



FACULTAD DE INGENIERÍA Y CIENCIAS AGROPECUARIAS

**SISTEMA GRÁFICO DE APOYO AL DIAGNÓSTICO DE PERIODONTITIS
PARA MEJORAR LA ATENCIÓN DE LOS PACIENTES**

Autor: Camilo Hernán Fonseca Cardozo

2013



FACULTAD DE INGENIERÍA Y CIENCIAS AGROPECUARIAS

**SISTEMA GRÁFICO DE APOYO AL DIAGNÓSTICO DE PERIODONTITIS
PARA MEJORAR LA ATENCIÓN DE LOS PACIENTES**

**Trabajo de titulación presentado en conformidad a los requisitos para obtener el
título de Ingeniero en Sistemas de Computación e Informática.**

Profesor Guía

Ing. Juan José León

Autor

Camilo Hernán Fonseca Cardozo

Año

2013

DECLARACIÓN DEL PROFESOR GUÍA

“Declaro haber dirigido este trabajo a través de reuniones periódicas con el estudiante, orientando sus conocimientos para un adecuado desarrollo del tema escogido, y dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación.”

Juan José León

Ingeniero de Sistemas

C.C.:

DECLARACIÓN DE AUTORÍA DEL ESTUDIANTE

“Declaro que este trabajo es original, de mi autoría, que se han citado las fuentes correspondientes y que en su ejecución se respetaron las disposiciones legales que protegen los derechos de autor vigentes.”

Camilo Hernán Fonseca Cardozo

C.C.: 1713439543

AGRADECIMIENTO

El agradecimiento del presente proyecto es especialmente para mis padres y esposa por el apoyo incondicional durante todo este tiempo.

DEDICATORIA

Dedicado a mis padres y esposa

RESUMEN

El presente trabajo representa la organización de información de un paciente para la especialidad de periodoncia dentro de la rama médica de la odontología, con el sistema desarrollado se podrá administrar de una forma más clara y precisa los datos de periodontogramas tomados en cada paciente. La información es administrada por el sistema de tal forma que todas las acciones de cada paciente realizadas en el consultorio dental como consultas, medicamentos, evaluaciones y toma de sondajes se conservan en un perfil que permite brindarle al paciente un mejor servicio. Así mismo los cálculos y gráficos realizados manualmente por los periodoncistas son hechos directamente por el sistema generando un diagnóstico directo y ahorrando tiempo que será utilizado en atención a más pacientes.

ABSTRACT

This project involves the organization of patient information for periodontics, a specialty within the medical branch of dentistry. By using this system, the information taken from periodontal charts can be administered in a more clear and precise way. The system manages the data so that all activity related to a patient in a dental office, such as appointments, medications, assessments and drillings, are stored in a profile, resulting in better service for the patient. In addition, the system can produce calculations and graphics that periodontists have typically done manually. This improvement will generate direct diagnosis and save time that can be used to care for more patients.

INDICE

1. Capítulo I. Introducción	1
1.1 Antecedentes	2
1.2 La enfermedad periodontal	3
1.3 Necesidades médicas	3
1.4 Situación actual de los gráficos	4
1.5 Obtención de los datos	5
2. Capítulo II. Metodología Scrum	6
2.1 Desarrollo ideal de un proyecto de software	6
2.1.1 Problemas en los proyectos de desarrollo de software	6
2.1.2 Metodologías tradicionales de desarrollo	7
2.1.3 Una alternativa ágil para el desarrollo de software	8
2.1.4 Orígenes de la metodología ágil Scrum	9
2.2. Scrum	9
2.2.1 El equipo Scrum	10
2.2.1.1 El cliente	11
2.2.1.2 Scrum Master	11
2.2.1.3 Equipo de desarrollo	12
2.3 Historias de usuario	13
2.4 Lista de objetivos	14
2.4.1 Elementos de la lista de objetivos	14
2.4.2 Priorización de objetivos	18
2.4.3 Velocidad del equipo en objetivos completados	18
2.4.5 Re-priorización de los objetivos debido a factores económicos	18
2.5 Sprint	19
2.5.1 Planificación de un sprint	21
2.5.1.1 Sprint Backlog	22
2.5.1.2 Elementos del Sprint Backlog	22
2.5.1.3 Elaboración del Sprint Backlog	23
2.5.1.4 Burndown chart	24

2.5.2 Reunión de sincronización	25
2.5.2.1 ¿Qué he hecho?	25
2.5.2.2 ¿Qué voy a hacer?	25
2.5.2.3 ¿Qué impedimentos tengo?	25
2.5.3 Beneficio de la reunión de sincronización	26
2.5.4 Restricciones de la reunión de sincronización	26
2.5.5 Retrospectiva del Sprint	27
2.6 Scrum vs metodologías tradicionales	27
2.7 Beneficios de utilizar Scrum	31
2.8 Mejores prácticas de Scrum	32
2.9 Porque utilizar Scrum para el presente proyecto	33
2.10 Empresas que han usado Scrum	33
3. Capítulo III. Tecnología	35
3.1 El lenguaje PHP	35
3.1.1 Características de PHP	36
3.1.2 Base para el uso de PHP	37
3.1.3 ¿Cómo trabaja PHP?	37
3.1.3.1 El servidor Web	38
3.1.3.2 Características de un servidor Web	39
3.1.3.3 Servidores web disponibles en la actualidad	40
3.1.4 Pasos para el funcionamiento de PHP	40
3.1.5 Requisitos para trabajar con PHP	40
3.2 Apache HTTP Server	41
3.2.1 Características de Apache HTTP Server	41
3.2.2 Ventajas de utilizar Apache HTTP Server	42
3.2.3 Popularidad de Apache HTTP Server	42
3.2.4 Funcionamiento de Apache Server	43
3.2.4.1 Arquitectura de Apache Server	44
3.2.4.2 Módulos de Apache HTTP Server	45
3.2.5 Requisitos para el funcionamiento de Apache en la plataforma Windows	46
3.2.6 Instalación para Windows	46

3.3 MySQL	49
3.3.1 ¿Por qué utilizar MySQL?	49
3.3.2 Características de MySQL	49
3.3.3 Usos de MySQL	50
3.3.4 Requerimientos para instalación de MySQL en plataforma Windows	50
3.3.5 Tipos de datos en MySQL	51
3.4 El servidor XAMPP	52
3.4.1 Características del servidor XAMPP	52
3.4.2 ¿Por qué utilizar el servidor XAMPP?	53
3.5 HTML5	53
3.5.1 Características de HTML 5	53
3.5.2 ¿En qué proyectos se utiliza HTML5?	54
3.5.3 Ventajas de HTML5	54
3.5.4 Funciones principales de HTML5	54
3.5.4.1 Navegadores compatibles con HTML5	55
3.5.5 ¿Por qué utilizar HTML5?	56
3.6 Frameworks web	56
3.6.1 Ventajas de utilizar frameworks	56
3.6.2 Tipos de framework	57
3.6.2.1 Características de los frameworks	58
3.6.2.2 Ejemplos de framework	59
3.7 Patrón MVC	59
3.7.1 Modelo	60
3.7.2 Vista	61
3.7.3 Controlador	61
3.7.4 Características del patrón MVC	62
3.7.4.1 Orientación a Objetos con MVC	62
3.7.5 Beneficios de utilizar MVC	63
3.7.6 Flujo en un patrón MVC	64
3.7.7 Creación de código sin MVC	65
3.8 Yii Framework	66

3.8.1 Características de Yii	66
3.8.2 Requerimientos para usar Yii	67
3.8.3 Ventajas de Yii	67
3.8.4 Versionamiento de Yii	68
3.8.5 Comparativa de Yii contra otros frameworks populares	69
3.9 Eclipse PDT	70
3.9.1 Características Eclipse PDT	70
3.9.2 Requisitos de instalación para Eclipse PDT	71
3.9.3 Compatibilidad con Yii	71
4. Capítulo IV. Desarrollo de la solución	72
4.1 Metodología Scrum en el proyecto	72
4.1.1 Equipo de trabajo	72
4.1.2 Descripción de los roles	72
4.1.2.1 Product Owner	72
4.1.2.1 Scrum Master	72
4.1.2.3 Equipo de desarrollo	73
4.2 Inicio del proyecto	73
4.2.1 Actividades previas al arranque del proyecto	73
4.3 Sprint 0	73
4.4 Requerimientos de MF Dental	73
4.4.1 Requisitos no funcionales	77
4.5 Historias de Usuario	77
4.5.1 Actores del sistema	79
4.5.2 Priorización de historias de usuario	79
4.6 Product Backlog del Sistema	81
4.7 Duración de cada sprint	82
4.7.1 Tareas de Sprint 0	83
4.7.2 Tareas de Sprint 1	84
4.7.3 Tareas de Sprint 2	86
4.7.4 Tareas de Sprint 3	87
4.7.5 Tareas de Sprint 4	88

4.8	Parámetros de calidad	89
4.9	Nombre del Sistema	90
4.10	Instalación de Aplicaciones de Desarrollo	90
4.10.1	Sistema Operativo	90
4.10.2	Aplicaciones de Desarrollo	90
4.11	Diseño de Bases de Datos	91
4.11.1	Modelo Conceptual de Base de Datos	91
4.11.2	Modelo Físico de Base de Datos	92
4.11.4	Generación de Script para Creación de Base de Datos	97
4.11.5	Carga de Base de Datos	97
4.12	Diagrama DFD Nivel 0	98
4.12.1	Diagrama DFD Nivel 1	99
4.12.2	Módulos Identificados	100
4.13	Análisis de Procesos	101
4.13.2	Gestión de Usuario del Sistema	102
4.13.3	Creación Actualización y Eliminación de Usuarios	102
4.13.4	Asignar Perfil de Usuario	103
4.13.5	Registro de Paciente	103
4.13.6	Creación, Eliminación y Actualización de Pacientes	104
4.13.7	Validar Información de Registro	104
4.13.8	Registro de Historial Clínico	105
4.13.9	Creación, Actualización o Eliminación de Historia Clínica	105
4.13.10	Validar Información de Registro de Historial	106
4.13.11	Ingreso o Actualización de Odontograma	106
4.14	Diagrama de Clases	107
4.15	Creación del proyecto SAGAP	115
4.15.1	Generación de Modelos del Sistema	116
4.16	Prototipos de Interfaces	116
4.17	Diseño de Pantallas	122
5.	Capítulo 5. Conclusiones y Recomendaciones	127

5.1 Conclusiones	127
5.2 Recomendaciones	129
REFERENCIAS	131
ANEXOS	143

CAPITULO 1

INTRODUCCIÓN

En el Ecuador las empresas cada vez optan más por desarrollar sistemas que sean capaces de mejorar el servicio ofrecen, los sistemas son capaces de administrar el negocio adaptándose a las necesidades que tiene la empresa. Para la ortodoncia que señala como “especialidad de la odontología que se ocupa del desarrollo, la prevención y la corrección de las irregularidades de los dientes, la mordida y los maxilares” (Odontologicosanpablo, s.f.), los sistemas pueden entrar dentro de muchos campos de desarrollo, ya sea para la gestión de talento humano, administración de finanzas, seguimiento de clientes y hasta para obtener medios gráficos en los que los odontólogos de apoyo para realizar sus cirugías.

Para proponer una solución de software fue necesario mantener entrevistas con el personal de MF Dental, con lo cual se pudo observar que toda la información y datos de pacientes se los lleva de forma poco adecuada tomando en cuenta la magnitud de la empresa y la importancia de los datos. En cuanto a las falencias en la administración de información se pudo apreciar lo siguiente.

- Hojas de Excel para guardar información confidencial del paciente.
- Formatos para tomar datos de la enfermedad a tratar deteriorados.
- Calculo de valores para determinación de niveles de enfermedad realizados con calculadora, cabe destacar que los datos son 192 valores de toma de datos y 96 de cálculos.
- Lentitud para determinar los cálculos.
- Tendencia al error por manipulación manual.
- Graficación de líneas de nivel de enfermedad realizadas a mano alzada.

La información clave de los pacientes podría perderse si se la sigue manejando de la misma manera, se realizaron entrevistas con el personal de MF Dental documentándolas en video y verificando archivos de trabajo propios de la empresa para poder entender mejor los métodos de administración de datos.

Es por eso que el presente proyecto de investigación propone una solución enfocada a fines médicos especialmente en el área de la odontología, la enfermedad a tratar se llama periodontitis que es la inflamación de las encías destruyendo tejido y hueso del aparato estomatognático que se refiere al “conjunto de órganos y tejidos ubicados en la cavidad oral como dientes, encías, huesos, órganos, glándulas, lengua, etc.” (Wikipedia, s.f.), en el caso que no se trate a tiempo la enfermedad, las encías se destruyen a tal punto que las piezas dentales no pueden sostenerse en el tejido deteriorado y como consecuencia el desprendimiento de dicha pieza dental.

El sistema a ser desarrollado comprende crear un medio para que los periodoncistas que son “especialistas en el campo de la odontología responsables del cuidado y la prevención de las enfermedades periodontales, la regeneración guiada del hueso y los implantes dentales” (Odontologicosanpablo, s.f.), puedan generar gráficos bidimensionales y que muestre a manera de apoyo, la gravedad de la enfermedad en las encías de los pacientes, visualizado en el sistema hasta qué punto la enfermedad ha avanzado, esto a fin de realizar un diagnóstico que permita ofrecer un tratamiento adecuado, que se ajuste a los perfiles de cada paciente.

1.1 Antecedentes

La empresa en la cual se implementará el sistema se llama MF Dental, se encuentra en la dirección Edificio Fortune Plaza, 4° piso oficina 407, con una antigüedad de 18 años, MF dental brinda salud odontológica en periodoncia cuya misión es prestar atención periodontal y como visión en todas sus áreas es brindar una salud estomatológica completa a la comunidad.

MF Dental tiene un total aproximado anual de 1880 tratamientos. Existe una guía de precios basada en los materiales que el especialista necesita y el tipo de tratamiento en cada diagnóstico. Los tratamientos que el periodoncista puede determinar comprenden medicación y en algunos casos cirugías, en el caso de cirugías el sistema crea un gráfico de apoyo para el odontólogo, indicando los sectores de las encías en el que se procederá a eliminar la enfermedad

quirúrgicamente. El periodoncista tiene que examinar cada pieza dental de la boca a fin de obtener el nivel que la enfermedad ha alcanzado en cada sector de las encías, ésta información es ingresada al sistema, el cual generará un mapa de todas las piezas dentales del paciente señalando los niveles de enfermedad por cada sector.

El sistema permite guardar el tratamiento aplicado a cada cita médica del paciente, creando un perfil en el que se archiva toda la información de su historial médico.

1.2 La enfermedad periodontal

La enfermedad periodontal es una enfermedad que afecta las encías y las estructuras de soporte de sus dientes. La etapa más temprana de la enfermedad periodontal es la gingivitis y se caracteriza por el enrojecimiento de las encías, que se inflaman y sangran fácilmente. Existen muy pocas molestias en esta etapa, pero en tal caso el peligro está latente. Si no se trata en este punto, la enfermedad periodontal puede progresar hacia periodontitis, donde hay daños irreversibles en las encías y en las estructuras de soporte de los dientes.

La periodontitis es la etapa más avanzada de la enfermedad periodontal, el hueso y los tejidos que circundan los dientes se vuelven móviles o "suelos", haciendo que éstos eventualmente se caigan.

1.3 Necesidades médicas

Normalmente los periodoncistas tienen una plantilla estándar del conjunto de piezas dentales de cada paciente, obtienen los datos de la enfermedad en cada pieza dental y en base a esta información dibujan estos diagramas a mano, para luego diagnosticar un tratamiento específico de acuerdo al tipo de gráfico que se obtuvo.

En la plantilla están prediseñados espacios para tomar apuntes de 3 áreas de cada pieza dental, luego mediante cálculos matemáticos obtienen un cuarto valor por registro en cada pieza dental, el cual será el indicador del avance de la enfermedad. Una vez obtenido el valor referencial el periodoncista dibuja las

líneas en la plantilla, estas líneas servirán para ayudar al periodoncista a diagnosticar qué tipo de tratamiento debe utilizar para cada paciente, al mismo tiempo las líneas dibujadas son el apoyo y guía en el caso que el periodoncista decida realizar una intervención quirúrgica. El problema de tener una plantilla estándar es que cada paciente es diferente y los gráficos dibujados a mano por un periodoncista son susceptibles a errores humanos y que conlleven eventualmente a un diagnóstico no adecuado de la enfermedad, no ajustándose a las necesidades del paciente.

Este sistema se implementará de una forma tal que los periodoncistas simplemente ingresen los valores que toman del paciente directamente en la aplicación, la cual determinará el grado de enfermedad de cada paciente.

1.4 Situación actual de los gráficos

Cada pieza dental está formada por 3 zonas que son corona “Sección de la pieza dental que es visible fuera de la encía, está recubierta por una capa de protección llamada esmalte” (Odontologicosanpablo, s.f.), raíz “soporte del diente, se encuentra recubierta por dentina y cemento y se encuentra adherida en la cavidad alveolar dentro de las encías” (Odontologicosanpablo, s.f.) y cuello “Es la unión marcada entre la corona y la raíz de una pieza dental” (Odontologicosanpablo, s.f.), los periodoncistas dibujan la línea indicadora del nivel de enfermedad con un diferente nivel del cuello en cada pieza dental de forma irregular y desnivelada, lo óptimo sería que el cuello se encuentre en un mismo nivel en toda la dentadura, y a su vez cada pieza dental tendrá su propio diagnóstico e ingreso de valores, este valor es de 3 dígitos, representativo del avance de la enfermedad en las 3 secciones distal “área de la pieza dental que encuentra con referencia hacia las muelas” (Wikipedia, s.f.), medio “área de la pieza dental que encuentra con referencia hacia las muelas” (Wikipedia, s.f.) y mesial “área que está entre la superficie distal y mesial de cada diente o muela” (Wikipedia, s.f.), el sistema realizará cálculos matemáticos en base a los dígitos ingresados a fin de generar el gráfico de las líneas guía para determinar el diagnóstico de la enfermedad.

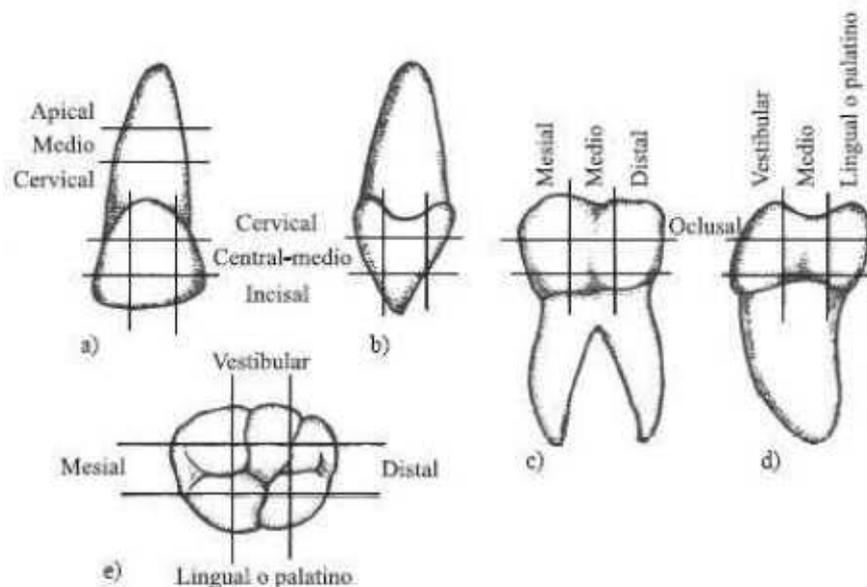


Figura 1. División de áreas en las diferentes piezas dentales.
Tomado de (Mobypicture, 2012).

1.5 Obtención de los datos

En la plantilla existen líneas que marcan niveles superiores e inferiores de una pieza dental tomando como punto central el cuello, las líneas tienen una separación de un milímetro, hacia el extremo de la corona se dibujan 15 líneas y hacia el extremo de la raíz se dibujan 20 líneas, el periodoncista examina cada pieza dental obteniendo un valor numérico, así el diagnóstico se da de acuerdo al avance de la enfermedad en cada pieza, los niveles son: Leve: 1 a 2 milímetros. Moderada: 2 a 4 milímetros; Severa: 5 milímetros en adelante.

Para diferenciar los distintos elementos que tiene la dentadura, los periodoncistas dibujan una cruz para representar que no existe un diente, o formas gráficas que representen un perno; el sistema podrá diferenciar gráficamente cada una de las piezas dentales, es decir, que si un paciente tiene pernos, tornillos, implantes o si no existe diente, se podrá agregar estos objetos a la dentadura virtual de cada usuario expresado en su perfil obteniendo un gráfico más real adaptado a cada paciente.

CAPITULO 2

METODOLOGÍA SCRUM

2.1 Desarrollo ideal de un proyecto de software

En los primeros pasos de un proyecto de desarrollo de software, existe una visión ideal de cómo se va a desarrollar el proyecto, esto teóricamente significa que inicialmente el cliente sabe exactamente qué es lo que quiere, conoce perfectamente sus necesidades y cuenta con un tiempo razonable para que el proyecto sea desarrollado. También el desarrollador del producto piensa que entiende al cien por ciento que es lo que necesita el cliente, y además sabe cómo realizarlo, se realiza el levantamiento de los requerimientos y tanto cliente como desarrollador no se volverán a ver hasta la entrega del producto final.

En un ambiente ideal el equipo de desarrollo realiza la documentación del proyecto, verifica que todo se cumple en los plazos establecidos y no piensa en preocupaciones porque todo el desarrollo está yendo bien como lo acordaron, el proyecto no ha cambiado durante el camino y al final se entrega el producto con la satisfacción del cliente que recibe exactamente lo que necesitaba y del desarrollador ya que no hay la necesidad de realizar cambios. Lamentablemente lo ideal no siempre es lo real.

2.1.1 Problemas en los proyectos de desarrollo de software

¿Al final del proyecto el cliente realmente se encuentra satisfecho con el producto entregado? La realidad del desarrollo de software es muy distante a la visión ideal, desde el inicio del proyecto el cliente muchas veces no sabe exactamente que producto qué es lo que desea obtener y peor aún, si realmente lo que pide es lo que necesita, simplemente sabe que tiene un problema y no tiene una idea clara del tiempo que al equipo de desarrollo le toma crear su solución.

Por otro lado, el desarrollador tiene que empaparse mucho con el negocio de tal forma que pueda comprender su núcleo, a fin de entender las necesidades reales del cliente y así poder brindarle una solución práctica, muchas veces esto tampoco se cumple, por ejemplo en la etapa de levantamiento de requerimientos

con el cliente en ocasiones quedan cabos sueltos que se detectan en la marcha del proyecto, estos cabos son los responsables para que el equipo de desarrollo tenga que realizar cambios durante la creación del producto, haciendo que el equipo tenga que volver unos pasos para enmendar el cabo suelto detectado, esto produce un efecto incremental en tiempos de entrega ya que por cada requerimiento no detectado al inicio, afectando los plazos de entrega del producto.

Muchas veces el producto ha cambiado tanto durante el desarrollo, que desencadena una serie de factores como falta de motivación del equipo de desarrollo, en otras ocasiones el cliente pide los cambios al final del proyecto complicando aún más la entrega ya que no existe mucho tiempo para arreglar las deficiencias y además el desarrollador está con cada vez menos presupuesto.

Para MF Dental es indispensable poder mantener todos los datos de los pacientes de forma ordenada, cada historia clínica se crea en fichas con un formato estandarizado por el gobierno nacional y guardadas en carpetas dentro de un archivador, algunos datos se mantienen en hojas de trabajo en Excel, se imprime una hoja para cada paciente para conservar los datos que sean necesarios, y para los gráficos, existe un formato dibujado a mano el mismo que es copiado para cada paciente y se lo utiliza como muestra del avance o retroceso de la enfermedad periodontal.

MF Dental nos podría sugerir cambios en los objetivos del proyecto durante la construcción del mismo, es por eso que es muy útil Scrum debido a que puede cubrir deficiencias en la determinación de objetivos, puede que algunas de las funcionalidades que se crearon inicialmente en el producto al final no son indispensables para cubrir las necesidades de la empresa.

2.1.2 Metodologías tradicionales de desarrollo

Las metodologías de desarrollo son un conjunto de reglas y métodos utilizados para el desarrollo de software y sirven para producir software de una manera ordenada. Las metodologías tradicionales han sido adecuadas y eficientes

durante mucho tiempo para proyectos muy grandes, pero debido a que actualmente las necesidades para el desarrollo de software son muy cambiantes, los plazos cada vez son más cortos, el producto tiene que recurrir a nueva tecnología y los equipos de trabajo son limitados, estas metodologías no han resultado un modelo muy adecuado en la actualidad. Una forma real de ver esto en las metodologías tradicionales es cuando se emplea mucho tiempo en pruebas después que la totalidad del producto ya ha sido desarrollado, en algunas ocasiones el cliente puede requerir un cambio durante esta fase, dando origen en el peor de los casos a no poder dar vuelta atrás.

Otro problema muy frecuente es que las metodologías tradicionales son muy estrictas en su definición como roles, actividades, modelado, documentación, y además existe tal cantidad de tareas por cumplir que estos mismos factores son los que conllevan a que la metodología se estanque con en su propio ritmo cayendo en los tan temidos retrasos.

2.1.3 Una alternativa ágil para el desarrollo de software

El aparecimiento de nuevas metodologías vienen en los últimos años como solución a las falencias que los métodos tradicionales han generado, éstas nuevas soluciones son conocidas como metodologías ágiles, que son metodologías creadas para cubrir las necesidades de constantes cambios que pueden darse durante el desarrollo del software, una de las particularidades es que no son muy enfocadas a la documentación, reduciéndola solo a lo que es necesario e importante registrar como documento.

La documentación solo es una pequeña diferencia de las metodologías ágiles, su naturaleza es adaptarse a cambios que puedan surgir sobre la marcha del proyecto, también son más enfocados a la motivación del equipo de desarrollo, teniendo en cuenta que el proceso apoya al grupo de desarrolladores en lugar de limitar sus capacidades, haciendo así que el desarrollo se convierta en una actividad grata para el equipo, éstas metodologías tienen una constante interacción con el cliente.

2.1.4 Orígenes de la metodología ágil Scrum

El origen de Scrum viene a mediados de los años 80 por Hirotaka Kateuchi e Ikujiro Nonaka, el surgimiento de Scrum no fue precisamente para el desarrollo de software más bien se enfocaba como una alternativa para la creación de productos tecnológicos, esto debido a que necesitaban poder desarrollar productos con rapidez, con requisitos difusos y flexibilidad a cambios durante la formación del producto.

Estas situaciones se incorporaron al desarrollo de productos de software, es por esto que en 1993 Jeff Sutherland aportó con Scrum a la creación de software para videojuegos, en 1996 junto se formalizó la metodología Scrum y en 2001 pasaron a formar parte de las metodologías ágiles con Agile Alliance "Organización sin fines comerciales encargada de promover el desarrollo de software ágil" (Agilealliance, s.f.).

2.2. Scrum

La metodología Scrum no es una metodología compleja pero si requiere de trabajo fuerte y en equipo, tampoco fija sus metas en un plan de trabajo, más bien se adapta a las circunstancias que conllevan la evolución de cada producto. Básicamente consiste en dividir todo el desarrollo de un producto de software en pequeñas partes de desarrollo llamadas carreras cortas o iteraciones, cada una de las iteraciones tienen un tiempo de duración de 2 a 4 semanas. Son entregas parciales de un producto final dando prioridad desde el inicio al beneficio del cliente.

Scrum se orienta a realizar una gestión de desarrollo de forma incremental, es decir, cada ciclo de iteración terminado es una porción de software funcionando que se incrementa en el producto final. Scrum es iterativo porque se basa en reuniones de seguimiento diarias y si existe alguna necesidad adicional o cambio en el proyecto de parte del cliente, se planifica en las reuniones de seguimiento y se aplican a tiempo los cambios requeridos.



Este procedimiento se realiza utilizando la gestión del trabajo en equipo de una forma colaborativa con el fin de obtener un proyecto eficiente en el menor tiempo posible.

2.2.1 El equipo Scrum

Scrum involucra 3 actores y son indispensables para el correcto funcionamiento de la metodología, la propuesta ágil se basa en que se debe obtener resultados con el menor esfuerzo posible por esta razón se ha reducido el número de involucrados en el proyecto.

- Cliente
- Facilitador "Scrum Master"
- Equipo de desarrollo

2.2.1.1 El cliente

El cliente “*product owner*” es una sola persona y es el responsable de decidir qué es lo que la empresa necesita, el representa al grupo de personas involucradas dentro de la empresa que previamente evaluaron obtener un producto para beneficio de la empresa, el cliente también coexiste con el equipo de trabajo para:

- Determinar que producto tiene que ser entregado al final del proyecto.
- Decidir la priorización de objetivos.
- Controlar los cambios que se realizarán a lo largo del proyecto si amerita realizarlos.
- Trabajar con el equipo de desarrollo en la reunión de planificación del sprint.
- Ser responsable de parar el sprint en el caso que sea necesario.
- Controlar los costos del proyecto y de cada sprint.
- Tener una buena comunicación con el equipo de desarrollo.
- Tener claridad en cuanto a los objetivos que desea obtener en cuanto al producto de software.

2.2.1.2 Scrum Master

Es el líder del equipo de desarrollo llamado facilitador, hace que el proyecto se encamine por la práctica de Scrum, sus responsabilidades son:

- Creación de ítems dentro de la lista de objetivos “product backlog”.
- Buscar las mejores prácticas para el desarrollo del producto.
- Gestiona las necesidades del proyecto.
- Ayuda a controlar el desarrollo del producto con la visión de cómo va a ser el resultado final.
- Organiza al equipo de desarrollo ayudando a que las funciones sean distribuidas a través de todos sus miembros.
- Dirige las reuniones de sincronización enfocándolas a que se traten los temas propuestos objetivamente sin que el equipo salga del contexto a tratar.
- Busca soluciones a los problemas que pueda tener el equipo.
- Ayuda a entender a los patrocinadores que se refiere a “entidades que

aportan económicamente al desarrollo del proyecto” (Scrum, s.f.), prácticas de Scrum.

- Dirige la reunión de planificación del sprint con el cliente, tratando de buscar las mejores vías para que el sprint sea completado correctamente.

2.2.1.3 Equipo de desarrollo

Es un grupo de profesionales que son los responsables de la creación del producto de software solicitado. El equipo debe ser de entre 5 y 9 integrantes debido a que Scrum ha desarrollado requisitos que faciliten el correcto desempeño de un equipo de trabajo.

La experiencia ha hecho que Scrum tenga un mínimo de 5 personas, el equipo no se podría sostener en el caso de que existe alguna eventualidad en la que no se pueda contar más con algún miembro del equipo, causando el no cumplimiento de objetivos, y cuando el equipo tiene más de 9 integrantes el proyecto empieza a tener falencias de comunicación entre sus miembros.

Otras características que tiene un equipo de desarrollo Scrum son:

- El equipo debe ser organizado.
- La forma de trabajo debe ser de forma unificada.
- No existen sub equipos que se encarguen de realizar otras actividades relacionadas.
- El equipo debe comportarse de manera casi independiente de factores externos al grupo.
- El equipo debe trabajar de manera estable, de ser necesario no debería haber cambios en los miembros del equipo.
- El trabajo que el equipo realiza debe ser realizado en el mismo lugar físico.
- Cualidades de un miembro del equipo:
- Conocer las habilidades del resto de sus integrantes son claves para el desarrollo incremental del producto.
- Cada miembro deberá tener habilidades especializadas que pertenece a todo el equipo.
- Compromiso con el proyecto.

- No estar involucrado en otros proyectos de desarrollo.
- Confianza entre miembros del equipo.
- Compartir información entre los integrantes del equipo.

Scrum trata de enfocar el desarrollo de relaciones interpersonales entre los miembros del equipo a fin de crear una atmósfera más agradable y con mayor confianza, de esta forma el grupo se compacta emocionalmente y adquiere una mayor motivación y compromiso para completar el proyecto.

2.3 Historias de usuario

Las historias de usuario son simples conversaciones acerca de los requisitos de software que el cliente necesita y son traducidas como recordatorios en notas pequeñas, el objetivo de las historias de usuario es desechar la gran cantidad de documentación formal y tiempo que lleva al equipo administraron de requerimientos detallados, quedándose con la esencia de lo que el cliente desea ver en el producto de software que se va a crear.

Entre sus características más importantes se destacan.

- Rapidez de solución cuando los requisitos son cambiantes
- No deben ser muy extensas debido a la falta de apreciación de costos que produce desarrollarla.
- Estimación del tiempo en el que la historia de usuario será completada
- Es perfecta para cuando las necesidades del cliente son volátiles.
- Divide el desarrollo del producto en varias partes que serán consideradas como entregas.
- Proporciona un entendimiento de los requisitos por parte el cliente acercándolo al desarrollo del proyecto.
- Permite conocer cuales con los requerimientos más importantes que priorizando su orden de desarrollo de acuerdo a la conveniencia del cliente.
- Tiempo estimado de entrega de historia de usuario está entre 10 horas y 2 semanas.
- Se asocia a pruebas de validación.
- Si una historia es mayor a dos semanas debe dividirse en historias de

- menor tiempo de desarrollo.
- Tienen pruebas que permiten al usuario determinar la validez de la del funcionamiento del requerimiento al momento de la entrega.
- Implementación rápida al producto final.

2.4 Lista de objetivos

Conocida como *Product Backlog* representa una percepción del desarrollo y avance del proyecto para el cliente, el mismo que plantea sus necesidades ante el *Scrum Master* y señala qué objetivos requieren una mayor prioridad para la empresa, el cliente indica el orden a cumplir de objetivos de producto terminado que el equipo debe realizar, así se puede apreciar la evolución del proyecto de forma más perceptible.

Las principales características que posee el product backlog se definen en los siguientes puntos:

- Los objetivos que se necesitan para completar el producto o proyecto.
- Costos necesarios para completar cada objetivo.
- Lista de posibles iteraciones que se realizarán durante el desarrollo del proyecto.
- Detalle de las entregas que el cliente requiere y que se realizarán durante todo el proyecto.
- Posibles riesgos que puede enfrentar el equipo durante el desarrollo del producto.
- Soluciones para mitigar los posibles riesgos que puedan afectar al desarrollo del producto.

2.4.1 Elementos de la lista de objetivos

El detalle de los elementos nos ayudará a entender que es lo que se representa en cada uno la lista de objetivos (Product Backlog).

- **Objetivo:** Es una parte del software que el equipo desarrollará en un número determinado de días y que servirá para completar un sprint de producto terminado.

- **Valor:** Es el costo que la empresa asigna y que el equipo necesita que se la creación del objetivo.
- **Estimación inicial:** Es el número de días que se demora el equipo en completar en objetivo.
- **Factor ajuste:** Es el porcentaje que indica cuanto tiempo es necesario ajustar para que el objetivo sea completado.
- **Estimación ajustada:** Es el nuevo valor de estimación que se creó a partir del valor de estimación inicial incrementado el factor de ajuste.
- **Entregas:** Es el proporcionar al cliente una parte del producto desarrollado.
- **Sprint:** Es el incremento y finalización de una porción de software desarrollado.
- **Pendiente:** Es el número de días restantes para la culminación de un sprint y el proyecto.

Tabla 1. Uso del Product Backlog listando todos los objetivos necesarios para crear un proyecto

ID	Objetivo	Valor	Estimación Inicial	Factor Ajuste	Estimación Ajustada	Sprint: 1 2 3 4 5				
						Pendiente: 225 170 114 57				
1	Objetivo A	2000	15		15	15	0			
2	Objetivo B	1750	20		20	20	0			
3	Objetivo C	1500	20		20	20	0			
	Sprint 1	5250	55		55	55	0	0	0	0
4	Objetivo D	1250	15	0,2	18	18	18	0		
5	Objetivo E	1250	20		20	20	20	0		
6	Objetivo F	1000	15	0,2	18	18	18	0		
	Sprint 2	3500	50		56	56	56	0	0	0
	Primera entrega	8750	105		111	111	56	0	0	0
10	Objetivo G	1250	20	0,2	24	24	24	24	0	
8	Objetivo H	750	15		15	15	15	15	0	
11	Objetivo I	750	15	0,2	18	18	18	18	0	
	Sprint 3	2750	50		57	57	57	57	0	0
7	Objetivo J	700	15	0,2	18	18	18	18	18	
9	Objetivo K	500	10	0,5	15	15	15	15	15	
12	Objetivo L	500	10	0,2	24	24	24	24	24	
	Sprint 4	1700	45		57	57	57	57	57	0
	Segunda entrega	4450	995		114	114	114	114	57	0

Nota. La tabla describe un formato para crear un product backlog, identificando los objetivos, las entregas que van a realizarse, detalle de costos y la estimación de tiempos para crear un proyecto. Tomado de (Proyectosagiles, s.f.).

Scrum utiliza los valores del product backlog para determinar el esfuerzo que equipo tiene que hacer en cada etapa del proyecto.

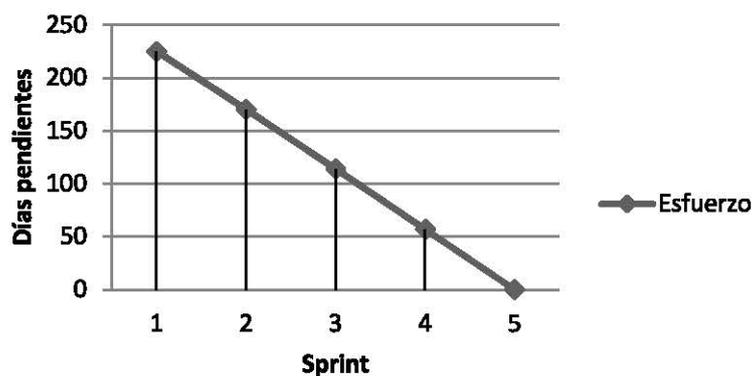


Figura 3. La gráfica muestra el esfuerzo que realiza un equipo de trabajo a lo largo de todo el proyecto en cada sprint basando los valores en la Tabla 1.

Mediante el gráfico de esfuerzo se puede detectar en cual objetivo el equipo podría tener dificultades para lograr una entrega determinada.

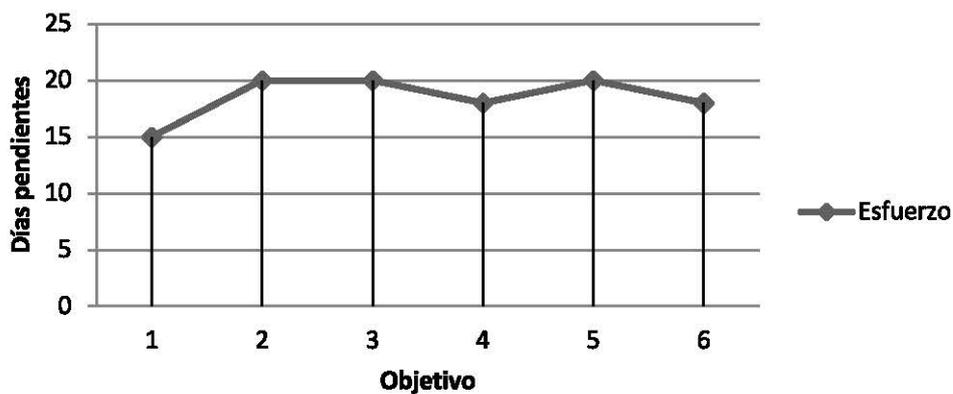


Figura 4. La gráfica muestra en qué objetivo el equipo de desarrollo requiere hacer más esfuerzo para cumplir la entrega primera entrega basando los valores en la Tabla 1.

2.4.2 Priorización de objetivos

Para realizar la priorización de objetivos el cliente debe saber qué producto va a obtener y debe detallar una lista de requisitos, al inicio del proyecto se realiza una reunión en donde se lanza una lista de objetivos a que el equipo va a realizar durante el desarrollo del software, al principio no es necesario que estén detallados todos los requisitos necesarios para el desarrollo del producto, basta con identificar qué requerimientos tienen mayor prioridad de desarrollo, se agrupan los objetivos de acuerdo al orden de prioridad establecidos formando la lista de objetivos definitiva para cada *sprint* esta a su vez se actualiza conforme el proyecto avance ganando tiempo desarrollando los objetivos más importantes que benefician al cliente.

Los beneficios de se obtiene de la priorización de objetivos son:

- Aprovechar el tiempo en empezar a crear objetivos que van a ser la base del producto.
- Ahorrar esfuerzo en detallar objetivos que no son prioritarios y que durante el transcurso del proyecto puedan ser cambiantes.
- Mayor adaptación del equipo hacia el producto en el caso que no sea necesario desarrollar algún objetivo debido a factores económicos.

