



FACULTAD DE INGENIERÍA Y CIENCIAS AGROPECUARIAS

DESAROLLAR UN APLICATIVO EN PLATAFORMA WEB PARA RESERVA DE HABITACIONES  
DE HOTEL Y ANÁLISIS DE LA INFORMACIÓN PARA LA TOMA DE DECISIONES  
BASADO EN LAS RESERVAS

Trabajo de Titulación presentado en conformidad con los requisitos establecidos  
para optar por el título de Ingeniero en Sistemas de Computación e Informática.

Profesor Guía

MBA. Christian Anibal Bastidas Romero

Autor

Vicente Fernando Piedra Veintimilla

Año  
2016

## DECLARACIÓN DEL PROFESOR GUÍA

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

---

Christian Anibal Bastidas Romero  
Master in Business Administration  
CI. 171052854-6

## 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.”

---

Vicente Fernando Piedra Veintimilla  
CI. 172248165-0

## AGRADECIMIENTOS

Agradezco a mi madre, quien siempre será un apoyo fundamental en mi vida; asimismo a mi hermano y a todas aquellas personas que brindaron su aporte de creatividad e ingenio a este proyecto.

## DEDICATORIA

Este proyecto está dedicado a mis amigos que siempre confiaron en mí, ya que sin su apoyo no hubiera sido posible el desarrollo de este trabajo.

## RESUMEN

El proyecto en mención está orientado al desarrollo de un motor de reservas a nivel nacional. Será un sitio en plataforma web que servirá para efectuar reservas de habitaciones de hotel. Mediante este sistema, se pretende obtener reportes que sirvan para que los hoteles y gobiernos autónomos descentralizados realicen los análisis correspondientes.

El objetivo de este proyecto es recopilar y brindar estadísticas de visitas, reservas, comparaciones de precios a cada región de tal forma que ayude a los hoteles en la toma de decisiones mediante la visualización de gráficas con datos estadísticos. Para la consecución de este objetivo se planificaron reuniones con los hoteles patrocinadores, quienes fueron los encargados de establecer las necesidades del sitio web junto con una reconocida organización, la cual se encargará de promocionar y vender el proyecto a los hoteles de todas las regiones del país.

Este proyecto busca implementar herramientas para la industria hotelera nacional, agilizando las reservas y proporcionando información de análisis. Para obtener una satisfacción adecuada del proyecto, se realizaron análisis del estado actual de la industria, controles de calidad del producto y pruebas unitarias. Estas herramientas ayudaron a identificar si se han cumplido con éxito las necesidades de los hoteles patrocinadores. Finalmente se lograron varias conclusiones, dando como resultado una medida de las mejoras obtenidas.

## ABSTRACT

The goal of the project is to develop a nation-wide booking engine. It will be a web site that will serve as platform to make hotel rooms reservations. Through this system, it will generate reports that will help hotels and autonomous governments' agencies make related analysis.

The project objective is to collect, provide and graphically display statistics about visits, reservations, and price comparisons between each region to help hotels in decision making.

In order to achieve this goal, several meetings were held with sponsoring hotels – those that were responsible for establishing the website needs – and with a recognized organization responsible for promoting and selling the project to the hotels located in all the regions of the country.

This project seeks to implement tools to support the national hotel industry by expediting reservations and by providing information analysis. To assure an adequate project satisfaction, analysis of the current state of the industry, the use of product quality controls and unit tests. These tools helped identify if the needs of the sponsoring hotels have been successfully met.

Finally, several conclusions were attained that provided the improvements achieved.

## ÍNDICE

INTRODUCCIÓN .....	1
1. MARCO TEÓRICO.....	3
1.1. Sector de Alojamiento .....	3
1.1.1. Hotel .....	3
1.2. Aspectos Macroeconómicos .....	3
1.2.1. Producto Interno Bruto (PIB) .....	3
1.2.2. Inflación .....	5
1.2.3. Empleo .....	6
1.3. Arquitectura de Software .....	8
1.3.1. Estilo Arquitectónico .....	10
1.3.2. Estilos de Llamada y Retorno .....	11
1.3.3. Arquitectura Multicapas .....	11
1.3.4. Patrón Arquitectónico .....	14
1.3.5. Patrón Arquitectónico Modelo Vista Controlador .....	15
1.3.6. Patrón de Diseño .....	17
1.4. Gestión Ágil de Proyectos de Software .....	18
1.4.1. Objetivos de la gestión ágil .....	20
1.5. Metodologías ágiles .....	21
1.5.1. Manifiesto Ágil .....	21
1.5.2. Metodologías Ágiles vs Tradicionales.....	26
1.5.3. Scrum .....	29
1.5.4. Visión general del proceso .....	30
1.5.5. Artefactos de Scrum .....	33

1.5.6.	Roles o responsabilidades de Scrum .....	34
1.5.7.	Pruebas Ágiles.....	35
1.5.8.	Agile Testing Quadrants .....	37
<b>2.</b>	<b>DESARROLLO DE LA SOLUCIÓN.....</b>	<b>40</b>
2.1.	Visión de la Solución .....	40
2.1.1.	Alcance Inicial.....	40
2.2.	Planificación del Proyecto .....	54
2.2.1.	Cronograma del Proyecto.....	54
2.2.2.	Plan de Liberación .....	54
2.2.3.	Planificación Ágil.....	59
2.2.4.	Condiciones de Satisfacción.....	64
2.3.	Construcción de la solución .....	65
2.3.1.	Sprint 0 .....	65
2.3.2.	Diagrama de Arquitectura .....	65
2.3.3.	Iteraciones de Desarrollo.....	69
2.3.4.	Release Burndown Chart.....	72
2.3.5.	Despliegue del Producto.....	72
2.3.6.	Implementación de Integración Continua .....	73
2.4.	Cierre del Proyecto .....	75
2.4.1.	Acta de Cierre del Proyecto.....	75
<b>3.</b>	<b>CASO DE ESTUDIO .....</b>	<b>76</b>
3.1.	Estado Actual .....	76
3.2.	Diseño del Plan Piloto .....	77
3.3.	Medida de la Mejora Obtenida .....	78

4.	PRUEBAS DE ACEPTACIÓN Y RESULTADOS .....	79
4.1.	Pruebas de aceptación .....	79
4.2.	Resultados .....	90
5.	CONCLUSIONES Y RECOMENDACIONES .....	95
5.1.	Conclusiones .....	95
5.2.	Recomendaciones.....	96
	REFERENCIAS .....	98
	ANEXOS .....	103

## ÍNDICE DE FIGURAS

Figura 1. PIB Sectorial .....	4
Figura 2. PIB Total .....	5
Figura 3. IPC Sectorial .....	6
Figura 4. Composición de los empleados .....	7
Figura 5. Composición de los empleados por actividad .....	7
Figura 6. Modelo conceptual de Arquitectura Multicapas .....	12
Figura 7. Interrelación entre los elementos del patrón MVC .....	17
Figura 8. Relación de abstracción entre estilos y patrones .....	18
Figura 9. Valores del desarrollo ágil .....	23
Figura 10. Agilidad en la industria del Desarrollo de Software .....	28
Figura 11. Agile vs Cascada.....	28
Figura 12. Sprint.....	31
Figura 13. Visión general del proceso .....	34
Figura 14. Equipos tradicionales vs. Un equipo ágil.....	35
Figura 15. Pruebas Tradicionales vs Agile Testing .....	36
Figura 16. Agile Testing Quadrants.....	38
Figura 17. Cronograma de las fases del proyecto.....	54
Figura 18. Cono de incertidumbre .....	60
Figura 19. Elaboración progresiva del plan .....	62
Figura 20. Cebolla de planificación .....	62
Figura 21. Condiciones de satisfacción en liberación.....	64
Figura 22. Tramo crítico de renderizado .....	66
Figura 23. Esquema de Arquitectura de Software.....	67
Figura 24. Infografía de Arquitectura de Software.....	68
Figura 25. Iteraciones de desarrollo .....	69
Figura 26. Iteraciones de desarrollo .....	70
Figura 27. Desglose de historias de usuario .....	71
Figura 28. Burndown chart (Ideal) .....	72
Figura 29. Cliente GUI (SourceTree).....	74
Figura 30. Proceso del sitio web AHOTEC.....	76
Figura 31. Diseño Plan Piloto .....	77

Figura 32. Reporte 1 visión general .....	90
Figura 33. Reporte 1 .....	91
Figura 34. Reporte 2 visión general .....	91
Figura 35. Reporte 2 .....	92
Figura 36. Reporte 3 visión general .....	92
Figura 37. Reporte 3 .....	93
Figura 38. Reporte 4 .....	93
Figura 39. Reporte de reservaciones por persona .....	94

## ÍNDICE DE TABLAS

Tabla 1. Estilos Arquitectónicos .....	10
Tabla 2. Estilo arquitectónico y patrón arquitectónico .....	15
Tabla 3. Diferencias entre metodologías ágiles y tradicionales.....	26
Tabla 4. Diferencias entre <i>Scrum</i> y modelo clásico de desarrollo .....	30
Tabla 5. Historia de usuario 1.....	41
Tabla 6. Historia de usuario 2.....	41
Tabla 7. Historia de usuario 3.....	42
Tabla 8. Historia de usuario 4.....	42
Tabla 9. Historia de usuario 5.....	43
Tabla 10. Historia de usuario 6.....	43
Tabla 11. Historia de usuario 7.....	44
Tabla 12. Historia de usuario 8.....	44
Tabla 13. Historia de usuario 9.....	45
Tabla 14. Historia de usuario 10.....	45
Tabla 15. Historia de usuario 11.....	46
Tabla 16. Historia de usuario 12.....	46
Tabla 17. Historia de usuario 13.....	47
Tabla 18. Historia de usuario 14.....	47
Tabla 19. Historia de usuario 15.....	48
Tabla 20. Historia de usuario 16.....	48
Tabla 21. Historia de usuario 17.....	49
Tabla 22. Historia de usuario 18.....	49
Tabla 23. Historia de usuario 19.....	50
Tabla 24. Historia de usuario 20.....	50
Tabla 25. Historia de usuario 21.....	51
Tabla 26. Historia de usuario 22.....	51
Tabla 27. Historia de usuario 23.....	52
Tabla 28. Historia de usuario 24.....	52
Tabla 29. Historia de usuario 25.....	53
Tabla 30. Historia de usuario 26.....	53
Tabla 31. Plan de liberación con <i>sprints</i> .....	54

Tabla 32. Planificación tradicional vs planificación ágil .....	61
Tabla 33. Situación actual y las posibles mejoras. ....	78
Tabla 34. Plantilla pruebas de aceptación.....	79
Tabla 35. Prueba de aceptación 1.....	81
Tabla 36. Prueba de aceptación 2.....	81
Tabla 37. Prueba de aceptación 3.....	82
Tabla 38. Prueba de aceptación 4.....	82
Tabla 39. Prueba de aceptación 5.....	83
Tabla 40. Prueba de aceptación 6.....	83
Tabla 41. Prueba de aceptación 7.....	84
Tabla 42. Prueba de aceptación 8.....	84
Tabla 43. Prueba de aceptación 9.....	85
Tabla 44. Prueba de aceptación 10.....	85
Tabla 45. Prueba de aceptación 11.....	86
Tabla 46. Prueba de aceptación 12.....	86
Tabla 47. Prueba de aceptación 13.....	87
Tabla 48. Prueba de aceptación 14.....	87
Tabla 49. Prueba de aceptación 15.....	88
Tabla 50. Prueba de aceptación 16.....	88
Tabla 51. Prueba de aceptación 17.....	89
Tabla 52. Prueba de aceptación 18.....	89

## INTRODUCCIÓN

En la actualidad el sector hotelero es uno de los más sobresalientes en la economía ecuatoriana, debido a su dinamismo en los últimos años. En los últimos 5 años su Producto Interno Bruto (PIB), tuvo un crecimiento promedio anual de 5.18% con una tendencia favorable, por lo que se prevé un incremento de la actividad en este año. (Maldonado y Proaño, 2015, p. 15)

De aquí surge la necesidad de otorgar un valor agregado a este importante sector, mediante el desarrollo de un aplicativo web que impulse y brinde valor a esta actividad y a sus agremiados a nivel nacional.

En este proyecto se pretende desarrollar un motor de reservas de habitaciones que muestre de manera centralizada hacia el exterior el conjunto de hoteles y atractivos turísticos potenciales del país en su conjunto.

Se busca como objetivo principal brindar estadísticas e información para el sector hotelero y gobiernos autónomos descentralizados (GAD's), mediante análisis y comparaciones, creando valor para las empresas turísticas y el turismo nacional.

En la actualidad, no existe un sitio en plataforma web que centre toda la actividad del sector de alojamiento en el país. Por esta razón, se necesita desarrollar un software de alta calidad para competir internacionalmente. De esta manera, las agencias de turismo a nivel nacional e internacional obtendrán la información necesaria del sector hotelero ecuatoriano en un solo lugar.

CAPITULO I.- En el marco teórico se explicarán los conceptos involucrados en la definición del proyecto y la metodología utilizada en el mismo.

CAPITULO II.- Se realizará el desarrollo de la ingeniería para la solución planteada.

CAPITULO III.- Aquí se realizará un análisis para conocer el estado actual del sector nacional de alojamiento. También se diseñará un plan piloto para implementar dicha solución informática.

CAPITULO IV.- En este capítulo se detallarán las pruebas de aceptación y los resultados.

CAPITULO V.- En el último capítulo se determinarán las conclusiones y recomendaciones una vez terminado el proyecto.

### **Objetivo General**

Construir un sitio web que permita recopilar y brindar estadísticas de visitas, reservas y comparaciones de precios por fechas a cada región, de tal forma que ayude a los hoteles en la toma de decisiones mediante la visualización de gráficas con datos estadísticos.

### **Objetivos específicos**

- Implementar un sitio web que ofrezca alojamiento en determinada región a todo tipo de viajero, con la selección del hotel de su preferencia y presupuesto dentro de un rango específico de fechas.
- Desarrollar y diseñar reportes estadísticos de los movimientos del sector de alojamiento, que permitan un análisis adecuado de la información.
- Generar valor de manera temprana y continua para el cliente potencial del producto, el cual se desarrollará por medio de la utilización de mejores prácticas del desarrollo ágil.
- Desarrollar un administrador de contenidos centralizado para el aplicativo.

## **CAPÍTULO I**

### **1. MARCO TEÓRICO**

#### **1.1. Sector de Alojamiento**

##### **1.1.1. Hotel**

De acuerdo al Reglamento General de Actividades Turísticas del Ecuador, hotel “es todo establecimiento que de modo habitual mediante precio preste al público en general, servicios de alojamiento, comidas y bebidas y disponga de un mínimo de treinta habitaciones”. (AHOTEC, 2002)

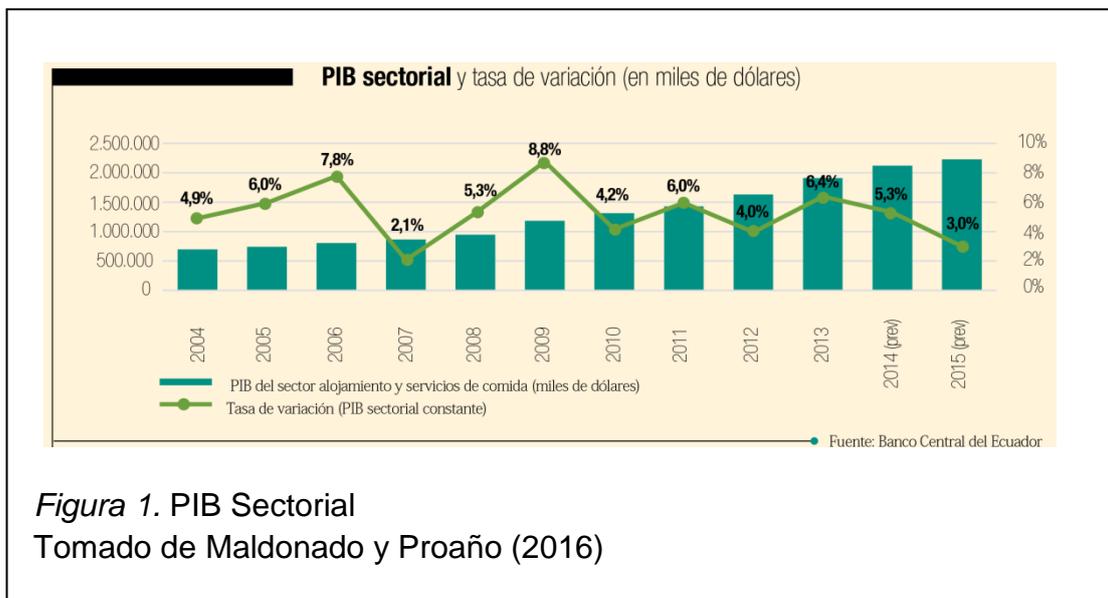
#### **1.2. Aspectos Macroeconómicos**

El sector de alojamiento es considerado un factor importante dentro de la economía ecuatoriana, ya que dinamiza la economía interna por su capacidad de generar fuentes de trabajo y por su fuerte influencia en el crecimiento de otros sectores como los de servicios públicos, la construcción, el transporte, servicios de comida y el entretenimiento (Maldonado y Proaño, 2015, p. 16).

Ecuador es considerado como una potencia turística debido a la gran biodiversidad de riquezas naturales y culturales que existen en su territorio. La clave es convertir todos estos recursos en productos turísticos, generando así un mayor ingreso de divisas para el país.

##### **1.2.1. Producto Interno Bruto (PIB)**

Maldonado y Proaño (2015, pp. 16-17) señalan que los negocios de alojamiento han experimentado un crecimiento sostenido en los últimos 10 años, llegando a considerarse como uno de los sectores más dinámicos en la economía ecuatoriana. Este crecimiento se produce debido al mayor movimiento turístico y a las inversiones realizadas en este sector.



*Figura 1. PIB Sectorial*  
Tomado de Maldonado y Proaño (2016)

Por otro lado, Maldonado y Proaño señalan que para el año 2015 existiría una misma tendencia en la economía ecuatoriana, es decir de una desaceleración económica. De esta manera se proyecta un menor nivel de crecimiento en el país en comparación a años anteriores, con una estimación para este sector con una tasa cercana al 3%.

Según estadísticas del Banco Central, la previsión del PIB sectorial del año 2014 muestra un crecimiento del sector del 5.3%, siendo 1,1% puntos porcentuales menos que el año 2013. Esto fue ocasionado, entre otros factores, por el menor dinamismo de la economía y la inversión en el país. Sin embargo, la aportación de esta industria representa un rubro importante en el PIB nacional, ya que su participación mantiene una tendencia creciente. El PIB en dólares corrientes para el año 2015 se estima que supere los 2.229.100 millones de dólares.

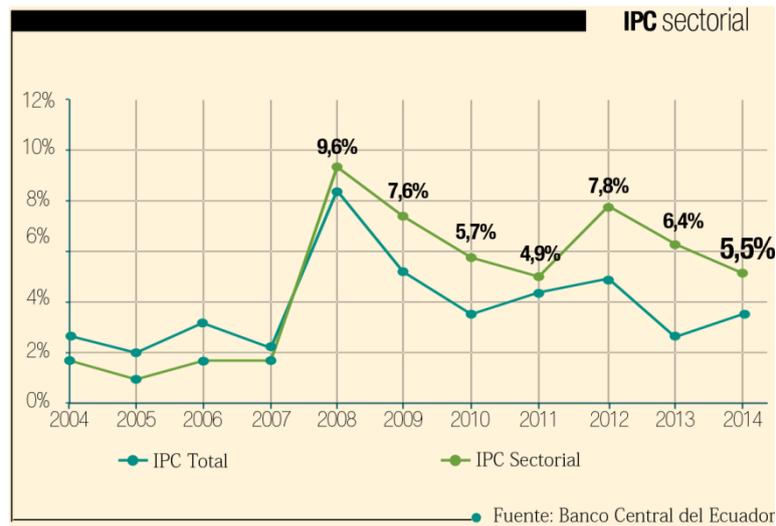


### 1.2.2. Inflación

En cuanto a la inflación sectorial, se observa que esta variable llega a sus picos más altos a partir del año 2008. Este efecto ocurre principalmente por el incremento del salario básico unificado y ciertas normativas vigentes desde el mismo año.

“En esta coyuntura se destacan las dificultades que presenta el sector, frente a los desafíos que tiene para realizar los ajustes de precios, lo cual incide en su competitividad. Para el año 2015, estas presiones van a seguir para el sector.” (Maldonado y Proaño, 2015, p. 17)

Esta visión empata con la coyuntura actual, que refleja un alto nivel de incertidumbre y cautela por los bajos precios del petróleo a nivel mundial, que repercuten en el Ecuador.



**Figura 3. IPC Sectorial**  
Tomado de Maldonado y Proaño (2016)

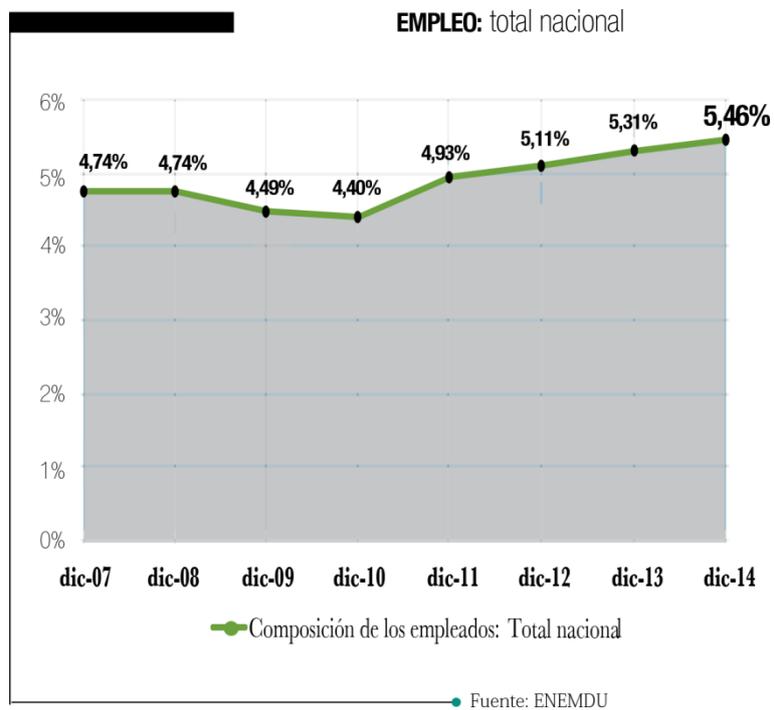
### 1.2.3. Empleo

Un factor importante que indica la importancia del sector de alojamiento en la economía ecuatoriana, es el empleo. (Maldonado y Proaño, 2015, p.18)

Según encuestas trimestrales realizadas por el Instituto Nacional de Estadísticas y Censos (INEC), señalan el peso que tiene este sector en el total de trabajadores, con un 5.46% en el empleo total del país.

Respecto a la composición de los empleados por esta actividad en el año 2013, según la Superintendencia de Compañías se observa que 25.557 personas trabajan en el sector de alojamiento y restaurantes; de los cuales el 67,5% se concentra en los negocios de restaurantes y el 32,5% en el sector de alojamiento.

Es decir que aproximadamente 8.300 personas están inmersas directamente en el sector de alojamiento.



*Figura 4. Composición de los empleados*  
Tomado de Maldonado y Proaño (2016)

**COMPOSICIÓN DE LOS EMPLEADOS POR ACTIVIDAD: Total nacional (2013)**



*Figura 5. Composición de los empleados por actividad*  
Tomado de Maldonado y Proaño (2016)

### 1.3. Arquitectura de Software

A partir del desarrollo del software, desde la década de 1990, muchos autores comenzaron a proponer sus teorías, conceptos y enfoques para poder convertir a la Arquitectura de Software en una disciplina. Se mencionará a algunos de los más destacados, quienes dieron grandes aportes a la Arquitectura de *Software* hasta la actualidad.

Primeramente, para Kazman la Arquitectura de *Software* es una forma de representar sistemas complejos mediante el uso de la abstracción. Por lo que planteó que la necesidad del diseño y el análisis de las arquitecturas de *software* llevarán al desarrollo de herramientas *Computer Aided Software Engineering* (CASE), por sus siglas en inglés, o Ingeniería de Software Asistida por Computadora. Por tanto, estas herramientas permitirán hacer el diseño y el análisis de las arquitecturas de software una actividad más confiable y mejor documentada (Kazman, 1999).

Bredemeyer propuso que los requerimientos arquitectónicos son necesarios para guiar las actividades de estructuración de un sistema de *software*. El evaluar tempranamente la arquitectura de *software* provee indicadores que permiten tener la oportunidad de resolver problemas que puedan darse a nivel arquitectónico (Bredemeyer, 2002).

Asimismo, Bredemeyer señala que independientemente de la metodología implementada, lo que se debe obtener es una arquitectura con la documentación necesaria, y de esta forma asegurar que el sistema cumpla con las especificaciones funcionales solicitadas por el cliente o usuario final.

Según Paul Clements y otros autores, una arquitectura de *software* es la estructura o estructuras del sistema, que consiste en los elementos, sus propiedades, y las relaciones entre ellos (Clements et al., 2010, p.49).

Paul Clements es Miembro Senior del equipo técnico en *Carnegie Mellon Software Engineering Institute* (SEI), donde ha trabajado desde 1994 liderando proyectos en la línea de ingeniería y arquitectura de *software*. Además, Clements es coautor de varios libros y docenas de artículos quien desde su punto de vista menciona que la “Arquitectura de *Software* es, a grandes rasgos, una vista del sistema que incluye los componentes principales del mismo, la conducta de sus componentes y las formas en que los mismos interactúan y se coordinan para alcanzar la misión del sistema” (Caicedo, Peña y Hernández, 2015).

Así es como se puede presentar un breve repaso de los conceptos del término Arquitectura de *Software*, cada una con varios planeamientos pero a su vez complementarios. Es incuestionable que su definición está aún en discusión, ya que no se puede encontrar en una sola fuente su significado, y menos aún hay un estándar que pueda ser utilizado globalmente.

Cabe recalcar, que existe toda una comunidad (*Community Software Architecture Definitions*), conformada por arquitectos expertos a nivel mundial, en la que se enlistan cientos de definiciones de Arquitectura de *Software*.

### 1.3.1. Estilo Arquitectónico

Los estilos arquitectónicos ayudan a construir bases fundamentales para la estructuración de un determinado software que ayude a promover buenas prácticas de diseño, gracias a la experiencia colectiva de ingenieros especializados que ya tienen soluciones efectivas utilizadas anteriormente.

Cada estilo se encarga de un problema específico, que puede repetirse en el diseño o en la implementación de software. A su vez, los estilos pueden ser usados para construir arquitecturas de software con propiedades específicas (Buschmann, 1996, p. 1).

Respecto a la clasificación general de los estilos, se elaboró la siguiente tabla:

Tabla 1. Estilos Arquitectónicos

<b>Estilo de Flujo de Datos</b>	<b>Estilos de Código Móvil</b>
<ul style="list-style-type: none"> <li>• Tubería y filtros</li> </ul>	<ul style="list-style-type: none"> <li>• Arquitectura de máquinas virtuales</li> </ul>
Estilos Centrados en Datos <ul style="list-style-type: none"> <li>• Arquitecturas de Pizarra o Repositorio</li> </ul>	Estilos Heterogéneos <ul style="list-style-type: none"> <li>• Sistemas de control de procesos</li> <li>• Arquitecturas basadas en atributos</li> </ul>
Estilos de Llamada y Retorno <ul style="list-style-type: none"> <li>• Modelo Vista Controlador (MVC)</li> <li>• Arquitectura en capas</li> <li>• Arquitecturas orientadas</li> </ul>	Estilos Peer-to-Peer <ul style="list-style-type: none"> <li>• Arquitecturas basadas en eventos</li> <li>• Arquitecturas orientadas a servicios</li> </ul>

Estilo de Flujo de Datos	Estilos de Código Móvil
a objetos <ul style="list-style-type: none"> <li>● Arquitecturas basadas en componentes</li> </ul>	<ul style="list-style-type: none"> <li>● Arquitecturas basadas en recursos</li> </ul>

Tomado de Acuña, J. 2014.

