

**FACULTAD DE INGENIERÍA**

CARRERA DE INGENIERÍA DE SISTEMAS COMPUTACIONALES

“IMPLEMENTACIÓN DE UNA APLICACIÓN SOFTWARE BASADA EN TECNOLOGÍAS MÓVIL Y WEB PARA MEJORAR EL ACCESO A SERVICIOS DE SALUD DE LOS HABITANTES DEL DISTRITO DE TRUJILLO”

Tesis para optar el título profesional de:

**Ingeniero de Sistemas Computacionales**

**Autor(es):**

Br. Jerry Randy Guzmán Valderrama

Br. Eder Martin Haro Araujo

**Asesor:**

Ing. Alberto Carlos Mendoza de los Santos

**Trujillo – Perú**

**2016**

APROBACIÓN DE LA TESIS

El asesor y los miembros del jurado evaluador asignados, **APRUEBAN** la tesis desarrollada por el Bachiller **Jerry Randy Guzmán Valderrama** y el Bachiller **Eder Martin Haro Araujo**, denominada:

**“IMPLEMENTACIÓN DE UNA APLICACIÓN SOFTWARE BASADA EN TECNOLOGÍAS MÓVIL Y WEB PARA MEJORAR EL ACCESO A SERVICIOS DE SALUD DE LOS HABITANTES DEL DISTRITO DE TRUJILLO”**

|  |
| --- |
| Ing. Alberto Carlos Mendoza de los Santos  **ASESOR** |

|  |
| --- |
| Ing. Bady Cruz Díaz  **JURADO**  **PRESIDENTE** |

|  |
| --- |
| Ing. Lourdes Roxana Díaz Amaya  **JURADO** |

|  |
| --- |
| Ing. Lain Jardiel Cardenas Escalante  **JURADO** |

DEDICATORIA

*Esta tesis se la dedicamos nuestros padres porque son nuestra fuente de inspiración en todo lo que hacemos.*

*A nuestros familiares y amigos que siempre esperaron lo mejor de nosotros.*

*A nuestros maestros de la facultad que se esforzaron por formarnos como buenos profesionales.*

AGRADECIMIENTO

*Queremos agradecer a nuestros padres por todos los esfuerzos que hicieron para educarnos y por motivarnos a alcanzar nuestras metas.*

*Así también, agradecer a nuestro asesor por sus recomendaciones y consejos los cuales sirvieron para guiar este proyecto por buen camino.*

*Agradecemos también, a los docentes de la facultad de ingeniería de sistemas computacionales por sus enseñanzas en nuestra etapa universitaria ya que estas sirvieron de base para la elaboración de esta tesis.*

*A los jurados, por sus observaciones, que sirvieron para mejorar esta investigación.*

RESUMEN

Los sistemas de salud ejercen una influencia determinante en la salud y la vida de las personas. Asimismo tienen la responsabilidad no solo de mejorar la salud y responder a las expectativas de la población sino de protegerlas contra las pérdidas perdidas propias de la enfermedad.

El acceso a los servicios de salud, se logra mediante la plena satisfacción de una necesidad de atención percibida. Sin embargo entre aquella necesidad de salud y la atención de la misma suelen presentarse múltiples barreras. Algunas de tipo económica, física o psicológica y otras de carácter administrativo producto de la organización del sistema de salud, sus niveles de atención, sectorización, cobertura y costo.

Este proyecto de tesis se presenta como una solución informática para mejorar el acceso a servicios de salud. Para ello, se implementaron dos aplicaciones móviles que permitirán interactuar a pacientes y médicos con la finalidad de atender consultas, asignar citas médicas, acceder a ubicaciones de centros médicos e intercambiar información sobre temas de salud. Por otro lado se implementó una plataforma web para la administración de los datos tanto de médicos como pacientes.

La investigación está enmarcada en un ámbito pre experimental. Se tomará como muestra un sector de la población del distrito de Trujillo y luego se le aplicará una pre y post prueba para determinar la validez de la hipótesis.

En la construcción del software se escogió como marco de desarrollo la metodología ágil Scrum. Las tecnologías empleadas en la aplicación web fueron Java Enterprise Edition junto con un servidor de aplicaciones Glassfish para el despliegue de sus componentes. Las aplicaciones móviles fueron desarrolladas para ejecutarse en la plataforma Android y consumirán datos por medio de un web service implementado en la parte web.

ABSTRACT

ÍNDICE DE CONTENIDOS

[APROBACIÓN DE LA TESIS ii](#_Toc387424847)

[DEDICATORIA iii](#_Toc387424848)

[AGRADECIMIENTO iv](#_Toc387424849)

[RESUMEN v](#_Toc387424850)

[ABSTRACT vi](#_Toc387424851)

[ÍNDICE DE CONTENIDOS vii](#_Toc387424852)

[ÍNDICE DE TABLAS ix](#_Toc387424853)

[ÍNDICE DE GRÁFICOS x](#_Toc387424854)

[CAPÍTULO 1. INTRODUCCIÓN 1](#_Toc387424855)

[1.1. Realidad problemática 1](#_Toc387424856)

[1.2. Formulación del problema 1](#_Toc387424857)

[1.3. Justificación 1](#_Toc387424858)

[1.4. Limitaciones 2](#_Toc387424859)

[1.5. Objetivos 2](#_Toc387424860)

[1.5.1. Objetivo General 2](#_Toc387424861)

[1.5.2. Objetivos Específicos 2](#_Toc387424862)

[CAPÍTULO 2. MARCO TEÓRICO 3](#_Toc387424863)

[2.1. Antecedentes 3](#_Toc387424864)

[2.2. Bases Teóricas 3](#_Toc387424865)

[2.3. Definición de términos básicos 3](#_Toc387424866)

[CAPÍTULO 3. HIPÓTESIS 4](#_Toc387424867)

[3.1. Formulación de la hipótesis 4](#_Toc387424868)

[3.2. Operacionalización de variables 4](#_Toc387424869)

[CAPÍTULO 4. PROPUESTA DE APLICACIÓN PROFESIONAL 8](#_Toc387424870)

[CAPÍTULO 5. MATERIALES Y MÉTODOS 9](#_Toc387424871)

[5.1. Tipo de diseño de investigación. 9](#_Toc387424872)

[5.2. Material de estudio. 9](#_Toc387424873)

[5.2.1. Unidad de estudio. 9](#_Toc387424874)

[5.2.2. Población. 9](#_Toc387424875)

[5.2.3. Muestra. 9](#_Toc387424876)

[5.3. Técnicas, procedimientos e instrumentos. 10](#_Toc387424877)

[5.3.1. Para recolectar datos. 10](#_Toc387424878)

[5.3.2. Para procesar datos. 10](#_Toc387424879)

[CAPÍTULO 6. RESULTADOS 11](#_Toc387424880)

[CAPÍTULO 7. DISCUSIÓN 12](#_Toc387424881)

[CONCLUSIONES 13](#_Toc387424882)

[RECOMENDACIONES 14](#_Toc387424883)

[REFERENCIAS 15](#_Toc387424884)

[ANEXOS 16](#_Toc387424885)

ÍNDICE DE TABLAS

ÍNDICE DE GRÁFICOS

# INTRODUCCIÓN

## Realidad problemática

De acuerdo a un estudio realizado por el Ministerio de Salud y la Dirección General de Epidemiologia del Perú, la demanda por servicios de salud se basa en el autorreporte de las personas de identificarse como enfermas. Esta decisión se considera que es selectiva y varía según la edad del individuo o de acuerdo a las condiciones de vida. Sin embargo, también se ve influenciado por diversos factores socioeconómicos, entre ellos, el nivel educativo, la situación de pobreza, la condición de asegurado, el ámbito geográfico del que procede, etc.

En el año 2011, el 69.5% de la población peruana autorreportó que en las últimas 4 semanas, tuvo algún problema de salud, cifra que se ha incrementado en los últimos años (2005: 54.1%; 2008: 63.0%; 2010: 68.0%) (Gráfico 1).

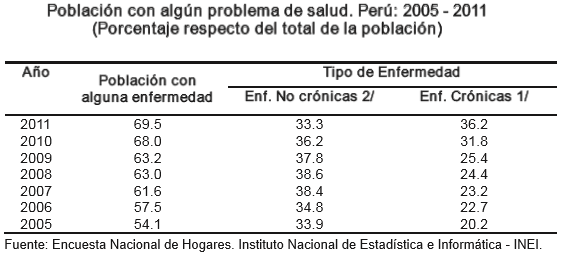


Gráfico . Población con algún problema de salud. Perú: 2005 – 2011(Porcentaje respecto del total de la población). *(Ministerio de Salud del Peru y Direccion General de Epidemiologia, 2013)*

Asimismo, se observa que del total de la población que declaró haber presentado algún problema de salud, el 56.9% realizó una consulta para aliviar el mal que le aquejaba. Este porcentaje de consulta fue mayor cuando el problema de salud no fue de carácter crónico (63.8%), en comparación para las enfermedades crónicas (49.3%)(Gráfico 2).

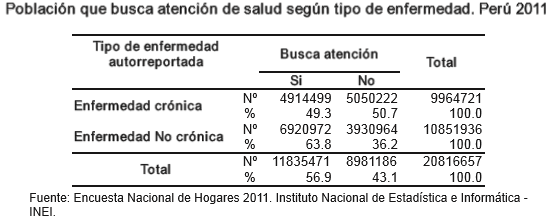


Gráfico . Población que busca atención de salud según tipo de enfermedad. Perú 2011. *(Ministerio de Salud del Peru y Direccion General de Epidemiologia, 2013).*

En las macro regiones se observa que se mantienen las diferencias entre los ámbitos urbanos y rurales sobre todo en las regiones de la sierra y selva, tanto en las zonas norte, centro y sur. Mientras que en la región de la costa, no se perciben diferencias marcadas entre las zonas urbanas y rurales (Gráfico 3).