2.4.3 Velocidad del equipo en objetivos completados

Es la cantidad de objetivos desarrollados por el equipo de trabajo en un espacio de tiempo, se lo puede medir en base a la cantidad de ítems desarrollados en cada *sprint* o en la cantidad de *sprints* realizados para completar el proyecto. Su objetivo es analizar el comportamiento del equipo durante el desarrollo del proyecto y encontrar en qué punto el equipo tiene dificultades de desarrollo permitiendo no caer en bucles de tiempo que retrasen la entrega del proyecto.

2.4.5 Re-priorización de los objetivos debido a factores económicos

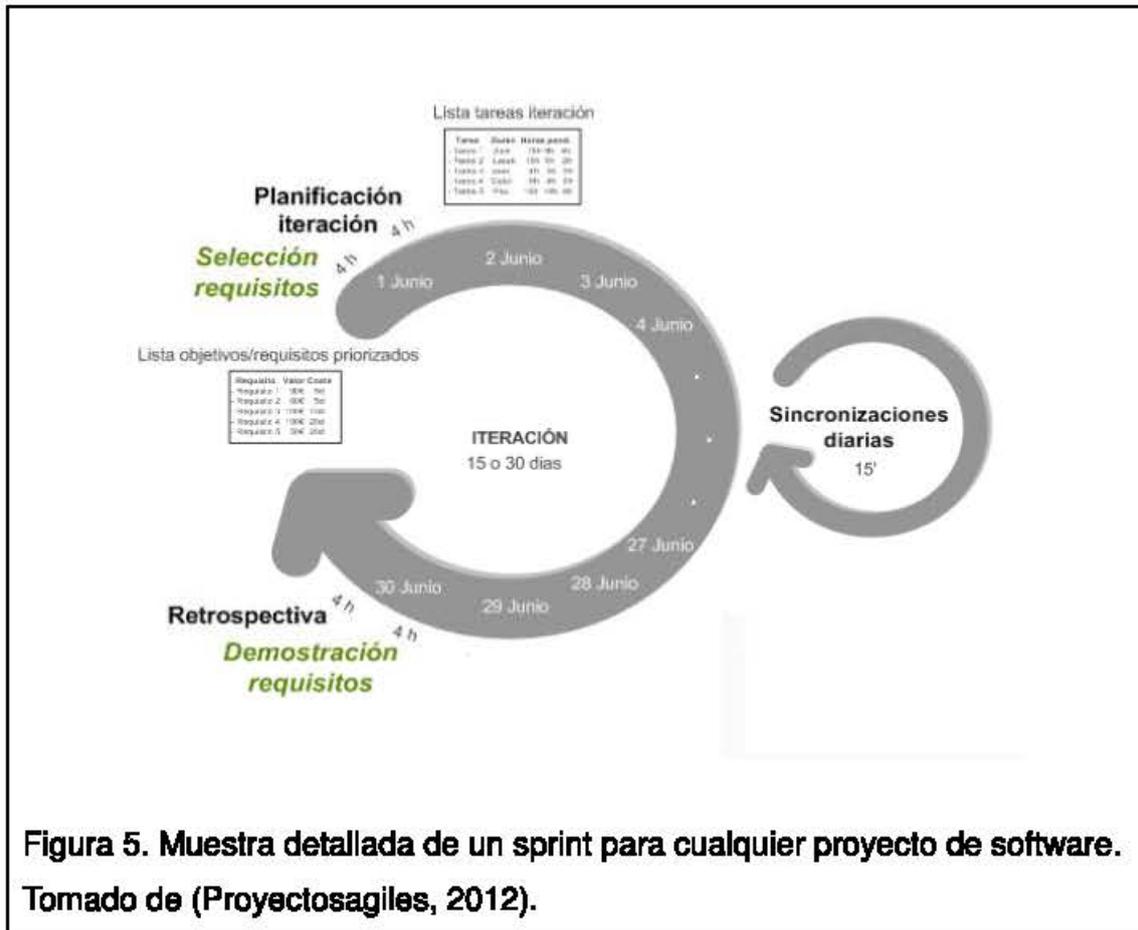
Cuando el cliente tiene la intención de realizar un proyecto de software es porque existen requerimientos que necesitan ser construidos para que la empresa siga en crecimiento, ayudando a obtener mayores ingresos económicos mejorando los resultados.

Existen ocasiones en las que hay que re-priorizar los objetivos periódicamente porque el costo que produce desarrollar el objetivo puede ser mayor que retorno de inversión (ROI) que indica “el beneficio generado después de realizar una inversión” (Proyectosagiles, s.f.), que se obtiene, el cliente en ese momento puede determinar si obtener el resto de objetivos es financieramente útil para la empresa y de ser el caso tiene toda la potestad de parar el proyecto y quedarse con lo desarrollado, así tendría beneficios en lugar de pérdidas.

Volver a priorizar los objetivos empieza cuando el cliente en ocasiones no puede apreciar cuales son los requisitos fundamentales que le ayudarán a tener beneficios, la flexibilidad de Scrum se da con evaluaciones periódicas de los objetivos, el cliente puede enmendar esa falta de visión para determinar un nuevo orden de prioridad de los entregables del producto.

2.5 Sprint

Cada *sprint* “iteración” representa una porción del producto de software a construir en un período de tiempo. Durante este tiempo el equipo realiza varias actividades que son necesarias para cumplir las metas de desarrollo de una solución de software.



Entre las principales características del sprint podemos observar las siguientes:

- Cada sprint tiene una duración aproximada de entre 2 a 4 semanas.
- Cada sprint es un pequeña parte terminada del producto.
- El equipo realiza tareas y actividades que son necesarias para cumplir los objetivos.
- Al final del sprint el cliente contará con una parte del software trabajando y probado.
- El sprint no puede detenerse.

Un sprint solo puede detenerse si:

- Se ha sobrepasado el coste de desarrollo del sprint.
- El equipo decide que los objetivos son imposibles de alcanzar.

- El cliente decide que el sprint en curso ya no es prioridad para el proyecto.
- El producto necesita realizar cambios representativos que no puedan esperar hasta el final del sprint.

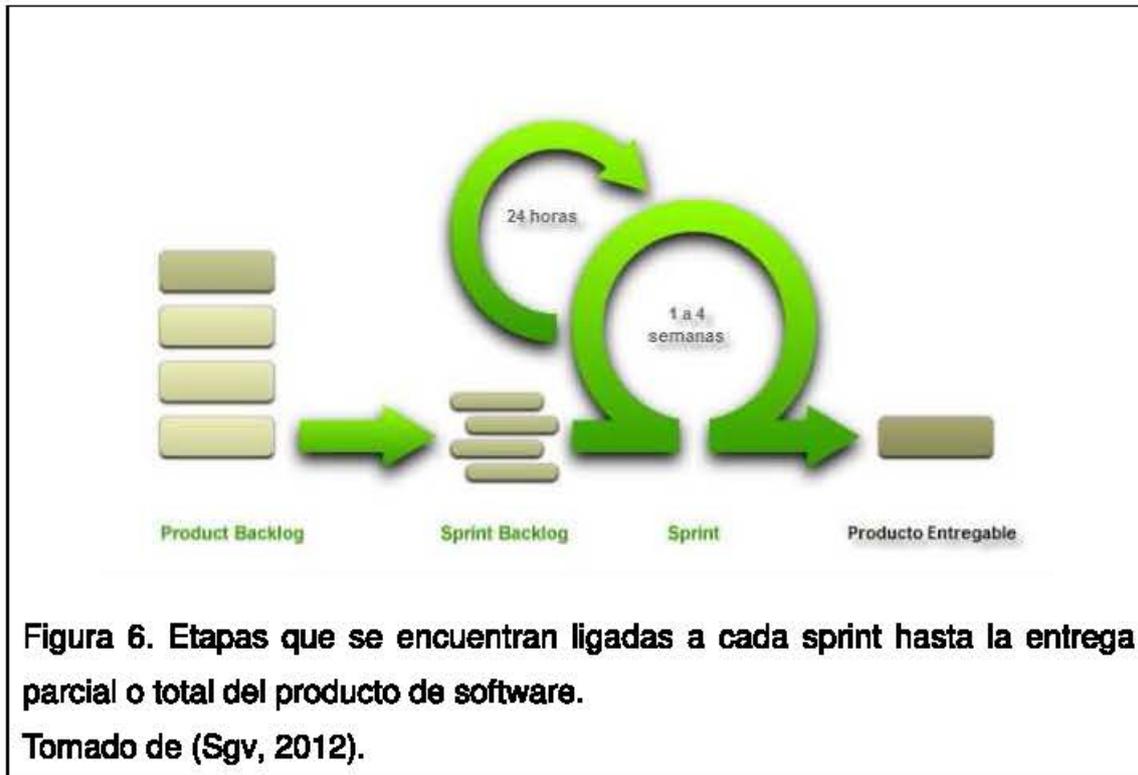


Figura 6. Etapas que se encuentran ligadas a cada sprint hasta la entrega parcial o total del producto de software.

Tomado de (Sgv, 2012).

2.5.1 Planificación de un *sprint*

Antes del inicio de cada *sprint* el equipo de desarrollo tiene una reunión en la que se discuten temas relacionados con los objetivos propuestos para el *sprint*. Al finalizar la planificación el equipo ha creado el *sprint backlog* "Lista de tareas" que representan las actividades a seguir durante el *sprint*.

Características de la reunión de planificación:

- Duración hasta de 8 horas.
- Se determina el objetivo a construir durante el sprint.
- La primera parte de la reunión se involucra al cliente para aclarar dudas al equipo y seleccionar el objetivo a construir.
- La segunda parte de la reunión el equipo realiza la lista de tareas que se van a realizar durante el sprint.

- En algunos casos se invita a los patrocinadores del proyecto.
- El equipo decide cuantas tareas son suficientes y apropiadas para realizar completar un sprint, no tienen que ser demasiados.
- El scrum master ayuda a decidir al equipo si las metas son realistas basado en su propia experiencia de anteriores iteraciones con respecto a la velocidad de desarrollo.

Una vez que el equipo ha completado las tareas que van a ser necesarias para realizar el *sprint* hace la entrega de los resultados de la planificación y el equipo empieza a trabajar.

2.5.1.1 *Sprint backlog*

Es la lista de tareas que se necesitan completar para cada *sprint*, como resultado de la reunión de planificación el equipo ha desglosado cada objetivo de un *sprint* en múltiples tareas las cuales deberán ser necesariamente finalizadas para la entrega del incremento del producto.

2.5.1.2 Elementos del *Sprint Backlog*

Los elementos para realizar una Sprint Backlog son:

- **Objetivo:** Representa la entrega que se realizará al cliente.
- **Tarea:** Es un trabajo que tiene que ser completado, un objetivo se construye en base a muchas tareas.
- **Responsable:** Es la persona que está a cargo de cumplir con la tarea asignada.
- **Estado:** Indica si la tarea se encuentra en estado Completado, en progreso o no iniciado.
- **Día:** Es un listado del número de días asignados a completar el proyecto.
- **Horas pendientes:** Es el número de horas que están asignadas a cada día de trabajo.

Tabla 2. Cuadro referencial de sprint backlog

Requisito	Tarea	Quien	Estado (No iniciada / en progreso /completada)	Día: 1 2 3 4 5 6 7 8 9 10																
				Horas Pendientes:																
				1120	1088	1076	1048	1040	1032	1020	1008	992	972							
Objetivo A	Tarea 1	Camilo	Completada	16	8															
Objetivo A	Tarea 2	Santiago	Completada	4																
Objetivo A	Tarea 3	Jorge	Completada	4																
Objetivo A	Tarea 4	Camilo	Completada	8																
Objetivo A	Tarea 5	Santiago	Completada	16	8	4														
Objetivo A	Tarea 6	Jorge	Completada	8	8	8														
Objetivo A	Tarea 7	Camilo	Completada	16	16	16	8													
Objetivo A	Tarea 8	Santiago	Completada	8	8	8														
Objetivo A	Tarea 9	Jorge	Completada	8	8	8	8	8												
Objetivo A	Tarea 10	Camilo	Completada	8	8	8	8	8	8	4										
Objetivo A	Tarea 11	Camilo	Completada	16	16	16	16	16	16	8										
Objetivo B	Tarea 12	Camilo	Completada	16	16	16	16	16	16	16	16	16	16	8						
Objetivo B	Tarea 13	Santiago	Completada	16	16	16	16	16	16	16	16	16	16	8						
Objetivo B	Tarea 14	Jorge	En progreso	8	8	8	8	8	8	8	8	8	8	8	4					
Objetivo B	Tarea 15	Camilo	En progreso	8	8	8	8	8	8	8	8	8	8	8	8					
Objetivo B	Tarea 16	Santiago	En progreso	8	8	8	8	8	8	8	8	8	8	8	8					
Objetivo C	Tarea 17	Jorge	No iniciada	4	4	4	4	4	4	4	4	4	4	4	4					
Objetivo C	Tarea 18	Camilo	No iniciada	8	8	8	8	8	8	8	8	8	8	8	8					
Objetivo C	Tarea 19	Santiago	No iniciada	16	16	16	16	16	16	16	16	16	16	16	16					
Objetivo C	Tarea 20	Camilo	No iniciada	8	8	8	8	8	8	8	8	8	8	8	8					

Nota: Cada objetivo del product backlog se divide en tareas y a su vez cada tarea es asignada a un miembro del equipo que tiene un plazo en días para ser completada. Tomado de (Proyectosagiles, 2012).

2.5.1.3 Elaboración del *sprint backlog*

La elaboración del *Product Backlog* se basa en los objetivos, en primer lugar se detectan cuáles son los objetivos más importantes y con mayor trascendencia para el proyecto, el orden de los objetivos está dispuesto en relación a la prioridad que requiere el cliente. En algunos casos se puede ver los puntos débiles de un equipo de trabajo al observar el progreso de la lista de tareas del proyecto, por ejemplo, una lista de tareas de forma desequilibrada significa que en algunas ocasiones el equipo completa objetivos que no son de prioridad para el cliente, o cuando el proyecto está compuesto por un alto número de objetivos desarrollándose al mismo tiempo.

Las tareas tienen siempre un orden de dependencia, es decir, que si una tarea se debe realizar en base a otra, esta deberá siempre ir por debajo de la tarea a completarse.

La identificación de la tarea se la realiza de forma tan reducida como para poder completarla en un período de tiempo de entre 4 y 16 horas, de esta forma se puede analizar el progreso diario, además de medir la velocidad a la que el equipo se encuentra desarrollando.

2.5.1.4 *Burndown chart*

Es un cuadro en el que se puede apreciar los días que faltan para completar una iteración o el proyecto total, por medio de este cuadro se puede medir la velocidad de desarrollo del equipo así como también apreciar como el equipo responde cuando se realizan cambios en los objetivos o aplazamiento de fechas de entrega.

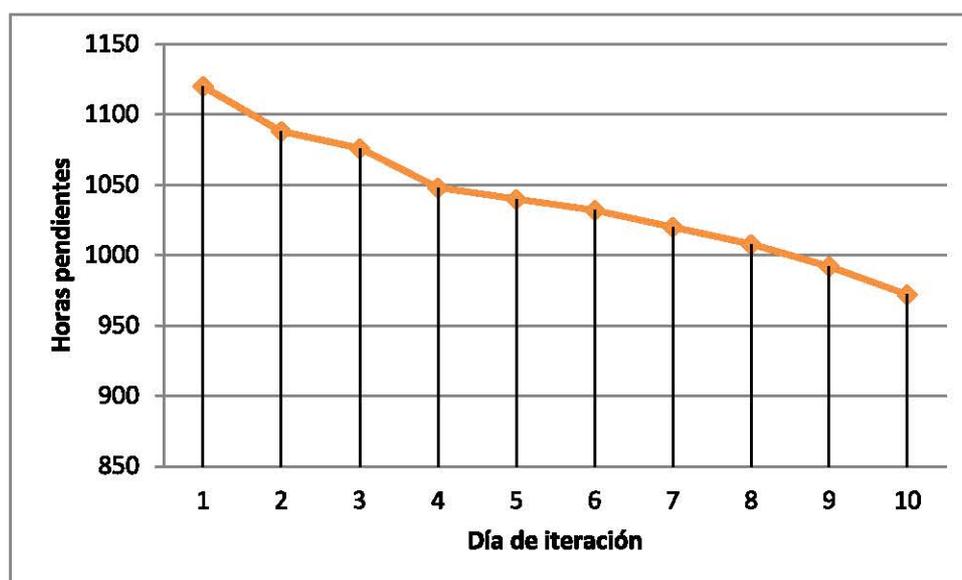


Figura 7. Burndown chart con relación a las horas de trabajo pendientes para completar el sprint con respecto a los valores de la Tabla 2.

2.5.2 Reunión de sincronización

Todos los días el equipo de desarrollo tiene reuniones de 15 minutos que son conocidas como sincronizaciones diarias o *timebox*, en las que el equipo debe transmitir información del *sprint* al resto de miembros del equipo de desarrollo con el fin de exponer los puntos de vista para que el equipo pueda alcanzar la meta y lograr el *sprint*.

Todos los miembros deben estar atentos a las posibles eventualidades que puedan surgir durante el desarrollo del *sprint*, proponiendo ideas de apoyo y ayuda al resto del equipo a fin de superar obstáculos y cumplir los objetivos trazados para el *sprint*. En las reuniones de sincronización el equipo debate en 3 aspectos clave que sirven de eje para la adaptación de todas las partes ¿Qué he hecho?, ¿Qué voy a hacer? y ¿Qué impedimentos tengo?.

2.5.2.1 ¿Qué he hecho?

Se analiza el avance que el equipo ha tenido desde la última reunión de sincronización, permitiendo visualizar si se ha hecho todo lo planificado y dando una mirada hacia los problemas que imposibiliten cumplir con los objetivos trazados para el día.

2.5.2.2 ¿Qué voy a hacer?

En ese momento el equipo determina cuales son las metas a cumplir hasta la siguiente reunión de sincronización.

2.5.2.3 ¿Qué impedimentos tengo?

Se analiza brevemente cuales serían los posibles problemas que el equipo de desarrollo tenga que enfrentar, de éste modo si se presentan los obstáculos el equipo pueda recurrir a una estrategia que permita superarlos y reducir su impacto en el proyecto.

Es recomendable que las reuniones se las realice en forma de pie para evitar que la reunión se alargue mucho y que los miembros empiecen a relajarse y así comenzar a hablar de otros temas que no son beneficio para la sincronización.

2.5.3 Beneficio de la reunión de sincronización

El beneficio más importante que se obtiene al realizar reuniones de sincronización periódicas es la productividad, debido a que cada miembro del equipo debe exponer su autoanálisis del progreso del proyecto, para luego lanzar ideas que colaboren a afianzar el compromiso del grupo hacia el proyecto.

También se puede listar los siguientes beneficios:

- Conocer en qué punto se encuentra el avance de la iteración.
- Identificar de forma oportuna los inconvenientes que puedan presentarse durante el desarrollo del producto.
- Todos los miembros del equipo saben cuáles son las falencias de sus compañeros y pueden aportar con ideas que ayuden a superarlas.
- Ayuda a identificar las prioridades para no perder tiempo en tareas que pueden retrasar al desarrollo del producto.
- Conocer el rendimiento de cada integrante del equipo.
- Evita el cuestionamiento entre miembros de grupo ya que todos conocen las tareas en las que cada uno se encuentra trabajando.
- Aprendizaje continuo de los miembros del equipo por asimilación de las propuestas para dar solución a problemas por parte de miembros de equipo más experimentados o especializados.
- Hacer notorias las tareas pendientes.

2.5.4 Restricciones de la reunión de sincronización

Entre las restricciones más importantes tenemos:

- Falta de atención a los detalles expuestos ya que algunos miembros pueden no considerarlo de su interés.
- Pérdida de tiempo tratando de planificar reuniones de colaboración de conocimientos para otros miembros del equipo.
- En las reuniones no resuelven los inconvenientes de desarrollo, solo lanzan ideas para que en el transcurso del día puedan ser resueltos.

- Pueden formarse pequeños grupos de discusión y perder la atención de todo el grupo hacia el tema a tratar.
- En algunas ocasiones el tiempo de la reunión puede emplearse para divergir entre compañeros.

2.5.5 Retrospectiva del *Sprint*

Sirve para que el equipo de desarrollo mejore su rendimiento y facilite la mejora en cuanto a la creación del producto para las siguientes iteraciones, esta retrospectiva se la realiza antes de empezar la planificación de un nuevo sprint, es una reunión que dura aproximadamente unas 3 horas durante las cuales el equipo regresa para ver en qué parte del desarrollo ha tenido dificultades y como las ha podido superar a fin de seguir desarrollando el producto cada vez más robusto.

2.6 Scrum vs metodologías tradicionales

Scrum dentro de las metodologías ágiles surge como alternativa ante las metodologías tradicionales debido a la experiencia y necesidades que han tenido que enfrentar los diferentes equipos de desarrollo durante la evolución del desarrollo de software a lo largo del tiempo especialmente cuando se observa que los proyectos generalmente tienden a ser cambiantes de su objetivo inicial.

Existen varios factores que influyen a los desarrolladores en optar por Scrum a la hora de optar por una metodología de desarrollo, como principales diferencias tenemos:

Tabla 3. Comparación entre Scrum y las Metodologías Tradicionales.

	Metodologías Ágiles	Metodologías Tradicionales
Equipo de desarrollo	Miembros multifuncionales	Miembros especializados
	Retroalimentación de conocimiento por todos los miembros del equipo	Conocimientos especializados en subequipos
	Tiene una comunicación constante con el Scrum master, todos los miembros del equipo y el cliente	Los diferentes subequipos se comunican de acuerdo al rol que representen
	Enfoca en el resultado final	Enfocado a la entrega de la siguiente fase
	Enfocado en interacción con el equipo	Enfocado en procesos
	El cliente es parte del equipo	El cliente actúa mediante negociaciones
Artefactos	Documentación necesaria	Excesiva documentación
	La documentación sirve a lo largo del proyecto	La documentación son marcadas y divide cada fase de desarrollo
	Gana tiempo de desarrollo a la documentación excesiva	Pérdida de tiempo en detalle de documentación
Producto	Incremento gradual del producto de software final probado y operativo	No se realizan pruebas sino hasta que deba entrar a la fase de pruebas de un producto que tiene que estar parcialmente terminado
	Entrega de resultados probados en menor tiempo	Mayor tiempo para entregas
	Lanzamiento de primeras versiones de acuerdo a la prioridad del cliente	Se lanza una única versión después de las pruebas
Cambios	Adaptable a cambios durante el desarrollo	El desarrollo tiene que ser concebido de acuerdo a un plan inicial
	Tiene retrospectiva al final de cada sprint	No puede regresar a ver porque tiene establecido un plan
	Menor impacto	Mayor impacto
Riesgos	Reducción de riesgos debido al incremento de producto final probado	Alto riesgo debido a que el software no es probado sino hasta antes de su implementación
Costos	Costos estimados con respecto a los cambios que pueden darse durante el proyecto	Si el software necesita un cambio va a incrementar en su costo de desarrollo
	Si las necesidades del cliente lo requieren debido a excesos financieros no estipulados se puede hacer una para de desarrollo de objetivos no prioritarios sin dejar de entrega lo esencial para el cliente	Tiene un costo fijo establecido que se incrementará de acuerdo al retraso y los cambios
Esfuerzo	Es medible durante cualquier día o etapa del proyecto	Solamente se puede medir de acuerdo a como el equipo vaya avanzando con su plan
	Se mide a través del tiempo	Se mide a través del producto

El cliente puede visualizar más rápidamente lanzamientos de producto terminado y funcionando, en lugar de esperar al final del ciclo de desarrollo para empezar tener resultados.

Tabla 4. Costos, ROI y Beneficios en un proyecto de software utilizando metodologías tradicionales y ágiles.

	Costo Total	Beneficios	B/CR	ROI%
ISO 9001	\$ 173.000	\$ 569.841	3:1	229
CMMI	\$ 1.108.233	\$ 3.023.064	3:1	173
SW CMM	\$ 311.433	\$ 3.023.064	10:1	871
Agile	\$ 217.712	\$ 4.292.285	20:1	1872

Nota: Representación de 79 estudios para ROI equivalente a \$26 por cada \$1 invertido en un proyecto de software. Tomado de (RICO, 2009).

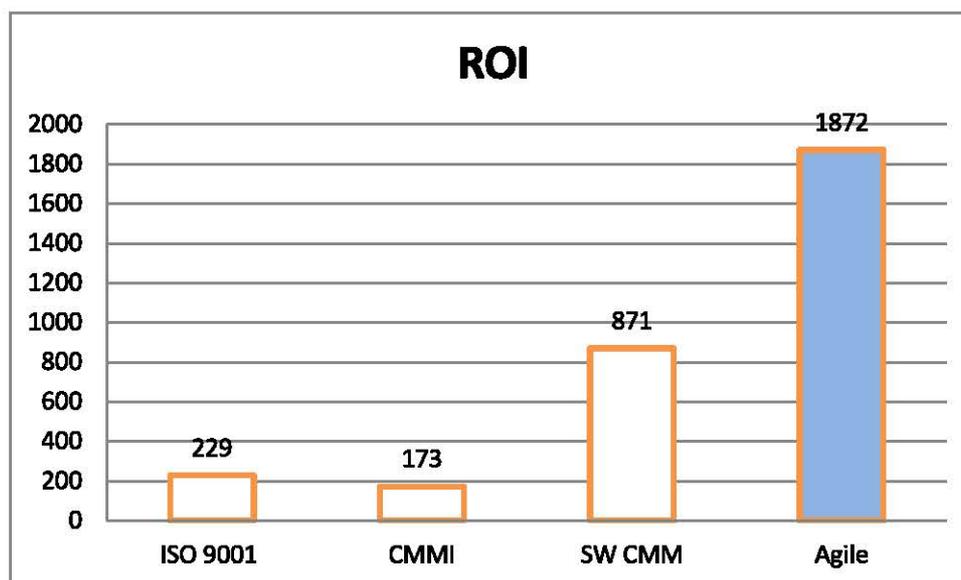


Figura 8. Retorno de inversión de un proyecto de software usando metodología ágil de acuerdo a la Tabla 4.

Los beneficios económicos se miden en función del retorno de inversión obtenido.

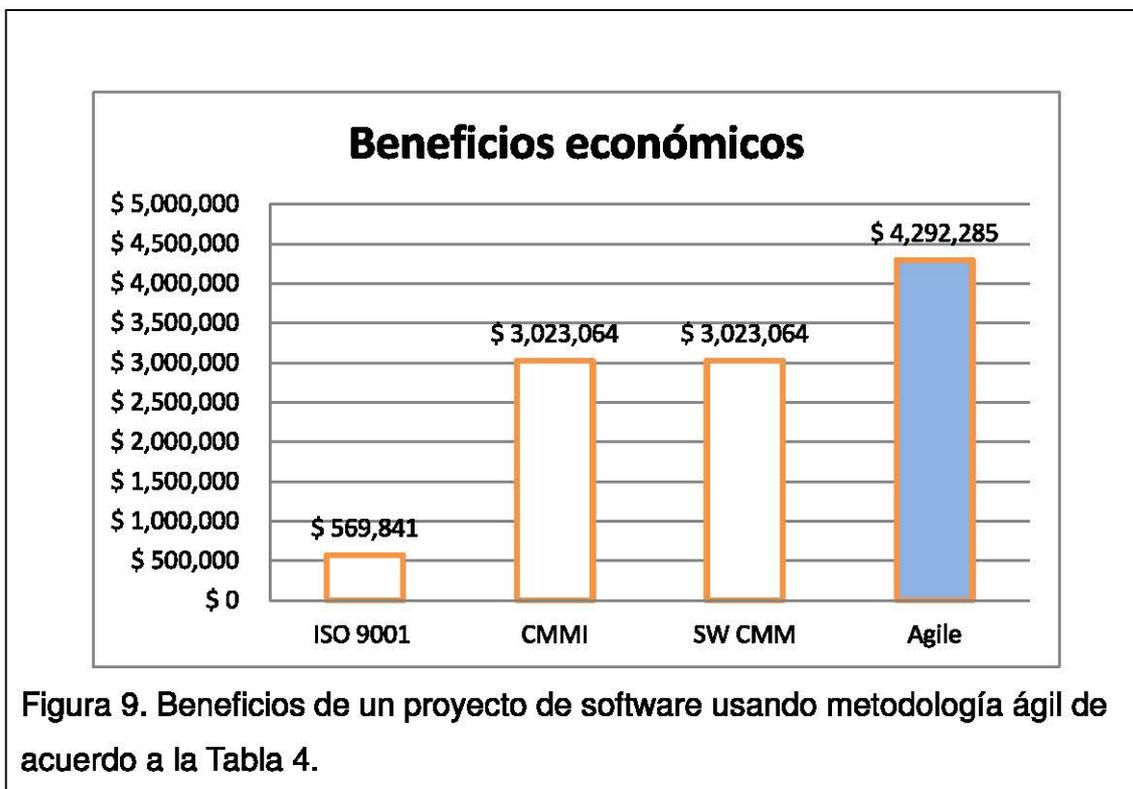
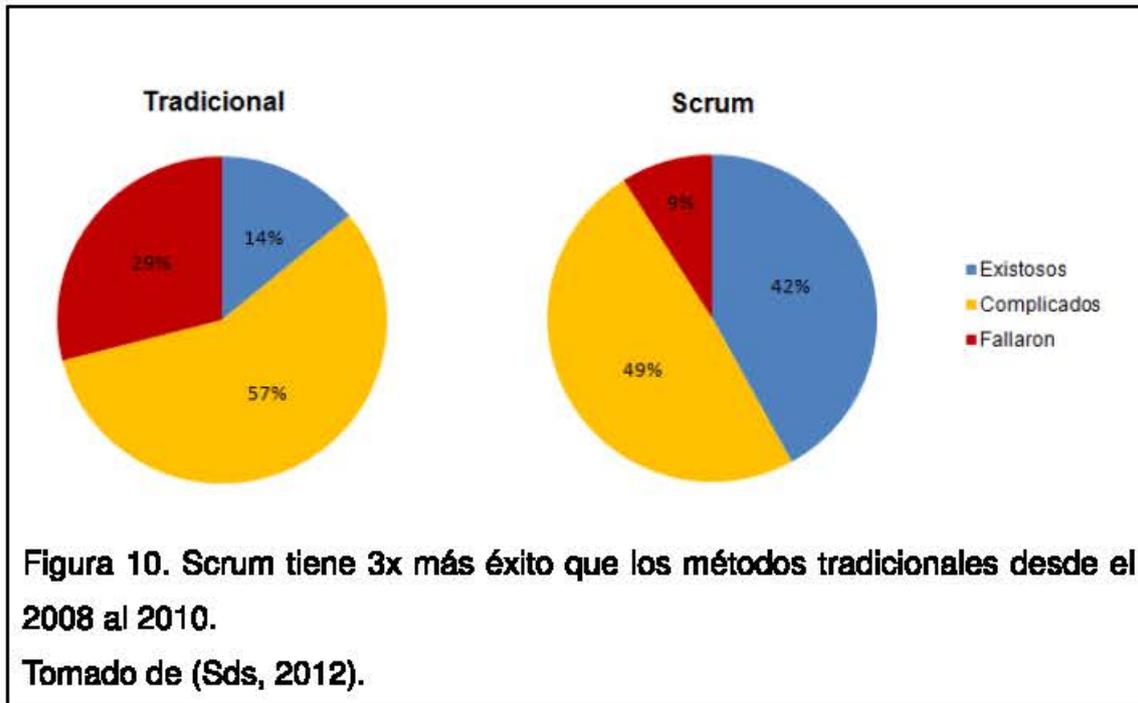


Figura 9. Beneficios de un proyecto de software usando metodología ágil de acuerdo a la Tabla 4.

Se ha comprobado que la metodología Scrum tiene mayor porcentaje de éxito con respecto a las metodologías de desarrollo tradicionales, debido a su adaptación al núcleo del negocio.



2.7 Beneficios de utilizar Scrum

Existen muchos beneficios al utilizar Scrum, de entre los cuales podemos citar los siguientes:

- Mitigación de riesgos rápida y oportuna.
- Rápidas entregas parciales del producto.
- Mejor seguimiento de la evolución del proyecto a ser desarrollado de desarrollo.
- Apreciación clara al trabajo y esfuerzo que realiza el equipo de desarrollo.
- Mejor calidad de desarrollo del producto.
- Más oportunidad de satisfacer al cliente en caso de requerir cambios.
- Mejor ambiente de trabajo entre miembros del equipo.
- Mayor oportunidad de aprendizaje entre los integrantes del equipo.

2.8 Mejores prácticas de Scrum

Para que Scrum sea lo más efectivo posible es recomendable utilizar lo mejor de la metodología para así obtener los resultados esperados, como mejores prácticas de Scrum pueden tomar en cuenta:

- Identificar las historias de usuario como la base de todo el proyecto, no empezar con la priorización sin tener una clara visión acerca de las historias de usuario que quiere comunicar el cliente.
- El equipo debe regirse lo más estricto posible a las reglas empleadas por Scrum para la reunión de sincronización diaria, utilizando siempre los 15 minutos propuestos y no salir del contexto sobre ¿Qué he hecho?, ¿Qué voy a hacer? Y ¿Qué impedimentos tengo?, el equipo debe ser muy objetivo en cuanto a los temas a tratar.
- Siempre tener bien definido y claramente detallados el Product Backlog y Sprint Backlog, si existe algún cambio inmediatamente actualizar los artefactos de tal forma que nunca quede una tarea pendiente o sin fundamento para el desarrollo.
- No se debe marcar como completada una tarea que no haya sido previamente probada.
- El Scrum master debe siempre tratar de elegir un equipo de trabajo ideal en el que los miembros se sientan cómodos entre sí, que tengan facilidad de comunicación, que trabajen organizadamente y tengan un compromiso con el proyecto para un supuesto caso de haber eventualidades.
- Siempre debe existir una buena comunicación entre todo el equipo incluso hacia el cliente, es la base para superar los obstáculos que pueden surgir durante el proyecto y entender cualquier necesidad del cliente.
- Siempre terminar todas las tareas propuestas para cada Scrum.
- Asignar un valor representativo único a cada objetivo para que si es el caso de no desarrollarlo o re-priorizarlo el proyecto siga fácilmente administrado.
- Siempre tener una amplia comunicación con el cliente de esta forma el cliente puede hacer entender al equipo necesidades que no se divisaban inicialmente.

- Como herramienta para las tareas con mayor grado de prioridad se puede utilizar el ROI.
- Utilizar constantemente los artefactos Product Backlog, Sprint Backlog, Burndown Chart, y los diagramas de esfuerzo, de esta manera podemos apreciar como evaluación del avance del proyecto, como el equipo se encuentra trabajando y para donde van a futuro.

2.9 Porque utilizar Scrum para el presente proyecto

Se ha optado por Scrum para el presente proyecto debido a que los requisitos que la empresa MF Dental requiere pueden ser cambiantes durante el desarrollo del proyecto, debido a que al no existir un producto de software similar el cliente necesite agregar cambios que lo beneficien a fin de brindar una mejor organización de la información registrada de cada paciente en el sistema.

El objetivo principal se mantiene claro en cuanto a cómo el sistema va a generar los gráficos de apoyo, pero en MF Dental los periodoncistas al manejar gráficos dibujados a mano y toda la documentación de los pacientes organizada de forma de archivos de carpetas es muy posible que surjan requerimientos adicionales que pueden ser aprovechados para un mejor rendimiento de la empresa generando mejor rentabilidad e ingresos.

2.10 Empresas que han usado Scrum

Existen empresas que actualmente se encuentran liderando los sectores a los cuales pertenecen, estas grandes empresas son la demostración de la confianza que han puesto en Scrum como metodología de desarrollo para sus soluciones de software.

Tabla 5. Empresas de que utilizan la metodología Scrum para desarrollar su software.

Sector	Empresas
Media	BBC, BellSouth, British Telecom, DoubleYou, Motorola, Nokia,
Software, Hardware	Adobe, Autentia, Biko2, Central Desktop, Citrix, Gailén,
Internet	Amazon, Google, mySpace, Yahoo
ERP	SAP
Banca e Inversión	Bank of America, Barclays Global Investors, Key Bank, Merrill Lynch
Defensa y Aeroespacial	Boeing, General Dynamics, Lockheed
Juegos	Blizzard, High Moon Studios, Crytek, Ubisoft, Electronic Arts
Otros	3M, Bose, GE, UOC, Ferrari

Nota: ERP (Enterprise Resource Planning) son sistemas que comprenden negocios, producción y distribución en una empresa. Tomado de (Proyectosagiles, s.f.).

CAPITULO 3

TECNOLOGÍA

3.1 El lenguaje PHP

PHP significa Hypertext Pre-Processor, es un lenguaje de alto nivel orientado a objetos, el lenguaje fue concebido en sus inicios en Perl que es un lenguaje basado en C que es “un lenguaje de programación creado en el año 1972 enfocado inicialmente hacia soluciones para sistemas operativos” (Wikipedia, s.f.), básicamente su programación era desarrollada en bloques y procesamiento de texto, luego evolucionó en 1994 a PHP/FI utilizando CGI “Common Gateway Interface que eran los estilos básicos de programación web” (W3, s.f.), continuó su desarrollo hasta transformarse en PHP Tools que básicamente fue el pilar para el desarrollo de páginas y aplicaciones web de tal forma que puedan tener una comunicación simple con un motor de base de datos, debido a que PHP Tools compartió su código con la comunidad de desarrolladores, estos se dedicaron conjuntamente a solventar los errores y ayudaron a que el lenguaje sea cada vez más robusto.

El código fue re-escrito, PHP/FI siguió creciendo y llamado PHP 2.0 en ese punto ya tenía una comunicación con motores de bases de datos y funciones adicionales que los desarrolladores podían crear. Todavía existía mucha limitación en cuanto a su evolución ya que Rasmus Lerdorf “Creador del lenguaje PHP” seguía haciéndolo casi solo, a pesar de que ya existía una comunidad de desarrolladores que aportaban con soluciones

En 1998 se re-escribió nuevamente el código y se publicó PHP 3.0 el cual ya era orientada a objetos contaba con una interfaz más compleja, varias extensiones y principalmente trabajaba con varias bases de datos, en ese punto ya contaba con 70000 dominios en el mundo que trabajaban con PHP 3.0. PHP 4.0 fue lanzado en el año 2000, en esta versión ya se contaba con características muy robustas y firmes en cuanto a servidores web, seguridad y comunicación con la base de datos.

PHP ha crecido tanto que en el año 2004 fue publicada PHP 5.0 que es la versión en la que los desarrolladores trabajan actualmente, depurando y mejorando notablemente la seguridad, fortaleciendo su estructura y simplificando la forma en la que se crean las soluciones web, siendo uno de los primeros lenguajes en trabajar directamente en el servidor.

3.1.1 Características de PHP

Muchas son las características que ofrece PHP entre algunas podemos citar:

- PHP es orientado a objetos.
- Se la utiliza para creación de soluciones web dinámicas.
- Tiene una curva de aprendizaje baja con respecto a otros lenguajes de programación.
- Puede utilizar una base de datos para trabajar la información.
- El código se ejecuta desde el servidor.
- Trabaja con páginas en utilizando el formato HTTP “HyperText Transfer Protocol es un protocolo para intercambio de datos en la web” (Wikipedia, s.f.).
- Tiene alta seguridad en sus soluciones debido a que solo muestra los resultados trabajados en el servidor mediante un navegador.
- PHP tiene una amplia gama de expansión debido a que tiene muchas funciones que se pueden instalar mediante bibliotecas o extensiones.
- Es software de distribución libre.
- Debido a su popularidad PHP tiene una gran cantidad de documentación, instructivos, libros videos y ejemplos para una mejor experiencia de desarrollo.
- Salida como página HTML “HyperText Markup Language es un traductor a forma de texto de la información que pasa por la red” (Wikipedia, s.f.).
- Pueden ser adaptables a un patrón de diseño de software.
- PHP puede tener una conexión con casi todas las bases de datos.