### 1.3.2. Estilos de Llamada y Retorno

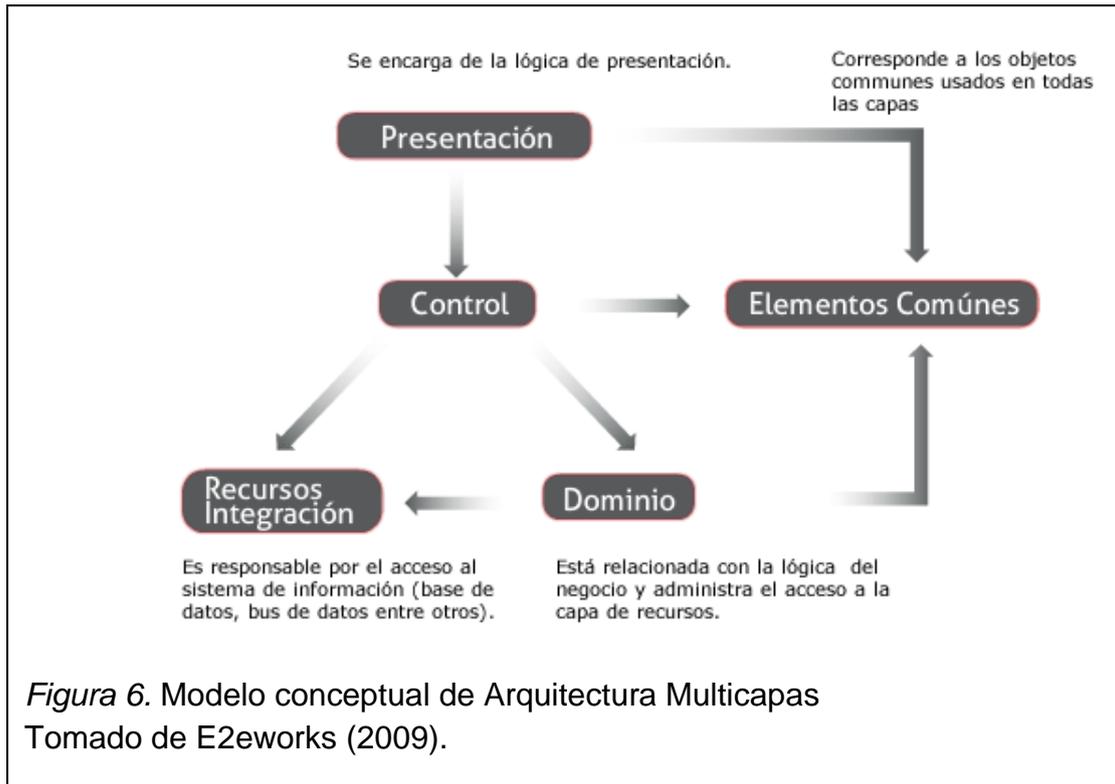
Este estilo arquitectónico hace énfasis en la escalabilidad y posibilidad de modificación, es decir que los cambios que puedan presentarse, se los puede llevar a cabo con mayor facilidad. Generalmente estos estilos se aplican en soluciones informáticas de gran tamaño.

Existen los siguientes sub-estilos:

- Arquitecturas en capas.
- Arquitecturas de programa principal.
- Arquitecturas de llamada de procedimiento remoto.
- Arquitecturas orientadas a objetos.

### 1.3.3. Arquitectura Multicapas

Este modelo es un conjunto ordenado de subsistemas basado en una estrategia de asignación de responsabilidades, cuyo objetivo es la separación de la lógica de negocios de la lógica de diseño. “De esta forma cada uno de los niveles tiene solamente la lógica que le compete” (E2eworks, 2009).



La arquitectura multicapas está compuesta por **capas y niveles**. Las capas son: capa de presentación, capa de negocio y la capa de datos

- **Capa de presentación**

Es la encargada de la interacción del sistema con el usuario final, es a su vez conocida como la interfaz gráfica (*user interface*) la misma que debe ser de fácil entendimiento para el usuario (Ecured, s.f.).

- **Capa de negocio (Middle Tier)**

Esta capa también denominada lógica, se encarga de tomar las peticiones realizadas por el usuario y enviar respuestas.

## Capa de datos

También denominada de almacenamiento, es donde reside la información y todos los elementos dedicados a la persistencia del sistema.

Los **niveles** hacen referencia a las capas lógicas y como se encuentran posicionadas físicamente, es decir:

- Una solución de tres capas (capa de presentación, lógica de negocio y capa de datos) ubicadas en un solo ordenador. Esta es una arquitectura de solución de tres capas y un nivel
- Una solución de tres capas (capa de presentación, lógica de negocio y capa de datos) ubicadas en dos ordenadores. Esta es una arquitectura de solución de tres capas y dos niveles.

Las ventajas de las arquitecturas multicapa son:

- Centralización del control: a nivel de acceso e integridad de la información.
- Escalabilidad: según el número de clientes y para no ver afectado el rendimiento del mismo se pueden aumentar o mejorar los nodos, los mismos que pueden ser clientes o servidores.
- Mantenimiento: al existir una redundancia de equipos con las mismas o diferentes responsabilidades esto hace posible su mantenimiento más fácil. (Janium, s.f.)

Un claro ejemplo del uso de esta arquitectura es Google, con el manejo de protocolos estandarizados que permiten flexibilidad, portabilidad y escalabilidad. Por ejemplo: millones de clientes realizando peticiones a sus servidores. Esta situación llevó a que

contemplaran dentro de su arquitectura multicapa servidores de indexación, servidores de documentos y servidores de balance entre otros (SlideShare, s.f.).

#### 1.3.4. Patrón Arquitectónico

Para Buschmann (1996, p. 3) cada patrón arquitectónico es una regla de tres partes, que expresa una relación entre:

- Contexto: Situación de diseño que da lugar a un problema de diseño (Buschmann, 1996, p. 11).
- Problema: Conjunto de fuerzas que se producen repetidamente en el contexto (Buschmann, 1996, p. 11).
- Solución: Estructura con componentes y relaciones (Buschmann, 1996, p. 11).

Entonces, los patrones arquitectónicos son instrucciones que muestran cómo resolver problemas particulares y recurrentes de diseño, dados por un sistema de fuerzas, donde el contexto hace que sea relevante dicha situación. (Buschmann, 1996, pp. 2-3).

Por lo tanto, en definitiva un patrón arquitectónico son situaciones que ocurren en el mundo del desarrollo, e indican una serie de reglas que describen y señalan el proceso para crear de manera correcta determinados objetos (Buschmann, 1996, p. 3).

Respecto a las diferencias entre estilo arquitectónico y patrón arquitectónico, se elaboró la siguiente tabla:

Tabla 2. Estilo arquitectónico y patrón arquitectónico

<b>Estilo Arquitectónico</b>	<b>Patrón Arquitectónico</b>
Describe la estructura general para el desarrollo de aplicaciones	Puntualizan la estructura básica de una aplicación
Son autónomos e independientes del contexto al que puedan ser ejecutados	Requiere del detalle del contexto de un problema
Cada uno de los estilos es independiente del resto	Depende de patrones reducidos, los cuales contienen a patrones con los que interactúa u otros que lo contengan
Hablan de técnicas de diseño desde una perspectiva independiente de la situación actual del diseño	Expresa un constante problema de diseño y presenta una solución para el mismo, dependiendo el punto de vista del contexto en el que se presenta
Pueden ser considerados como una clasificación de sistemas	Parte de soluciones generales que resuelven problemas comunes

Tomado de Kazman et al., 1999

### 1.3.5. Patrón Arquitectónico Modelo Vista Controlador

Al momento de desarrollar un sistema interactivo, hay que tener en cuenta dos aspectos:

- Los cambios en la interfaz de usuario deben ser fáciles, y posibles en un tiempo corto de ejecución.

- La adaptación de la interfaz de usuario no debe afectar el código en su núcleo funcional de la aplicación (Buschmann, 1996, p. 4).

“El patrón MVC es un patrón de arquitectura de *software* encargado de separar la lógica de negocio de la interfaz del usuario y es el más utilizado en aplicaciones web, ya que facilita la funcionalidad, mantenibilidad y escalabilidad del sistema” (Bahit, 2011, p. 36).

Según Steven Burbeck, como se lee en la página de la Pontificia Universidad Javeriana de Cali, “el patrón *Model-View-Controller* (MVC), separa el modelado del dominio, la presentación y las acciones basados en las entradas del usuario en tres clases apartes”.

Ante lo expuesto anteriormente, se concluye que el patrón arquitectónico Modelo Vista Controlador, es un estándar para el diseño de aplicaciones el cual divide sus partes en el Modelo, las Vistas y los Controladores. Esto permite reutilizar código y efectuar un mantenimiento de aplicaciones más sencillo, así como tener escalabilidad y flexibilidad.

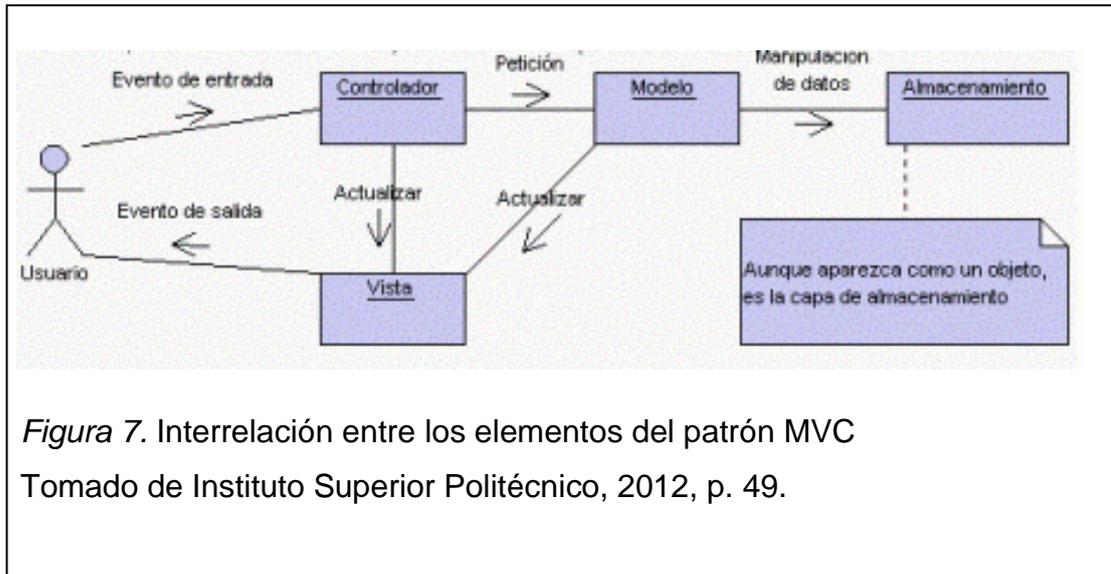


Figura 7. Interrelación entre los elementos del patrón MVC

Tomado de Instituto Superior Politécnico, 2012, p. 49.

### 1.3.6. Patrón de Diseño

Buschman (1996, p. 346) señala que un Patrón de Diseño representan patrones de bajo nivel, y describen cómo resolver efectivamente problemas específicos de la implementación en un determinado lenguaje de programación.

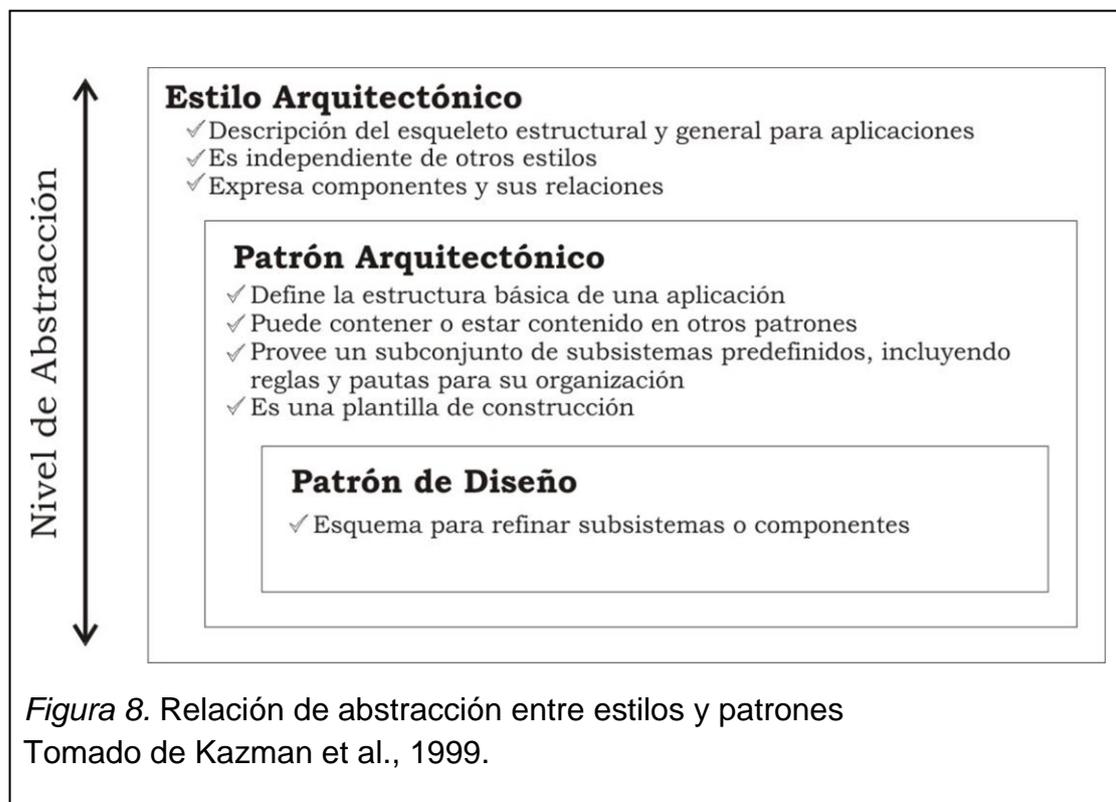
Al desarrollar aplicaciones de software pueden darse casos de encontrar problemas y resolverlos una y otra vez, incluso dentro de un mismo proyecto. Por esta razón, los patrones de diseño son los encargados de definir una solución general a inconvenientes específicos.

Para Christopher Alexander,

*“Each pattern describes a problem which occurs over and over again in our environment, and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice.”* (Citado en Gamma et al., 1997)

Entonces, a partir de estas definiciones se puede decir que cada Patrón de Diseño describe problemas que ocurren reiteradamente en el desarrollo de *software* y cada uno describe la solución a dichos problemas; de esta manera se podrá reutilizar el código sin hacer dos veces lo mismo.

A continuación se presentará una figura donde se muestra el nivel de abstracción entre los conceptos mencionados anteriormente:



#### 1.4. Gestión Ágil de Proyectos de Software

“La agilidad es la capacidad de diseñar los pasos a seguir en función del aprendizaje que surge de los resultados producidos por los pasos anteriores y de la habilidad de adaptarse a los cambios del contexto, con el fin de alcanzar los objetivos propuestos de manera efectiva” (Caminoagil, s.f.).

Pero para definir qué es agilidad dentro de un proyecto de desarrollo de software, es necesario desglosar los siguientes conceptos:

- **Proyecto**

“Un proyecto es un esfuerzo temporal emprendido para satisfacer un objetivo o resultado de negocio a través de la creación de un producto, servicio o resultado único” (Americapmi, s.f.).

“Un proyecto es un esfuerzo temporal emprendido para crear un producto, servicio o resultado único” (Managedagile, s.f.).

- **Gestión de proyectos**

“Es la aplicación de conocimientos, habilidades, técnicas y herramientas para ejecutar proyectos en forma eficiente y efectiva” (Americapmi, s.f.).

- **Gestión ágil de proyectos**

“Es la aplicación de conocimientos, habilidades, técnicas y herramientas dentro del contexto del entorno de negocio para maximizar el valor de lo que el proyecto produce” (Managedagile, s.f.).

Varios autores mencionan y recalcan que quizá ya no hay productos de software terminados en su totalidad, es decir, productos finales, sino que existen productos en continua evolución y mejora. Finalmente, uno de ellos concluye que “la gestión de proyectos ágil no se formula sobre la necesidad de anticipación, sino sobre la de adaptación continua” (Palacio, 2014).

En conclusión, la gestión ágil de proyectos de software consiste en aplicar las técnicas adecuadas sobre un proyecto de software garantizando entregables de calidad en tiempos óptimos.

### 1.4.1. Objetivos de la gestión ágil

*“Una teoría debe ser lo más simple posible pero no simplista”*  
Albert Einstein.

Según Palacio (2014), los objetivos de la gestión ágil son:

- **Valor**

El valor consiste en la innovación continua y flexibilidad a los cambios que puedan presentarse en un proyecto de software.

El principal valor para las empresas que promueven el desarrollo ágil es la capacidad de innovar, es decir, incluyendo nuevas y mejores ideas.

- **Reducción del tiempo de desarrollo**

Se basa en desarrollar un entregable de calidad, en el tiempo más corto posible, sin olvidarse que un producto puede estar en continua evolución.

La reducción del tiempo es un factor competitivo de primer orden.

- **Agilidad y flexibilidad**

Este objetivo se basa en responder rápidamente a los cambios solicitados en un determinado proyecto, tomando dichos cambios como actualizaciones o mejoras continuas.

- **Fiabilidad**

Un modelo de gestión ágil es fiable si los resultados son entregados a tiempo y tienen valor innovador.

La gestión ágil funciona porque:

- Se realizan planes en diferentes niveles.
- Re-planificación ocurre frecuentemente.
- Los planes se basan en *features*(unidades de valor para el cliente), y no en tareas.
- Se estima primero el tamaño; mientras que la duración se deriva del tamaño estimado.
- El trabajo se elimina en cada iteración.
- Se mide el progreso del equipo completo y no individualmente.
- La incertidumbre es conocida y la planificamos.

## 1.5. Metodologías ágiles

Lo ágil se define “como la habilidad de responder en forma versátil al cambio para maximizar los beneficios” (Citón, 2006, pp. 10-35).

Este tipo de metodologías tuvieron origen a inicios del año 2001, donde 17 empresarios de la industria del software se reunieron con el fin de establecer principios y valores que se deben cumplir en el desarrollo de software, haciendo énfasis en la calidad, tiempos cortos y flexibilidad a los cambios.

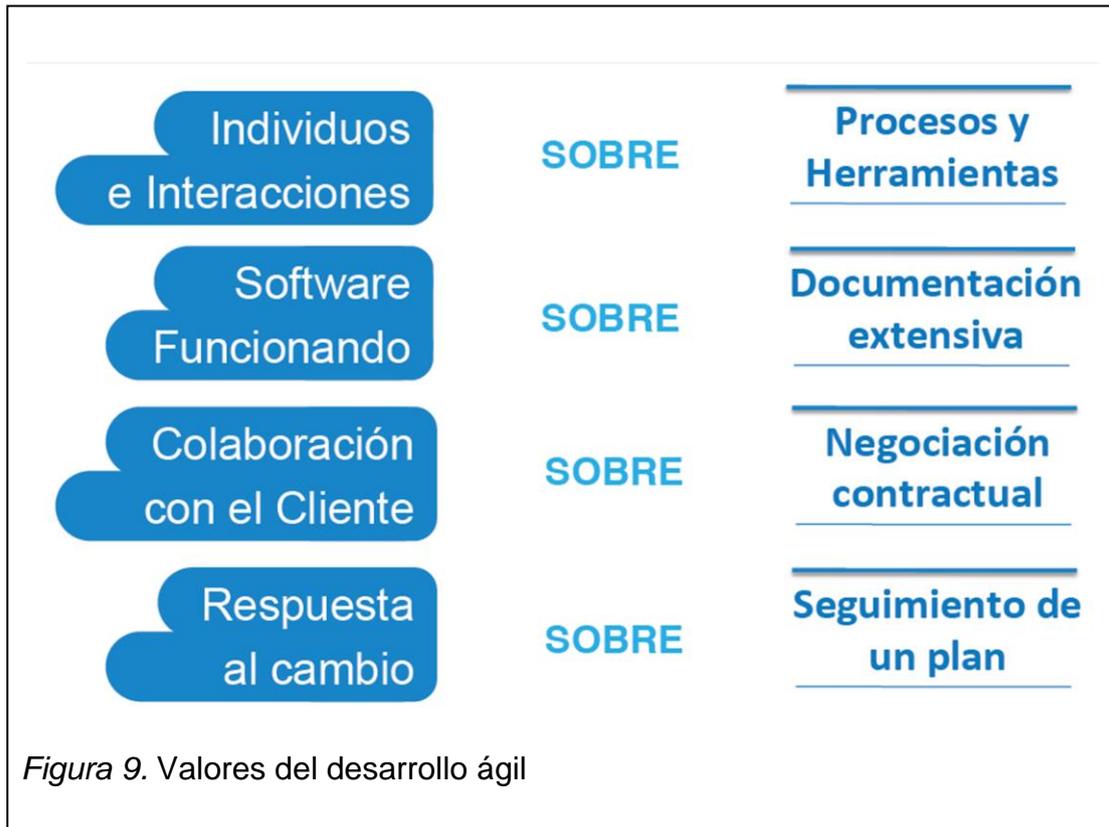
Los integrantes de la reunión resumieron la filosofía ágil en un documento denominado manifiesto ágil. Además, se creó “*The Agile Alliance*”, una organización enfocada a promover el desarrollo ágil de *software*.

### 1.5.1. Manifiesto Ágil

Las metodologías ágiles varían en su forma de responder al cambio, pero tienen características o valores similares:

- Los individuos y sus interacciones son más importantes que los procesos y las herramientas: las personas son el principal factor de éxito en un proyecto de software, es decir, si existe un buen equipo se puede llegar con mucha más facilidad al objetivo final. Sin embargo, no se debe dejar de lado los procesos, razón por la cual es importante que los mismos se adapten a los equipos multidisciplinarios y no al revés.
- El software que funciona es más importante que la documentación exhaustiva: este valor no afirma que no sea necesaria la documentación, más bien aclara que es un punto fundamental para transferir conocimiento, registrar información, entre otros. Resalta que tienen menos importancia que los productos que funcionan.
- La colaboración con el cliente en lugar de la negociación de contratos: en el desarrollo ágil se considera más importante los resultados de cada iteración que tener un documento con las normativas o garantías de un contrato. “Un contrato no aporta valor al producto” (Palacio, 2007, p. 50).
- La respuesta al cambio en lugar de aferrarse a un plan: la habilidad de responder a los cambios, la evolución rápida y continua y la capacidad de respuesta son los puntos fuertes de las metodologías ágiles, por su manera de adaptarse ante cualquier situación. “Los principales valores de la gestión ágil son la anticipación y la adaptación” (Palacio, 2007, p. 50).

A continuación en la figura se observa una síntesis de los principales valores descritos, que según Palacio (2007 p. 48), son los principales puntos para desarrollar *software*.



Una vez descritos los cuatro valores, de estos se derivaron los doce principios del manifiesto (Palacio, 2007, pp. 51-54).

***I. Nuestra principal prioridad es satisfacer al cliente a través de la entrega temprana y continua de software de valor***

Un proceso es ágil si se entrega lo máximo y con el mayor valor que se pueda hacer, en un corto plazo, es decir, a una fecha determinada por el cliente.

***II. Son bienvenidos los requisitos cambiantes, incluso si llegan tarde al desarrollo. Los procesos ágiles se doblan al cambio como ventaja competitiva para el cliente***

Los cambios solicitados por el cliente son capturados como un valor agregado al producto final, y a su vez, se logra una mayor satisfacción del cliente.

**III. *Entregar con frecuencia software que funcione, en periodos de un par de semanas hasta un par de meses, con preferencia en los periodos breves***

El objetivo principal de la gestión ágil consiste en entregar de manera temprana software que genere valor para el producto final.

**IV. *Las personas del negocio y los desarrolladores deben trabajar juntos de forma cotidiana a través del proyecto***

En el proceso de desarrollo de software es necesario trabajar en conjunto con el cliente, es decir, las personas que tienen el conocimiento del negocio.

**V. *Construir el proyecto en torno a individuos motivados, dándoles la oportunidad y el respaldo que necesitan y confiar en ellos para que lleven a cabo sus tareas***

El personal es un factor clave y el activo más importante en una empresa, por lo que siempre debe existir una motivación adecuada para cada uno de ellos.

**VI. *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***

Los miembros del equipo deben comunicarse directamente para transferir con mayor precisión cualquier tipo de mensaje. Y así evitar posibles ambigüedades en una comunicación escrita.

**VII. *El software que funciona es la principal medida del progreso***

En la gestión ágil se mide el progreso por entregables de código que funcione y que aporte algún valor para el objetivo final. Su principal métrica es ver partes terminadas.

A las métricas ágiles lo que les interesa es que si ya se ha terminado y funciona un producto satisfactoriamente.

**VIII. *Los procesos ágiles promueven el desarrollo sostenible***

La gestión ágil señala que desarrolladores y usuarios deben tener la capacidad de mantener un ritmo de trabajo constante durante un determinado proyecto, manteniendo a todo momento la calidad del producto.

**IX. *La atención continua a la calidad técnica y al buen diseño mejora la agilidad***

Un factor importante es la calidad técnica, la cual define todo lo relacionado a la estructura del producto y que a su vez, definirá su robustez y flexibilidad al cambio, características principales en un entorno ágil.

**X. *La simplicidad es esencial***

Desarrollar código simple y de alta calidad permitirá adaptarlo a posibles cambios.

**XI. *Las mejores arquitecturas, requisitos y diseños emergen de equipos que se auto-organizan***

Este punto consiste en el trabajo de equipo especializado y auto-organizado, lo que permitirá mejores resultados de diseño y arquitectura.

***XII. En intervalos regulares, el equipo reflexiona sobre la forma de ser más efectivo y ajusta su conducta en consecuencia***

El equipo debe estar siempre dispuesto a adaptarse a un posible nuevo escenario con el fin de mejorar el modelo ágil empleado.

En conclusión, los dos primeros principios del manifiesto describen la importancia del cliente y la eficiencia con la que se debe cumplir alguna tarea encomendada. Luego existen una serie de principios que señalan el proceso de desarrollo de software a seguir. Para finalizar, los últimos principios destacan la importancia del trabajo en equipo.