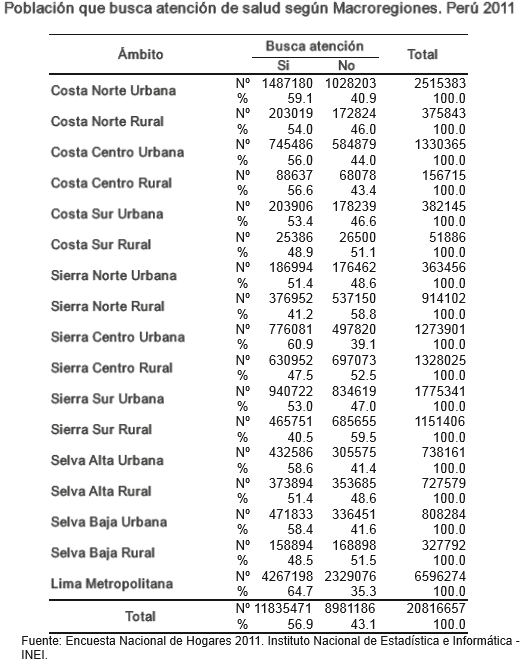


Gráfico . Población de que busca atención de salud según Macroregiones. Perú 2011. *(Ministerio de Salud del Peru y Direccion General de Epidemiologia, 2013)*

Al analizar las tendencias de cada una de las razones por la que las personas que autorreportaron enfermedad, no fueron en busca de atención en los últimos 8 años, se observa un incremento sostenido en la razón “no fue necesario”, y “falta de tiempo”. Así, la “falta de tiempo” subió de 9 % en el 2004 a 18.1% en el 2011 y el “no fue necesario” subió de 34.3% a 40.5%.

También se incrementaron las razones relacionadas a la dimensión de la calidad de atención, como el maltrato del personal, falta de confianza y demora en la atención, que en el año 2004 representó el 8.3% pasó al 13.6% en el 2011.

La falta de dinero como una justificación para no buscar atención, ha ido disminuyendo en los últimos años. Así, en el 2004 representó el 23.7%, bajando al 12.9% en el año 2011, lo que significa una reducción en los últimos 8 años del 11.8%.

Similar comportamiento se observó en el uso de remedios caseros o automedicación como razón para no buscar atención, la cual disminuyo de 42.5% en el 2004 a 28.8% en el 2011 (Gráfico 4).

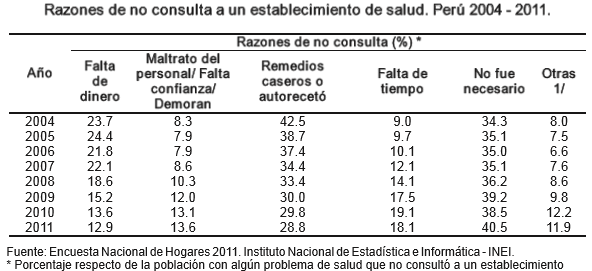


Gráfico . Razones de no consulta a un establecimiento de salud. Peru 2004 – 2011. *(Ministerio de Salud del Peru y Direccion General de Epidemiologia, 2013)*

Por otro lado, el uso de dispositivos y aplicaciones móviles se ha venido incrementando en los últimos años, siendo Android el sistema operativo móvil con mayor presencia en el Perú. Según estadísticas de StartCounter, Android mantiene una tendencia creciente en su participación de mercado de sistemas operativos móviles. En junio del 2012, la participación de Android era de un 49% mostrando un crecimiento continuo en los años posteriores.

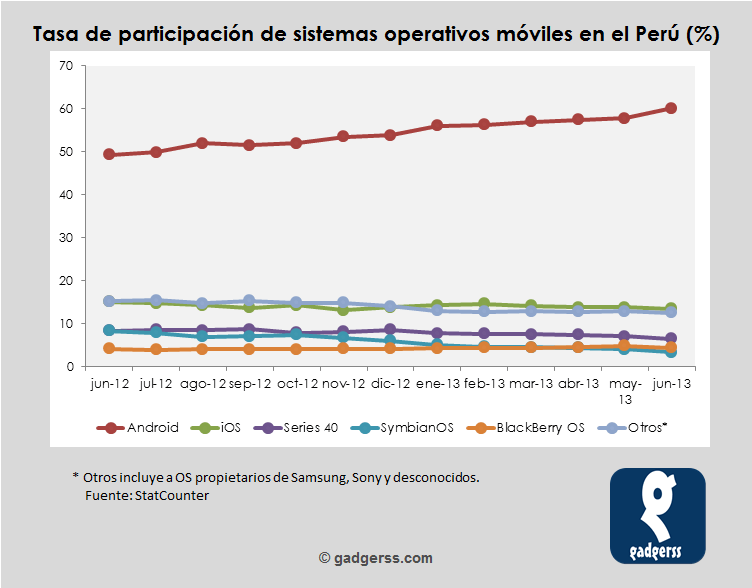


Gráfico . Tasa de partipación de sistemas operativos moviles en el Perú (%)

Asimismo, Android domina el mercado peruano a lo largo de las diferentes categorías de contenido (Gráfico 6).



Gráfico . Participación de visitantes únicos en Smartphones de las principales categorías por sistema Operativo.

De acuerdo a las estadísticas mostradas se puede identificar una creciente demanda por servicios de salud que no es atendida satisfactoriamente por factores de tipo económico, físico o psicológico y otros de carácter administrativo producto de la organización del sistema de salud, sus niveles de atención, sectorización, cobertura y costo. Aprovechando la tendencia móvil que se viene dando en el Perú con el uso de dispositivos con sistema operativo Android y con el fin de mejorar el acceso a servicios de salud, en la presente tesis se pretende implementar una aplicación software basada en tecnologías móvil y web que permita a los habitantes del distrito de Trujillo tener acceso a consultas médicas primarias basadas en síntomas e imágenes, búsqueda de información de proveedores de salud del medio, reserva de citas médicas con médicos de su preferencia e información sobre temas de salud.

## Formulación del problema

¿De qué manera la implementación de una aplicación software basada en tecnologías móvil y web mejora el acceso a servicios de salud de los habitantes del distrito de Trujillo?

## Justificación

El presente proyecto de investigación se justifica de manera aplicativa porque permitirá solucionar el problema actual que tienen los habitantes del Distrito de Trujillo para acceder a servicios de salud como atención de consultas, reserva de citas médicas, búsqueda de información acerca de temas de salud y búsqueda de proveedores de salud que en consecuencia generan insatisfacción de la demanda de estos servicios. Asimismo se justifica de manera académica porque se aplicarán los conocimientos aprendidos en la carrera de ingeniería de sistemas computacionales basados en tecnologías web y móvil para el desarrollo de la solución. Por último presenta valor y relevancia social, ya que se tiene como fin contribuir en mantener una sociedad informada en cuanto a temas de salud.

## Limitaciones

* Mantener la simplicidad en el código fuente requiere trabajo adicional. Bajo la presión de fechas de entrega es posible que se carezca de tiempo para realizar las simplificaciones deseables a la aplicación. A pesar de eso, el código fuente estará abierto a modificaciones y mantenimientos posteriores.
* Debido a que no se contará con la participación de médicos especialistas, la plataforma móvil diseñada para los pacientes contará con datos de prueba al ser presentada a las personas que serán encuestadas.

## Objetivos

### Objetivo General

Mejorar el acceso a servicios de salud de los habitantes del distrito de Trujillo mediante la implementación de una aplicación software basada en tecnologías móvil y web.

### Objetivos Específicos

* Implementar una aplicación software basada en tecnologías móvil y web que garantice la usabilidad, fiabilidad, mantenibilidad y funcionalidad.
* Incrementar el nivel de acceso de los habitantes a información de proveedores de salud.
* Incrementar el nivel de acceso de los habitantes a consultas médicas.
* Incrementar el nivel de acceso de los habitantes a reserva de citas médicas.
* Incrementar el nivel de acceso de los habitantes a información sobre temas de salud.

# MARCO TEÓRICO

## Antecedentes

Se consideraron aplicaciones móvil y web que brindan acceso a servicios de salud en mercados nacionales e internacionales.

### Second Options

Es una aplicación móvil posicionada actualmente en los Estados Unidos, para personas con dominio del idioma Ingles que cuenten con un dispositivo con sistema operativo Android, iOS o acceso a la página web oficial de la aplicación. Esta aplicación permite realizar consultas por especialidad por medio escrito o a través de un audio grabado, coordina una reunión en tiempo real con alguno de los especialistas, muestra una lista de los especialistas disponibles, ofrece la venta de distintos servicios médicos en establecimientos asociados y ofrece servicios de subscripción al servicio por planes, los cuales obtienen diferentes beneficios según el plan seleccionado.

### Healthcare Magic

Es una aplicación móvil actualmente posicionada en la India para pacientes que cuenten con un dispositivo con sistema operativo Android, iOS o acceso a la página web oficial de la aplicación. Esta aplicación permite realizar consultas por especialidad, la respuesta puede ser recepcionada vía e-mail sin necesidad de subscripción, muestra algunas de las últimas consultas realizadas por los usuarios y la respectiva respuesta, para obtener una respuesta es necesario pagar por adelantado.

### iDoc

Es una aplicación móvil posicionada actualmente en Estados Unidos, para personas con dominio del idioma Ingles que cuenten con un dispositivo con sistema operativo Android, iOS o acceso a la página web oficial de la aplicación. Esta aplicación ofrece la posibilidad de realizar consultas médicas de medicina general, información acerca de las especialidades médicas, tips diarios sobre cuidados médicos y la opción de ver a doctores disponibles para responder las consultas.

### Médicos Perú

Es una aplicación desarrollada en plataforma web, posicionada actualmente en el Perú. Ofrece la posibilidad de realizar consultas médicas de medicina general y en algunas especialidades, cuenta con un directorio médico, con foros de artículos médicos publicados por doctores y tiene un apartado para tests y calculadoras.

Las funcionalidades de las aplicaciones mencionadas se tomaran en cuenta para la implementación y evolución de la aplicación software.

## Bases Teóricas

### Demanda y Acceso de Servicios de Salud

La demanda por servicios de salud está asociada a una decisión secuencial por parte de los individuos; en una primera etapa, se decide si se busca o no servicios de salud, la cual está condicionada a que se perciba un síntoma de enfermedad, se sufra un accidente o se requiera de controles preventivos (prenatal, dental, etc). En una segunda etapa, se opta por elegir a un tipo de proveedor de servicios de salud (público o a un servicio privado). Las dos primeras etapas están asociadas al concepto de demanda de acceso, el cual determina la cantidad de demandantes de servicios o la probabilidad de que un individuo acceda a un determinado servicio de salud.

