



INSTITUTO TECNOLÓGICO
“CORDILLERA”

CARRERA DE ANÁLISIS DE SISTEMAS

MEJORAMIENTO DE UN SISTEMA PARA RASTREO DE HOTELES
DEFINIDOS EN EL DISTRITO METROPOLITANO DE QUITO MEDIANTE
UNA APP PARA DISPOSITIVOS MÓVILES EN LA EMPRESA STAFF
VACATION CIA. LTDA.

Proyecto de investigación previo a la obtención del Título de Tecnólogo en Análisis
de Sistemas

Autor: Darío Xavier Yanacallo Yanacallo

Tutor: Ing. Roberto Morales

Quito, Octubre 2015

DECLARACION DE APROBACIÓN DE TUTOR Y LECTOR

MEJORAMIENTO DE UN SISTEMA PARA RASTREO DE HOTELES
DEFINIDOS EN EL DISTRITO METROPOLITANO DE QUITO MEDIANTE
UNA APP PARA DISPOSITIVOS MÓVILES EN LA EMPRESA STAFF
VACATION CIA. LTDA.

DECLARACION DE AUTORIA DEL ESTUDIANTE

Declaro que la investigación es absolutamente original, autentica, personal, que se citado las fuentes correspondientes y que en su ejecución se representaron las disposiciones legales que protegen los derechos de autor vigentes. Las ideas, doctrina, resultados y conclusiones a lo que he llegado son de mi absoluta responsabilidad.

Darío Xavier Yanacallo Yanacallo

C.C. 1724141088

CONTRATO DE CESIÓN SOBRE DERECHOS PROPIEDAD INTELECTUAL

Comparecen a la celebración del presente contrato de cesión y transferencia de derechos de propiedad intelectual, por una parte, el estudiante Darío Xavier Yanacallo Yanacallo, por sus propios y personales derechos, a quien en lo posterior se le denominará el “CEDENTE”; y, por otra parte, el INSTITUTO SUPERIOR TECNOLÓGICO CORDILLERA, representado por su Rector el Ingeniero Ernesto Flores Córdova, a quien en lo posterior se lo denominará el “CESIONARIO”. Los comparecientes son mayores de edad, domiciliados en esta ciudad de Quito Distrito Metropolitano, hábiles y capaces para contraer derechos y obligaciones, quienes acuerdan al tenor de las siguientes cláusulas:

PRIMERA: ANTECEDENTE.- a) El Cedente dentro del pensum de estudio en la carrera de análisis de sistemas que imparte el Instituto Superior Tecnológico Cordillera, y con el objeto de obtener el título de Tecnólogo en Análisis de Sistemas, el estudiante participa en el proyecto de grado denominado “Mejoramiento de un sistema para rastreo de hoteles definidos en el D.M.Q., mediante una App para dispositivos móviles para la empresa STAFF VACATIONS CIA. LTDA”, el cual incluye la creación y desarrollo del programa de ordenador o software, para lo cual ha implementado los conocimientos adquiridos en su calidad de alumno. b) Por iniciativa y responsabilidad del Instituto Superior Tecnológico Cordillera se desarrolla la creación del programa de ordenador, motivo por el cual se regula de forma clara la cesión de los derechos de autor que genera la obra literaria y que es producto del

proyecto de grado, el mismo que culminado es de plena aplicación técnica, administrativa y de reproducción.

SEGUNDA: CESIÓN Y TRANSFERENCIA.- Con el antecedente indicado, el Cedente libre y voluntariamente cede y transfiere de manera perpetua y gratuita todos los derechos patrimoniales del programa de ordenador descrito en la cláusula anterior a favor del Cesionario, sin reservarse para sí ningún privilegio especial (código fuente, código objeto, diagramas de flujo, planos, manuales de uso, etc.). El Cesionario podrá explotar el programa de ordenador por cualquier medio o procedimiento tal cual lo establece el Artículo 20 de la Ley de Propiedad Intelectual, esto es, realizar, autorizar o prohibir, entre otros: a) La reproducción del programa de ordenador por cualquier forma o procedimiento; b) La comunicación pública del software; c) La distribución pública de ejemplares o copias, la comercialización, arrendamiento o alquiler del programa de ordenador; d) Cualquier transformación o modificación del programa de ordenador; e) La protección y registro en el IEPI el programa de ordenador a nombre del Cesionario; f) Ejercer la protección jurídica del programa de ordenador; g) Los demás derechos establecidos en la Ley de Propiedad Intelectual y otros cuerpos legales que normen sobre la cesión de derechos de autor y derechos patrimoniales.

TERCERA: OBLIGACIÓN DEL CEDENTE.- El cedente no podrá transferir a ningún tercero los derechos que conforman la estructura, secuencia y organización del programa de ordenador que es objeto del presente contrato, como tampoco emplearlo o utilizarlo a título personal, ya que siempre se deberá guardar la exclusividad del programa de ordenador a favor del Cesionario.

CUARTA: CUANTIA.- La cesión objeto del presente contrato, se realiza a título gratuito y por ende el Cesionario ni sus administradores deben cancelar valor alguno o regalías por este contrato y por los derechos que se derivan del mismo.

QUINTA: PLAZO.- La vigencia del presente contrato es indefinida.

SEXTA: DOMICILIO, JURISDICCIÓN Y COMPETENCIA.- Las partes fijan como su domicilio la ciudad de Quito. Toda controversia o diferencia derivada de éste, será resuelta directamente entre las partes y, si esto no fuere factible, se solicitará la asistencia de un Mediador del Centro de Arbitraje y Mediación de la Cámara de Comercio de Quito. En el evento que el conflicto no fuere resuelto mediante este procedimiento, en el plazo de diez días calendario desde su inicio, pudiendo prorrogarse por mutuo acuerdo este plazo, las partes someterán sus controversias a la resolución de un árbitro, que se sujetará a lo dispuesto en la Ley de Arbitraje y Mediación, al Reglamento del Centro de Arbitraje y Mediación de la Cámara de comercio de Quito, y a las siguientes normas: a) El árbitro será seleccionado conforme a lo establecido en la Ley de Arbitraje y Mediación; b) Las partes renuncian a la jurisdicción ordinaria, se obligan a acatar el laudo arbitral y se comprometen a no interponer ningún tipo de recurso en contra del laudo arbitral; c) Para la ejecución de medidas cautelares, el árbitro está facultado para solicitar el auxilio de los funcionarios públicos, judiciales, policiales y administrativos, sin que sea necesario recurrir a juez ordinario alguno; d) El procedimiento será confidencial y en derecho; e) El lugar de arbitraje serán las instalaciones del centro de arbitraje y mediación de la Cámara de Comercio de Quito; f) El idioma del arbitraje será el español; y, g) La reconvenición, caso de haberla, seguirá los mismos procedimientos antes indicados para el juicio principal.

SÉPTIMA: ACEPTACIÓN.- Las partes contratantes aceptan el contenido del presente contrato, por ser hecho en seguridad de sus respectivos intereses.

En aceptación firman a los 19 días del mes de Octubre del dos mil quince.

f) _____

C.C. N° 1724141088

CEDENTE

f) _____

Instituto Superior Tecnológico Cordillera

CESIONARIO

AGRADECIMIENTO

Primeramente agradezco a Dios por darme salud, vida, sabiduría, paciencia para poder afrontar este difícil reto de terminar con éxito la tesis.

A mis abuelitos que fueron el pilar fundamental que junto con mi tío supieron darme ánimos, fuerzas, ganas de seguir adelante y ayudarme a cumplir mi sueño de culminar los estudios superiores.

A mis maestros quienes me inculcaron de buenos conocimientos profesionales y técnicos, por ello me siento muy orgulloso de haberla elegido esta carrera ya que ellos contribuyeron en mi proceso de formación no solo profesional si no personal y a mis amigos que han hecho que la vida en el Instituto sea llena de experiencias maravillosas que me han ayudado a crecer como persona y ser humano.

DEDICATORIA

Este proyecto está dedicado a Dios por ser el que hizo posible este sueño de culminar la carrera dándome sabiduría, aguante y ganas de seguir adelante.

El presente proyecto está dedicado a tres personas fundamentales en mi vida; mis dos abuelitos maternos y mi tío Raúl que siempre están presentes en mi vida apoyándome moralmente y dándome fuerzas para seguir adelante.

A mi madre la cual también estuvo pendiente de mi desempeño educativo, apoyándome económicamente y dándome ánimos para llegar a mí objetivo planteado.

A mi hermana Tatiana que siempre me dio ánimos para seguir adelante y no quedarme a medio camino de cumplir este sueño maravilloso.

Y a mi padre que también estuvo ayudándome de muchas maneras, y a cada uno de ellos que depositó su confianza en mí.

INDICE GENERAL

DECLARACION DE APROBACIÓN DE TUTOR Y LECTOR	i
DECLARACION DE AUTORIA DEL ESTUDIANTE.....	ii
CONTRATO DE CESIÓN SOBRE DERECHOS PROPIEDAD INTELECTUAL.....	iii
AGRADECIMIENTO.....	vii
DEDICATORIA.....	viii
RESUMEN EJECUTIVO	xxii
ABSTRACT	xxiv
INTRODUCCION	xxvi
Capítulo I: Antecedentes.....	1
1.01 Contexto	1
1.02 Justificación	2
1.03 Definición de problema	3
Capítulo II: Análisis de Involucrados.....	5
2.01. Requerimientos	5
2.01.1. Descripción del sistemas actual:	5
2.01.2. Visión y alcance.....	5
2.01.3. Entrevista	7
2.01.4. Matriz de requerimientos	8
2.01.5. Descripción detallada.....	9

2.02.	Mapeo de Involucrados	12
2.03.	Matriz de Involucrados	13
Capítulo III: Problemas y Objetivos		14
3.01.	Árbol de Problemas.....	14
3.02.	Árbol de Objetivos	15
3.03.	Diagramas de caso de uso	16
3.04.	Especificaciones de caso de uso.....	17
3.05.	Caso de uso de realización	18
3.06.	Diagramas de secuencia del sistema	20
Capítulo IV: Análisis de Alternativas.....		23
4.01.	Matriz de Análisis de Alternativas	23
4.02.	Matriz de Impactos de Objetivos.....	23
4.03.	Estándares para el Diseño de Clases.....	23
4.04.	Diagrama de Clases.....	26
4.05.	Modelo físico	26
4.06.	Diagrama de Componentes	27
4.07.	Diagrama de Estrategias	28
4.08.	Matriz de Marco Lógico.	29
4.09.	Vistas arquitectónicas	30
4.09.01.	Vista lógica	30
4.09.02.	Vista física.....	30

4.09.03.	Vista de desarrollo	31
4.09.04.	Vista de procesos	32
Capítulo V: Propuesta.....		33
5.01.	Especificación de estándares de programación	33
5.02.	Diseño de interfaces de usuario	34
5.03.	Especificación de pruebas de unidad	49
5.04.	Especificación de pruebas de aceptación	52
5.05.	Especificación de pruebas de carga	55
5.06.	Configuración del ambiente mínima/ideal	63
CAPÍTULO VI: ASPECTOS ADMINISTRATIVOS.....		66
6.01.	Recursos	66
6.02.	Presupuestos.....	67
6.03.	Cronograma	67
CAPÍTULO VII: CONCLUSIONES Y RECOMENDACIONES.....		68
7.01.	Conclusiones.....	68
7.02.	Recomendaciones	68
ANEXOS.....		69
Anexo A.01.	Matriz de análisis de Alternativas	70
Anexo A.02.	Matriz de Impacto de Objetivos.....	71
Anexo A.03.	Especificaciones de estándares de programación	72
Anexo A.04.	Cronograma de actividades.....	73

Anexo A.05. Manual de instalación de aplicaciones	74
Anexo A.06. Manual de administrador y usuario	91
Manual para el administrador	91
Manual de usuario.....	96
Anexo A.07. Manual técnico	101
Script de la base de datos.	120
Bibliografía.....	133

INDICE DE TABLAS

Tabla 1. Matriz de fuerzas T	4
Tabla 2. Escala valorativa	4
Tabla 3. Entrevista	7
Tabla 4. Matriz de Requerimientos	8
Tabla 5. Descripción de Requerimiento RF001	9
Tabla 6. Descripción de Requerimiento RF002	10
Tabla 7. Descripción de Requerimiento NRF001	11
Tabla 8. Matriz de Involucrados	13
Tabla 9. Especificaciones de caso de Uso	17
Tabla 10. Matriz de Marco Lógico	29
Tabla 11. Descripción del ingreso al sistema	34
Tabla 12. Detalle de usuarios	35
Tabla 13. Detalle de ingreso de usuario	35
Tabla 14. Detalle de sectores	36
Tabla 15. Detalle de ingreso de nuevo sector	37
Tabla 16. Detalle de roles	37
Tabla 17. Detalle de ingreso de nuevo rol	38
Tabla 18. Detalle de hoteles	38
Tabla 19. Detalle de ingreso de nuevo hotel	39
Tabla 20. Detalle de los hoteles	40
Tabla 21. Ingreso de nuevo servicio	40
Tabla 22. Detalle de atributos	41
Tabla 23. Detalle de ingreso a la App	42

Tabla 24.	Registro de nuevo usuario en la App	43
Tabla 25.	Detalle de los sectores	44
Tabla 26.	Detalle de hoteles, búsqueda y cerrar sesión	45
Tabla 27.	Detalle de servicios de hotel	46
Tabla 28.	Detalle de como desea llegar al hotel requerido	47
Tabla 29.	Búsqueda de hoteles	48
Tabla 30.	Especificación de la prueba de unidad PDU001	50
Tabla 31.	Especificación de la prueba de unidad PDU002	50
Tabla 32.	Especificación de la prueba de unidad PDU003	51
Tabla 33.	Especificación de la prueba de unidad PDU004	51
Tabla 34.	Especificación de la prueba de unidad PDU005	52
Tabla 35.	Especificación de prueba de aceptación EPDA001 aprobar App	52
Tabla 36.	Especificación de prueba de aceptación EPDA002 desarrollo App	53
Tabla 37.	Especificación de prueba de aceptación EPDA003 entrega App	53
Tabla 38.	Especificación de prueba de aceptación EPDA004 registro de cliente	54
Tabla 39.	Especificación de prueba de aceptación EPDA005 uso de la App	54
Tabla 40.	Especificación de la prueba de carga EPC001.	55
Tabla 41.	Especificación de la prueba de carga EPC002	60
Tabla 42.	Recursos humanos y económicos	66
Tabla 43.	Detalle de los equipos utilizados	66
Tabla 44.	Presupuesto del proyecto	67
Tabla 45.	Matriz de alternativas	70
Tabla 46.	Matriz de impacto	71
Tabla 47.	Estándares de programación	72