### **1.5.2. Metodologías Ágiles vs Tradicionales**

Para las diferencias entre las metodologías ágiles y las metodologías tradicionales, se debe tomar en cuenta no solo el proceso que conlleva cada una de ellas, sino también considerar el contexto de equipo, que pueden ser las siguientes metodologías que están mostradas en la tabla a continuación:

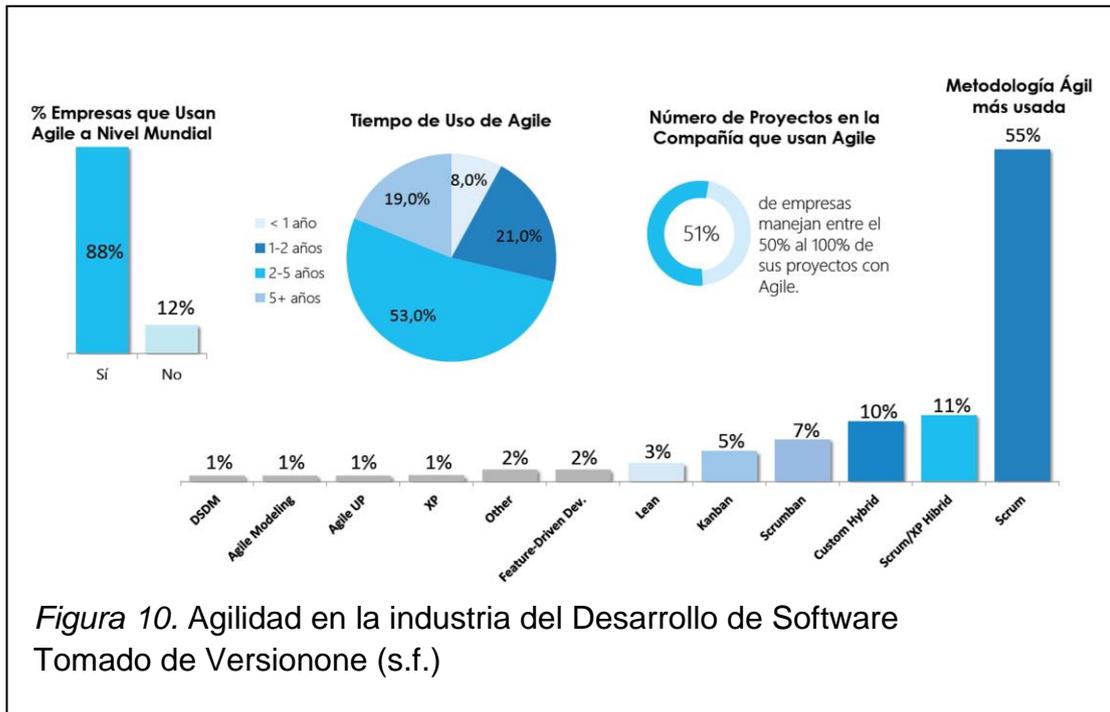
Tabla 3. Diferencias entre metodologías ágiles y tradicionales

<b>Metodologías Ágiles</b>	<b>Metodologías Tradicionales</b>
La planificación del trabajo sólo comprende el ciclo en el que se está trabajando.	Trabajo y gestión guiada por un plan general del proyecto que compete todo un ciclo de desarrollo.

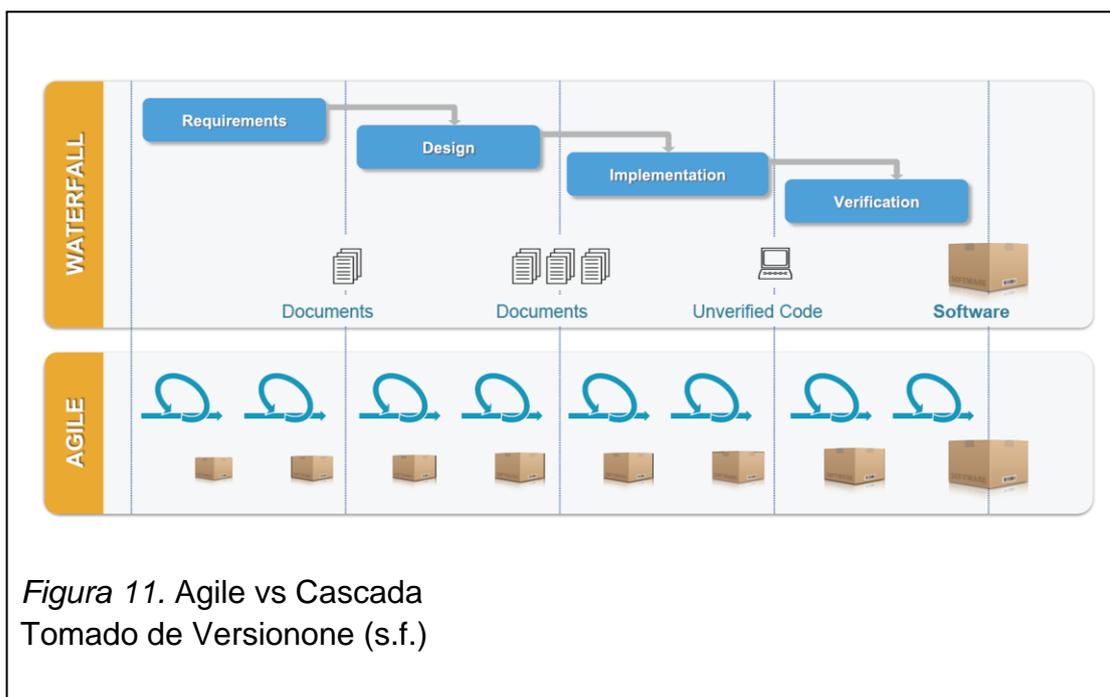
<b>Metodologías Ágiles</b>	<b>Metodologías Tradicionales</b>
Descubrimiento progresivo de requisitos, e incorporación de cambios en cualquier iteración del desarrollo.	Conocimiento detallado de los requisitos antes de comenzar el diseño del proyecto. “Hacerlo bien a la primera”
Refactorización de código como modelo de trabajo compatible con el punto anterior.	Evitar la re-codificación y el re-trabajo que supone una pérdida de eficiencia.
Comunicación directa entre los integrantes del equipo.	Comunicación formal según el plan de comunicación del proyecto.
Equipos auto-gestionados.	Gestión de equipos y personas centralizada en el gesto del proyecto.
No existe contrato tradicional o al menos es bastante flexible.	Existe un contrato prefijado.
El cliente es parte del equipo de desarrollo	El cliente interactúa con el equipo de desarrollo mediante reuniones.
Grupos pequeños, trabajado en el mismo sitio	Grupos grandes y posiblemente distribuidos.
Pocos roles.	Más roles.

Tomado de Citón, 2006, p.17.

Las metodologías ágiles y su constante evolución, han revolucionado la manera de desarrollar software, y a su vez han generado un amplio debate con los partidarios de usar metodologías tradicionales.



La figura anterior especifica la acogida a nivel mundial de *Agile* y las metodologías más usadas en la industria del desarrollo de *software*. Estos datos fueron expuestos en la octava encuesta “*State of Agile*”, llevada a cabo entre el 04 de agosto al 16 de octubre de 2013. Hubo un total de 3.501 respuestas recolectadas entre todas las comunidades de desarrollo.



En la figura anterior se muestra software liberado continuamente con el valor que el usuario final espera usando una metodología ágil, y su comparativa de entregables usando una metodología tradicional.

### 1.5.3. Scrum

La palabra *Scrum* es un término usado en el juego *rugby*, la cual se define como la acción de avanzar en equipo pasando un balón entre todos los jugadores. Esta palabra a su vez fue utilizada por Jeff Sutherland para definir un nuevo marco de trabajo ágil en el desarrollo de software, por la analogía del trabajo en equipo. El cual consiste en entregar software funcional en períodos cortos de tiempo.

“*Scrum* es un marco de trabajo para la gestión y desarrollo de productos complejos, en un proceso iterativo e incremental utilizado comúnmente en entornos donde existe gran incertidumbre” Twileshare (s.f.).

Según Schwaber y Sutherland (2011, p. 5), “*Scrum* consiste en los equipos *Scrum* y en sus roles, eventos, artefactos y reglas asociadas. Cada componente dentro del marco de trabajo sirve a un propósito específico y es esencial para el éxito de *Scrum* y para su uso”.

Entonces, a partir de estas definiciones se puede decir que *Scrum* es un marco de trabajo cuyo objetivo es aumentar al máximo la productividad del equipo de desarrollo, mediante un proceso iterativo e incremental que permite construir el mejor producto posible ajustando las necesidades y entregando valor visible con cada iteración.

Tabla 4. Diferencias entre *Scrum* y modelo clásico de desarrollo

<b>Desarrollo Tradicional</b>	<b>Desarrollo Ágil</b>
Especialización	Equipo multidisciplinario
Fases	Solapamiento
Requisitos detallados	Visión del producto
Seguimiento del plan	Adaptación a los cambios

Tomado de Palacio, 2007, p. 40.

En la tabla anterior, se muestra las principales diferencias entre el desarrollo tradicional y el desarrollo ágil empleando el marco de trabajo *Scrum*. De estas se destaca el equipo multidisciplinario, es decir, un equipo conformado por profesionales competentes con conocimientos en diferentes áreas que pueden sacar adelante un producto.

#### 1.5.4. Visión general del proceso

Al tratarse de un proceso constructivo, iterativo e incremental hasta llegar a la entrega de un producto existe una serie de iteraciones denominadas *Sprint*.

Un *sprint* “es el período de tiempo durante el que se desarrolla un incremento de funcionalidad. Constituye el núcleo de *Scrum*, que de esta forma divide el desarrollo de un proyecto en un conjunto de pequeñas carreras” (Citón, 2006, p. 26).

Para Schwaber y Jeff Sutherland (2011, p. 8), “el corazón de *Scrum* es el *sprint*, un bloque de tiempo de un mes o menos, durante el cual se crea un incremento de producto utilizable potencialmente entregable”.

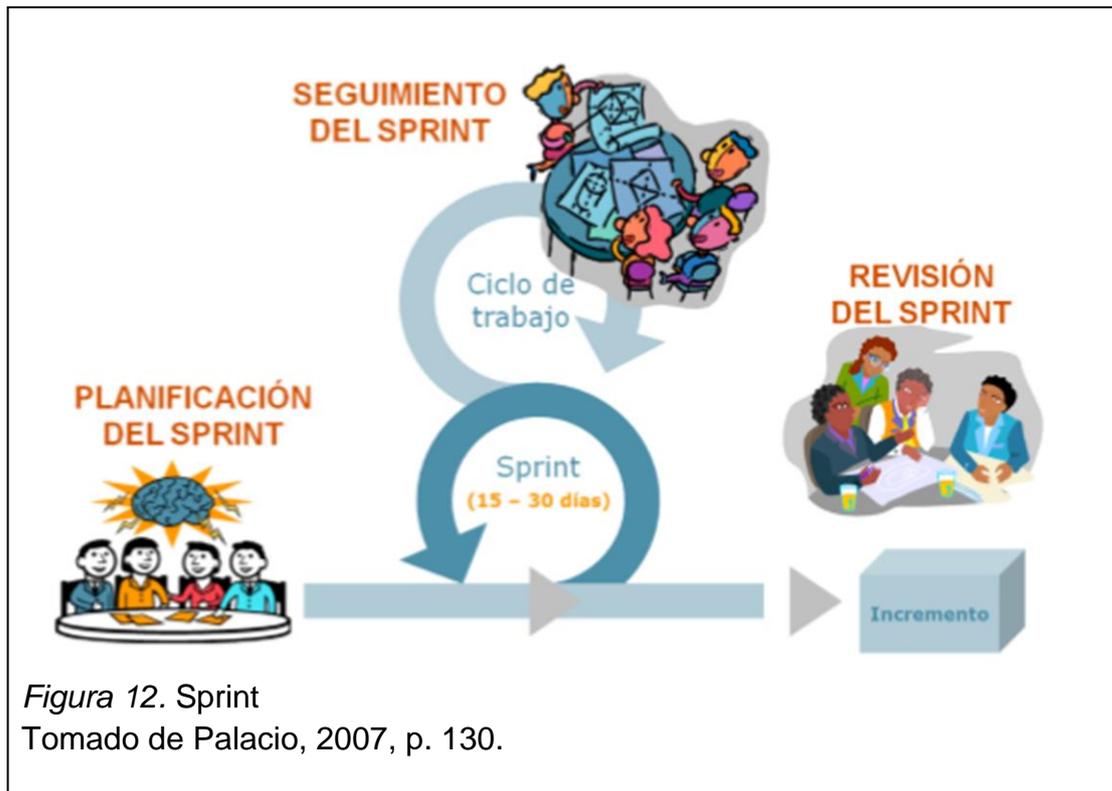


Figura 12. Sprint  
Tomado de Palacio, 2007, p. 130.

Por tanto, en la imagen anterior, el ciclo del *sprint* presenta el comienzo de cada iteración en el que se determina qué partes se van a desarrollar, “tomando como criterios la prioridad para el negocio y la cantidad de trabajo que se podrá abordar durante la iteración” (Palacio, 2007, p. 129).

Para Palacio (2007, pp. 129-130) los conceptos empleados en *Scrum* son:

- **Planificación del *sprint* (*Sprint Planning Meeting*)**

Reunión de planificación del *sprint* en la que se determina las unidades de valor a desarrollar y los objetivos a cubrirse en esa iteración por todo el equipo *Scrum*. El conjunto de unidades de valor (*Features*) se denomina *sprint backlog*.

La reunión de planificación del *sprint* tiene una duración de 8 horas para un *sprint* de un mes. En caso de haber *sprints* más

cortos el tiempo es proporcional y siempre estará dividida por la mitad de la totalidad del tiempo; se dará respuesta a las siguientes preguntas:

- ¿Qué será entregado en el incremento resultante del sprint que comienza?
- ¿Cómo se conseguirá hacer el trabajo necesario para entregar el incremento?

- **Seguimiento del *sprint* (*Daily Scrum*)**

*Scrum* diario es una reunión con un límite de 15 minutos para revisar el avance de cada actividad y crear un plan para las siguientes 24 horas.

Esta reunión debe ser llevada a cabo en el mismo lugar y a la misma hora todos los días; allí solo intervienen los miembros del equipo, quienes explican lo siguiente:

- ¿Qué se ha conseguido desde la última reunión?
- ¿Qué se hará antes de la próxima reunión?
- ¿Qué obstáculos se encuentran en el camino?

- **Revisión del *sprint* (*Sprint Review*)**

Es una reunión informal al final del *sprint* en la que el dueño del producto realiza un análisis y revisión del incremento generado. Tiene una duración de 4 horas para *sprints* de un mes. En caso de haber *sprints* más cortos el tiempo es proporcionalmente menor.

La cancelación de un *sprint* causa un gran impacto en el equipo *Scrum*, lo cual es muy poco común. De darse este caso cada uno de los elementos de la pila que se hayan llevado a cabo deber ser revisados; los que estén por realizarse se vuelven a estimar.

La decisión de cancelación del *sprint* solo puede ser tomada por el dueño del producto (*product owner*).

### 1.5.5. Artefactos de Scrum

Para Schwaber y Sutherland (2011, pp. 13-16), *Scrum* parte de tres artefactos fundamentales:

- ***Product Backlog***

Para Schwaber (2011, p, 13), en pocas palabras, “la pila del producto lista todas las características, funcionalidades, requerimientos, mejoras y correcciones” a realizar para cumplir el objetivo del producto.

El Dueño del Producto (*Product Owner*), es el responsable de este inventario dinámico de características, el cual es ordenado por valor, riesgo, prioridad y necesidad. Estos elementos a su vez tienen los siguientes atributos: descripción, ordenación y estimación.

- ***Sprint Backlog***

Enumera las unidades de valor que se llevarán a cabo durante el *sprint* para entregar el incremento de producto deseado. Cada una de las tareas enlistadas deberá ser estimada en horas, en un rango de 1 a 16.

- **Incremento**

Es el resultado de cada *sprint*, es decir el incremento de funcionalidad totalmente utilizable. Una vez probado exhaustivamente, el dueño del producto es el encargado de liberar cada incremento.

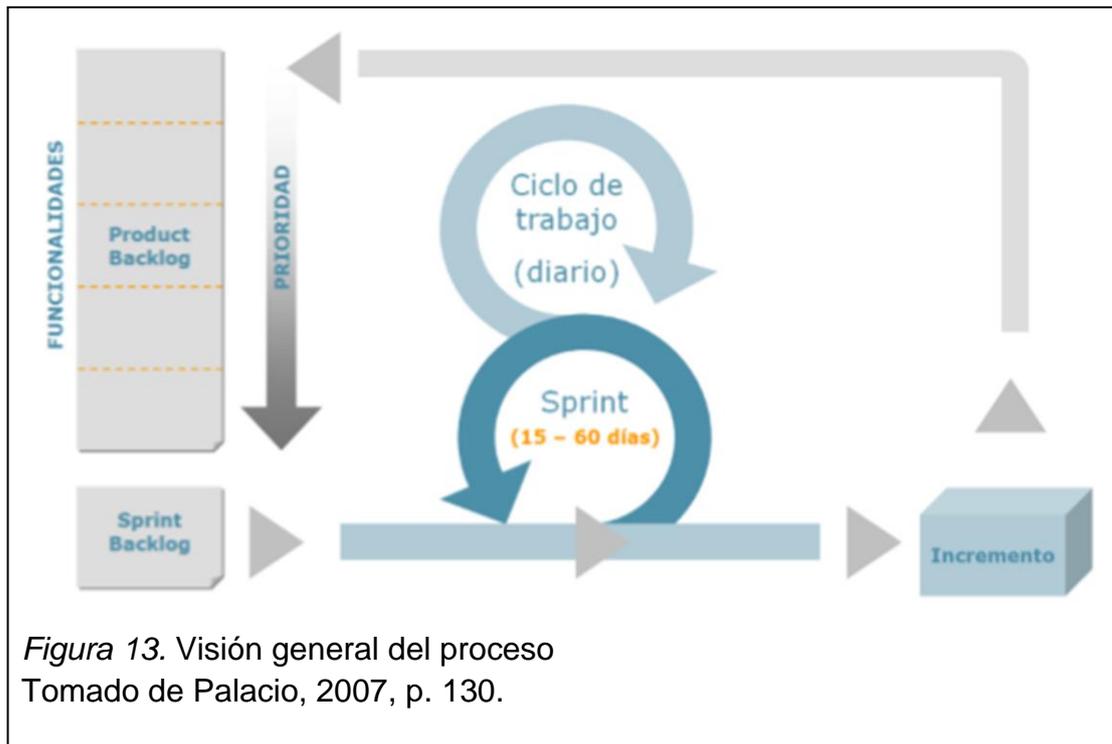


Figura 13. Visión general del proceso  
Tomado de Palacio, 2007, p. 130.

### 1.5.6. Roles o responsabilidades de Scrum

Según Schwaber y Sutherland (2011, pp. 5-7), los roles de *Scrum* son:

- **El Dueño del Producto (*Product Owner*)**

“Es el responsable de maximizar el valor del producto y del trabajo del Equipo de desarrollo” (Schwaber y Sutherland, 2011, p. 5).

Además, sus áreas de responsabilidad incluyen:

- Responsable oficial del proyecto
- Responsable del resultado final
- Financiamiento para el proyecto
- Retorno de la inversión del proyecto
- Gestionar la Pila del Producto

- **El equipo de Desarrollo (*Development Team*)**

Para Palacio (2007, p. 132) “es un equipo multidisciplinario que cubre todas las habilidades necesarias para generar el resultado”.

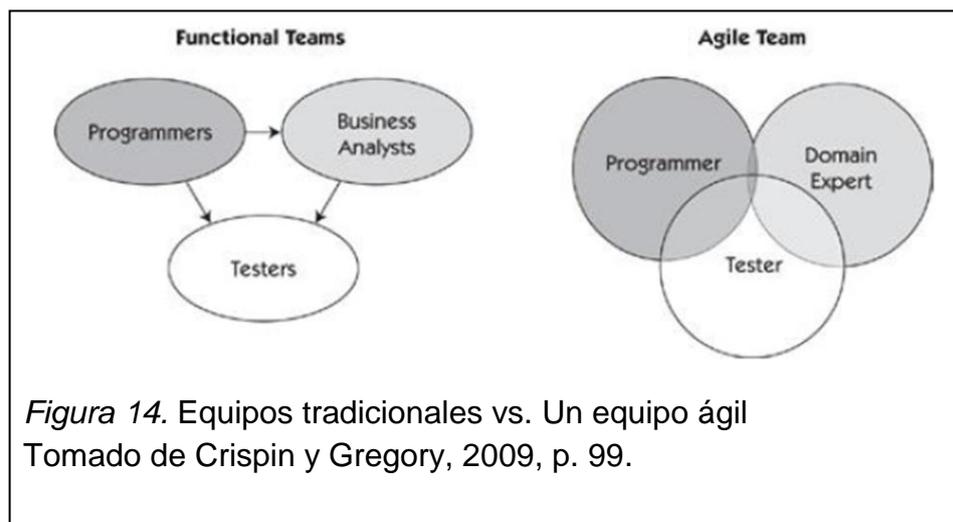
Los Equipos *Scrum* son auto-organizados y multifuncionales, ya que se encargan de entregar productos de forma iterativa e incremental.

- **Líder del proyecto (*Scrum Master*)**

“El *Scrum Master* es un líder servil, al servicio del Equipo *Scrum*” (Schwaber y Sutherland, 2011, p.16).

### 1.5.7. Pruebas Ágiles

Las pruebas ágiles (*Agile Testing*) son un esquema para facilitar el proceso de desarrollo de software y sus pruebas de aseguramiento de calidad, en el que cada miembro del equipo ágil se centra en entregar productos de alta calidad y de valor para el negocio.



”Agile es iterativo e incremental, lo que significa que los *testers* prueban cada incremento de codificación tan pronto es terminado. Una vez que estén seguros que funciona correctamente se sigue con la siguiente pieza a ser desarrollada” (Crispiny Gregory, 2009, p. 56).

A continuación se presentará una figura donde se muestra la diferencia entre metodologías ágiles y las metodologías tradicionales en su respectiva etapa de pruebas. En la imagen se presenta el trabajo en cada iteración e indica que el equipo de desarrollo no podrá continuar si una historia no está “hecha” (*done*), hasta que haya sido probada y todo funcione correctamente

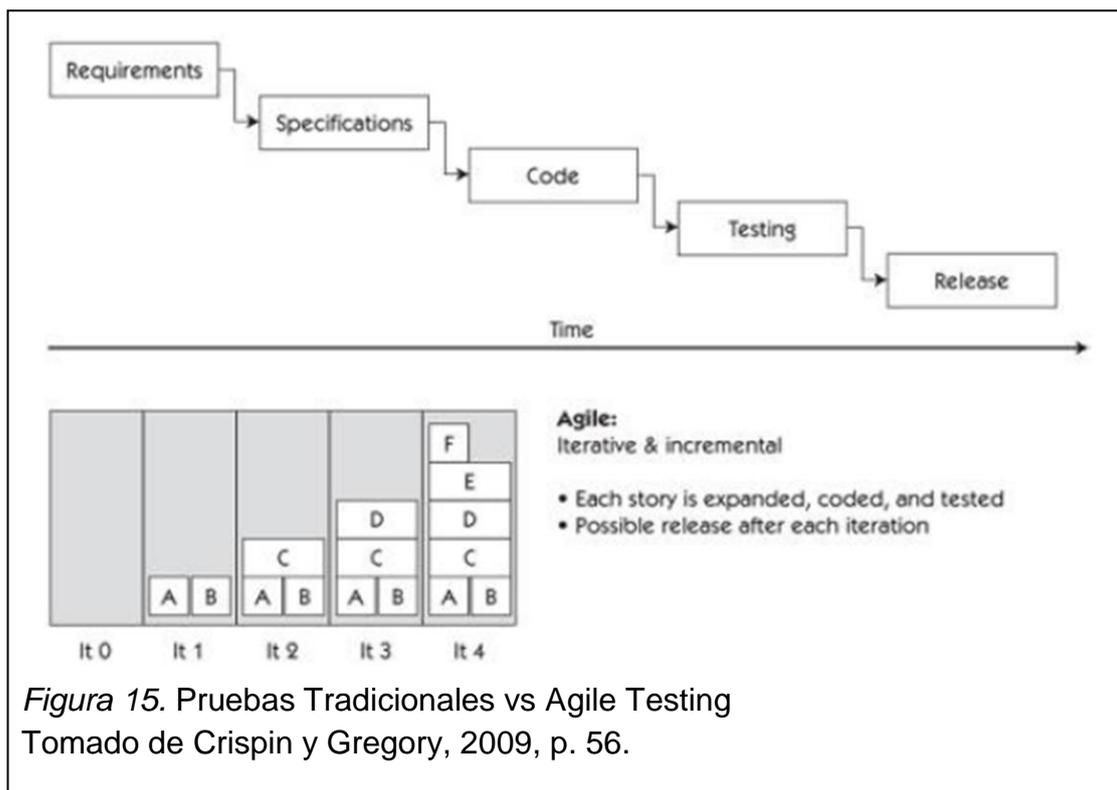


Figura 15. Pruebas Tradicionales vs Agile Testing  
Tomado de Crispin y Gregory, 2009, p. 56.

“Cada proyecto, cada equipo y a veces cada iteración es diferente. Cómo el equipo resuelva posibles problemas debe depender del problema, las personas y las herramientas que se tengan a disposición” (Crispiny Gregory, 2009, p. 56).

Por lo tanto, ser miembro de un equipo ágil, significa poder adaptarse a las necesidades del equipo y aportar rápidamente con la retroalimentación de las pruebas para así continuar con el siguiente hito.

#### **1.5.8. Agile Testing Quadrants**

Crispiny Gregory señala (2009, pp. 127-132) que las principales razones para realizar pruebas de aseguramiento de calidad son: para encontrar errores, para asegurarse de que el código es fiable y a veces sólo para ver si el código es utilizable.

Por esta razón. Brian Marick propuso una matriz con cuatro cuadrantes que permite a los *testers* asegurarse que se han realizado y considerado todos los diferentes tipos de pruebas que se necesitan con el fin de entregar valor a un producto de software.

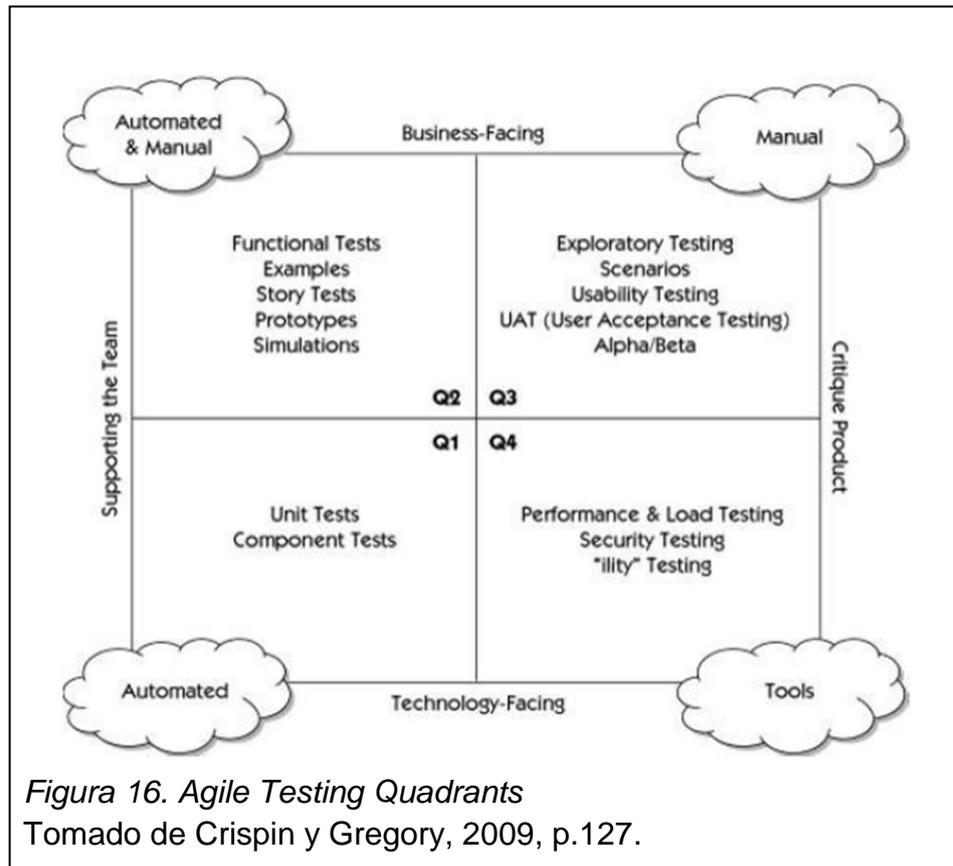
A continuación en la figura se observan los cuatro cuadrantes (*The Agile Testing Quadrants*), en el que cada uno de ellos refleja las diferentes razones por las que se realizan las pruebas de calidad de software.

