaroja.net/2013/10/como-seleccionar-plataforma-tecnologica/>
- Accessed January 2016. [Online]. Available [https://msdn.microsoft.com/es-es/library/4w3ex9c2\(v=vs.100\).aspx](https://msdn.microsoft.com/es-es/library/4w3ex9c2(v=vs.100).aspx)
- J. Hanley, Accessed January 2016. [Online]. Available http://www.syllogisticsoftware.com/papers/Web_Development_Technology_Comparison.html
- B. Kohan, Accessed January 2016. [Online]. Available <http://www.comentum.com/php-vs-asp.net-comparison.html>

- Accessed January 2016. [Online]. Available <http://dev.mysql.com/doc/refman/5.7/en/installing.html>
- Accessed January 2016. [Online]. Available http://www.pearsonhighered.com/assets/hip/us/hip_us_pearsonhighered/samplechapter/0672328127.pdf
- (2015), Accessed January 2016. [Online]. Available <http://blog.panel.es/index.php/software-qa-cuales-son-los-tipos-de-pruebas-software/>
- (2014) Accessed January 2016. [Online]. Available http://testeandosoftware.com/istqb-que-es-cuales-son-los-niveles-de-certificacion/_20/01/2016
- (2014), Tipos de pruebas de software definidos por el ISTQB, Accessed January 2016. [Online]. Available <http://www.pmoinformatica.com/2014/01/tipos-de-pruebas-de-software-istqb.html>
- Accessed January 2016. [Online]. Available <http://www.istqb.org/downloads/finish/16/15.html>
- (2010), Accessed January 2016. [Online]. Available https://josepablosarco.files.wordpress.com/2012/10/sstqb_file95-a69acf.pdf
- Accessed January 2016. [Online]. Available <http://en.globetesting.com/glosario/>
- (2013), Accessed December 2015. [Online]. Available <http://es.slideshare.net/dumethvah/pruebas-software-c2>
- J. Sigüenza, Introducción a las Bases de Datos Relacionales, Accessed December 2015. [Online]. Available <http://arantxa.ii.uam.es/~siguenza/Bases%20de%20datos%20I.ppt>
- Accessed December 2015. [Online]. Available <http://cursos.aiu.edu/Base%20de%20Datos/pdf/Tema%204.pdf>
- Accessed December 2015. [Online]. Available <http://www.ual.es/~mtorres/BD/bdtransp3.pdf>
- Accessed December 2015. [Online]. Available <http://www.uazuay.edu.ec/analisis/EI%20modelo%20relacional.pdf>
- Accessed December 2015. [Online]. Available https://es.wikipedia.org/wiki/Base_de_datos_relacional#Caracter.C3.ADsticas
- Accessed January 2016. [Online]. Available <http://uwe.pst.ifi.lmu.de/>

- N. Koch, A.Knapp and G. Zhang, Accesed January 2016. [Online]. Available <http://www.pst.ifi.lmu.de/veroeffentlichungen/uwe.pdf>
- N. Koch, (2007), Accesed January 2016. [Online]. Available <http://es.slideshare.net/adorepump/umlbased-web-engineering-uwe-presentation>
- N. Koch, (2005) Accesed January 2016. [Online]. Available http://www.pst.ifi.lmu.de/personen/kochn/presentations/UWE-sevilla_210405_norakoch.pdf
- D. Rodríguez, (2014), Accesed January 2016. [Online]. Available <http://slideplayer.es/slide/1687281/>
- (2006), Accesed January 2016. [Online]. Available <http://www.uazuay.edu.ec/analisis/Modelo%20Entidad%20Relacion.pdf>

ANEXOS

1. Instalación del SGBD

Una vez logueado en el perfil de administrador:

- Identificar el Sistema Operativo en el que se va realizar la instalación, en este caso se utiliza una PC con Sistema Operativo Windows 7.
- Se descarga el paquete enlistado en el inciso anterior y según el Sistema Operativo con el que se cuente se elige de la siguiente liga: <http://dev.mysql.com/downloads/installer/>

En la **Figura 4.3.1.1** se muestra la pestaña de descargas, se elige “web community” si se cuenta con una conexión a internet mientras se ejecuta el instalador; caso contrario se elige la opción restante.