INDICE DE FIGURAS

<i>Figura 1. Mapeo de involucrados.....</i>	<i>12</i>
<i>Figura 2. Árbol de problemas.....</i>	<i>14</i>
<i>Figura 3. Árbol de objetivos.....</i>	<i>15</i>
<i>Figura 4. Diagrama de casos de uso.....</i>	<i>16</i>
<i>Figura 5. Caso de uso de realización (Aprobación App).....</i>	<i>18</i>
<i>Figura 6. Caso de uso de realización (Compra del sistema)</i>	<i>18</i>
<i>Figura 7. Caso de uso de realización (Desarrollo App).....</i>	<i>19</i>
<i>Figura 8. Caso de uso de realización (Entrega App).....</i>	<i>19</i>
<i>Figura 9. Caso de uso de realización (Usuario).....</i>	<i>20</i>
<i>Figura 10. Diagrama de secuencia (Aprobación App).....</i>	<i>20</i>
<i>Figura 11. Diagrama de secuencia (Compra App).....</i>	<i>21</i>
<i>Figura 12. Diagrama de secuencia (Desarrollo App)</i>	<i>21</i>
<i>Figura 13. Diagrama de secuencia (Entrega App).....</i>	<i>22</i>
<i>Figura 14. Diagrama de secuencia (Usuario).....</i>	<i>22</i>
<i>Figura 15. Clase.....</i>	<i>23</i>
<i>Figura 16. Atributos.....</i>	<i>23</i>
<i>Figura 17. Métodos.....</i>	<i>24</i>
<i>Figura 18. Primary Key (código principal).....</i>	<i>24</i>
<i>Figura 19. Foreign Key (código heredado).....</i>	<i>24</i>
<i>Figura 20. Asociación de objetos.....</i>	<i>25</i>
<i>Figura 21. Generalización de objetos</i>	<i>25</i>
<i>Figura 22. Relación.....</i>	<i>25</i>
<i>Figura 23. Diagrama de clases.....</i>	<i>26</i>

<i>Figura 24. Modelo Físico</i>	<i>26</i>
<i>Figura 25. Diagrama de Componentes</i>	<i>27</i>
<i>Figura 26. Diagrama de Estrategias.....</i>	<i>28</i>
<i>Figura 27. Vista lógica.....</i>	<i>30</i>
<i>Figura 28. Vista física</i>	<i>30</i>
<i>Figura 29. Vista de desarrollo</i>	<i>31</i>
<i>Figura 30. Vista de procesos</i>	<i>32</i>
<i>Figura 31. Conexión a la base de datos.....</i>	<i>33</i>
<i>Figura 32. Ingreso al sistema</i>	<i>34</i>
<i>Figura 33. Usuarios del sistema</i>	<i>35</i>
<i>Figura 34. Ingreso de nuevo usuario</i>	<i>36</i>
<i>Figura 35. Sectores de ubicación de los hoteles.....</i>	<i>36</i>
<i>Figura 36. Ingreso de nuevo sector.....</i>	<i>37</i>
<i>Figura 37. Roles de usuarios</i>	<i>37</i>
<i>Figura 38. Ingreso de nuevo rol</i>	<i>38</i>
<i>Figura 39. Detalle de hoteles.....</i>	<i>38</i>
<i>Figura 40. Ingreso de nuevo hotel</i>	<i>39</i>
<i>Figura 41. Detalle de los servicios de hoteles</i>	<i>40</i>
<i>Figura 42. Descripción de un servicio</i>	<i>40</i>
<i>Figura 43. Atributos de un hotel.....</i>	<i>41</i>
<i>Figura 44. Ingreso a la App.....</i>	<i>42</i>
<i>Figura 45. Registro de nuevo usuario</i>	<i>43</i>
<i>Figura 46. Detalle de los sectores</i>	<i>44</i>
<i>Figura 47. Lista de hoteles, búsqueda y cerrar sesión.....</i>	<i>45</i>

<i>Figura 48. Servicios y ubicación de hoteles</i>	<i>46</i>
<i>Figura 49. Llegar al hotel deseado</i>	<i>47</i>
<i>Figura 50. Búsqueda de un hotel</i>	<i>48</i>
<i>Figura 51. Software para realizar las pruebas de carga</i>	<i>56</i>
<i>Figura 52. Inicio de la pruebas de carga</i>	<i>57</i>
<i>Figura 53. Pruebas realizadas</i>	<i>57</i>
<i>Figura 54. Cantidad de solicitudes realizadas por el usuario</i>	<i>58</i>
<i>Figura 55. Velocidad de transmisión de solicitudes enviadas al servidor</i>	<i>58</i>
<i>Figura 56. Tiempo de carga de las páginas ejecutadas</i>	<i>59</i>
<i>Figura 57. Informe general de las pruebas de carga con dos usuarios</i>	<i>59</i>
<i>Figura 58. Interfaz de LoadUI Web para 15 usuarios</i>	<i>61</i>
<i>Figura 59. Pruebas de carga con 15 usuarios, solicitudes enviadas por el usuario</i>	<i>61</i>
<i>Figura 60. Pruebas de carga con 15 usuarios, tiempo de carga de las paginas</i>	<i>62</i>
<i>Figura 61. Pruebas de carga con 15 usuarios, transmisión de solicitudes</i>	<i>62</i>
<i>Figura 62. Pruebas de carga con 15 usuarios, información de la prueba generada</i>	<i>63</i>
<i>Figura 63. Cronograma de actividades para el desarrollo del sistema</i>	<i>73</i>
<i>Figura 64. NET descarga del framework</i>	<i>74</i>
<i>Figura 65. Inicio de instalación de Mysql</i>	<i>74</i>
<i>Figura 66. Aceptación de licencia MySQL</i>	<i>75</i>
<i>Figura 67. Opción de instalación completa de MySQL</i>	<i>75</i>
<i>Figura 68. Finalización de la instalación</i>	<i>75</i>
<i>Figura 69. Instalación de mysql workbench</i>	<i>76</i>
<i>Figura 70. Aceptación de licencia del workbench</i>	<i>76</i>
<i>Figura 71. Iniciación de la instalación del workbench</i>	<i>76</i>

Figura 72. Opción de instalación developer machine, server machine, dedicated MySQL Server.....	77
Figura 73. Opción multifuncional Data base	77
Figura 74. Agregar usuarios y contraseñas	78
Figura 75. Asignar contraseña al usuario creado.....	78
Figura 76. Final de la instalación de MySQL.....	79
Figura 77. Descargar e instalar el java Development Kit (JDK).....	79
Figura 78. Descargar Netbeans.....	80
Figura 79. Comienzo de la instalación	80
Figura 80. Elegir la carpeta de instalación y JDK.....	81
Figura 81. Utilización de Netbeans	82
Figura 82. Inicio de instalación.....	82
Figura 83. Términos y condiciones de la instalación	82
Figura 84. Seleccionamos la dirección donde se instalara.....	83
Figura 85. Progreso de instalación	83
Figura 86. Selección de navegador.....	84
Figura 87. Instalación por defecto.....	84
Figura 88. Finalización de la instalación	85
Figura 89. Seleccionaremos el idioma.....	85
Figura 90. Descarga de Android Studio	86
Figura 91. Inicio de la instalación.....	86
Figura 92. Seleccionamos los componentes que queremos instalar (seleccionar todas las funciones disponibles).....	87
Figura 93. Aceptamos los términos de uso.....	87

<i>Figura 94. Selección de ruta de instalación.....</i>	<i>88</i>
<i>Figura 95: Selección de memoria.....</i>	<i>88</i>
<i>Figura 96: Proceso de instalación.....</i>	<i>89</i>
<i>Figura 97. Instalación completa (siguiente).....</i>	<i>89</i>
<i>Figura 98. Descarga de componentes SDK.....</i>	<i>90</i>
<i>Figura 99. Instalación completa.....</i>	<i>90</i>
<i>Figura 100. Ingreso al sistema</i>	<i>91</i>
<i>Figura 101. Añadir nuevo.....</i>	<i>91</i>
<i>Figura 102. Botón modificar</i>	<i>92</i>
<i>Figura 103. Botón eliminar</i>	<i>92</i>
<i>Figura 104. Ingreso al sistema</i>	<i>92</i>
<i>Figura 105. Clic en el botón de nuevo sector.....</i>	<i>93</i>
<i>Figura 106. Ingresar nuevo sector.....</i>	<i>93</i>
<i>Figura 107. Clic nuevo hotel</i>	<i>94</i>
<i>Figura 108. Ingreso de todos los datos del nuevo hotel.....</i>	<i>94</i>
<i>Figura 109. Clic en servicio</i>	<i>95</i>
<i>Figura 110. Ingreso detallado de los servicios.....</i>	<i>95</i>
<i>Figura 111. Instalar Apk</i>	<i>96</i>
<i>Figura 112. Finaliza la instalación.....</i>	<i>96</i>
<i>Figura 113. Abrimos la App instalada</i>	<i>97</i>
<i>Figura 114. Nuevo registro</i>	<i>97</i>
<i>Figura 115. Inicio de sesión</i>	<i>98</i>
<i>Figura 116. Sectores de Quito</i>	<i>98</i>
<i>Figura 117. Hoteles de un sector.....</i>	<i>99</i>

Figura 118.	Búsqueda de hoteles.....	99
Figura 119.	Servicios de un hotel	100
Figura 120.	Como llegar al hotel.....	100
Figura 121.	Mapa de ubicación.....	101
Figura 122.	Interfaz de ingreso al sistema	102
Figura 123.	Código para ingreso al sistema	102
Figura 124.	Interfaz de usuario	103
Figura 125.	Código para visualizar los usuarios	103
Figura 126.	Interfaz de ingreso nuevo usuario.....	104
Figura 127.	Código para ingresar nuevo usuario parte 1	104
Figura 128.	Código de validación de contraseña para ingreso de usuario parte 2	105
Figura 129.	Código de validación de contraseña para ingreso de usuario parte 3	105
Figura 130.	Método para borrar y modificar usuario	106
Figura 131.	Método para registrar nuevo usuario	106
Figura 132.	Interfaz de hoteles	107
Figura 133.	Código para visualizar los hoteles parte 1.....	107
Figura 134.	Código para visualizar los hoteles parte 2.....	108
Figura 135.	Código para confirmar eliminar un hotel	108
Figura 136.	Interfaz de ingreso nuevo hotel.....	109
Figura 137.	Código para modificar e ingresar nuevo hotel.....	109
Figura 138.	Código para verificar si el hotel existe al momento de ingresar nuevo y borrar hotel	110
Figura 139.	Interfaz de registro nuevo usuario en Androi.....	111
Figura 140.	Código de interfaz, registro de nuevo usuario en Android	111

<i>Figura 141. Código Java de registro nuevo usuario en Android parte 1</i>	<i>112</i>
<i>Figura 142. Código Java de registro nuevo usuario en Android parte 2</i>	<i>112</i>
<i>Figura 143. Interfaz de ingreso al sistema en android</i>	<i>113</i>
<i>Figura 144. Código de interfaz para el ingreso al sistema de android</i>	<i>113</i>
<i>Figura 145. Código Java para el ingreso al sistema de android parte 1</i>	<i>114</i>
<i>Figura 146. Código Java para el ingreso al sistema de android parte 2</i>	<i>114</i>
<i>Figura 147. Interfaz de los sectores de Quito, android.....</i>	<i>115</i>
<i>Figura 148. Código de interfaz de los sectores en android.....</i>	<i>115</i>
<i>Figura 149. Código Java para la carga de sectores en android.....</i>	<i>116</i>
<i>Figura 150. Código Java para obtener datos de los sectores en android</i>	<i>116</i>
<i>Figura 151. Interfaz de la ubicación de los hoteles, android.....</i>	<i>117</i>
<i>Figura 152. Código de interfaz para la ubicación de los hoteles parte 1.....</i>	<i>117</i>
<i>Figura 153. Código de interfaz para la ubicación de los hoteles parte 2.....</i>	<i>118</i>
<i>Figura 154. Código Java para la obtención de datos (coordenadas), android</i>	<i>118</i>
<i>Figura 155. Código Java para recibir coordenadas y mostrar el mapa, android .</i>	<i>119</i>
<i>Figura 156. Código Java para cerrar sesión en android y re direccionar al login</i>	<i>119</i>

RESUMEN EJECUTIVO

Este proyecto tiene como objetivo mostrar una aplicación de ubicación hotelera para los turistas, dando así la apertura de desarrollar a un Sistema de geo localización para el D.M.Q.

La garantía de este proyecto es aplicar el conocimiento adquirido durante el último semestre en el instituto, materias como PHP, Desarrollo de sistemas operativos, Proyectos, así como también las demás asignaturas de los semestres pasados como base de datos y programación orientada a objetos.

Consideramos muy importante el desarrollo de dicha aplicación ya que es en beneficio del campo laboral de las empresas turísticas, sabemos que la aplicación será de gran apoyo y será muy utilizado para la fomentación del área turística en la ciudad.

Para ello se ha dividido en siete capítulos para poder desarrollar de una forma efectiva y entendible, como por ejemplo en el capítulo uno y dos se detalla la situación de la empresa, los problemas que tiene y así poder mejorar con la implementación de este sistema.

En el siguiente capítulo detallamos la situación actual de la empresa para luego plantear los objetivos principales de la empresa a la que quiere llegar.

En el cuarto capítulo se llega a analizar las alternativas, luego ver las distintas formas de solución o cambio que la empresa debe efectuar, en este capítulo también se encuentra el marco lógico el cual nos indica los resultados previos del proyecto, otro punto muy importante que se detalla aquí es la estructura de la base de datos y su representación en el diagrama de clases, modelo lógico, físico y la vista de desarrollo del proyecto.

En el capítulo cinco se detalla la propuesta del proyecto donde podemos ver los estándares de programación es decir los lenguajes de programación que hemos utilizado para la implementación del sistema y a su vez el diseño de las interfaces de usuario, a continuación también se detalla cada una de las pruebas que se hace al sistema para su correcto funcionamiento, a la vez el ambiente ideal para el funcionamiento del sistema.

Los recursos y presupuestos para la elaboración del proyecto se detalla en el capítulo seis y también el cronograma de trabajo, para ello se debe considerar desde el inicio (aprobación del tema) hasta la finalización del proyecto.

Como último capítulo tenemos el siete donde van las conclusiones es decir los resultados obtenidos durante el desarrollo del sistema, también las recomendaciones que se puede dar después de haber desarrollado el sistema.

Para luego terminar con los anexos y la bibliografía de donde se obtuvo ayuda durante todo el desarrollo del proyecto.

ABSTRACT

This project aims to show an application of hotel location for tourists, giving the opening to develop a geo localization system for DMQ

The guarantee of this project is to apply the knowledge acquired during the last semester in high school subjects such as PHP, development of operating systems, projects, as well as other subjects of past semesters database and object-oriented programming.

We consider very important the development of this application as it is for the benefit of the labor field of tourism businesses, we know that the application will be of great support and will be widely used for the furthering of the tourist area in the city.

For it has been divided into seven chapters to develop an effective and understandable way, such as in chapter one two the situation of the company, the problems you have and thus improve with the implementation of this system is detailed.

In the next chapter we detail the current situation of the company and then pose the main objectives of the company you want to go.

In the fourth chapter comes analyze the alternatives, then see the different forms of solution or change an enterprise should, in this chapter is also the logical framework which indicates the previous results of the project, another very important point detailed here is the structure of the database and its representation in the class diagram, logical, physical model and the view of the project.

In chapter five the project proposal where we can see the programming standards ie programming languages we have used to implement the system and turn the design

of user interfaces, then also detailed each detailed testing is done to the system for proper operation while the ideal system operation environment.

Resources and budgets for the drafting detailed in chapter six and the work schedule, this will be considered from the beginning (approval of the issue) until completion of the project.

As a last chapter which will have seven conclusions ie the results obtained during the development of the system, also the recommendations that can be given after developing the system.

And then end with the annexes and the literature of where help is obtained throughout the project.

INTRODUCCION

Este sistema se desarrolla en la empresa STAFF VACATIONS CIA. LTDA. Esta tiene como finalidad establecer un sistema de geo localización hotelera para facilitar a los turistas que llegan a la ciudad a ubicar con facilidad los hoteles.