El acceso a los servicios de salud, se logra mediante la plena satisfacción de una necesidad de atención percibida. Sin embargo también se sabe que entre aquella necesidad de salud y la atención de la misma suelen presentarse múltiples barreras. Algunas de tipo económica, física o psicológica y otras de carácter administrativo producto de la organización del sistema de salud, sus niveles de atención, sectorización, cobertura y costo. *(Ministerio de Salud del Peru y Direccion General de Epidemiologia, 2013)*

### Android

Android es una plataforma de código abierto basada en Linux y promovida por Google. Así como un framework de desarrollo para crear aplicaciones usando una mezcla de Java y XML que además permite desplegarlas en una amplia variedad de dispositivos móviles, tablets y más.

Android está compuesto por diferentes componentes. Esto incluye aplicaciones básicas como Contactos, Calendario, Mapas y un Navegador, APIs que permiten controlar la apariencia y comportamiento de las aplicaciones. Android Runtime que viene con un conjunto de librerías que implementan la mayor parte del lenguaje de programación Java, cada aplicación Android se ejecuta en su propio proceso. Debajo del framework de aplicación se encuentran un conjunto de librerías C y C++ y debajo de todo se encuentra el kernel Linux, Android se basa en el kernel para drivers, así como para servicios básicos de seguridad y administración de memoria (Gráfico 7).

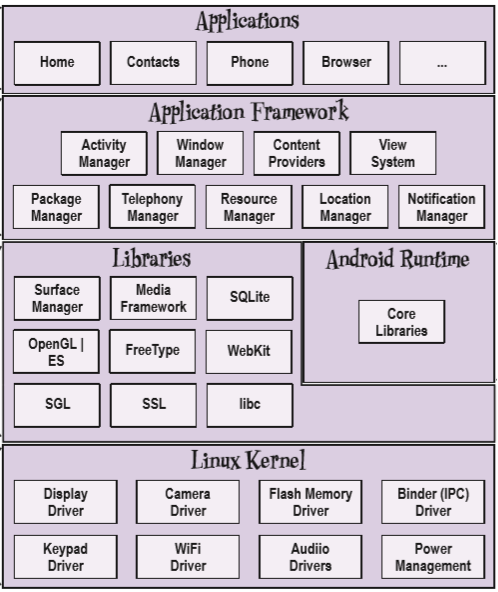


Gráfico . Componentes de Android (Griffiths, 2015)

### SCRUM

* **Manifiesto Ágil**

En marzo de 2001, 17 críticos de los modelos de producción basados en procesos, convocados por Kent Beck, que había publicado un par de años antes el libro en el que explicaba la nueva metodología Extreme Programming se reunieron en Salt Lake City para discutir sobre el desarrollo de software. En la reunión se acuñó el término “Métodos Ágiles” para definir a aquellos que estaban surgiendo como alternativa a las metodologías formales: CMM-SW, (precursor de CMMI) PMI, SPICE (proyecto inicial de ISO 15504), a las que consideraban excesivamente “pesadas” y rígidas por su carácter normativo y fuerte dependencia de planificaciones detalladas, previas al desarrollo.

Los integrantes de la reunión resumieron en cuatro postulados lo que ha quedado denominado como “Manifiesto Ágil”, que son los valores sobre los que se asientan estos métodos.

*“Estamos poniendo al descubierto mejores métodos para desarrollar software, haciéndolo y ayudando a otros a que lo hagan.*

*Con este trabajo hemos llegado a valorar:*

* *A los individuos y su interacción, por encima de los procesos y las herramientas.*
* *El software que funciona, por encima de la documentación exhaustiva.*
* *La colaboración con el cliente, por encima de la negociación contractual.*
* *La respuesta al cambio, por encima del seguimiento de un plan.*

*Aunque hay valor en los elementos de la derecha, valoramos más los de la izquierda.”*

El manifiesto ágil, tras los postulados de estos cuatro valores en los que se fundamenta, establece 12 principios:

* *Nuestra principal prioridad es satisfacer al cliente a través de la entrega temprana y continua de software de valor.*
* *Son bienvenidos los requisitos cambiantes, incluso si llegan tarde al desarrollo. Los procesos ágiles se doblegan al cambio como ventaja competitiva para el cliente.*
* *Entregar con frecuencia software que funcione, en periodos de un par de semanas hasta un par de meses, con preferencia en los periodos breves.*
* *Las personas del negocio y los desarrolladores deben trabajar juntos de forma cotidiana a través del proyecto.*
* *Construcción de proyectos en torno a individuos motivados, dándoles la oportunidad y el respaldo que necesitan y procurándoles confianza para que realicen la tarea.*
* *La forma más eficiente y efectiva de comunicar información de ida y vuelta dentro de un equipo de desarrollo es mediante la conversación cara a cara.*
* *El software que funciona es la principal medida del progreso.*
* *Los procesos ágiles promueven el desarrollo sostenido. Los patrocinadores, desarrolladores y usuarios deben mantener un ritmo constante de forma indefinida.*
* *La atención continua a la excelencia técnica enaltece la agilidad.*
* *La simplicidad como arte de maximizar la cantidad de trabajo que se hace, es esencial.*
* *Las mejores arquitecturas, requisitos y diseños emergen de equipos que se autoorganizan.*
* *En intervalos regulares, el equipo reflexiona sobre la forma de ser más efectivo y ajusta su conducta en consecuencia.*
* **Origen de Scrum**

Scrum es un modelo de desarrollo ágil caracterizado por:

* + Adoptar una estrategia de desarrollo incremental, en lugar de la planificación y ejecución completa del producto.
  + Basar la calidad del resultado más en el conocimiento tácito de las personas en equipos auto organizados, que en la calidad de los procesos empleados.
  + Solapamiento de las diferentes fases del desarrollo, en lugar de realizar una tras otra en un ciclo secuencial o de cascada.

Scrum fue identificado y definido por Ikujiro Nonaka e Hirotaka Takeuchi a principios de los 80, al analizar cómo desarrollaban los nuevos productos las principales empresas de manufactura tecnológica: Fuji-Xerox, Canon, Honda, Nec, Epson, Brother, 3M y Hewlett-Packard (Nonaka & Takeuchi, The New New Product Development Game, 1986).

En su estudio, Nonaka y Takeuchi compararon la nueva forma de trabajo en equipo, con el avance en formación de scrum de los jugadores de Rugby, a raíz de lo cual quedó acuñado el término “scrum” para referirse a ella.

Aunque esta forma de trabajo surgió en empresas de productos tecnológicos, es apropiada para proyectos con requisitos inestables y para los que requieren rapidez y flexibilidad, situaciones frecuentes en el desarrollo de determinados sistemas de software.

Scrum se puede adoptar de forma técnica, aplicando reglas definidas, o pragmática, adoptando los valores originales scrum con reglas personalizadas. Para el desarrollo de este proyecto de tesis se usara Scrum técnico.

Scrum técnico está formado por un conjunto de prácticas y reglas que resultan válidos para dar respuesta a los siguientes principios de desarrollo ágil:

* Gestión evolutiva del avance, en lugar de la tradicional o predictiva.
* Trabajar basando la calidad del resultado en el conocimiento tácito de las personas, más que en el explícito de los procesos y la tecnología empleada.
* Estrategia de desarrollo incremental a través de iteraciones (sprints) y revisiones.
* Seguir los pasos del desarrollo ágil: desde el concepto o visión general de la necesidad del cliente, construcción del producto de forma incremental a través de iteraciones breves que comprenden fases de especulación, exploración y revisión. Estas iteraciones (en scrum llamadas sprints) se repiten de forma continua hasta que el cliente da por cerrada la evolución del producto.

Se comienza con la visión general de lo que se desea obtener, y a partir de ella se especifica y da detalle a las partes de mayor prioridad, y que se desean tener cuanto antes. Cada ciclo de desarrollo o iteración (sprint) finaliza con la entrega de una parte operativa del producto (incremento). La duración de cada sprint puede ser desde una, hasta seis semanas, aunque se recomienda que no excedan de un mes. En scrum, el equipo monitoriza la evolución de cada sprint en reuniones breves diarias donde se revisa en conjunto el trabajo realizado por cada miembro el día anterior, y el previsto para el día en curso. Esta reunión diaria es de tiempo prefijado de 5 a 15 minutos máximo, se realiza de pie junto a un tablero o pizarra con información de las tareas del sprint, y el trabajo pendiente en cada una. Esta reunión se denomina “reunion de pie” o “scrum diario” y si se emplea la terminología inglesa: “stand-up meeting”, también: “daily scrum” o “morning rollcall”.

#### Gestión de la evolución del proyecto

* **Revisión de iteraciones**

Al finalizar cada sprint se revisa funcionalmente el resultado, con todos los implicados en el proyecto. Es por tanto la duración del sprint, el período de tiempo máximo para descubrir planteamientos erróneos, mejorables o malinterpretaciones en las funcionalidades del producto.

* **Desarrollo Incremental**

No se trabaja con diseños o abstracciones durante toda la construcción del producto. El desarrollo incremental ofrece al final de cada iteración una parte de producto operativa, que se puede usar, inspeccionar y evaluar. Scrum resulta adecuado en proyectos con requisitos inciertos y, o inestables. ¿Por qué predecir la versión definitiva de algo que va a estar evolucionando de forma continua? scrum considera a la inestabilidad como una premisa, y adopta técnicas de trabajo para facilitar la evolución sin degradar la calidad de la arquitectura y permitir que también evolucione durante el desarrollo. Durante la construcción se depura el diseño y la arquitectura, y no se cierran en una primera fase del proyecto. Las distintas fases que el desarrollo en cascada realiza de forma secuencial, en scrum se solapan y realizan de forma continua y simultánea.