En uno de los ejes, se divide la matriz en:

- Pruebas que apoyan al equipo de desarrollo y
- Pruebas que critican al producto

El otro eje está dividido en:

- Pruebas desde la perspectiva del negocio y
- Pruebas desde la perspectiva de tecnología



### Cuadrante 1

El cuadrante inferior izquierdo representa el desarrollo de software basado en pruebas, que es el núcleo en el desarrollo ágil. Está compuesto por pruebas unitarias, las cuales se encargan de verificar la funcionalidad de pequeños subconjuntos del sistema, tal como un objeto o método.

También se compone por pruebas de componentes, las cuales verifican el comportamiento de una parte más grande del sistema, tal como un grupo de clases.

**Cuadrante 2**

Las pruebas en el cuadrante 2 también están centradas en el trabajo del equipo de desarrollo, pero a alto nivel. Está compuesto por pruebas funcionales, prototipos y simulaciones, entre otros, los cuales se encargan de definir la calidad externa y las características que los clientes solicitan.

**Cuadrante 3**

Este cuadrante está compuesto por las pruebas de exploración, escenarios, pruebas de usabilidad y las pruebas de aceptación del usuario. Estas ayudan al equipo a diseñar el producto deseado.

**Cuadrante 4**

En este cuadrante se critican las características del producto, tales como el rendimiento, robustez y seguridad. Todo se realiza desde el punto de vista tecnológico.

## CAPÍTULO II

### 2. DESARROLLO DE LA SOLUCIÓN

#### 2.1. Visión de la Solución

La visión de este sitio web es representar y liderar al sector de alojamiento a nivel nacional, ayudando a turistas y agencias de viajes a realizar reservas de habitaciones; y además proporcionar información de análisis para los hoteles afiliados.

##### 2.1.1. Alcance Inicial

En este proceso ágil para el desarrollo del prototipo, todo el inventario de características y requisitos del sistema (historias de usuario), estarán detalladas en la pila del producto (*Product Backlog*), para así llegar al incremento que el propietario del producto desea obtener.

Una historia de usuario es una solicitud de un requerimiento funcional, detalle por el usuario final de la aplicación, el mismo que es expresado en un lenguaje natural.

Las historias de usuario se clasifican en:

- Una épica: es una historia muy larga.

Las épicas y los temas son más difíciles de estimar

Una épica por si sola es considerada un tema.

- Un tema: es un conjunto de historias relacionadas.

A continuación se especificará las historias de usuario, detalladas desde el punto de vista (roles):

- Usuario final y
- Administrador de la aplicación.

## Usuario final

Tabla 5. Historia de usuario 1

Historia de usuario	
<b>Número:</b> 1	<b>Usuario:</b> Cliente
<b>Nombre historia:</b> Elegir ciudad de la reserva	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Media
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b> 1
<b>Descripción:</b> Como usuario de la aplicación, quiero elegir una ciudad para reservar una habitación de hotel.	
<b>Validación:</b> El cliente puede seleccionar cualquiera de las ciudades del país.	

Tabla 6. Historia de usuario 2

Historia de usuario	
<b>Número:</b> 2	<b>Usuario:</b> Cliente
<b>Nombre historia:</b> Seleccionar dentro de un rango de fechas	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alta
<b>Puntos estimados:</b> 5	<b>Iteración asignada:</b> 1
<b>Descripción:</b> Como usuario de la aplicación, quiero realizar una reserva dentro de un determinado rango de fechas.	
<b>Validación:</b> El cliente solo podrá seleccionar desde la fecha actual o superior.	

## Usuario final

Tabla 7. Historia de usuario 3

Historia de usuario	
<b>Número:</b> 3	<b>Usuario:</b> Cliente
<b>Nombre historia:</b> Filtrar por precio	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en desarrollo:</b> Baja
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 1
<b>Descripción:</b> Como usuario de la aplicación, quiero filtrar los hoteles dentro de un rango de precios.	
<b>Validación:</b> Ninguna	

Tabla 8. Historia de usuario 4

Historia de usuario	
<b>Número:</b> 4	<b>Usuario:</b> Cliente
<b>Nombre historia:</b> Filtrar por nombre del hotel	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en desarrollo:</b> Baja
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 1
<b>Descripción:</b> Como usuario de la aplicación, quiero filtrar los hoteles por el nombre o en orden alfabético.	
<b>Validación:</b> Las opciones de filtrado serían: de la A-Z y de la Z-A	

## Usuario final

Tabla 9. Historia de usuario 5

Historia de usuario	
<b>Número:</b> 5	<b>Usuario:</b> Cliente
<b>Nombre historia:</b> Compaginación	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en desarrollo:</b> Baja
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 1
<b>Descripción:</b> Como usuario de la aplicación, quiero compaginar los hoteles.	
<b>Validación:</b> Del listado de hoteles quiero poder ver cinco, diez o todos.	

Tabla 10. Historia de usuario 6

Historia de usuario	
<b>Número:</b> 6	<b>Usuario:</b> Cliente
<b>Nombre historia:</b> Filtrar por categoría	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en desarrollo:</b> Baja
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 1
<b>Descripción:</b> Como usuario de la aplicación, quiero filtrar los hoteles por su categoría o número de estrellas.	
<b>Validación:</b> El rango para asignar una categoría es de 1 a 5 estrellas.	

## Usuario final

Tabla 11. Historia de usuario 7

Historia de usuario	
<b>Número:</b> 7	<b>Usuario:</b> Cliente
<b>Nombre historia:</b> Galería de fotos	
<b>Prioridad en negocio:</b> Media	<b>Riesgo en desarrollo:</b> Baja
<b>Puntos estimados:</b> 5	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Fernando Piedra	
<b>Descripción:</b> Como usuario de la aplicación, quiero ver imágenes del hotel y sus habitaciones.	
<b>Validación:</b> El cliente puede visualizar las imágenes en dimensiones más altas al hacer clic sobre ellas.	

Tabla 12. Historia de usuario 8

Historia de usuario	
<b>Número:</b> 8	<b>Usuario:</b> Cliente
<b>Nombre historia:</b> Descripción de los hoteles	
<b>Prioridad en negocio:</b> Media	<b>Riesgo en desarrollo:</b> Baja
<b>Puntos estimados:</b> 5	<b>Iteración asignada:</b> 2
<b>Descripción:</b> Como usuario de la aplicación, quiero ver una breve descripción de todos los hoteles.	
<b>Validación:</b> Ninguna.	

## Usuario final

Tabla 13. Historia de usuario 9

Historia de usuario	
<b>Número:</b> 9	<b>Usuario:</b> Cliente
<b>Nombre historia:</b> Reservación	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alta
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b> 2
<b>Descripción:</b> Como usuario de la aplicación, quiero ingresar mis datos y realizar la reserva	
<b>Validación:</b> El cliente puede visualizar solo las habitaciones disponibles.	

Tabla 14. Historia de usuario 10

Historia de usuario	
<b>Número:</b> 10	<b>Usuario:</b> Cliente
<b>Nombre historia:</b> Diseño	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alta
<b>Puntos estimados:</b> 8	<b>Iteración asignada:</b> 2
<b>Descripción:</b> Como usuario de la aplicación, quiero acceder desde cualquier dispositivo móvil.	
<b>Validación:</b> Diseño adaptativo ( <i>responsive</i> )	

A continuación se presentarán las historias de usuario desde la perspectiva de la alta gerencia del sitio web:

Tabla 15. Historia de usuario 11

<b>Historia de usuario</b>	
<b>Número:</b> 11	<b>Usuario:</b> Administrador
<b>Nombre historia:</b> Ingreso usuarios	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Media
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 2
<b>Descripción:</b> Como gerente del <i>booking</i> , quiero ingresar al administrador con un usuario y contraseña.	
<b>Validación:</b>	

Tabla 16. Historia de usuario 12

<b>Historia de usuario</b>	
<b>Número:</b> 12	<b>Usuario:</b> Administrador
<b>Nombre historia:</b> Creación de hoteles	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en desarrollo:</b> Baja
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 2
<b>Descripción:</b> Como gerente del <i>booking</i> , quiero gestionar la creación de hoteles con sus respectivas características.	
<b>Validación:</b>	

## Administrador

Tabla 17. Historia de usuario 13

Historia de usuario	
<b>Número:</b> 13	<b>Usuario:</b> Administrador
<b>Nombre historia:</b> Listado de hoteles	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en desarrollo:</b> Baja
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 2
<b>Descripción:</b> Como gerente del <i>booking</i> , quiero ver un listado de todos los hoteles del sitio web	
<b>Validación:</b> El administrador puede ver todos los hoteles o filtrarlos y ver 10, 25 ó 50.	

Tabla 18. Historia de usuario 14

Historia de usuario	
<b>Número:</b> 14	<b>Usuario:</b> Administrador
<b>Nombre historia:</b> Exportar información	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en desarrollo:</b> Baja
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 3
<b>Descripción:</b> Como gerente del <i>booking</i> , quiero exportar la información de hoteles a diferentes formatos	
<b>Validación:</b> El administrador puede descargar la información en formato .pdf y .xlsx.	

## Administrador

Tabla 19. Historia de usuario 15

Historia de usuario	
<b>Número:</b> 15	<b>Usuario:</b> Administrador
<b>Nombre historia:</b> Listado de ciudades	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en desarrollo:</b> Baja
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 3
<b>Descripción:</b> Como gerente del <i>booking</i> , quiero crear ciudades y ver un listado respectivo con todas.	
<b>Validación:</b> El administrador puede ver todas las ciudades o filtrarlas y ver 10, 25 ó 50.	

Tabla 20. Historia de usuario 16

Historia de usuario	
<b>Número:</b> 16	<b>Usuario:</b> Administrador
<b>Nombre historia:</b> Exportar información	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en desarrollo:</b> Baja
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 3
<b>Descripción:</b> Como gerente del <i>booking</i> , quiero exportar las ciudades en diferentes formatos	
<b>Validación:</b> El administrador puede descargar la información en formato .pdf y .xlsx.	

## Administrador

Tabla 21. Historia de usuario 17

Historia de usuario	
<b>Número:</b> 17	<b>Usuario:</b> Administrador
<b>Nombre historia:</b> Listado de provincias	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en desarrollo:</b> Baja
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 3
<b>Descripción:</b> Como gerente del <i>booking</i> , quiero crear provincias y ver un listado respectivo con todas.	
<b>Validación:</b> El administrador puede ver todas las provincias o filtrarlas.	

Tabla 22. Historia de usuario 18

Historia de usuario	
<b>Número:</b> 18	<b>Usuario:</b> Administrador
<b>Nombre historia:</b> Exportar información	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en desarrollo:</b> Baja
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 3
<b>Descripción:</b> Como gerente del <i>booking</i> , quiero exportar las provincias en diferentes formatos.	
<b>Validación:</b> El administrador puede descargar la información en formato .pdf y .xlsx.	

## Administrador

Tabla 23. Historia de usuario 19

Historia de usuario	
<b>Número:</b> 19	<b>Usuario:</b> Administrador
<b>Nombre historia:</b> Listado de regiones	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en desarrollo:</b> Baja
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 3
<b>Descripción:</b> Como gerente del <i>booking</i> , quiero crear regiones y ver un listado respectivo con todas.	
<b>Validación:</b> El administrador puede ver todas las regiones o filtrarlas.	

Tabla 24. Historia de usuario 20

Historia de usuario	
<b>Número:</b> 20	<b>Usuario:</b> Administrador
<b>Nombre historia:</b> Exportar información	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en desarrollo:</b> Baja
<b>Puntos estimados:</b> 1	<b>Iteración asignada:</b> 3
<b>Descripción:</b> Como gerente del <i>booking</i> , quiero exportar las regiones del país en diferentes formatos.	
<b>Validación:</b> El administrador puede descargar la información en formato .pdf y .xlsx.	

## Administrador

Tabla 25. Historia de usuario 21

Historia de usuario	
<b>Número:</b> 21	<b>Usuario:</b> Administrador
<b>Nombre historia:</b> Edición información	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en desarrollo:</b> Baja
<b>Puntos estimados:</b> 5	<b>Iteración asignada:</b> 3
<b>Descripción:</b> Como gerente del <i>booking</i> , quiero editar información de los hoteles, ciudades, provincias y regiones ingresadas en el sistema.	
<b>Validación:</b>	

Tabla 26. Historia de usuario 22

Historia de usuario	
<b>Número:</b> 22	<b>Usuario:</b> Administrador
<b>Nombre historia:</b> Reporte de reservas por hoteles	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alta
<b>Puntos estimados:</b> 8	<b>Iteración asignada:</b> 3
<b>Descripción:</b> Como gerente del <i>booking</i> , quiero ver un reporte de cada hotel y contabilizar el número de reservas en un determinado rango de fechas.	
<b>Validación:</b> El administrador puede ver un reporte general o por ciudad.	

## Administrador

Tabla 27. Historia de usuario 23

Historia de usuario	
<b>Número:</b> 23	<b>Usuario:</b> Administrador
<b>Nombre historia:</b> Estado reservas	
<b>Prioridad en negocio:</b> Media	<b>Riesgo en desarrollo:</b> Media
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 4
<b>Descripción:</b> Como gerente del <i>booking</i> , quiero ingresar un número de cédula o pasaporte y ver la o las reservas de dicho cliente.	
<b>Validación:</b>	

Tabla 28. Historia de usuario 24

Historia de usuario	
<b>Número:</b> 24	<b>Usuario:</b> Administrador
<b>Nombre historia:</b> Estado habitaciones	
<b>Prioridad en negocio:</b> Baja	<b>Riesgo en desarrollo:</b> Baja
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b> 4
<b>Descripción:</b> Como gerente del <i>booking</i> , quiero ver en cualquier instante la cantidad de habitaciones disponibles de cualquier hotel.	
<b>Validación:</b>	

## Administrador

Tabla 29. Historia de usuario 25

Historia de usuario	
<b>Número:</b> 25	<b>Usuario:</b> Administrador
<b>Nombre historia:</b> Reporte ganancias	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alta
<b>Puntos estimados:</b> 8	<b>Iteración asignada:</b> 4
<b>Descripción:</b> Como gerente del <i>booking</i> , quiero ver un reporte de ganancias según el número de reservaciones realizadas en las diferentes provincias dentro de un determinado rango de fechas.	
<b>Validación:</b>	

Tabla 30. Historia de usuario 26

Historia de usuario	
<b>Número:</b> 26	<b>Usuario:</b> Administrador
<b>Nombre historia:</b> Reporte ingresos por tipo de habitación	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alta
<b>Puntos estimados:</b> 8	<b>Iteración asignada:</b> 4
<b>Descripción:</b> Como gerente del <i>booking</i> , quiero ver un reporte con el total de ingresos por cada tipo de habitación en un determinado rango de fechas.	
<b>Validación:</b>	

## 2.2. Planificación del Proyecto

### 2.2.1. Cronograma del Proyecto

El desarrollo del prototipo se llevará a cabo en 150 (ciento cincuenta) días.

A continuación en la figura se observa el cronograma por fases del proyecto.



Figura 17. Cronograma de las fases del proyecto

### 2.2.2. Plan de Liberación

Tabla 31. Plan de liberación con *sprints*

#	Rol	Historia	Orden de liberación (Prioridad)	Story Points	Sprint
1	Usuario	Quiero elegir una ciudad para reservar una habitación de hotel.	1	3	1
2	Usuario	Quiero realizar una reserva dentro de un determinado rango de fechas.	2	5	1
3	Usuario	Quiero filtrar los hoteles dentro de un	5	2	1

#	Rol	Historia	Orden de liberación (Prioridad)	Story Points	Sprint
		rango de precios.			
4	Usuario	Quiero filtrar los hoteles por el nombre o en orden alfabético.	4	2	1
5	Usuario	Quiero compaginar los hoteles.	7	1	1
6	Usuario	Quiero filtrar los hoteles por su categoría o número de estrellas.	3	2	1
7	Usuario	Quiero ver imágenes del hotel y sus habitaciones	6	5	1
8	Usuario	Quiero ver una breve descripción de todos los hoteles.	10	5	2
9	Usuario	Quiero ingresar mis datos y realizar la reserva	8	3	2
10	Usuario	Como cliente del sitio web, quiero acceder desde cualquier dispositivo móvil.	9	8	2
11	Gerente	Como gerente del <i>booking</i> , quiero ingresar al administrador con un usuario y contraseña.	11	1	2
12	Gerente	Como gerente del	12	1	2

#	Rol	Historia	Orden de liberación (Prioridad)	Story Points	Sprint
		<i>booking</i> , quiero gestionar la creación de hoteles con sus respectivas características.			
13	Gerente	Como gerente del <i>booking</i> , quiero ver un listado de todos los hoteles	13	1	2
14	Gerente	Como gerente del <i>booking</i> , quiero exportar la información de hoteles a diferentes formatos	16	2	3
15	Gerente	Como gerente del <i>booking</i> , quiero crear ciudades y ver un listado respectivo.	17	1	3
16	Gerente	Como gerente del <i>booking</i> , quiero exportar las ciudades en diferentes formatos	18	1	3
17	Gerente	Como gerente del <i>booking</i> , quiero crear provincias y ver un listado respectivo.	19	1	3
18	Gerente	Como gerente del <i>booking</i> , quiero exportar las provincias	20	1	3

#	Rol	Historia	Orden de liberación (Prioridad)	Story Points	Sprint
		en diferentes formatos.			
19	Gerente	Como gerente del booking, quiero crear regiones y ver un listado	21	1	3
20	Gerente	Como gerente del <i>booking</i> , quiero exportar las regiones del país en diferentes formatos	22	1	3
21	Gerente	Como gerente del <i>booking</i> , quiero editar información de los hoteles, ciudades, provincias y regiones ingresadas en el sistema.	15	5	3
22	Gerente	Como gerente del <i>booking</i> , quiero ver un reporte de cada hotel y contabilizar el número de reservas en un determinado rango de fechas.	14	8	3
23	Gerente	Como gerente del <i>booking</i> , quiero ingresar un número de cédula o pasaporte y ver la o las reservas	26	2	4

#	Rol	Historia	Orden de liberación (Prioridad)	Story Points	Sprint
		de dicho cliente.			
24	Gerente	Como gerente del <i>booking</i> , quiero ver en cualquier instante la cantidad de habitaciones disponibles de cualquier hotel.	25	2	4
25	Gerente	Como gerente del <i>booking</i> , quiero ver un reporte de ganancias según el número de reservaciones realizadas en las diferentes provincias dentro de un determinado rango de fechas	24	8	4
26	Gerente	Como gerente del <i>booking</i> , quiero ver un reporte con el total de ingresos por cada tipo de habitación en un determinado rango de fechas	23	8	4
		<b>Total</b>		<b>80</b>	

La tabla anterior muestra detalladamente el número de tareas a realizar en los 4 *sprints* (ciclos de desarrollo). Como se puede apreciar, existe una velocidad esperada de trabajo de 20 *story points* por mes.

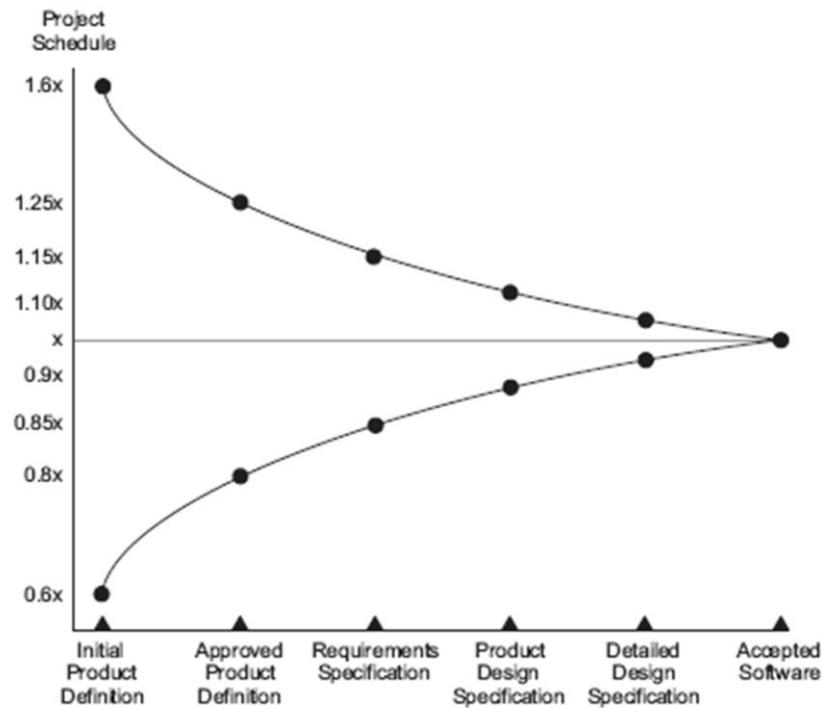
### 2.2.3. Planificación Ágil

“La estimación y planificación son fundamentales para el éxito de cualquier proyecto de desarrollo de software de cualquier tamaño o consecuencia” (Cohn, 2005, p. 3).

El propósito de la planificación ágil es:

- Dar una respuesta a la pregunta ¿Cuándo vas hacer?
- Dar una respuesta a la pregunta ¿Qué se debe construir?
- Reducir el riesgo
- Reducir la incertidumbre
- Mejorar el soporte para la toma de decisiones
- Establecer confianza
- Actividad de continuo cambio de planes

Según Cohn (2005, pp. 3-5), la dificultad de la estimación y planificación no es algo nuevo. En el año 1981, Barry Boehm dibujó la primera versión de un “cono de incertidumbre”, denominado así en el año 1998 por Steve McConnell. Es decir, desde la década de los ochenta se presentaron rangos iniciales de incertidumbre en diferentes puntos en un proceso secuencial (metodología en cascada) de desarrollo de software.



*Figura 18.* Cono de incertidumbre  
Tomado de Cohn, 2005, p 4.

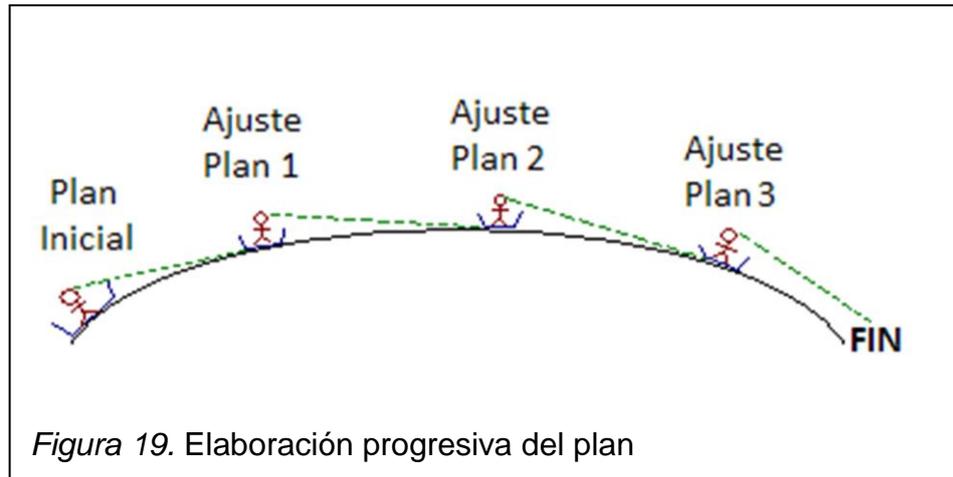
La figura anterior especifica que durante la fase de viabilidad de un proyecto la estimación puede variar de 60% a 160%. Es decir, un proyecto que puede durar 20 semanas podría llevarse a cabo en un plazo de 12 a 32 semanas. Una vez escritos los requerimientos funcionales, la estimación podría ser del 15% en cualquier dirección. Por lo tanto una estimación de 20 semanas significa un trabajo que lleva de 17 a 23 semanas de duración (Cohn, 2005, p. 4).

Respecto a la diferencia con la planificación tradicional, se elaboró una tabla con el fin de señalar sus principales diferencias.

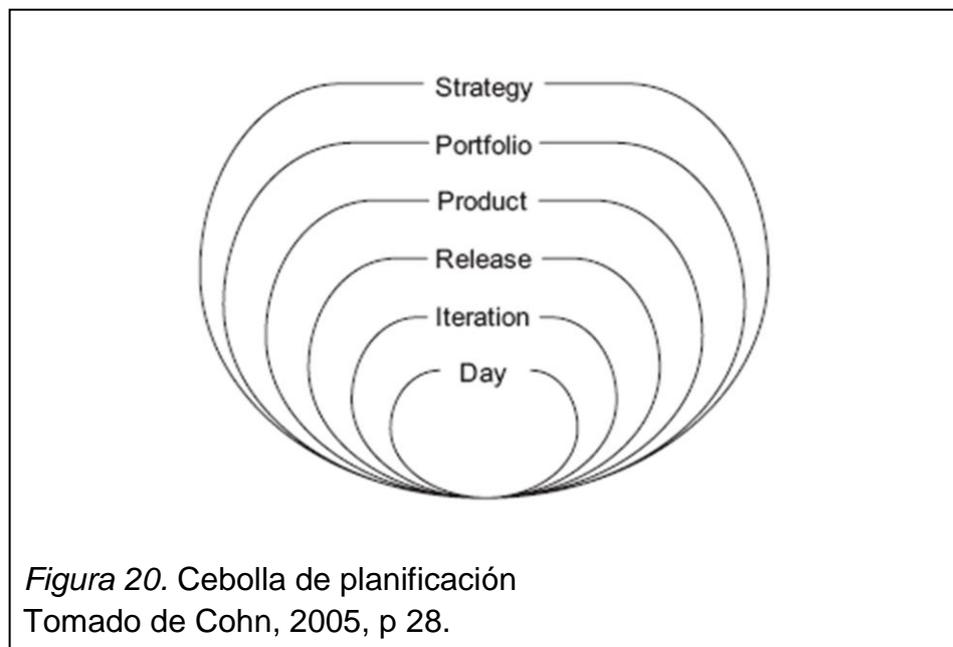
Tabla 32. Planificación tradicional vs planificación ágil

<b>Planificación Tradicional</b>	<b>Planificación Ágil</b>
Se planifica por actividades.	Se planifica por <i>features</i> (unidades de valor para el cliente).
Las actividades no son independientes.	<i>Features</i> pueden ser independientes.
Demasiadas tareas simultáneas. Se trata de mantener lo más ocupados posibles a todos.	Mantiene una holgura para hacer frente a la variabilidad de los proyectos de <i>software</i> .
Entregables no se desarrollan en orden de prioridad.	Se ordena por valor para el cliente.
Las estimaciones se convierten en compromisos.	No se pueden hacer compromisos sobre probabilidades.
Un proyecto se planifica como la ejecución de pasos	Un proyecto no se debe planificar como la ejecución de pasos.