3.1.2 Base para el uso de PHP

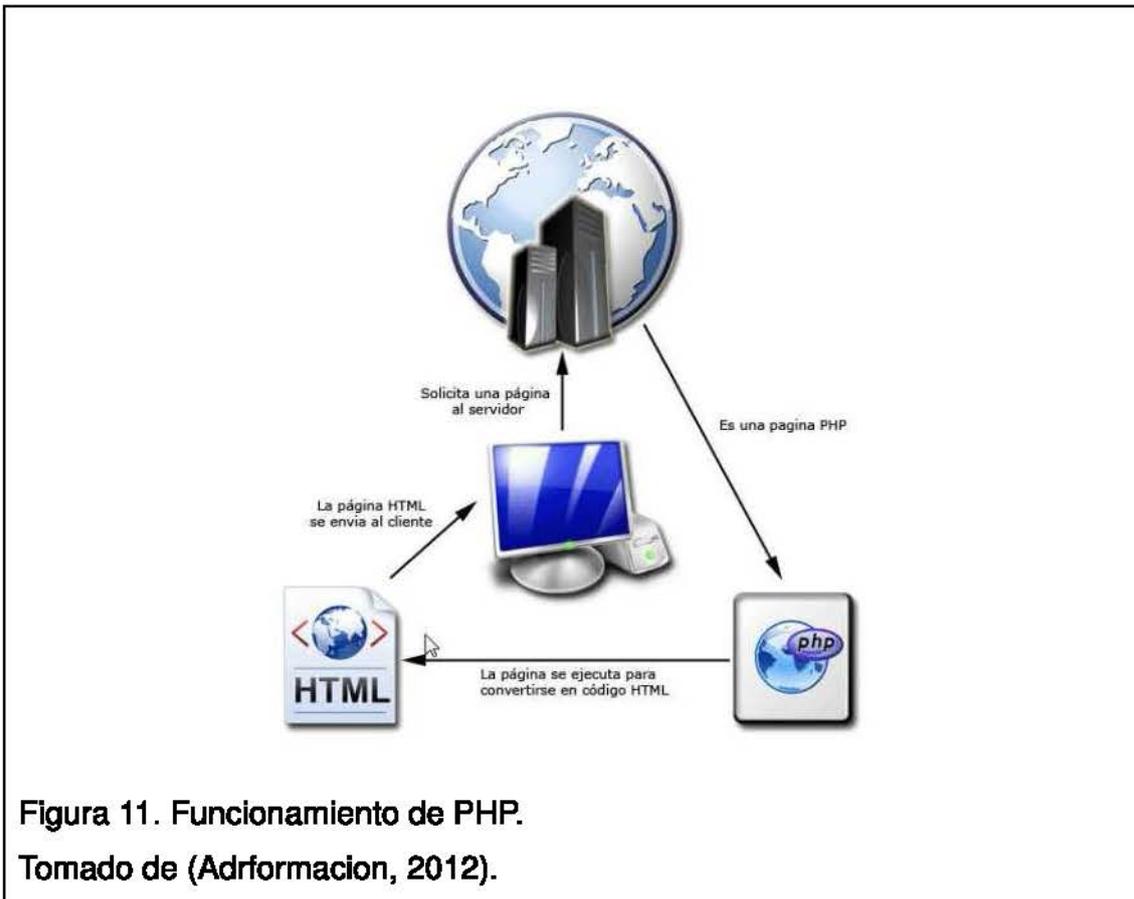
PHP tiene principios en los cuales basar su desarrollo y pueden interpretarse de acuerdo al criterio de cada desarrollador, sus principios se muestran como:

- Se empleará su utilización exclusivamente cuando sea necesario.
- Creación de múltiples tablas a fin que sea cada vez más escalable.
- Desconfianza en usuarios.
- En lo posible utilizar IDE "Integrated Development Environment es un programa que sirve para desarrollar en un lenguaje de programación determinado, provisto de herramientas de compatibilidad" (Wikipedia, s.f.).
- Mejorar la utilización de las funciones.
- Utilizar un framework basado en PHP.
- Mantener habilitado el indicador de errores.

3.1.3 ¿Cómo trabaja PHP?

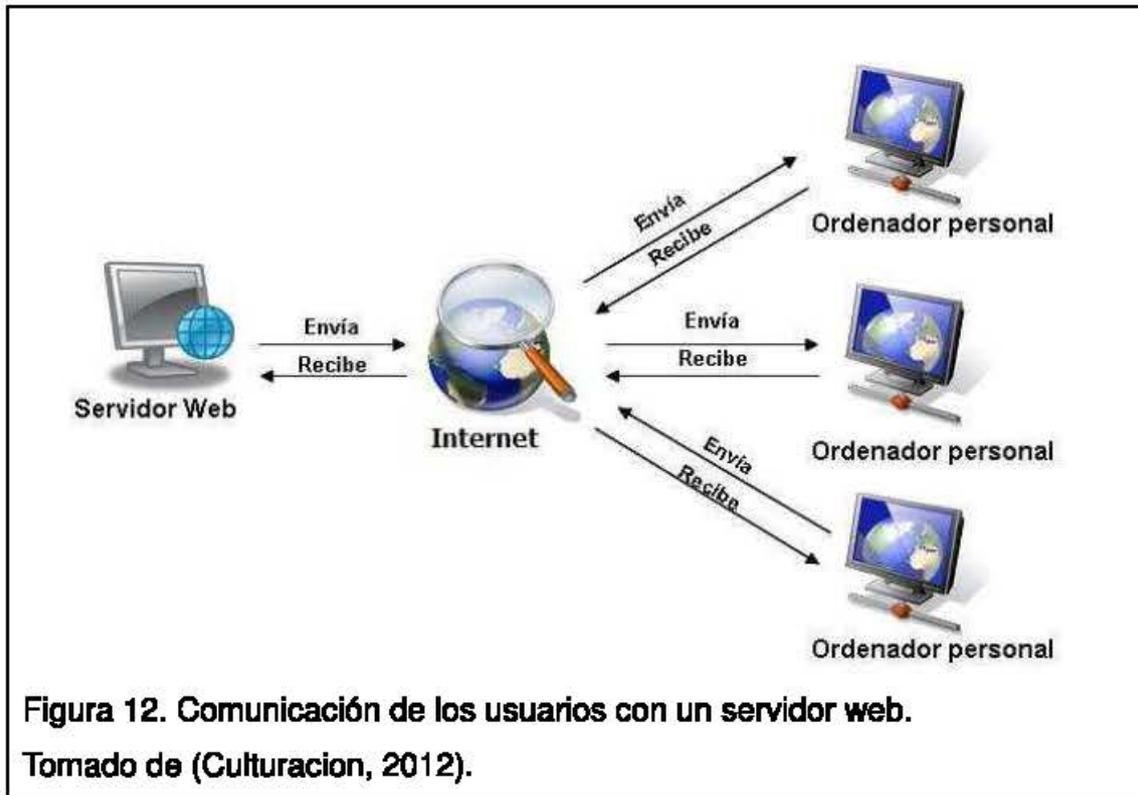
Uno de los fuertes de PHP es que es un lenguaje multiplataforma, tiene la capacidad de adaptarse tanto a ambientes Unix como Windows, por lo tanto existe una sola manera de implementar PHP sin requerir de una necesidad de alteraciones en la codificación.

PHP también posee la ventaja de trabajar directamente en el servidor web, en el caso del sistema operativo Unix debe usar algún servidor web como Apache y para el caso del sistema operativo Windows puede usar IIS que se refiere a "Internet Information Service que es un servidor web desarrollado por Windows" (Wikipedia, s.f.).



3.1.3.1 El servidor Web

Un servidor web es una aplicación de ejecución constante que se encuentra alojada en un disco de un computador o en un espacio de disco alquilado y es accedido a través de internet, el servidor web se mantiene a la espera para responder a las solicitudes que los usuarios realizan en una aplicación que se muestra mediante un navegador web, el formato de comunicación entre el cliente y el servidor está establecido en protocolo HTTP que enlaza el sitio alojado en un espacio de servidor y la interfaz del cliente para mostrarla mediante un navegador web.



3.1.3.2 Características de un servidor Web

Como principales características de servidores web tenemos:

- Alta velocidad de procesamiento.
- Trabaja entregando las respuestas a solicitudes utilizando el protocolo HTTP.
- Se mantiene en ejecución a la espera solicitudes.
- En la actualidad los servidores web se implementan en la misma compañía supliendo el alojamiento en entidades externas.
- Responde a las peticiones por la dirección IP.
- Se pueden aplicar en todo tipo de empresas.
- Se puede trabajar con todo tipo de aplicaciones web desde las más simples hasta las que tienen un alto grado de complejidad.
- Alta versatilidad en cuanto al tipo de aplicación ya que se los puede utilizar para sitios enfocados en comercio electrónico o sitios que simplemente necesiten mostrar datos estadísticos.
- Soporte la comunica con un motor de base de datos.

- Puede trabajar con otro tipo de protocolo que no sea en formato HTTP como SSL o HTTPS.
- Entrega páginas web que se muestran en un navegador web.
- Generación dinámica de páginas web, es decir, la página se forma a partir de los datos obtenidos desde una base de datos, la información no se encuentra estática grabada en la página.

3.1.3.3 Servidores web disponibles en la actualidad

Los servidores Web más utilizados actualmente son:

- Apache Server.
- Internet Information Services (IIS).
- Nginx.
- Cherokee.
- Google Web Server (GWS).
- Light HTTP Server.
- Sun Java System Web Server.
- IBM HTTP Server.
- Oracle HTTP Server.

3.1.4 Pasos para el funcionamiento de PHP

- PHP tiene un funcionamiento que se resume en los siguientes pasos:
- El usuario a través de la interfaz realiza una petición de una página al servidor web.
- En el servidor web ejecuta código PHP solicitado por el usuario.
- PHP procesa la respuesta y la convierte en código HTML.
- El navegador recibe la respuesta procesada en PHP como una página en formato HTML.
- El usuario observa los resultados como una página web.

3.1.5 Requisitos para trabajar con PHP

Para el correcto funcionamiento de PHP es necesario tener:

- Casi cualquier versión de Sistema Operativo (Windows, Mac OS, Linux)
- Servidor web (Apache, Internet Information Server IIS)

- El lenguaje PHP
- Navegador Web (Internet Explorer, Google Chrome, Firefox)
- Motor de base de datos (Opcional si se desea realizar transacciones de datos)
- Editor de texto (Notepad, Wordpad).
- Entorno de desarrollo compatible.

También es necesario especificar algunos componentes de acuerdo a las plataformas y el uso que se va a requerir, para que trabaje en Windows se pueden ocupar los siguientes elementos:

- Instalador PHP 5.1, 5.2 en adelante.
- Microsoft IIS 5.1, 6.0, 7.0 en adelante
- Apache 1.3.x, 2.x
- Servidores Sun, iPlanet y Netscape en Microsoft Windows
- Sambar Server en Microsoft Windows
- Xitami en Microsoft Windows
- Instaladores de extensiones.

3.2 Apache HTTP Server

Es un servidor web que sirve para gestionar datos e información entre páginas web estáticas y dinámicas, su comunicación se basa en el protocolo HTTP, su estructura se basa en el núcleo principal de Apache Server e interactúa con los módulos que son extensiones con las que Apache Server se vuelve compatible para el lenguaje de programación deseado.

3.2.1 Características de Apache HTTP Server

- Tiene soporte para entornos virtualizados que indica que es un “software que simula recursos tecnológicos físicos reales” Wikipedia. (s.f.).
- Soporta un motor de base de datos.
- Transaccionalidad con SSL “Secure Socket Layer que son protocolos para establecer conexiones incrementando la seguridad” Wikipedia. (s.f.).
- Es código abierto.
- Multiplataforma.

- Alta velocidad.
- Destaca por su eficiencia.
- Soporta autenticación.
- Soporta autorización.
- Capacidad alta de mejorar velocidad de procesamiento y respuesta de las peticiones HTTP Caching que es “almacenamiento temporal de recursos utilizados para incrementar la velocidad de trabajo” Wikipedia. (s.f.).
- Mejora de la seguridad en nuevas versiones de Apache HTTP Server.
- Puede trabajar con los lenguajes PHP, Perl “Lenguaje de programación creado en 1987 con cualidades del lenguaje C” Perl. (s.f.), CGI.
- Protección del Kernel que afirma que es la “esencia de las cualidades y recursos del cualquier sistema operativo” Wikipedia. (s.f.), al utilizar extensiones de forma modular,

3.2.2 Ventajas de utilizar Apache HTTP Server

Sus principales ventajas son:

- Su realización es código abierto por Apache HTTP Server Foundation.
- Su instalación e implementación es sencilla.
- Posee una gran cantidad de extensiones lo que hace que sea muy versátil en su funcionamiento.
- Tiene una comunidad que se dedica a mejorar el producto.
- Es el servidor más utilizado en la red.

3.2.3 Popularidad de Apache HTTP Server

Apache es el servidor más utilizado con un 57% de popularidad a nivel mundial en comparación al resto de servidores web disponibles.

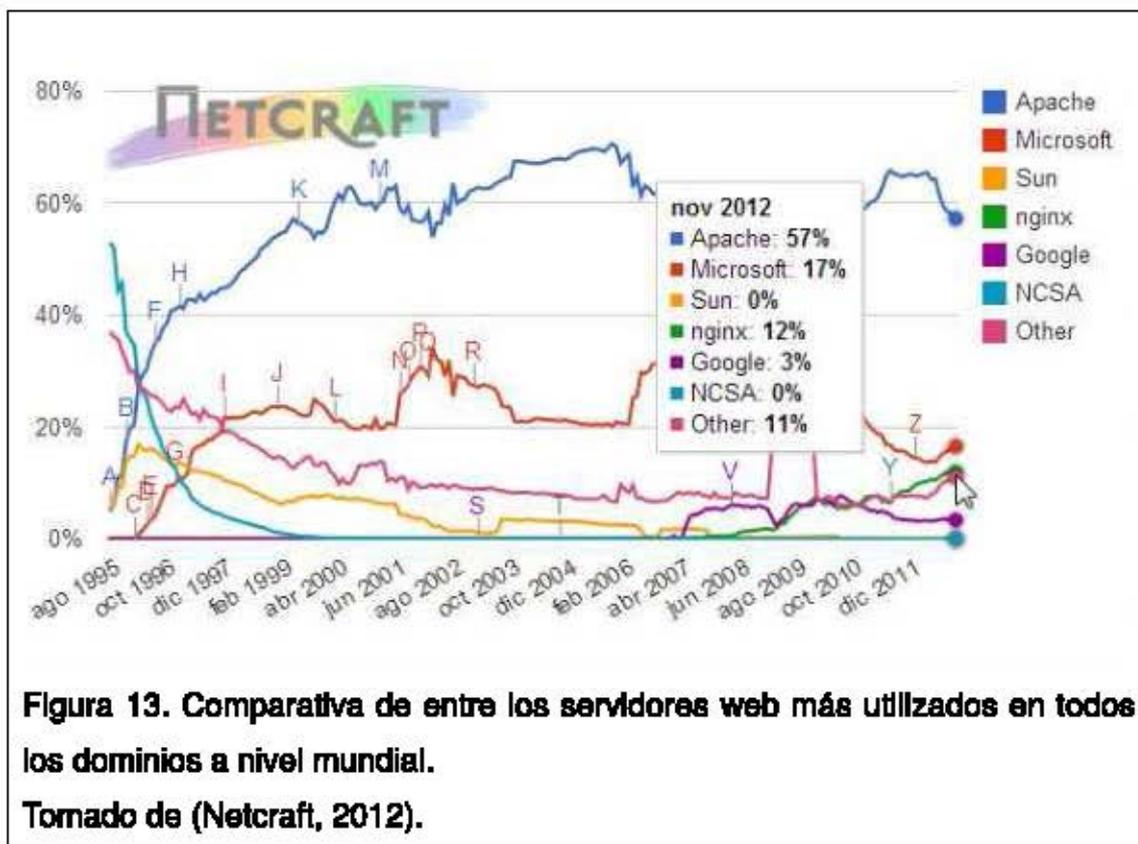


Tabla 6. Mejores servidores entre todos los dominios

Desarrollador	Noviembre 2012	Porcentaje
Apache	357,865,215	57.23%
Microsoft	103,333,170	16.52%
nginx	74,437,764	11.90%
Google	21,090,410	3.37%

Nota: Se describe al servidor Apache como el servidor más utilizado de entre todos los dominios existentes. Tomado de (Netcraft, 2012).

3.2.4 Funcionamiento de Apache Server

En la carpeta donde fue instalado el servidor Apache se encuentran todos los elementos que hacen posible que Apache entre en funcionamiento, pero existen dos directorios en donde se realiza gran parte de su configuración.

- En el directorio conf que se encuentran todos los archivos para configuración y personalización del servidor web, ahí se encuentra httpd.conf que se configura alterando su código.
- La carpeta htdocs es el contenedor contiene las páginas web que fueron desarrolladas y se mostrarán al cliente.

3.2.4.1 Arquitectura de Apache Server

Para entender la arquitectura de Apache HTTP Server es necesario comprender que núcleo que la conforma, y está integrada por los siguientes componentes:

- http_protocol.c: Es el componente de implementación, establece el diálogo http con el usuario además de la transferencia de los datos.
- http_main.c: Es el responsable de tener el servidor cargado a la espera de las conexiones y gestionar los tiempos de conexión.
- http_request.c: Establece comunicación con los módulos si es necesario.
- http_core.c: Se encarga de la funcionalidad principal del servidor web Apache.
- alloc.c: Asigna un conjunto de recursos como memoria o de almacenamiento.
- http_config.c: Contiene funciones de lectura del archivo de configuración.

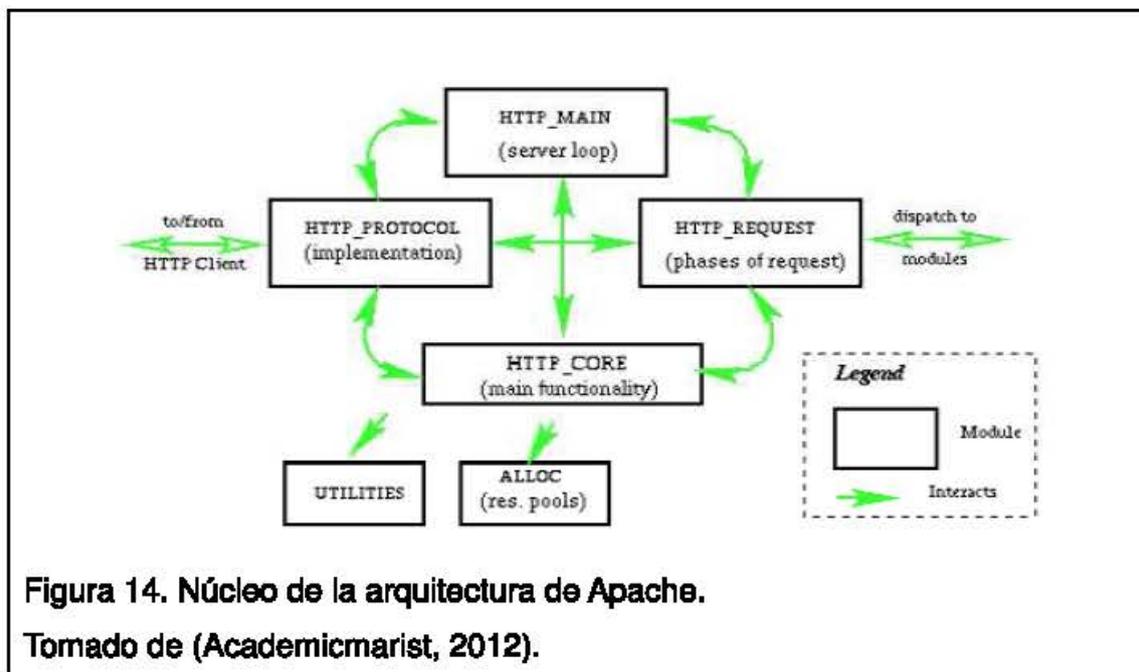


Figura 14. Núcleo de la arquitectura de Apache.
Tomado de (Academicmarist, 2012).

3.2.4.2 Módulos de Apache HTTP Server

Los módulos son extensiones que se agregan al núcleo y sirven para darle al servidor web la funcionalidad que se necesite, por ejemplo podemos agregar el módulo para que el núcleo trabaje con PHP o con Perl.

- mod_php
- mod_perl

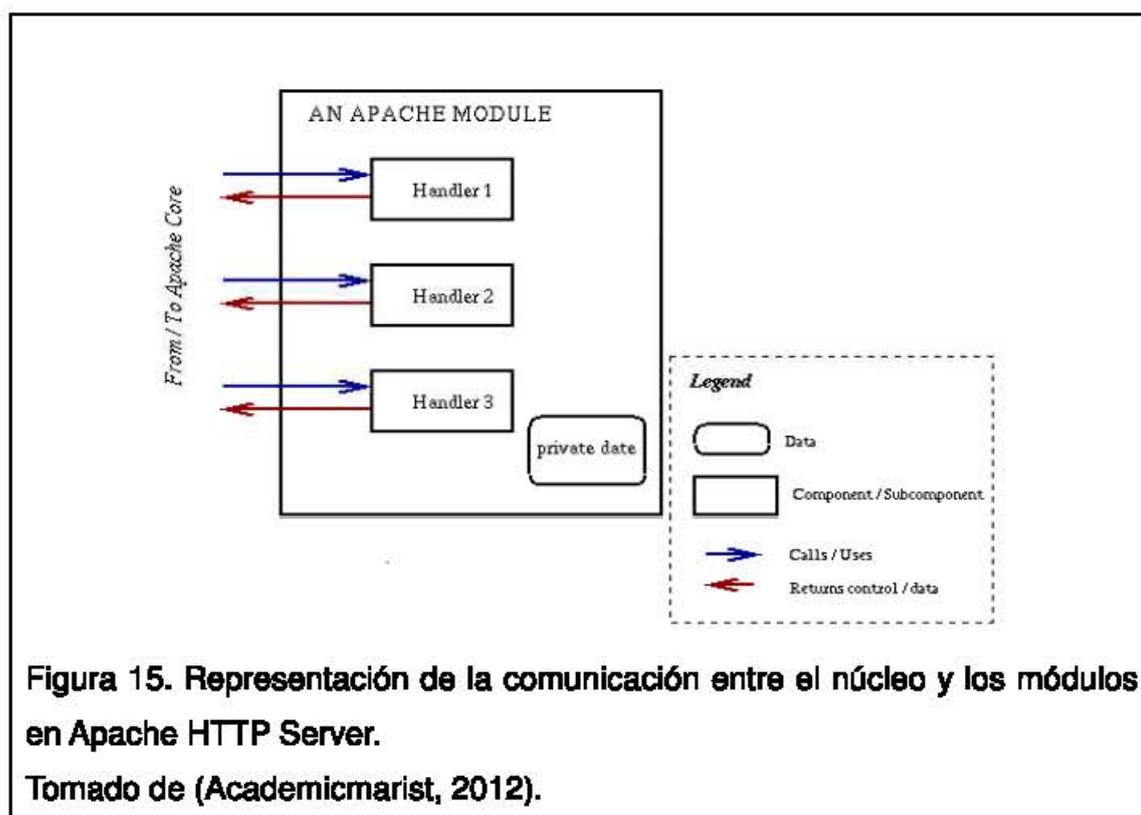
También podemos agregar módulos que nos permitan trabajar con comunicaciones seguras utilizando SSL.

- mod_ssl

O puede ser el trabajo con la gestión de autenticación de usuarios.

- mod_auth_ldap

La ventaja que Apache Server sea modular implica que se ajuste a las necesidades sin perder su funcionalidad inicial debido a que el núcleo permanece intacto.



3.2.5 Requisitos para el funcionamiento de Apache en la plataforma Windows

Apache es soportada en las siguientes versiones de la plataforma Microsoft de servidores.

- Microsoft Windows NT 4.0 con Service Pack 6
- Microsoft Windows Server 2003, 2008, 2008 r2 y 2012

Para Microsoft Windows Cliente.

- Microsoft Windows 2000, XP, Vista, 7 y 8.

Requerimiento mínimo de RAM.

- 256 MB.
- 512 MB en adelante para un mejor rendimiento.

Requisito mínimo para Disco Duro.

- 200 MB de espacio libre

Procesadores x86

- AMD
- Intel

Se recomienda mantener actualizada la versión de Service Pack que es un “programa que corrige las vulnerabilidades y actualiza los sistemas operativos de plataforma Windows” (Windowsmicrosoft, s.f.), de acuerdo al sistema operativo en uso.

3.2.6 Instalación para Windows

La instalación del servidor web Apache es muy sencilla especialmente bajo la plataforma Windows, en donde se debe simplemente bajar desde el sitio oficial <http://httpd.apache.org/download.cgi> cualquier versión disponible para Windows, como por ejemplo una instalación pura de apache sin módulos desde:

- httpd-2.0.64-win32-x86-no_ssl.msi

Una vez que tenemos el instalador lo ejecutamos, captamos las condiciones de uso y se mantiene una Instalación por defecto, existen dos consideraciones a tomar en cuenta al momento de una instalación en la plataforma Windows:

- Inicialmente se deja la configuración del dominio por defecto para posteriormente cambiarla al dominio deseado o el número de dirección IP del servidor.

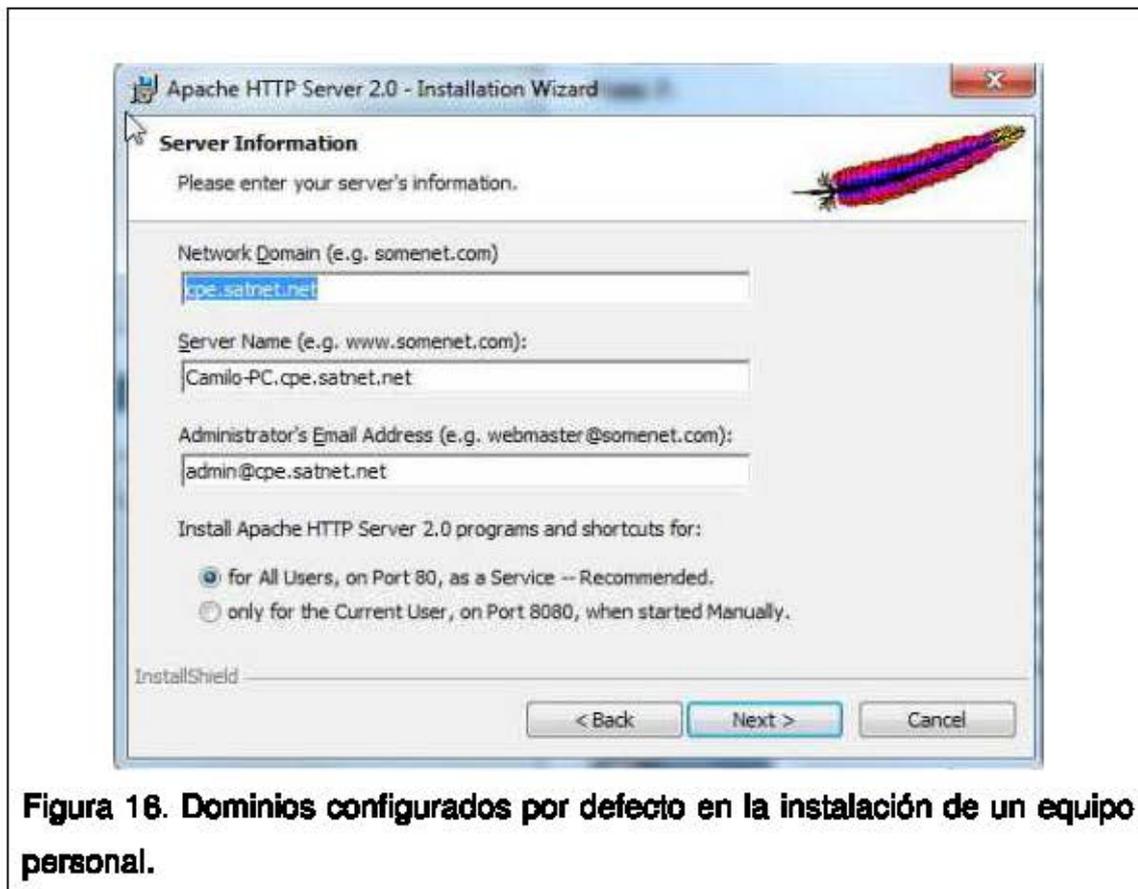


Figura 16. Dominios configurados por defecto en la instalación de un equipo personal.

La ruta en donde se instalará la carpeta puede funcionar con la Instalación por defecto, pero como recomendación se podría instalar en la ruta C:/apache o cualquier ruta que resulte una forma más accesible e independiente para uso compartido, de esta forma se tiene mejor control de permisos y accesos.

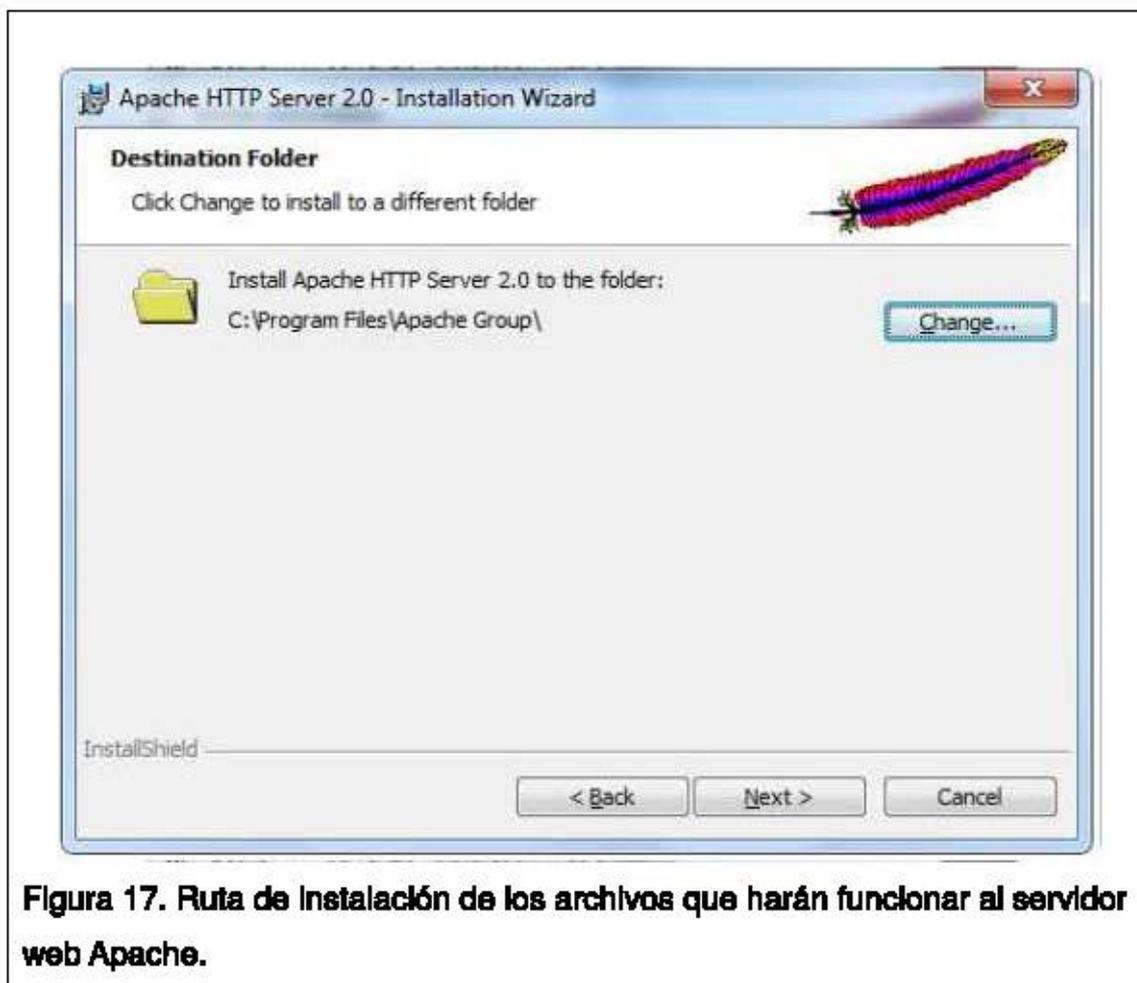


Figura 17. Ruta de instalación de los archivos que harán funcionar al servidor web Apache.

Para verificar el correcto funcionamiento del servidor web Apache se debe ingresar desde cualquier navegador a la dirección IP que es un "identificador único de una tarjeta de red utilizando protocolo IP" (Wikipedia, s.f.), 127.0.0.1, se podrá visualizar la página de confirmación en el que se muestra el mensaje que funciona correctamente.

3.3 MySQL

Es un motor de base de datos MySQL que sirve almacenar y gestionar información guardada en la misma, está desarrollado en C, C++ conocido como el “lenguaje en C extendido creado en 1980” (Cplusplus, s.f.), y es software de distribución libre, existe la venta de licencias para empresas que desean agregarla para gestionar su información debido a que ofrece soporte en sus versiones pagadas como MySQL Enterprise Edition.

3.3.1 ¿Por qué utilizar MySQL?

MySQL es la base de datos de licenciamiento libre más utilizada en el mundo, su comunidad de desarrolladores permiten que siempre sea mejorada, además es compatible con casi todos los sistemas operativos.

Grandes empresas con un alto índice de transaccionalidad a nivel mundial han confiado en la base de datos MySQL para administrar su información debido a su rendimiento, seguridad y fácil manejo, algunos ejemplos son:

- Facebook
- Google
- Ebay
- Adobe

3.3.2 Características de MySQL

- Las principales características son:
- Se trabaja en base al modelo de base de dato relacional.
- Se realizan muchas tareas simultáneamente y de forma concurrente por lo que se dice que MySQL es multihilo que afirma que son “muchas formas básicas de procesamiento que llamados hilos” (Wikipedia, s.f.).
- Es multiusuario porque muchos usuarios pueden acceder a la base de datos al mismo tiempo.
- Seguridad mediante SSL.
- Manejo de contraseñas con cifrado.
- Manejo de portapapeles GNU que afirma “Gnu is Not Unix y es un proyecto para crear software libre” (Gnu, s.f.).

- Se conoce que ha sido usado para albergar 50 000 000 000 registros.
- Soporte para ODBC.
- Soporta *store procedures* que afirman “procedimientos que se guardan directamente en una base de datos para facilitar acceso directo a la información” (Msdn.microsoft, s.f.), *triggers* “Ejecución de técnicas de ingreso a base de datos bajo condiciones” (Docs.oracle, s.f.).
- Maneja encriptamiento de datos que se refiere a la “utilización de algoritmos para ocultar información a fin de obtener confidencialidad” (Aceproject, s.f.) de datos.
- Tiene la capacidad de distribución debido a que puede transaccionar entre varias bases de datos distintas.
- Es multiplataforma, puede trabajar con casi todas las plataformas existentes entre las cuales las más comunes.
 - o Linux
 - o Mac OS
 - o OS/2
 - o Solaris
 - o Windows 95, 98, NT, 2000, XP, Vista, 7, 8.
 - o Windows Server 200, 2003 y 2008.

3.3.3 Usos de MySQL

MySQL es muy conocido en cuanto a la administración de bases de datos basándose prácticamente en:

- Aplicaciones transaccionales de alta o baja concurrencia.
- Aplicaciones de comercio electrónico.

3.3.4 Requerimientos para instalación de MySQL en plataforma Windows

Los requerimientos mínimos que establece MySQL para la instalación en los sistemas operativos Windows son:

- Windows OS de 32 bits.

- Debe tener protocolo TCP/IP “*Transmission Control Protocol/Internet Protocol* es un modelo que se utiliza para tener comunicación y conectividad a través de la red usando protocolos de red IP” (Yale, s.f.).
- Mínimo 200 MB de espacio disponible en el disco
- Para administración mayor a 4GB debe ser trabajar con el disco en formato NTFS “*New Technology File System* es un formato que permite gestionar información en discos de alta capacidad” (Wikipedia, s.f.).

3.3.5 Tipos de datos en MySQL

MySQL soporta la gran mayoría de tipos de datos que existen

Tabla 7. Principales tipos de datos que maneja MySQL.

Grupo	Tipo	Características
Texto	CHAR()	Tamaño fijo, entre 0 y 255 caracteres.
	VARCHAR()	Tamaño variable, entre 0 y 255 caracteres.
	TINYTEXT	Cadena con longitud máxima de 255 caracteres.
	TEXT	Cadena con longitud máxima de 65.535 caracteres.
	BLOB	Cadena con longitud máxima de 65.535 caracteres.
	MEDIUMTEXT	Cadena con longitud máxima de 16.777.215 caracteres.
	LONGTEXT	Cadena con longitud máxima de 4.294.967.295 caracteres.
	LOB	Cadena con longitud máxima de 4.294.967.295 caracteres.
Numéricos	TINYINT()	Desde -128 a 127 con signo. Desde 0 a 255 sin signo (<i>UNSIGNED</i>).
	SMALLINT()	Desde -32.768 a 32.767 con signo. Desde 0 a 65.535 sin signo (<i>UNSIGNED</i>).
	MEDIUMINT()	Desde -8.388.608 a 8.388.607 con signo. Desde 0 a 16.777.215 sin signo (<i>UNSIGNED</i>).
	INT()	Desde -2.147.483.648 a 2.147.483.647 con signo. Desde 0 a 4.294.967.295 sin signo (<i>UNSIGNED</i>).
	BIGINT()	Desde -9.223.372.036.854.775.808 a 9.223.372.036.854.775.807 con signo. Desde 0 a 18.446.744.073.709.551.615 sin signo (<i>UNSIGNED</i>).
	FLOAT	Números pequeños en coma flotante.
	DOUBLE(,)	Números grandes en coma flotante.
	DECIMAL(,)	Double almacenado como una cadena, para números en coma fija.
Tiempo	DATE	AAAA-MM-DD
	DATETIME	AAAA-MM-DD HH:MM:SS
	TIMESTAMP	AAAAMDDHHMMSS
	TIME	HH:MM:SS
Otros	ENUM()	Cada columna puede tener uno de los valores especificados.
	SET	Similar a ENUM(), pero en este caso cada columna puede tener más de uno de los valores especificados.

Nota: Tomado de (W3resources, 2012).

3.4 El servidor XAMPP

Es un servidor de código abierto que sirve para administrar la comunicación y gestión entre el lenguaje de programación PHP, el servidor web Apache la base de datos MySQL, de ahí su nombre WAMP (Windows, Apache, MySQL, PHP), está integrada por los siguientes elementos de software en la versión XAMPP Windows 1.8.1.

Tabla 8. Elementos que incluye el paquete XAMPP Server.

Elementos de XAMPP Sever
Apache 2.4.2
MySQL 5.5.27
PHP 5.4.7
OpenSSL 1.0.1c
PhpMyAdmin 3.5.2.2
XAMPP Control Panel 3.1.0
Webalizer 2.23-04
Mercury Mail Transport System v4.62
FileZilla FTP Server 0.9.41
Tomcat 7.0.30 (with mod_proxy_ajp as connector)
Strawberry Perl 5.16.0.1 Portable
Soporte para Windows 2000, XP, Vista, 7.

Nota: Tomado de (Apachefriends, 2012).

3.4.1 Características del servidor XAMPP

Las principales características del servidor Xampp son:

- Es multiplataforma soporta las más importantes del existentes.
- Gestiona los servicios entre MySQL y Apache
- Contiene Apache, MySQL y PHP en un solo paquete integrado.
- WAMP es un servidor seguro.
- Instalación sencilla.
- Incluye PHPMyAdmin que es una interfaz para la gestión de MySQL.

3.4.2 ¿Por qué utilizar el servidor XAMPP?

Porque es un servidor para un entorno de desarrollo en lenguaje PHP y tiene las siguientes ventajas son:

- Después de ser instalado ya está listo para ser utilizado
- Las actualizaciones se basan en mejoras a nivel de archivos
- El registro no tiene que ser modificado,
- Es configurable para implementar conexiones seguras.
- Existe una comunidad encargada de solventar *bugs* que son "Fallas no esperadas en aplicaciones de software" (Wikipedia, s.f.).
- Es de software libre.
- Existe amplia documentación sobre el servidor XAMPP.
- Puede servir tanto como para pruebas como para producción.
- Su interfaz es muy intuitiva y adaptable incluso para usuarios sin mucha experiencia.
- Políticas de seguridad configurables de acuerdo a las necesidades.
- Acceso a los servicios es muy sencilla.

3.5 HTML5

Es una de las nuevas tecnologías web y es la sucesión del lenguaje de desarrollo web HTML a la versión 5, HTML5 nos permite crear y editar páginas web de una forma muy versátil debido a que se han registrado amplias mejoras del lenguaje que incluso ha sido capaz de reemplazar a aplicaciones como flash. Estas mejoras nos permite desarrollar soluciones para ambientes de desarrollo modernos, HTML5 es un estándar de la W3C.