* **Autoorganización**

Son muchos los factores impredecibles en un proyecto. La gestión predictiva asigna al rol de gestor del proyecto la responsabilidad de su gestión y resolución. En scrum los equipos son autoorganizados, con un margen de maniobra suficiente para tomar las decisiones que consideren oportunas.

* **Colaboración**

Es un componente importante y necesario para que a través de la autoorganización se pueda gestionar con solvencia la labor que de otra forma realizaría un gestor de proyectos. Todos los miembros del equipo colaboran de forma abierta con los demás, según sus capacidades y no según su rol o su puesto.

El marco de Scrum técnico está formado por:

* **Roles:** 
  + **El equipo Scrum**

Grupo o grupos de trabajo que desarrollan el producto.

* + **El dueño del producto**

Es la persona responsable de lograr el mayor valor de producto para los clientes, usuarios y resto de implicados.

* + **El Scrum Master**

Proporciona la asesoría y formación necesaria al propietario del producto y al equipo.

* **Artefactos:** 
  + **Pila del producto**

Lista de requisitos de usuario, que a partir de la visión inicial del producto crece y evoluciona durante el desarrollo.

* + **Pila del sprint**

Lista de los trabajos que debe realizar el equipo durante el sprint para generar el incremento previsto.

* + **Sprint**

Nombre que recibe cada iteración de desarrollo. Es el núcleo central que genera el pulso de avance por tiempos prefijados.

* + **Incremento**

Resultado de cada Sprint.

* **Eventos** 
  + **Reunión de planificación del sprint**

Reunión de trabajo previa al inicio de cada sprint en la que se determina cuál va a ser el objetivo del sprint y las tareas necesarias para conseguirlo.

* + **Scrum diario**

Breve reunión diaria del equipo en la que cada miembro responde a tres cuestiones: el trabajo realizado el día anterior, el que tiene previsto realizar y cosas que puede necesitar o impedimentos que deben eliminarse para poder realizar el trabajo.

* + **Revisión del sprint**

Análisis e inspección del incremento generado, y adaptación de la pila del producto si resulta necesario.

* + **Retrospectiva del sprint**

Revisión de lo sucedido durante el Sprint. Reunión en la que el equipo analiza aspectos operativos de la forma de trabajo y crea un plan de mejoras para aplicar en el próximo sprint.

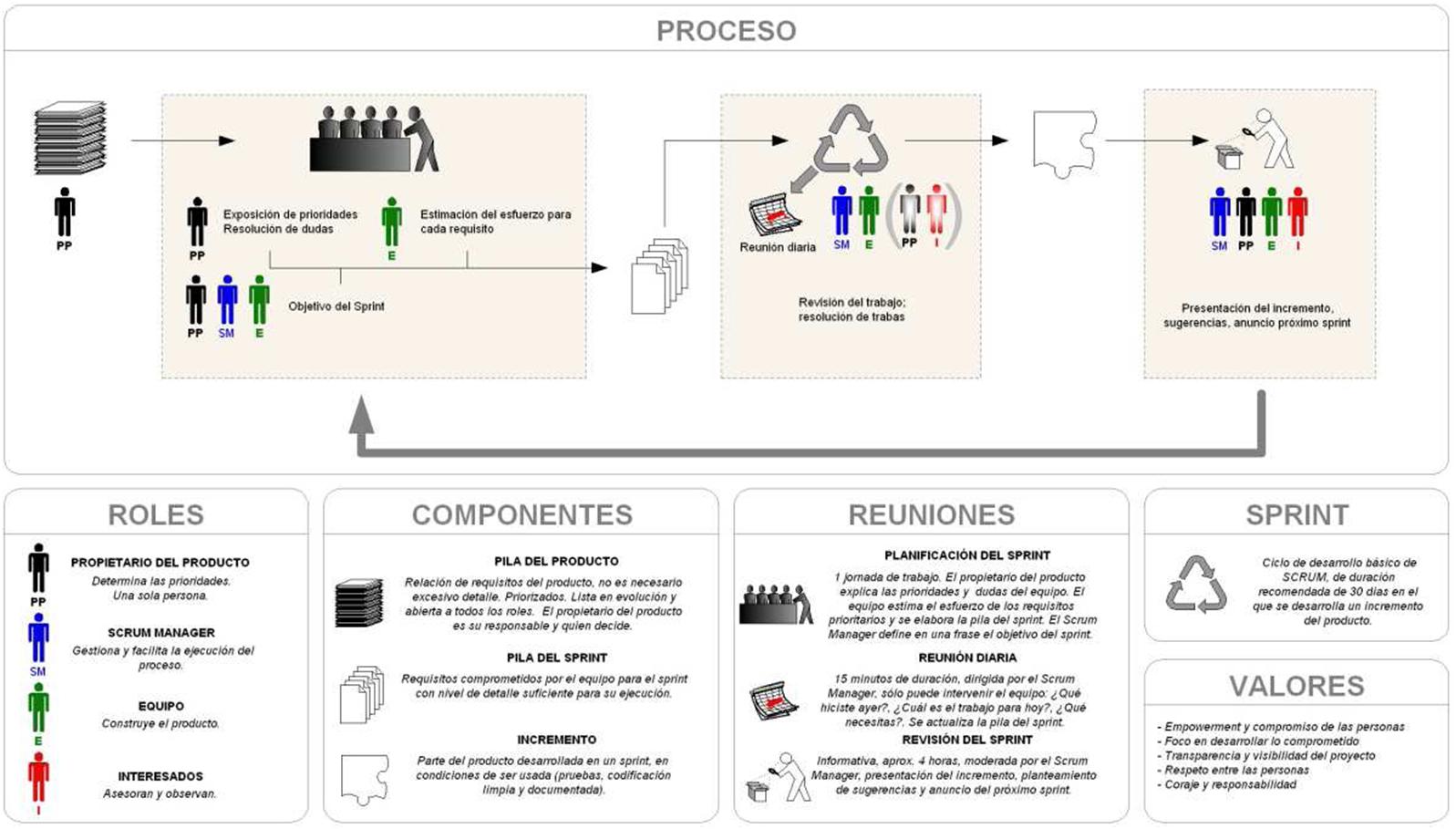


Gráfico . Marco Scrum Técnico (Palacio, 2014).

### Java Platform, Enterprise Edition

#### Introducción

Java Platform, Enterprise Edition o Java EE (Oracle, s.f.) es una plataforma de programación que permite el desarrollo y ejecución de aplicaciones diseñadas en arquitecturas multicapa o n-capas en un entorno distribuido, y está desarrollada por Java Community Process (JPC), la cual es responsable de todas las tecnologías de Java.

El modelo de aplicación de Java EE define una arquitectura para la implementación de servicios como aplicaciones multicapa que brindan escalabilidad, accesibilidad y capacidad de gestión. Este modelo divide el trabajo en:

* La presentación y lógica del negocio para ser implementada por el desarrollador.
* Los estándares de servicios del sistema proporcionados por la plataforma Java EE.

La plataforma Java EE usa un modelo de aplicación multinivel distribuida en aplicaciones empresariales. La lógica de la aplicación está dividida en componentes de acuerdo a su función, y los componentes de aplicación que conforman Java EE son instalados en diferentes máquinas de acuerdo al nivel o capa al cual pertenecen.

El Gráfico 9 muestra dos aplicaciones Java EE multicapa divididas en:

* Componentes del nivel-cliente ejecutados en la maquina cliente.
* Componentes de nivel-web ejecutados en el servidor Java EE.
* Componentes del nivel-negocio ejecutados en el servidor Java EE.
* El nivel de sistema de información empresarial se ejecuta en el servidor EIS.

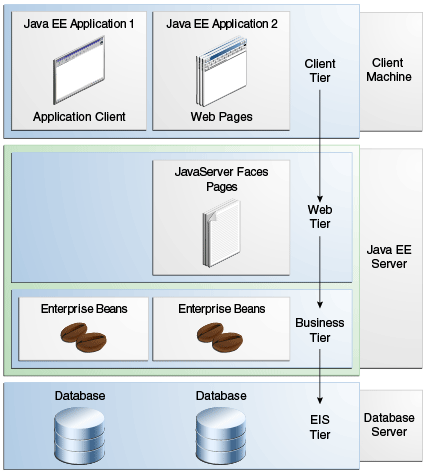


Gráfico . Aplicaciones Multicapa Java EE

#### Historial de Versiones

La plataforma era conocida como como Java 2 Platform, Enterprise Edition or J2EE hasta la versión 1.4, a partir de la cual se denominó sencillamente Java Platform, Enterprise Edition o Java EE. A lo largo del tiempo, la plataforma fue extendiendo la funcionalidad del modelo estándar Java Platform, Standard Edition (JSE) con el desarrollo de diferentes versiones:

* J2EE 1.2 (12 de Diciembre de 1999)
* J2EE 1.3 (24 de Septiembre de 2001)
* J2EE 1.4 (11 de Noviembre de 2003)
* Java EE 5 (11 de Mayo de 2006)
* Java EE 6 (10 de Diciembre de 2009)
* Java EE 7 (12 de Junio de 2013)

Cada versión extiende la anterior con nuevas APIs que aportan nuevos servicios y/o funcionalidades. Puede consultar los detalles concretos de cada versión en [5]. La versión 6 es la utilizada en el desarrollo del proyecto. Como se puede observar, Java EE es una tecnología en constante evolución, y la nueva versión Java EE 7, acaba de ser presentada en el mes de Junio de este año.

#### Servicios y APIs

#### Componentes Java EE

Un componente Java EE es una unidad funcional de software que se ejecuta en una aplicación Java EE junto con sus clases, archivos relacionados y que además se comunica con otros componentes. Java EE define los siguientes componentes:

* Applets y aplicaciones cliente son componentes que se ejecutan en el cliente.
* Las tecnologías Java Servlet, JavaServer Faces y JavaServer Pages (JSP) son componentes web que se ejecutan en el servidor.
* Los EJBs (Enterprise Beans) son componentes de negocio que se ejecutan en el servidor.

Estos componentes son escritos en lenguaje de programación java y son compilados del mismo modo que cualquier programa en este lenguaje. La diferencia entre componentes Java EE y clases de Java Estándar es que estos componentes son ensamblados en una aplicación Java EE, se organizan de acuerdo a la especificación Java EE y son desplegados en producción donde se ejecutan y manejan por un servidor Java EE.