Otro punto crítico son los múltiples niveles de planificación, ya que recalcan la importancia de recordar que no podemos ver más allá del horizonte, por lo que se debe tener una pausa para analizar y hacer cambios de ser necesario. En la siguiente figura se muestra la representación de lo anteriormente mencionado.



Para lograr esto el enfoque ágil planea 3 horizontes representados en la siguiente figura.



La figura anterior especifica los horizontes de la planificación ágil. Los horizontes que se llevaron a cabo en el desarrollo del prototipo fueron:

- Liberación

Alcance, cronograma, recursos. Ajustados usualmente al inicio de cada iteración.

– Iteración

Tareas para convertir un requerimiento en software que funcione y se pueda probar.

– Día

Coordinar trabajo y sincronizar esfuerzo diario.

Otros horizontes

– Producto

Evolución del producto más allá de una liberación.

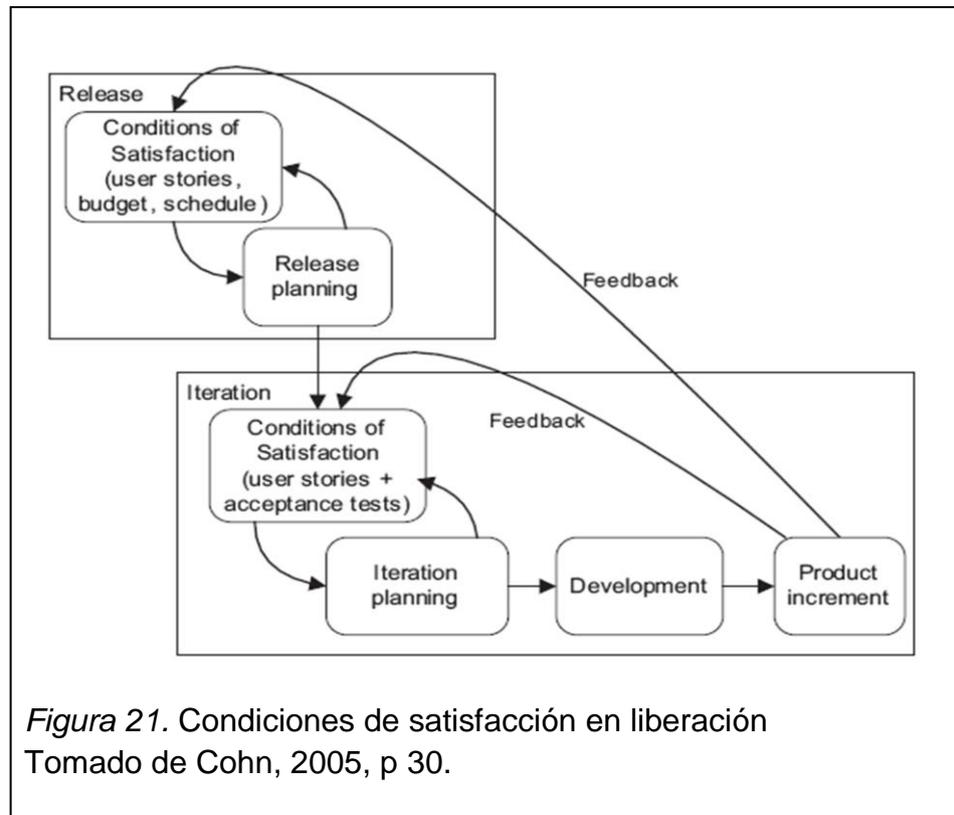
– Portafolio

Selección de productos que mejor implementen la **planificación estratégica** de la organización.

Ante lo expuesto anteriormente, se concluye que la planificación ágil:

- Se enfoca en la planificación más que en el plan.
- Estimula el cambio.
- Produce un conjunto de planes que pueden ser cambiados fácilmente.
- Se extiende a lo largo de todo el proyecto.

## 2.2.4. Condiciones de Satisfacción



*Figura 21. Condiciones de satisfacción en liberación*  
 Tomado de Cohn, 2005, p 30.

La figura anterior indica los bucles de retroalimentación que se llevaron a cabo para obtener un nuevo incremento del producto, tomando en cuenta la retroalimentación de los clientes y desarrolladores en cada iteración. La retroalimentación de la última iteración se usó antes de estimar una nueva con los comentarios, ideas y cambios, entre otros, llevados a cabo en la misma.

Al igual que la planificación de la liberación, la planificación de la iteración es iterativa. El dueño del producto y el equipo discuten de las diversas formas que existen para dar cumplimiento a las condiciones de satisfacción para la iteración (Cohn, 2005, p. 30).

## 2.3. Construcción de la solución

### 2.3.1. Sprint 0

El *Sprint* inicial (*sprint 0*) consiste en crear el esqueleto básico y las bases del proyecto para que los *sprints* futuros puedan agregar verdadero valor.

Para llevar a cabo las reuniones con el patrocinador, en el desarrollo del prototipo se utilizó un espacio para la creatividad y la creación de un diseño innovador. Se empezó a identificar a los usuarios del sitio, o el público objetivo al que desea llegar el cliente. Luego de algunas charlas se identificaron claramente los objetivos del proyecto y se realizó una investigación del sector al que se desea brindar información con el uso de este aplicativo web.

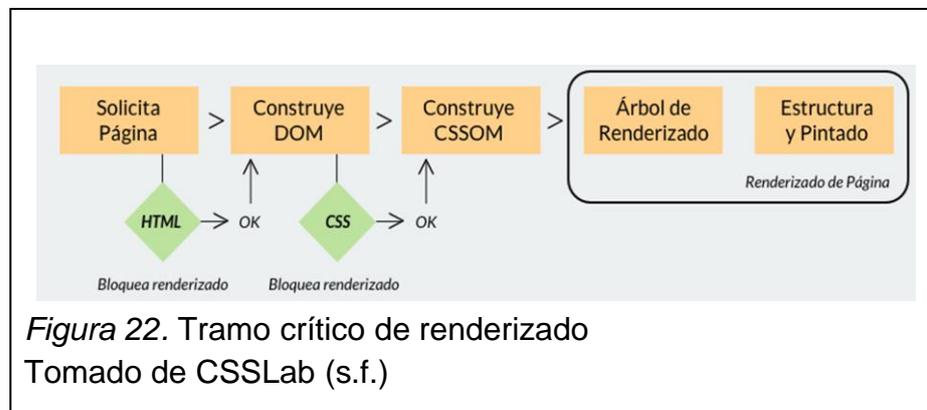
### 2.3.2. Diagrama de Arquitectura

El diagrama de arquitectura es el diseño con el nivel más alto de abstracción de la estructura de un sistema en el que se representa la interacción entre los usuarios y componentes del aplicativo.

El proceso del funcionamiento de la arquitectura del prototipo es el siguiente:

1. El usuario realiza una petición (*request*) a través del protocolo *Hypertext Transfer Protocol* (HTTP), desde cualquier dispositivo móvil o computador personal, por medio de un disparador de acciones. (Ej. Pulsando un botón).
2. El controlador captura el evento y los objetos enviados e invoca peticiones o llamadas al modelo para resolver la acción solicitada por el usuario.

3. El modelo se encargará de interactuar con la base de datos directamente.
4. El modelo accederá a la base de datos para realizar consultas, actualizaciones o inserción de información, y retornará estos datos al controlador.
5. El controlador recibe los datos enviados por el modelo y envía a la vista.
6. La vista obtiene los datos, y convierte el código y recursos asociados con una secuencia de pasos (*critical rendering path*), como se observa en la siguiente figura:



La figura anterior especifica los pasos para renderizar los correspondientes códigos HTML y CSS. Estos pasos son: parsear (funciones del compilador) el código HTML dentro del *Document Object Model* (DOM); construir el *CSS Object Model* para que los estilos sean entendidos por el navegador. Y, finalmente se construye el árbol de renderizado (*render tree*) (CSSLab, s.f.).

7. La vista despliega como salida (*response*) la interfaz apropiada para usuario (*Front-end*).

En las siguientes figuras se muestra la representación de lo anterior mencionado:

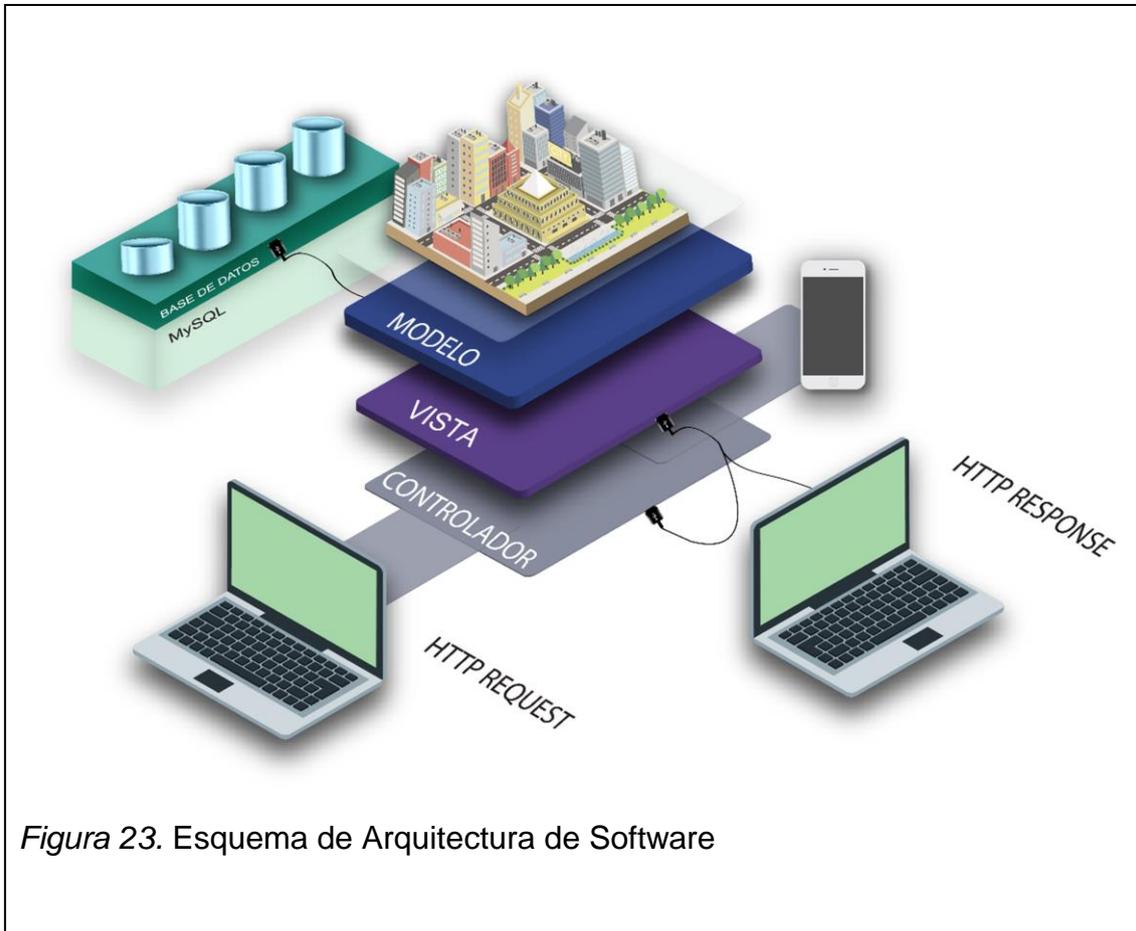


Figura 23. Esquema de Arquitectura de Software

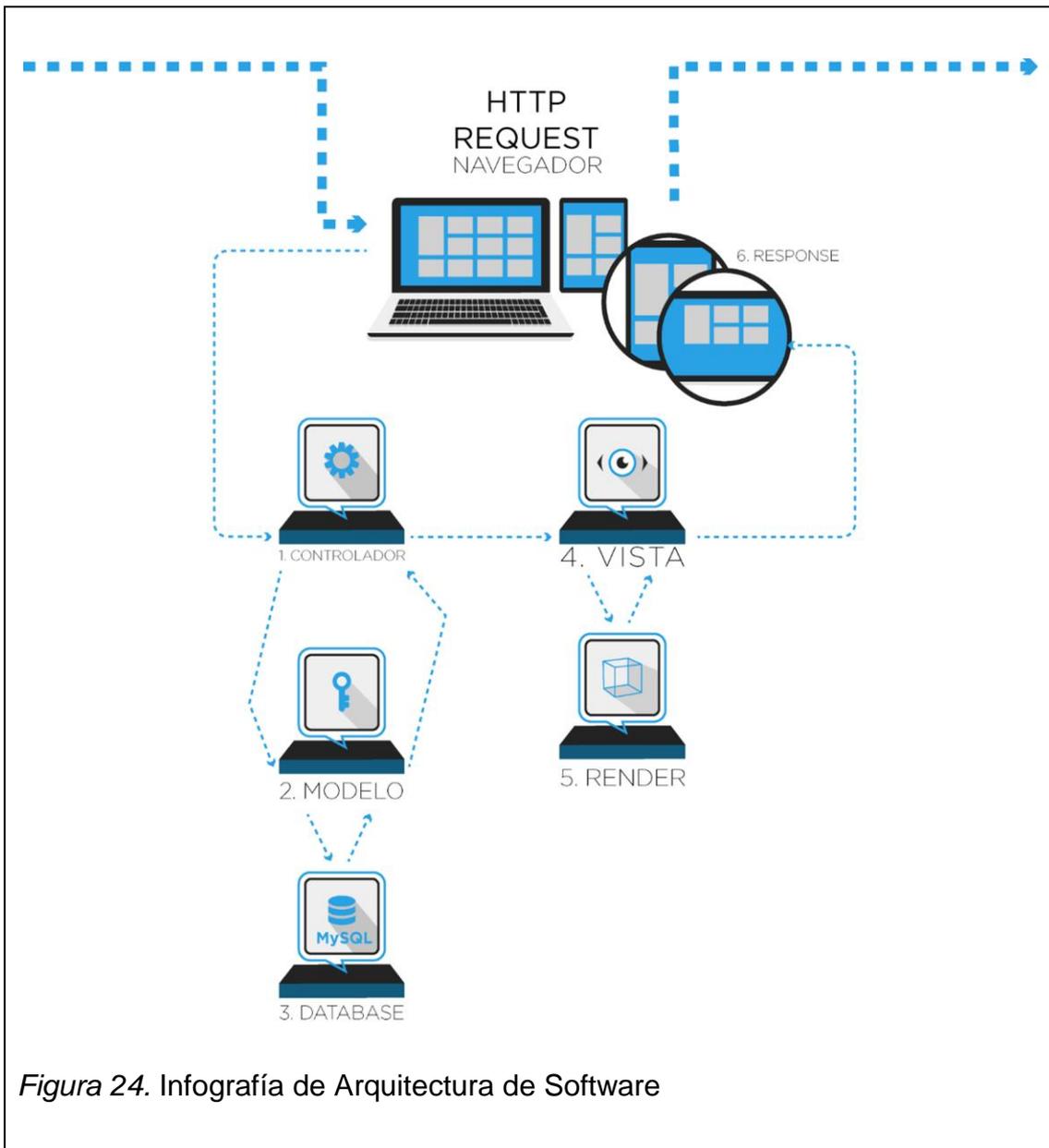


Figura 24. Infografía de Arquitectura de Software

### 2.3.3. Iteraciones de Desarrollo

A continuación en la figura se observan las cuatro iteraciones de desarrollo con sus respectivas historias de usuario, prioridad, riesgos y estimaciones.

N.-	HISTORIAS DE USUARIO	Prioridad	Riesgo	Iteración				Puntos Estimados
				I	II	III	IV	
1	Quiero elegir una ciudad para reservar una habitación de hotel.	Alta	Medio	X				3
2	Quiero realizar una reserva dentro de un determinado rango de fechas.	Alta	Alta	X				5
3	Quiero filtrar los hoteles dentro de un rango de precios.	Baja	Bajo	X				2
4	Quiero filtrar los hoteles por el nombre o en orden alfabético.	Baja	Bajo	X				2
5	Quiero compaginar los hoteles.	Baja	Bajo	X				1
6	Quiero filtrar los hoteles por su categoría o número de estrellas.	Baja	Bajo	X				2
7	Quiero ver imágenes del hotel y sus habitaciones	Media	Bajo	X				5
8	Quiero ver una breve descripción de todos los hoteles.	Media	Bajo		X			5
9	Quiero ingresar mis datos y realizar la reserva	Alta	Alta		X			8
10	Como cliente del sitio web, quiero acceder desde cualquier dispositivo móvil.	Alta	Alta		X			3
11	Como gerente del booking, quiero ingresar al administrador con un usuario y contraseña.	Alta	Medio		X			1
12	Como gerente del booking, quiero crear hoteles con sus respectivas características.	Bajo	Bajo		X			1
13	Como gerente del booking, quiero ver un listado de todos los hoteles	Bajo	Bajo		X			1
14	Como gerente del booking, quiero exportar la información de hoteles a diferentes formatos	Bajo	Bajo			X		2
15	Como gerente del booking, quiero ver un listado de las ciudades.	Bajo	Bajo			X		1
16	Como gerente del booking, quiero exportar las ciudades en diferentes formatos	Bajo	Bajo			X		1
17	Como gerente del booking, quiero ver un listado de las provincias.	Bajo	Bajo			X		1
18	Como gerente del booking, quiero exportar las provincias en diferentes formatos.	Bajo	Bajo			X		1
19	Como gerente del booking, quiero ver un listado de las regiones del país.	Bajo	Bajo			X		1
20	Como gerente del booking, quiero exportar las regiones del país en diferentes formatos	Bajo	Bajo			X		1
21	Como gerente del booking, quiero editar información de los hoteles, ciudades, provincias y regiones ingresadas en el sistema.	Bajo	Bajo			X		5
22	Como gerente del booking, quiero ver un reporte de cada hotel y contabilizar el número de reservas en un determinado rango de fechas.	Alta	Alta			X		8
23	Como gerente del booking, quiero ingresar un número de cédula o pasaporte y ver la o las reservas de dicho cliente.	Media	Medio				X	2
24	Como gerente del booking, quiero ver un reporte del estado de las habitaciones de cualquier hotel.	Bajo	Bajo				X	2
25	Como gerente del booking, quiero ver un reporte de ganancias según el número de reservaciones realizadas en las diferentes provincias dentro de un determinado rango de fechas	Alta	Alta				X	8
26	Como gerente del booking, quiero ver un reporte con el total de ingresos por cada tipo de habitación en un determinado rango de fechas	Alta	Alta				X	8
								<b>80</b>

Figura 25. Iteraciones de desarrollo

Los reportes presentados a continuación fueron realizados con la herramienta en línea **Pivotal Tracker** en su versión gratuita, que permite el seguimiento de todo el proyecto y sus historias de usuario, cristalizando prioridades. De esta manera ayuda a los equipos a que se centren en lo que es esencial de una manera visual con la ayuda de tableros dinámicos (PivotalTracker, s.f.).

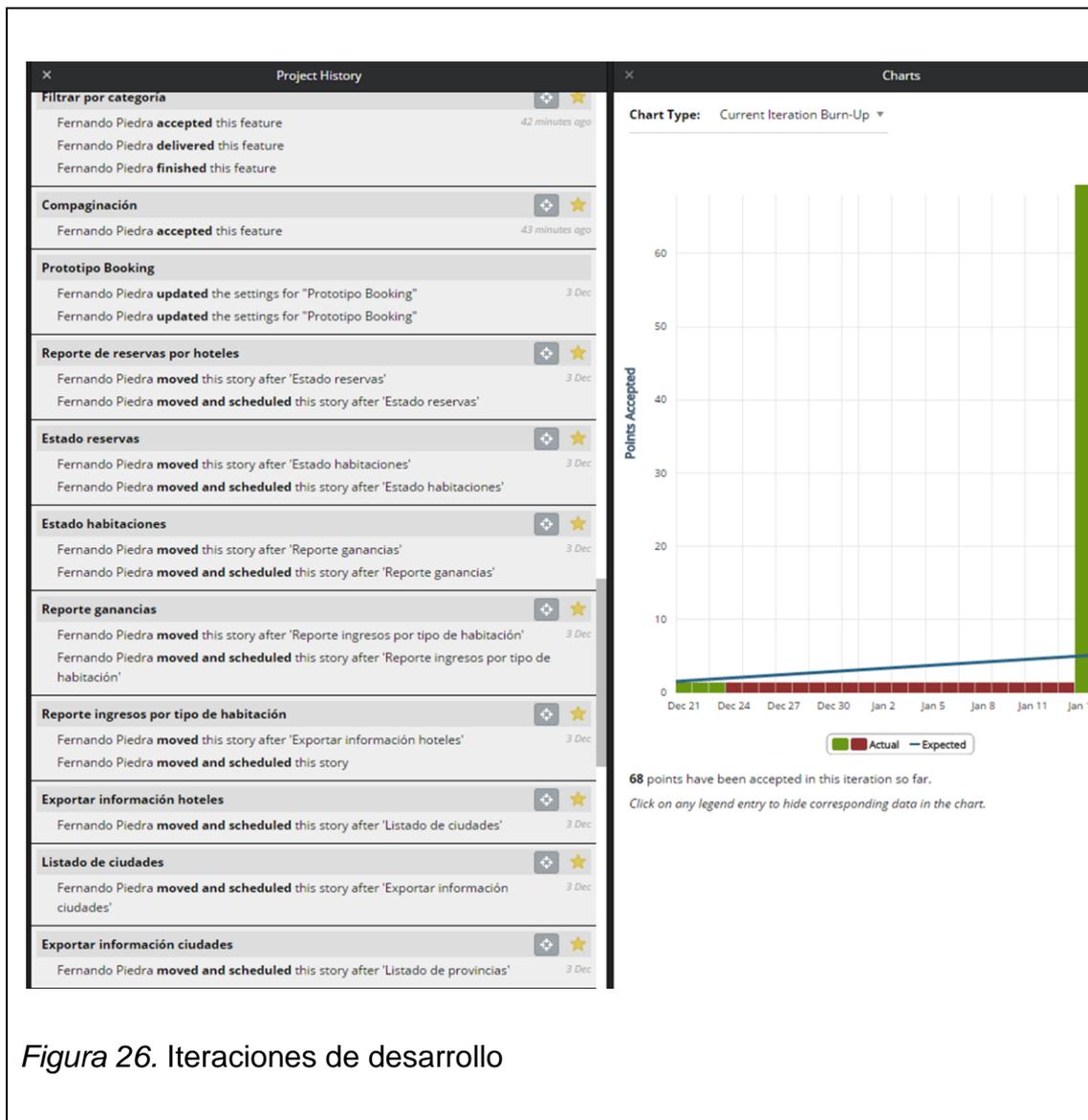


Figura 26. Iteraciones de desarrollo

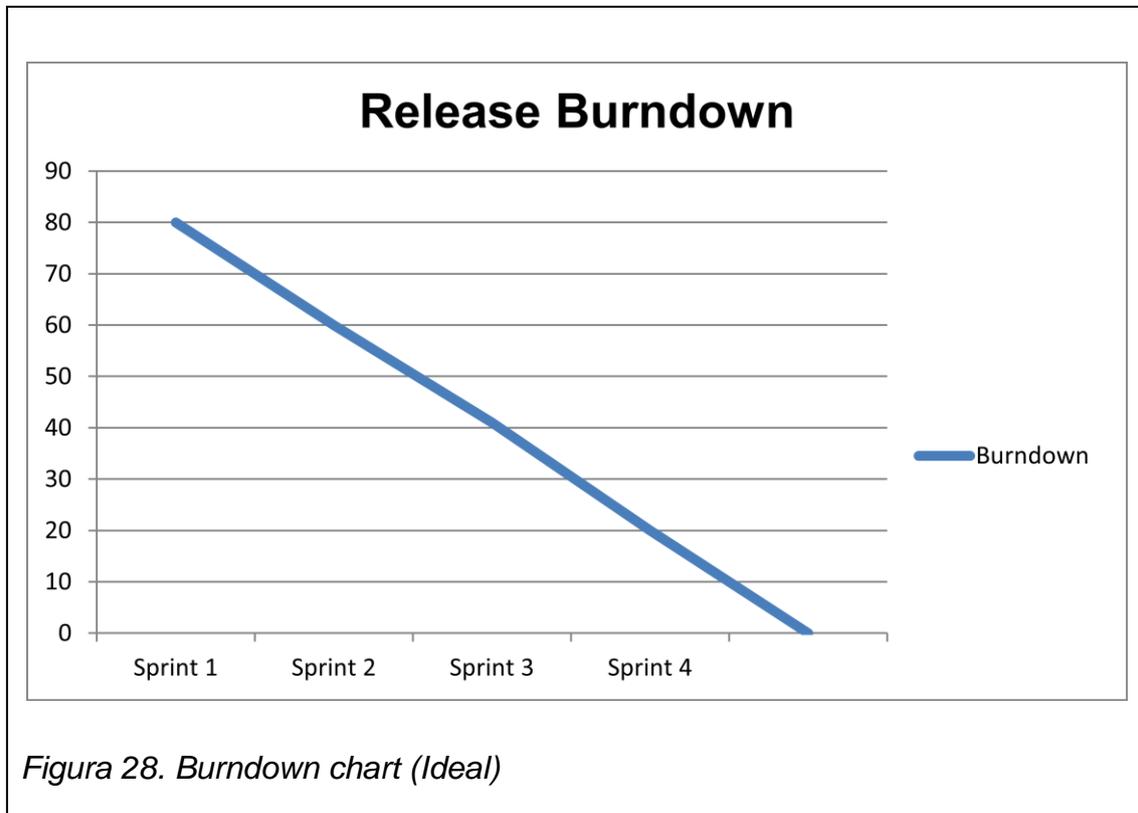
A continuación en la figura se observa el desglose diario del proyecto y como se mueven las historias a través de las diferentes fases de ejecución o estado. Las barras de colores muestran el total de puntos de historia realizados al final de cada día.



Figura 27. Desglose de historias de usuario

### 2.3.4. Release Burndown Chart

A continuación en la figura se observa el *release burndown chart*, que muestra el trabajo completado en cada *sprint*.



### 2.3.5. Despliegue del Producto

El objetivo del despliegue del producto consiste en “asegurarse que el producto está preparado para suministrarlo al cliente, ajustar el producto de *software* a las necesidades del usuario, y proceder a su entrega y recepción por parte del usuario” (Alonso, Martínez, y Segovia, 2005, p. 344).

Cabe recalcar que el despliegue inicia cuando el código ha sido exhaustivamente probado para su liberación.

Para el despliegue del producto se recomienda las siguientes características en un servidor dedicado (*Dedicated server*):

- CPU: 4 Núcleos (*Quad Core* 3.3GHZ)
- RAM: 16 GB
- Espacio del disco duro (*disk space*): 240 GB
- Ancho de banda (*bandwidth*): 20 TB
- Dirección IP (*IP Addresses*): 4 IPs
- Opciones de almacenamiento flexibles (RAID-1, SATA o SAS opcionales)
- Sistema operativo Linux
- Versión PHP: 5.5.9

### **2.3.6. Implementación de Integración Continua**

La implementación de integración continua conlleva una mejora de la calidad de software en todos sus aspectos: procesos, productos y equipo (Garzas, 2014).

#### **Calidad del proceso**

La integración continua usa la calidad del proceso para dar al equipo visibilidad del proceso de desarrollo en general, es decir, el equipo se mantendrá informado a todo momento sobre compilaciones, pruebas y versiones en ambientes de pruebas, entre otros.

En el desarrollo del presente proyecto se utilizó como cliente *graphical user interface* (GUI), para el manejo del repositorio GIT un *distributed version control system* (DVCS), llamado *SourceTree*, un *software* de control de versiones entre los mejores en la actualidad, destacado por su eficiencia, integridad de datos, confiabilidad del mantenimiento de versiones, gestión distribuida y visualización de historial en el desarrollo no lineal.