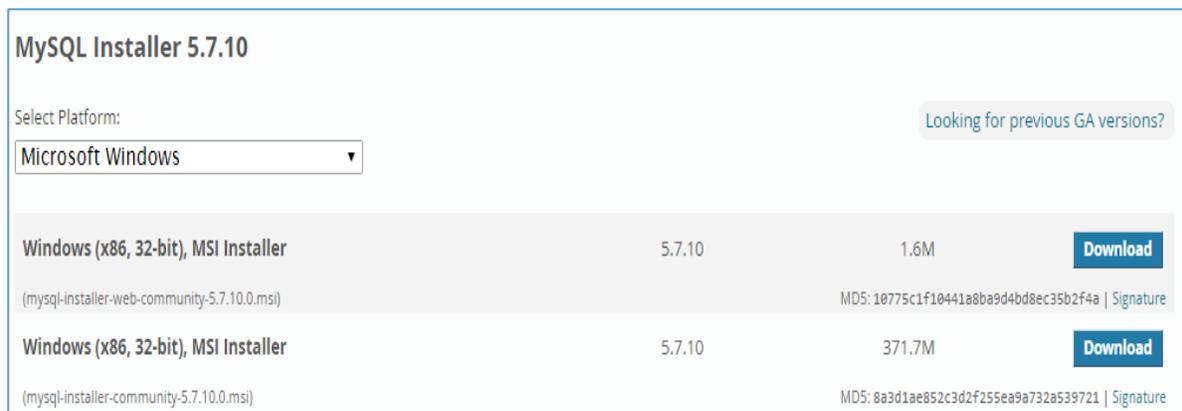


Figura 4.3.1.1 Ventana para la elección del Instalador

El paquete descargado es un instalador que guía paso a paso para la instalación, si se cuenta con otra versión de MySQL éstas pueden coexistir sin ningún problema.

A continuación dar clic en “No thanks, just start my download” como se muestra en la **Figura 4.3.1.2**.

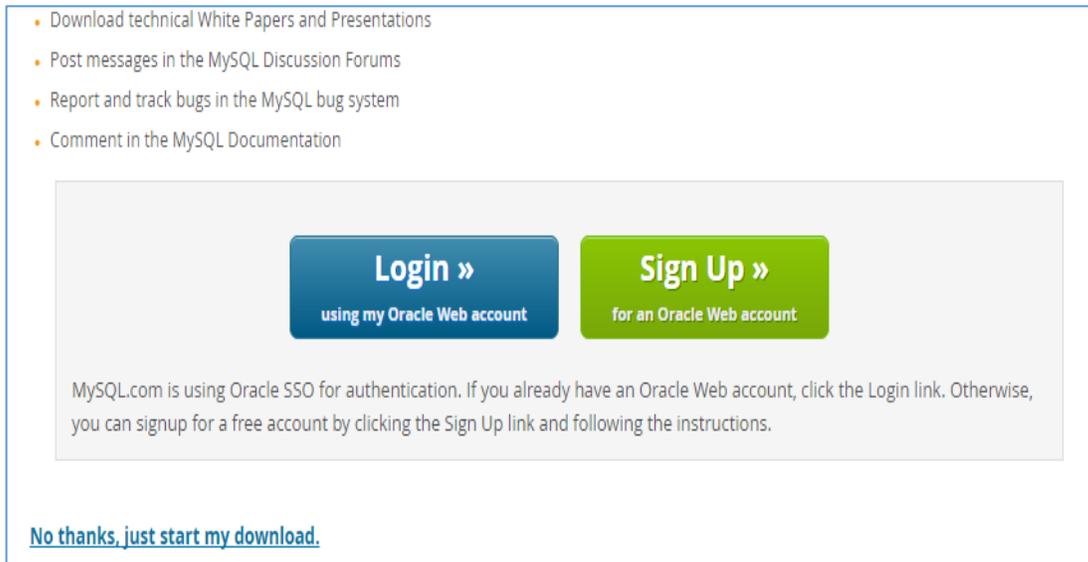


Figura 4.3.1.2 Iniciar descarga.

- Ya que se descargó el instalador se abre el ejecutable y se sigue la guía de instalación paso a paso eligiendo los valores por defecto que ofrece el paquete.
 - Dar clic y seleccionar la opción “Ejecutar” como se muestra en la **Figura 4.3.1.3.**

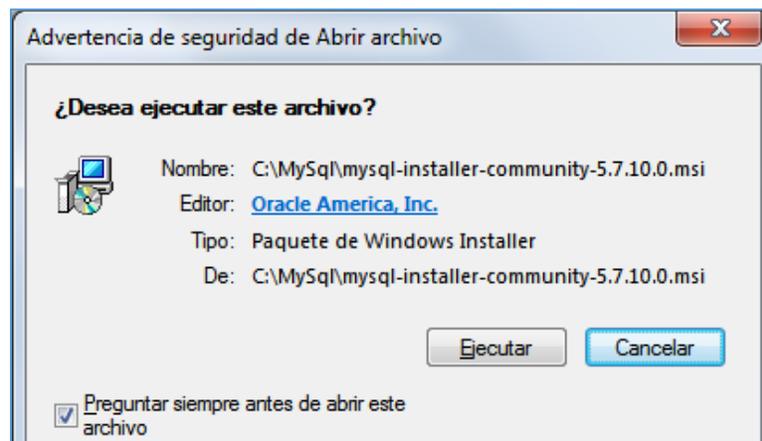


Figura 4.3.1.3 Ventana de ejecución del paquete de instalación.