Es esta sección se describirán los componentes Enterprise JavaBeans y Java ServerFaces que son los que se utilizaron en el desarrollo del proyecto.

##### Enterprise JavaBeans (EJBs)

Un Enterprise JavaBean es un componente del lado del servidor que se ejecutan dentro de un contenedor EJB y encapsula la lógica de negocio de una aplicación.

Los Enterprise JavaBeans simplifican el desarrollo de aplicaciones distribuidas de gran tamaño, por las siguientes razones:

* El contenedor EJB provee a los Enterprise Beans de servicios a nivel de sistema para que el desarrollador puede concentrarse en resolver problemas del negocio.
* Los Enterprise JavaBeans son componentes portables, pueden ser ensamblados y re-ensamblados en diferentes aplicaciones. Siempre y cuando utilicen la API estándar, estas aplicaciones se pueden ejecutar en cualquier servidor compatible con Java EE.

Existen 3 tipos de EJBs:

* **EJB de entidad (Entity Beans):**

Estos modelan conceptos de negocio como objetos persistentes asociados a datos. A partir de la versión 3.0 los Entity Beans pasan a pertenecer al API de persistencia (JPA). Se recuerda que el objetivo de esta API es no perder las ventajas de la orientación a objetos al interactuar con una base de datos. Se define un lenguaje propio para la persistencia, el Java Persistence Query Language (JPQL) y se permite la interacción con objetos simples, que en este contexto pasan a denominarse Plain Old Java Objects (POJOS).

* **EJBs de sesión (Session Beans):**

Gestionan el flujo de información del servidor. Representan procesos ejecutados en respuesta a una solicitud del cliente. Existen dos tipos distintos de EJBs de sesión:

* + Con estado (stateful): En estos beans de sesión un proceso de cliente involucra múltiples invocaciones, a través de las cuales se mantiene el estado, denominado conversational state. El estado no es persistente puesto que se pierde cuando el cliente libera el bean. Cada instancia tiene una entidad diferente.
  + Sin estado (stateless): Aquí un proceso cliente implica una invocación simple. Carecen de estado, así que no permiten almacenar datos específicos de cliente entre invocaciones. Todas las entidades tienen la misma instancia.
* **EJBs activados por mensaje (Message-driven Beans):**

Representan procesos ejecutados como respuesta a la recepción de un mensaje. Su funcionamiento es asíncrono. Utilizan el Java Messaging System (JMS), se suscriben a un tema (topic) o a una cola (queue) y se activan al recibir un mensaje dirigido a dicho tema o cola. No requieren instanciación por parte del cliente.

##### Características de EJBs 3.0

##### JavaServer Faces Tecnology (JSF)

JavaServer Faces es un componente y framework del lado del servidor que permite la construcción de tecnologías Java basadas en aplicaciones web. JSF incluye las siguientes características:

* Un conjunto de APIs para representar componentes de una interfaz de usuario y administrar su estado, manejar eventos, definir un esquema de navegación de las páginas y dar soporte para internacionalización y accesibilidad.
* Librerías de etiquetas para agregar componentes a páginas web y para conectar componentes a objetos del lado del servidor.

La funcionalidad proporcionada por una aplicación JavaServer Faces es similar a la de cualquier otra aplicación web Java. Una aplicación JSF incluye las siguientes partes:

* Un conjunto de páginas web en las que los componentes ya están diseñados.
* Un conjunto de etiquetas para agregar componentes a la página web.
* Un conjunto de beans gestionados, que son objetos contenedores gestionados ligeros con requisitos mínimos. Estos soportan un pequeño conjunto de servicios básicos, como la inyección de recursos, las devoluciones de llamada de ciclo de vida y los interceptores.
* Un descriptor de despliegue web (archivo web.xml).
* Opcionalmente, uno o más archivos de recursos de configuración de la aplicación, como por ejemplo un archivo faces-config.xml, que se puede utilizar para definir reglas de navegación de páginas y configurar los beans y otros objetos personalizados, tales como los componentes personalizados.

A continuación se muestra la interacción entre cliente y servidor en una aplicación JSF.

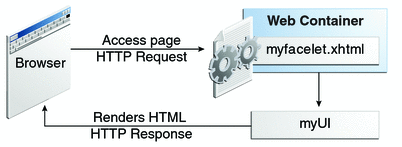


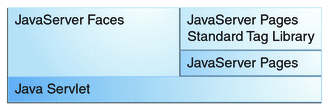
Gráfico .

La página web, myfacelet.xhtml, está construida usando etiquetas de componentes JSF. Las etiquetas de componentes son usadas para agregar componentes a la view (representada por myUI en el diagrama), el cual es la representación de la página del lado del servidor. A petición del cliente, la vista se presenta como una respuesta. *Rendering* es el proceso por el cual, basado en la vista del lado del servidor, el contenedor web genera una salida, tales como HTML o XHTML, que puede ser leído por el cliente, como un navegador.

Una de las mayores ventajas de la tecnología JavaServer Faces es que ofrece una separación limpia entre el comportamiento y la presentación de las aplicaciones web. Una aplicación JavaServer Faces puede mapear peticiones HTTP a la gestión de eventos de componentes específicos y administrar componentes como objetos con estado en el servidor.

La separación de la lógica de presentación también permite que cada miembro de un equipo de desarrollo de aplicaciones web pueda centrarse en una sola pieza del proceso de desarrollo y proporcionar un modelo de programación sencillo para enlazar las piezas.

Otro logro objetivo importante de la tecnología JSF, es aprovechar componentes familiares y conceptos web sin tener que limitarse a una tecnología particular de escritura o lenguaje de marcado. La tecnología JavaServer Faces está construida sobre la API de Servlet, como se muestra en la siguiente figura. (Oracle, s.f.)



Gráfico

### Principios de Programación Orientada a Objetos SOLID

* + **Principio de Responsabilidad Única (SRP)**

*“No debe haber más de una razón para que una clase cambie”*

Cuando los requerimientos cambian, estos se manifiestan a través de cambios en las responsabilidades de las clases. Si una clase asume más de una responsabilidad entonces habrá más de una razón para que esta cambie.

Por lo tanto, si una clase tiene más de una responsabilidad, entonces sus responsabilidades estarán acopladas y los cambios en una de las responsabilidades pueden afectar o inhibir la capacidad de la clase para satisfacer a otras. Este tipo de acoplamiento conduce a diseños frágiles que se rompen de manera inesperada.

Este principio indica que una clase debe tener una responsabilidad o un solo motivo para cambiar.(Martin, objectmentor.com, 2002)*.*

* + **Principio de Abierto-Cerrado (OCP)**

*“Un módulo debe estar abierto para la extensión pero cerrado para la modificación”*

El más importante de todos los principios de diseño orientado a objetos, se originó de la obra de Bertrand Meyer. Significa que deberíamos escribir nuestros módulos de modo que puedan ser extendidos, sin que tengan que ser modificados. En otras palabras, queremos ser capaces de cambiar lo que los módulos hacen, sin cambiar el código fuente de los módulos.

Existen varias técnicas para lograr este principio, tales como, polimorfismo dinámico y polimorfismo estático. Usando estas técnicas podemos crear módulos que sean extensibles.

Incluso si el principio OCP no puede lograrse plenamente, su cumplimiento parcial puede hacer mejoras dramáticas en la estructura de una aplicación. Siempre es mejor si los cambios no se propagan en el código existente que ya funciona.(Martin, objectmentor.com, 2000)*.*

* + **Principio de Sustitución de Liskov (LSP)**

*“Las subclases deben ser sustituibles por sus clases bases”*

Este principio fue introducido por Barbar Liskov, en su trabajo con respecto a la abstracción de datos. También deriva del concepto de Desing by Contract (DBC) de Bertrand Meyer.

Este principio indica que un usuario de una clase base debería seguir funcionando correctamente si un derivado de esa clase base se pasa a ella.

Asimismo, está estrechamente relacionado con el anterior en cuanto a la extensibilidad de las clases cuando ésta se realiza mediante herencia o subtipos. Si una función no cumple el LSP entonces rompe el OCP puesto que para ser capaz de funcionar con subtipos (clases hijas) necesita saber demasiado de la clase padre y por tanto, modificarla.(Martin, objectmentor.com, 2000)*.*

* + **Principio de Segregación de Interfaces (ISP)**

*“Haz interfaces que sean específicas para un tipo de cliente”*

La esencia de este principio indica que, si tienes una clase que tiene varios clientes, en vez de cargar la clase con todos los métodos que los clientes necesitan, crea interfaces específicas para cada cliente y herédalas a la clase.

El ISP no recomienda que cada clase que utiliza un servicio tenga una interfaz propia que el servicio deba heredar. Si ese fuera el caso, el servicio sería dependiente de cada cliente de una manera extraña y poco viable. En vez de eso, los clientes deben ser clasificados por su tipo, y las interfaces deber ser creadas para cada tipo de cliente.(Martin, objectmentor.com, 2000)*.*

* + **Principio de Inversión de Dependencias (DIP)**

1. *Módulos de alto nivel no deben depender de módulos de bajo nivel. Ambos deben depender de abstracciones.*
2. *Las abstracciones no deben depender de detalles. Los detalles deben depender de las abstracciones.*

Si se consideran las implicaciones que tienen los módulos de alto nivel que dependen de los módulos de bajo nivel. Son los módulos de alto nivel los que contienen decisiones de política y modelos del negocio importantes de una aplicación. Cuando estos módulos dependen de módulos de bajo nivel, los cambios a los módulos de bajo nivel pueden afectarlos directamente; y forzarlos a cambiar. Asimismo al existir esta dependencia, se dificulta los reutilización de los módulos de alto nivel.

Sin embargo, cuando los módulos de alto nivel son independientes de los módulos de bajo nivel, estos pueden ser reutilizados de forma sencilla.