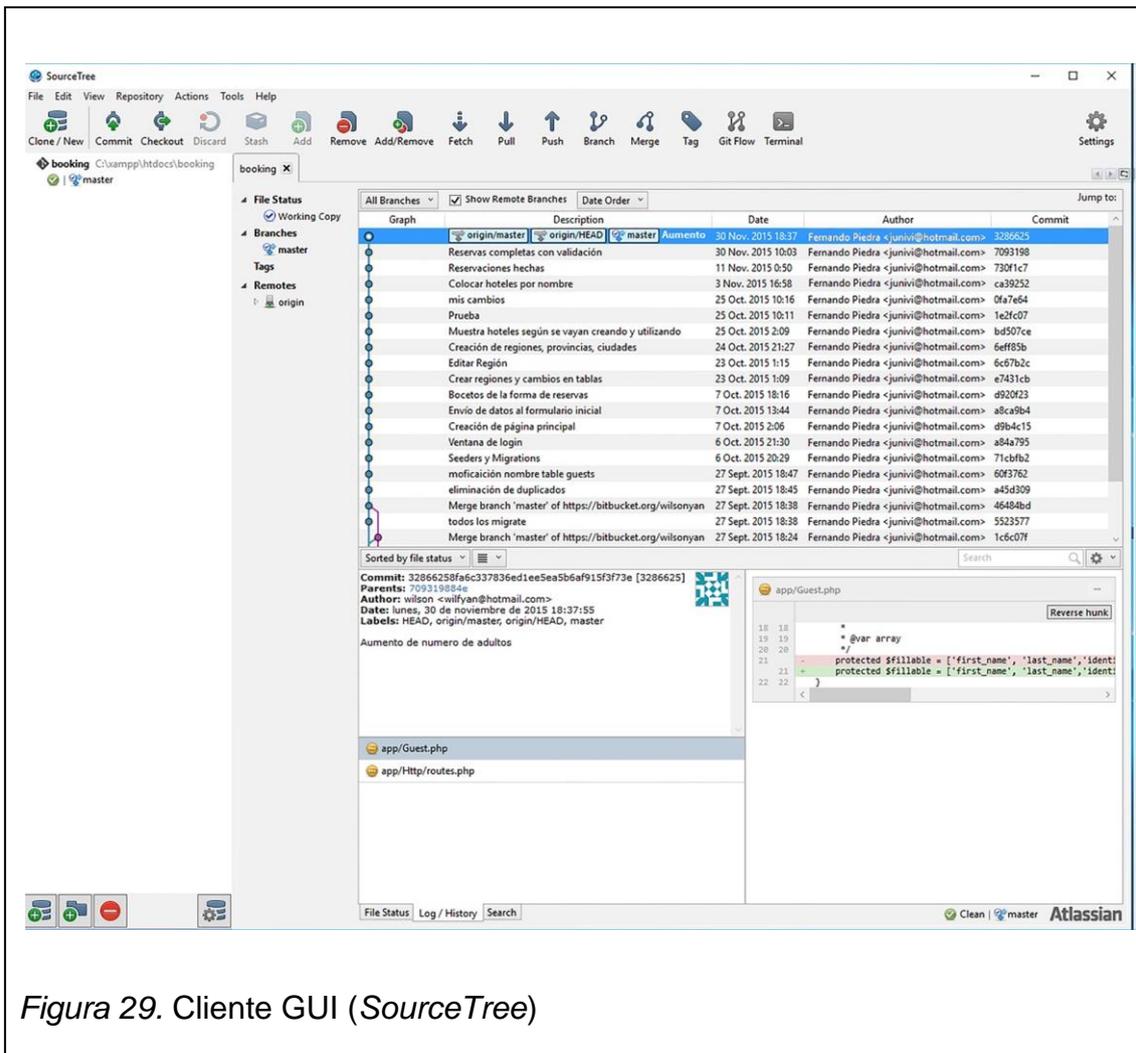


Figura 29. Cliente GUI (SourceTree)

La figura anterior muestra el *distributed version control system* (DVCS), y su historial de cambios con su respectiva descripción y fecha, entre otros datos de la versión en desarrollo.

### Calidad del producto

Por otro lado, un objetivo más de la integración continua es detectar errores lo antes posible, por lo que en cada iteración se llevaran a cabo pruebas funcionales y de aceptación para de esta manera realizar el incremento de producto con la calidad y valor deseado.

## **Calidad del equipo**

La calidad del equipo en este caso se puede resumir en tener todo el proceso de desarrollo bien claro, con sus respectivas pruebas, detectado errores tempranamente y solucionándolos en cada iteración, dejando como valor agregado nuevos conocimientos en cada etapa del desarrollo de la solución.

## **2.4. Cierre del Proyecto**

### **2.4.1. Acta de Cierre del Proyecto**

Una vez finalizado el proyecto de desarrollo del prototipo en plataforma web para reserva de habitaciones de hotel y análisis de la información se elaboró un documento en el cual se detalla los entregables, con el fin de conseguir y dejar por escrito la aceptación del mismo.

Con la firma del cliente se certifica que se ha cumplido con cabalidad cada uno de los puntos mencionados como entregables y que el mismo está conforme con ellos.

Revisar anexo 11 para revisión del modelo de la carta.

## CAPÍTULO III

### 3. CASO DE ESTUDIO

En el desarrollo del prototipo se han introducido características y mejoras, todas ellas diseñadas con el objetivo de impulsar el sector de alojamiento. Pero antes de iniciar con la implementación del software, es preciso realizar una evaluación o inspección del entorno actual del sector, para tener una visión clara de los procesos actuales y posteriormente verificar la mejora obtenida.

#### 3.1. Estado Actual

El problema radica en que no existe un sistema que centralice la información del movimiento de los hoteles y hosterías, entre otros, en el Ecuador. Los sitios principales que brindan información del sector son:

- Hoteles Ecuador (<http://goo.gl/O8SLqk>)
- Abordo (<http://goo.gl/5nHp4M>)

La dinámica actual de estos sitios es poco informativa, ya que muestra datos diferentes de un mismo hotel, creando así confusión y malestar entre los usuarios.

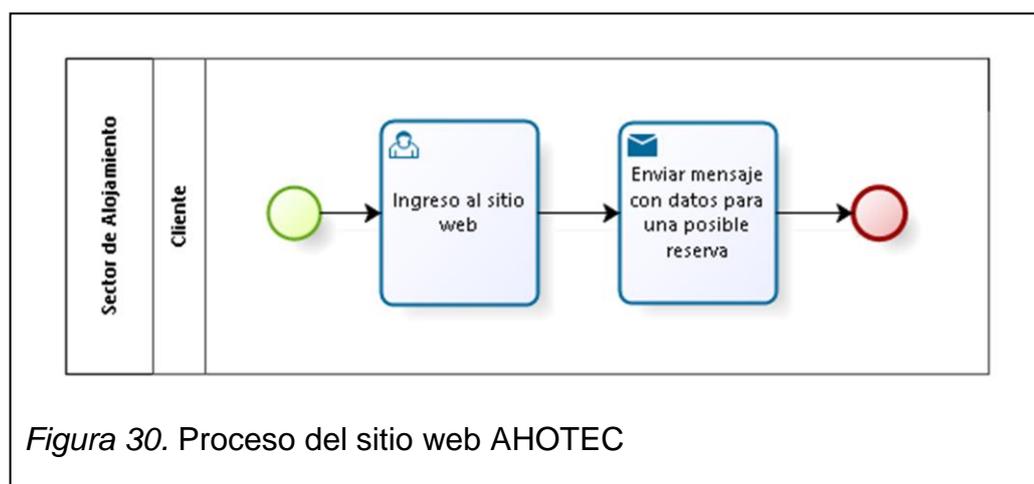


Figura 30. Proceso del sitio web AHOTEC

La figura anterior indica el proceso que realizan los usuarios al ingresar al sitio web oficial (interfaz no adaptativa), de Hoteles Ecuador. En el sitio web solamente se puede enviar un mensaje al hotel en el que se desee realizar una posible reservación.

### 3.2. Diseño del Plan Piloto

El plan piloto se encarga de realizar diversas funciones, entre ellas pruebas, capacitación y retroalimentación de los usuarios, esto con la finalidad de lograr una implementación sin inconvenientes y así realizar mejoras de la funcionalidad en caso de ser necesario.

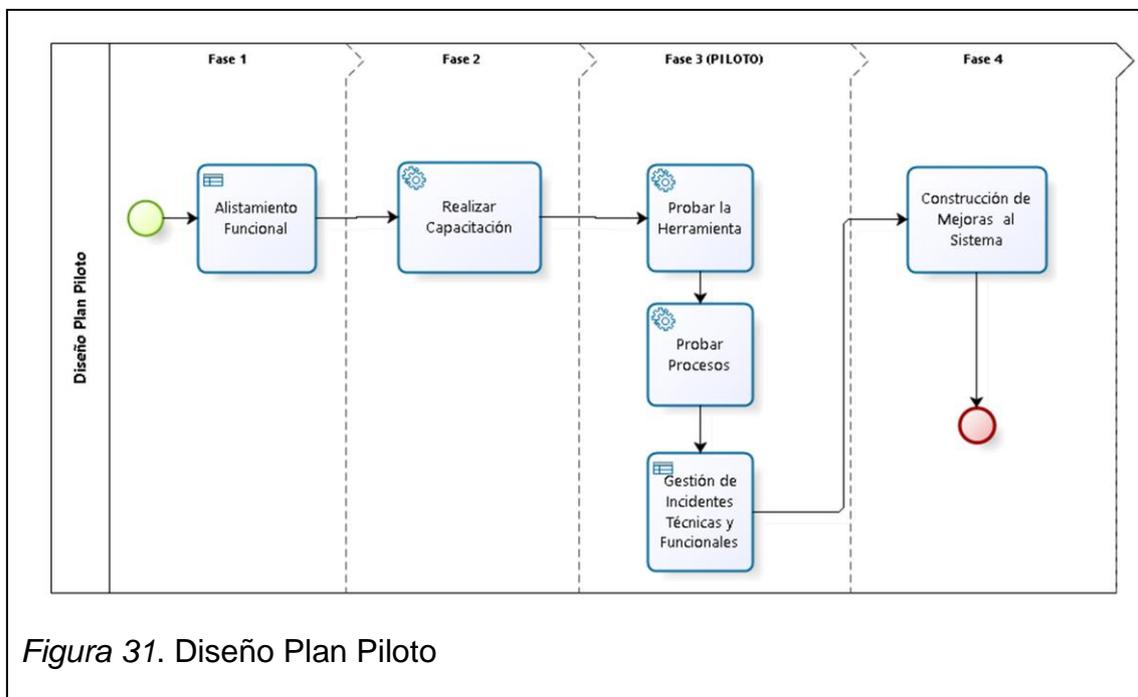


Figura 31. Diseño Plan Piloto

La figura anterior indica el proceso del plan piloto previo a la implementación del *software*, el cual consiste en una serie de pruebas funcionales que nos ayudarán a verificar si se han cumplido los objetivos planteados o resultados esperados desde la perspectiva del cliente.

### 3.3. Medida de la Mejora Obtenida

Tabla 33. Situación actual y las posibles mejoras.

<b>Medida (#)</b>	<b>Antes</b>	<b>Después</b>
1	Proceso obsoleto para el sector de alojamiento. (Ver figura 27)	Proceso estandarizado de reservas en línea. (Ver anexo 1)
2	Elaboración de reportes de manera manual.	Proceso automatizado de generación de estadísticas
3	Comparación de información investigando en diferentes sitios web.	Conocer y comparar servicios, prestaciones y precios de mercado en un solo sitio web.
4	Desarrollo de <i>software</i> con metodologías tradicionales.	Generación de valor a la finalización de cada <i>Sprint</i> .
5	Visualización no adaptativa para dispositivos móviles.	Optimización para visualización en dispositivos móviles (Ver anexo 2).
6	Existen diferentes sitios web de hoteles y algunos lugares de alojamiento sin ningún sitio web.	Herramienta en línea que centraliza el sector, que ayuda a las pequeñas empresas en la publicación de información y reservas.

## CAPÍTULO IV

### 4. PRUEBAS DE ACEPTACIÓN Y RESULTADOS

Luego del desarrollo del prototipo y una vez realizados los cambios solicitados en determinados *sprints*, el hotel patrocinador interesado ha desarrollado las pruebas y las verificaciones funcionales y técnicas necesarias del aplicativo. De esta forma, ya puede certificar su correcto funcionamiento y la cobertura de todos los requerimientos técnicos y funcionales acordados a través de las historias de usuario.

#### 4.1. Pruebas de aceptación

Las pruebas de aceptación se realizaron para comprobar que el software cumple con las necesidades y expectativas del cliente.

Tabla 34. Plantilla pruebas de aceptación

PRUEBAS DE ACEPTACIÓN	
Número (#) de Escenario:	Historia de usuario en prueba (#):
Criterio de Aceptación (Título):	
Contexto:	
Evento:	
Resultado:	
Evaluación:	

Tomado de PMOinformatica (s.f.)

En la tabla anterior, se muestra la plantilla utilizada en las pruebas de aceptación la misma que consta de:

- **Número de escenario**

Número en orden ascendente que se utiliza para enumerar cada escenario

- **Historia de usuario en prueba**

Número de la historia de usuario asociada a la prueba de aceptación.

- **Criterio de aceptación (Título)**

Descripción macro del escenario según un comportamiento.

- **Contexto**

Descripción a detalle de las condiciones esperadas.

- **Evento**

Es una acción ejecutada por el usuario final.

- **Resultado**

Consecuencia del comportamiento del prototipo, según el contexto y evento en la prueba de aceptación.

- **Evaluación**

Determina si la prueba de aceptación es o no aprobada.

A continuación se presentarán las pruebas de aceptación realizadas en el desarrollo del presente prototipo.

Tabla 35. Prueba de aceptación 1

PRUEBAS DE ACEPTACIÓN	
<b>Número (#) de Escenario: 1</b>	<b>Historia de usuario en prueba (#): 1</b>
<b>Criterio de Aceptación:</b>	Ciudad sin hoteles.
<b>Contexto:</b>	En caso que exista una ciudad sin hoteles,
<b>Evento:</b>	Cuando el cliente despliegue el listado para seleccionar el mismo,
<b>Resultado:</b>	La aplicación no mostrará un mensaje de error o de no disponibilidad, pero no deberá caerse la aplicación por esta razón.
<b>Evaluación:</b>	<b>Aprobado.</b> Se cumple con los resultados esperados.

Tabla 36. Prueba de aceptación 2

PRUEBAS DE ACEPTACIÓN	
<b>Número (#) de Escenario: 2</b>	<b>Historia de usuario en prueba (#): 2</b>
<b>Criterio de Aceptación:</b>	Fecha anterior
<b>Contexto:</b>	En caso que se intente seleccionar una fecha pasada,
<b>Evento:</b>	Cuando el cliente despliegue cualquier seleccionador de días o fechas,
<b>Resultado:</b>	La aplicación no permitirá dicha selección.
<b>Evaluación:</b>	<b>Aprobado.</b> Se cumple con los resultados esperados.

Tabla 37. Prueba de aceptación 3

PRUEBAS DE ACEPTACIÓN	
<b>Número (#) de Escenario:</b> 3	<b>Historia de usuario en prueba (#):</b> 3,4, 6
<b>Criterio de Aceptación:</b>	Filtros hoteles
<b>Contexto:</b>	<p>En caso que se quiera filtrar los hoteles</p> <ul style="list-style-type: none"> <li>– Dentro de un rango de precios,</li> <li>– categoría y</li> <li>– adicionalmente en orden alfabético,</li> </ul>
<b>Evento:</b>	Cuando el cliente despliegue el listado de hoteles para realizar su reserva,
<b>Resultado:</b>	La aplicación deberá desplegar los hoteles bajo las condiciones señaladas
<b>Evaluación:</b>	<p><b>Aprobado.</b></p> <p>Se cumple con los resultados esperados.</p>

Tabla 38. Prueba de aceptación 4

PRUEBAS DE ACEPTACIÓN	
<b>Número (#) de Escenario:</b> 4	<b>Historia de usuario en prueba (#):</b> 5
<b>Criterio de Aceptación:</b>	Compaginación hoteles
<b>Contexto:</b>	En caso que existan más de 5 hoteles,
<b>Evento:</b>	Cuando el cliente seleccione cualquier ciudad y las condiciones para realizar una reserva,
<b>Resultado:</b>	La aplicación permitirá ver en pantalla 5, 10 o todos los hoteles.
<b>Evaluación:</b>	<p><b>Aprobado.</b></p> <p>Se cumple con los resultados esperados.</p>

Tabla 39. Prueba de aceptación 5

<b>PRUEBAS DE ACEPTACIÓN</b>	
<b>Número (#) de Escenario: 5</b>	<b>Historia de usuario en prueba (#): 7</b>
<b>Criterio de Aceptación:</b>	Galería de imágenes
<b>Contexto:</b>	En caso que se quiera ver a detalle las principales imágenes del hotel,
<b>Evento:</b>	Cuando el cliente seleccione o utilice el cursor sobre una de ellas,
<b>Resultado:</b>	La aplicación mostrará una descripción corta y permitirá hacer un zoom sobre la imagen.
<b>Evaluación:</b>	<b>Aprobado.</b> Se cumple con los resultados esperados.

Tabla 40. Prueba de aceptación 6

<b>PRUEBAS DE ACEPTACIÓN</b>	
<b>Número (#) de Escenario: 6</b>	<b>Historia de usuario en prueba (#): 8</b>
<b>Criterio de Aceptación:</b>	Descripción hoteles
<b>Contexto:</b>	En caso que se enlisten todos o algunos hoteles,
<b>Evento:</b>	Cuando el cliente quiera hacer la reserva,
<b>Resultado:</b>	La aplicación mostrará una corta descripción de cada uno de los hoteles.
<b>Evaluación:</b>	<b>Aprobado.</b> Se cumple con los resultados esperados.

Tabla 41. Prueba de aceptación 7

PRUEBAS DE ACEPTACIÓN	
<b>Número (#) de Escenario: 7</b>	<b>Historia de usuario en prueba (#): 9</b>
<b>Criterio de Aceptación:</b>	Formulario sin datos
<b>Contexto:</b>	En caso que exista un formulario sin completar todos los campos,
<b>Evento:</b>	Cuando el cliente despliegue el formulario para realizar la reservación,
<b>Resultado:</b>	La aplicación mostrará el siguiente texto debajo de la caja de texto  <i>“Oops! Forgot something? Let try again”.</i>
<b>Evaluación:</b>	<b>Aprobado.</b> Se cumple con los resultados esperados.

Tabla 42. Prueba de aceptación 8

PRUEBAS DE ACEPTACIÓN	
<b>Número (#) de Escenario: 8</b>	<b>Historia de usuario en prueba (#): 10</b>
<b>Criterio de Aceptación:</b>	Acceso desde dispositivos móviles
<b>Contexto:</b>	En caso que se quiera realizar una reservación,
<b>Evento:</b>	Cuando el cliente acceda desde un dispositivo móvil,
<b>Resultado:</b>	La aplicación se adaptará a la pantalla del mismo.
<b>Evaluación:</b>	<b>Aprobado.</b> Se cumple con los resultados esperados.

Tabla 43. Prueba de aceptación 9

PRUEBAS DE ACEPTACIÓN	
<b>Número (#) de Escenario:</b> 9	<b>Historia de usuario en prueba (#):</b> 11
<b>Criterio de Aceptación:</b>	Acceso usuarios al administrador
<b>Contexto:</b>	En caso que exista un dato erróneo,
<b>Evento:</b>	Cuando el administrador ingrese los datos en el acceso de usuarios ( <i>login</i> ),
<b>Resultado:</b>	La aplicación mostrará un mensaje de error.
<b>Evaluación:</b>	<b>Aprobado.</b> Se cumple con los resultados esperados.

Tabla 44. Prueba de aceptación 10

PRUEBAS DE ACEPTACIÓN	
<b>Número (#) de Escenario:</b> 10	<b>Historia de usuario en prueba (#):</b> 12,13
<b>Criterio de Aceptación:</b>	Creación hoteles
<b>Contexto:</b>	En caso que no se completen los campos requeridos,
<b>Evento:</b>	Cuando el administrador ingrese los datos para crear hoteles,
<b>Resultado:</b>	La aplicación mostrará el siguiente texto debajo del cuadro "Este campo es requerido".
<b>Evaluación:</b>	<b>Aprobado.</b> Se cumple con los resultados esperados.

Tabla 45. Prueba de aceptación 11

PRUEBAS DE ACEPTACIÓN	
<b>Número (#) de Escenario:</b> 11	<b>Historia de usuario en prueba:</b> 15,17, 19
<b>Criterio de Aceptación:</b>	Creación ciudades, provincias y regiones
<b>Contexto:</b>	En caso que no se completen los campos requeridos,
<b>Evento:</b>	<p>Cuando el administrador ingrese los datos para crear</p> <ul style="list-style-type: none"> <li>– ciudades,</li> <li>– provincias o</li> <li>– regiones,</li> </ul>
<b>Resultado:</b>	La aplicación no habilitará el botón de Guardar
<b>Evaluación:</b>	<p><b>Aprobado.</b></p> <p>Se cumple con los resultados esperados.</p>

Tabla 46. Prueba de aceptación 12

PRUEBAS DE ACEPTACIÓN	
<b>Número de Escenario:</b> 12	<b>Historia de usuario en prueba:</b> 14,16,18,20
<b>Criterio de Aceptación:</b>	Exportar información
<b>Contexto:</b>	En caso que se deseé exportar información,
<b>Evento:</b>	<p>Cuando el administrador ingrese al listado de:</p> <ul style="list-style-type: none"> <li>– hoteles,</li> <li>– ciudades,</li> <li>– provincias o</li> <li>– regiones,</li> </ul>
<b>Resultado:</b>	La aplicación desplegará botones para exportar dicha información en diferentes formatos.
<b>Evaluación:</b>	<p><b>Aprobado.</b></p> <p>Se cumple con los resultados esperados.</p>

Tabla 47. Prueba de aceptación 13

<b>PRUEBAS DE ACEPTACIÓN</b>	
<b>Número de Escenario:</b> 13	<b>Historia de usuario en prueba (#):</b> 21
<b>Criterio de Aceptación:</b>	Edición información
<b>Contexto:</b>	En caso que se deba realizar algún cambio de información,
<b>Evento:</b>	Cuando el administrador seleccione un listado sea de hoteles, ciudades, provincias o regiones,
<b>Resultado:</b>	El sistema deberá mostrar botones de acción: <ul style="list-style-type: none"> <li>- Editar</li> <li>- Eliminar</li> </ul> Y deberá actualizar dicha información en su base de datos.
<b>Evaluación:</b>	<b>Aprobado.</b> Se cumple con los resultados esperados.

Tabla 48. Prueba de aceptación 14

<b>PRUEBAS DE ACEPTACIÓN</b>	
<b>Número de Escenario:</b> 14	<b>Historia de usuario en prueba (#):</b> 22
<b>Criterio de Aceptación:</b>	Reporte reservas
<b>Contexto:</b>	En caso que se deseé información para análisis,
<b>Evento:</b>	Cuando el administrador despliegue el menú de reportes,
<b>Resultado:</b>	El sistema deberá generar reportes estadísticos de cada hotel y contabilizar el número de reservas en un determinado rango de fechas.
<b>Evaluación:</b>	<b>Aprobado.</b> Se cumple con los resultados esperados.

Tabla 49. Prueba de aceptación 15

<b>PRUEBAS DE ACEPTACIÓN</b>	
<b>Número de Escenario:</b> 15	<b>Historia de usuario en prueba (#):</b> 23
<b>Criterio de Aceptación:</b>	Reporte clientes
<b>Contexto:</b>	En caso que se deseé información para análisis,
<b>Evento:</b>	Cuando el administrador ingrese <ul style="list-style-type: none"> <li>– número de cédula o</li> <li>– pasaporte,</li> </ul>
<b>Resultado:</b>	El sistema deberá generar un reporte de las reservas realizadas por un determinado cliente
<b>Evaluación:</b>	<b>Aprobado.</b> Se cumple con los resultados esperados.

Tabla 50. Prueba de aceptación 16

<b>PRUEBAS DE ACEPTACIÓN</b>	
<b>Número de Escenario:</b> 16	<b>Historia de usuario en prueba (#):</b> 24
<b>Criterio de Aceptación:</b>	Reporte habitaciones
<b>Contexto:</b>	En caso que se deseé información para análisis,
<b>Evento:</b>	Cuando el administrador seleccione el reporte,
<b>Resultado:</b>	El aplicativo mostrará la cantidad de habitaciones disponibles de cualquier hotel.
<b>Evaluación:</b>	<b>Aprobado.</b> Se cumple con los resultados esperados.

Tabla 51. Prueba de aceptación 17

<b>PRUEBAS DE ACEPTACIÓN</b>	
<b>Número de Escenario:</b> 17	<b>Historia de usuario en prueba (#):</b> 25
<b>Criterio de Aceptación:</b>	Reporte ganancias
<b>Contexto:</b>	En caso que se deseé información para análisis,
<b>Evento:</b>	Cuando el administrador seleccione el reporte,
<b>Resultado:</b>	El aplicativo generará un reporte de ganancias según el número de reservaciones realizadas en las diferentes provincias dentro de un determinado rango de fechas.
<b>Evaluación:</b>	<b>Aprobado.</b> Se cumple con los resultados esperados.

Tabla 52. Prueba de aceptación 18

<b>PRUEBAS DE ACEPTACIÓN</b>	
<b>Número de Escenario:</b> 18	<b>Historia de usuario en prueba (#):</b> 26
<b>Criterio de Aceptación:</b>	Reporte ingresos
<b>Contexto:</b>	En caso que se deseé información para análisis,
<b>Evento:</b>	Cuando el administrador seleccione el reporte,
<b>Resultado:</b>	El aplicativo generará un reporte con el total de ingresos por cada tipo de habitación en un determinado rango de fechas.
<b>Evaluación:</b>	<b>Aprobado.</b> Se cumple con los resultados esperados.

## 4.2. Resultados

A continuación se detallan en las figuras los reportes desarrollados para el sector de alojamiento, en base a las siguientes métricas:

- Reservas,
- Ingresos,
- Tipos de habitación,
- Entre otros.



*Figura 32.* Reporte 1 visión general

a. Reporte generado por cada hotel seleccionado, se encarga de contabilizar el número de reservas en un determinado rango de fechas.