3.5.1 Características de HTML 5

- Posee etiquetas que permite una mayor amplitud para desarrollo en Canvas que es un "elemento integrado en HTML que sirve para trabajar con gráficos y realizar animaciones" (W3schools, s.f.), 2D y 3D o cualquier animador compatible.
- Soporte con audio y video.
- Incorpora controladores para mostrar

- Compatibilidad con JavaScript “Lenguaje complementario de HTML utilizado para controlar los elementos en un navegador” (W3schools, s.f.).
- Permite realizar animaciones.
- Soporte para reproducción de audio en vivo.

3.5.2 ¿En qué proyectos se utiliza HTML5?

- Aplicaciones web
- Aplicaciones móviles
- Soluciones empresariales
- Desarrollo de juegos
- Comercio electrónico

3.5.3 Ventajas de HTML5

- Posee extensión con la que puede ser compatible también con el navegador Google Chrome.
- Existe una vasta documentación y tutorías de desarrollo.
- Es código abierto.
- Es una alternativa libre de la popular Flash que se enfoca especialmente en desarrollo multimedia.
- Se pueden crear animaciones y fácilmente exportarlas desde una PC a cualquier dispositivo multimedia.
- Reducción de costos de implementación.
- Debido a sus nuevas funciones es la seguridad de las aplicaciones web son simplificadas.

3.5.4 Funciones principales de HTML5

- Aplicaciones “*always-offline*” sin conexión que se pueden adaptar a trabajar con un navegador personal como lo hacen los iPad.
- Aplicaciones “*online-offline*” en las que la aplicación puede adaptarse a tiempos sin conexión para sincronizarse con una nube “Conjunto de servicios accedidos mediante internet desde cualquier dispositivo” (Wikipedia, s.f.).
- Rompe la clásica barrera en la que el almacenamiento va en un espacio en el servidor, HTML5 está capaz de almacenar datos del lado del cliente.

- Se ha mejorado la calidad con respecto a conectividad de esta forma se puede trabajar con chats o juegos en línea de una forma optimizada.
- Permite el acceso a archivos de forma recurrente.
- Existe una nueva semántica mejorada para componentes multimedia, atributos, formularios y sintaxis internacional mejorada.
- El contenido multimedia de audio y video ha mejorado enormemente como por ejemplo ahora se pueden aplicaciones para mezclar música que interactúe con selección de un ambiente de video.
- Para crear elementos gráficos en 2D y 3D solo se podía realizar mediante Flash o Javascript, ahora HTML5 incorporó elementos Canvas, WebGL y SVG que permite realizarlo directamente.
- Gracias a CSS3 la presentación es mejorada para sitios modernos en los que ya se puede incluir casi cualquier tipo de elemento o animaciones.
- Mejora en el rendimiento al punto de compararse con el rendimiento de una aplicación de escritorio.
- Soporte para elementos de ubicación geográfica incluida.

3.5.4.1 Navegadores compatibles con HTML5

- Internet Explorer
- Chrome
- Firefox
- Safari
- Opera (Mini, Mobile)
- iOS
- Android

3.5.5 ¿Por qué utilizar HTML5?

HTML5 se ha desarrollado tanto que en una sola tecnología ha sido capaz de integrar una gran variedad de soporte para elementos que antes se lo tenía que hacer mediante aplicaciones externas y pagadas, convirtiéndose en una herramienta muy versátil para desarrolladores web, HTML5 ha evolucionado tanto que su rendimiento ha incrementado de tal forma que puede convertirse en el lenguaje para aplicaciones seguras, veloces y con un diseño adaptable a las necesidades actuales. Además que es de uso de licencia libre.

3.6 Frameworks web

En toda actividad relacionada con la información sea o no de computo, se ha llegado a la conclusión que es mejor administrar la información de la forma más ordenada posible, así facilitamos el acceso al almacén de datos de manera más eficiente. Sean medios digitales o físicos, la administración la información bien estructurada se vuelve sencilla implementando el orden de los datos. De esta forma para los desarrolladores de software la información que manejan en creación de software es mejor administrarla ordenadamente así podemos estructurar toda la información utilizada para producir software como por ejemplo librerías, código fuente, archivos, etc.

Un framework es un modelo que sirve a los desarrolladores para crear una solución de software, cada lenguaje de programación tiene su propio framework en el cual basar su desarrollo. Este modelo es una base definida e incompleta en la cual que los desarrolladores se basan para configurarla y obtener un producto final de software que puede ser una aplicación web, un servicio web o un sitio web dinámico.

3.6.1 Ventajas de utilizar frameworks

Las ventajas principales de utilizar frameworks son:

- El código puede ser utilizado nuevamente.
- Mayor velocidad de desarrollo.
- Facilita el desarrollo de una solución de software.
- Fomenta las prácticas de programación a causa de los patrones a seguir.

- Ventaja a medida que aumente la dificultad y el tamaño del software.
- Disponibilidad de librerías adaptadas del framework.
- Cualquier cambio en los integrantes de desarrollo puede ser solventado con mayor facilidad debido a los patrones establecidos.
- Integración con Base de datos.
- Dentro del framework se incluye la autenticación de usuarios.

3.6.2 Tipos de framework

Existen varios tipos de frameworks que se utilizan para dar soluciones a múltiples necesidades tecnológicas de una empresa como:

- Gobierno "Normas y procedimientos para administrar recursos tecnológicos" (Tepsi, s.f.), de TI
- Rendición de cuentas
- Autorización
- Controles TI
- Procesos

Para el presente proyecto se utilizará un framework para desarrollo de aplicaciones web, la mayoría de frameworks son basados en el patrón MVC, se diferencian de la siguiente manera:

- Modelo – Vista – Controlador (MVC)
- Push / Pull
- Tres niveles

MVC. Patrón que se caracteriza por separar la lógica del negocio de la interfaz de usuario y de los datos.

- Push. Empuja los datos mediante acciones que el framework realiza directamente hacia la capa de presentación.
- Pull. Obtiene los resultados desde múltiples controladores.

- Tres niveles Trabaja utilizando la capa de cliente que hace peticiones HTTP, datos que se refiere al a base de datos y aplicación que contiene la lógica del negocio, a diferencia del patrón MVC en tres niveles la lógica del negocio se encuentra separada del controlador.

3.6.2.1 Características de los frameworks

Como principales características se encuentran:

- Usa plantillas para la creación de páginas dinámicas que son “paginas HTML que se ejecutan obteniendo y cargando información desde el servidor” (Wikipedia, s.f.), puede utilizar en base a una sola plantilla dinámica la conexión hacia múltiples registros de datos.
- Utilización de caching que son medios para reducir el ancho de banda mediante reutilización de datos para múltiples escenarios.
- Inclusión de autenticación y autorización para mejoramiento de la seguridad.
- Trabajan con acceso y administración de la base de datos.
- Descripción creada por el programador llamada *Scaffolding* que se refiere a “cómo utilizar y acceder al motor de base de datos desde un framework” (Wikipedia, s.f.), que sirve para que el framework interactúe con la base de datos o realizando un CRUD que significa “*Create Read Update Delete* son las manipulaciones más esenciales en una base de datos” (Wikipedia, s.f.).
- El mapeo URL “Uniform Resource Locator es un conjunto de caracteres que utilizan un formato para acceder a recursos mediante internet” (Docs.oracle, s.f.), simplifica los elementos de los URL haciendo que las búsquedas y la indexación sea más eficientes.
- Soporte Ajax que significa “*Asynchronous JavaScript And XML* es una forma de acceder y manipular datos en el servidor utilizando JavaScript en segundo plano mientras se visualiza una página” (W3schools, s.f.).
- Algunos frameworks permiten el soporte para la creación de servicios web.
- Existen frameworks software libre o de propiedad.

3.6.2.2 Ejemplos de framework

Todos los framework se encargan de separar la lógica del negocio, los datos y la presentación pero utilizan una diferente técnica de administración y manejo. Los siguientes frameworks web son los más importantes y destacados para el año 2012.

Tabla 9. Framework web destacados para el año 2012

Web Application Frameworks 2012	
Ruby on Rails	Framework MVC basado en Ruby, orientado al desarrollo de aplicaciones web
CodeIgniter	Poderoso framework PHP liviano y rápido
Django	Framework Python que promueve el desarrollo rápido y el diseño limpio
CakePHP	Framework MVC para PHP de desarrollo rápido
Zend Framework	Framework para PHP 5, simple, claro y open-source
Yii	Framework PHP de alto rendimiento basado en componentes
Pylons	Framework web para Python que enfatiza la flexibilidad y el desarrollo rápido
Catalyst	Framework para aplicaciones web MVC elegante
Symfony	Framework full-stack
TurboGears	Próxima generación construido sobre Pylons

Nota: Descripción general de los frameworks para aplicaciones web más destacados para el año 2012. Tomado de (Linuxtoday, 2012).

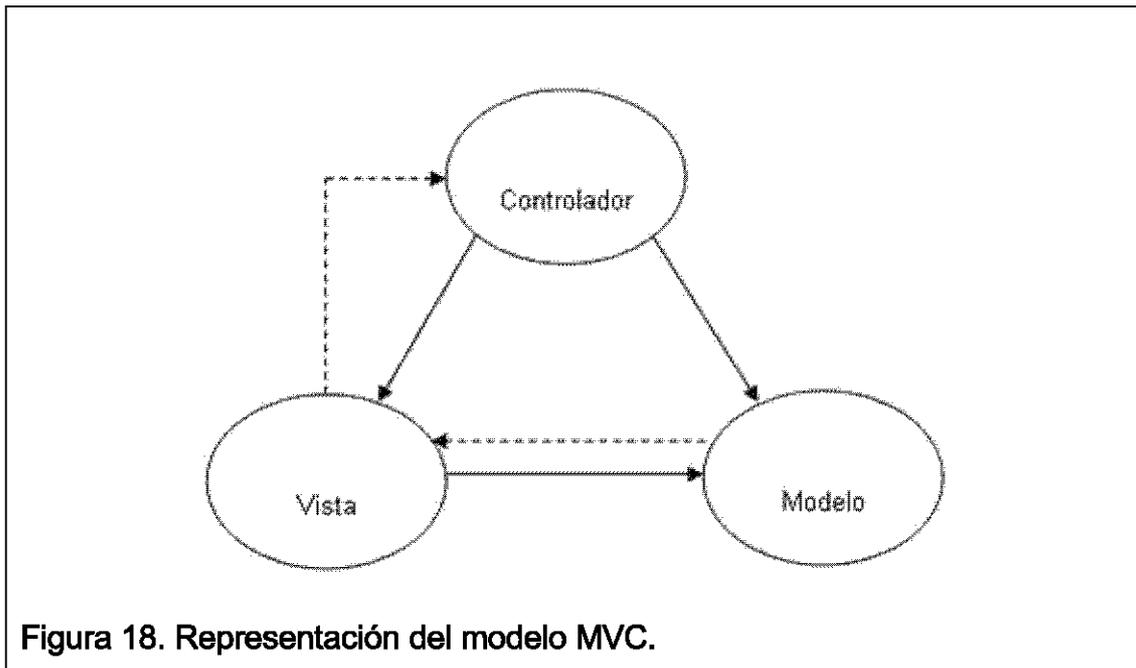
3.7 Patrón MVC

Es un patrón para el desarrollo de software web, sirve para trabajar en 3 capas separando la lógica del negocio, los datos y la presentación hacia el usuario final. Es un patrón de desarrollo muy utilizado en frameworks web, tiene bien definidas las funciones con las que trabaja, evitando mezclar código utilizando independientemente el Modelo – Vista – Controlador que hace referencia a su nombre MVC. Su diseño fue creado para facilitar la construcción de aplicaciones reduciendo el esfuerzo de desarrollo a los programadores.

Sus aplicaciones sirven para:

- Creación de sitios web
- Aplicaciones web
- Aplicaciones de escritorio

Tradicionalmente se utilizaba el código en una misma página llevando a que los roles de los desarrolladores sean dependientes entre sí, agilitando los típicos problemas debido a retraso entre roles, ya que el acceso a datos no podría seguir adelante mientras el diseñador no acabe su trabajo. Estas situaciones son solucionadas con el patrón de diseño MVC debido a que los roles pueden trabajar de manera independiente en su capa.



3.7.1 Modelo

Es la capa encargada de interactuar directamente con la base de datos, como sus características principales podemos encontrar:

- Trabajo directamente con la base de datos.
- Dentro del modelo se encuentra la lógica del negocio.
- EL cambio que se realice en el modelo se refleja en la vista.
- No hace referencias a controladores.
- No hace referencias a vistas.
- Operaciones de acceso a base de datos.
- Contenedor de clases del sistema.

3.7.2 Vista

Es la capa que se encarga de mostrar al usuario la presentación del software desarrollado, representa visualmente las acciones que realiza el usuario y que van a ser interpretadas por el sistema.

Sus principales características son:

- Representación visual de datos y eventos
- Permite la interacción directa con el usuario.
- Muestra datos obtenidos del modelo.
- Se pueden vincular muchas vistas a un solo modelo.
- Es adaptable a las necesidades del cliente.
- Se actualiza de acuerdo a los cambios en el modelo.
- Es la página HTML propiamente dicha.
- Permite recibir eventos generados por el usuario.

3.7.3 Controlador

En esta capa se encuentra la interacción entre el modelo y la vista, recibe eventos y controla el flujo entre los datos y lo que se muestra al usuario.

Sus principales características son:

- Comunicación entre la vista y modelo.
- Recibe las acciones enviadas por el usuario a través de la vista.
- Solicita al modelo las operaciones necesarias enviadas desde la vista.
- Envía los resultados desde el modelo hacia la vista.
- Obtención del código HTML.
- Puede modificar el modelo.
- Trabajo dinámico.

3.7.4 Características del patrón MVC

Las principales características del patrón MVC son:

- Funcionalidades de Modelo, Vista y Controlador bien definidas.
- Generalmente se utiliza con web browser HTML.
- Independencia para acceder a la capa de datos.
- Utiliza reglas de negocio bien estructuradas y detalladas.
- Independencia de la capa de almacenamiento.
- Se pueden crear muchas vistas diferentes en base a un solo modelo.
- Implementación sencilla.
- Escalabilidad.

3.7.4.1 Orientación a Objetos con MVC

Para entender la relación entre MVC y POO "Programación Orientada a Objetos que son elementos o unidades de software que tienen un estado" (Msdn.microsoft, s.f.), debemos citar las características y elementos esenciales que llevan a crear la POO.

Los principales elementos de la POO son:

- Clases es un modelo en el que se guarda la información de un objeto como puede ser una "persona".
- Objetos son conceptos del mundo real que son representados como una instancia de clases, puede ser una persona llamada "Camilo".
- Métodos es un algoritmo que muestra que es lo que puede hacer un objeto como una persona llamada Camilo puede "correr".
- Eventos son la forma cómo interactúan las diferentes clases.
- Atributos se refiere a la información y datos de una clase.
- La POO tiene como características principales:
- Abstracción son las propiedades fundamentales de un objeto.
- Herencia sirve para utilizar como extensión los comportamientos y estados de una clase padre hacia clases hijas.
- Polimorfismo es la propiedad para utilizar un método para diferentes objetos de esta forma tenemos muchos comportamientos de respuesta a un mismo

mensaje, podemos decir que una persona llamada Camilo “duerme” pero no todas las personas lo hacen de la misma manera.

- Encapsulamiento es capaz de mantener oculta la información de datos y que los objetos sean modificables mediante la definición de métodos.

El patrón de diseño MVC está ligada estrechamente a la POO debido a que para que el patrón funcione necesita tanto de los elementos que intervienen en la POO como de sus características, el patrón simplemente es una comunicación entre las diferentes capas modelo, vista y controlador que le permite al usuario acceder e interactuar con los objetos que fueron creados para desarrollar una solución de software.

3.7.5 Beneficios de utilizar MVC

Son muchos los beneficios de utilizar el patrón de diseño MVC, se pueden nombrar los siguientes:

- Desarrollo modular.
- Escalable a aplicaciones de gran dimensión.
- Funcionamiento de componentes de forma independiente.
- Los cambios se producen de forma automática en su ejecución.
- Vistas personalizables.
- Rápido desarrollo de aplicaciones.
- Reutilización de código.
- Desarrollo modular de aplicaciones.
- Patrón de desarrollo de funcionamiento comprobado.
- Organización en cuanto al código.
- Escalabilidad de aplicaciones.
- Actualización automática de los datos en las vistas.
- Creación de soluciones rápida.
- Mejora en tiempos de mantenimiento debido a su estructura ordenada y estructurada.
- En caso de cambio de personal es mucho más sencillo que el siguiente desarrollador tome la posta de las funciones debido al código limpio que

indica que son “métodos y maneras para mantener el código de programación ordenada y con buenas prácticas” (Dosideas, s.f.).

- Mejor manejo en cuanto a la realización de pruebas.
- Coexistencia de aplicaciones compartiendo datos.

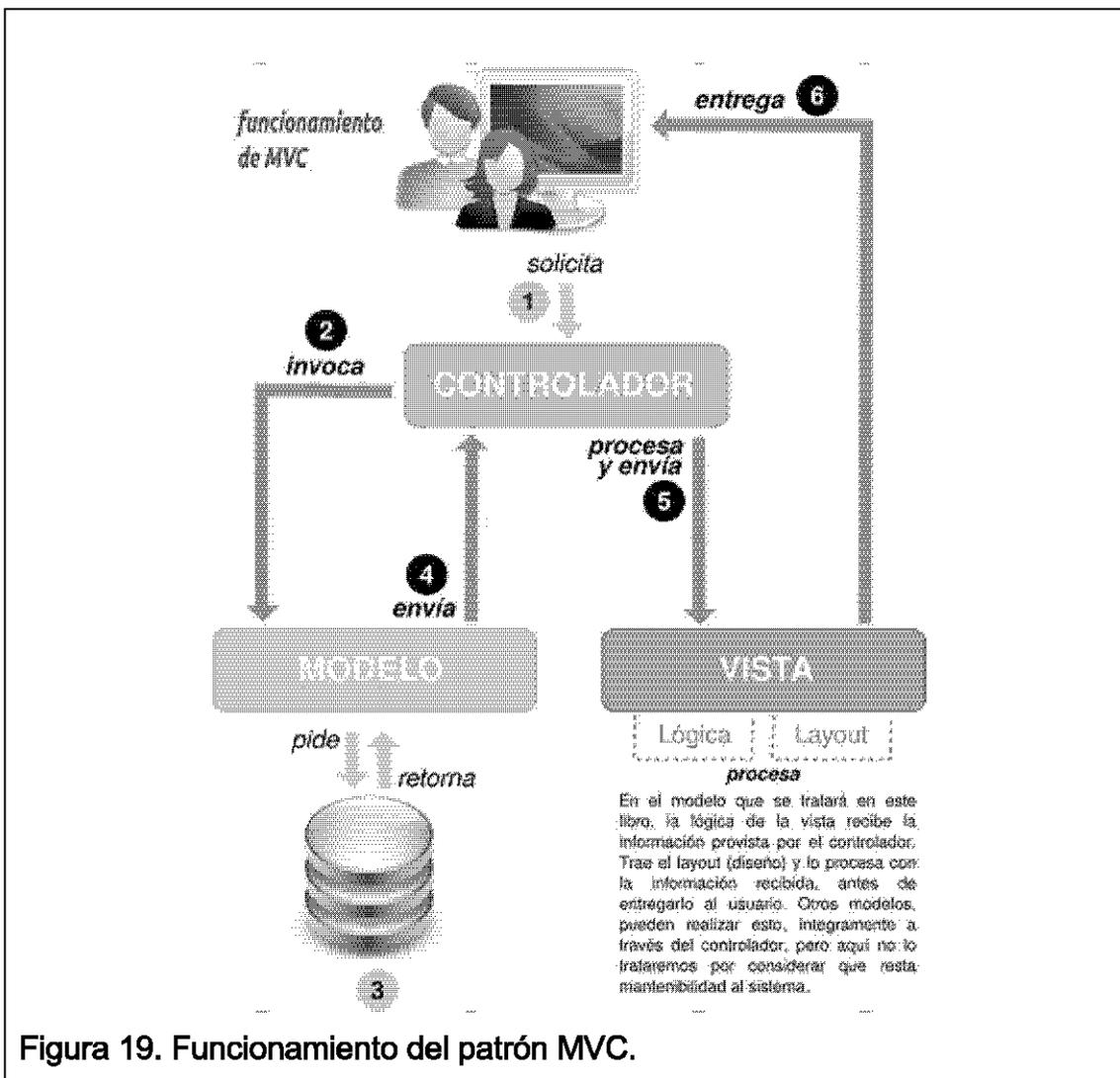
En general el patrón MVC es simplemente un patrón de desarrollo, una forma a la que el desarrollador se debe adaptar para crear soluciones, debido a la diversas formas del pensamiento humano es posible que para unas personas resulte más práctico utilizar un patrón o una metodología diferente, esto debido a la curva de aprendizaje del framework o simplemente por la falta de adaptación al mismo.

3.7.6 Flujo en un patrón MVC

El patrón MVC tiene su propia forma de trabajo, establece una comunicación definida entre las diferentes capas que la conforman, estas capas se comunican estableciendo un flujo de trabajo con el que se obtiene su correcto funcionamiento.

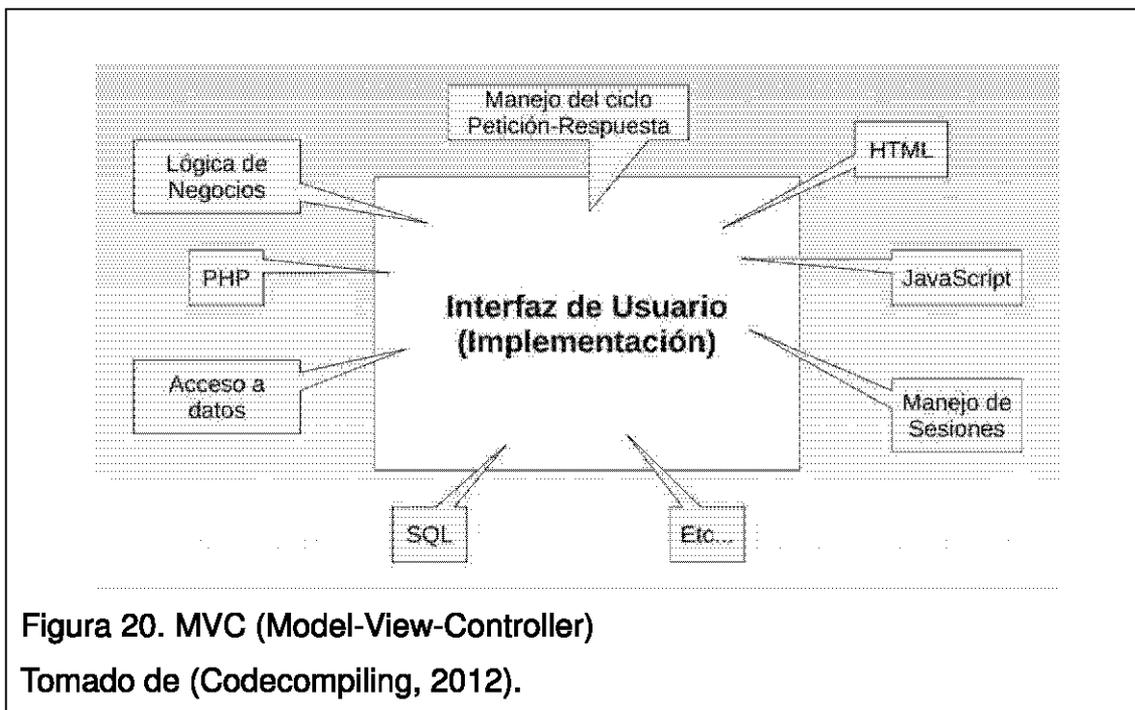
Los pasos que un patrón de desarrollo MVC utiliza son:

- Se produce la interacción del usuario con la presentación de la aplicación, se ejecuta una acción decidida por el usuario.
- La acción es enviada hacia la capa de controlador el cual recibe el evento y decide que es lo que va a realizar.
- Se produce una comunicación con la capa modelo diciéndole que parte la base de datos la capa modelo tiene que modificar u obtener los resultados.
- La capa modelo devuelve los resultados a la capa controlador.
- La vista recibe los resultados obtenidos de la modificación u obtención de información desde la base de datos y los muestra al usuario.
- Todo el proceso se repita cuantas veces el usuario quiera interactuar con la aplicación para obtener resultados o modificar información.



3.7.7 Creación de código sin MVC

Para la creación de soluciones web que es el “desarrollo de aplicaciones web que ayudan a solventar necesidades en una empresa” (Wikipedia, s.f.), los programadores muchas veces tienen a mezclar código, esto se va corrigiendo de acuerdo a la experiencia y utilización de buenas prácticas que se van adquiriendo con el transcurso del tiempo, los MVC nos obligan a emplearlas desde el inicio.



3.8 Yii Framework

Es un framework basado en el patrón Modelo – Vista – Controlador para desarrollar soluciones de software web en lenguaje PHP, su origen data del año 2008 y fue en base a solucionar inconvenientes con respecto a rendimiento, compatibilidad, personalización y facilidad de uso del framework PRADO “PHP Rapid Application Development Object-oriented es un componente basado en el lenguaje PHP para crear aplicaciones web” (Pradosoft, s.f.), y fue nombrado Yii por su idea conceptual “*Yes It Is*”.

3.8.1 Características de Yii

- Las principales características que tiene Yii son:
- Es un framework de lenguaje PHP.
- Se puede desarrollar todo tipo de soluciones.
- Basado en MVC.
- Es un framework muy liviano para ejecutar
- Mayor velocidad de desarrollo de soluciones de aplicaciones web.
- Acceso a base de datos.

- Soporte para compatibilidad con jQuery que son “librerías de JavaScript para facilitar la comunicación con elementos HTML” (jQuery, s.f.).
- Soporte para validación autenticación y autorización.
- Es personalizable
- Personalización para el control y gestión de errores.
- CRUD automático.
- Funcionalidades de Ajax incluidas.
- Caching
- Es un framework orientado a objetos
- Soporta migración de base de datos.
- Alta seguridad con la implementación del componente CHtmlPurifier que elimina código malicioso especialmente de ataques externos.
- Posee extensiones que lo hacen aún más versátil.
- Generación automática de código.

3.8.2 Requerimientos para usar Yii

Los requerimientos mínimos para que el framework Yii trabaje son:

- PHP 5.1.0 o superior
- Apache HTTP Server
- Plataformas Windows y Linux

3.8.3 Ventajas de Yii

Las ventajas que ofrece Yii son:

- Utilizable para aplicaciones de alto rendimiento.
- Amplia documentación.
- Video tutoriales en línea.
- La comunidad de desarrolladores de Yii desarrollan un gran número de extensiones adaptándose a cada necesidad.
- Versiones disponibles de Yii.
- JQuery integrado.
- Validación desde el cliente.
- Framework sencillo pero potente.

- Baja curva de aprendizaje.
- Librerías de JavaScript integradas.
- Es software de distribución libre.
- En muchas comparativas está considerado como el MVC número 1 basado en PHP por su facilidad y rendimiento.
- Soporta aplicaciones con alta concurrencia.
- Ayuda a prevenir ataques como el Inyección SQL que es un “método que inserta código para obtener datos o modificar una base de un sitio que contenga vulnerabilidades” (Php, s.f.), XSS “*Cross-Site Scripting* es una vulnerabilidad de software que permite el ataque introduciendo código *script*” (Owasp, s.f.), CSRF “*Cross-site request forgery es una técnica que utiliza código no autorizado para obtener autorización y acceso a sitios web*” (Owasp, s.f.).

3.8.4 Versionamiento de Yii

Desde su creación en Diciembre de 2008 Yii ha evolucionado hasta encontrarse en la versión 1.1.12.

Tabla 10. Versiones de Yii desde sus inicios.

Version	Lanzamiento	Fecha de lanzamiento	Fin de mantenimiento	Requerimientos
1.0	1.0	3 de Diciembre 2008		
	1.0.1	4 de enero, 2009		
	1.0.2	1 de febrero, 2009		
	1.0.3	1 de marzo, 2009		
	1.0.4	5 de abril, 2009		
	1.0.5	10 de mayo, 2009		
	1.0.6	7 de junio, 2009	31 de diciembre, 2010	PHP 5.1.0 o superior.
	1.0.7	5 de julio, 2009		
	1.0.8	9 de agosto, 2009		
	1.0.9	6 de septiembre, 2009		
	1.0.10	18 de octubre, 2009		
	1.0.11	13 de diciembre, 2009		
1.0.12	14 de marzo, 2010			
1.1	1.1.0	10 de enero, 2010		
	1.1.1	14 de marzo, 2010		
	1.1.2	2 de mayo, 2010		
	1.1.3	4 de julio, 2010		
	1.1.4	5 de septiembre, 2010		
	1.1.5	14 de noviembre, 2010		
	1.1.6	16 de enero, 2011	31 de Diciembre, 2013 o más	PHP 5.1.0 o superior
	1.1.7	27 de marzo, 2011		
	1.1.8	26 de junio, 2011		
	1.1.9	1 de enero, 2012		
	1.1.10	12 de febrero, 2012		
	1.1.11	29 de julio, 2012		
1.1.12	19 de agosto, 2012			
2.0	TBD	TBD	TBD	PHP 5.3.0 o superior

Nota: Tomado de (Wikipedia, s.f.).

3.8.5 Comparativa de Yii contra otros frameworks populares

Se puede hacer una comparación entre 3 framework PHP muy utilizados en el desarrollo de aplicaciones web.

Tabla 11. Cuadro comparativo entre 3 frameworks PHP populares.

PHP Framework	PHP4	PHP5	MVC	Múltiples BD	ORM	Objetos BD	Plantillas	Caching	Validación	Ajax	Auth	Modules	EDP
CakePHP	✓	✓	✓	✓	✓	✓	-	✓	✓	✓	✓	✓	-
Yii	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Symfony	-	✓	✓	✓	✓	✓	-	✓	✓	✓	✓	✓	-

Nota: ORM Indica si el marco apoya un mapeador objeto-relacional, por lo general una implementación de Active Record que es un “patrón de almacenamiento de datos en una base” (Yiiframework, s.f.). Auth: Indica si el marco tiene un módulo incorporado para manejar la autenticación de usuario. EDP: Programación orientada a eventos. Tomado de (Phpframework, 2012).

3.9 Eclipse PDT

Es un IDE “*Integrated Development Environment* es una aplicación que integra los recursos para la utilización de determinado lenguaje de programación” (Wikipedia, s.f.), de desarrollo que sirve para facilitar el desarrollo en lenguaje PHP, por sus siglas significa PHP Developer Tools, su utilización es muy popular ya que Eclipse posee IDE de desarrollo para varios lenguajes y se ha posicionado como uno de los más completos y confiables entornos de creación de software.

Eclipse PDT es de licenciamiento libre gracias a eso existe una comunidad de desarrolladores que brindan soporte para los bugs que vayan saliendo en la marcha de un proyecto, además se pueden agregar componentes y extensiones a fin de tener una herramienta más diversa para desarrollo en PHP.

3.9.1 Características Eclipse PDT

- Tiene su funcionalidad principal para el lenguaje PHP.
- Mejora con los bugs producidos en las primeras versiones.
- Realiza autocompletado en código.

- Adaptación a frameworks y API "*Application Programming Interface* es la conexión para manipulación de elementos y componentes de software" (Wikipedia, s.f.).
- Se acopla con todas las herramientas de desarrollo, HTML5, CSS3, MVC, etc.

3.9.2 Requisitos de instalación para Eclipse PDT

Los requisitos para la instalación de Eclipse PDT son:

- Java Runtime Environment (JRE) preferiblemente la última versión.
- Configurado el servidor XAMPP.
- Estructurado Yii Framework en el servidor XAMPP.
- Eclipse SDK "*Software Development Kit* es un conjunto de herramientas basadas en un cierto lenguaje de programación para que sirve para crear software" (Wikipedia, s.f.), Classic desde la versión Helios hasta la más actual.
- Extensión PHP Development Tools (PDT).

Una vez instalada cualquier versión de Eclipse SDK Classic se configura para que se extienda a la versión PDT.

3.9.3 Compatibilidad con Yii

Eclipse PDT es compatible con Yii con su IDE PDT, existen muchos usuarios en el mundo que lo utilizan, adicional se puede utilizar una extensión de Yii framework para Eclipse PDT que se llama Yiclipse que es un complemento exclusivamente para la funcionalidad del framework Yii.

Existe otra extensión para Eclipse SDK Classic que es un *plug-in que se refiere a "componentes que se extienden al software para agregar funcionalidades específicas"* (Wikipedia, s.f.), de Aptada es un IDE que su enfoque siempre ha sido hacia PHP.

CAPITULO 4

DESARROLLO DE LA SOLUCIÓN

4.1 Metodología Scrum en el proyecto

Para el siguiente proyecto de desarrollo se seguirá la secuencia de eventos y pasos necesarios para el correcto desenvolvimiento de la metodología Scrum.

4.1.1 Equipo de trabajo

Las personas involucradas para la creación del proyecto son:

- Product Owner
Dr. Marco Medina Vega
- Scrum Master
Ing. Juan José León
- Equipo de desarrollo
Camilo Fonseca

4.1.2 Descripción de los roles

4.1.2.1 Product Owner

- El Dr. Marco Medina indicará las necesidades del área de periodoncia de la empresa MF Dental y requerimientos del sistema las cuales se transformarán en historias de usuario.
- Priorización de historias de usuario.
- Revisión de producto terminado.
- Notificación para el cambio de priorización en historias de usuario de ser el caso.

4.1.2.1 Scrum Master

- Dirigir la reunión de sincronización diaria.
- Dirigir reuniones de inicio de sprint y con el cliente.
- Aseguramiento del desempeño de procedimientos correctos en el desarrollo del producto.

- Motivación al equipo de trabajo.
- Facilita que el cumplimiento de cada *sprint*
- Mitigación de riesgos que impidan el cumplimiento del *sprint*.

4.1.2.3 Equipo de desarrollo

Debido a que el proyecto involucra una sola persona para el desarrollo Camilo Fonseca tomará las capacidades de desarrollador y diseñador.

4.2 Inicio del proyecto

Para iniciar el proyecto los involucrados tomarán un tiempo de 5 días en los cuales se tendrán reuniones para establecer los requerimientos e historias de usuario de MF Dental, a este tiempo lo llamaremos Sprint 0.

4.2.1 Actividades previas al arranque del proyecto

Para el arranque del proyecto se ha llevado a cabo la siguiente lista de tareas que serán cumplidas en el Sprint 0.

Reuniones de toma de requerimientos con el Dr. Marco Medina, estas citas se harán en paralelo junto con otras actividades que involucran la puesta a punto del sistema y del arranque del proyecto debido a la agenda que maneja el doctor como periodoncista y como representante latinoamericano en periodoncia.

4.3 Sprint 0

4.4 Requerimientos de MF Dental

Se plantearon los requerimientos necesarios para el desarrollo del proyecto en la empresa MF Dental.

Tabla 12. Requerimientos del sistema.

Requerimientos			
ID	Nombre	Usuario	Descripción
R1	Creación de usuarios para acceso al sistema.	Administrador	Usuarios que se registrarán roles como administrador, doctor o asistente.
R2	Control de acceso a usuarios del sistema de acuerdo a un rol.	Administrador	Los usuarios deben únicamente tener acceso a ciertas responsabilidades de acuerdo a su rol.
R3	Creación de perfiles de pacientes.	Administrador Doctor Asistente	El administrador creará los usuarios que podrán ingresar al sistema.
R4	Creación de historial clínico para pacientes.	Administrador Doctor Asistente	Todos los roles podrán crear el historial médico del paciente.
R5	Ingreso de nueva consulta médica.	Administrador Doctor	Se registran datos de la consulta.
R6	Generación de recetas.	Administrador Doctor	Se registra las indicaciones para cada receta.
R7	Visualización de historia clínica del paciente.	Administrador	Se verán los registros que el paciente ha tenido durante todo su ciclo en MF dental.
R8	Ingreso de medicamentos en recetas.	Doctor	
R9	Generación de reportes de historial médico de pacientes.	Administrador Doctor	Se registran los medicamentos para el tratamiento del paciente.
R10	Generación de reportes de	Administrador Doctor	Verificación de datos de los perfiles.

	perfiles de usuarios del sistema.		
R11	Actualización de datos de paciente.	Administrador	
R12	Actualización de datos de usuarios del sistema.	Administrador	
R13	Recomendaciones de tratamiento.	Administrador Doctor	
R14	Ingreso de notas de evolución en la consulta.	Administrador Doctor	Se ingresa las notas de evolución del paciente de acuerdo a su tratamiento.
R15	Ingreso de datos para margen gingival de cada pieza dental del paciente.	Administrador Doctor	Ingreso de datos tomados por el doctor de acuerdo a los niveles especificados de margen gingival.
R16	Ingreso de datos para sondaje inicial del paciente.	Administrador Doctor	Ingreso de datos tomados por el doctor de acuerdo a los niveles especificados de sondaje inicial.
R17	Ingreso de datos de sondaje para reevaluación del paciente.	Administrador Doctor	Ingreso de datos tomados por el doctor de acuerdo a los niveles especificados de sondaje de reevaluación.
R18	Ingreso de límite muco-gingival para cada pieza dental.	Administrador Doctor	
R19	Autogeneración de diagnóstico	Administrador	Reporte del diagnóstico.

	médico.		
R20	Guardar todos los registros que se realicen en el paciente durante todo si historial clínico.	Administrador Doctor	
R21	Personalización de las piezas dentales de acuerdo al paciente.	Administrador Doctor	Se personalizará las piezas dentales de acuerdo a la dentadura del paciente.
R22	Gráficos automáticos de las líneas de nivel de enfermedad periodontal.	Administrador Doctor	Generación de líneas indicadoras del nivel de enfermedad.
R23	Alineamiento del cuello de todas las piezas dentales en el gráfico.	Administrador Doctor	Todas las piezas dentales deben estar alineadas.
R24	Calculo automático de valores de enfermedad.	Administrador Doctor	
R25	Diagnóstico de la enfermedad en las piezas dentales.	Administrador Doctor	

Nota: Detalle de cada historia de usuario requerido por MF Dental.

4.4.1 Requisitos no funcionales

Existen otros requisitos que se han presentado con respecto al funcionamiento del software a desarrollar.

Tabla 13. Requisitos no funcionales del sistema

ID	Detalle
RNF1	Software a desarrollar estable.
RNF2	Navegación y uso intuitivo.
RNF3	Evitar colores opacos en las pantallas.
RNF4	Funcionamiento en varias plataformas de sistemas operativos Windows.

4.5 Historias de Usuario

Una vez que se registran los requerimientos que debe ofrecer el sistema, estos se convierten en historias de usuario.

Tabla 14. Historias de usuario.

Historias de usuario	
ID	Nombre
H1	Creación de usuarios para acceso al sistema.
H2	Control de acceso a usuarios del sistema de acuerdo a un rol.
H3	Creación de perfiles de pacientes.
H4	Creación de historial clínico para pacientes.
H5	Ingreso de nueva consulta médica.
H6	Generación de recetas.
H7	Visualización de historia clínica del paciente.
H8	Ingreso de medicamentos en recetas.
H9	Generación de reportes de historial médico de pacientes.

H10	Generación de reportes de perfiles de usuarios del sistema.
H11	Actualización de datos de paciente.
H12	Actualización de datos de usuarios del sistema.
H13	Recomendaciones de tratamiento.
H14	Ingreso de notas de evolución en la consulta.
H15	Ingreso de datos para margen gingival de cada pieza dental del paciente.
H16	Ingreso de datos para sondaje inicial del paciente.
H17	Ingreso de datos de sondaje para reevaluación del paciente.
H18	Ingreso de límite muco-gingival para cada pieza dental.
H19	Autogeneración de diagnóstico médico.
H20	Guardar todos los registros que se realicen en el paciente durante todo su historial clínico.
H21	Personalización de las piezas dentales de acuerdo al paciente.
H22	Gráficos automáticos de las líneas de nivel de enfermedad periodontal.
H23	Alineamiento del cuello de todas las piezas dentales en el gráfico.
H24	Calculo automático de valores de enfermedad.
H25	Diagnóstico de la enfermedad en las piezas dentales.

4.5.1 Actores del sistema

Las siguientes personas corresponden al grupo de personas que tienen relación con en el sistema.

Tabla 15. Detalle de actores que interactúan con el sistema.

ACT-ID	Administrador
Descripción	Encargado de administrar el control de usuarios y visualizar reportes de historiales médicos.
ACT-ID	Doctor
Descripción	Encargado de crear consultas, recetar medicinas e ingresar los datos de los sondajes periodontales.
ACT-ID	Asistente
Descripción	Este actor se encarga de crear los perfiles de clientes.
ACT-ID	Paciente
Descripción	No es un actor que interactúe directamente con el sistema pero es el principal involucrado debido a que sus datos son almacenados en el mismo y además es la razón para que el sistema se desarrolle.