Por ello, DIP explica que un módulo concreto A, no debe depender directamente de otro módulo concreto B, sino de una abstracción de B. Tal abstracción es una interfaz o una clase (que podría ser abstracta) que sirve de base para un conjunto de clases hijas.(Martin, objectmentor.com, 1996)

## Definición de términos básicos

### Aplicación Software basada en tecnologías móvil y web

* + **Software de Aplicación**

Es el conjunto de programas de cómputo, procedimientos, reglas, documentación y datos asociados, que forman parte de las operaciones de un sistema de computación. Estos permiten a los usuarios llevar a cabo una o varias tareas específicas, en cualquier campo de actividad susceptible de ser automatizado o asistido.(Sommerville, 2011)

* + **Aplicación móvil**

Software de aplicación diseñado para ser ejecutado en teléfonos inteligentes, tabletas y otros dispositivos móviles. Se encuentran disponibles a través de plataformas de distribución, operadas por las compañías propietarias de los sistemas operativos móviles como Android, iOS, BlackBerry OS, Windows Phone, entre otros.

* + **Aplicación Web**

Aplicación software que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una intranet mediante un navegador.(Sergio, 2002)

Se usara el estándar internacional ISO 9126 para evaluar la calidad de la aplicación software

### Acceso a Servicios de Salud

* **Servicios de Salud**

Prestaciones que brindan asistencia sanitaria, estas son entregadas por personal de salud en forma directa o bajo supervisión de estas, con el propósito de diagnosticar, promover, prevenir, mantener y/o recuperar la salud de una población objetivo, tanto a nivel individual como colectivo. *(Ministerio de Salud del Peru y Direccion General de Epidemiologia, 2013)*

* **Acceso**

Acción de llegar o acercarse.

Se evaluaran los servicios de salud que son susceptibles a ser automatizados por una aplicación software.

# HIPÓTESIS

## Formulación de la hipótesis

Una aplicación software basada en tecnologías móvil y web mejora el acceso a servicios de salud de los habitantes del distrito de Trujillo.

## Operacionalización de variables

|  |  |  |  |
| --- | --- | --- | --- |
| **VARIABLE** | **DEFINICIÓN CONCEPTUAL** | **DIMENSIONES** | **INDICADORES** |
| Aplicación software basada en tecnologías móvil y web  (Variable Independiente) | Conjunto de programas de cómputo, procedimientos, reglas, documentación y datos asociados, que forman parte de las operaciones de un sistema de computación. Estos permiten a los usuarios llevar a cabo una o varias tareas específicas, en cualquier campo de actividad susceptible de ser automatizado o asistido.  Se evaluara la calidad de la aplicación software en base al estándar internacional ISO 9126. | Usabilidad | Grado de esfuerzo cognitivo de los usuarios con el uso de la aplicación software. |
| Fiabilidad | Nivel de rendimiento óptimo de la aplicación software durante un periodo determinado. |
| Mantenibilidad | Nivel de esfuerzo necesario para realizar modificaciones a la aplicación software. |
| Funcionalidad | Nivel de satisfacción de las funcionalidades propuestas para la aplicación software. |
| Acceso a servicios de salud  (Variable Dependiente) | Prestaciones que brindan asistencia sanitaria, estas son entregadas por personal de salud en forma directa o bajo supervisión de estas, con el propósito de diagnosticar, promover, prevenir, mantener y/o recuperar la salud de una población objetivo, tanto a nivel individual como colectivo.  Se evaluaran los servicios de salud que son susceptibles a ser automatizados por una aplicación software. | Información de Proveedores de Salud | Nivel de acceso de los habitantes a información de proveedores de salud. |
| Consultas Médicas | Nivel de acceso de los habitantes a consultas médicas. |
| Reserva de Citas Medicas | Nivel de acceso de los habitantes a reserva de citas médicas. |
| Información acerca de Temas de Salud | Nivel de acceso de los habitantes a información sobre temas de salud |

Tabla . Cuadro detallado de la Operacionalizacion de las variables

# 

# PROPUESTA DE APLICACIÓN PROFESIONAL

En este capítulo se detalla el planteamiento, la arquitectura y construcción de la solución software. Asimismo se describen las tecnologías y herramientas utilizadas en su desarrollo.

## Planteamiento de la Solución

Las características de la solución software fueron tomadas en base a las necesidades expuestas en la realidad problemática. Para ello se tiene como objetivo implementar dos aplicaciones móviles y una web administrativa, las cuales se detallan a continuación.

### Aplicación Móvil Paciente

Esta aplicación móvil está dirigida a los habitantes del distrito de Trujillo y tendrá las siguientes características:

* Registro de Usuarios
* Registro de antecedentes médicos
* Registro de familiares
* Envió de consultas medicas
* Historial de consultas medicas
* Reserva de citas medicas
* Historial de citas medicas
* Búsqueda de proveedores de salud
* Agregar doctores favoritos
* Consulta de temas de salud
* Consulta de clínicas cercanas
* Consulta de información de especialidades medicas

En la lógica del negocio de la aplicación se tomaron las siguientes consideraciones:

* Los pacientes deberán registrar sus antecedentes médicos junto con sus datos personales, antes de utilizar la aplicación.
* Los pacientes podrán registrar a personas menores de edad que forman parte de su familia. Al registrarlos deben indicar también sus antecedentes médicos.
* Al enviar una consulta médica los pacientes deben indicar la especialidad y una descripción en la cual registran los síntomas de su dolencia. Asimismo se puede adjuntar una imagen si se considera necesaria.
* La opción “Información de especialidades” representa una descripción acerca de especialidades médicas, las cuales servirán de ayuda a los pacientes al momento de escoger una especialidad en él envió de sus consultas médicas.
* Los pacientes recibirán 3 respuestas de diferentes médicos. Las respuestas pueden ser puntuadas, esto ayudara a los doctores a acumular puntos.
* Los pacientes podrán agregar a doctores a su lista de favoritos para así enviar peticiones de citas médicas con facilidad.
* Para reservar una cita el paciente envía una notificación al médico, este responderá a la notificación indicando la fecha y hora de la cita. El paciente podrá aceptar o rechazar la cita programada.
* La opción temas de salud o casos de salud representa un foro en el cual los administradores subirán problemáticas médicas y los médicos podrán comentar indicando las recomendaciones que consideren adecuadas. Estas recomendaciones podrán ser calificadas por los pacientes.

### Aplicación Móvil Doctor

Esta aplicación móvil está dirigida a los médicos y tendrá las siguientes funcionalidades:

* Consultar historial de citas medicas
* Responder a consultas medicas
* Consultar historial de citas medicas
* Asignar citas medicas
* Consultar temas de salud
* Responder a temas de salud
* Consultar información de pacientes

Las consideraciones para la lógica del negocio son las siguientes:

* Los médicos serán registrados por los administradores
* Las consultas llegaran a los médicos de acuerdo a la especialidad seleccionada por el paciente. Solo las 3 primeras respuestas de diferentes médicos serán enviadas al paciente.

### Aplicación Web Administrativa

Esa aplicación web tendrá las siguientes funcionalidades:

* Gestionar doctores
* Gestionar usuarios
* Gestionar ubigeo
* Gestionar clínicas
* Gestionar casos de salud

## Arquitectura de la Solución

### Modelo Físico

La solución usa un modelo de arquitectura cliente-servidor, en el cual las tareas son distribuidas entre los proveedores de recursos (servidores) y los demandantes (clientes). El Gráfico 11 muestra el diagrama de despliegue de la solución, en este, cada elemento físico está representado por un nodo, los cuales se describen a continuación:

* **Nodo Cliente Administrador**

Este nodo representa una pc u otro dispositivo que disponga de un navegador web por el cual los administradores tendrán acceso al sistema.

* **Nodo Cliente Paciente**

Este nodo representa un dispositivo móvil con sistema operativo Android en el cual se ejecutara el archivo MedFinderPA.apk, que permitirá a los pacientes instalar y desplegar la aplicación. Esta aplicación realizara peticiones al servidor por medio de un web service implementado en restful.

* **Nodo Cliente Doctor**

Este nodo al igual que el nodo cliente paciente representa un dispositivo móvil con sistema operativo Android en el cual se ejecutara el archivo MedFinderPA.apk, que permitirá a los pacientes instalar y desplegar la aplicación. Esta aplicación realizara peticiones al servidor por medio de un web service implementado en restful.

* **Nodo Servidor**

Este nodo representa un servidor de aplicaciones VPS (Virtual Private Server). En este se encuentran dos nodos anidados los cuales representan un servidor Payara basado en Glassfish encargado de ejecutar los wars y web service de la aplicación web mientras el otro servidor está encargado de ejecutar la base de datos.

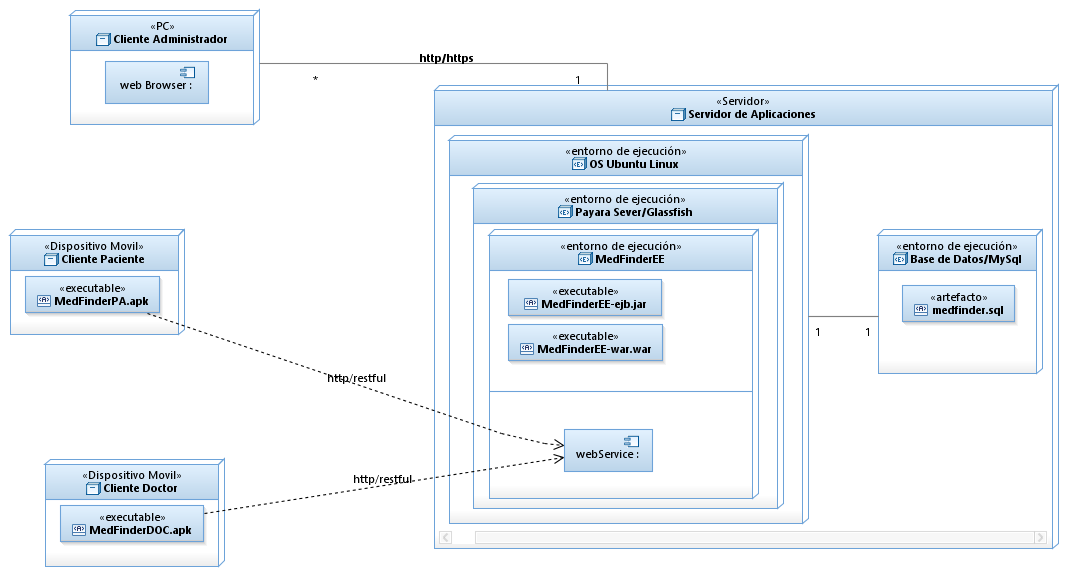


Gráfico . Diagrama de Despliegue de la solución

### Modelo Lógico

#### Web

El Gráfico 12 muestra el diagrama de componentes de la aplicación web. Esta aplicación está desarrollada en la plataforma Java Enterprise Edition (J2EE) y sus componentes se describen a continuación:

* En la presentación, se usó la tecnología Java Server Faces la cual emplea el framework MVC (Modelo Vista Controlador). Las vistas están representadas por JSF Pages, el controlador está representado por Managed Beans los cuales permitirán la comunicación con el modelo (Entity) a través de Session Beans.
* En la lógica de negocio se usó la tecnología Enterprise Java Beans. En esta, se empleara el patrón session facade el cual permite a los Enterprise Beans encapsular la lógica de negocio y lógica de datos, y exponerlas mediante interfaces (Facades).
* En el acceso a datos se usó Java Persistence API, esta tecnología nos permite generar clases entity para cada una de las tablas de la base de datos. Asimismo nos genera los queries básicos de base de datos.