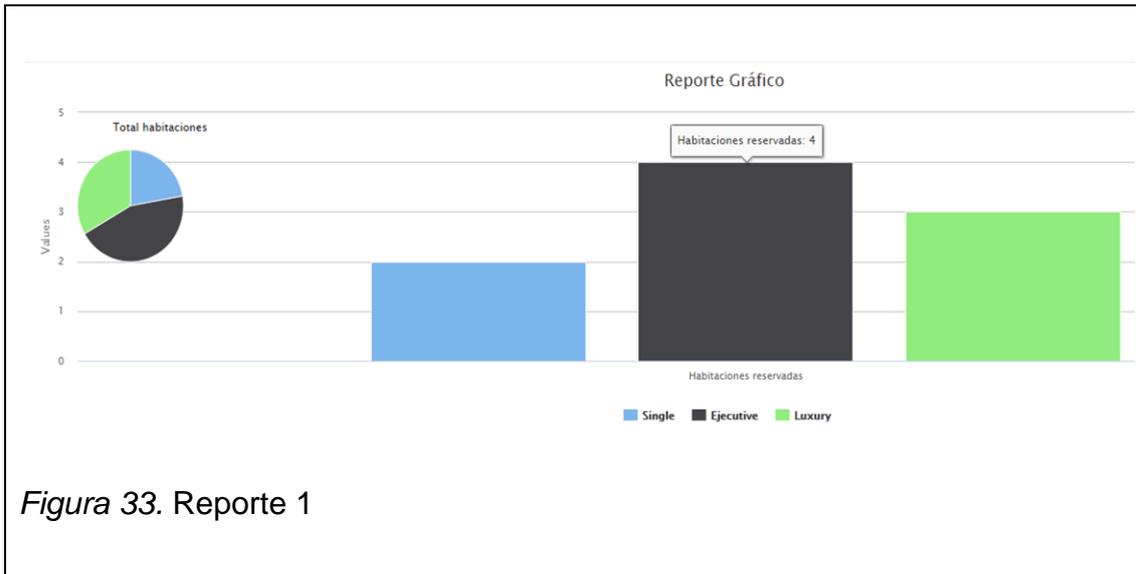


Figura 33. Reporte 1

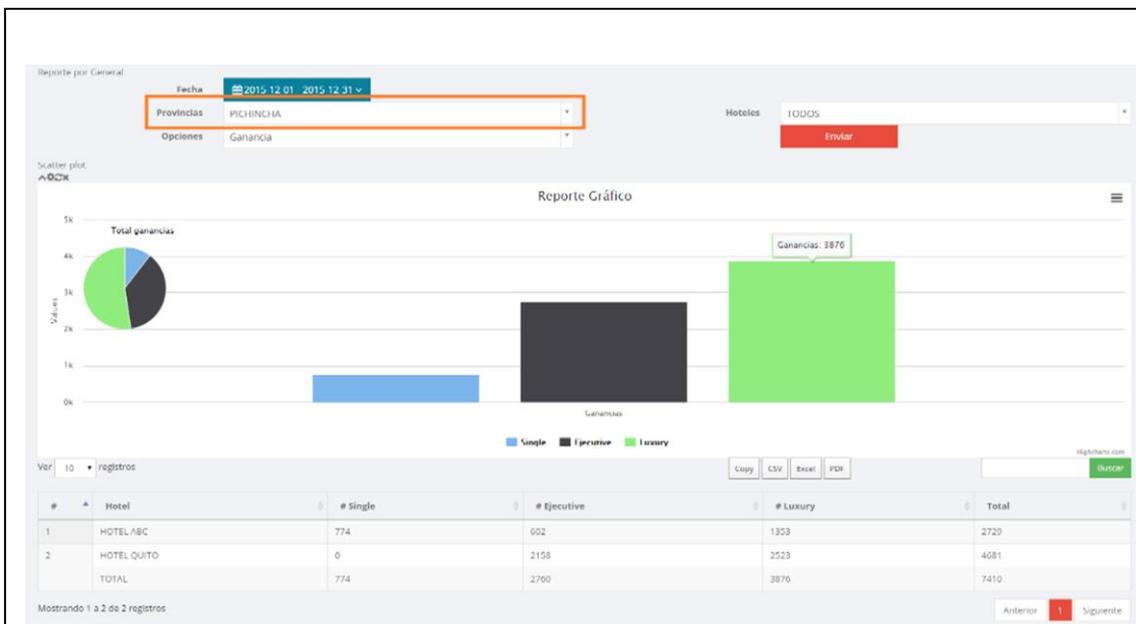


Figura 34. Reporte 2 visión general

- Reporte generado por cada provincia seleccionada, se encarga de desplegar los ingresos según el número de reservaciones realizadas en un determinado rango de fechas.

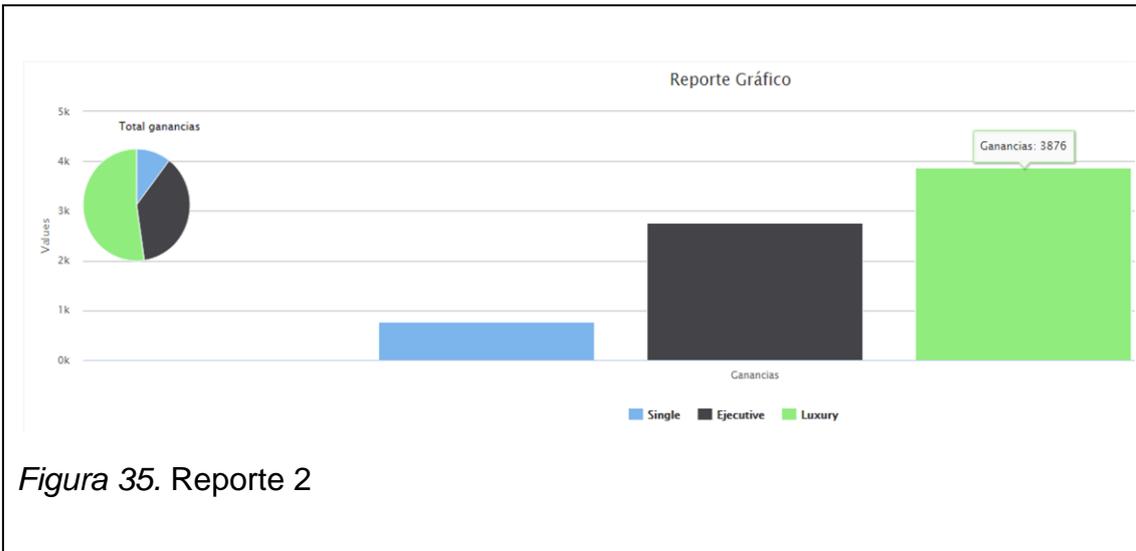


Figura 35. Reporte 2

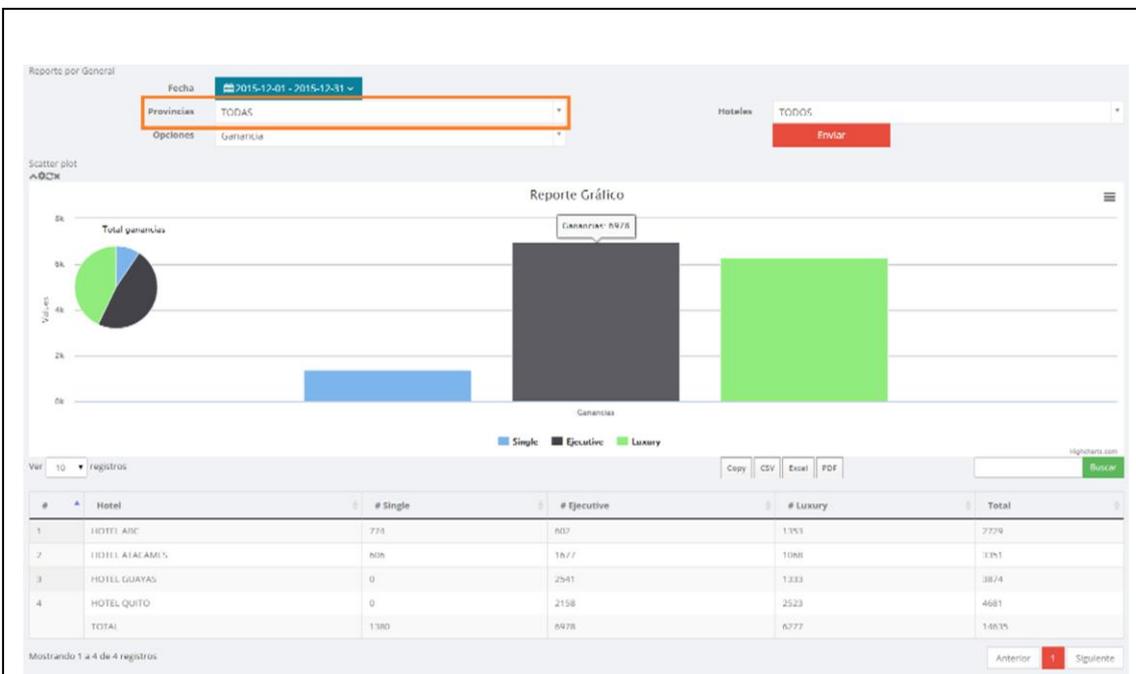
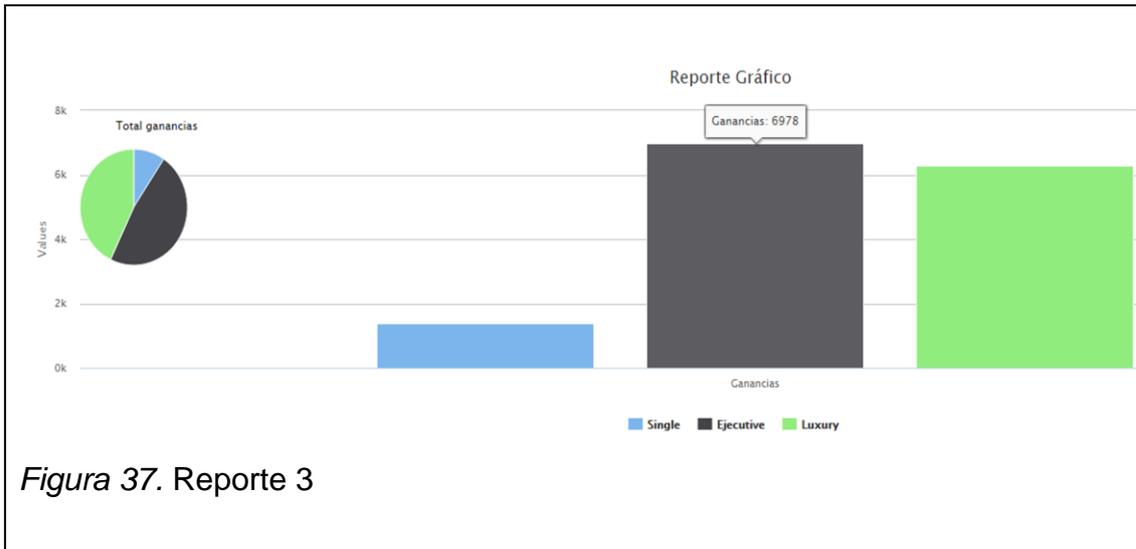


Figura 36. Reporte 3 visión general

a. Reporte generado con el total de ingresos por cada tipo de habitación en un determinado rango de fechas.



Reporte de habitaciones disponibles

### Habitaciones disponibles

Ver  registros

#	Hotel	Ciudad	#Single	#Ejecutive	#Luxurury	Total
1	HOTEL ABC	QUITO	7	14	39	60
2	HOTEL ATACAMES	ATACAMES	19	20	21	60
3	HOTEL GUAYAS	GUAYAQUIL	4	15	21	40
4	HOTEL QUITO	QUITO	7	14	38	59

Mostrando 1 a 4 de 4 registros

Figura 38. Reporte 4

a. Reporte generado para conocer en cualquier instante la cantidad de habitaciones disponibles de cualquier hotel.

Reporte de reservaciones por persona

Identificación

Información del Cliente

Nombres: Jamey Kling Identificación: 1249975583  
 Email: vLemke@Walker.info Teléfono: 08739771

Reservas del Cliente

Ver  registros

#	Fecha Inicio	Fecha Fin	Hotel	Número de Cuarto
1	2015-09-18	2016-01-27	HOTEL ABC	52
2	2015-04-04	2015-05-23	HOTEL QUITO	52
3	2015-07-17	2015-09-19	HOTEL ATACAMES	60
4	2015-10-19	2015-12-25	HOTEL QUITO	29
5	2015-04-02	2015-10-27	HOTEL ATACAMES	47
6	2015-11-12	2015-12-11	HOTEL GUAYAS	12
7	2015-07-21	2015-10-15	HOTEL ATACAMES	10
8	2015-08-12	2015-10-14	HOTEL ABC	21
9	2015-03-27	2015-11-11	HOTEL ATACAMES	32
10	2015-03-05	2015-07-07	HOTEL QUITO	56

Mostrando 1 a 10 de 12 registros

*Figura 39. Reporte de reservaciones por persona*

a. Reporte generado para conocer las reservas realizadas por un determinado cliente.

## CAPÍTULO V

### 5. CONCLUSIONES Y RECOMENDACIONES

Gracias a la investigación realizada se pudo llegar a las siguientes conclusiones y recomendaciones.

#### 5.1. Conclusiones

- Pese a la gran importancia del sector de alojamiento en el país y sus grandes ventajas competitivas, no se ha considerado desarrollar un motor de reservas a nivel nacional, por lo que es necesario un mayor dinamismo en las organizaciones gremiales, gubernamentales y no gubernamentales para mejorar la situación del sector.
- Después de la construcción del prototipo, se concluye que el desarrollo de un motor de reservas es viable porque podría beneficiar al sector de alojamiento y a una gran cantidad de lugares de alojamiento que no poseen un sitio web y mucho menos mecanismos de reservas y reportes para realizar análisis de información. De esta manera se mejora la atracción de visitantes y su alcance a mercados potencialmente amplios.
- De acuerdo a la metodología ágil de desarrollo utilizada se concluye que la principal característica de estas metodologías, que es la de dividir en tiempos cortos las entregas (*Sprint*), ésta retroalimentación continúa genera confianza en el cliente, pues este tiene la certeza de qué es lo que tiene que recibir al final de cada ciclo. De existir algún problema lo conocerá rápidamente y tendrá margen de tiempo para reaccionar según el caso y no cuando sea demasiado tarde. De esta manera, tanto la gerencia del proyecto como los clientes, conocen el estado del proyecto en períodos cortos de tiempo.

- Tomando en cuenta la arquitectura de software, se puede concluir que el aplicativo web puede centrar el movimiento del sector de alojamiento e ir creciendo exponencialmente, hasta el punto que podría interactuar con otros sectores como restaurantes, eventos sociales, bares entre otros.
- El caso de estudio concluyó que mediante la estandarización de procesos se obtuvo un óptimo desenvolvimiento en la realización de reservas en los ambientes de prueba.
- Finalmente, se concluye que la hipótesis del proyecto de titulación sí se cumple mediante el desarrollo de un aplicativo web con interfaz adaptativa (*responsive*). Se trata de una excelente opción para satisfacer la necesidad de generar reportes de manera automática con datos estadísticos. Su uso principalmente está destinado a los agremiados de la Federación Hotelera del Ecuador para una efectiva toma de decisiones.

## 5.2. Recomendaciones

Para la consecución de los objetivos planteados por la Asociación Hotelera Nacional del Ecuador (AHOTEC), y su plan de acción, en primer lugar se recomienda optar por una solución tecnológica que brinde valor al sector de alojamiento mediante un motor de reservas que recopile información y permita la generación de reportes estadísticos para la toma de decisiones.

Con el fin de informar sobre esta herramienta al sector de alojamiento, es fundamental trabajar en conjunto con otras organizaciones para llevar a cabo capacitaciones en línea (webinars) destinadas a agencias de turismo nacionales e internacionales.

El proceso de comercialización de reservas de habitación en hoteles del país, debería llevarse a cabo en conjunto con: tour operadores, agencias de viaje, agencias de transporte de turismo

Además, es recomendable recibir una retroalimentación por parte de todas las asociaciones hoteleras del país para mejorar continuamente el aplicativo web.

Es necesaria la actualización en el diseño de la página web oficial de Hoteles del Ecuador, con el fin de estar a la par con otros sitios web de la región y competir internacionalmente.

Se recomienda a futuro adicionar al aplicativo una galería de fotos más amplia y con destinos favoritos de los viajeros, para así captar mucho mejor la atención de los usuarios para la selección de determinado lugar.

Por último, a futuro es necesario el desarrollo e integración con motores de reserva de las grandes cadenas hoteleras del país. De esta manera se cubrirían en gran parte las necesidades de todos los agremiados, creando una solución multicapa con una interface adicional de servicios web (*Web Services*), o un sistema de bases de datos distribuidas.

## REFERENCIAS

Acuña, C. (2014). *Arquitecturas de Software*. Recuperado el 06 de septiembre de 2015 de <http://pegasus.javeriana.edu.co/~mad/Arquitecturas%20de%20SW.pdf>

AHOTEC. (2012). *Reglamento General de Actividades Turísticas (Decreto No.3400)*. Recuperado el 26 de julio de 2015 de <http://www.hotelesecuador.com.ec/downloads/Reglamento%20General%20de%20Actividades%20Turisticas.pdf>

Alonso, F., Martínez, L. y Segovia, F. (2005). *Introducción a la Ingeniería de Software Modelo de desarrollo de programas*. (1.<sup>a</sup> ed.). Zaragoza, España: Delta Publicaciones Universitarias.

Americalatinapmi. (s.f.). Dirección de Proyectos. Recuperado el 19 de septiembre de 2015 de <https://americalatina.pmi.org/latam/AboutUS/QueEsLaDireccionDeProyectos.aspx>

Bahit, E. (2011). *POO y MVC en PHP*. Buenos Aires, Argentina: Autoedición.

Bredemeyer, D. (2002). *The Visual Architecting Process*. Recuperado el 29 de Agosto de 2015 de <http://prof.usb.ve/Imendoza/Documentos/PS-6116/Guia%20Arquitectura%20v.2.pdf>

Buschmann, F. (1996). *Pattern-Oriented Software Architecture – A System of Patterns*. West Sussex, Inglaterra: John Wiley & Sons Ltd.

Caicedo, I., Peña, P. y Hernández, M. (2015). *Arquitecturas de Software*. Recuperado el 29 de agosto de 2015 de [http://www.aves.edu.co/ovaunicor/recursos/1/index\\_Arquitectura\\_de\\_software.pdf](http://www.aves.edu.co/ovaunicor/recursos/1/index_Arquitectura_de_software.pdf)

Caminoagil. (s.f.). Qué es en realidad la agilidad. Recuperado el 16 de septiembre de 2015 de <http://www.caminoagil.com/2014/11/que-es-en-realidad-la-agilidad.html>

Cic.javerianacali. (s.f.). Patrón de Arquitectura de Software MVC. Recuperado el 12 de septiembre de 2015 de <http://cic.javerianacali.edu.co/wiki/lib/exe/fetch.php?media=materias:patronmvc.pdf>

Citón, L. (2006). *Método Ágil Scrum*. Buenos Aires, Argentina: Universidad de Mendoza.

Clements, P., Rachman, F., Bass, L., Garlan, D., Ivers, J., Little, R., Merson, P., Nord, R., y Stafford, J. (2010). *Documenting Software Architecture: Views and Beyond*. (2.<sup>a</sup> ed.). Texas, Estados Unidos: Addison-Wesley Professional.

Cohn, M. (2005). *Agile Estimating and Planning*. New Jersey, Estados Unidos: Prentice Hall.

Crispin, L. y Gregory, J. (2009). *Agile Testing*. New Jersey, Estados Unidos: Addison-Wesley.

CSSLab. (s.f.). Tramo crítico de renderizado. Recuperado el 20 de diciembre de 2015 de <http://www.csslab.cl/2014/11/26/conoce-como-funciona-el-browser-y-optimiza-tu-codigo-front-end/>

E2eWorks. (2009). *Modelo conceptual de Arquitectura Multicapas (N-Tier)*. Recuperado el 12 de septiembre de 2015 de [http://www.e2ew.com/espanol/arquitectura\\_modelo.htm](http://www.e2ew.com/espanol/arquitectura_modelo.htm)

Ecured. (2009). Arquitectura de tres niveles. Recuperado el 19 de junio de 2016 de [http://www.ecured.cu/Arquitectura\\_de\\_tres\\_niveles](http://www.ecured.cu/Arquitectura_de_tres_niveles)

- Gamma, E., Helm, R., Johnson, R. y Vlissides, J. (1997). *Design Patterns: Elements of Reusable object-Oriented Software*. Illinois, Estados Unidos: Addison-Wesley.
- Garzas, J. (2014). *Aprende a implantar integración continua desde cero*. Recuperado el 24 de diciembre de 2015 de <http://www.javiergarzas.com/2014/08/implantar-integracion-continua.html>
- Instituto Superior Politécnico. (2012). *Patrón Modelo-Vista-Controlador*. (11.<sup>a</sup> ed.). Habana, Cuba: Instituto Superior Politécnico.
- Janium. (s.f.). Janium adopta una arquitectura multicapa. Recuperado el 12 de septiembre de 2015 de <http://www.janium.com/janium-adopta-arquitectura-multicapa/>
- Kazman, R., Bass, L., Barbacii, M., Carriere, J., Klein, M. y Lipson, H. (1999). *Attribute Based Arquitectural Styles*. Recuperado el 29 de agosto de 2015 de <http://prof.usb.ve/lmendoza/Documentos/PS-6116/Guia%20Arquitectura%20v.2.pdf>
- Maldonado, F. y Proaño, G. (2015). *Comunidad de negocios para restaurantes, hoteles, cafetería y catering*. (18<sup>a</sup> ed.). Quito, Ecuador: La Barra.
- Maldonado, F. y Proaño, G. (2016). *Comunidad de negocios para restaurantes, hoteles, cafetería y catering*. Recuperado el 14 de julio de 2016 de [http://www.labarra.ec/revistas/id\\_r/18#/page/3](http://www.labarra.ec/revistas/id_r/18#/page/3)
- Managedagile. (s.f.). Agile Project Management. Recuperado el 19 de septiembre de 2015 de <http://managedagile.com/2014/06/29/what-is-a-project/>

Palacio, A. (2007). *Flexibilidad con Scrum*. Zaragoza, España: Safe Creative.

Palacio, J. (2014). *Gestión de proyectos de Scrum Manager*. Recuperado el 19 de septiembre de 2015 de <http://www.scrummanager.net/>

PivotalTracker. (s.f.). Build Better Software Faster. Recuperado el 16 de enero 2016 de <https://www.pivotaltracker.com/n/projects/1482238>

PMOinformatica. (s.f.). Plantillas Scrum: historias de usuario y criterios de aceptación. Recuperado el 28 de diciembre de 2015 de <http://www.pmoinformatica.com/2012/10/plantillas-scrum-historias-de-usuario.html>

Schwaber, K. y Sutherland, J. (2011). *La Guía de Scrum*. Recuperado el 10 de octubre de 2015 de <http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-ES.pdf>

Scrummanager. (s.f.). Historias de usuario. Recuperado el 01 de diciembre de 2015 de [http://www.scrummanager.net/bok/index.php?title=Historia\\_de\\_usuario](http://www.scrummanager.net/bok/index.php?title=Historia_de_usuario)

Scrum Alliance. (s.f.). What is sprint zero?. Recuperado el 17 de diciembre de 2015 de <https://www.scrumalliance.org/community/articles/2013/september/what-is-sprint-zero>

SlideShare. (s.f.). Arquitectura multicapa. Recuperado el 12 de septiembre de 2015 de <http://es.slideshare.net/maryme/arquitectura-multicapa>

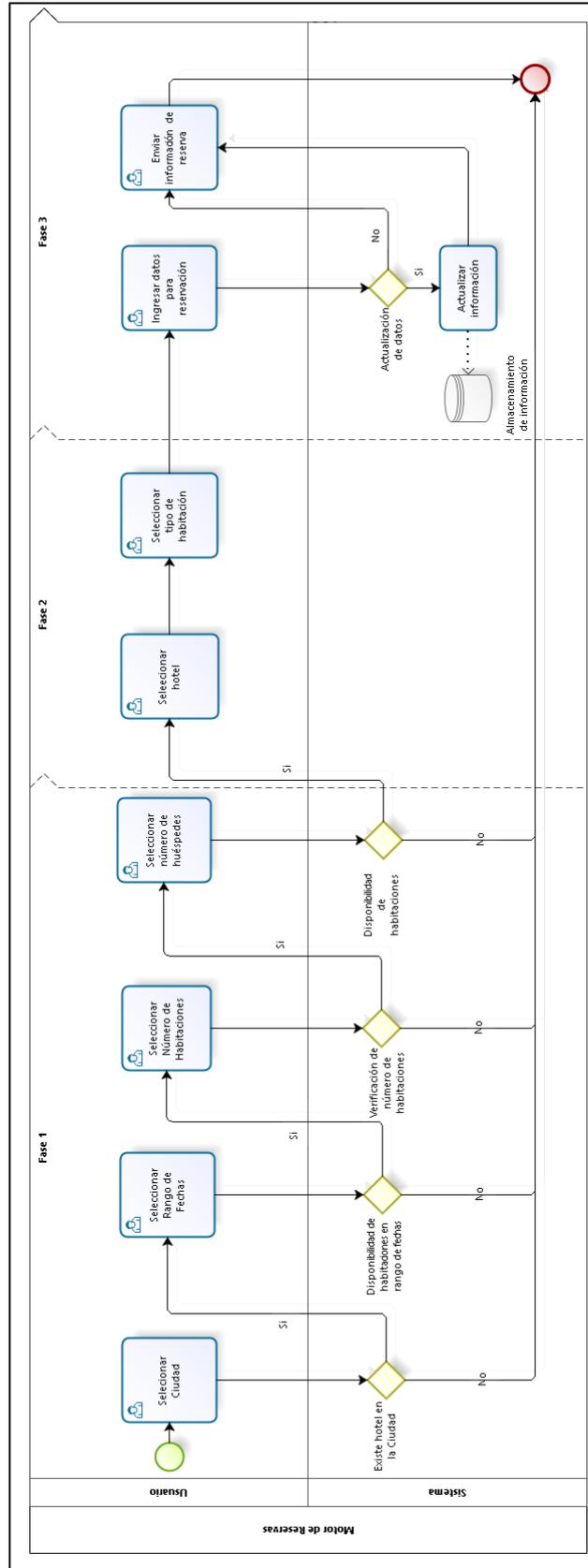
SlideShare. (s.f.). Metodología ágil para gestión de proyectos. Recuperado el 03 de octubre de 2015 de <http://es.slideshare.net/RobertoVelazquezB/scrum-v1>

Twileshare. (s.f.). Introducción a la Agilidad y Scrum. Recuperado el 10 de octubre de 2015 de <http://twileshare.com/uploads/EFF46d01.pdf>

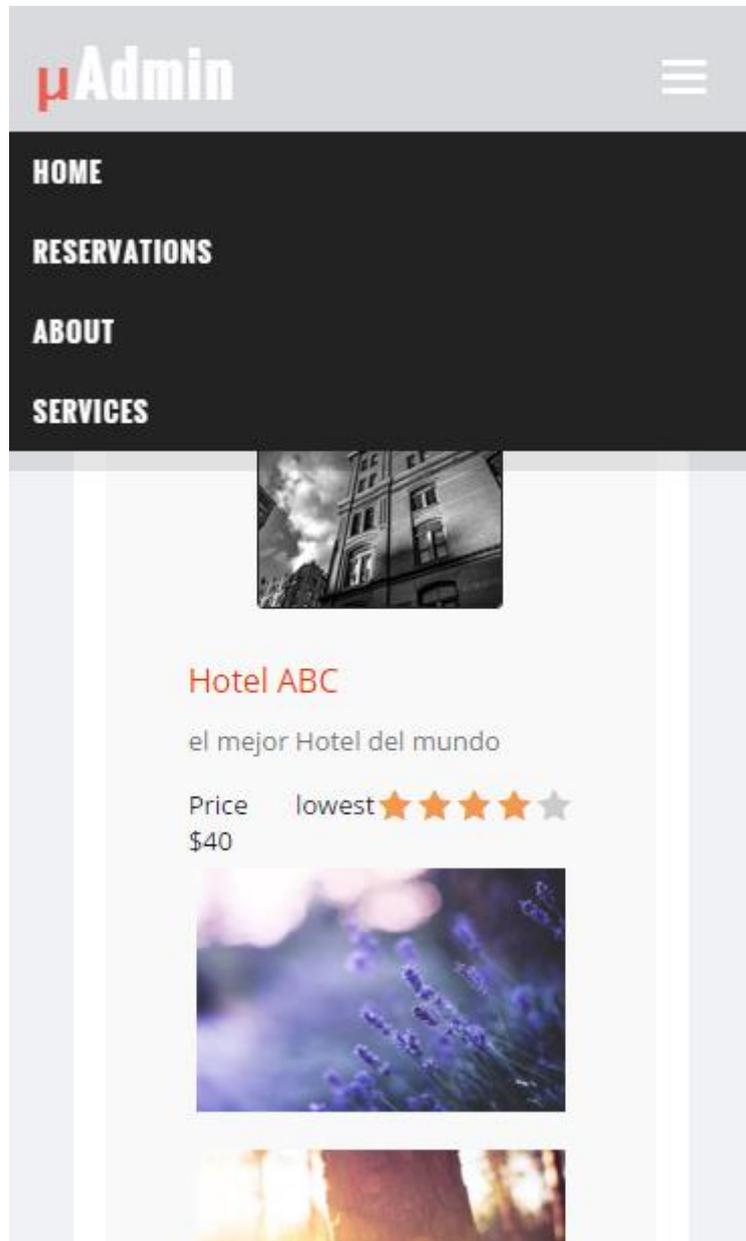
Versionone. (s.f.). 8th Annual State of Agile Survery. Recuperado el 10 de octubre de 2015 de <https://www.versionone.com/pdf/2013-state-of-agile-survey.pdf>