4.5.2 Priorización de historias de usuario

Los requisitos del sistema ahora se tratarán como historias de usuarios para cumplir con la metodología Scrum, se utilizando la letra H para identificar cada historia. Se ha determinado con el cliente el orden de prioridad de cada historia de usuario.

Tabla 16. Orden de prioridad de historias de usuario.

Historias de usuario		
ID	Nombre	Prioridad
H1	Creación de usuarios para acceso al sistema.	1
H2	Control de acceso a usuarios del sistema de acuerdo a un rol.	1
H3	Creación de perfiles de pacientes.	1
H4	Creación de historial clínico para pacientes.	1
H5	Ingreso de nueva consulta médica.	1
H6	Generación de recetas.	3
H7	Visualización de historia clínica del paciente.	3
H8	Ingreso de medicamentos en recetas.	4
H9	Generación de reportes de historial médico de pacientes.	4
H10	Generación de reportes de perfiles de usuarios del sistema.	4
H11	Actualización de datos de paciente.	1
H12	Actualización de datos de usuarios del sistema.	2
H13	Recomendaciones de tratamiento.	3
H14	Ingreso de notas de evolución en la consulta.	3
H15	Ingreso de datos para margen gingival de cada pieza dental del paciente.	1
H16	Ingreso de datos para sondaje inicial del paciente.	1
H17	Ingreso de datos de sondaje para reevaluación del paciente.	1
H18	Ingreso de límite muco-gingival para cada pieza dental.	1
H19	Autogeneración de diagnóstico médico.	3
H20	Guardar todos los registros que se realicen en el paciente durante todo su historial clínico.	2
H21	Personalización de las piezas dentales de acuerdo al paciente.	2

H22	Gráficos automáticos de las líneas de nivel de enfermedad periodontal.	1
H23	Alineamiento del cuello de todas las piezas dentales en el gráfico.	2
H24	Calculo automático de valores de enfermedad.	1
H25	Diagnóstico de la enfermedad en las piezas dentales.	1

Nota: El orden de prioridad se determina en un rango del número 1 al 5 siendo 1 la prioridad más elevada y 5 la que menos impacto tendría en el sistema.

4.6 Product Backlog del Sistema

Una vez priorizadas las historias de usuario se ha dividido la lista de objetivos en 4 sprints, al final del desarrollo se contará con 5 sprints debido a que para el arranque del proyecto se ha definido el sprint 0.

Tabla 17. Relación de cada sprint con las historias de usuario generadas por el cliente.

Historias de usuario		
Sprint	ID	Nombre
0	H0	Arranque del proyecto
1	H1	Creación de usuarios para acceso al sistema.
	H2	Control de acceso a usuarios del sistema de acuerdo a un rol.
	H3	Creación de perfiles de pacientes.
	H4	Creación de historial clínico para pacientes.
	H5	Ingreso de nueva consulta médica.
	H12	Actualización de datos de usuarios del sistema.
2	H11	Actualización de datos de paciente.
	H20	Guardar todos los registros que se realicen en el paciente durante todo su historial clínico.
	H7	Visualización de historia clínica del paciente.
	H14	Ingreso de notas de evolución en la consulta.
	H8	Ingreso de medicamentos en recetas.

3	H21	Personalización de las piezas dentales de acuerdo al paciente.
	H23	Alineamiento del cuello de todas las piezas dentales en el gráfico.
	H6	Generación de recetas.
	H13	Recomendaciones de tratamiento.
	H19	Autogeneración de diagnóstico médico.
4	H15	Ingreso de datos para margen gingival de cada pieza dental del paciente.
	H16	Ingreso de datos para sondaje inicial del paciente.
	H17	Ingreso de datos de sondaje para reevaluación del paciente.
	H18	Ingreso de límite muco-gingival para cada pieza dental.
	H22	Gráficos automáticos de las líneas de nivel de enfermedad periodontal.
	H24	Calculo automático de valores de enfermedad.
	H25	Diagnóstico de la enfermedad en las piezas dentales.
	H9	Generación de reportes de historial médico de pacientes.
H10	Generación de reportes de perfiles de usuarios del sistema.	

4.7 Duración de cada sprint

Se ha decidido que cada sprint va a tener una duración diferente, debido a que las tareas seleccionadas para cada sprint cumplen diferentes funciones.

- Sprint 0 tendrá una duración de 20 días
- Sprint 1 tendrá una duración de 30 días
- Sprint 2 tendrá una duración de 30 días
- Sprint 3 tendrá una duración de 20 días
- Sprint 4 tendrá una duración de 25 días

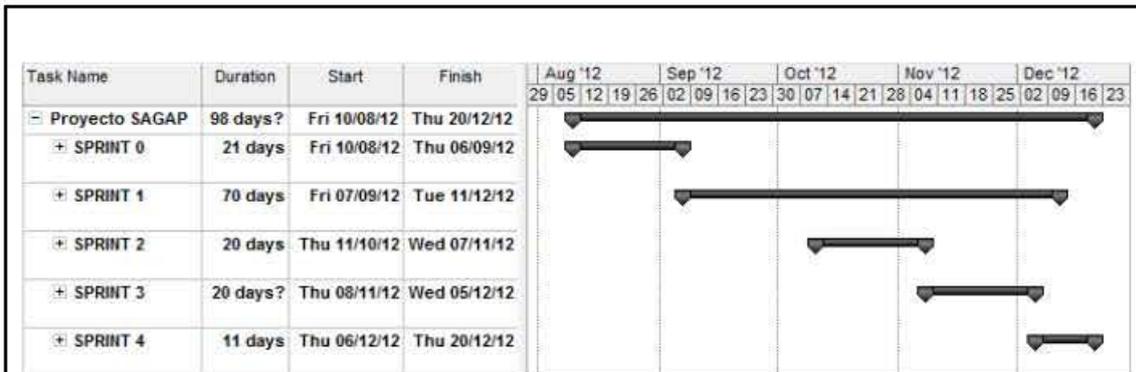


Figura 21. Diagrama de Gantt para el desarrollo del proyecto SAGAP.

4.7.1 Tareas de Sprint 0

Para el arranque del proyecto se ha tomado un plazo de 10 días, durante el sprint 0 se realizarán las siguientes tareas. Existen tareas dentro del sprint 0 en las que únicamente se refiere al cliente y sus necesidades.

- Reuniones con el cliente para determinar los requisitos del sistema.
- Ubicación de las necesidades de la empresa.
- Verificar el funcionamiento del negocio.
- Determinar posibles riesgos que puedan ocasionar el fracaso del proyecto.
- Elaboración del plan de proyecto.
- Reunir documentación de MF Dental necesaria para el desarrollo.
- Especificación de procedimientos específicos de los cálculos matemáticos en periodontitis.

También existen tareas con respecto al equipo de trabajo en el que se definen los siguientes puntos.

- Instalación de herramientas para el desarrollo.
- Prueba de funcionamiento de las herramientas.
- Determinación de historias de usuario.
- Creación del Product Backlog.
- Generación de sprints necesarios para cumplir el proyecto.
- Especificación del nombre del software.
- Establecer parámetros para determinar la calidad del desarrollo.

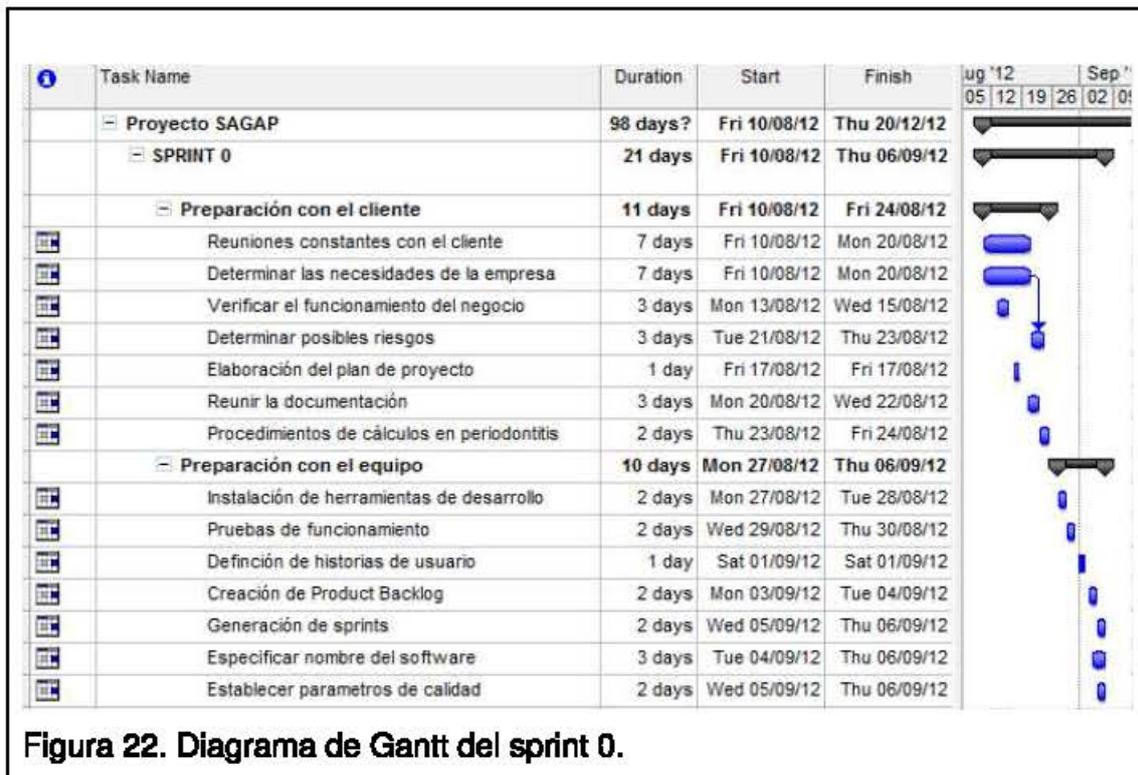


Figura 22. Diagrama de Gantt del sprint 0.

4.7.2 Tareas de Sprint 1

Las tareas asignadas al sprint 1 son:

- Diagrama DFD Nivel 1.
- Diagrama DFD Nivel 2.
- Diseño conceptual de base de datos.
- Creación del modelo físico de base de datos.
- Creación de usuarios para acceso al sistema.
- Diagramas de procesos.
- Diagrama de Clases.
- Generación de Script de Base de datos.
- Cargar base de datos a MySQL.
- Crear nueva aplicación Yii Sagap.
- Generación de todos los modelos para cada tabla de la base de datos.
- Generar formulario de registro de usuario.

- Generar formulario de creación de usuario.
- Validar el login de un nuevo usuario en el sistema.
- Generación de opciones para actualización de usuario.
- Validar actualización de usuarios del sistema.
- Crear perfiles de usuario.
- Control de acceso de acuerdo a perfil de usuario.
- Creación de formulario para creación de perfiles de pacientes.
- Validación de ingreso de datos para creación de perfiles.
- Creación de formulario para historia clínica.
- Validación de información para ingreso de nueva historia.
- Creación de formulario para consulta médica.
- Validación de datos para ingreso en el formulario de consulta.

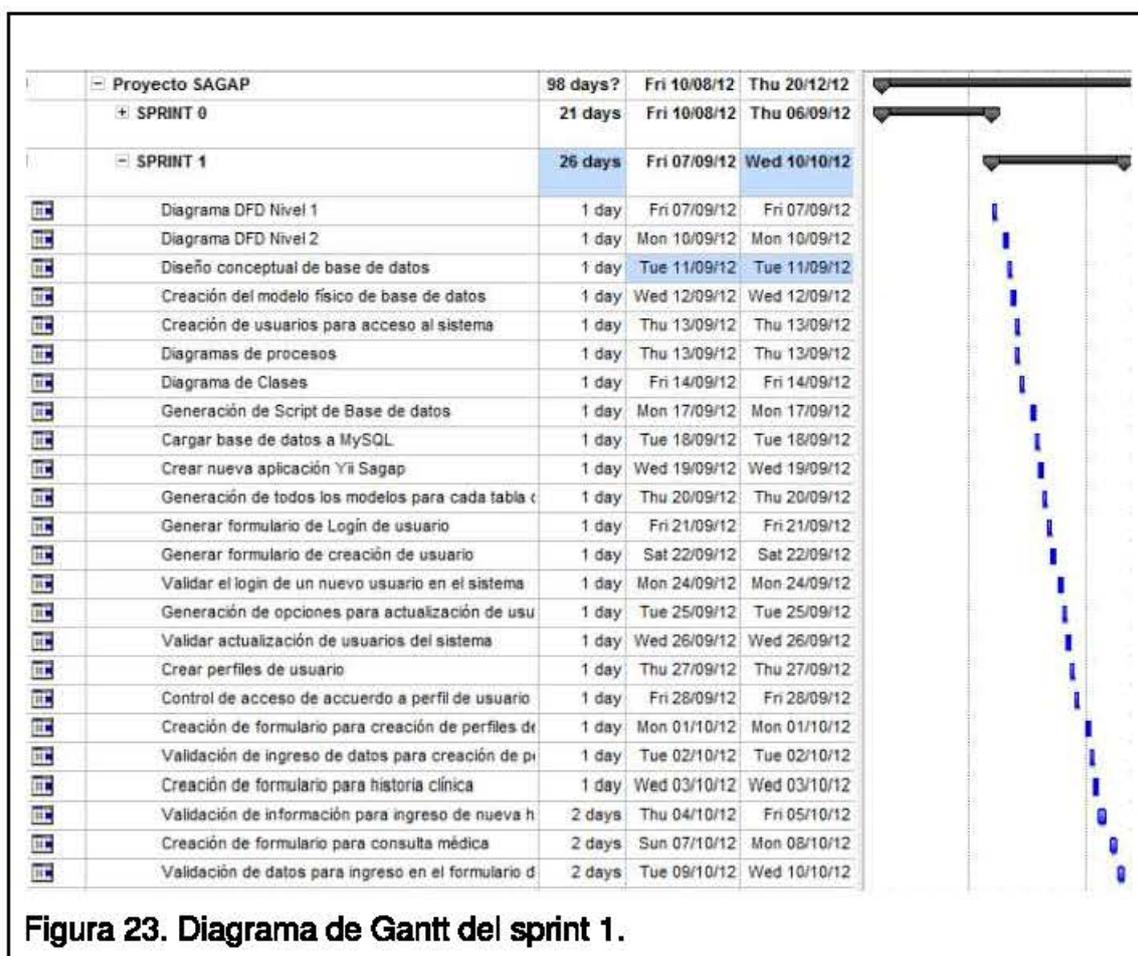


Figura 23. Diagrama de Gantt del sprint 1.

4.7.3 Tareas de Sprint 2

Las tareas asignadas al sprint 2 son:

- Generación de todos los controladores de la aplicación.
- Generación de formularios de salida.
- Creación de menús.
- Widgets para todos los calendarios.
- Permisos de acceso.
- Apariencia del sistema.
- Creación de gráfico principal con logo de la empresa.
- Creación de formulario de tratamiento.
- Creación de formulario de medicamentos.
- Validación de ingreso de datos para formulario de tratamiento.
- Validación de ingreso de datos para formulario de medicamentos.
- Vincular formularios de medicamentos a plan de tratamiento.
- Vincular formularios de tratamiento a consultas.
- "Mostrar en el historial clínico la información de pacientes, consultas, tratamientos y medicamentos".
- Validar la actualización de datos del paciente.
- Creación de formulario de notas de evolución.
- Validar ingreso de datos para formulario de notas de evolución.
- Vincular datos de notas de evolución al historial clínico.

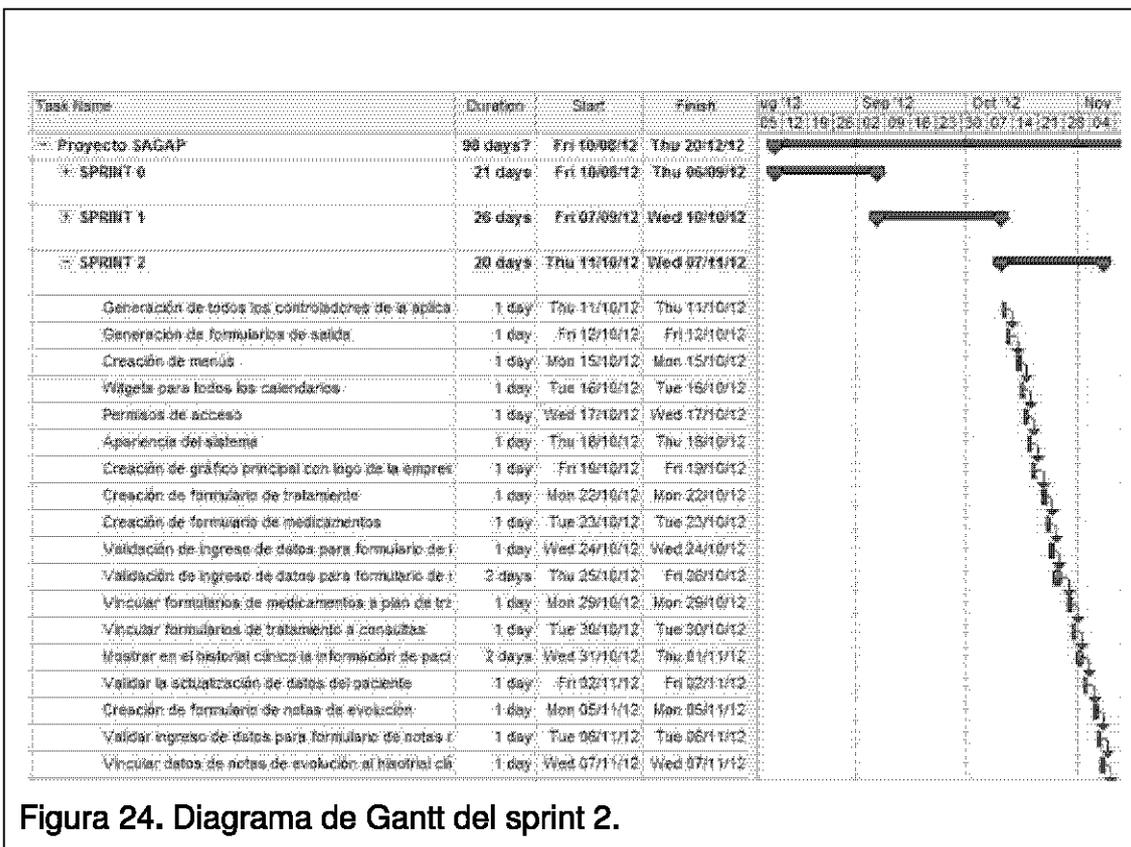


Figura 24. Diagrama de Gantt del sprint 2.

4.7.4 Tareas de Sprint 3

Las tareas asignadas al sprint 3 son:

- Creación de formulario de odontograma.
- Creación de gráfico de cada pieza dental.
- Mostrar piezas dentales en formulario de odontograma.
- Creación de botones con diferentes tipos de registro para cada pieza dental.
- Asociar un estado a una pieza dental.
- Guardar el resultado en la base de datos.
- Validar el ingreso de datos.
- Asociar el formulario de odontograma al menú de opciones.
- Superposición de paneles usando HTML5.

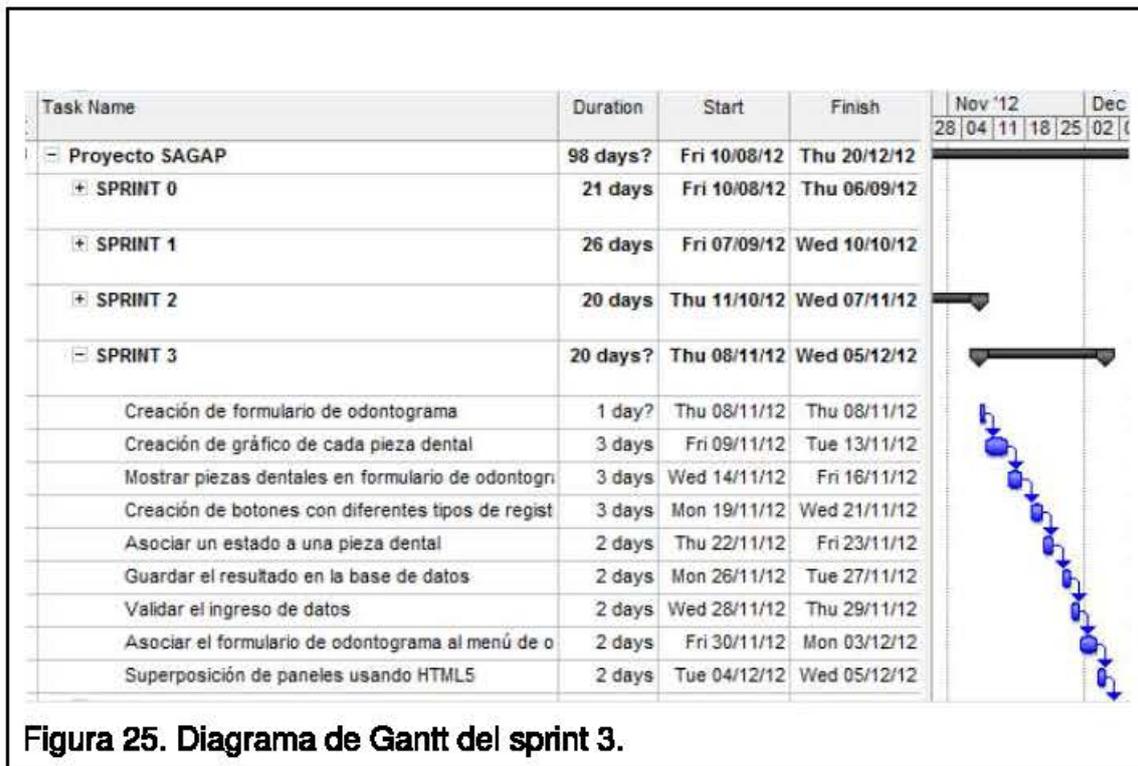


Figura 25. Diagrama de Gantt del sprint 3.

4.7.5 Tareas de Sprint 4

Las tareas asignadas al sprint 4 son:

- Creación de formulario para periodontograma.
- Asociar imagen de odontograma en periodontograma.
- Validación de ingreso de información en el formulario.
- Superposición de panel con líneas guías de nivel.
- Creación e variables para las líneas de nivel de enfermedad.
- Asociación de datos con variables.
- Cálculo de niveles de enfermedad.
- Superposición de panel para gráfico de líneas.
- Graficación de líneas HTML5.
- Creación de variables para diagnóstico.
- Cálculo de diagnóstico por pieza dental.
- Cálculo de diagnóstico total.

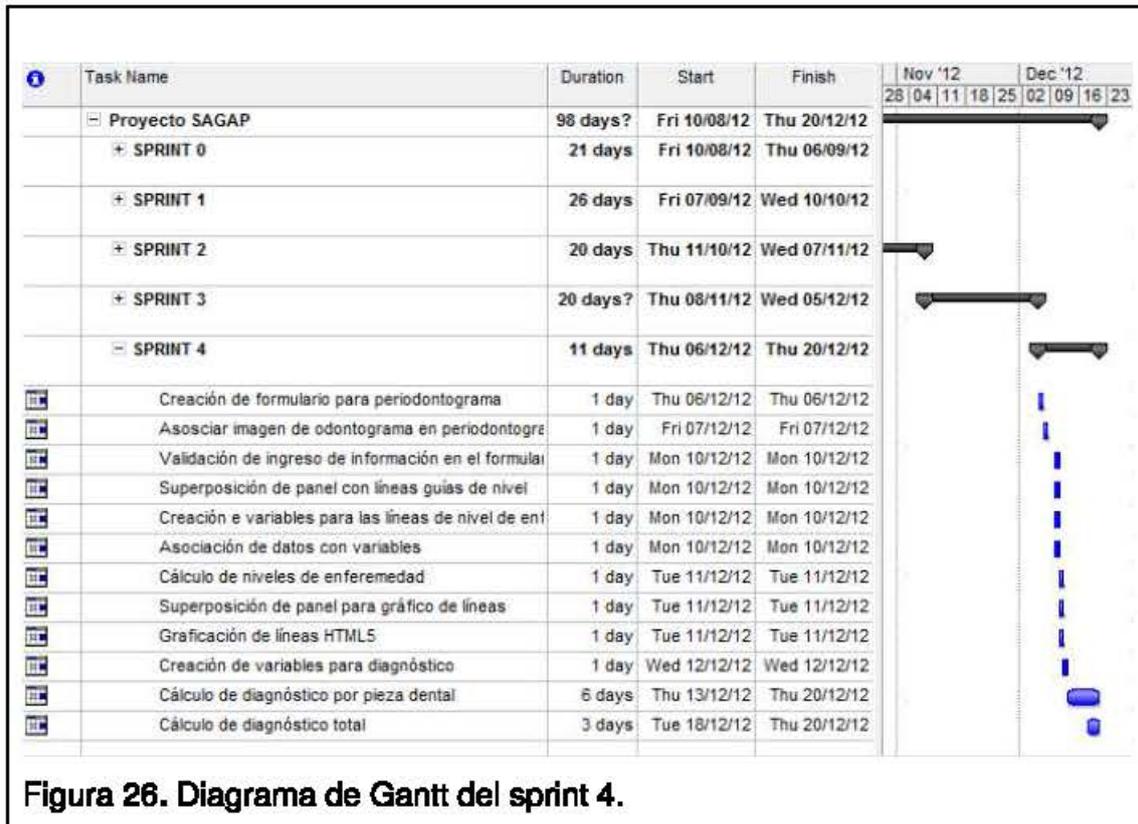


Figura 26. Diagrama de Gantt del sprint 4.

4.8 Parámetros de calidad

Para el desarrollo del producto se han contemplado los siguientes parámetros de calidad que se llevarán a cabo durante cada sprint hasta obtener el producto terminado:

- Pruebas modulares: Pruebas aisladas de cada sprint.
- Pruebas de sistema: Para asegurar el correcto funcionamiento en el entorno del cliente.
- Pruebas de aceptación: Demostración al cliente de la porción de software entregado.

Con esto el equipo asegura la calidad del software entregado al cliente, de hecho el uso de Yii framework asegura la calidad en la construcción de software.

4.9 Nombre del Sistema

El nombre escogido para el sistema periodontal fue SAGAP que es el acrónimo de Sistema de Apoyo Gráfico para Atención Periodontal.

4.10 Instalación de Aplicaciones de Desarrollo

Para emprender el proyecto SAGAP es necesario poner a punto el software de desarrollo a utilizar. (Para mayor detalle verificar el manual de instalación de las aplicaciones que se encuentra en los anexos).

4.10.1 Sistema Operativo

El proyecto se desarrollará en un equipo laptop de las siguientes características.

- Procesador Intel Core 2 Duo T7100 1.8GH.
- RAM 2 GB.
- Sistema Windows 7 Home Premium de 32 bits.
- Disco Duro de 200GB.

4.10.2 Aplicaciones de Desarrollo

Las siguientes aplicaciones se utilizaron para la creación del proyecto SAGAP, se descargaron los instaladores desde el sitio web respectivo en versión compatible con Windows 7 de 32 bits.

- XAMPP Server 2.5 (xampp-win32-1.7.4-VC6-installer.exe).
- MySQL.
- PHP.
- Apache Web Server.
- Yii Framework (yii-1.1.12.b600af.zip).
- Eclipse Helios (eclipse-php-helios-SR2-win32.zip).
- PowerDesigner 12 (PowerDesigner12.5.rar).
- Notepad++ (npp.6.2.2.Installer.exe).

NOMBRE TABLA	estado_civil
DESCRIPCIÓN	Guarda la información de de estado civil de cada paciente

NOMBRE TABLA	factor
DESCRIPCIÓN	Información del factor RH sanguíneo que posee cada paciente.

NOMBRE TABLA	grupo_sanguineo
DESCRIPCIÓN	Información referente al tipo de sangre de cada paciente.

NOMBRE TABLA	documento
DESCRIPCIÓN	Información si el documento que posee cada paciente es una cédula o un pasaporte en el caso de extranjeros.

NOMBRE TABLA	sexo
DESCRIPCIÓN	Identificación si el paciente tiene sexo masculino o femenino.

NOMBRE TABLA	historia_clinica
DESCRIPCIÓN	Información con los datos del paciente y que hace referencia a otra información con el objetivo de guardar todos los eventos registrados por el paciente mientras sea atendido en MF Dental.

NOMBRE TABLA	consulta
DESCRIPCIÓN	Información referente a cada cita realizada por el paciente en MF Dental.

NOMBRE TABLA	notas_evolution
DESCRIPCIÓN	Almacena registros que realiza el doctor de acuerdo a la evolución del paciente y está asociada a una consulta.

NOMBRE TABLA	diagnostico
DESCRIPCIÓN	Indica una descripción de los resultados de la consulta que realiza el periodoncista.

NOMBRE TABLA	plan_tratamiento
DESCRIPCIÓN	Descripción de cómo se va a proceder de acuerdo a la consulta.

NOMBRE TABLA	medicamento
DESCRIPCIÓN	Indica una descripción completa de cada medicamento que el paciente va a consumir durante un plan de tratamiento.

NOMBRE TABLA	establecimiento
DESCRIPCIÓN	Identifica el nombre de la institución a la que pertenece la consulta, es una referencia.

NOMBRE TABLA	odontograma
DESCRIPCIÓN	Indica el estado en el que se encuentra cada pieza dental de un paciente.

NOMBRE TABLA	estado_pieza
DESCRIPCIÓN	Representa los diferentes tipos de condición que puede tener una pieza dental.

NOMBRE TABLA	registro_periodontograma
DESCRIPCIÓN	Información sobre los registros de niveles de enfermedad que el periodoncista realiza al paciente con sondas.

NOMBRE TABLA	tipo_registro
DESCRIPCIÓN	Indica si el patrón de registro para cada medición total de niveles de periodontitis.

NOMBRE TABLA	usuario
DESCRIPCIÓN	Información de qué personas que pueden hacer uso del sistema.

NOMBRE TABLA	perfil
DESCRIPCIÓN	Información del nivel de acceso de cada usuario.

NOMBRE TABLA	antecedentes_personales
DESCRIPCIÓN	Información en el historial personal acerca de enfermedades delicadas.

NOMBRE TABLA	periodontograma
DESCRIPCIÓN	Información proveniente de los registros periodontales que se realicen en cada paciente, además se vincula a la información de diagnóstico total de la enfermedad y diagnóstico por cada pieza dental

NOMBRE TABLA	diagnostico_pieza
DESCRIPCIÓN	Información de diagnóstico si la enfermedad después de realizar cálculos matemáticos es severa, leve y moderada únicamente por cada pieza dental.

NOMBRE TABLA	diagnostico_total
DESCRIPCIÓN	Información si la enfermedad registrada en cada paciente es severa, moderada o leve.

4.11.4 Generación de Script para Creación de Base de Datos

Una vez obtenido el modelo físico de datos se utiliza la herramienta PowerDesigner para la generación del script de creación de la base de datos para MySQL en un archivo SagapDB.sql.

4.11.5 Carga de Base de Datos

Para la carga y creación de la base de datos es necesario seguir los siguientes pasos:

- Crear una nueva base de datos llamada SAGAP.
- Copiar el contenido del archivo SagapDB.sql.
- Pegar las líneas de código SQL en la herramienta para ejecutar scripts de PhpMyAdmin.
- Ejecutar el script.
- Al final debe aparecer un mensaje indicando que la consulta se ejecutó con éxito.



Figura 29. Visualización de carga exitosa del scrip de base de datos.

4.12 Diagrama DFD Nivel 0

El sistema trabajará de acuerdo al tipo de usuario, el mismo que tendrá un rol establecido y de acuerdo a eso el sistema gestionará la información necesaria, las tres personas que interactúan con el sistema son:

- **Administrador**
- **Doctor**
- **Asistente**

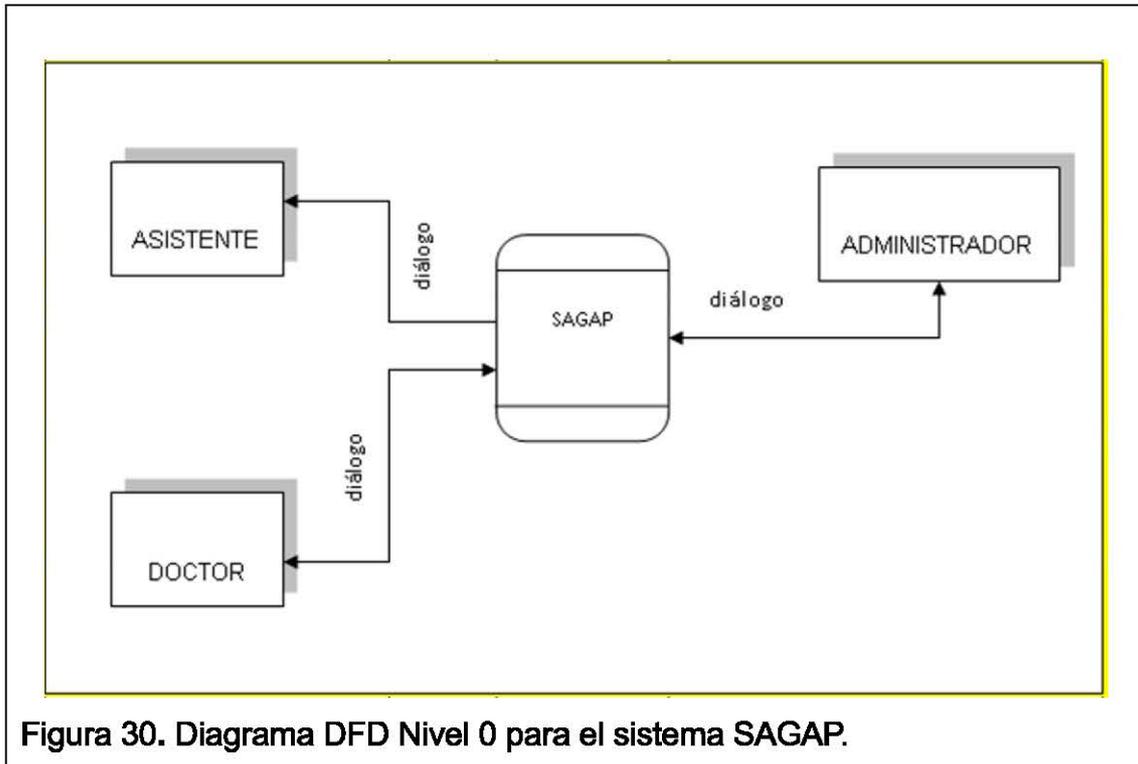
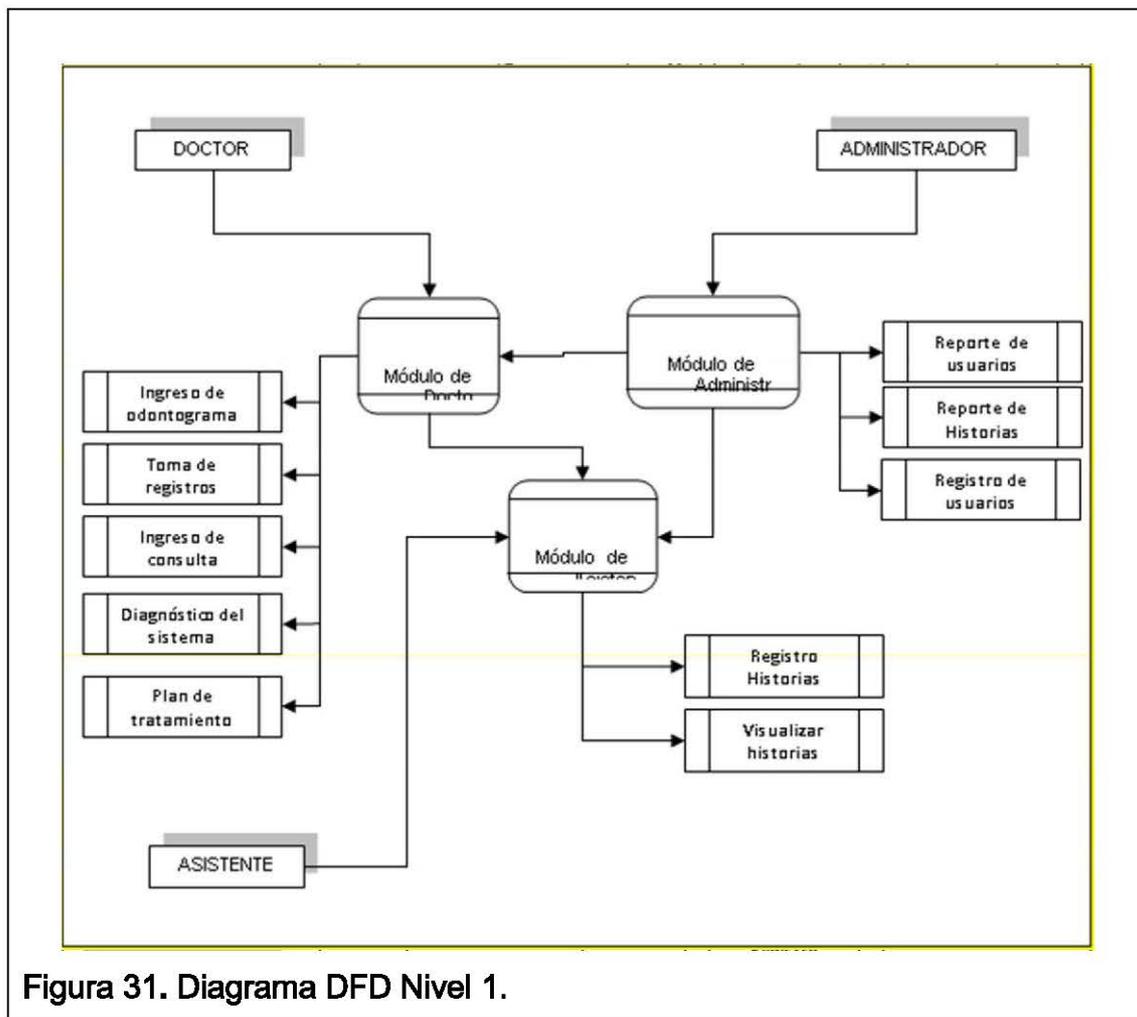


Figura 30. Diagrama DFD Nivel 0 para el sistema SAGAP.

4.12.1 Diagrama DFD Nivel 1

En el diagrama Nivel 1 se identifica los subsistemas que componen SIGSAP, los accesos que tienen los diferentes roles de usuarios a sus respectivos módulos.



4.12.2 Módulos Identificados

El diagrama DFD Nivel 1 identificado en la Figura 31 nos permite apreciar los diferentes subsistemas generados.

- Módulo Asistente
- Módulo Doctor
- Módulo Administrador

4.13 Análisis de Procesos

Para cada módulo existen diferentes procesos que van a ser accedidos de acuerdo a los diferentes roles que ocupen los usuarios, en algunos casos los usuarios van a poder entrar a procesos de diferentes módulos.