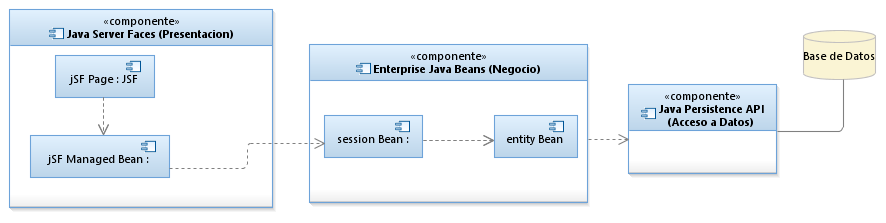


Gráfico . Diagrama de Componentes Web

#### Móvil

## Construcción

Las aplicaciones software fueron construidas usando el marco de desarrollo ágil Scrum. A continuación se detallan cada una de las fases con sus respectivos artefactos.

### Pila del Producto

En la Tabla 2 se muestra el inventario de funcionalidades o historias de usuario que se realizarán a lo largo del proyecto. Estas historias están agrupadas en sprints y tienen un tiempo estimado para ser terminadas. La unidad de trabajo empleada es de 4 horas diarias por integrante.

|  |  |  |  |
| --- | --- | --- | --- |
| **PILA DEL PRODUCTO** | | | |
| **ID** | **PRIORIDAD** | **HISTORIAS** | **TRABAJO** |
| **APLICACIÓN MOVIL-PACIENTE** | | | |
| P01 | Media | Creación y diseño de layouts | 16h |
| P02 | Media | Creación y validación de base de datos | 8h |
| P03 | Media | Registrar Usuarios | 12h |
| P04 | Media | Registrar antecedentes médicos | 12h |
| P05 | Media | Registrar familiares | 12h |
| P06 | Alta | Enviar de consultas medicas | 12h |
| P07 | Alta | Consultar historial de consultas medicas | 12h |
| P08 | Alta | Reservar citas medicas | 12h |
| P09 | Alta | Consultar historial de citas medicas | 12h |
| P10 | Media | Consultas temas de salud | 12h |
| P11 | Media | Búsqueda de proveedores de salud | 12h |
| P12 | Media | Agregar doctores favoritos | 12h |
| P13 | Media | Consultar clínicas cercanas | 12h |
| P14 | Media | Consultar información de especialidades medicas | 12h |
| **APLICACIÓN MOVIL-DOCTOR** | | | |
| D1 | Media | Creación y diseño de layouts | 12h |
| D2 | Alta | Consultar historial de consultas medicas | 12h |
| D3 | Alta | Responder a consultas medicas | 12h |
| D4 | Alta | Consultar historial de citas medicas | 12h |
| D5 | Alta | Asignar citas medicas | 12h |
| D6 | Media | Consultar temas de salud | 12h |
| D7 | Media | Responder a temas de salud | 12h |
| D8 | Media | Consultar información de pacientes | 12h |
| **WEB ADMINISTRATIVA** | | | |
| A1 | Media | Creación de layouts y estilos del sitio web | 24h |
| A2 | Alta | Gestionar doctores | 12h |
| A3 | Alta | Gestionar usuarios | 12h |
| A4 | Media | Gestionar ubigeo | 12h |
| A5 | Media | Gestionar clínicas | 12h |
| A6 | Media | Gestionar casos de salud | 12h |

Tabla . Pila del Producto

Tercer Sprint estimación: 96h

Segundo Sprint estimación: 96h

Primer Sprint estimación: 168h

## PRIMER SPRINT

### Pila del Sprint

La Tabla 3 muestra la pila para el primer sprint. Este sprint tiene una duración de 4 semanas y especifica las actividades diarias para el desarrollo de la aplicación móvil paciente.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **PRIMER SPRINT** | | | | | **Horas Pendientes** | | | | | | | | | | | | | | | | | | | | | | | |
| **Desarrollar una aplicación móvil para el paciente en la plataforma Android** | | | | | 168 | 160 | 156 | 156 | 156 | 152 | 148 | 148 | 144 | 128 | 120 | 104 | 96 | 84 | 80 | 68 | 60 | 40 | 24 | 16 | 8 | 0 | 0 | 0 |
| **Categoría** | **Tarea** | **Responsable** | **Est. en Horas** | **Estado** | **18-Ene** | **19-Ene** | **20-Ene** | **21-Ene** | **22-Ene** | **23-Ene** | **25-Ene** | **26-Ene** | **27-Ene** | **28-Ene** | **29-Ene** | **30-Ene** | **1-Feb** | **2-Feb** | **3-Feb** | **4-Feb** | **5-Feb** | **6-Feb** | **8-Feb** | **9-Feb** | **10-Feb** | **11-Feb** | **12-Feb** | **13-Feb** |
| Diseño | Creación y diseño de layouts | Jerry Guzmán V. Eder Haro A. | 16h | Terminando | 16 | 8 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Diseño | Creación y validación de base de datos | Jerry Guzmán V. Eder Haro A. | 8h | Terminando | 8 | 8 | 8 | 8 | 8 | 8 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Registrar Usuarios** | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de activities y métodos | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de queries, conexión a Sqlite y webservice | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Realización de pruebas y validación de datos | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Enviar Consultas Medicas** | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de activities y métodos | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de queries, conexión a Sqlite y webservice | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Realización de pruebas y validación de datos | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Registrar Antecedentes Médicos** | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de activities y métodos | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de queries, conexión a Sqlite y webservice | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Realización de pruebas y validación de datos | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |  |  |  |
| **Consultar historial de Consultas** | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de activities y métodos | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de queries, conexión a Sqlite y webservice | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Realización de pruebas y validación de datos | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |  |  |  |
| **Registrar Familiares** | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de activities y métodos | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de queries, conexión a Sqlite y webservice | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Realización de pruebas y validación de datos | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |  |
| **Reservar Citas Medicas** | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de activities y métodos | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de queries, conexión a Sqlite y webservice | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |
| Desarrollo | Realización de pruebas y validación de datos | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |
| **Consultar Casos de Salud** | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de activities y métodos | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de queries, conexión a Sqlite y webservice | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |
| Desarrollo | Realización de pruebas y validación de datos | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |
| **Historial de Citas Medicas** | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de activities y métodos | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |
| Desarrollo | Implementación de queries, conexión a Sqlite y webservice | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |
| Desarrollo | Realización de pruebas y validación de datos | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |
| **Consultar Clinicas Cercanas** | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de activities y métodos | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |
| Desarrollo | Implementación de queries, conexión a Sqlite y webservice | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |
| Desarrollo | Realización de pruebas y validación de datos | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |
| **Consultar Información de Especialidades Medicas** | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de activities y métodos | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |
| Desarrollo | Implementación de queries, conexión a Sqlite y webservice | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |
| Desarrollo | Realización de pruebas y validación de datos | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |
| **Agregar Doctores Favoritos** | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de activities y métodos | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |
| Desarrollo | Implementación de queries, conexión a Sqlite y webservice | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |
| Desarrollo | Realización de pruebas y validación de datos | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |
| **Búsqueda de Proveedores de Salud** | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de activities y métodos | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |
| Desarrollo | Implementación de queries, conexión a Sqlite y webservice | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |
| Desarrollo | Realización de pruebas y validación de datos | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |

Tabla . Pila del Primer Sprint

### Diagrama Burn-Down

El Gráfico 12 muestra el diagrama burn-down del primer sprint. Este diagrama representa la comparación entre el tiempo de estimación que se hizo en la pila del producto y el tiempo real que tomo la realización del primer sprint, descritos por la línea roja y azul respectivamente.