- Si el sistema solicita algún permiso extra para ejecutar el archivo otorgarlos.
- Aceptar los términos de la Licencia.
- Elegir la instalación típica “Developer Default” y dar en el botón siguiente “next”, como se muestra en la **Figura 4.3.1.4.**

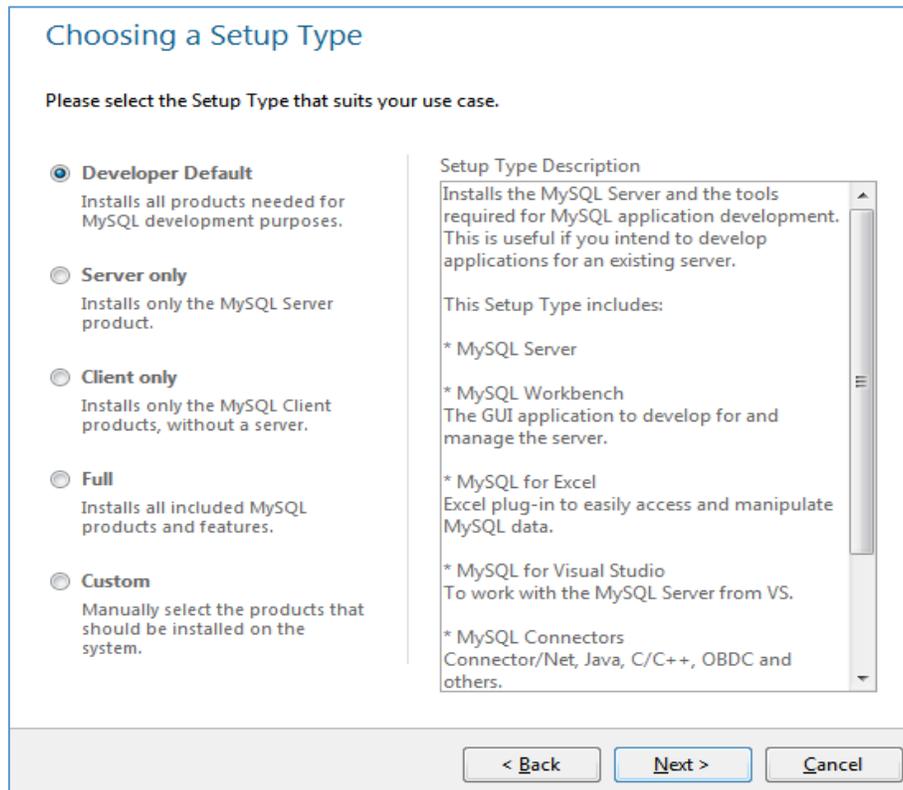


Figura 4.3.1.4 Tipo de instalación.

- Para el caso en que el sistema se desarrolle en .NET, Python, etc., se necesitará tener previamente el framework o bien en la pantalla siguiente como se muestra en la **Figura 4.3.1.5** seleccionar los productos faltantes para que los descargue de la red y dar clic en el botón ejecutar “Execute”, para este caso dichas opciones no son requeridas y sólo damos clic en siguiente “Next”

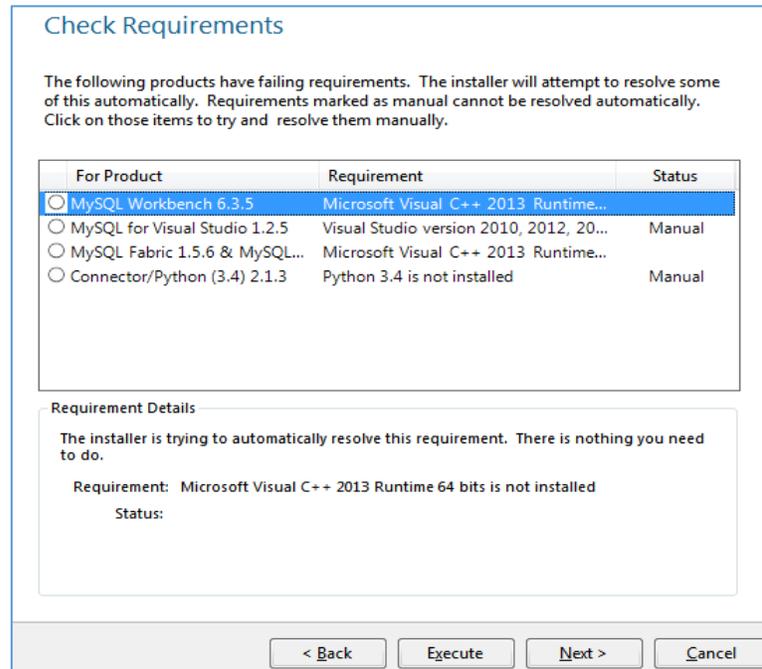


Figura 4.3.1.5 Requerimientos adicionales.