El software consta de dos módulos, el primero que es para la parte administrativa que su desarrollo se ha efectuado en php con base de datos MySQL y lo segundo que está desarrollado para el usuario el cual tendrá manipulación de dicho sistema y está desarrollado en Android Studio ya que es el lenguaje más común para el desarrollo de App para dispositivos móviles.

Uno de los principales objetivos del presente sistema es la fácil ubicación y en poco tiempo de los hoteles del D.M.Q., dando gran beneficio a los usuarios ya que uno de los principales problemas es la falta de conocimiento de los hoteles que existe.

También permitirá a la empresa STAFF VACATIONS brindar un buen servicio de localización de hoteles ya que hoy día no existe un sistema similar, también conocer al final el aprendizaje del estudiante y el beneficio que se dará a los usuarios de esta App.

Dependiendo de los resultados y de la acogida que tenga este sistema se verá en la necesidad de mantener en el Distrito Metropolitano de Quito o implementar en las demás provincias llegando así a nivel nacional.

Capítulo I: Antecedentes

1.01 Contexto

El presente software (Mejoramiento de un sistema para rastreo de hoteles definidos en el D.M.Q. mediante una App para dispositivos móviles para la empresa STAFF VACATIONS CIA. LTDA.) está ligado a varios requerimientos de acuerdo a la necesidad del cliente. Este sistema es indispensable para la ubicación exacta de hoteles.

El principal propósito que pretende el presente proyecto es tener en cuenta todos los hoteles del D.M.Q. y así poder controlar y/o ubicar eficientemente, dando seguridad al usuario al momento de ubicar un hotel a su beneficio, ya que uno de los principales problemas es la falta de conocimiento de todos los hoteles que existen.

El software tiene como objetivo una correcta ubicación de hoteles por parte de los usuarios, de aquí la importancia del presente sistema de geo localización. Este permitirá a la empresa STAFF VACATIONS brindar un excelente servicio de localización ya que hoy en día no existe, también conocer al final, el aprendizaje del estudiante y el beneficio que se dará a los usuarios de esta App.

Se ha planteado efectuar o poner en marcha este sistema en vista de la falta de un sistema de ubicación, ya que hoy solo se cuenta con sistemas generales, es decir, de todos los lugares como es el google maps y demás sistemas, pero no un sistema que se enfoque precisamente a un solo lugar o cosa, en este caso los hoteles del D.M.Q. y así se brindará un excelente servicio a los usuarios especialmente a los turistas que llegan a la ciudad.

Dependiendo de los resultados y de las promociones que brinde este sistema se verá la necesidad de mantener la App solo en el D.M.Q o implementarlo en toda la provincia y a su vez a nivel nacional para el beneficio de más usuarios.

1.02 Justificación

El uso de varias herramientas inadecuadas de parte del turista ha generado la creatividad del estudiante al desarrollar una App de rastreo de hoteles definidos o asociados a la empresa STAFF VACATIONS CIA. LTDA., que deseen hacerse conocer en el medio turístico y promocionar sus beneficios, o también el desarrollo de alguna aplicación, llena de motivación, ya que permite investigar e innovar nuevas formas del uso de la programación, con diferentes herramientas de ayuda el cual sirve como motor de arranque para la creación del mismo. El desarrollo de esta App dará lugar a métodos de investigación, horas de trabajo y el uso de herramientas asociadas a la programación para así llegar al fin determinado.

Además nos puede ayudar proyectos ya realizados anteriormente como apoyo para facilitar el desarrollo del sistema utilizando herramientas adecuadas de programación, para llegar a obtener un excelente sistema de geo localización, para ayudar a los usuarios a ubicar fácilmente los hoteles de su elección evitando un gasto de impresiones en mapas de ubicación, ya que este sistema no solo desplegara un mapa, sino, nos dará a conocer de cómo llegar al lugar desde cualquier punto que nos encontremos.

1.03 Definición de problema

Debido al inadecuado proceso de ubicación de hoteles el gerente y presidente de la empresa STAFF VACATIONS CIA.LTDA necesitan la ayuda de una App de información capaz de rastrear correctamente los hoteles del D.M.Q.

Ya que en la actualidad es muy importante que el turista cuente con una herramienta de información veraz y oportuna con respecto al rastreo de hoteles y para ello se debe contar con una App que dote de dicha facilidad, esta permitirá un rastreo exacto y reducirá el tiempo de búsqueda.

Además los usuarios de dispositivos móviles no cuentan con esta herramienta tan efectiva que ayudara de gran manera a reducir el tiempo y dinero en busca de un hotel a gusto de cada usuario.

Tabla 1. *Matriz de fuerzas T*

Mala información de la ubicación de los hoteles en el D.M.Q.	Inadecuado Proceso de ubicación de hoteles en el D.M.Q.				Mejorar la ubicación de hoteles en el D.M.Q.
	I	PC	I	PC	
Fuerzas Impulsadoras					Fuerzas Bloqueadoras
Ubicación confiable	4	5	5	3	Ubicación no confiable
Ubicación adecuada de los hoteles del D.M.Q.	3	4	4	4	Desconocimiento de la ubicación de los hoteles del D.M.Q.
Mejoramiento del tiempo de llegada al conocer la ruta exacta de los hoteles del D.M.Q.	4	5	4	3	Retraso en el tiempo de llegada por el desconocimiento de la ruta de los hoteles del D.M.Q.
	-	+	+	-	

Tabla 2. *Escala valorativa*

Escala de valorización	
1 Muy bajo	
2 Bajo	I: intensidad
3 Medio	PC: potencial de cambio
4 Medio Alto	
5 Alto	

Capítulo II: Análisis de Involucrados

2.01. Requerimientos

2.01.1. Descripción del sistemas actual:

Considerando la importancia para los turistas el proceso de ubicación de los hoteles, que requiere de varias horas para su localización en el D.M.Q., normalmente los turistas hacen este proceso mediante mapas, internet o servicio de taxi, incomodando y quitando tiempo que se puede invertir en otras actividades, por falta de información y datos de ubicación.

Analizando los factores señalados y evaluando cada uno de ellos, se está desarrollando este sistema, el cual ayudará el avance que puede tener el medio turístico en la ciudad.

2.01.2. Visión y alcance

Dentro de la empresa STAFF VACATIONS CIA. LTDA., uno de los objetivos principales es la generación de software que facilite el trabajo a las personas y por ello este sistema es favorable para el cumplimiento de sus metas, ya sea a corto, mediano o largo plazo tomando en cuenta los beneficios que se puede obtener.

La empresa tiene la visión de liderar el mercado mediante el desarrollo de software y así dando mayores beneficios para los socios. Dar facilidad de acceso al mayor número de clientes con la finalidad de promover el turismo inicialmente en Quito y luego el resto del país.

Abarcar establecimientos que se encuentren inmersos dentro de la diversión y necesidades vacacionales de nuestros socios como son hoteles, hosterías, hostales, spa, restaurantes, parques de diversiones, bares, discotecas, renta de autos, cines, etc.

El alcance de este proyecto se enfoca en la automatización de los procesos de búsqueda de los hoteles que existen en Quito, y por ende ayuda mucho a las personas e instituciones involucradas en este sistema; se detalla a continuación;

Usuarios:

Aquí los usuarios que deseen se contactaran con la empresa y firmaran un contrato para obtener el servicio que deseen o en nuestro caso la App.

Hoteles e instituciones:

Mediante la aplicación, aquí se ve el crecimiento de los clientes y la acogida que da a cada una de ellas.

Empresa:

De igual manera se verá el crecimiento de los usuarios en la empresa y los beneficios que se brindara para satisfacer sus necesidades.

2.01.3. Entrevista

Tabla 3. Entrevista

DISEÑO ENTREVISTA		
Identificador		
Preguntas	Objetivos	Análisis posterior
¿Cuál es la problemática por el que se pretende realizar este sistema?	Conocer el lugar específico donde está el hotel con una dirección exacta para así tener un mayor beneficio en el sistema.	Se desea conocer la dirección exacta del hotel mediante un aplicativo para así obtener información eficaz, real y en el menor tiempo posible.
¿Cuál será la estrategia que se utilizara para saber la dirección exacta de los hoteles?	Llegar con exactitud a la dirección del hotel que se desea hospedar.	Investigar sobre las coordenadas satelitales, la cual es la única forma que se lograra direccionar correctamente en esta App la dirección del hotel
¿Cuántos y que usuarios tendrán acceso a este sistema?	Conocer el tipo y la cantidad de usuarios que van a obtener la App	Investigar o hacer una encuesta de la necesidad del sistema para el público si es beneficioso y las ventajas que prestara.

2.01.4. Matriz de requerimientos

Tabla 4. *Matriz de Requerimientos*

Matriz de Requerimientos						
Identificador	Descripción	Fuente	Prioridad	Tipo	Estado	Usuarios Involucrados
<u>REQUERIMIENTOS FUNCIONALES</u>						
RF001	Contar con un mapa de ubicación de los lugares que se encuentran los hoteles	Presidente de la empresa STAFF VACATIONS CIA.LTDA.	Alta	App	Revisión	Desarrollador
RF002	Investigar sobre las coordenadas satelitales para una correcta ubicación en el mapa.	Presidente de la empresa STAFF VACATIONS CIA.LTDA.	Alta	App	Revisión	Desarrollador
<u>REQUERIMIENTOS NO FUNCIONALES</u>						
NRF001	Conocer el tipo y la cantidad de usuarios que van a tener la App	Presidente de la empresa STAFF VACATIONS CIA.LTDA.	Media	App	Revisión	Desarrollador

2.01.5. Descripción detallada

Tabla 5. Descripción de Requerimiento RF001

Descripción del requerimiento		Estado	
Creado por:	Xavier Yanacallo	Actualizado por:	Xavier Yanacallo
Fecha Creación	03/07/2015	Fecha Actualización	08/07/2015
Identificador		RF001	
Estado de Requerimiento	Critico	Tipo de Requerimiento	Funcional
Datos de Entrada	Visualización de la ubicación de hoteles		
Descripción	Contar con un adecuado diseño del mapa de ubicación. Correcta utilización de la App. Es necesario contar con la App para poderse ubicar.		
Datos de Salida	Correcta ubicación de los hoteles del D.M.Q.		
Resultados Esperados	Correcta ubicación.		
Origen	Presidente de la empresa STAFF VACATIONS CIA.LTDA		
Dirigido a	Empresa STAFF VACATIONS CIA.LTDA		
Prioridad	Alto		
Requerimientos Asociados	RF001		
ESPECIFICACIÓN			
Precondiciones	Diseño u obtención del mapa		
Pos condiciones	Ubicación correcta de los hoteles en el mapa		
Criterios de Aceptación	Conocer la ubicación de los Hoteles		

Tabla 6. Descripción de Requerimiento RF002

Descripción del requerimiento		Estado	
Creado por:	Xavier Yanacallo	Actualizado por:	Xavier Yanacallo
Fecha Creación	03/07/2015	Fecha Actualización	08/07/2015
Identificador		RF002	
Estado de Requerimiento	Critico	Tipo de Requerimiento	Funcional
Datos de Entrada	Datos precisos de donde esta y a dónde quiere llegar		
Descripción	Basarse en coordenada satelitales para el desarrollo de la App. Correcta utilización de las coordenadas satelitales. Utilización correcta las coordenadas en el mapa.		
Datos de Salida	Obtención de la ubicación correcta		
Resultados Esperados	Correcta ubicación.		
Origen	Presidente de la empresa STAFF VACATIONS CIA.LTDA		
Dirigido a	Empresa STAFF VACATIONS CIA.LTDA		
Prioridad	Alto		
Requerimientos Asociados	RF002		
ESPECIFICACIÓN			
Precondiciones	Investigar cómo utilizar las coordenadas satelitales		
Pos condiciones	Ubicar las coordenadas en la App		
Criterios de Aceptación	Conocer la ubicación de los Hoteles		

Tabla 7. Descripción de Requerimiento NRF001

Descripción del requerimiento		Estado	
Creado por:	Xavier Yanacallo	Actualizado por:	Xavier Yanacallo
Fecha Creación	03/07/2015	Fecha Actualización	08/07/2015
Identificador		NRF001	
Estado de Requerimiento	Critico	Tipo de Requerimiento	No Funcional
Datos de Entrada		Desarrollar la App	
Descripción	Acogida que tendrá esta App en los turistas que visitan el D.M.Q. Necesidad del usuario de esta App. Una buena utilidad que le dará el usuario.		
Datos de Salida		Satisfacción del usuario por la App	
Resultados Esperados		Correcta ubicación.	
Origen	Presidente de la empresa STAFF VACATIONS CIA.LTDA		
Dirigido a	Empresa STAFF VACATIONS CIA.LTDA		
Prioridad		Medio	
Requerimientos Asociados		NRF001	
ESPECIFICACIÓN			
Precondiciones		Desarrollar una correcta App	
Pos condiciones		Acogida de la App de parte de usuario	
Criterios de Aceptación		Conocer la ubicación de los Hoteles	

2.02. Mapeo de Involucrados

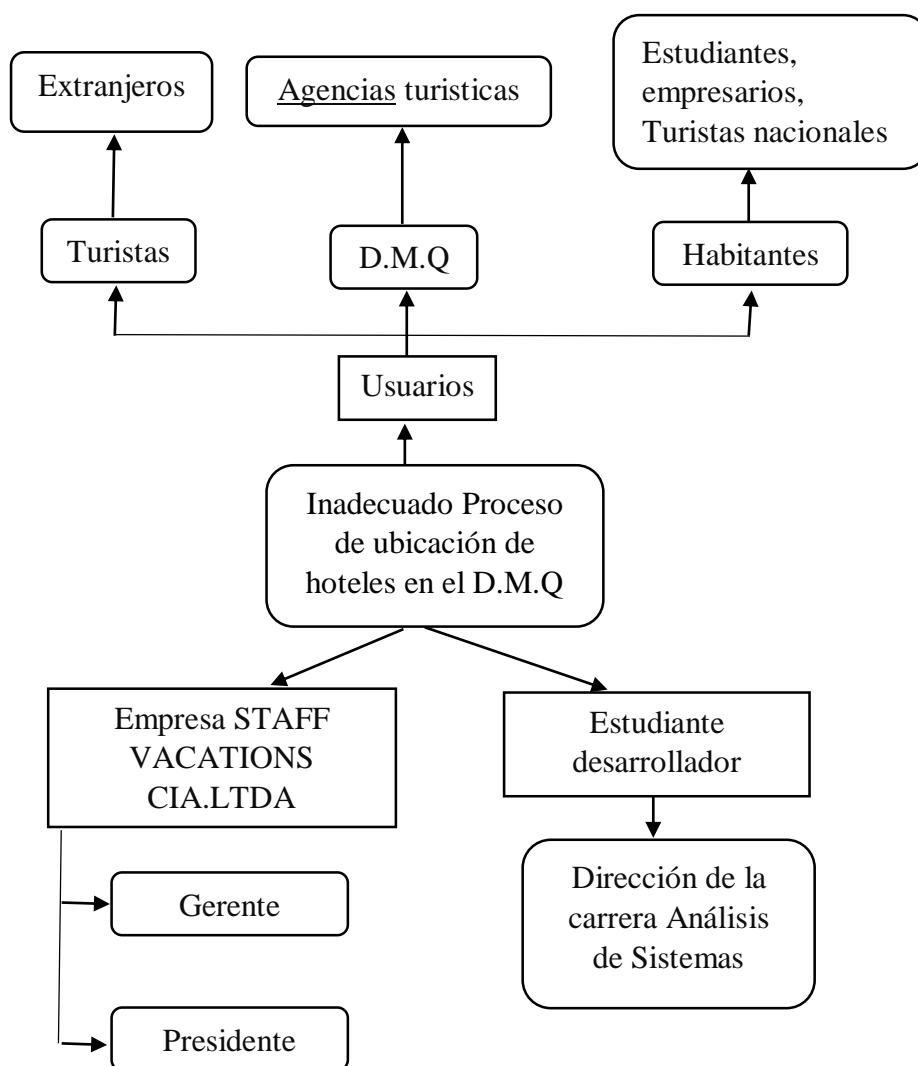


Figura 1. Mapeo de involucrados

2.03. Matriz de Involucrados

Tabla 8. *Matriz de Involucrados*

Inadecuado Proceso de ubicación de hoteles en el D.M.Q					
Involucrados	Intereses en el Problema	Problemas Percibidos	Recursos y Mandatos	Interés en el Proyecto	Conflictos y cooperación
Usuarios	<ul style="list-style-type: none"> • Interés en conocer la ubicación correcta de hoteles. • Ahorro de tiempo en la búsqueda de un hotel. • Reducir el costo de impresiones de guías. 	Desconocimiento y pérdida de tiempo en la ubicación de hoteles en el D.M.Q.	Buena acogida de la App	Conocer la ubicación exacta de los hoteles, para así reducir el tiempo de búsqueda en el D.M.Q.	Existe desconocimiento de los turistas de la ubicación geográfica de los hoteles. Ahorro de tiempo en dirigirse al hotel más cercano.
Empresa STAFF VACATIONNS CIA.LTDA	<ul style="list-style-type: none"> • Ayudar al usuario a ubicarse de manera eficaz los hoteles. • Dar alcance a todo público información y variedad de hoteles. 	Desconocimiento de parte de los usuarios y/o turistas de la variedad de hoteles que existe en el D.M.Q.	Facilidad de ubicación de hoteles	Ayudar a los usuarios a la ubicación de hoteles mediante una App para dispositivos móviles.	
Estudiante	<ul style="list-style-type: none"> • Tienen interés en alcanzar y mejorar la competencia • Ejercen interés en mejorar la calidad de servicios para los turistas. 	Desconocimiento informativo de la ubicación geográfica y distancia de los hoteles del D.M.Q	Capacitación de la App para una buena utilización	Conocer a fondo la necesidad de los usuarios para la ubicación de hoteles.	