4.13.1 Login de Usuario

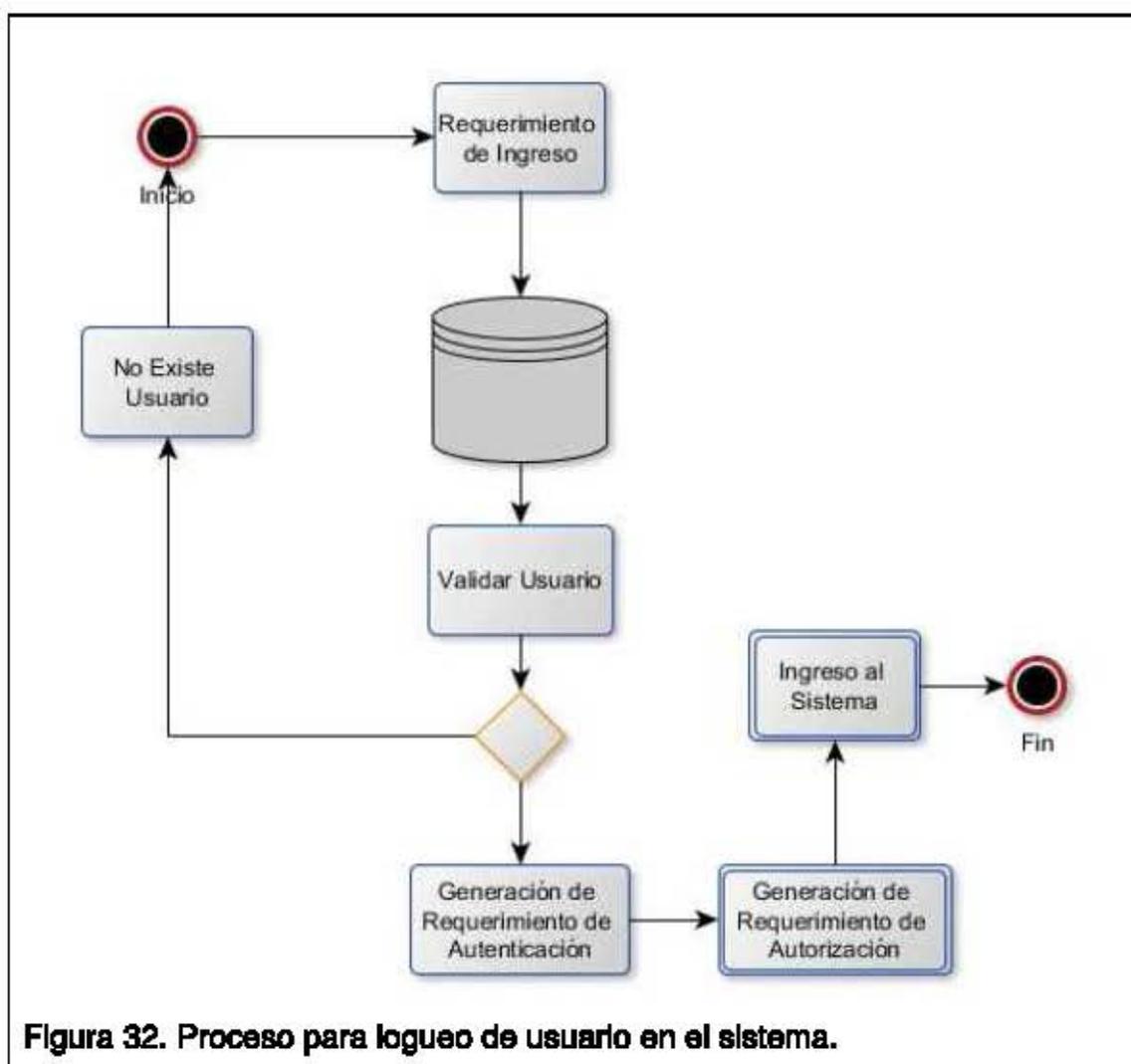


Figura 32. Proceso para logueo de usuario en el sistema.

4.13.2 Gestión de Usuario del Sistema

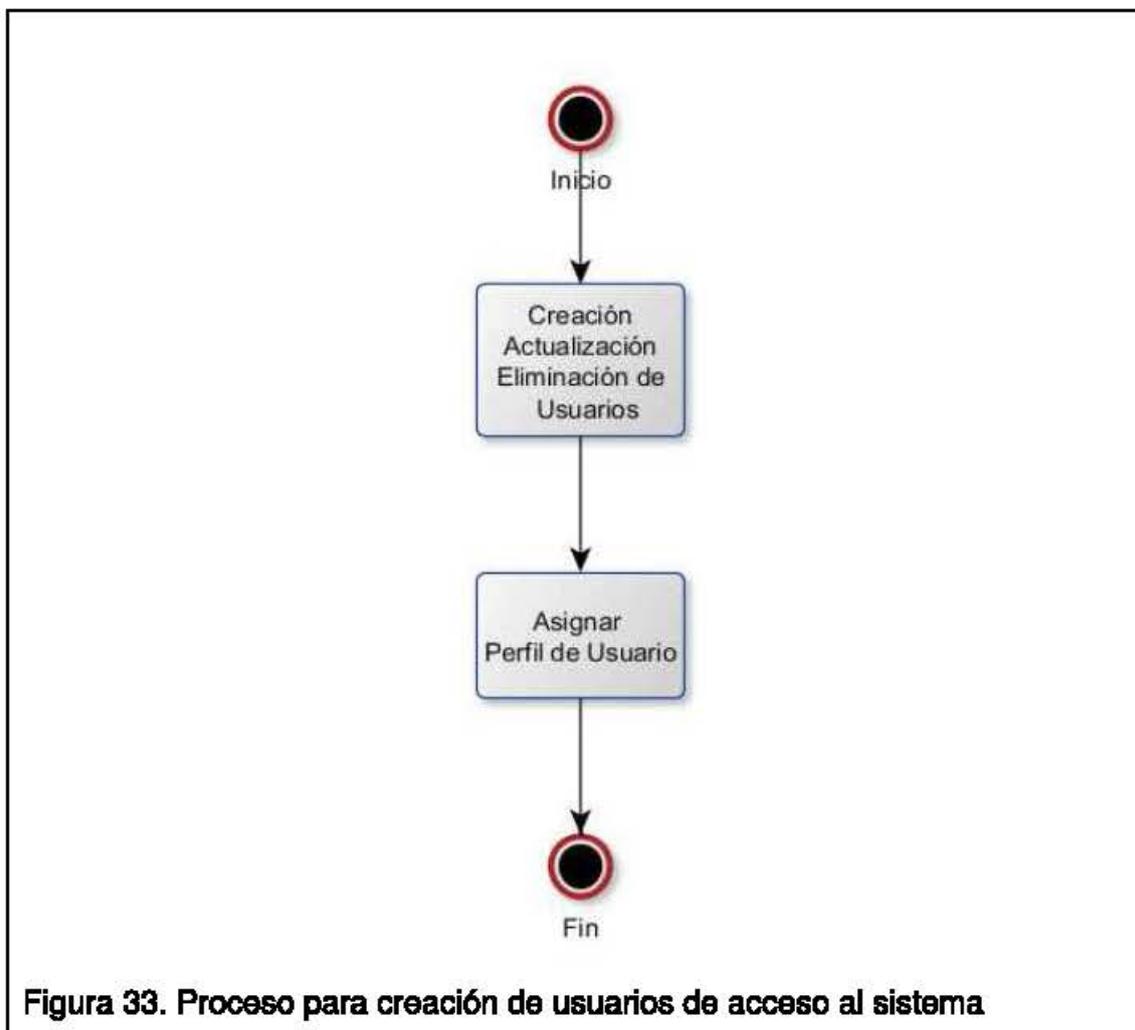


Figura 33. Proceso para creación de usuarios de acceso al sistema

4.13.3 Creación Actualización y Eliminación de Usuarios

Descripción

El administrador del sistema podrá hacer uso del mismo para crear, actualizar o eliminar usuarios del sistema. Los campos que se desplegarán son:

- Perfil
- Nombre
- Apellido
- Número de documento
- Username
- Password

4.13.4 Asignar Perfil de Usuario

Descripción

La asignación del perfil de usuario consiste en identificar el tipo de acceso que el usuario va a tener en el sistema, el mismo que puede ser Administrador, Doctor y Asistente.

4.13.5 Registro de Paciente

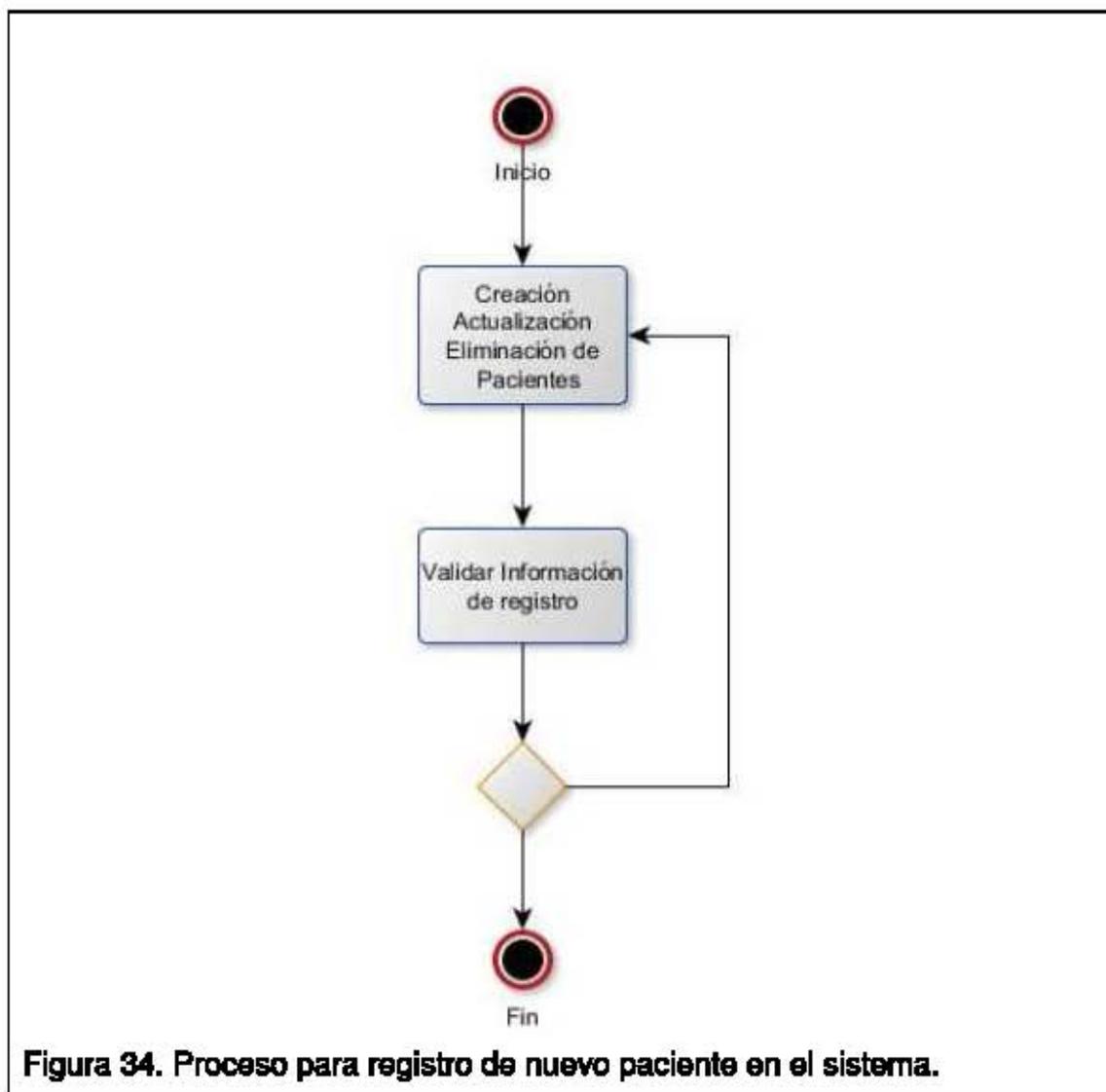


Figura 34. Proceso para registro de nuevo paciente en el sistema.

4.13.6 Creación, Eliminación y Actualización de Pacientes

Descripción

Se registrará la información inicial cuando un paciente ingrese a MF Dental, esto podrá ser controlado por el Administrador, Doctor y Asistente, los datos que se desplegarán para el registro son:

- Nombre del Paciente
- Apellido Materno
- Apellido Paterno
- Sexo
- Grupo Sanguíneo
- Factor Sanguíneo
- Tipo de Documento
- Detalle del Documento
- Nacionalidad
- Ciudad de Nacimiento
- Provincia
- Dirección
- Ocupación
- Email
- Foto
- Estado Civil
- Teléfono

4.13.7 Validar Información de Registro

Descripción

El sistema valida si los datos ingresados por el usuario son correctos caso contrario el usuario debe ingresar la información hasta que los sea perfecta.

4.13.8 Registro de Historial Clínico

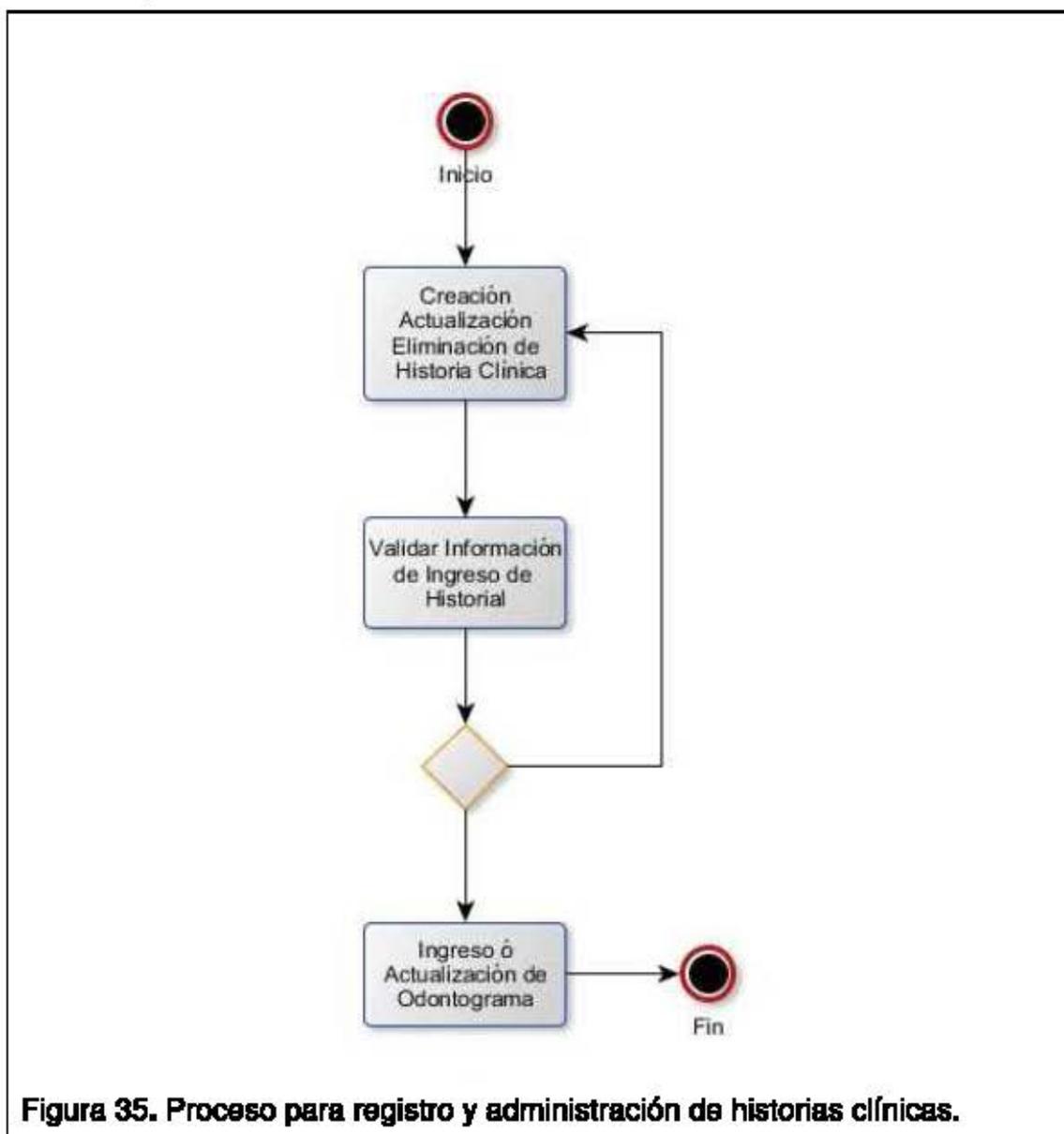


Figura 35. Proceso para registro y administración de historias clínicas.

4.13.9 Creación, Actualización o Eliminación de Historia Clínica

Descripción

El usuario gestiona la Información que se Ingresará como historia clínica del paciente, esta información quedará permanentemente o de ser el caso podrá ser actualizada conforme pase el tiempo o eliminada si el cliente ya no se atiende más en MF Dental.

- La información que el sistema despliega es:

- Antecedentes Personales
- Fecha de creación
- Establecimiento
- Periodontograma
- Odontograma

4.13.10 Validar Información de Registro de Historial

Descripción

El sistema verifica que la información ingresada contenga los patrones correctos de acuerdo al tipo de dato del campo caso contrario el usuario debe seguir ingresando los datos hasta que la validación sea perfecta.

4.13.11 Ingreso o Actualización de Odontograma

Descripción

El usuario ingresa o actualiza la información del estado de cada pieza dental del paciente, los campos que se despliegan serán los siguientes:

- Estado de cada una de las 38 piezas dentales

(Detalle del resto de procesos se incluye en los anexos).

4.14.1 Descripción del Diagrama de Clases

Tabla 19. Clases del sistema.

NOMBRE CLASE	PACIENTE
TIPO CLASE	PADRE
DESCRIPCIÓN	Registra la información general del paciente que servirá para su historial clínico.
CLASES HIJAS	<ul style="list-style-type: none"> - Documento - Grupo Sanguíneo - Factor - Sexo - Estado Civil - Teléfono

NOMBRE CLASE	SEXO
TIPO CLASE	HIJA
DESCRIPCIÓN	Registra el sexo que tiene cada paciente.
CLASES HIJAS	-

NOMBRE CLASE	DOCUMENTO
TIPO CLASE	HIJA
DESCRIPCIÓN	Registra si el tipo de documento del paciente es una cédula o un pasaporte.
CLASES HIJAS	-

NOMBRE CLASE	GRUPO SANGUÍNEO
TIPO CLASE	HIJA
DESCRIPCIÓN	Registra el tipo de sangre que tiene el paciente.
CLASES HIJAS	-

NOMBRE CLASE	FACTOR
TIPO CLASE	HIJA
DESCRIPCIÓN	Registra si el factor sanguíneo del paciente es positivo o negativo.
CLASES HIJAS	-

NOMBRE CLASE	ESTADO CIVIL
TIPO CLASE	HIJA
DESCRIPCIÓN	Registra información si el paciente se encuentra con estado civil soltero o casado.
CLASES HIJAS	-

NOMBRE CLASE	TELÉFONO
TIPO CLASE	HIJA
DESCRIPCIÓN	Registra los números de contacto de cada paciente.
CLASES HIJAS	-

NOMBRE CLASE	HISTORIA CLINICA
TIPO CLASE	PADRE
DESCRIPCIÓN	Registra toda la información y registro de actividades que el paciente realiza en MF Dental desde que ingresa, contiene la información de odontogramas, perfil, periodontogramas, consultas y tratamientos.
CLASES HIJAS	<ul style="list-style-type: none"> - Establecimiento - Paciente - Consulta - Antecedentes Personales - Periodontograma - Odontograma

NOMBRE CLASE	ANTECEDENTES PERSONALES
TIPO CLASE	HIJA
DESCRIPCIÓN	Registra el tipo de enfermedad que ha tendido previamente el paciente o sus familiares.
CLASES HIJAS	-

NOMBRE CLASE	ESTABLECIMIENTO
TIPO CLASE	HIJA
DESCRIPCIÓN	Registra la cede en donde se trata el paciente.
CLASES HIJAS	-

NOMBRE CLASE	ODONTOGRAMA
TIPO CLASE	HIJA
DESCRIPCIÓN	Registra la información del estado de cada pieza dental que presenta el paciente.
CLASES HIJAS	- Estado Pieza

NOMBRE CLASE	CONSULTA
TIPO CLASE	PADRE
DESCRIPCIÓN	Registra cada evento en el que el paciente acude a solicitar los servicios de MF Dental.
CLASES HIJAS	- Notas Evolución - Diagnóstico - Plan Tratamiento - Usuario

NOMBRE CLASE	NOTAS EVOLUCIÓN
TIPO CLASE	HIJA
DESCRIPCIÓN	Representa los registros de mejoría de cada paciente con respecto a su enfermedad periodontal.
CLASES HIJAS	-

NOMBRE CLASE	DIAGNÓSTICO
TIPO CLASE	HIJA
DESCRIPCIÓN	Representa el diagnóstico final que el periodoncista registra en base a los diagnósticos generados por el sistema.
CLASES HIJAS	-

NOMBRE CLASE	PLAN TRATAMIENTO
TIPO CLASE	PADRE
DESCRIPCIÓN	Registra el tipo de tratamiento que va a ser suministrado al paciente.
CLASES HIJAS	- Medicamento

NOMBRE CLASE	MEDICAMENTO
TIPO CLASE	HIJA
DESCRIPCIÓN	Registra la información detallada de cada medicamento que va a ser utilizado por el paciente.
CLASES HIJAS	-

NOMBRE CLASE	USUARIO
TIPO CLASE	PADRE
DESCRIPCIÓN	Registra los usuarios de MF Dental que van a ingresar al sistema.
CLASES HIJAS	- Perfil

NOMBRE CLASE	PERFIL
TIPO CLASE	HIJA
DESCRIPCIÓN	Registra la información del tipo de acceso que va a ser asignado a cada usuario del sistema.
CLASES HIJAS	-

NOMBRE CLASE	PERIODONTOGRAMA
TIPO CLASE	PADRE
DESCRIPCIÓN	Registra la información de todos los sondeos que se realiza a cada paciente, los datos contenidos sirven para realizar los cálculos de las líneas por donde se marcará el nivel de enfermedad periodontal.
CLASES HIJAS	- Diagnóstico Pieza - Diagnóstico Total - Registro Periodontograma

NOMBRE CLASE	DIAGNÓSTICO PIEZA
TIPO CLASE	HIJA
DESCRIPCIÓN	Registra el nivel de enfermedad que se obtuvo en base a los registros periodontales.
CLASES HIJAS	-

NOMBRE CLASE	DIAGNÓSTICO TOTAL
TIPO CLASE	HIJA
DESCRIPCIÓN	Registra el diagnóstico final de todo el nivel de enfermedad de cada paciente en base al número de piezas dentales afectadas.
CLASES HIJAS	-

NOMBRE CLASE	REGISTRO PERIODONTOGRAMA
TIPO CLASE	PADRE
DESCRIPCIÓN	Registra la información de sondajes iniciales, finales, enfermedad en las tres zonas de cada pieza dental.
CLASES HIJAS	TIPO REGISTRO

NOMBRE CLASE	TIPO REGISTRO
TIPO CLASE	HIJA
DESCRIPCIÓN	Detalla el tipo de registro de periodontograma que el doctor va a realizar.
CLASES HIJAS	-

4.15 Creación del proyecto SAGAP

Para la creación de un nuevo proyecto en Yii Framework se utiliza la línea de comandos CMD de la plataforma Windows y ejecutando la siguiente sentencia.

- `C:\xampp\htdocs\yii webapp sagap`

Se ha creado un nuevo proyecto Yii llamado SAGAP en la ruta `C:\xampp\htdocs`. Para su configuración es necesario seguir los siguientes pasos

- Copiar el contenido del directorio framework desde `C:\xampp\htdocs\yii` hasta la ruta `C:\xampp\htdocs\sagap`.
- Crear un nuevo proyecto PHP en Eclipse, con el mismo nombre que se creó la aplicación y se apunta hacia la ruta `C:\xampp\htdocs\sagap`.



Figura 37. Asociación de Eclipse con el proyecto generado en Yii.

4.15.1 Generación de Modelos del Sistema

Se procede a crear los modelos que sirven de enlace entre el sistema y los datos de cada tabla en la base de datos. (El detalle de los modelos generados se encuentra en los anexos).

4.16 Prototipos de Interfaces

El prototipo para las interfaces del sistema SAGAP tiene un formato estándar para todos sus formularios web, con el fin de dar facilidad de manejo y rápida adaptación de la funcionalidad del sistema al usuario.

El diseño elemental de pantallas se desarrolla de la siguiente manera:

Tabla 20. Bienvenida al sistema

PANTALLA	Bienvenida al sistema
DESCRIPCIÓN	Prototipo por donde el sistema le ofrece la puerta de entrada al sistema, dotándolo de menú para ingreso al sistema con una cuenta de usuario.

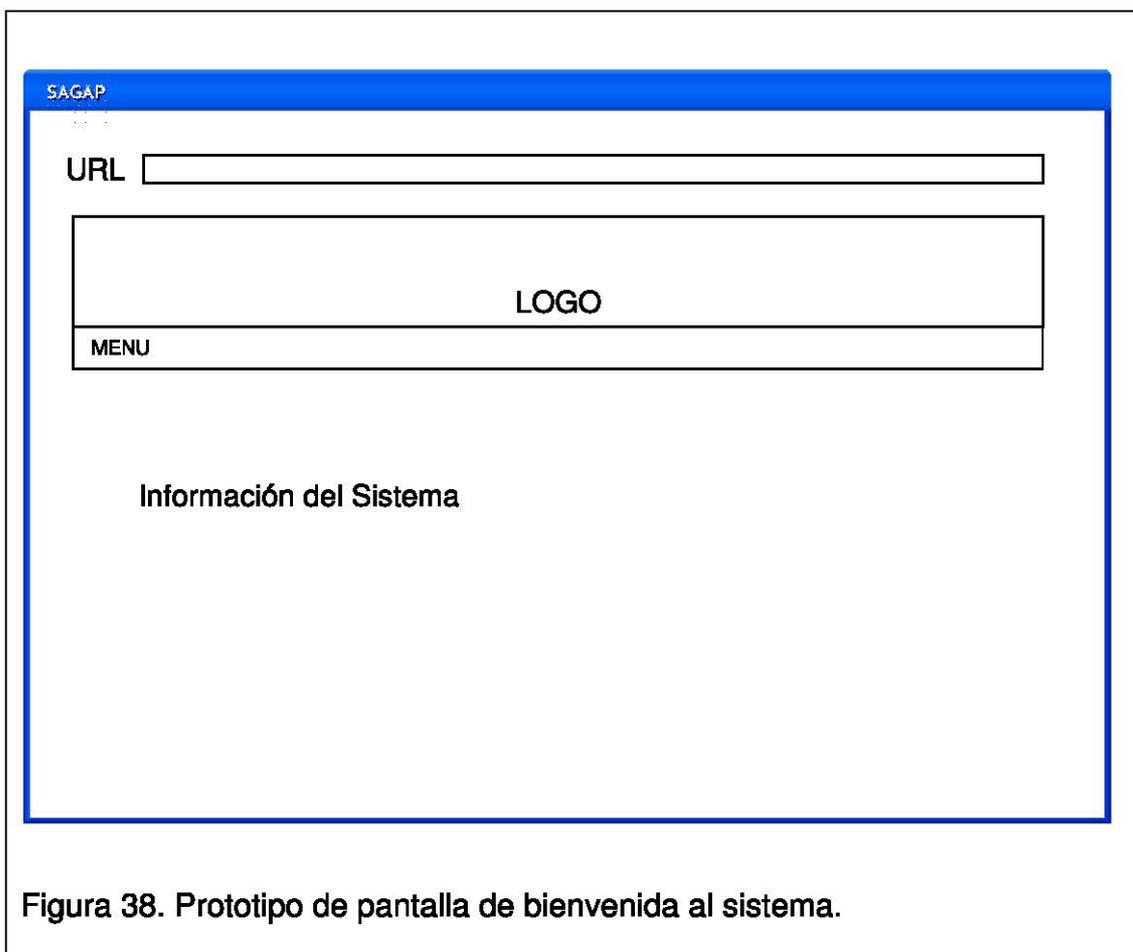


Figura 38. Prototipo de pantalla de bienvenida al sistema.

Tabla 21. Ingreso al sistema.

PANTALLA	Ingreso al sistema.
DESCRIPCIÓN	Prototipo de ingreso al sistema con nombre de usuario y contraseña para ser verificado y autorizado.

Figura 39. Prototipo de pantalla de bienvenida al sistema.

Tabla 22. Formulario web general.

PANTALLA	Formulario web general.
DESCRIPCIÓN	Prototipo en el cual se van a presentar los distintos formularios de creación de usuarios, historial clínicos, recetas, medicamentos, pacientes. Al mismo tiempo dentro del prototipo se pueden visualizar los datos ingresados previamente dependiendo de la opción escogida en el menú de opciones ubicado a la derecha del formulario o en el menú principal ubicado en la parte superior.

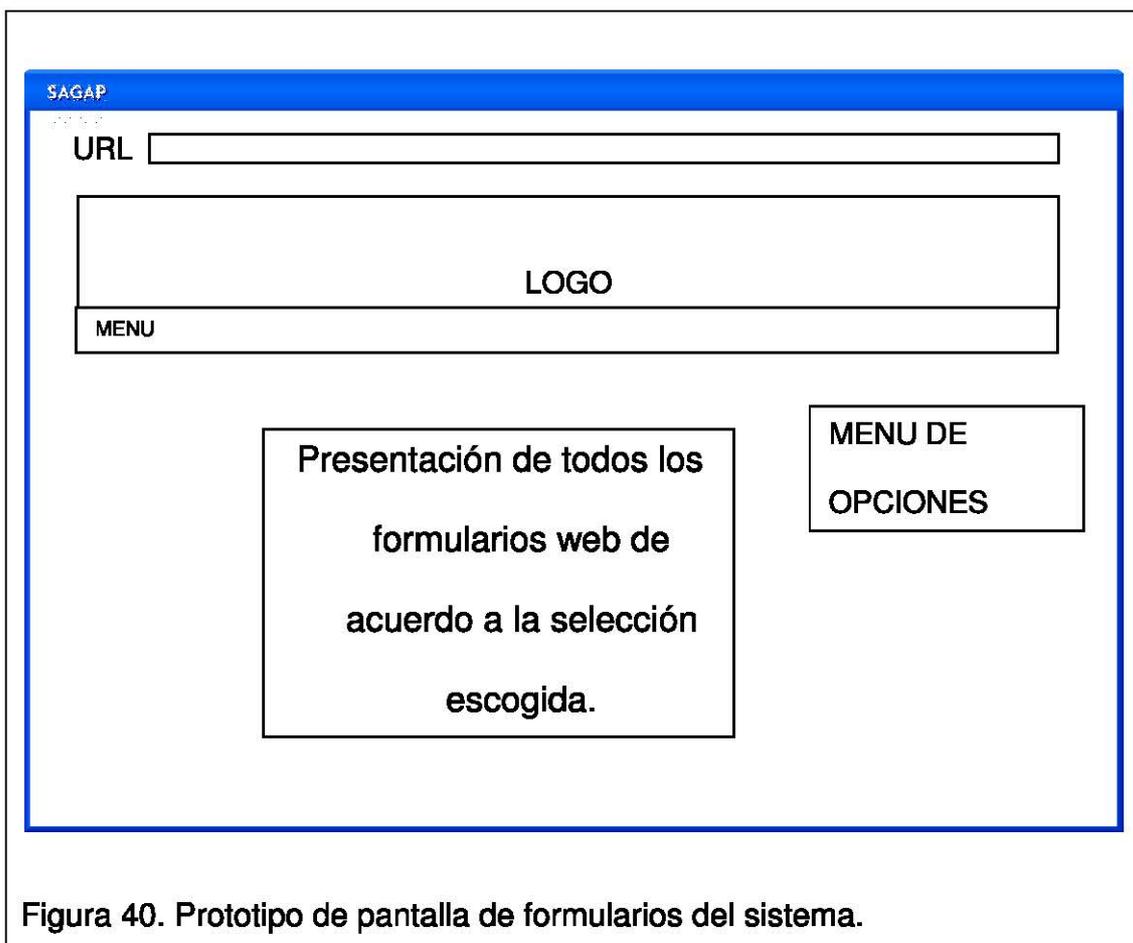


Tabla 23. Ingreso de datos para periodontograma.

PANTALLA	Ingreso de datos para periodontograma.
DESCRIPCIÓN	Prototipo por cual se ingresarán los datos de margen gingival, sondaje inicial, margen gingival 2, sondaje reevaluación para el diagnóstico final.

Sección

Tipo

Nombre del Paciente

Datos Periodontograma Superior

ODONTOGRAMA

Sección

Tipo

Datos Periodontograma Inferior

Botón Ingresar Datos

Figura 41. Prototipo de pantalla de ingreso de periodontograma.

Tabla 24. Ingreso odontograma.

PANTALLA	Ingreso odontograma.
DESCRIPCIÓN	Prototipo por donde se ingresarán los primeros datos de evaluación de cada pieza dental de acuerdo a la patología de cada pieza dental.

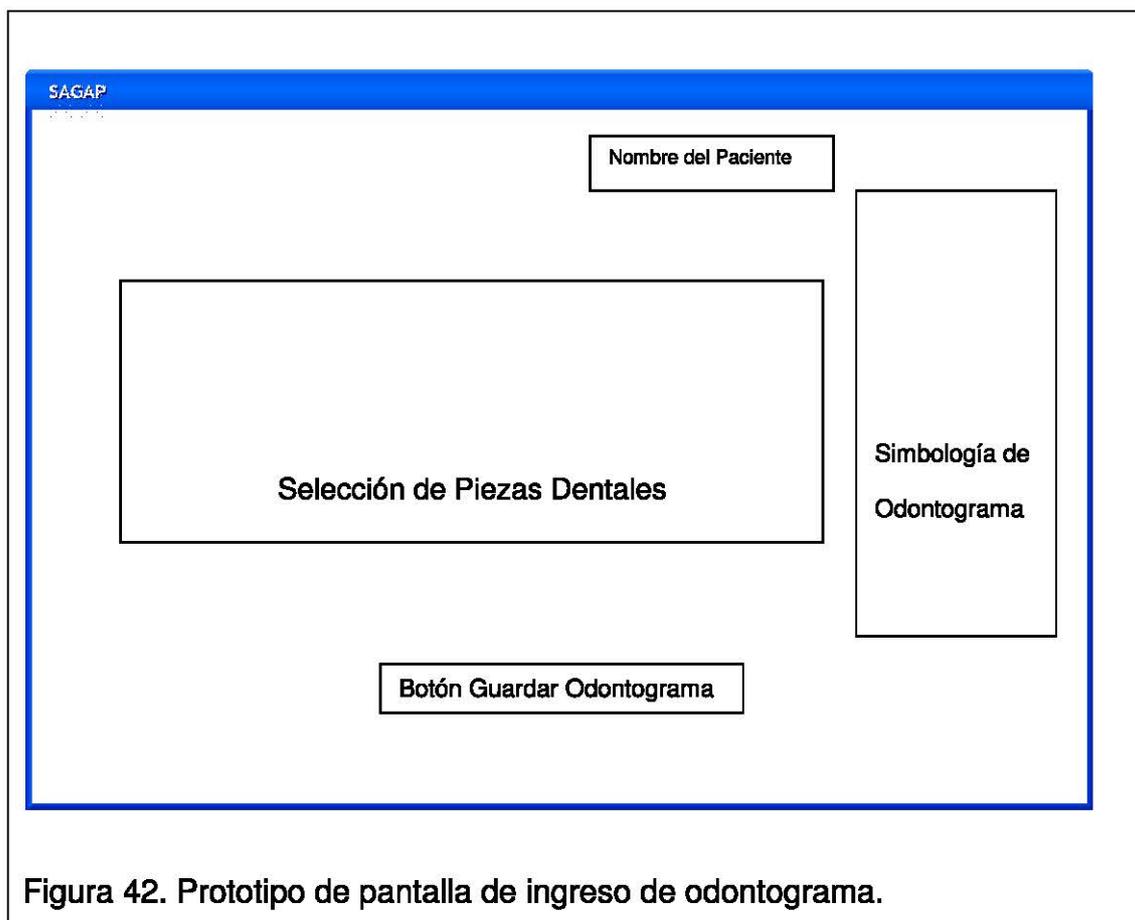


Tabla 25. Periodontograma

PANTALLA	Periodontograma
DESCRIPCIÓN	Prototipo de visualización de todas las muestras tomadas para evaluación de enfermedad, diagramación de líneas de nivel de enfermedad, diagnóstico de cada pieza dental y diagnóstico de enfermedad total de cada paciente.

SAGAP

Nombre del Paciente

Margen Gingival

Sondaje Inicial

Margen Gingival 2

Datos de Piezas Superiores

Periodontograma

Margen Gingival

Sondaje Inicial

Margen Gingival 2

Datos de Piezas Inferiores

Diagnóstico

Figura 43. Prototipo de pantalla de visualización de registros de periodontograma.

4.17 Diseño de Pantallas

A continuación el diseño de pantallas para el sistema SAGAP.

Tabla 26. Pantalla de bienvenida al sistema.

Descripción
La pantalla muestra la bienvenida al sistema, dando al usuario la oportunidad de ingresar a sesión para acceder a las funciones del sistema

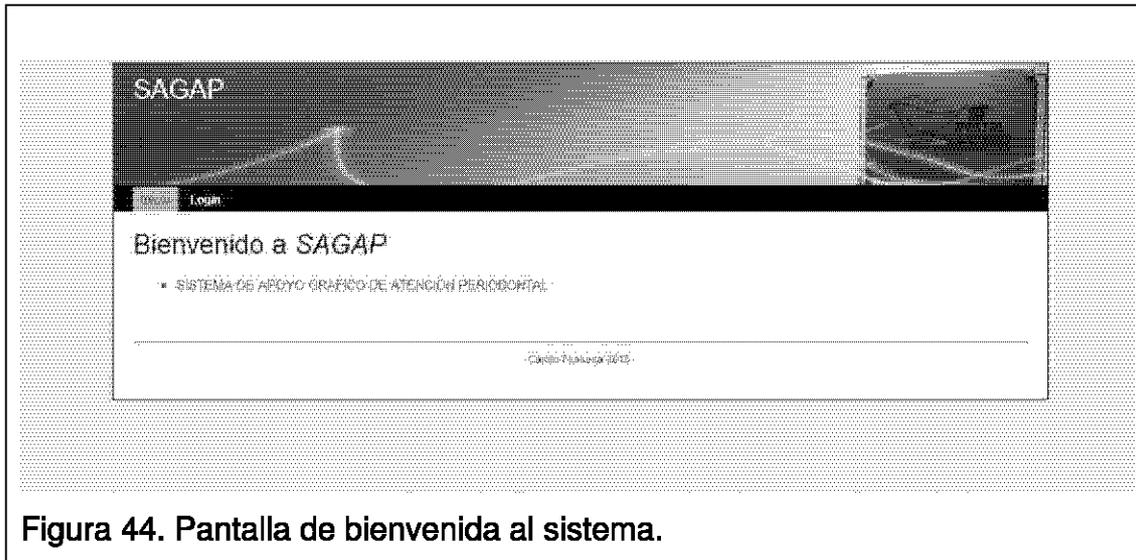


Figura 44. Pantalla de bienvenida al sistema.

Tabla 27. Pantalla de ingreso al sistema.

Botón	Activado	Acción
Entrar	Siempre	Permite ingresar al sistema verificando la autenticidad de las credenciales de sesión para cada usuario.

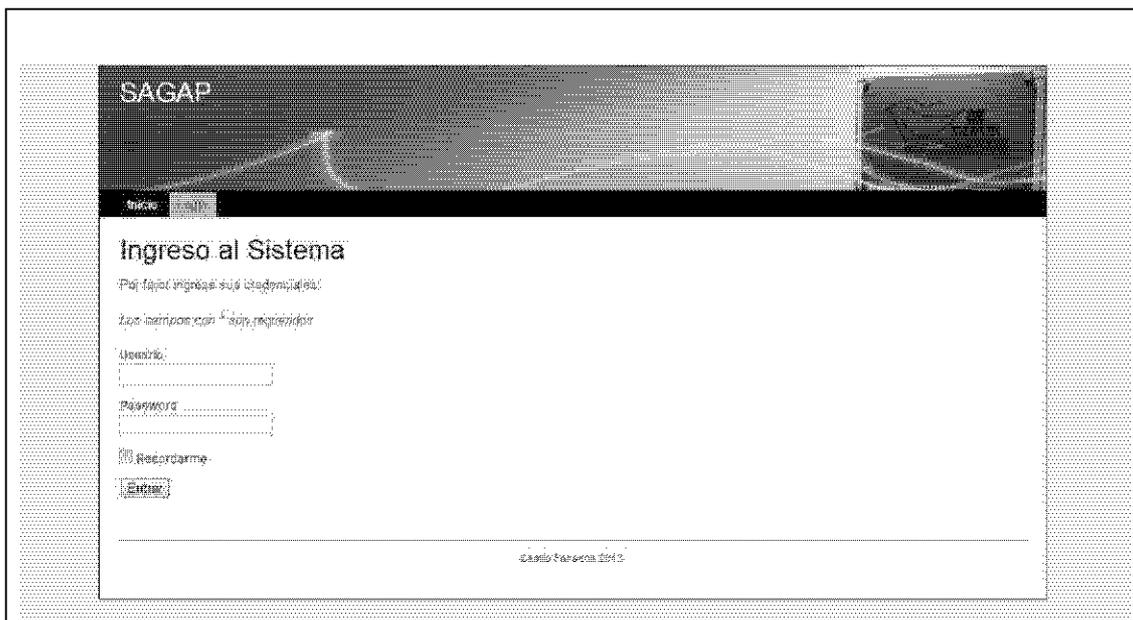


Figura 45. Pantalla de ingreso al sistema.