- Al término del asistente de instalación se deberá elegir el cuadro dando “clic” en “Next”, como se muestra en la **Figura 4.3.1.6**.

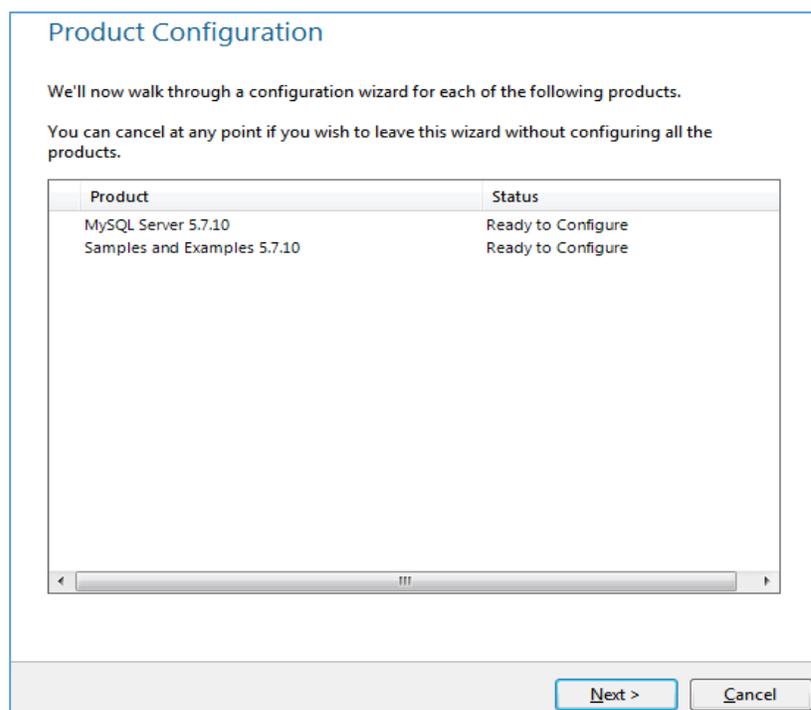


Figura 4.3.1.6 Instalación completada, inicia configuración de servidor de base de datos.

- Continúa con el tipo de configuración y conectividad en la cual elegiremos:
 - “Server Machine” para los equipos donde se accederá a la aplicación.
 - Dejamos los valores por defecto.
 - En ocasiones puede marcar el puerto por defecto como ya ocupado por un servicio, en ese caso se puede elegir alguno que se encuentre libre.

Damos clic en siguiente en el botón “Next” como se muestra en la **Figura 4.3.1.7.**

Type and Networking

Server Configuration Type

Choose the correct server configuration type for this MySQL Server installation. This setting will define how much system resources are assigned to the MySQL Server instance.

Config Type:

Connectivity

Use the following controls to select how you would like to connect to this server.

TCP/IP Port Number:

Open Firewall port for network access

Named Pipe Pipe Name:

Shared Memory Memory Name:

Advanced Configuration

Select the checkbox below to get additional configuration page where you can set advanced options for this server instance.

Show Advanced Options

Figura 4.3.1.7. Tipo de Configuración y Conectividad.

- A continuación se configuran las cuentas y los roles, se debe establecer una contraseña para el usuario *root* (raíz), que será el usuario administrador de la base.
- En la misma ventana seleccionar agregar usuario dando clic en el botón “Add User” y desplegará otra ventana en donde se podrá configurar dicho usuario:

- En el recuadro “*Username*” estableceremos el nombre del usuario en este caso se registra “*Owner*”.
 - En el recuadro “*Host*” se registra “*All Host*” para que pueda tener acceso a todos los servidores que se llegasen a crear en el SGBD.
 - En el recuadro “*Role*” se registra el rol que desempeñará el usuario en este caso se establece “*DB Admin*”
 - En “*Authentication*” seleccionamos la casilla MySQL
 - En “*Password*” establecemos la contraseña con la que se firmará el usuario “*Owner*”
 - Damos clic en “*OK*”
- El rol debería verse agregado como se muestra en la **Figura 4.3.1.8**.

Accounts and Roles

Root Account Password
Enter the password for the root account. Please remember to store this password in a secure place.

MySQL Root Password:

Repeat Password:

Password Strength: **Strong**

MySQL User Accounts
Create MySQL user accounts for your users and applications. Assign a role to the user that consists of a set of privileges.

MySQL Username	Host	User Role
Owner	%	DB Admin

Add User
Edit User
Delete

< Back Next > Cancel

Figura 4.3.1.8 Despliegue de roles.

- El siguiente paso es configurar MySQL como un servicio en Windows, dejando los valores por defecto que marca el asistente damos clic en siguiente en el botón “Next”, como se muestra en la **Figura 4.3.1.9**.