Capítulo III: Problemas y Objetivos

3.01. Árbol de Problemas

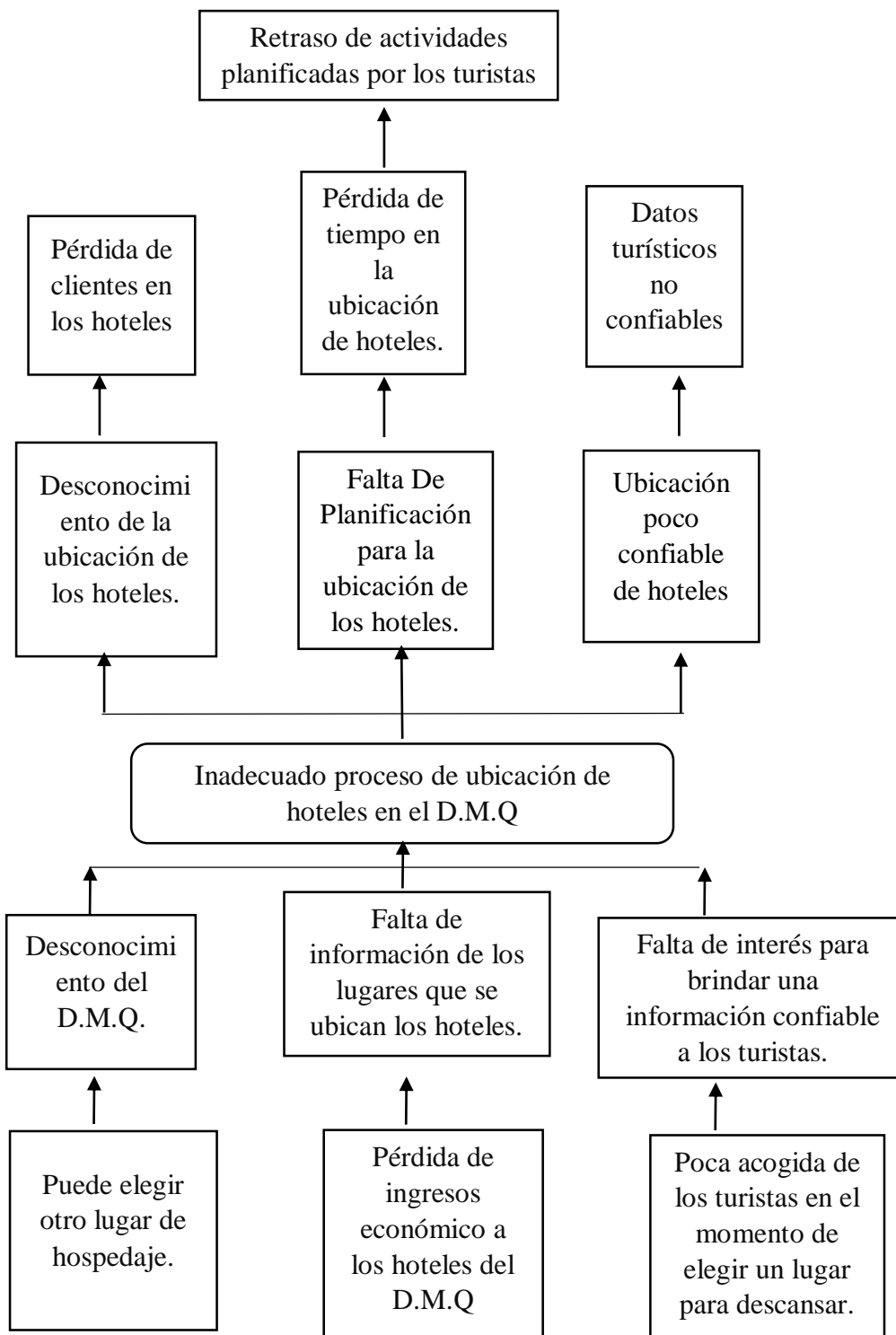


Figura 2. Árbol de problemas

3.02. Árbol de Objetivos

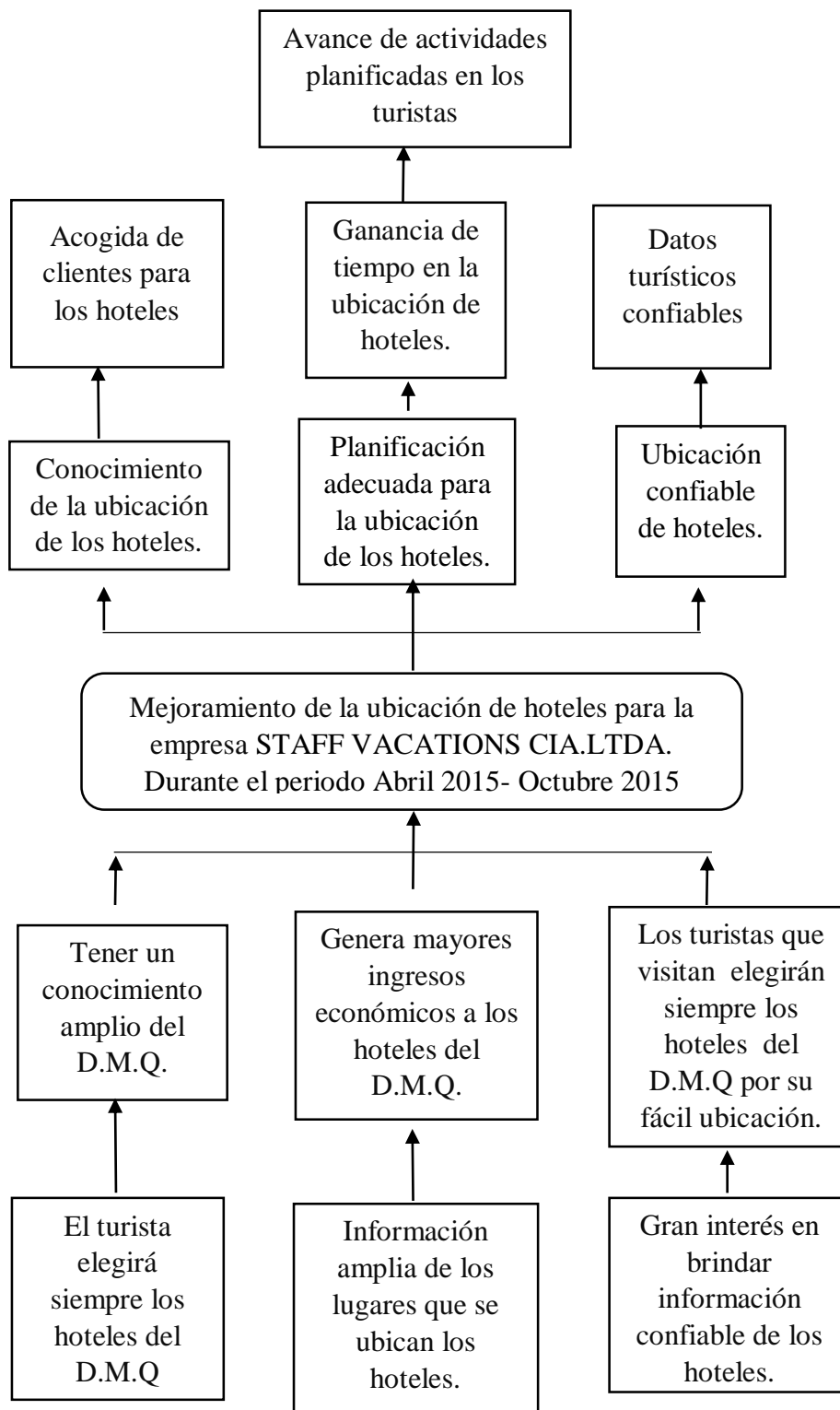


Figura 3. Árbol de objetivos

3.03. Diagramas de caso de uso

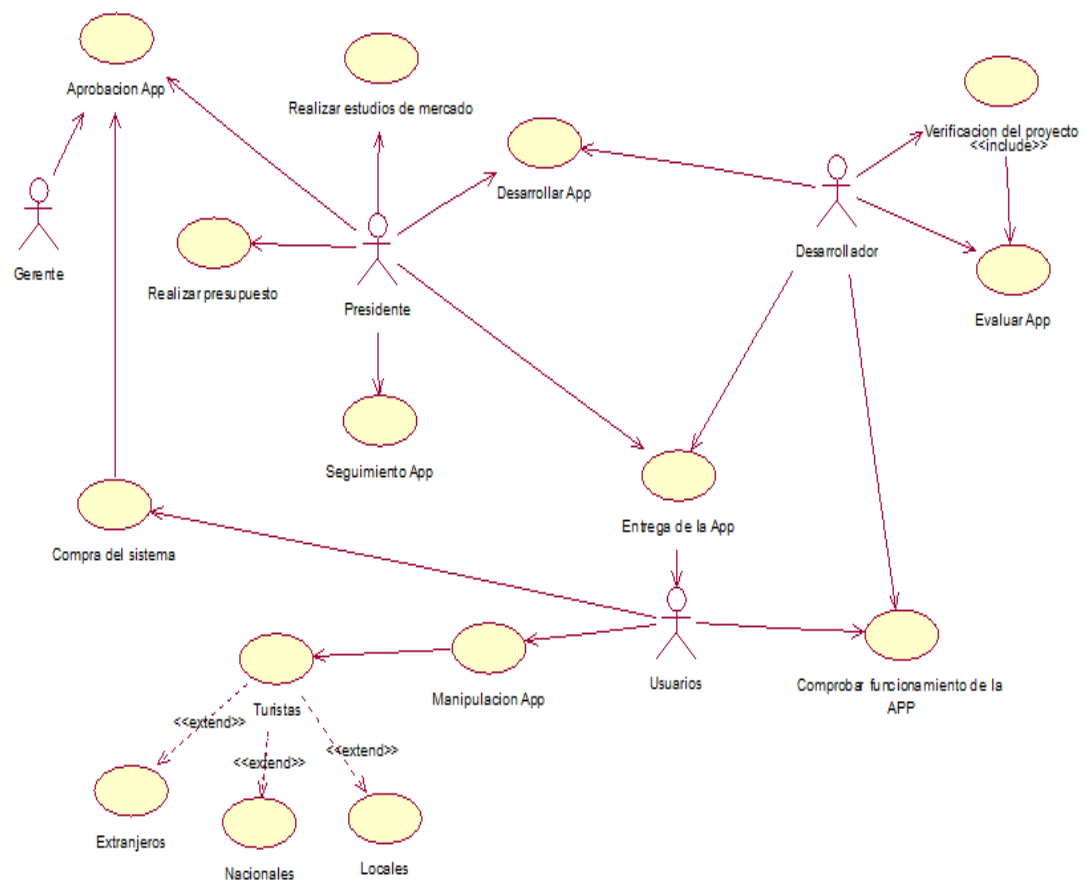


Figura 4. Diagrama de casos de uso

3.04. Especificaciones de caso de uso

Tabla 9. *Especificaciones de caso de Uso*

Caso de Uso	Buscar Ubicación de Hotel
Identificador	UC001
CURSO TÍPICO DE EVENTOS	
Usuario	Sistema
1.- El usuario se activa cuando se haya registrado en la App.	El usuario se puede ingresar al sistema siempre y cuando ya haya registrado sus datos, después de la descarga y solo basta con ingresar la contraseña para que los datos del usuario se restablezcan.
2.- El usuario podrá realizar la búsqueda ingresando:	Una vez ingresado los parámetros de búsqueda en el sistema, ejecuta:
a) El nombre del Hotel.	a) Si es por nombre del hotel filtra en la base de datos todas aquellas que coincidan con el nombre buscado.
b) Servicios del Hotel.	b) Si es por el servicio que ofrece filtra en la base de datos y saca los servicios de cada hotel.
c) Dirección del hotel.	c) Si es por dirección filtra en la base de datos y saca el hotel que se encuentra en la dirección buscada.
d) Mapa de ubicación.	d) Si es por ubicación el usuario ira directamente al mapa de la aplicación y realizará la búsqueda.
CURSOS ALTERNATIVOS	
Si el usuario no puede encontrar un hotel que desee, debe registrarse y/o comunicarse con la empresa ya que es la empresa la que facilita esta App, al cliente que desea afiliarse a este servicio.	