Tabla 28. Pantalla de administración de usuarios.

Descripción
Pantalla en la cual se puede observar todos los usuarios creados para ingreso al sistema.

Tabla 29. Menú de opciones de administración de usuarios.

Menú	Activado	Acción
Crear usuario	Siempre	Permite la creación de un nuevo usuario.
Administrar Usuario	Siempre	Permite realizar tareas de administración, creación actualización de un usuario del sistema.



Figura 46. Pantalla de administración de usuarios.

Tabla 30. Descripción de acción para creación de usuarios.

Botón	Activado	Descripción
Crear Usuario	Siempre	Valida que la información en el formulario se encuentre correcta y crea un nuevo usuario en la base de datos.

Tabla 31. Menú de opciones de creación de usuarios

Menú	Activado	Acción
Crear usuario	Siempre	Permite la creación de un nuevo usuario.
Administrar Usuario	Siempre	Permite realizar tareas de administración, creación actualización de un usuario del sistema.



Figura 47. Pantalla de creación de usuarios.

CAPITULO 5

CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

El proyecto de tesis "SISTEMA GRÁFICO DE APOYO AL DIAGNÓSTICO DE PERIODONTITIS PARA MEJORAR LA ATENCIÓN DE LOS PACIENTES" se construyó utilizando la metodología de desarrollo Scrum y en lenguaje PHP a través de Yii Framework. En la finalización del proyecto se arrojan las conclusiones siguientes.

El proyecto se cumplió en los plazos establecidos debido a que Scrum es una metodología que presiona para producir resultados completos y funcionando en el tiempo establecido, demostrando así la efectividad de realizar tareas en intervalos cortos a fin de generar la solución deseada.

Las historias de usuario fueron un factor clave porque fueron muy específicas, los periodoncistas sabían que es exactamente lo que necesitaban, eso influyó para que el proyecto no requiera de cambios significativos en la marcha o retrasos uno o varios cambios de planes eventuales en el sistema.

SAGAP gestiona los procesos de búsqueda, reevaluación en el análisis de criterios, almacenamiento datos específicos para cada paciente de forma satisfactoria, permitiendo así tener una base de datos organizada con el historial del paciente, cumpliendo así los objetivos propuestos al inicio del proyecto.

El desarrollo con Yii framework permitió crear una aplicación con una interfaz de características profesionales, debido a su organización se pudo comprobar la efectividad de la plataforma.

La metodología de desarrollo Scrum fue útil para desarrollar el proyecto SAGAP dejando ver que Scrum puede servir para el desarrollo de cualquier tipo de proyecto, si se cuenta con un equipo de desarrollo experimentado podemos fácilmente emplear la misma metodología en proyectos de desarrollo de software con un alcance mayor y más complejo.

En el medio ecuatoriano todavía prevalece el desarrollo de software con metodologías tradicionales debido a que en muchas instituciones educativas aún se mantiene la enseñanza basada en dichas metodologías evitando así que las nuevas generaciones de estudiantes y futuros profesionales se integren adecuadamente a las metodologías ágiles.

Muchas veces las empresas que requieren de servicios de desarrollo de soluciones de software a gran escala o en grandes proyectos en Ecuador conservan la teoría de emplear metodologías tradicionales evitando así dar el gran salto tecnológico que permita empezar a promover la eficiencia de las metodologías ágiles.

Yii framework es una herramienta muy potente para desarrollar software de calidad y seguro, con un conocimiento previo de MVC es una excelente opción para el desarrollo de aplicaciones web.

El sistema se mostró de manera sólida en cuanto a la carga de modelos con el framework Yii, las vistas y los controladores no presentaron bugs ni problemas, todas las dudas en cuanto al manejo de Yii se encontraron en el sitio web de Yii framework.

Para el presente proyecto se agregó diagramación UML, diagramas de clases, diagramas de base de datos, prototipos de interfaces, diagramas DFD, en base a la metodología tradicional con el propósito de mostrar de forma más clara el proceso de desarrollo y el contenido del sistema debido a que Scrum es muy limitado en cuanto a documentación, ya que el tiempo empleado para documentar se lo invierte en minimizar documentación y producir software de manera sólida.

5.2 Recomendaciones

El presente proyecto podría utilizarse con fin comercial porque para los profesionales ortodontistas en el mercado no existe una solución de software exactamente como la propuesta por SAGAP debido a que solo existen soluciones que abarcan la mayor cantidad de ramas de la odontología posible sin enfocarse en una solución completamente objetiva.

Principalmente cuando se utiliza Yii framework debe ser con un previo conocimiento de MVC y en lo posible del mismo Yii framework debido a que la información es muy específica y es entendible desde el punto de vista de una persona que maneja o trabaja mucho con MVC, no existe mucha documentación para empezar desde cero.

Es muy recomendable utilizar HTML5 ya que se obtiene resultados de calidad en cuanto a graficación y animaciones web por encontrarse a la altura de Adobe Flash y debido a su costo por ser de código abierto, hay una gran cantidad de documentación disponible para poner en marcha cualquier tipo de solución, además existe una comunidad colaborativa en base a experiencias que sirve para brindar cualquier ayuda propuesta en la marcha del desarrollo del software.

La graficación se podría implementar a un nivel superior empleando la misma solución pero para tener resultados de animaciones en 3D, HTML5 sí permite realizar éste tipo de gráficos de modo que sin duda SAGAP podría extenderse multiplicando significativamente su efectividad como entorno visual de una so

Se debe seguir estrictamente la metodología Scrum, por su importancia de completar tareas específicas en un tiempo determinado, si no se siguen fielmente las reglas de Scrum, esto puede ser un factor influyente para retrasos significativos en el Sprint o decisivo para el fracaso total del proyecto.

Si el proyecto se llevase a una gran escala debería trabajar con el complemento Yii como un servicio web, instalándolo en un servidor de manera que pueda proveer el servicio a todos los equipos la empresa requieran utilizar la aplicación, esto podría manejarse a nivel nacional.

Sería muy útil integrar el sistema a un módulo de manejo de citas, permitiendo al cliente registrarse, obtener información o modificar una cita a través de internet, dándole al cliente la opción de administrar su tiempo permitiendo que la empresa MF Dental no pare sus actividades ya que el sistema trataría de ubicar a otros pacientes en espacios de citas canceladas.

Como adición al sistema se podría implementar la compatibilidad multiplataforma para que el sistema pueda ser utilizado en un dispositivo móvil, tablet o pc, de ésta forma se podría usar cierto tipo de dispositivos para tareas específicas como simplemente tomar datos del paciente en un ipad o si el sistema se integrase a un módulo de citas, el paciente podría planificar su cita desde su celular.

REFERENCIAS

Academicmarist. (s.f.).

Apache Server Architecture.

Recuperado el 20 de Noviembre de 2012 de

<http://www.academic.marist.edu%2F~jzbv%2Farchitecture%2FProjects08%2FApacheServerArchitectureProject.ppt&ei=SDW7UOHALYGE9QS754GICw&usg=AFQjCNEElw7OhVE3lpmO5Fs8AuKg35A8jw>

Aceproject. (s.f.).

Encriptamiento.

Recuperado el 11 de Octubre de 2012 de

<http://aceproject.org/main/espanol/et/ete08.htm>

Adrformacion. (2012).

Curso de Desarrollo de páginas web con PHP.

Recuperado el 05 de Septiembre de 2012 de

<http://www.adrformacion.com/cursos/php2/leccion1/tutorial1.html>

Agilealliance. (s.f.).

WHAT IS THE ALLIANCE?

Recuperado el 11 de Octubre de 2012 de <http://www.agilealliance.org/>

Apachefriends. (s.f.).

XAMPP for Windows.

Recuperado el 20 de Noviembre de 2012 de

<http://www.apachefriends.org/en/xampp-windows.html#641>

Baker, S. (2012). *Introduction to agile and Scrum*. London, UK: ThinkBox.

Bozzuto, B. (2009).

Agile Software Development Beyond Cheaper, Faster & Less defects.

Boston: Big Visible Solutions.

Recuperado de <http://www.bigvisible.com>

Codecompiling. (2010).

¿Sin MVC?

Recuperado el 29 de Septiembre de 2012 de

http://www.codecompiling.net/files/slides/IS_clase_11_mvc.pdf

Cplusplus. (s.f.).

A brief description.

Recuperado el 11 de Octubre de 2012 de

<http://www.cplusplus.com/info/description/>

Culturación. (2011).

Cómo convertir un PC en servidor web.

Recuperado el 03 de Septiembre de 2012 de

<http://culturacion.com/2011/12/como-convertir-un-pc-en-servidor-web>

Docs.oracle. (s.f.).

Database Triggers.

Recuperado el 11 de Octubre de 2012 de

http://docs.oracle.com/cd/A57673_01/DOC/server/doc/SCN73/ch15.htm

Docs.oracle. (s.f.).

URL.

Recuperado el 11 de Octubre de 2012 de

<http://docs.oracle.com/javase/6/docs/api/java/net/URL.html>

Dosideas. (s.f.).

Las 9 cualidades del código limpio.

Recuperado el 11 de Octubre de 2012 de

<http://www.dosideas.com/noticias/desarrollo-de-software/965-las-9-cualidades-del-codigo-limpio.html>

Gnu. (s.f.).

What is GNU?.

Recuperado el 11 de Octubre de 2012 de <http://www.gnu.org/>

Jeremy, K. (2012). *Html5 for web designers*. Chicago, USA: A Book Apart.

Jquery. (s.f.).

What is jQuery.

Recuperado el 11 de Octubre de 2012 de <http://jquery.com/>

Linuxtoday. (2012). 10 Best Free Web Application Frameworks. Recuperado el 18 de Septiembre de 2012 de <http://www.linuxtoday.com/upload/10-best-free-web-application-frameworks-120525021957.html>

Lubbers, P. (2010). *Pro html5 programming*. San Antonio, USA: Apress.

Marakov, A. (2011). *Yii 1.1 Application Development Cookbook*. Birmingham, UK: Packt Publishing.

Metodología (s.f.).

Visión general del proceso.

Recuperado el 03 de Septiembre de 2012 de

<http://www.sgv.es/nueva/index.php/empresa/metodologia?tmpl=component&print=1>

Mobypicture. (2012).

Moby Picture.

Recuperado el 23 de Agosto de 2012 de

http://www.mobypicture.com/user/JoBrosMyHeros_/view/13599835

Msdn.microsoft. (s.f.).

Fundamentos de la POO.

Recuperado el 11 de Octubre de 2012 de [http://msdn.microsoft.com/es-](http://msdn.microsoft.com/es-es/library/bb972232.aspx)

[es/library/bb972232.aspx](http://msdn.microsoft.com/es-es/library/bb972232.aspx)

Msdn.microsoft. (s.f.).

Stored Procedures (Database Engine).

Recuperado el 11 de Octubre de 2012 de [http://msdn.microsoft.com/en-](http://msdn.microsoft.com/en-us/library/ms190782.aspx)

[us/library/ms190782.aspx](http://msdn.microsoft.com/en-us/library/ms190782.aspx)

Netcraft. (2012).

November 2012 Web Server Survey.

Recuperado el 28 de Noviembre de 2012 de

<http://news.netcraft.com/archives/2012/11/01/november-2012-web-server-survey.html>

Odontologicosanpablo. (s.f.).

Enciclopedia de la salud.

Recuperado el 11 de Octubre de 2012 de

<http://www.odontologicosanpablo.com.pe/enciclopedia.html>

Owasp. (s.f.).

Cross-Site Request Forgery (CSRF).

Recuperado el 11 de Octubre de 2012 de

[https://www.owasp.org/index.php/Cross-Site_Request_Forgery_\(CSRF\)](https://www.owasp.org/index.php/Cross-Site_Request_Forgery_(CSRF))

Owasp. (s.f.).

Cross-site Scripting (XSS).

Recuperado el 11 de Octubre de 2012 de

[https://www.owasp.org/index.php/Cross-site_Scripting_\(XSS\)](https://www.owasp.org/index.php/Cross-site_Scripting_(XSS))

Palacio, J. (2008). *Scrum Manager: Gestión de proyectos*. Barcelona, España: SafeCreative.

Perl. (s.f.). *The Perl Programming Language*. Recuperado el 11 de Octubre de 2012 de <http://www.perl.org/>

PHP. (s.f.).

El paradigma de la Programación Orientada a Objetos en PHP y el patrón de arquitectura de Software MVC.

Recuperado el 12 de Octubre de 2012 de

<http://es.scribd.com/doc/92524163/El-paradigma-de-la-Programacion-Orientada-a-Objetos-en-PHP-y-el-patron-de-arquitectura-de-Software-MVC>

Php. (s.f.).

History of PHP.

Recuperado el 11 de Octubre de 2012 de

<http://php.net/manual/en/history.php.php>

Php. (s.f.).

SQL Injection.

Recuperado el 11 de Octubre de 2012 de

<http://php.net/manual/en/security.database.sql-injection.php>

Phpframeworks. (2012).

We compare all kinds of PHP Frameworks.

Recuperado el 03 de Noviembre de 2012 de <http://www.phpframeworks.com>

Pradosoft. (s.f.).

What is PRADO?

Recuperado el 11 de Octubre de 2012 de <http://www.pradosoft.com/>

Proyectosagiles. (s.f.).

Beneficios de Scrum.

Recuperado el 11 de Octubre de 2012 de

<http://www.proyectosagiles.org/beneficios-de-scrum>

Proyectosagiles. (s.f.).

Historia de Scrum.

Recuperado el 03 de Septiembre de 2012 de

<http://www.proyectosagiles.org/historia-de-scrum>

Proyectosagiles. (s.f.).

Lista de objetivos/requisitos priorizada (Product Backlog).

Recuperado el 03 de Septiembre de 2012 de

<http://www.proyectosagiles.org/lista-requisitos-priorizada-product-backlog>.

Proyectosagiles. (s.f.).

Lista de tareas de la iteración (Sprint Backlog).

Recuperado el 03 de Septiembre de 2012 de

<http://www.proyectosagiles.org/lista-tareas-iteracion-sprint-backlog>

Proyectosagiles. (s.f.).

Qué es SCRUM?.

Recuperado el 03 de Septiembre de 2012 de

<http://www.proyectosagiles.org/que-es-scrum>

Rico, D. (2009). *The Business Value of Agile Software Methods for Systems Development*. Recuperado el 03 de Septiembre de 2012.

Scrum. (s.f.).

Partners.

Recuperado el 11 de Octubre de 2012 de <http://www.scrum.org/About/Work-With-Us/Partners>

Tcpsi. (s.f.).

Gobierno IT.

Recuperado el 20 de Noviembre de 2012 de http://www.tcpsi.com/servicios/gobierno_ti.htm

Thestadisgggroup (2012).

The chaos manifesto.

Recuperado el 22 de Septiembre de 2012 de <http://www.sds-consulting.com/blog/agile-projects-successful-3x-more-non-agile-projects>

Valade, J. (2005). *PHP & MYSQL Everyday Apps for Dummies*. Hoboken, USA: Wiley Publishing.

W3. (s.f.).

CGI: Common Gateway Interface.

Recuperado el 11 de Octubre de 2012 de <http://www.w3.org/CGI/>

W3resource. (2012).

MySQL Data Types.

Recuperado el 06 de Noviembre de 2012 de <http://www.w3resource.com/mysql/mysql-data-types.php>

W3schools. (s.f.).

AJAX Tutorial.

Recuperado el 11 de Octubre de 2012 de <http://www.w3schools.com/ajax/>

W3schools. (s.f.).

HTML5 Canvas.

Recuperado el 11 de Octubre de 2012 de

http://www.w3schools.com/html/html5_canvas.asp

W3schools. (s.f.).

JavaScript Introduction.

Recuperado el 11 de Octubre de 2012 de

http://www.w3schools.com/js/js_intro.asp

Wikipedia. (s.f.).

Aparato estomatognático.

Recuperado el 11 de Octubre de 2012 de

http://es.wikipedia.org/wiki/Aparato_estomatogn%C3%A1tico

Wikipedia. (s.f.).

Application programming interface.

Recuperado el 11 de Octubre de 2012 de

http://en.wikipedia.org/wiki/Application_programming_interface

Wikipedia. (s.f.).

C (lenguaje de programación).

Recuperado el 11 de Octubre de 2012 de

[https://es.wikipedia.org/wiki/C_\(lenguaje_de_programaci%C3%B3n\)](https://es.wikipedia.org/wiki/C_(lenguaje_de_programaci%C3%B3n))

Wikipedia. (s.f.).

Cache (computing).

Recuperado el 11 de Octubre de 2012 de

[http://en.wikipedia.org/wiki/Cache_\(computing\)](http://en.wikipedia.org/wiki/Cache_(computing))

Wikipedia. (s.f.).

Capa de aplicación.

Recuperado el 11 de Octubre de 2012 de

http://es.wikipedia.org/wiki/Capa_de_aplicaci%C3%B3n

Wikipedia. (s.f.).

Commonly used terms of relationship and comparison in dentistry.

Recuperado el 11 de Octubre de 2012 de

http://en.wikipedia.org/wiki/Commonly_used_terms_of_relationship_and_comparison_in_dentistry

Wikipedia. (s.f.).

Computación en la nube.

Recuperado el 11 de Octubre de 2012 de

http://es.wikipedia.org/wiki/Computaci%C3%B3n_en_la_nube

Wikipedia. (s.f.).

Create, read, update and delete.

Recuperado el 11 de Octubre de 2012 de

http://en.wikipedia.org/wiki/Create,_read,_update_and_delete

Wikipedia. (s.f.).

Diseño web.

Recuperado el 11 de Octubre de 2012 de

http://es.wikipedia.org/wiki/Dise%C3%B1o_web

Wikipedia. (s.f.).

HTML.

Recuperado el 11 de Octubre de 2012 de <http://es.wikipedia.org/wiki/Html>

Wikipedia. (s.f.).

IIS.

Recuperado el 11 de Octubre de 2012 de <http://en.wikipedia.org/wiki/IIS>

Wikipedia. (s.f.).

Integrated development environment.

Recuperado el 11 de Octubre de 2012 de

http://en.wikipedia.org/wiki/Integrated_development_environment

Wikipedia. (s.f.).

Integrated development environment.

Recuperado el 11 de Octubre de 2012 de

http://en.wikipedia.org/wiki/Integrated_development_environment

Wikipedia. (s.f.).

IP address.

Recuperado el 11 de Octubre de 2012 de

http://en.wikipedia.org/wiki/IP_address

Wikipedia. (s.f.).

Kernel (computing).

Recuperado el 11 de Octubre de 2012 de

[http://en.wikipedia.org/wiki/Kernel_\(computing\)](http://en.wikipedia.org/wiki/Kernel_(computing))

Wikipedia. (s.f.).

NTFS.

Recuperado el 11 de Octubre de 2012 de <https://en.wikipedia.org/wiki/NTFS>

Wikipedia. (s.f.).

Página web.

Recuperado el 11 de Octubre de 2012 de

https://es.wikipedia.org/wiki/P%C3%A1gina_web

Wikipedia. (s.f.).

Plug-in (computing).

Recuperado el 11 de Octubre de 2012 de [https://en.wikipedia.org/wiki/Plug-in_\(computing\)](https://en.wikipedia.org/wiki/Plug-in_(computing))

Wikipedia. (s.f.).

Scaffold (programming).

Recuperado el 11 de Octubre de 2012 de [http://en.wikipedia.org/wiki/Scaffold_\(programming\)](http://en.wikipedia.org/wiki/Scaffold_(programming))

Wikipedia. (s.f.).

Software bug.

Recuperado el 11 de Octubre de 2012 de http://en.wikipedia.org/wiki/Software_bug

Wikipedia. (s.f.).

Software development kit.

Recuperado el 11 de Octubre de 2012 de http://en.wikipedia.org/wiki/Software_development_kit

Wikipedia. (s.f.).

SSL.

Recuperado el 11 de Octubre de 2012 de <http://en.wikipedia.org/wiki/Ssl>

Wikipedia. (s.f.).

Thread (computing).

Recuperado el 11 de Octubre de 2012 de [http://en.wikipedia.org/wiki/Thread_\(computing\)](http://en.wikipedia.org/wiki/Thread_(computing))

Wikipedia. (s.f.).

Virtualization.

Recuperado el 11 de Octubre de 2012 de
<http://en.wikipedia.org/wiki/Virtualization>

Wikipedia. (s.f.).

Yii.

Recuperado el 22 de Noviembre de 2012 de <http://es.wikipedia.org/wiki/Yii>

Windows.microsoft. (s.f.).

Service Pack Center.

Recuperado el 11 de Octubre de 2012 de <http://windows.microsoft.com/en-us/windows/service-packs-download#sptabs=win8>

Yale. (s.f.).

Introduction to TCP/IP.

Recuperado el 11 de Octubre de 2012 de
<http://www.yale.edu/pclt/COMM/TCPIP.HTM>

Yiiframework. (s.f.).

Active Record.

Recuperado el 11 de Octubre de 2012 de
<http://www.yiiframework.com/doc/guide/1.1/en/database.ar>

ANEXOS

ANEXO 1

MANUAL DE INSTALACIÓN Y CONFIGURACIÓN DE APLICACIONES DEL SISTEMA

Instalación de XAMPP Server

Como recomendación XAMPP indica que no se instale XAMPP Server en la ruta C:/Program Files porque podría ocasionar conflictos con permisos del usuario. Para instalar Yii se debe ejecutar el instalador en modo de administrador y seguir los siguientes pasos.

Pantallas

- Welcome to the XAMPP 1.7.4 Setup Wizard presionar Next.
- Choose Install Location Folder se escoge C:\xampp como destination folder y presionar Next.
- XAMPP Options se habilita Install Apache as Service Install MySQL as Service y presionar Install

Configuración de XAMPP Server

Una vez que se ha instalado XAMPP los procedimientos para su correcto funcionamiento son:

- Agregar en la variable de entorno Path la siguiente ubicación C:\Xampp
- Para finalizar la configuración es necesario editar el archivo php.ini en la ruta C:\xampp\php\
- Se debe eliminar el signo ; para que se habiliten las siguientes extensiones de la siguiente forma:

[XDebug]

```
zend_extension = "C:\xampp\php\ext\php_xdebug.dll"
```

```
xdebug.remote_host = "localhost"
```

```
xdebug.remote_enable = 1
```

```
xdebug.remote_handler = "dbgp"
```

```
xdebug.remote_port = 9000
```

- Para el caso de remote_enable hay que también cambiar el valor de 0 a 1.

```
xdebug.remote_enable = 1
```

- Es necesario también desactivar IIS.

En el caso que el servidor apache no se encuentre habilitado es porque Apache ocupa el servidor 80 o 443(SSL) que son los mismos que utiliza la aplicación Skype, lo que se debe hacer es cerrar Skype o como mejor opción desinstalarlo.

Configuración de MySQL

Para configurar MySQL es necesario seguir los siguientes pasos:

- Editar el archivo config.inc.php que se encuentra en la ruta C:\xampp\phpMyAdmin.
- Habilitar las funciones eliminando /* */ de las siguientes líneas para que queden de la siguiente forma.

```
$cfg['Servers'][$i]['controluser'] = 'pma';
$cfg['Servers'][$i]['controlpass'] = 'pmapass';
$cfg['Servers'][$i]['pmadb'] = 'phpmyadmin';
$cfg['Servers'][$i]['bookmarktable'] = 'pma_bookmark';
$cfg['Servers'][$i]['relation'] = 'pma_relation';
$cfg['Servers'][$i]['table_info'] = 'pma_table_info';
$cfg['Servers'][$i]['table_coords'] = 'pma_table_coords';
```

```
$cfg['Servers'][$i]['pdf_pages'] = 'pma_pdf_pages';  
$cfg['Servers'][$i]['column_info'] = 'pma_column_info';  
$cfg['Servers'][$i]['history'] = 'pma_history';  
$cfg['Servers'][$i]['designer_coords'] = 'pma_designer_coords';
```

- Agregar la siguiente línea de código

```
$cfg['PmaNoRelation_DisableWarning'] = TRUE;
```

- Desde línea de comandos CMD ingresar C:\ mysql -u root -p con password vacío.

```
mysql> GRANT USAGE ON mysql.* TO 'pma'@'localhost' IDENTIFIED BY  
'pmapass';
```

```
mysql> GRANT SELECT (Host, User, Select_priv, Insert_priv, Update_priv,  
Delete_priv, Create_priv, Drop_priv, Reload_priv, Shutdown_priv, Process_priv,  
File_priv, Grant_priv, References_priv, Index_priv, Alter_priv, Show_db_priv,  
Super_priv, Create_tmp_table_priv, Lock_tables_priv, Execute_priv,  
Repl_slave_priv, Repl_client_priv) ON mysql.user TO 'pma'@'localhost';
```

```
mysql> GRANT SELECT ON mysql.db TO 'pma'@'localhost';
```

```
mysql> GRANT SELECT ON mysql.host TO 'pma'@'localhost';
```

```
mysql> GRANT SELECT (Host, Db, User, Table_name, Table_priv,  
Column_priv) ON mysql.tables_priv TO 'pma'@'localhost';
```

```
mysql> exit
```

- Reiniciar mysql y apache server

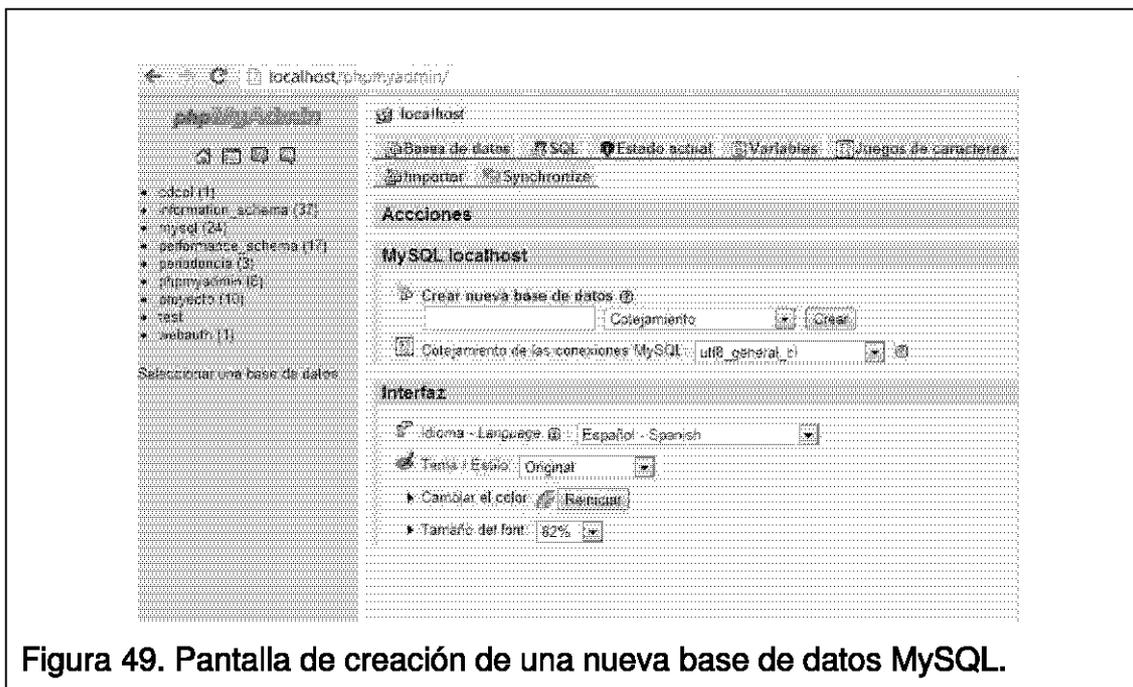


Figura 49. Pantalla de creación de una nueva base de datos MySQL.

Instalación de Yii Framework

Para instalar Yii Framework es necesario seguir los siguientes pasos:

- Descomprimir el archivo yii-1.1.12.b600af.zip.
- Cambiar de nombre de la carpeta yii-1.1.12.b600af por Yii

Configuración de Yii Framework

Para configurar Yii Framework se requiere agregar a la variable de entorno Path el path C:\xampp\htdocs\yii\framework que es la ruta donde se encuentran ubicados los archivos de configuración y motor del framework.

Verificación de la instalación de Yii Framework

Para verificar el correcto funcionamiento de Yii framework se puede ingresar a la dirección <http://localhost/yii/demos/phonebook/> desde cualquier navegador web y se debe mostrar la siguiente pantalla.

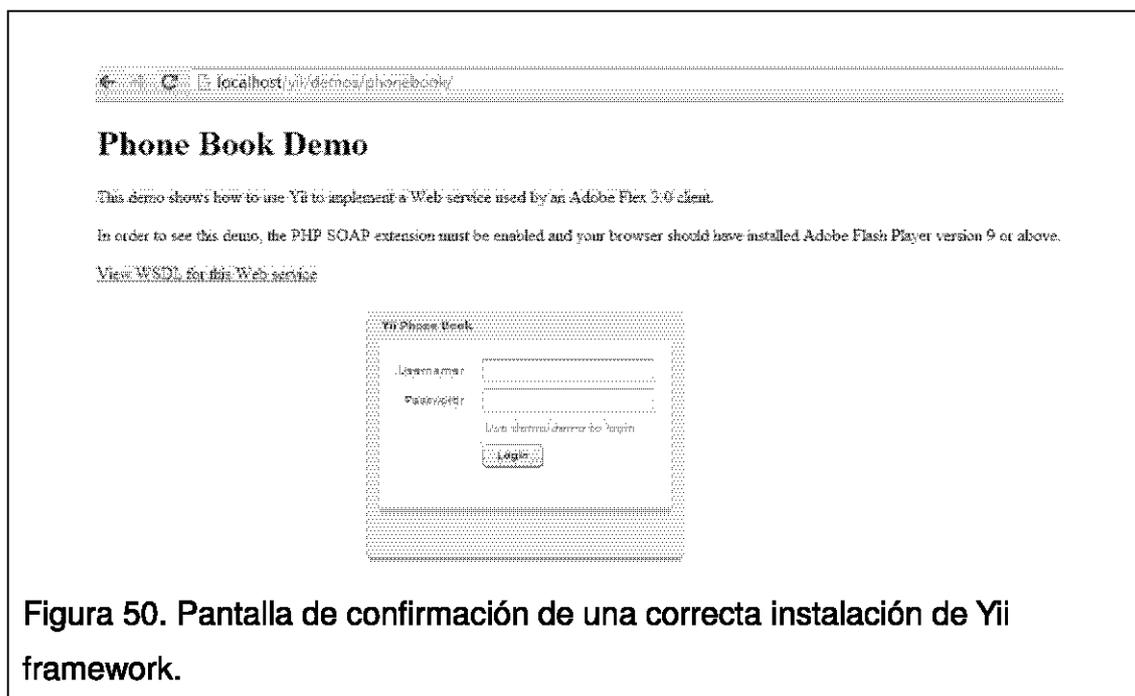


Figura 50. Pantalla de confirmación de una correcta instalación de Yii framework.

Instalación de Eclipse

Los siguientes pasos son necesarios para la instalación de Eclipse:

- Crear la carpeta C:\Eclipse.
- Descomprimir el archivo eclipse-php-helios-SR2-win32.zip.
- Copiar la carpeta descomprimida en C:\Eclipse.
- Ejecutar el archivo eclipse.exe de la ruta C:\Eclipse\eclipse-php-helios-SR2-win32\eclipse

Configuración de Eclipse

Para que eclipse pueda trabajar con PHP y Yii Framework se debe realizar las siguientes configuraciones:

- Seleccionar Internet Explorer como navegador predeterminado desde Windows – Preferences – Web Browser.

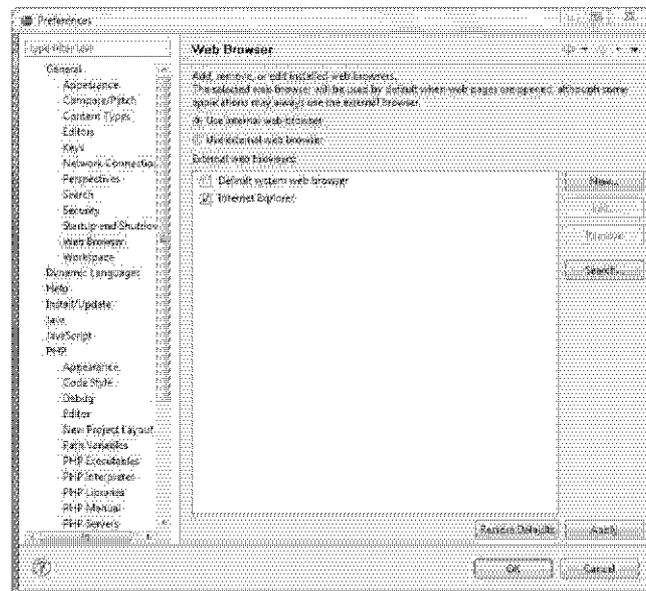


Figura 51. Configuración de Eclipse para el navegador web.

- Selección XDebug como el debugger predeterminado desde PHP – Debug.

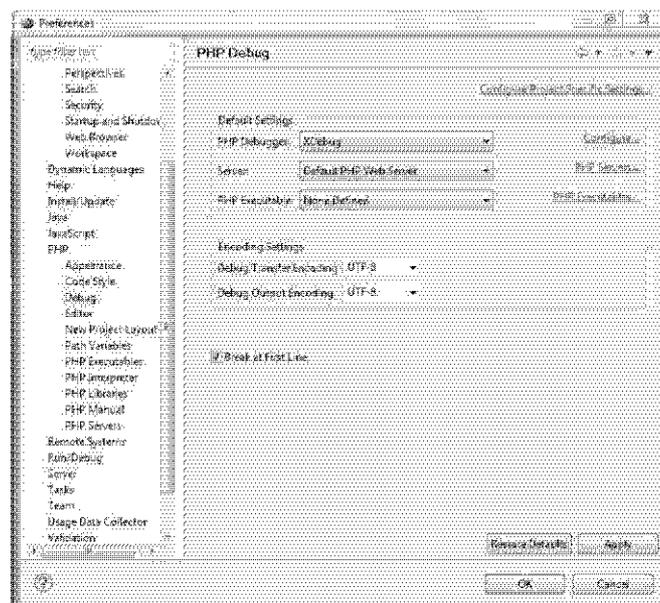


Figura 52. Configuración de Eclipse para PHP.

Instalación de complemento Yii para Eclipse

Yii tiene un complemento para Eclipse que se llama Yiiclipse, el método de instalación es el siguiente:

- Desde Help – Install new software ingresar:
- Name: Yiiclipse
- Location: <http://yiiclipse.maziarz.org/updatesite/STABLE/>

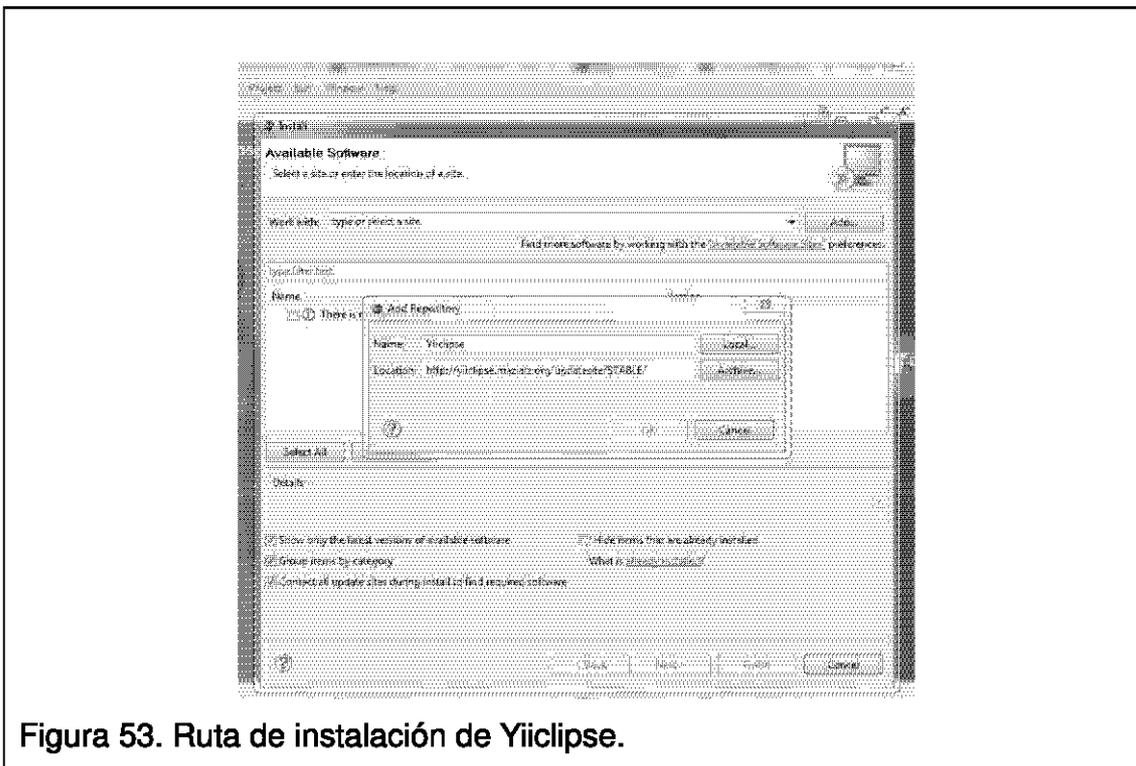


Figura 53. Ruta de instalación de Yiiclipse.

- Una vez que aparece el instalador de Yiiclipse se selecciona el complemento y se presiona Next.

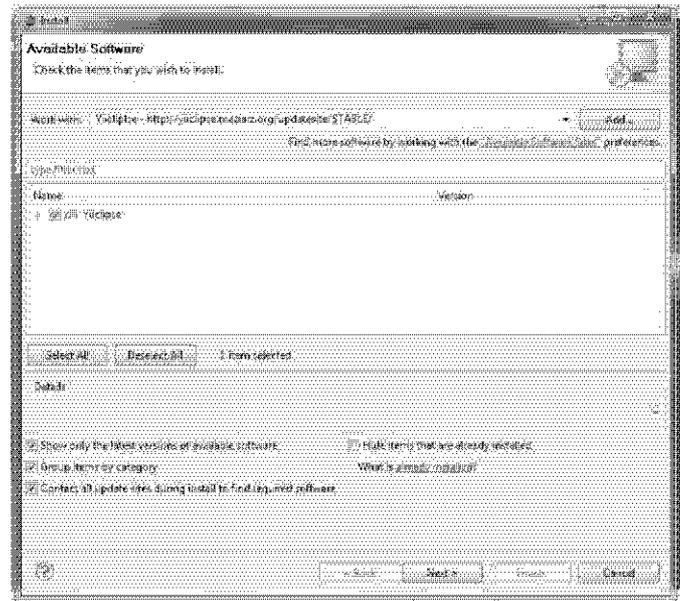


Figura 54. Pantalla de instalación de Yiclipse.

- Confirmación de la instalación presionando Finish.

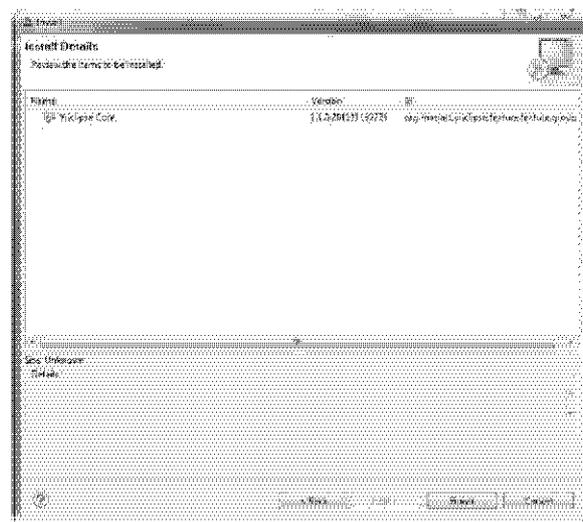


Figura 55. Pantalla de finalización de instalación de Yiclipse.

Instalación de PowerDesigner 12, y Notepad++

Para la instalación de Power Designer y Notepad++ es suficiente y necesario seguir la configuración por defecto hasta que cada asistente finalice la instalación.