The screenshot shows a 'Windows Service' configuration window. At the top, there is a checked checkbox labeled 'Configure MySQL Server as a Windows Service'. Below this, the 'Windows Service Details' section asks for a unique service name, with 'MySQL57' entered in the text box. Another checked checkbox is 'Start the MySQL Server at System Startup'. The 'Run Windows Service as ...' section has two radio button options: 'Standard System Account' (selected) and 'Custom User'. At the bottom, there are three buttons: '< Back', 'Next >' (highlighted with a dashed border), and 'Cancel'.

Figura 4.3.1.9 Configuración de servicios de Windows.

- Como último paso se aplican las configuraciones y damos clic en ejecutar en el botón “*Execute*”.
- Al finalizar de instalar el paquete del servidor comienza la instalación para la instalación de ejemplos dando clic en siguiente en el botón “Next”.
- La siguiente pantalla mostrará por defecto las credenciales del usuario root para continuar con la instalación en este paso se dará clic en siguiente en el botón “Check”, como se muestra en la **Figura 4.3.1.10**.

Connect To Server

Here are the compatible servers installed. If more than one, please select one.

Server	Architecture	Status	
MySQL Server 5.7.10	X64	Running	

Now give us the credentials we should use (needs to have root privileges). Click check to make sure they work.

User: Credentials provided in Server configuration

Password:

Figura 4.3.1.10 Conectando al servidor para instalar ejemplos.

- Como siguiente paso aplicará la configuración, damos clic en ejecutar en el botón “Execute”, como se muestra en la **Figura 4.3.1.11**.

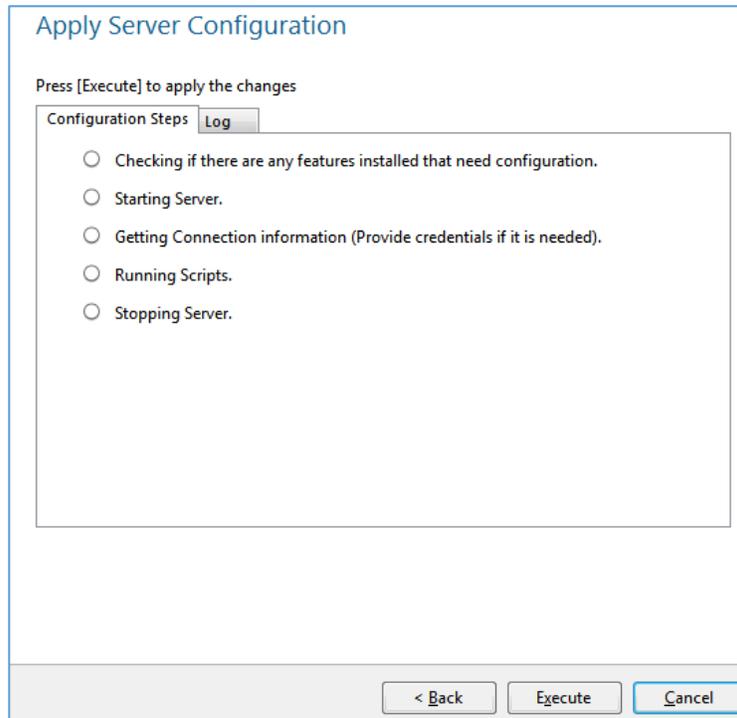


Figura 4.3.1.11 Aplicando configuración para los cambios efectuados.

- Una vez finalizado mostrará una ventana con los paquetes instalados y damos clic en siguiente en el botón “Next”, como se muestra la **Figura 4.3.1.12.**