3.05. Caso de uso de realización

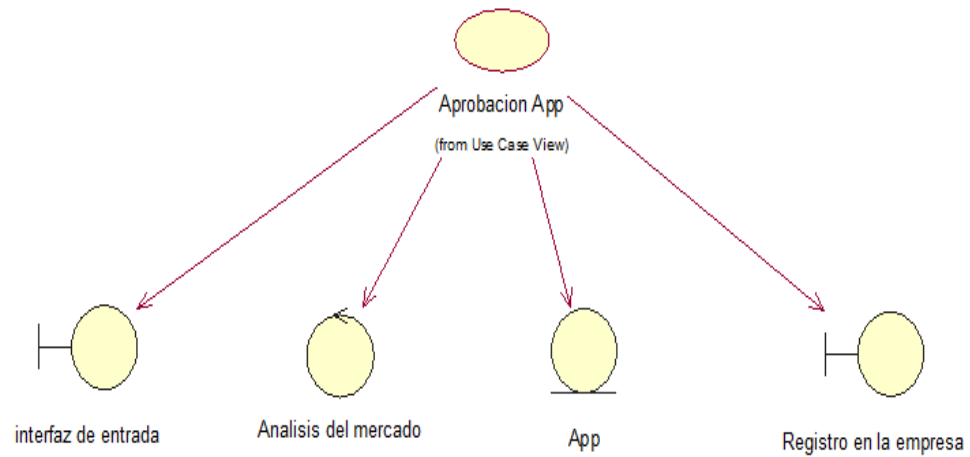


Figura 5. Caso de uso de realización (Aprobación App)

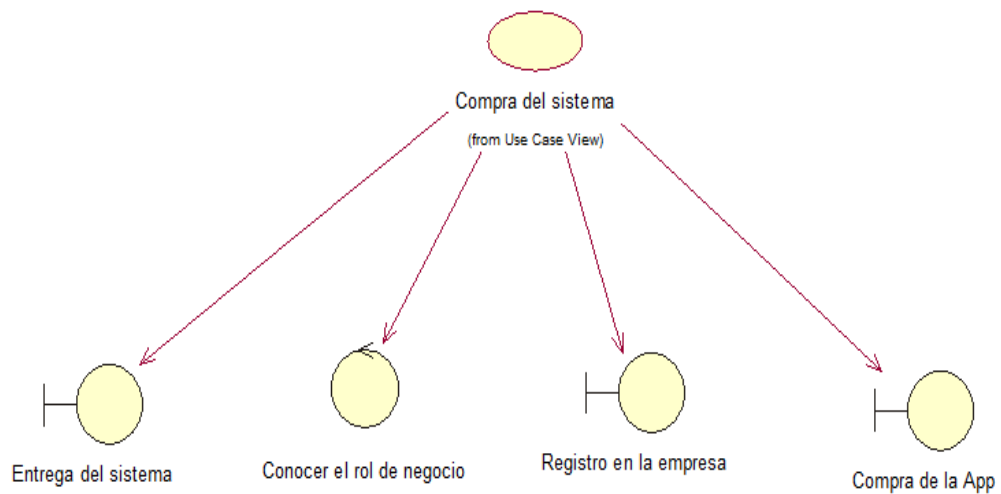


Figura 6. Caso de uso de realización (Compra del sistema)

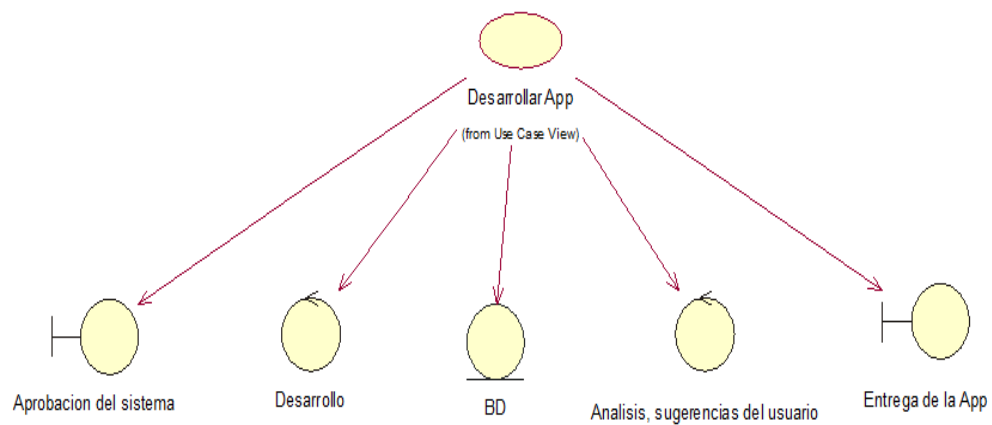


Figura 7. Caso de uso de realización (Desarrollo App)

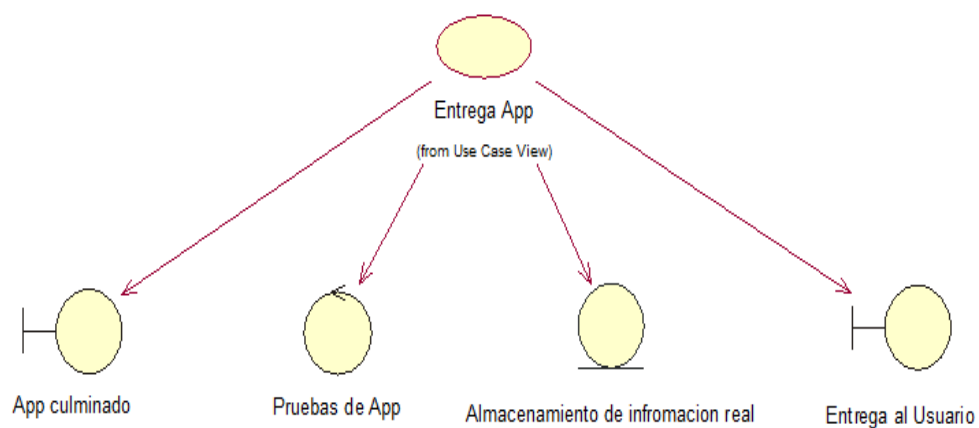


Figura 8. Caso de uso de realización (Entrega App)

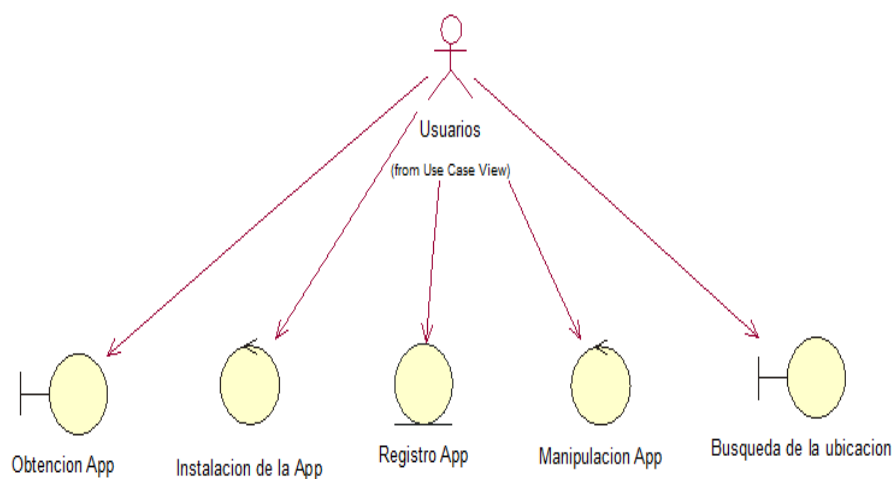


Figura 9. Caso de uso de realización (Usuario)

3.06. Diagramas de secuencia del sistema

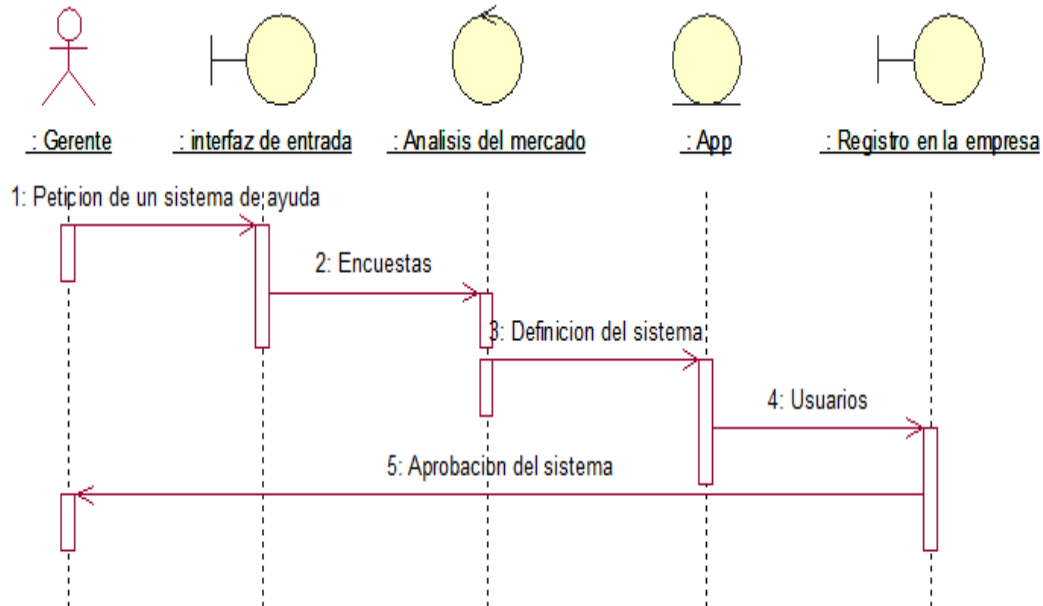


Figura 10. Diagrama de secuencia (Aprobación App)

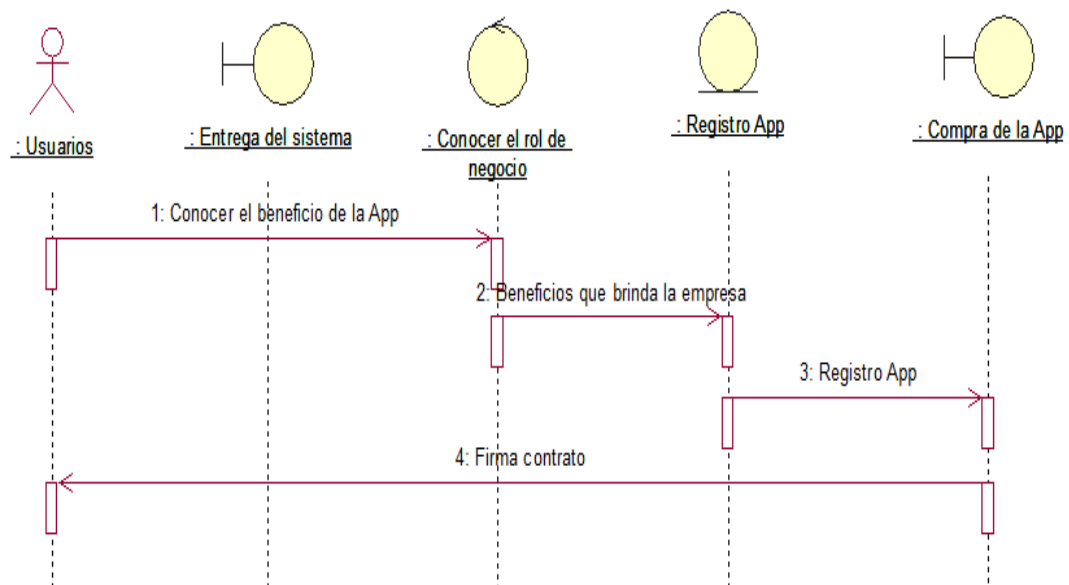


Figura 11. Diagrama de secuencia (Compra App)

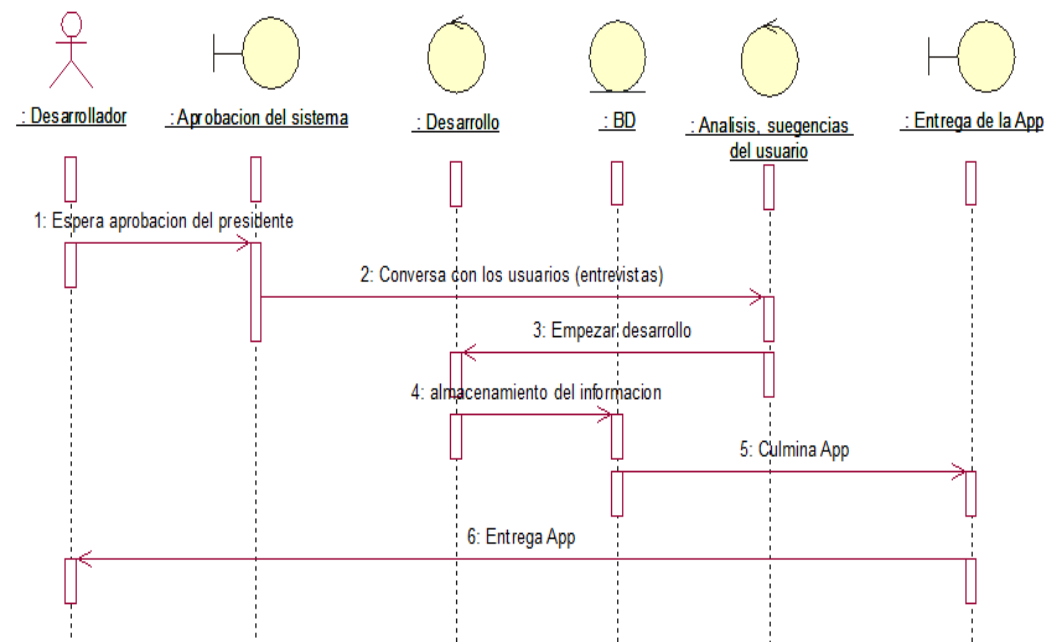


Figura 12. Diagrama de secuencia (Desarrollo App)

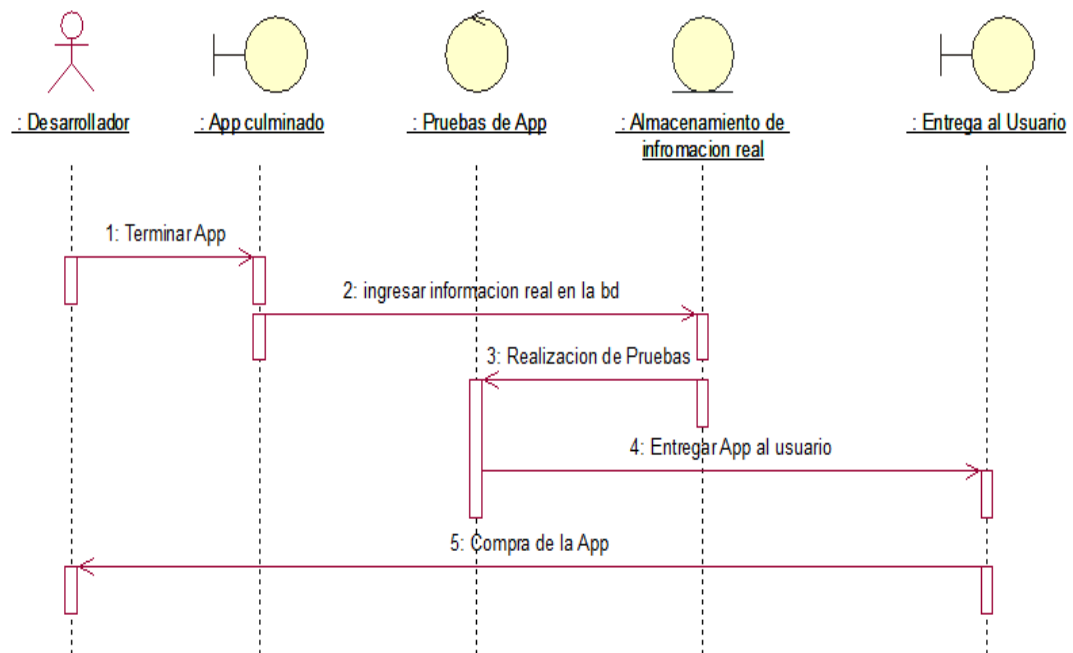


Figura 13. Diagrama de secuencia (Entrega App)