ANEXO 2

DIAGRAMAS DE PROCESOS DEL SISTEMA

Ingreso de Consulta Médica

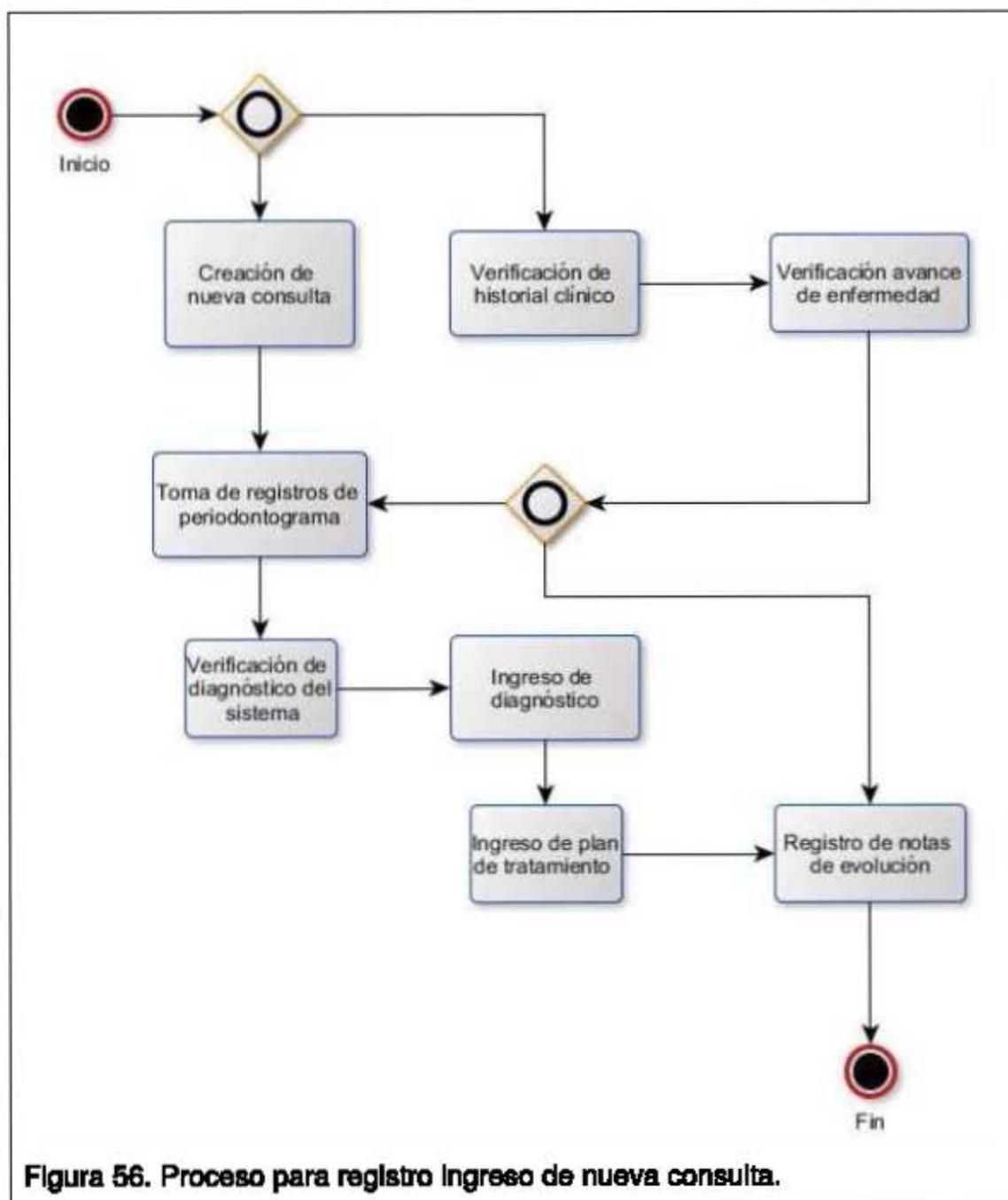


Figura 56. Proceso para registro Ingreso de nueva consulta.

Reporte de Usuarios

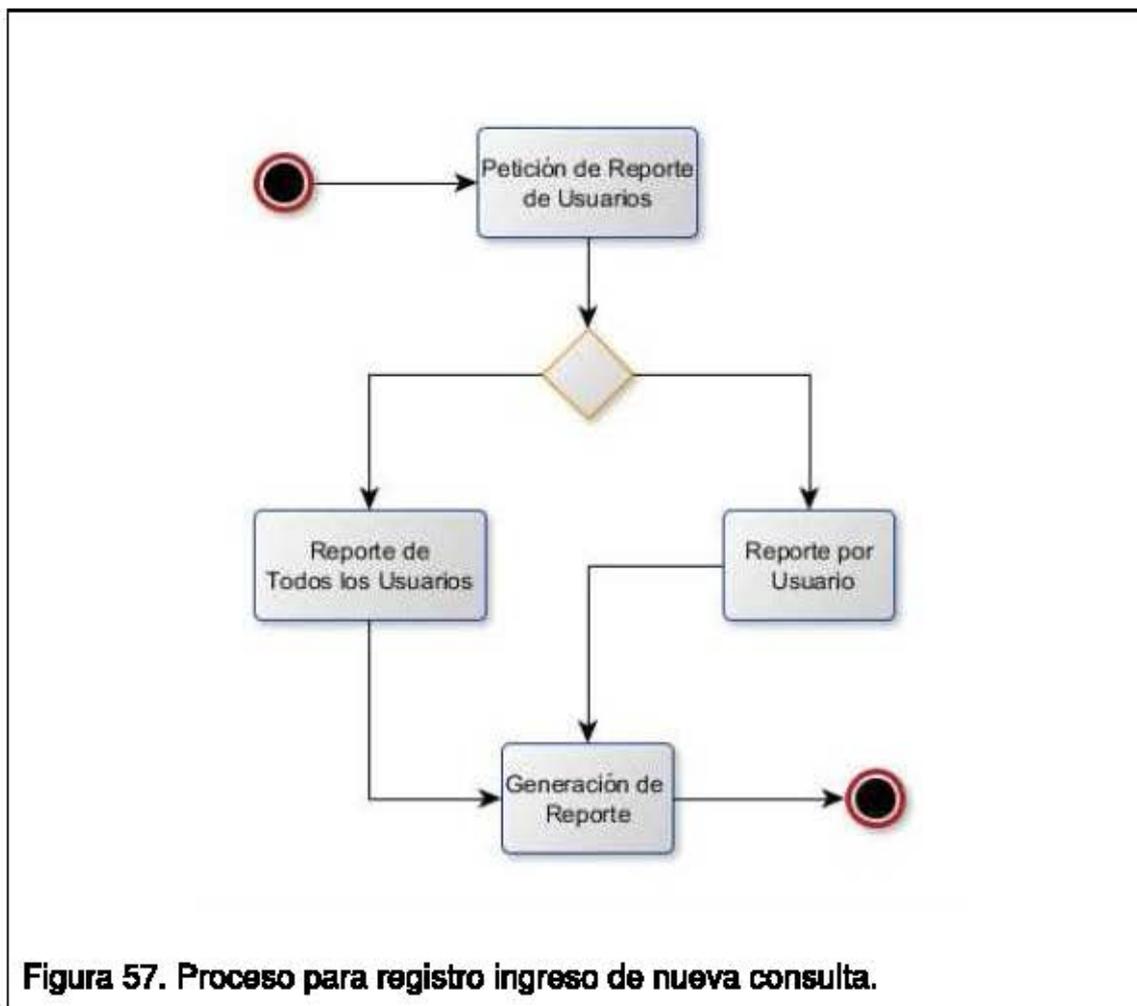


Figura 57. Proceso para registro ingreso de nueva consulta.

Reporte de Historias Clínicas

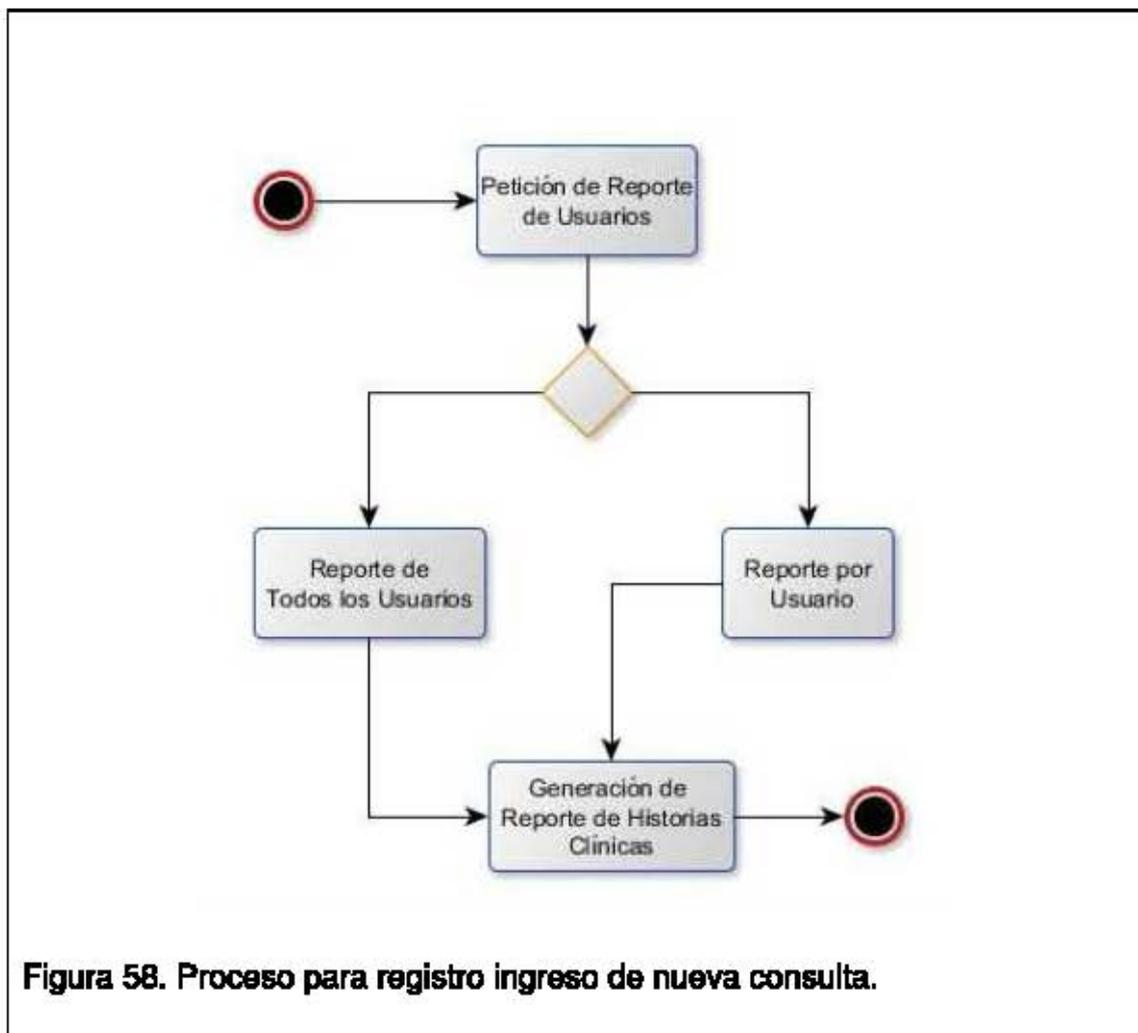


Figura 58. Proceso para registro ingreso de nueva consulta.

ANEXO 3

MANUAL DE USUARIO DEL SISTEMA

Descripción

Pantalla que permite la búsqueda, actualización y eliminación de usuarios de acceso al sistema.

Menú	Activado	Acción
Crear usuario	Siempre	Permite la creación de un nuevo usuario.
Listar Usuario	Siempre	Permite visualizar los usuarios de acceso al sistema existentes.

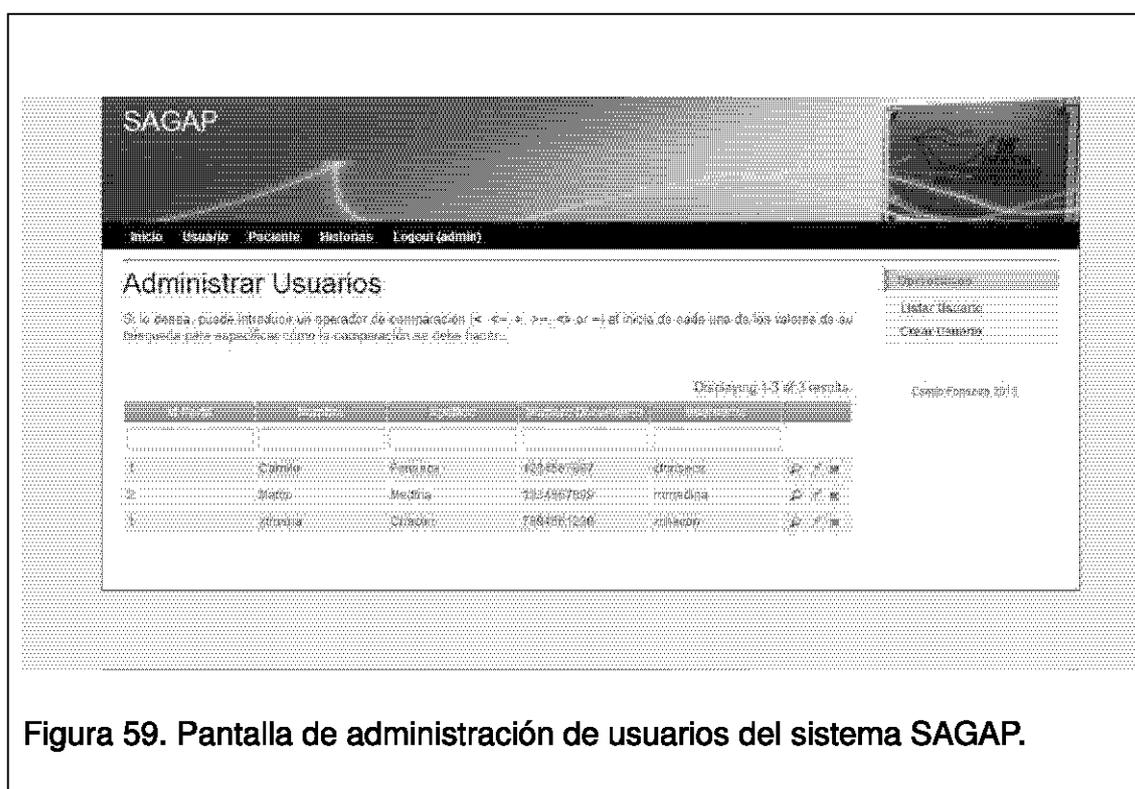


Figura 59. Pantalla de administración de usuarios del sistema SAGAP.

ANEXO 6

Descripción
Pantalla que permite la búsqueda, actualización y eliminación de usuarios de acceso al sistema.

Menú	Activado	Acción
Crear usuario	Siempre	Permite la creación de un nuevo usuario.
Listar Usuario	Siempre	Permite visualizar los usuarios de acceso al sistema existentes.
Actualizar Usuario	Siempre	Permite actualizar la información de un usuario del sistema.
Eliminar Usuario	Siempre	Permite eliminar un usuario del sistema.
Administrar Usuario	Siempre	Permite realizar tareas de administración, creación actualización de un usuario del sistema.

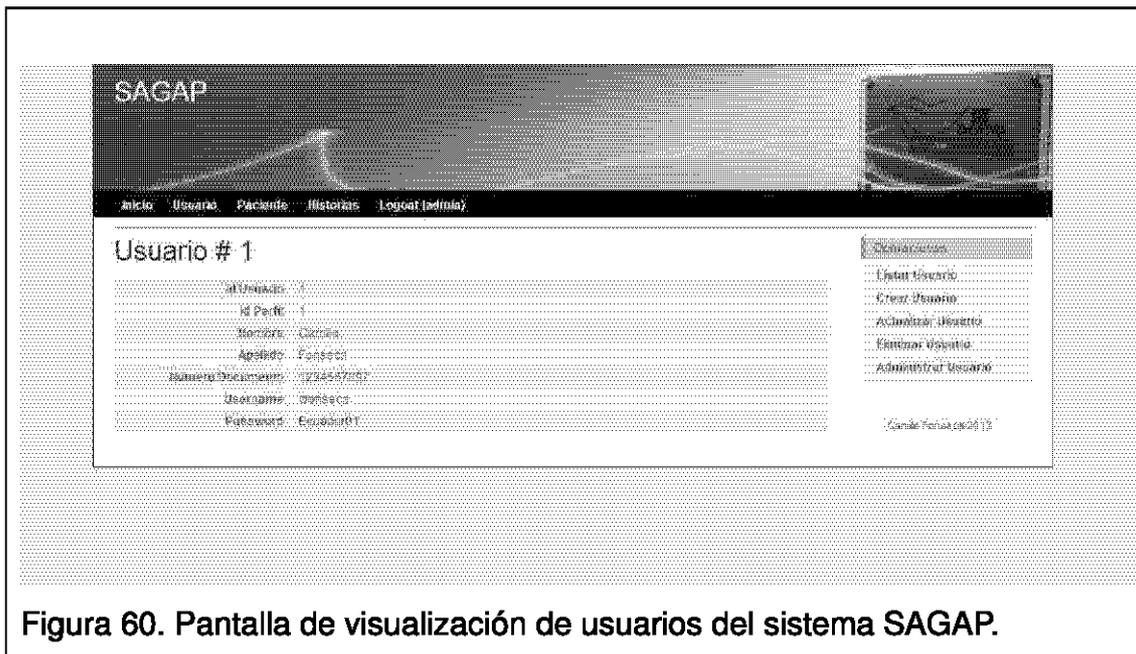


Figura 60. Pantalla de visualización de usuarios del sistema SAGAP.

ANEXO 7

Botón	Activado	Descripción
Guardar	Siempre	Permite validar la información ingresada por el administrador y actualizarla en el sistema.

Menú	Activado	Acción
Crear Usuario	Siempre	Permite la creación de un nuevo usuario.
Listar Usuario	Siempre	Permite visualizar los usuarios de acceso al sistema existentes.
Ver Usuario	Siempre	Permite ver los usuarios del sistema.
Administrar Usuario	Siempre	Permite realizar tareas de administración, creación actualización de un usuario del sistema.

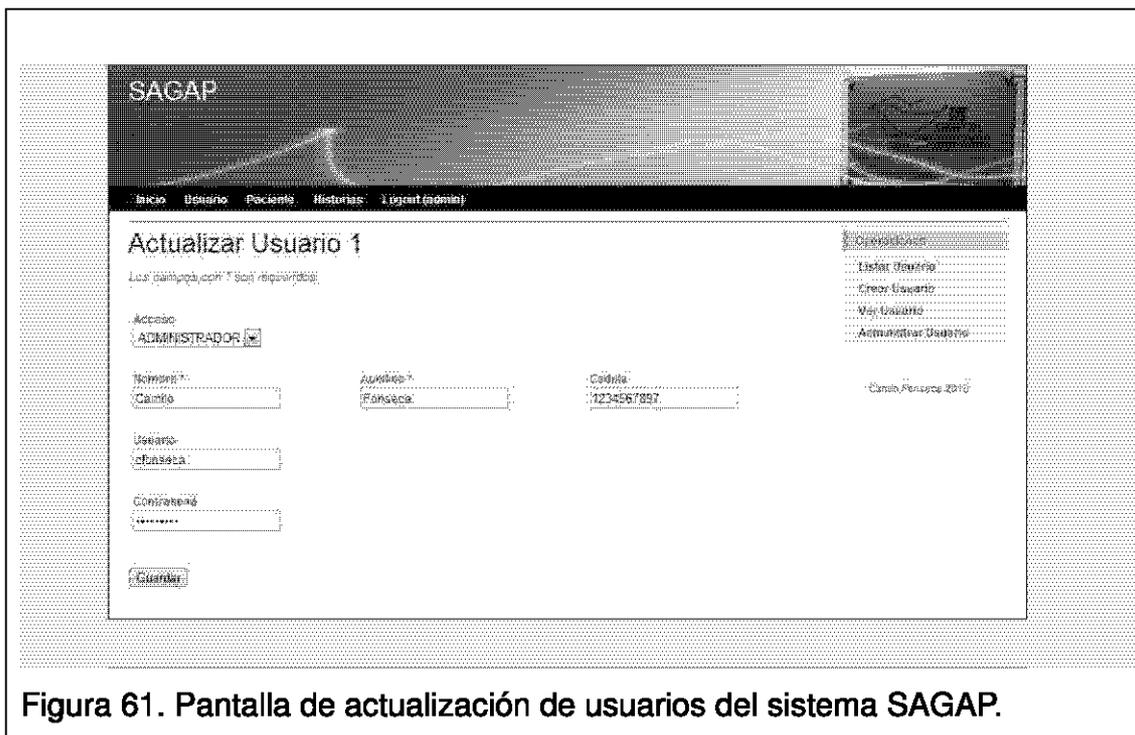


Figura 61. Pantalla de actualización de usuarios del sistema SAGAP.

ANEXO 8

Descripción

Pantalla que permite la eliminación de un paciente del sistema, antes de proceder con la eliminación el sistema pregunta si está seguro de proceder con la acción.



Figura 62. Pantalla de administración de usuarios del sistema SAGAP.

ANEXO 9

Descripción
Lista todos los pacientes ingresados en el sistema.

Menú	Activado	Acción
Crear Paciente	Siempre	Permite la creación de un nuevo paciente.
Listar Paciente	Siempre	Permite visualizar los pacientes existentes en el sistema.



Figura 63. Pantalla de visualización de pacientes del sistema SAGAP.

ANEXO 10

Botón	Activado	Descripción
Crear Paciente	Siempre	Valida la información ingresada en el formulario y crea un nuevo paciente del centro MF Dental.

Menú	Activado	Acción
Listar Paciente	Siempre	Permite visualizar los pacientes existentes en el sistema.
Administrar Paciente	Siempre	Permite realizar tareas de administración, creación actualización de un paciente del sistema.

SAGAP

Inicio Usuario Paciente Historias Login (Admin)

Crear Paciente

Los campos con * son requeridos

Nombre: Apellido Paterno: Apellido Materno:

Sexo: M F Tipo de Sangre: A B O AB Factor: Rh+ Rh-

Documento: * DNI * Sexo: M F Estado Civil: * SOLTERO * Fecha De Nacimiento: Nacionalidad:

Cuidador De Paciente: Parentesco / Estado:

Dirección:

Ocupación: Teléfono: Email:

Centro Paroquia 007

Figura 64. Pantalla de creación de pacientes del sistema SAGAP.

ANEXO 11

Descripción

Pantalla que permite realizar búsquedas de pacientes en el sistema, al mismo tiempo ofrece la posibilidad de actualizar, y eliminar la información de un paciente encontrado en el sistema.

Menú	Activado	Acción
Crear Paciente	Siempre	Permite la creación de un nuevo paciente.
Listar Paciente	Siempre	Permite visualizar los pacientes existentes en el sistema.

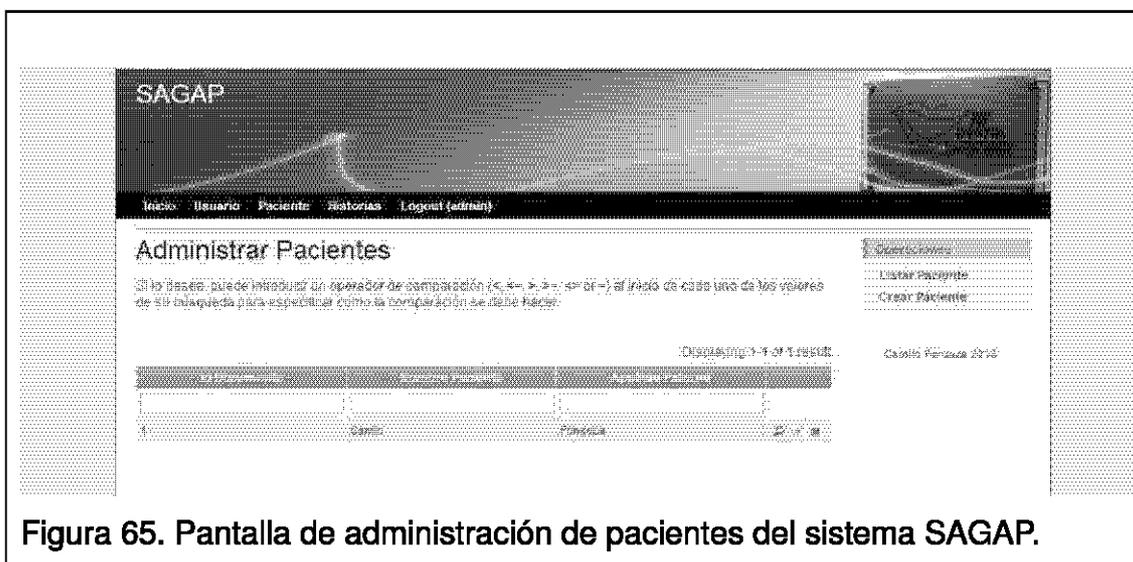


Figura 65. Pantalla de administración de pacientes del sistema SAGAP.

ANEXO 12

Descripción
Pantalla que permite la eliminación de un paciente del sistema, antes de proceder con la eliminación el sistema pregunta si está seguro de proceder con la acción.

Menú	Activado	Acción
Crear Paciente	Siempre	Permite la creación de un nuevo paciente.
Listar Paciente	Siempre	Permite visualizar los pacientes existentes en el sistema.

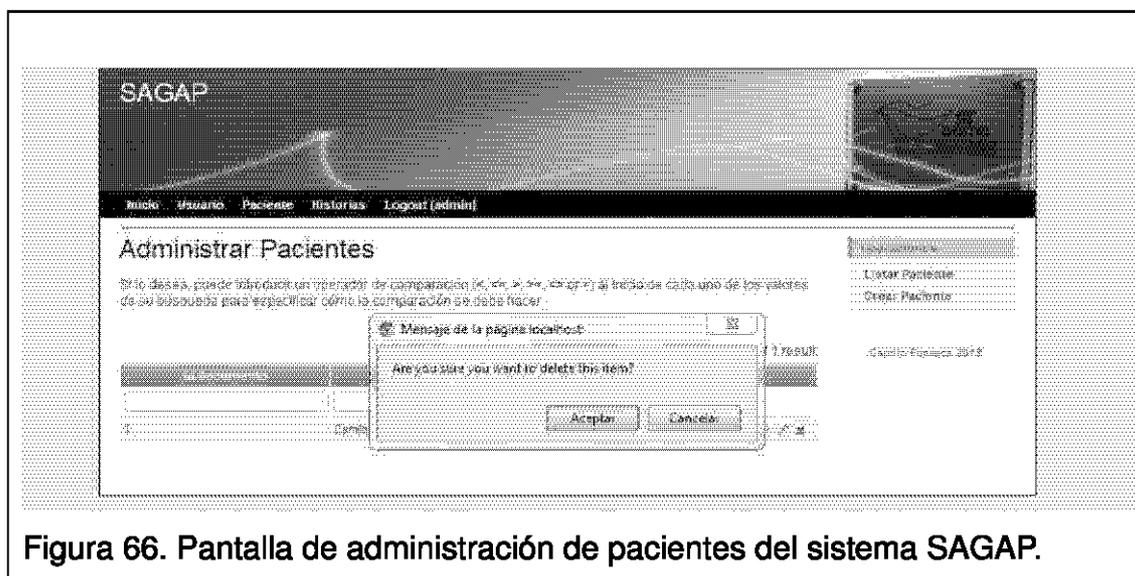


Figura 66. Pantalla de administración de pacientes del sistema SAGAP.

ANEXO 13

Descripción
Pantalla que permite visualizar un paciente en el sistema.

Menú	Activado	Acción
Crear Paciente	Siempre	Permite la creación de un nuevo paciente.
Listar Paciente	Siempre	Permite visualizar los pacientes existentes en el sistema.
Actualizar Paciente	Siempre	Permite actualizar la información de paciente.
Eliminar Paciente	Siempre	Permite eliminar un paciente del sistema.
Administrar Paciente	Siempre	Permite realizar tareas de administración, creación actualización de un paciente.

The screenshot displays the SAGAP web application interface. At the top, there is a navigation menu with links for 'Inicio', 'Usuarios', 'Paciente', 'Historias', and 'Logout (admin)'. The main content area is titled 'Paciente # 1' and contains a list of patient attributes. On the right side, there is a sidebar with several management actions and a version indicator.

Attribute	Value
Id Sexo	2
Id Estado C	2
Id Factor	1
Id Grupo	1
Id Documento	1
Nombre Paciente	Carlin
Apellido Materno	Pariona
Apellido Materno	Carozzo
Detalle Ingreso	Y 11/4 89/45
Fecha Ingreso	1996-02-23
Localidad	Esquel, Chubut
Ciudad Ingreso	Sesé
Provincia	Chubut
Organismo	Desarrollo
Ocupación	Estudiante
Telefono	029-422-4222
Email	carlin@carlin.com

Management actions in the sidebar:

- Listar Paciente
- Crear Paciente
- Actualizar Paciente
- Eliminar Paciente
- Administrar Paciente

Version: Carlin Paciente 02.12

Figura 67. Pantalla de administración de pacientes del sistema SAGAP.

ANEXO 14

Botón	Activado	Descripción
Crear	Siempre	Valida que la información ingresada en el formulario se correcta y guarda la información de una historia clínica asociada al id de un paciente..

Menú	Activado	Acción
Crear Historia	Siempre	Permite la creación de una nueva historia clínica.
Listar Historia	Siempre	Permite visualizar las historias clínicas de cada paciente en el sistema.



Figura 68. Pantalla de administración de pacientes del sistema SAGAP.

ANEXO 15

Descripción
Pantalla que permite realizar búsquedas de historias clínicas en el sistema, al mismo tiempo ofrece la posibilidad de actualizar, y eliminar la información de un paciente encontrado en el sistema.

Menú	Activado	Acción
Crear Historia	Siempre	Permite la creación de una nueva historia clínica.
Listar Historia	Siempre	Permite visualizar las historias clínicas de cada paciente en el sistema.



Figura 69. Pantalla de administración de pacientes del sistema SAGAP.

ANEXO 16

Descripción
Lista todas las historias clínicas ingresadas en el sistema.

Menú	Activado	Acción
Crear Historia	Siempre	Permite la creación de una nueva historia clínica.
Administrar Historia	Siempre	Permite realizar tareas de administración, creación actualización de una historia clínica.

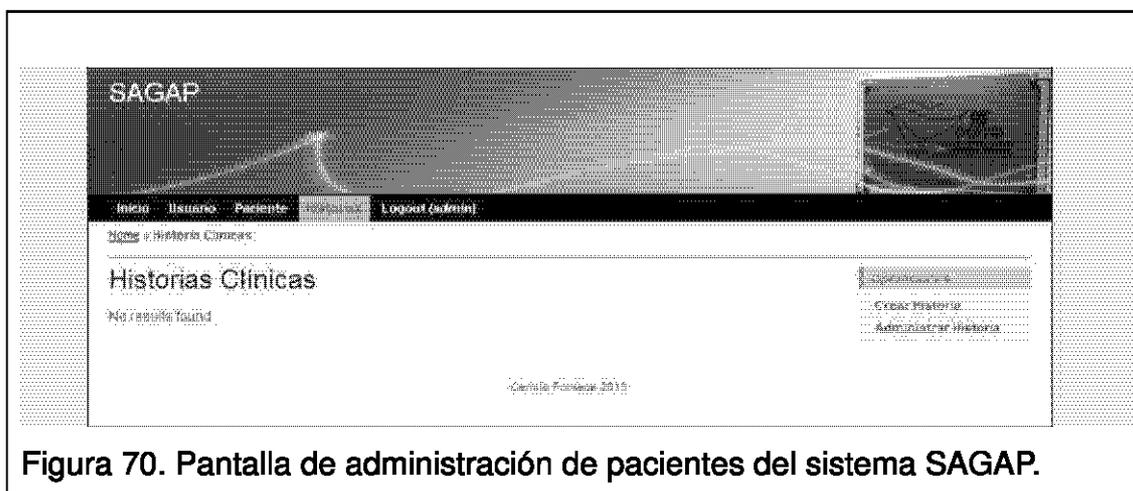


Figura 70. Pantalla de administración de pacientes del sistema SAGAP.

ANEXO 17

Descripción
Lista todas las consultas ingresadas en el sistema asociadas a un historial clínico.

Menú	Activado	Acción
Crear Consulta	Siempre	Permite la creación de una nueva consulta a la historia clínica de un paciente.
Administrar Consulta	Siempre	Permite realizar tareas de administración, creación actualización de una consulta.

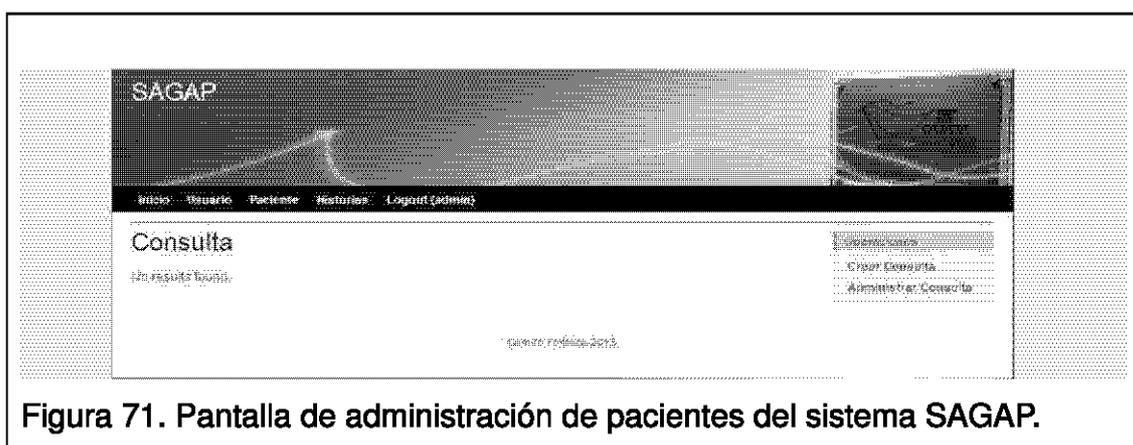


Figura 71. Pantalla de administración de pacientes del sistema SAGAP.

ANEXO 18

Botón	Activado	Descripción
Crear Consulta	Siempre	Valida la información ingresada en el sistema y crea una nueva consulta asociada a un paciente.

Menú	Activado	Acción
Crear Consulta	Siempre	Permite la creación de una nueva consulta a la historia clínica de un paciente.
Listar Consulta	Siempre	Permite visualizar las consultas asociadas a una historia clínica.

SAGAP

Inicio Usuario Paciente Historias Logout (admin)

Crear Consulta

Los pacientes con * saturación de:

Doctor:

Paciente:

Fecha:

Motivo:

Problema Actual:

Fecha siguiente consulta:

Ejecutar Plan Tratamiento

Crear Consulta

Calendario: Enero 2019

Lu	Ma	Mi	Ju	Vi	Se	Do
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

Calendario: Enero 2019

Listar Consultas

Administrar Consultas

Figura 72. Pantalla de administración de pacientes del sistema SAGAP.

ANEXO 20

Botón	Activado	Descripción
Crear Plan de Tratamiento	Siempre	Valida la información ingresada en el sistema y crea un nuevo plan de tratamiento asociado a la consulta del paciente.
Ingresar Medicamento	Siempre	Permite abrir una ventana para ingreso de medicamentos al plan de tratamiento.

Menú	Activado	Acción
Crear Plan	Siempre	Permite la creación de un nuevo plan de tratamiento asociado a la consulta de cada paciente.

The screenshot shows the SAGAP web application interface. At the top left, the logo 'SAGAP' is displayed. Below it is a navigation menu with links for 'Inicio', 'Usuario', 'Paciente', 'Historias', and 'Logout (admin)'. The main heading is 'Crear Plan de Tratamiento'. Below the heading, there is a text prompt: 'Crea un plan de tratamiento con los siguientes datos:'. To the right of this prompt, there are two buttons: 'Crear Nuevo Plan' and 'Cancelar Plan'. Below the prompt, there is a text input field labeled 'Consulta #' and a date input field labeled 'Consulta (Fecha) 2013'. Below these fields is a large text area labeled 'Indicaciones'. At the bottom left, there is a button labeled 'Ingresar Medicamento'. At the bottom center, there is a button labeled 'Crear Plan de Tratamiento'.

Figura 74. Pantalla de administración de pacientes del sistema SAGAP.

ANEXO 21

Botón	Activado	Descripción
Crear	Siempre	Valida la información ingresada en el sistema y crea un nuevo medicamento asociado a un plan de tratamiento.

Menú	Activado	Acción
Nuevo Medicamento	Siempre	Permite la creación de un nuevo medicamento asociado a un plan de tratamiento.

Ingresar Medicamento

Los campos con * son requeridos.

Nombre Medicamento*
 Concentración*
 Presentación*
 Camino Farmaco. S.I. 13:

Cantidad*
 Vía*

Término*
 Frecuencia*

Superficie

Figura 75. Pantalla de administración de pacientes del sistema SAGAP.

ANEXO 22

De la siguiente manera se guardaban los datos y se realizaba la graficación del nivel de enfermedad de cada paciente.

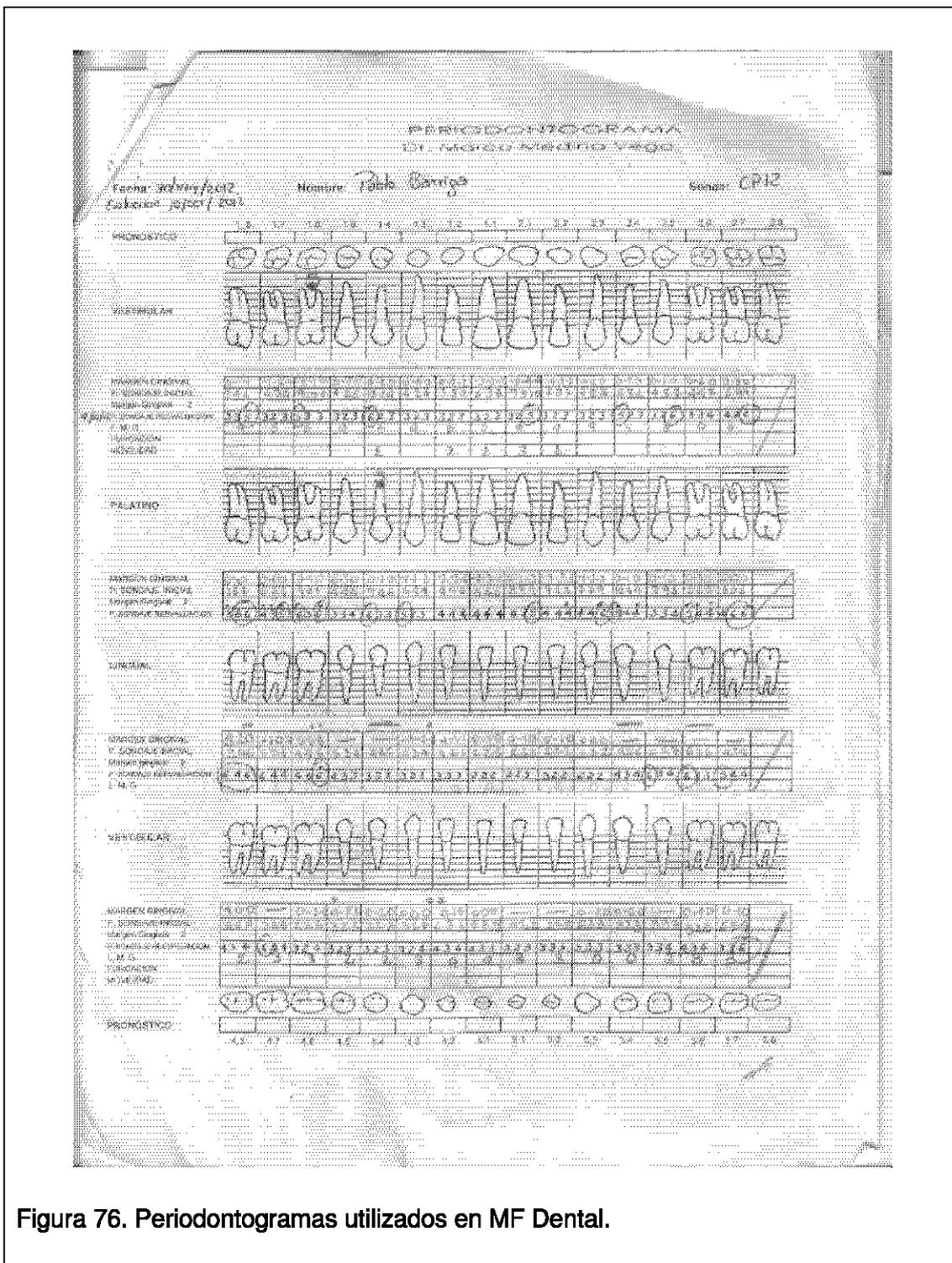


Figura 76. Periodontogramas utilizados en MF Dental.