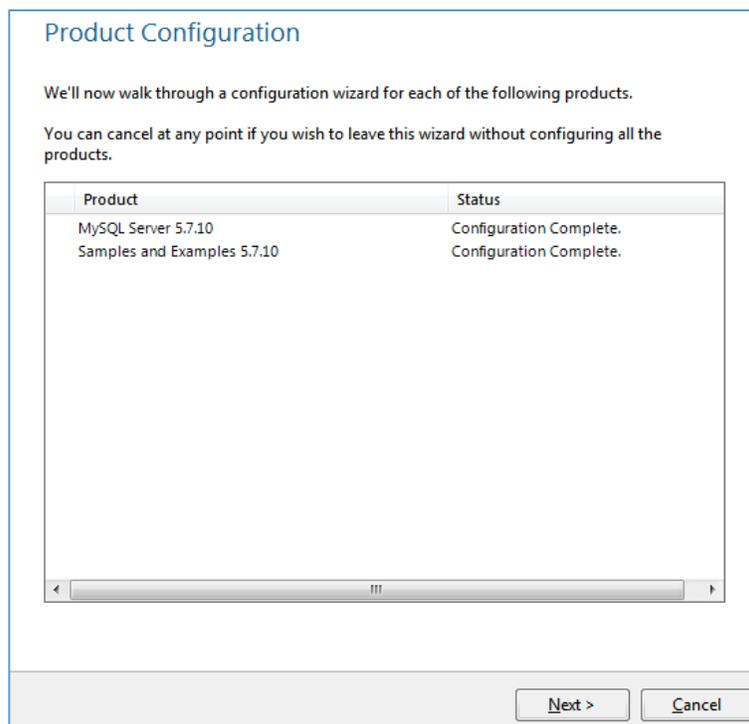
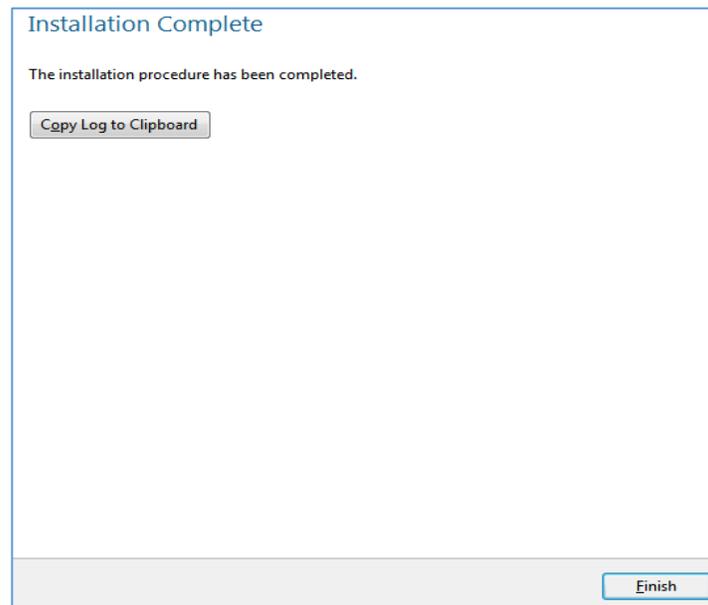


Figura 4.3.1.12 Configuración completada de los paquetes instalados.

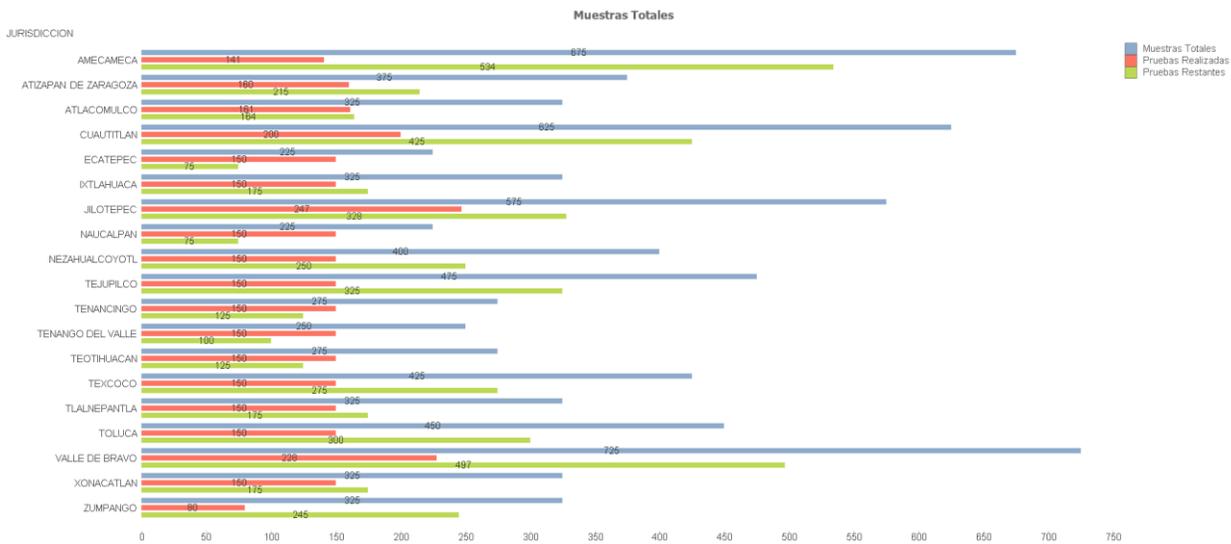
- Para Finalizar muestra la siguiente pantalla y damos clic en finalizar en el botón “Finish”, como se muestra en la **Figura 4.3.1.13**.