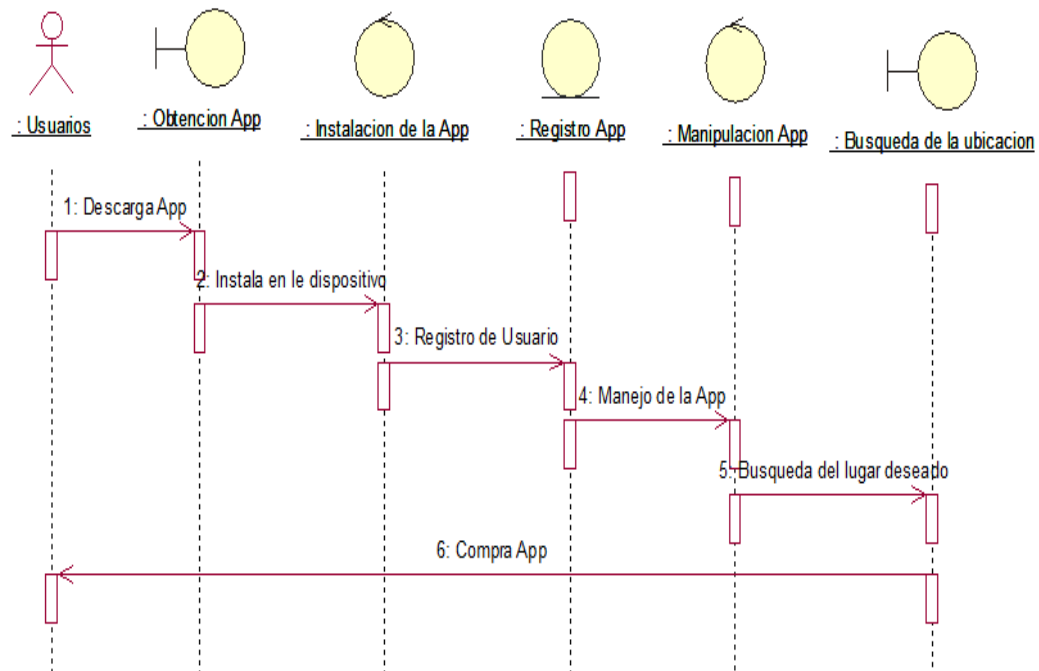


Figura 14. Diagrama de secuencia (Usuario)

Capítulo IV: Análisis de Alternativas

4.01. Matriz de Análisis de Alternativas

En esta matriz se detalla las estrategias, la necesidad, factibilidad, duración del proyecto y a la vez el respectivo puntaje de cada una de ellas. Véase *anexo A.01*.

4.02. Matriz de Impactos de Objetivos

En esta matriz se analizara el beneficio que se obtendrá con este sistema lo que se quiere lograr con su diseño e implementación y a la vez el impacto que tendrá en el medio ambiente. Véase *anexo A.02*.

4.03. Estándares para el Diseño de Clases

Clase: La clase es la unidad básica que contiene toda la información de un objeto que comparten los mismos atributos, métodos, relaciones y semántica. El nombre de las clases se indica la primera letra con mayúscula si existen varias palabras.

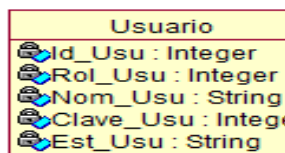


Figura 15. Clase

Atributos: Atributos o variables de instancia que caracterizan a la clase los mismos pueden ser private, protected o public.

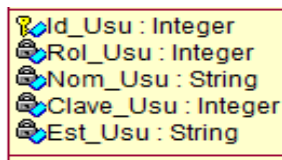


Figura 16. Atributos

Métodos: Los métodos u operaciones, son aquellos con los cuales interactúa el objeto con su entorno dependiendo de la visibilidad: private, protected o public. Especificando la operación por ejemplo: "Insertar (), Eliminar (), Modificar ()".

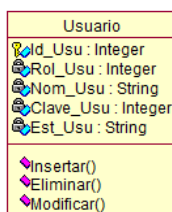


Figura 17. Métodos

Primary Key: Toda tabla deberá tener una llave principal la cual indica que el atributo no será accesible desde fuera de la clase, pero si podrá ser manipulado por métodos de la clase y de sus subclases.

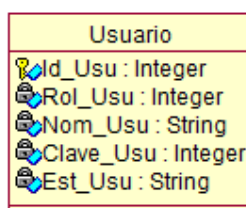


Figura 18. Primary Key (código principal)

Foreing Key: Representamos el foreing key antecediendo el nombre del atributo con el sufijo "FK", seguido de un "_" y con el respectivo nombre. Ej.: FK_Id_Hotel.

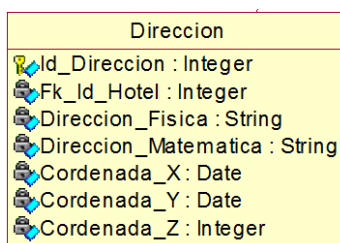


Figura 19. Foreing Key (código heredado)

Asociación: Permite asociar objetos que colaboran entre sí. Cabe destacar que no es una relación fuerte, es decir, el tiempo de vida de un objeto no depende del otro.



Figura 20. Asociación de objetos

Generalización: Es una relación de especificación o generalización en lo que los objetos del elemento especificado (hijo) puede sustituir a los elementos generales (padre).

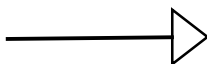


Figura 21. Generalización de objetos

Relación de uno a muchos: Sirve para relacionar una tabla con otra y así obtener la herencia entre tablas, evitando la duplicidad de la información.



Figura 22. Relación

4.04. Diagrama de Clases

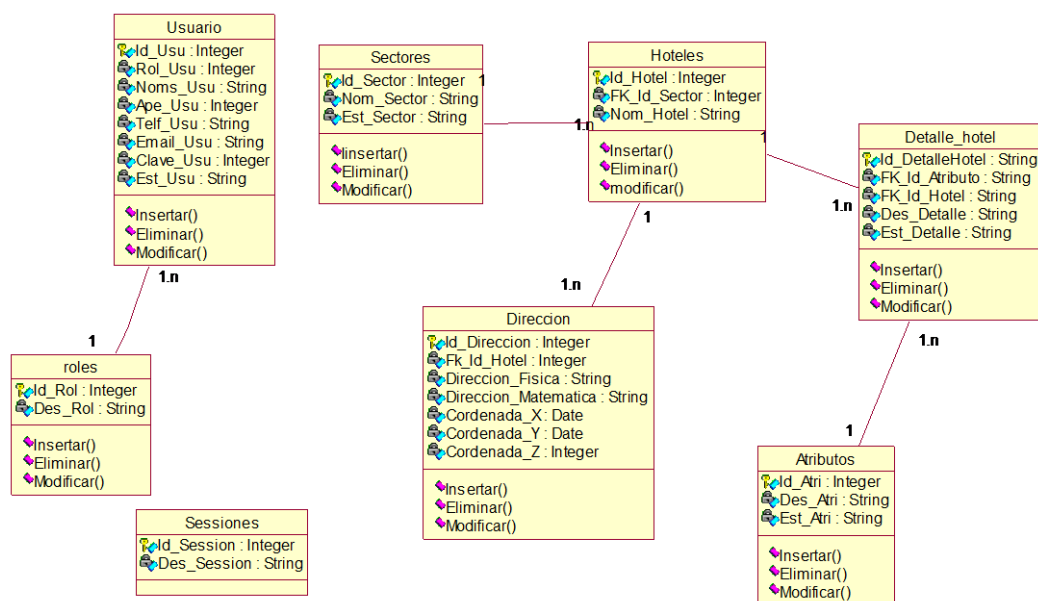


Figura 23. Diagrama de clases

4.05. Modelo físico

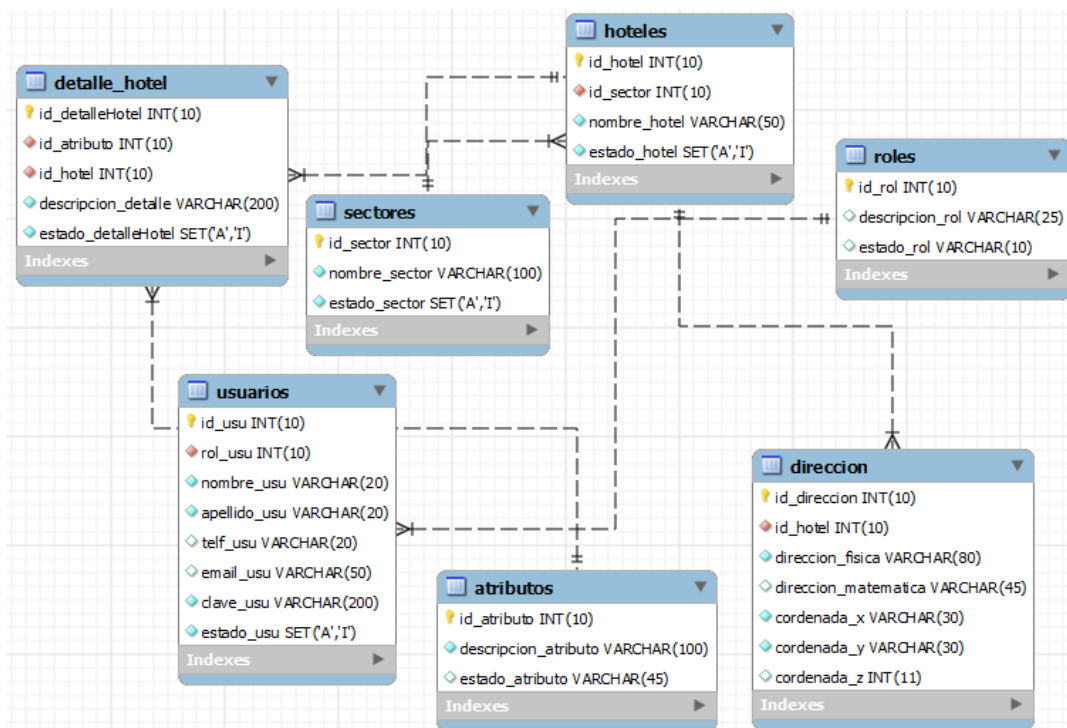


Figura 24. Modelo Físico

4.06. Diagrama de Componentes

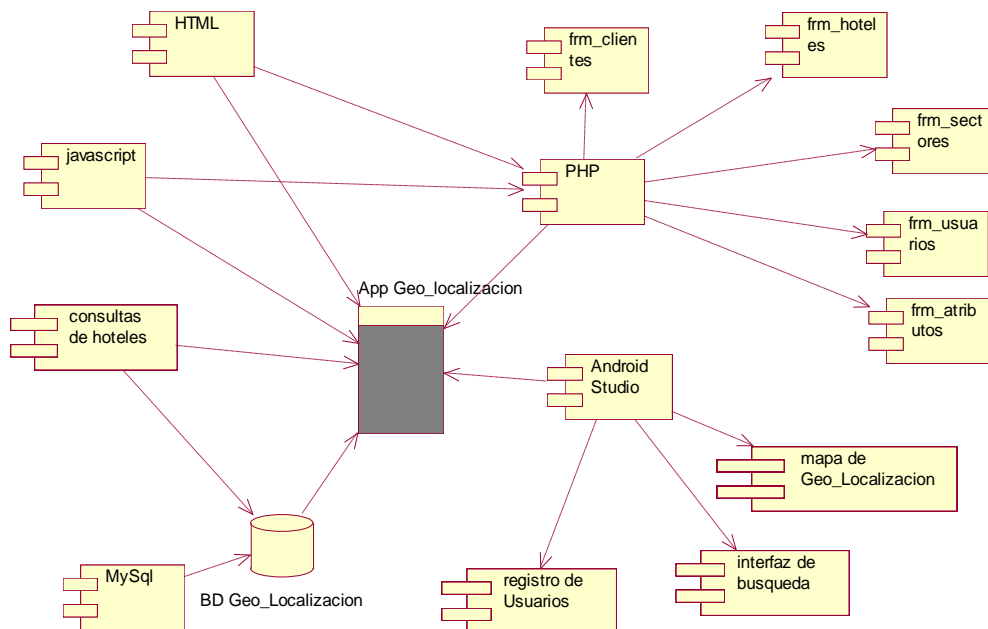


Figura 25. Diagrama de Componentes

4.07. Diagrama de Estrategias

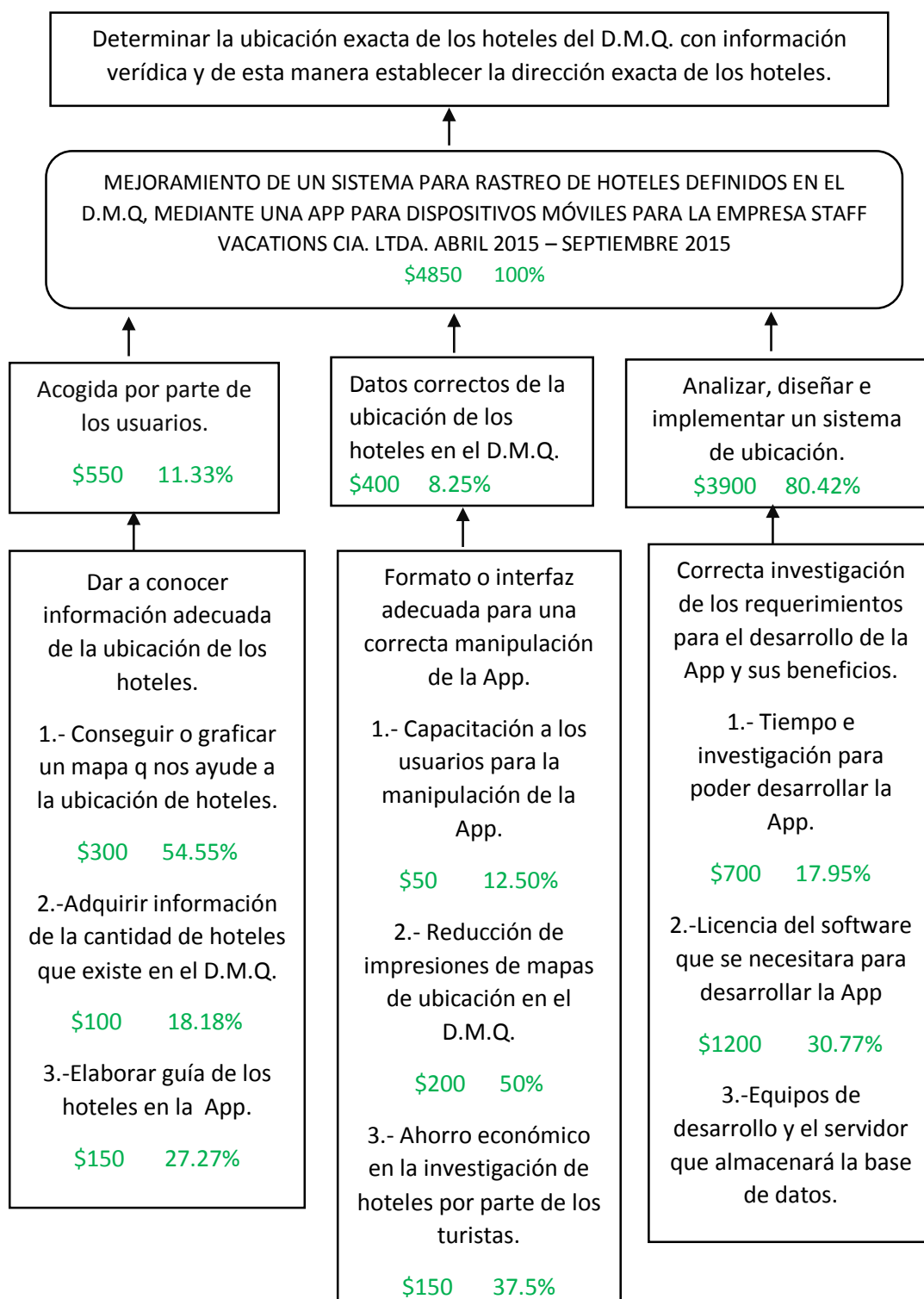


Figura 26. Diagrama de Estrategias

4.08. Matriz de Marco Lógico.

Tabla 10. *Matriz de Marco Lógico*

Resumen de objetivos	Indicadores	Medios de Verificación	Supuestos
Pérdida de clientes en los hoteles y a la vez Pérdida de tiempo en la ubicación por datos turísticos no confiables.	Avance de actividades planificadas en los turistas	Facturas de los gastos generados por la compra o impresión de mapas de ubicación.	El público tenga acogida de la App, para ellos se deberá hacer una encuesta, y así determinar el beneficio de la App.
Inadecuado proceso de ubicación de hoteles.	Desconocimiento de la ubicación de los hoteles. Falta De Planificación para la ubicación de los hoteles.		También se debe determinar si es necesario la utilización de
Ganancia de ingresos económico a los hoteles del D.M.Q	Acogida por parte de los usuarios. \$550 11.33% Datos correctos de la ubicación de los hoteles en el D.M.Q. \$400 8.25% Analizar, diseñar e implementar un sistema de ubicación. \$3900 80.42% TOTAL Mejoramiento de un sistema para rastreo de hoteles definidos en el D.M.Q., mediante una App para dispositivos móviles para la empresa STAFF VACATIONS CIA. LTDA. ABRIL 2015 – OCTUBRE 2015 \$4850 100%	Aumento en la economía de la empresa, en sus cuentas bancarias.	la App para optimizar el tiempo de ubicación de hospedaje y crecimiento económico de los hoteles del D.M.Q.
Los turistas elegirán siempre los hoteles del D.M.Q por su fácil ubicación.	Falta de información de los lugares donde se ubican los hoteles. Poco conocimiento del D.M.Q.	Ubicación correcta de los hoteles mediante un informe o listas detalladas.	