Gráfico . Diagrama Burn-Down del Primer Sprint

### Incremento

#### Diagrama de Clases

#### Diagramas de Secuencia

#### Flujo de Pantallas

## SEGUNDO SPRINT

### Pila del Sprint

La Tabla 4 muestra la pila para el segundo sprint. Este sprint tiene una duración de 2 semanas y especifica las actividades diarias para el desarrollo de la aplicación móvil doctor.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **SEGUNDO SPRINT** | | | | | **Horas Pendientes** | | | | | | | | | | | | | | |
| **Desarrollar una aplicación móvil Medico con funcionalidades de atención de consultas, asignación de citas y envió de recomendaciones para los casos de salud.** | | | | | 96 | 92 | 88 | 84 | 76 | 68 | 60 | 48 | 44 | 32 | 24 | 20 | 12 | 4 | 0 |
| **Categoría** | **Tarea** | **Responsable** | **Est. en Horas** | **Estado** | **15-Feb** | **16-Feb** | **17-Feb** | **18-Feb** | **19-Feb** | **20-Feb** | **22-Feb** | **23-Feb** | **24-Feb** | **25-Feb** | **26-Feb** | **27-Feb** | **29-Feb** | **1-Mar** | **2-Mar** |
| Diseño | Creación y diseño de layouts | Jerry Guzmán V.Eder Haro A. | 12h | Terminando | 12 | 8 | 4 | 0 |  |  |  |  |  |  |  |  |  |  |  |
| **Consultar Temas de Salud** | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de activities y métodos | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de queries, conexión a Sqlite y webservice | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |  |
| Desarrollo | Realización de pruebas y validación de datos | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |
| **Consultar Historial de Consultas Medicas** | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de activities y métodos | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de queries, conexión a Sqlite y webservice | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |  |
| Desarrollo | Realización de pruebas y validación de datos | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |
| **Responder a Temas de Salud** | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de activities y métodos | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |
| Desarrollo | Implementación de queries, conexión a Sqlite y webservice | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |
| Desarrollo | Realización de pruebas y validación de datos | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |
| **Responder a Consultas Medicas** | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de activities y métodos | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |
| Desarrollo | Implementación de queries, conexión a Sqlite y webservice | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |
| Desarrollo | Realización de pruebas y validación de datos | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |
| **Consultar Información de Pacientes** | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de activities y métodos | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |
| Desarrollo | Implementación de queries, conexión a Sqlite y webservice | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |
| Desarrollo | Realización de pruebas y validación de datos | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |
| **Consultar Historial de Citas Medicas** | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de activities y métodos | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |
| Desarrollo | Implementación de queries, conexión a Sqlite y webservice | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |
| Desarrollo | Realización de pruebas y validación de datos | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |
| **Asignar Citas Medicas** | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de activities y métodos | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |
| Desarrollo | Implementación de queries, conexión a Sqlite y webservice | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |
| Desarrollo | Realización de pruebas y validación de datos | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |

Tabla . Pila del Segundo Sprint

### Diagrama Burn-Down

El Gráfico 15 muestra el diagrama burn-down del segundo sprint. Este diagrama representa la comparación entre el tiempo de estimación que se hizo en la pila del producto y el tiempo real que tomo la realización del segundo sprint, descritos por la línea roja y azul respectivamente.

Gráfico . Diagrama Burn-Down del Segundo Sprint

### Incremento

#### Diagrama de Clases

#### Diagrama de Secuencia

#### Flujo de Pantallas

## TERCER SPRINT

### Pila del Sprint

La Tabla 5 muestra la pila para el tercer sprint. Este sprint tiene una duración de 3 semanas y especifica las actividades diarias para el desarrollo de la aplicación web.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **TERCER SPRINT** | | | | | **Horas Pendientes** | | | | | | | | | | | | | | | |
| **Desarrollar una plataforma web administrativa con funcionalidades para la gestión de pacientes, médicos, casos de salud, consultas médicas y citas.** | | | | | 96 | 92 | 88 | 80 | 76 | 72 | 60 | 52 | 40 | 32 | 16 | 12 | 4 | 0 | 0 | 0 |
| **Categoría** | **Tarea** | **Responsable** | **Est. en Horas** | **Estado** | **7-Mar** | **8-Mar** | **9-Mar** | **10-Mar** | **11-Mar** | **12-Mar** | **13-Mar** | **14-Mar** | **15-Mar** | **16-Mar** | **17-Mar** | **18-Mar** | **19-Mar** | **20-Mar** | **21-Mar** | **22-Mar** |
| Diseño | Creación del diseño y estructura de proyecto web | Jerry Guzmán V. Eder Haro A. | 24 | Terminando | 24 | 20 | 16 | 8 | 4 | 0 |  |  |  |  |  |  |  |  |  |  |
| **Gestionar Usuario** | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de session beans(Facades) | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de vistas, managed beans y converters | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |
| Desarrollo | Realización de pruebas y validación de datos | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |
| **Gestionar Ubigeo** | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de session beans(Facades) | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de vistas, managed beans y converters | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |  |
| Desarrollo | Realización de pruebas y validación de datos | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |  |
| **Gestionar Médicos** | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de session beans(Facades) | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |
| Desarrollo | Implementación de vistas, managed beans y converters | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |
| Desarrollo | Realización de pruebas y validación de datos | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |
| **Gestionar Clínicas** | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de session beans(Facades) | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |
| Desarrollo | Implementación de vistas, managed beans y converters | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |  |
| Desarrollo | Realización de pruebas y validación de datos | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |  |
| **Gestionar Casos de Salud** | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de session beans(Facades) | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |
| Desarrollo | Implementación de vistas, managed beans y converters | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |
| Desarrollo | Realización de pruebas y validación de datos | Jerry Guzmán V. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |
| **Gestionar Especialidad** | | | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desarrollo | Implementación de session beans(Facades) | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |
| Desarrollo | Implementación de vistas, managed beans y converters | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |
| Desarrollo | Realización de pruebas y validación de datos | Eder Haro A. | 4h | Terminando | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 0 |  |  |  |  |  |

Tabla . Pila del Tercer Sprint

### Diagrama Burn-Down

El Gráfico 16 muestra el diagrama burn-down del tercer sprint. Este diagrama representa la comparación entre el tiempo de estimación que se hizo en la pila del producto y el tiempo real que tomo la realización del tercer sprint, descritos por la línea roja y azul respectivamente.

Gráfico . Diagrama Burn-Down del Tercer Sprint

### Incremento

#### Diagrama de Casos de Uso

#### Diagrama de Clases

#### Flujo de Pantallas

# MATERIALES Y MÉTODOS

## Tipo de diseño de investigación

Pre experimental. Se tomara como muestra un sector de la población del Distrito de Trujillo para estimular la variable independiente (aplicación software basada en tecnologías móvil y web) y determinar su efecto en la variable dependiente (acceso a servicios de salud) utilizando un diseño de pre prueba – pos prueba con un solo grupo.

## Material de estudio

### Unidad de estudio

Habitante del distrito de Trujillo mayor de 14 años de edad.

### Población

Habitantes del Distrito de Trujillo, esta población será estratificada ya que se enfocara en las personas mayores de 14 años. De acuerdo al censo realizado por el INEI 2007, el distrito de Trujillo cuenta con un total de 294,899 habitantes, de los cuales las personas mayores de 14 años son 227,587.

### Muestra.

Para determinar la muestra se utilizara el muestreo probabilístico de tipo aleatorio simple.

n = F:\Informacion\UPN\Tesis - 2015\formula.png

Donde:

n = Número de elementos de la muestra

N = Número de elementos del universo

P y Q = Probabilidad con las que se presenta el fenómeno, si no se conoce P = 0.5 y Q =0.5

Z = Valor critico 1.96

E = Margen de error permitido 0.05

Nuestra unidad de estudio será cada uno de los habitantes del Distrito de Trujillo mayores de 14 años.

Nuestra muestra(n) será de **383** habitantes.

## Técnicas, procedimientos e instrumentos

### Para recolectar datos

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Objetivo Especifico** | **Fuente** | **Técnica** | **Instrumento** | **Logro** |
| Incrementar el nivel de acceso de los habitantes a información de proveedores de salud. | Habitantes del distrito de Trujillo | Encuesta | Cuestionario | Conocer el nivel de acceso de los habitantes del distrito de Trujillo a información de proveedores de salud antes y después del uso de la aplicación software. |
| Incrementar el nivel de acceso de los habitantes a consultas médicas | Habitantes del distrito de Trujillo | Encuesta | Cuestionario | Conocer el nivel de acceso de los habitantes del distrito de Trujillo a consultas médicas antes y después del uso de la aplicación software. |
| Incrementar el nivel de acceso de los habitantes a reserva de citas médicas. | Habitantes del distrito de Trujillo | Encuesta | Cuestionario | Conocer el nivel de acceso de los habitantes del distrito de Trujillo a reserva de citas médicas antes y después del uso de la aplicación software. |
| Incrementar el nivel de acceso de los habitantes a información sobre temas de salud. | Habitantes del distrito de Trujillo | Encuesta | Cuestionario | Conocer el nivel de acceso de los habitantes del distrito de Trujillo a información sobre temas de salud antes y después del uso de la aplicación software. |

Tabla .

### Para procesar datos.

* **Técnicas**
  + **Análisis Descriptivos**

De acuerdo a la escalas de las variables de estudio (ordinal), se procederá a calcular su mediana, moda tabulando los datos en tablas de frecuencias o gráficos de barras o circular según sea la naturaleza de los resultados.

* + **Análisis Ligados a la Hipótesis**

Para probar la hipótesis se hará uso de la prueba estadística T-Student.

* **Instrumentos**
  + **Microsoft Excel (Hoja de Cálculo)**

Software mediante el cual se realizaran tabulaciones y cálculo de datos.

# RESULTADOS

Comprende los resultados de la medición de los indicadores correspondientes a la variable dependiente, bajo los efectos de la variable independiente.

Si el producto de especialidad no se pudiera aplicar de manera real para medir algunos indicadores, es válido realizar la medición a través de una simulación o proyección estadística del mismo.

Los resultados concluyen con la prueba de hipótesis.

# DISCUSIÓN

Es un texto argumentativo a través del cual el autor valida su hipótesis, tomando como sustento los resultados obtenidos en la medición de indicadores y la prueba de hipótesis. Puede además mencionar aquí las características del producto-solución que permitieron alcanzar los resultados.

CONCLUSIONES

Las conclusiones se redactan en relación a los objetivos planteados. La primera conclusión debe responder si se logró o no el objetivo general y especificar alguna evidencia de ello.

Las demás conclusiones responden a los objetivos específicos en el orden en que fueron planteados.

RECOMENDACIONES

Las recomendaciones deben ser dirigidas a todos los actores interesados en el estudio. Por ejemplo, a la empresa, a otros investigadores interesados en el tema, a otros profesionales de la especialidad, entre otros.

REFERENCIAS

ANEXOS

En anexos puede ir:

* El formato de los instrumentos de registro utilizados (formato de encuesta, guía de entrevista, ficha de observación, etcétera).
* La transcripción de la norma en caso exista un marco legal referencia. - Otros documentos.

Cada uno de los instrumentos, evidencias u otros insertados en los anexos, va en hoja independiente. No pueden ir dos anexos en una misma hoja. Cada hoja que contiene un anexo debe ser numerada: ANEXO n.° 1. Título del anexo.