**Figura 4.3.1.13 Pantalla de término de la instalación.**

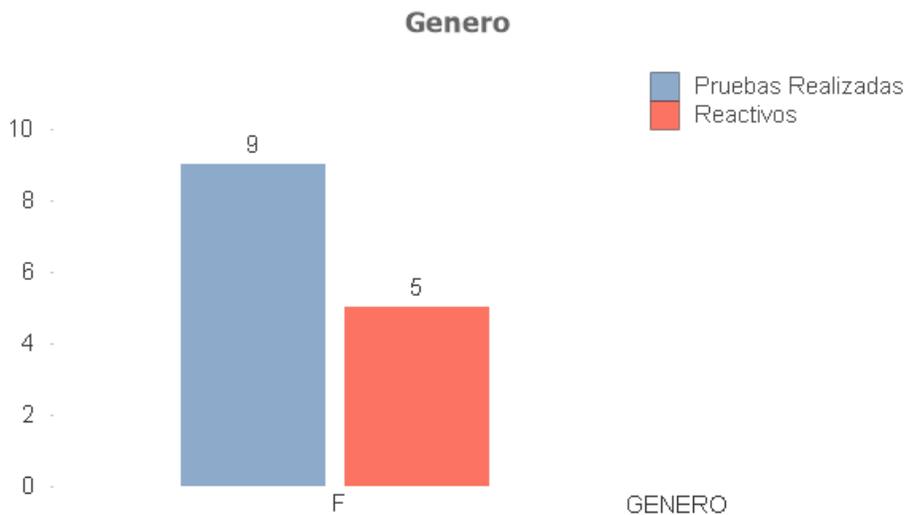
2. Gráficas de Reportes

A continuación se presentan las gráficas generadas a partir de los reportes de la aplicación SysPETS

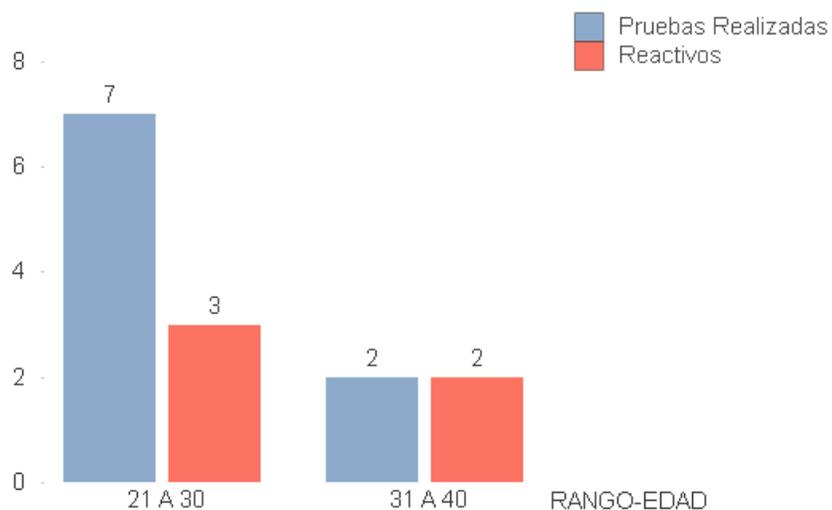
a) Por Jurisdicción



b) Por Género



c) Por Rango de Edad



3. Diccionario de datos

A continuación se enlista el diccionario de datos que compone el sistema.

Tabla: actual

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
ID	varchar(64)	No			
Fumador	varchar(2)	No			
Hipertensión	varchar(2)	No			
Diabetes	varchar(2)	No			
Cáncer	varchar(2)	No			
dislipidemia	varchar(2)	No			
Pruebald	varchar(64)	No			
Embarazo	varchar(2)	No			

marcapasos	varchar(2)	No			
------------	------------	----	--	--	--

Tabla antropo:

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
Id	varchar(64)	No			
Talla	float	No			
Peso	int(4)	No			
Cintura	int(4)	No			
Grasa	int(4)	No			
Pruebald	varchar(64)	No			
Frecuencia	varchar(3)	No			
Sistólica	varchar(3)	No			
Diastólica	varchar(3)	No			
fechaparto	varchar(12)	No			
feharegla	varchar(12)	No			

Tabla capacits:

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
idcapacit	varchar(64)	No			

Nombre	varchar(20)	No			
responsable	varchar(30)	No			
Correo	varchar(35)	No			
Horario	varchar(50)	No			
direccion	varchar(100)	No			
Teléfono	varchar(25)	No			

Tabla cardio:

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
Id	varchar(64)	No			
Sistólica	int(4)	No			
diastolica	int(4)	No			
frecuencia	int(4)	No			
colesterol	int(4)	No			
trigliceridos	int(4)	No			
Glucosa	int(4)	No			
microalbuminuria	int(4)	No			
Ayunas	varchar(2)	No			

diebetico	varchar(2)	No			
Pruebald	varchar(64)	No			

Tabla colonia:

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
CodEstado	varchar(4)	No			
CodMunicipio	int(10)	No			
CodColonia	int(10)	No			
Colonia	varchar(100)	No			
CP	char(5)	No			
CPOficina	char(5)	No			

Tabla consumibles:

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
pedido (<i>Primaria</i>)	int(4)	No			
Id	varchar(64)	No			
Nombre	varchar(30)	No			
Cantidad	int(3)	No			
Kit	varchar(10)	No			

Caducidad	varchar(12)	No			
Enviado	varchar(12)	No			
Recibido	varchar(12)	No			
Lote	varchar(20)	No			

Tabla cuestionario:

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
ID	varchar(64)	No			
Actividad	varchar(4)	No			
Frutasv	varchar(4)	No			
medicacion	varchar(4)	No			
glucosalta	varchar(4)	No			
Diabetes	varchar(4)	No			
Síntoma	varchar(4)	No			
Pruebald	varchar(64)	No			

Tabla estado:

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
CodEstado	varchar(4)	No			
Estado	varchar(100)	No			

Tabla estudio:

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
ID	varchar(64)	No			
Pruebald (<i>Primaria</i>)	varchar(64)	No			
Primera	int(2)	No			
Riesgo	int(2)	No			
Hospital	varchar(2)	No			
Medico	varchar(30)	No			
Diagnostico	Text	No			
Fechatoma	varchar(12)	No			
Capacit	varchar(64)	No			
Vih	int(2)	No			
Sífilis	int(2)	No			

Fechaparto	varchar(12)	No			
------------	-------------	----	--	--	--

Tabla frecuencia:

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
Sexo	varchar(1)	No			
Edadmin	int(3)	No			
Edadmax	int(3)	No			
Frecuenciamin	int(3)	No			
Frecuenciamax	int(3)	No			
Color	varchar(1)	No			
Estado	varchar(10)	No			

Tabla hospital:

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
HOSPID	int(11)	No			
MUNICIPIO	varchar(160)	No			
NOMBRE	varchar(200)	No			
PASSWD	varchar(70)	No			
DIRECCION	varchar(160)	No			

CONTACTO	varchar(160)	Sí	NULL		
CARGO	varchar(160)	Sí	NULL		
TELEFONO	varchar(160)	No			
CELULAR	varchar(160)	Sí	NULL		
RESPONSABLE	varchar(160)	No			
CORREO	varchar(50)	Sí	NULL		

Tabla jurisdicción:

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
ID (<i>Primaria</i>)	int(4)	No			
Clues	varchar(12)	No			
Userid	varchar(64)	No			
Jurisdicción	varchar(30)	No			
Región	int(3)	No			
Contacto	varchar(30)	No			

Tabla municipio:

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
---------	------	------	----------------	-----------	-------------

CodEstado	varchar(4)	No			
CodMunicipio	int(10)	No			
Municipio	varchar(100)	No			

Tabla paciente:

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
ID (<i>Primaria</i>)	varchar(32)	No			
NOMBRE	varchar(50)	No			
APP	varchar(50)	No			
APM	varchar(50)	No			
FECHANAC	date	Sí	<i>NULL</i>		
RFC	varchar(13)	Sí	<i>NULL</i>		
SEXO	varchar(1)	No			
EDAD	varchar(2)	No			
EDOCIVIL	varchar(1)	No			
TELEFONO	varchar(15)	No			
MAIL	varchar(30)	No			
LNACIMIENTO	varchar(50)	No			

FECHAATENCIÓN	date	No			
MARCA	varchar(2)	No			
Ocupación	varchar(30)	No			
Atrabajo	varchar(30)	No			

Tabla prueba:

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
Pruebald	varchar(64)	No			
Fecha	Date	No			
ID	varchar(64)	No			
jurisdiccion	int(2)	No			

Tabla recomendación:

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
ID	varchar(64)	No			
recomendacion	varchar(1000)	No			
Medico	varchar(50)	No			
Cedula	int(10)	No			
Pruebald	varchar(64)	No			

Tabla riesgo:

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
diabetico	varchar(2)	No			
Sexo	varchar(2)	No			
Fuma	varchar(2)	No			
Edad	varchar(2)	No			
Pas	varchar(3)	No			
colesterol	int(3)	No			
Color	varchar(1)	No			

Tabla salud:

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
Id	varchar(32)	No			
Primera	Float	Sí	NULL		
Segunda	Float	Sí	NULL		
Tercera	Float	Sí	NULL		
Marca	varchar(2)	Sí	NULL		
Pruebald	varchar(64)	No			

Tabla saluddiabetes:

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
Id	varchar(32)	No			
Primera	Float	Sí	NULL		
Segunda	Float	Sí	NULL		
Tercera	Float	Sí	NULL		
Marca	varchar(2)	Sí	NULL		
Pruebald	varchar(64)	No			

Tabla tomames:

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
Userid	varchar(64)	No			
Mes	int(2)	No			
Ano	int(4)	No			
Cantidad	int(3)	No			
fechaactual	varchar(12)	No			

Tabla ubica_paciente:

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
ID	varchar(64)	No			
Calle	varchar(60)	No			
Numero	varchar(4)	No			
ext	varchar(2)	No			
estado	varchar(3)	No			
municipio	varchar(3)	No			
colonia	varchar(5)	No			

Tabla usuario:

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
usuarioid (<i>Primaria</i>)	varchar(64)	No			
Usuario	varchar(12)	No			
Nombre	varchar(50)	No			
Contraseña	varchar(64)	No			
Tipo	varchar(1)	No			