4.09. Vistas arquitectónicas

4.09.01. Vista lógica

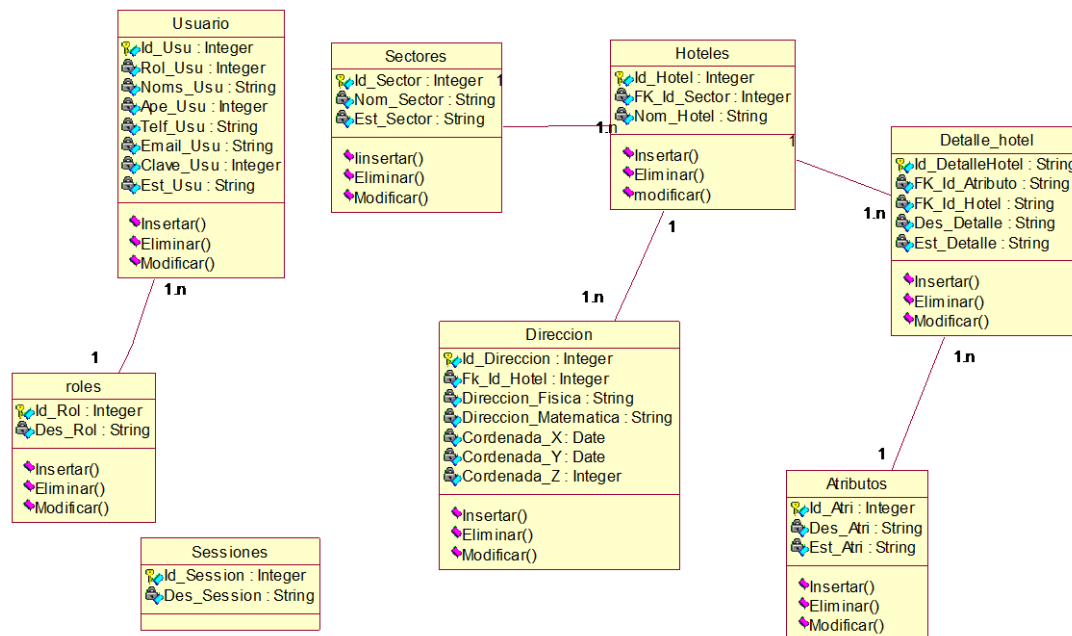


Figura 27. Vista lógica

4.09.02. Vista física

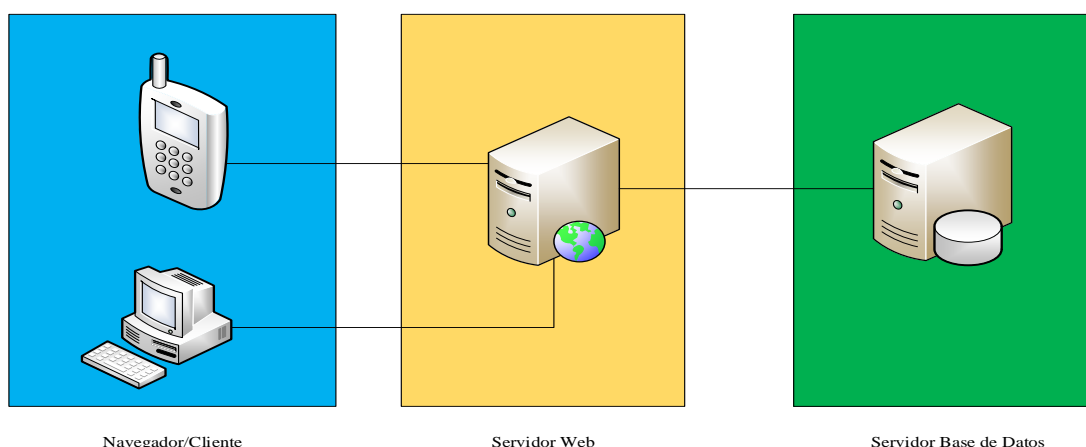


Figura 28. Vista física

4.09.03. Vista de desarrollo

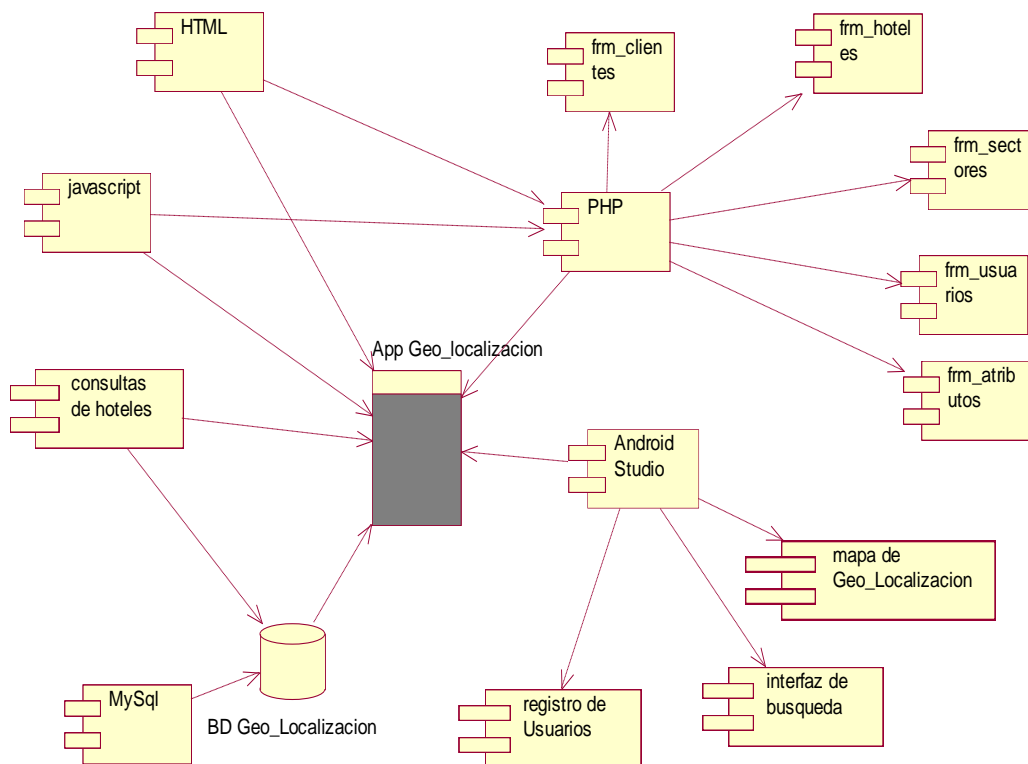


Figura 29. Vista de desarrollo

4.09.04. Vista de procesos

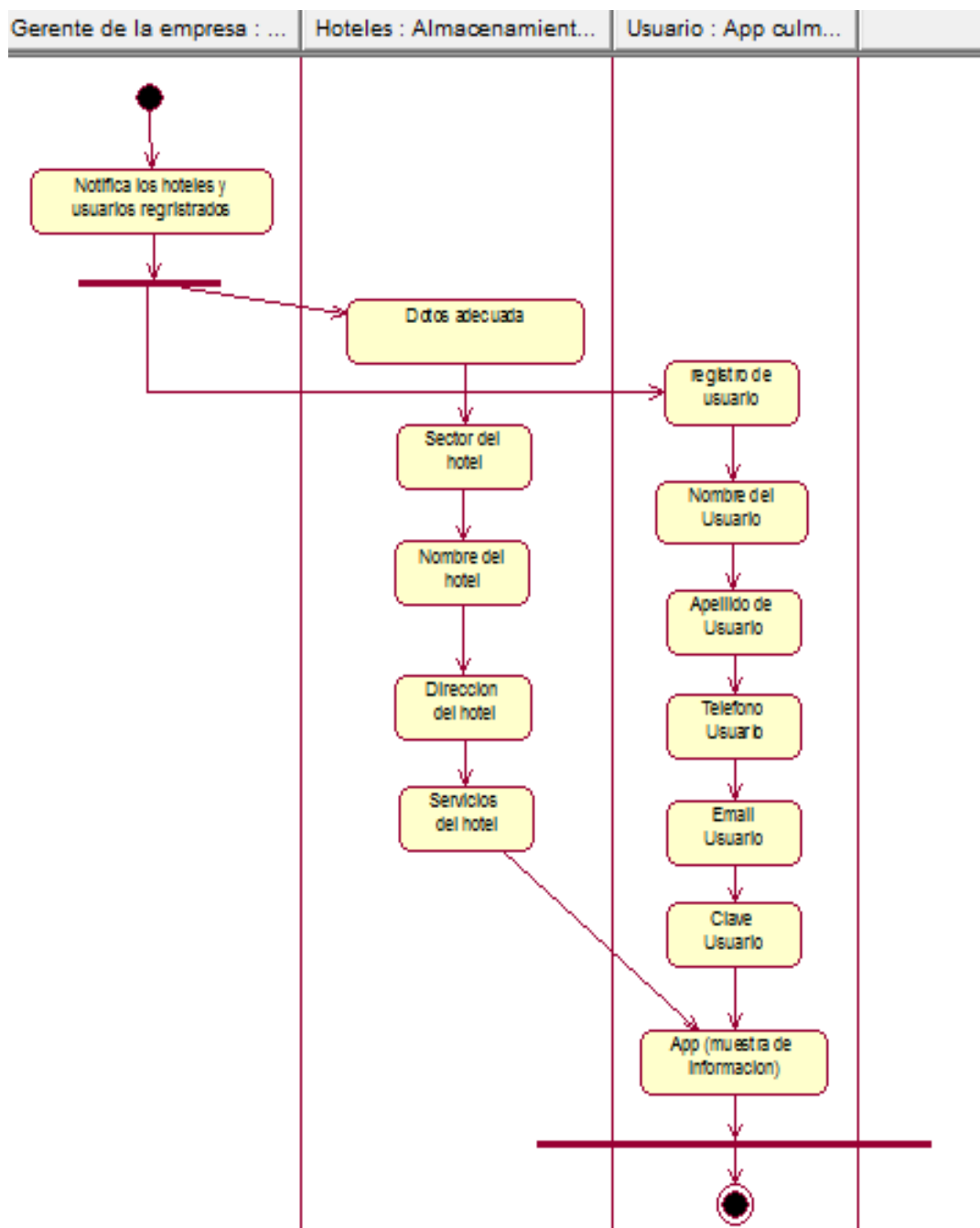


Figura 30. Vista de procesos

Capítulo V: Propuesta

5.01. Especificación de estándares de programación

En este punto se verá la forma de normalizar la programación, su estructura y organización, para el momento de trabajar en un proyecto, cualquier persona involucrada en el mismo puedan comprender el código. Véase Anexo A.03.

Podemos ver la forma o método de obtener los datos para la App y la utilización del try catch que detectará cualquier excepción. Por lo general, se recomienda, cuando se esté programando, detectar un tipo de excepción específico en lugar de utilizar la instrucción

```
public void obtener_datosDB(){  
    Thread nt = run() -> {  
        final String Resultado=sendByPost("http://katonproyect.tk/geo_localizacion/hoteles_remoto/consultar_sectores");  
        try {  
            runOnUiThread() -> {  
                cargarSectores(obtDatosJSON(Resultado));  
            };  
        } catch (Exception e) {  
            mostrarMensaje("Error " + e);  
        }  
    };  
    nt.start();  
}
```

Figura 31. Conexión a la base de datos

5.02. Diseño de interfaces de usuario

Las siguientes figuras se muestran la interfaz que utilizara el administrador del sistema:

Tabla 11. Descripción del ingreso al sistema

Ítem	Representación	Descripción
A	imagen	Imagen usuario
B	image	Imagen contraseña
C	input type="text"	Usuario
D	input type="password"	Contraseña
E	Button	Inicia Sesión
F	lebel	Muestra hora actual



Figura 32. Ingreso al sistema

Tabla 12. *Detalle de usuarios*

Ítem	Representación	Descripción
A	Button	Ingreso nuevo usuario
B	Button	Modificar usuario
C	Button	Página anterior
D	Button	Borrar usuario
E	menu	Menú del sistema



Figura 33. Usuarios del sistema

Tabla 13. *Detalle de ingreso de usuario*

Ítem	Representación	Descripción
A	<option value=""	Tipo usuario
B	input type="text"	Nombre usuario
C	input type="password"	Clave
D	input type="password"	Confirmación clave
E	Button	Guardar usuario

Figura 34. Ingreso de nuevo usuario

Tabla 14. Detalle de sectores

Ítem	Representación	Descripción
A	Button	Nuevo sector
B	Button	Modificar sector
C	Button	Página anterior
D	Button	Borrar sector

Figura 35. Sectores de ubicación de los hoteles

Tabla 15. Detalle de ingreso de nuevo sector

Ítem	Representación	Descripción
A	input type="text"	Nombre del nuevo sector
B	Button	Guardar sector

Figura 36. Ingreso de nuevo sector

Tabla 16. Detalle de roles

Ítem	Representación	Descripción
A	Button	Borrar rol usuario
B	Button	Página anterior
C	Button	Nuevo Rol
D	Button	Modificar rol

Figura 37. Roles de usuarios

Tabla 17. Detalle de ingreso de nuevo rol

Ítem	Representación	Descripción
A	Button	Guardar nuevo rol
B	input type="text"	Nombre nuevo rol

Figura 38. Ingreso de nuevo rol

Tabla 18. Detalle de hoteles

Ítem	Representación	Descripción
A	Button	Nuevo hotel
B	Button	Modificar hotel
C	Button	Borrar hotel
D	Button	Página anterior

Figura 39. Detalle de hoteles

Tabla 19. Detalle de ingreso de nuevo hotel

Ítem	Representación	Descripción
A	<option value=""	Elegir sector del hotel
B	input type="text"	Coordenadas (longitud)
C	Button	Guardar
D	input type="text"	Nombre hotel
E	input type="text"	Dirección hotel
F	input type="text"	Coordenadas (latitud)
G	<script type="text/javascript"	Ubicación del hotel en el mapa

The screenshot shows a web application titled "Sistema de Geo Localizacion Hotelera" with a navigation bar containing links: Inicio, Atributos, Servicios, Hoteles, Roles, Sectores, Usuarios, and Salir. The main form is titled "NUEVO HOTEL" and contains the following elements:

- A:** A dropdown menu for "Sector" with the text "Elije..." and a downward arrow.
- B:** A text input field for "Logitud" containing the value "-78.4305382".
- C:** A "Guardar" button.
- D:** A text input field for "Nombre hotel".
- E:** A text input field for "Dirección".
- F:** A text input field for "Latitud" containing the value "-0.1865944".
- G:** A map showing the location of the hotel in Quito, Ecuador, with a red pin marker.