## **ANEXOS**

# Anexo 1. Mapa de proceso motor de reservas



## Anexo 2. Escenario de FrontEnd (Apple iPhone 6)



### Anexo 3. Escenario de FrontEnd (Computador Personal)

**μAdmin** HOME RESERVATIONS ABOUT SERVICES

Reset  10 per page  Title A-Z

\$0  \$1000      1 estrellas  5 estrellas

Page 1 of 1

**Hotel ABC**  
el mejor Hotel del mundo  
Price lowest \$40



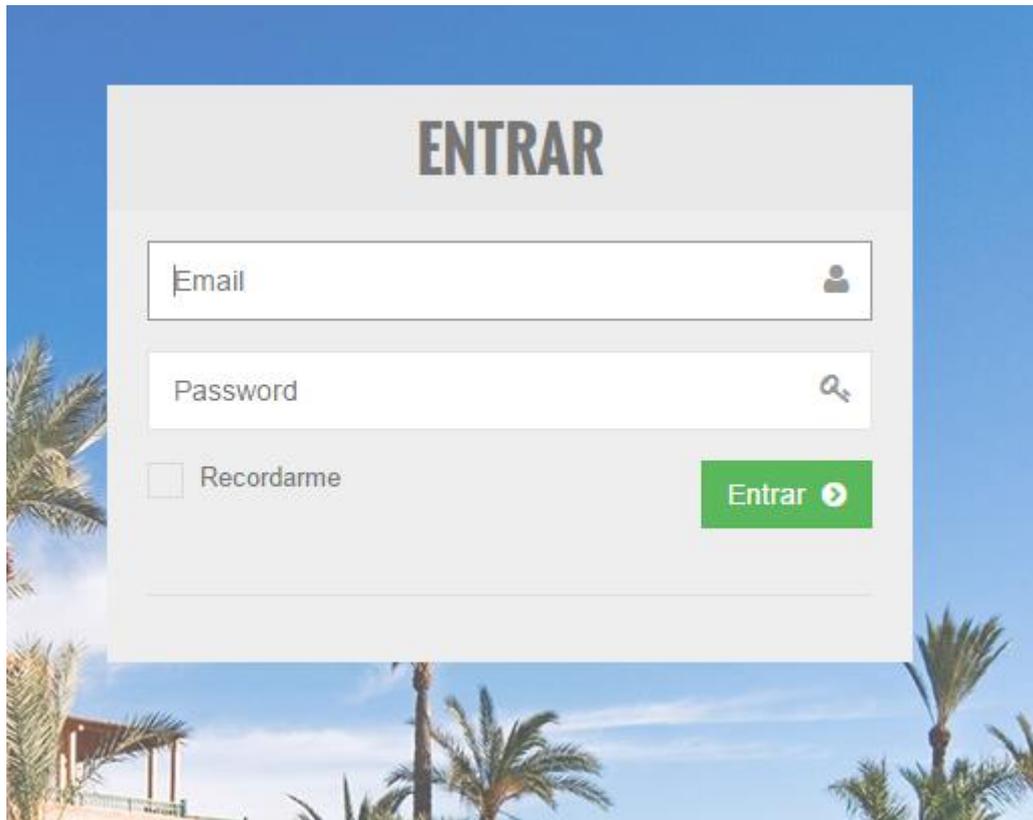
★★★★★

**HOTEL QUITO**  
Price lowest \$10

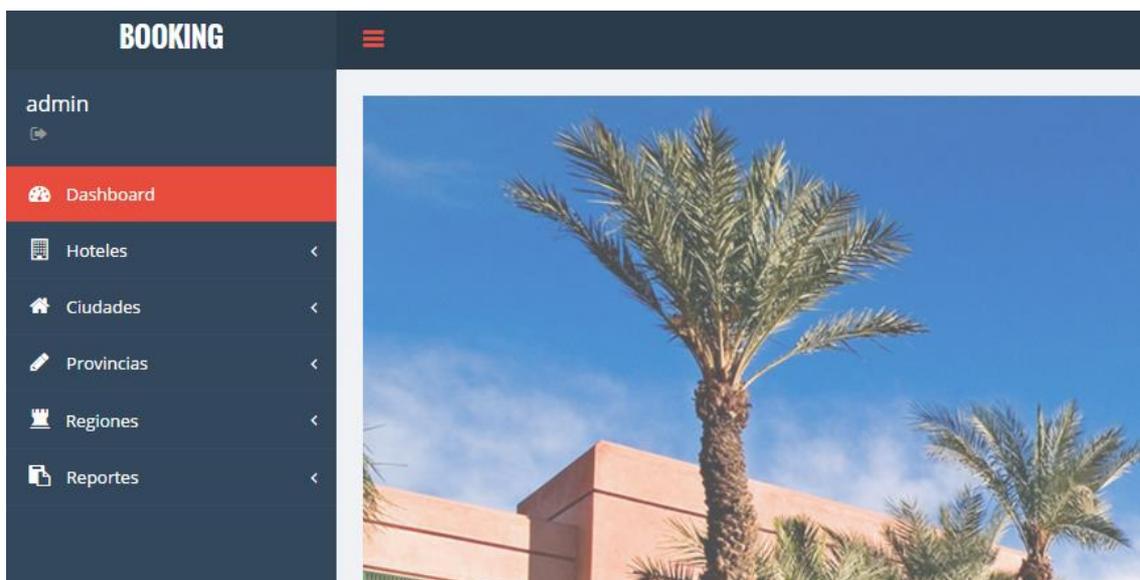


★★★★★

#### Anexo 4. Escenario de Backend (Acceso usuarios)



#### Anexo 5. Escenario de Backend (Administrador)



## Anexo 6. Reporte del progreso realizado en las historias de usuario en un lapso de tiempo determinado

### Generate Progress Report

Projects  
Prototipo Booking ▼

Date Range  
8/1/2015 to 1/16/2016

► Display Options

**GENERATE REPORT**

---

#### Progress Report: Prototipo Booking (Aug 1, 2015 to Jan 16, 2016)

---

★ Elegir ciudad de la reserva (Accepted)

2015-12-03 03:55PM PET	Fernando Piedra started this feature
2015-12-03 04:02PM PET	Fernando Piedra finished this feature
2015-12-03 04:02PM PET	Fernando Piedra delivered this feature
2015-12-03 04:02PM PET	Fernando Piedra accepted this feature

---

★ Seleccionar dentro de un rango de fechas (Accepted)

2015-12-03 04:01PM PET	Fernando Piedra started this feature
2015-12-03 04:18PM PET	Fernando Piedra finished this feature
2015-12-03 04:18PM PET	Fernando Piedra delivered this feature
2015-12-03 04:18PM PET	Fernando Piedra accepted this feature

---

★ Filtrar por precio (Accepted)

2015-12-03 04:12PM PET	Fernando Piedra moved and scheduled this story after 'Seleccionar dentro de un rango de fechas'
2015-12-03 04:18PM PET	Fernando Piedra started this feature
2015-12-03 04:18PM PET	Fernando Piedra finished this feature
2015-12-03 04:18PM PET	Fernando Piedra delivered this feature
2015-12-03 04:18PM PET	Fernando Piedra accepted this feature

---

★ Filtrar por nombre del hotel (Accepted)

2015-12-03 04:15PM PET	Fernando Piedra moved and scheduled this story after 'Filtrar por precio'
2015-12-03 04:18PM PET	Fernando Piedra started this feature
2015-12-03 04:18PM PET	Fernando Piedra finished this feature
2015-12-03 04:18PM PET	Fernando Piedra delivered this feature
2015-12-03 04:18PM PET	Fernando Piedra accepted this feature

---

★ Compaginación (Accepted)

2015-12-03 04:45PM PET	Fernando Piedra started this feature
2015-12-03 04:48PM PET	Fernando Piedra finished this feature
2015-12-03 04:48PM PET	Fernando Piedra delivered this feature
2016-01-14 09:54PM PET	Fernando Piedra accepted this feature

---

★ Filtrar por categoría (Accepted)

2015-12-03 04:45PM PET	Fernando Piedra started this feature
2016-01-14 09:54PM PET	Fernando Piedra finished this feature
2016-01-14 09:54PM PET	Fernando Piedra delivered this feature
2016-01-14 09:54PM PET	Fernando Piedra accepted this feature

---

## Anexo 7. Certificación de funcionalidad implementada Hostería Castellmar



### CERTIFICACIÓN DE FUNCIONALIDAD IMPLEMENTADA

Quito, D.M. 22 de enero de 2016

Luego del desarrollo del prototipo para realizar reserva de habitaciones de hotel, junto con los cambios solicitados en determinados *sprints*, la Hostería Castellmar, patrocinador requirente, ha realizado las pruebas y verificaciones funcionales y técnicas necesarias del aplicativo para certificar su correcto funcionamiento, así como la cobertura de todos los requerimientos técnicos y funcionales acordados a través de las historias de usuario.

Consecuentemente, luego de realizadas las pruebas funcionales y validación de las condiciones de certificación necesarias, la Hostería Castellmar formaliza a través del presente documento la certificación del desarrollo de cada uno de las historias solicitadas sobre el aplicativo y todos los entregables acordados, y certifica que el mismo se encuentra funcional y a conformidad de acuerdo a lo requerido, por lo tanto se encuentra apto para uso del sector de alojamiento, considerando mejoras antes de entrar a producción.

A continuación se detallan los productos certificados:

N.	PRODUCTO CERTIFICADO
	Desarrollar un aplicativo web que permita:
1	Motor de reservas que ofrece alojamiento en determinada región
2	Administrador de contenidos
3	Reportes

Suscriben el presente documento

Elaborado por:	Revisado por:	Aprobado por:
----- Nombre: Fernando Piedra Veintimilla	----- Nombre:	 ----- Nombre: Carmen Anda V.

## Anexo 8. Tarjeta de presentación patrocinador



## Anexo 9. Certificación de funcionalidad implementada Hotel Costa Brava

### CERTIFICACIÓN DE FUNCIONALIDAD IMPLEMENTADA

Quito, D.M. 22 de enero de 2016

Luego del desarrollo del prototipo para realizar reserva de habitaciones de hotel, junto con los cambios solicitados en determinados *sprints*, el Hotel Costa Brava Atacames patrocinador requirente ha realizado las pruebas y verificaciones funcionales y técnicas necesarias del aplicativo para certificar su correcto funcionamiento, así como la cobertura de todos los requerimientos técnicos y funcionales acordados a través de las historias de usuario.

Consecuentemente, luego de realizadas las pruebas funcionales y validación de las condiciones de certificación necesarias, el Hotel Costa Brava Atacames, formaliza a través del presente documento la certificación del desarrollo de cada uno de las historias solicitadas sobre el aplicativo y todos los entregables acordados, y certifica que el mismo se encuentra funcional y a conformidad de acuerdo a lo requerido, por lo tanto se encuentra apto para uso del sector de alojamiento, considerando mejoras antes de entrar a producción.

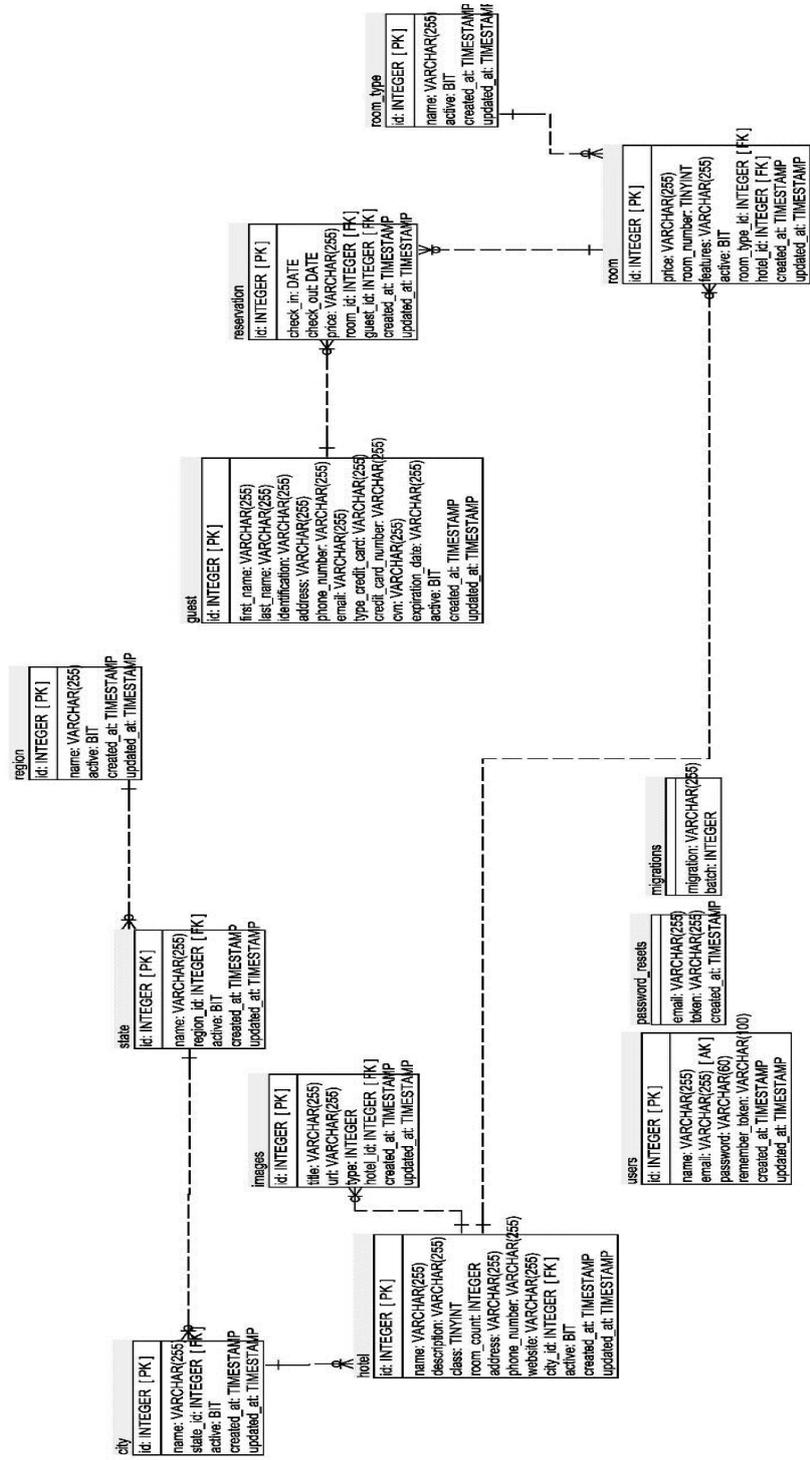
A continuación se detallan los productos certificados:

N.	PRODUCTO CERTIFICADO
Desarrollar un aplicativo web que permita:	
1	Motor de reservas que ofrece alojamiento en determinada región
2	Administrador de contenidos
3	Reportes

Suscriben el presente documento

Elaborado por:	Revisor por:	Aprobado por:
----- Nombre: Fernando Piedra Veintimilla	----- Nombre:	 ----- Nombre: Sandra de Cisneros

# Anexo 10. Diagrama Entidad / Relación



## Anexo 11. Acta de cierre del proyecto



Quito, 2 de Febrero de 2016

**Señores**

Departamento de Tecnologías & Innovación Digital  
Corporación EKOS

**Presente.-**

Atención:

**Diego Álvarez B.**  
Director del Departamento de Tecnología

Asunto: Finalización del proyecto de desarrollo de *"prototipo en plataforma web para reserva de habitaciones de hotel y análisis de la información"*

De mis consideraciones:

En relación al proyecto para la elaboración de un aplicativo en plataforma web para reserva de habitaciones de hotel y análisis de la información para la toma de decisiones basado en las reservas, se realiza la entrega final de los siguientes documentos / entregables del proyecto desarrollado, conforme al cronograma establecido para dicho proyecto,



Detalles de acta de entrega

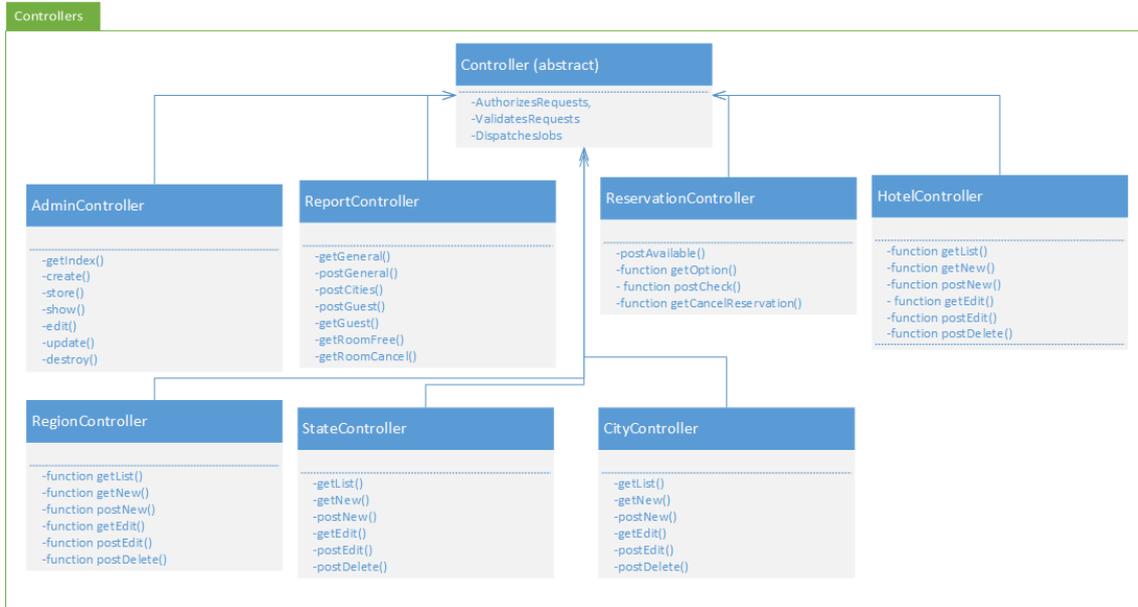
	Documento / Entregable	Referencia / Observación
Diseño de la Aplicación	Cronograma del proyecto	Se entrega el documento en el capítulo 2
	Artefactos de planificación, análisis, modelos generados.	Se entrega el documento: Anexo 01
	Código revisado versionado según estándares de la organización, aprobado por el área de Proyectos TI.	Se entrega el documento adjunto.
	Scripts de creación de la base de datos.	Se entrega el documento adjunto.
	Historias de usuario	Se entrega el documento adjunto.
	Diagrama Entidad / Relación	Se entrega el documento Anexo 10

Entregado por

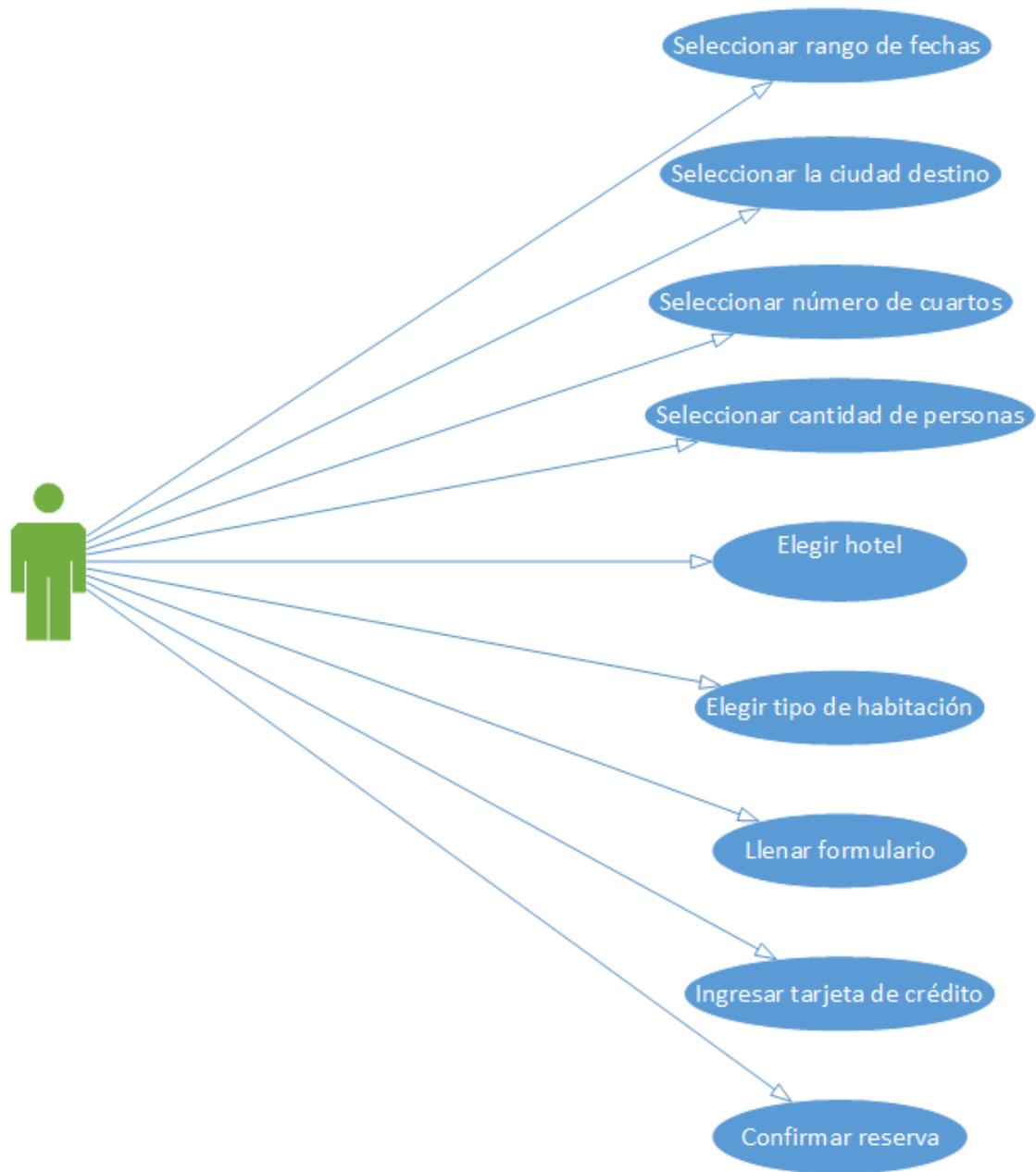
Recibido por

STARGROUP CIA. LTDA.

## Anexo 12. Diagrama de clases



### Anexo 13. Caso de uso para realizar una reserva



## Anexo 14. Entrevista cualitativa a líder de proyectos de software



1. **Una vez realizadas las reservas de habitación en el prototipo, ¿considera usted que esta podría ser una herramienta de gran utilidad para el sector hotelero?**

Tomando en cuenta la funcionalidad de la aplicación, considero que sería de gran utilidad una herramienta como estas en el Ecuador, ayudaría de gran manera al sector hotelero, ya que pondría al alcance de la comunidad nacional e internacional toda la información disponible y necesaria para tomar decisiones a la hora de emprender un viaje ya sea de negocios o familiar

2. **¿Ha respondido a una velocidad adecuada el prototipo al realizar consultas específicas, haciendo uso de los filtros de categoría hotelera, desarrollados en la plataforma?**

Considerando el tiempo de respuesta de las consultas realizadas y la facilidad de filtros a la hora de buscar, puedo decir que el tiempo de respuesta es aceptable, tomando en cuenta que se trata de un prototipo, creo que una vez puesto en producción se tomarán las medidas necesarias para mejorar dicho tiempo y pueda dar respuesta al sin número de usuarios que la aplicación pueda tener

3. **¿Considera usted que los reportes generados en la aplicación son de utilidad para los gobiernos autónomos descentralizados o al sector hotelero y turístico para una buena toma de decisiones?**

Desde el punto de vista administrativo, la información presentada en los reportes es de calidad, deberían verla desde el lado de los hoteleros para que de alguna manera haya una retroalimentación y puedan construir una herramienta que colabore en el crecimiento de este sector.

Ing. Juan Carlos Pérez., MBA  
Líder de proyectos de software

## Anexo 15. Entrevista cualitativa a funcionario de ComputerWorld Ecuador



1. Una vez realizadas las reservas de habitación en el prototipo, ¿considera usted que esta podría ser una herramienta de gran utilidad para el sector hotelero?

Si, creo que haría la diferencia de valor agregado para quienes buscan un servicio en línea y rápido.

2. ¿Ha respondido a una velocidad adecuada el prototipo al realizar consultas específicas, haciendo uso de los filtros de categoría hotelera, desarrollados en la plataforma?

Si, no tuve problemas al probarlo.

3. ¿Considera usted que los reportes generados en la aplicación son de utilidad para los gobiernos autónomos descentralizados o al sector hotelero y turístico para una buena toma de decisiones?

Si los saben usar si. Creo que se debe capacitar a las personas encargadas para que sepan leer la analítica.

A handwritten signature in blue ink, appearing to read "Daniel Loaiza Rodríguez".

Ing. Daniel Loaiza Rodríguez

Subgerente de Contenidos Digitales

## Anexo 16. Carta de recomendación por parte del Ministerio de Turismo del Ecuador



Quito, D.M. 15 de enero del 2016

A quien interese:

Ante el trabajo de investigación del Señor Fernando Piedra, previo la obtención de su titulación a "Ingeniero en Sistema en Computación e Informática" y luego de haber revisado el prototipo para realizar reservas de habitaciones de hotel y vistos los reportes basados en: Reservas, Ingresos y Tipos de habitación, me permito manifestar su correcto funcionamiento; además, de expresar que es una aplicación técnica para la eficiencia, eficacia y efectividad para la comercialización en el sector de alojamiento del país

En este sentido, este tipo de solución web contribuye a potenciar la comercialización de los servicios turísticos hoteleros, la cual es aplicable para portales web de entidades como la Asociación de Hoteles del Ecuador (AHOTEC), los gobiernos locales, seccionales y provinciales que cuentan con la competencia de turismo, dada su misión en materia de promoción turística.

Atentamente,

Ing. Rocio Castro Castro  
Dirección de Mercados Internacionales