Figura 40. Ingreso de nuevo hotel

Tabla 20. Detalle de los hoteles

Ítem	Representación	Descripción
A	Button	Ver servicios del hotel
B	Button	Página anterior



Figura 41. Detalle de los servicios de hoteles

Tabla 21. Ingreso de nuevo servicio

Ítem	Representación	Descripción
A	<option value=""	Tipo de servicio
B	input type="text"	Descripción del servicio
C	Button	Registrar



Figura 42. Descripción de un servicio

Tabla 22. *Detalle de atributos*

Ítem	Representación	Descripción
A	Button	Nuevo atributo
B	Button	Modificar atributo
C	Button	Borrar atributo
D	Button	Página anterior



Figura 43. Atributos de un hotel

Las siguientes figuras nos indicaran la interfaz donde el usuario podrá hacer uso de la App:

Tabla 23. *Detalle de ingreso a la App*

Ítem	Representación	Descripción
A	TextView	Usuario
B	ImageView	Imagen hotel
C	EditText	Ingreso de usuario
D	TextView	Clave
E	EditText	Ingreso clave
F	Button	Ingresar
G	Button	Registro nuevo usuario

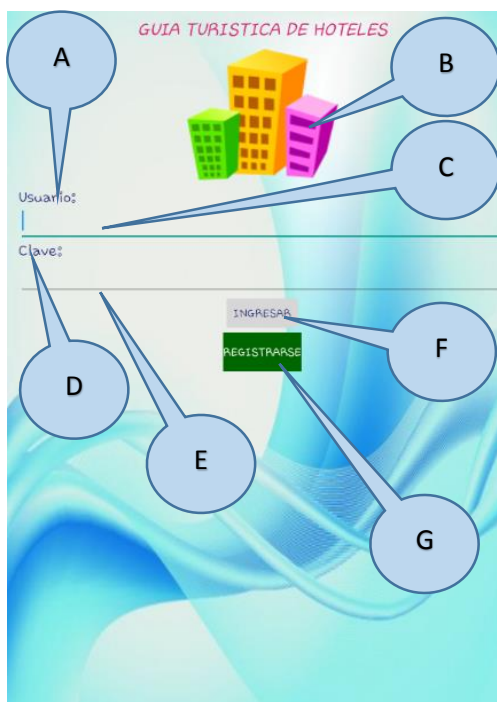


Figura 44. Ingreso a la App

Tabla 24. Registro de nuevo usuario en la App

Ítem	Representación	Descripción
A	TextView	Nombre
B	TextView	Apellido
C	TextView	Email
D	TextView	Teléfono
E	TextView	Contraseña
F	EditText	Nombre
G	EditText	Apellido
H	EditText	Email
I	EditText	Teléfono
J	EditText	Contraseña
K	Button	Registrar usuario

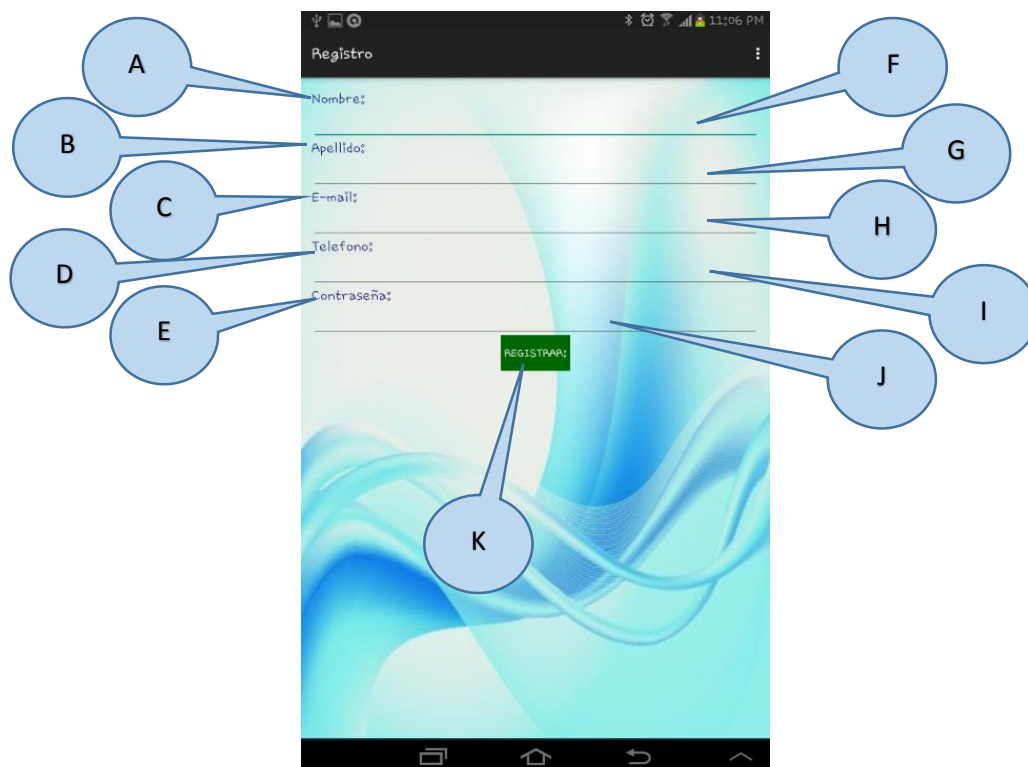


Figura 45. Registro de nuevo usuario

Tabla 25. *Detalle de los sectores*

Ítem	Representación	Descripción
A	TextView	Título sectores
B	ImageView	Imagen sectores
C	ListView	Sectores

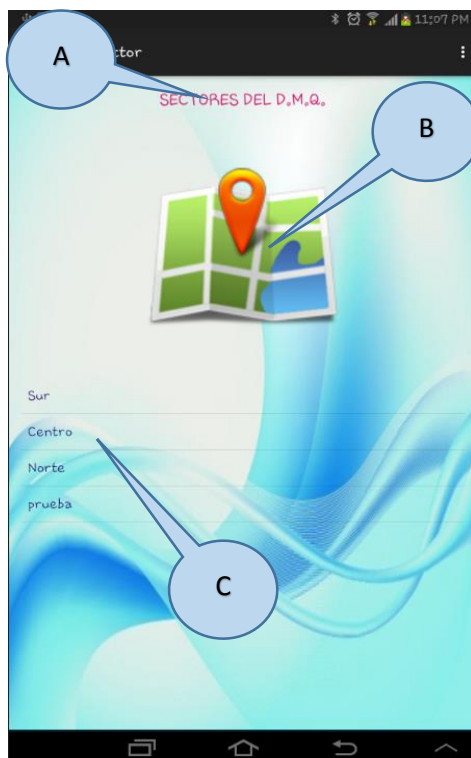


Figura 46. *Detalle de los sectores*

Tabla 26. Detalle de hoteles, búsqueda y cerrar sesión

Ítem	Representación	Descripción
A	ListView	Listado de hoteles
B	TabHost	Selección menú hoteles
C	TabHost	Selección búsqueda
D	<item android:title="Salir"	Cerrar sesión

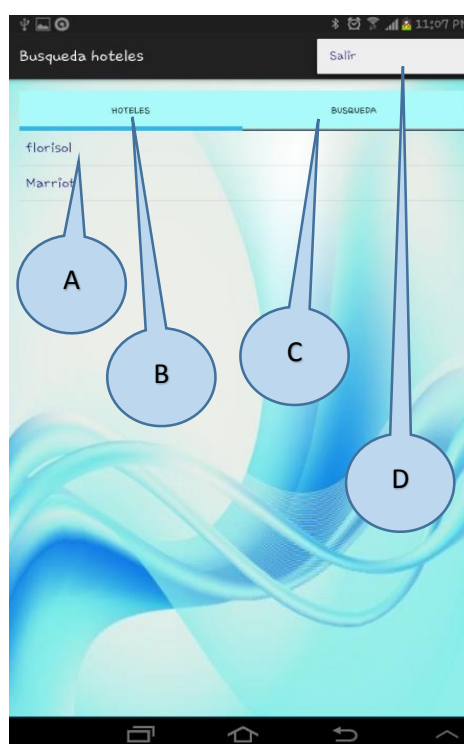


Figura 47. Lista de hoteles, búsqueda y cerrar sesión

Tabla 27. *Detalle de servicios de hotel*

Ítem	Representación	Descripción
A	ListView	Listado de servicios
B	TabHost	Selección menú servicios
C	TabHost	Selección Ubicación

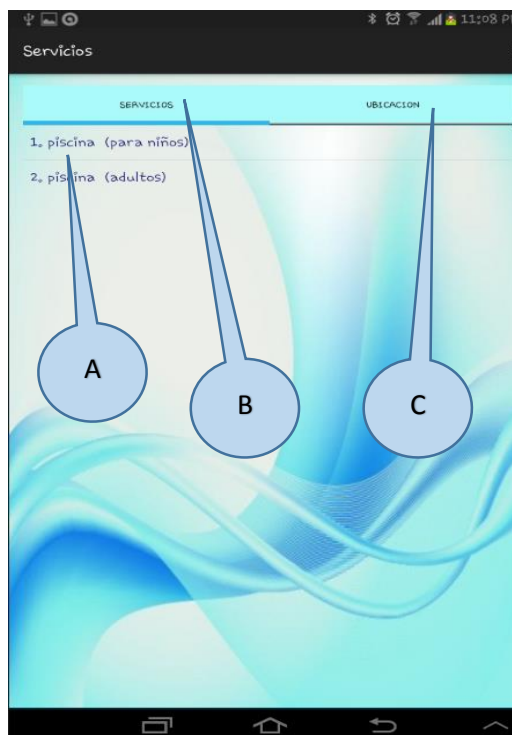


Figura 48. Servicios y ubicación de hoteles

Tabla 28. Detalle de como desea llegar al hotel requerido

Ítem	Representación	Descripción
A	TabHost	Selección menú servicios
B	TabHost	Selección Ubicación
C	ImageView	Imagen de ubicación
D	ImageView	Imagen conducir
E	Button	Mostrar del mapa conduciendo
F	ImageView	Imagen caminando
G	Button	Mostrar mapa caminando
H	ImageView	Imagen bicicleta
I	Button	Mostrar mapa bicicleta

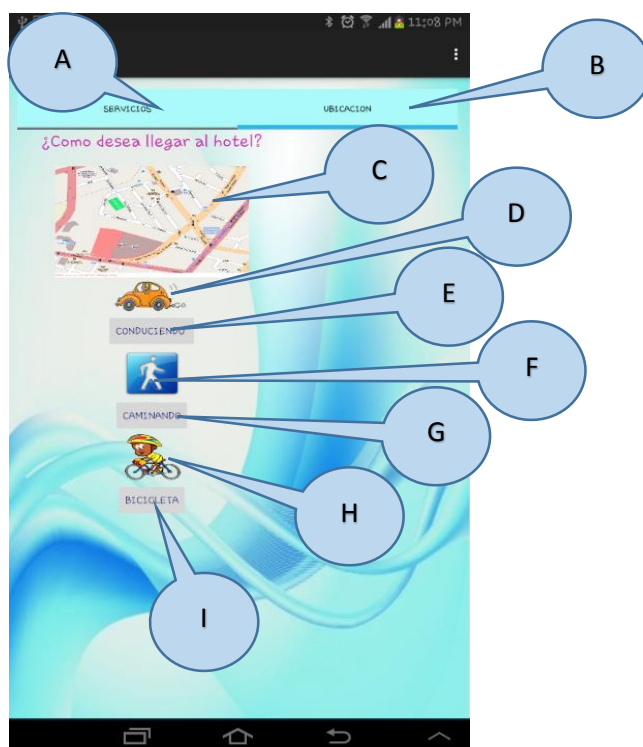


Figura 49. Llegar al hotel deseado

Tabla 29. *Búsqueda de hoteles*

Ítem	Representación	Descripción
A	TabHost	Selección menú hoteles
B	TabHost	Selección menú búsqueda
C	EditText	Parámetro de búsqueda
D	Button	Búsqueda de hotel
E	ImageView	Imagen Búsqueda

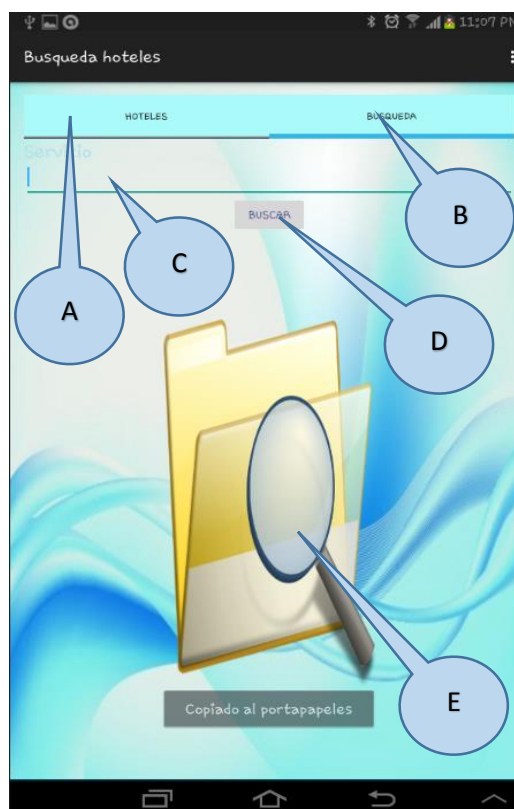


Figura 50. Búsqueda de un hotel

5.03. Especificación de pruebas de unidad

La construcción de este software tiene como objetivo satisfacer una necesidad planteada sobre todo por parte del gerente de la empresa STAFF VACATIONS CIA. LTDA., y también de los usuario "turistas".

Para asegurar que el software ha alcanzado los niveles de calidad acordados es necesario que el producto se vaya evaluando a medida que se va construyendo.

El procedimiento para realizar dichas pruebas de unidad definirá de forma detallada los pasos que se ejecuta en el sistema. Analiza en detalle cada una de las fases que forma este procedimiento, describiendo, las actividades a realizar y los datos de entrada y salida en el sistema.

El alcance es que el procedimiento está dirigido a realizar las pruebas de unidad.

¿Qué se va a probar? Las funciones individuales o métodos: se probarán las entradas y las salidas y se comprobará que los valores obtenidos son los esperados. Es decir, se prueba el código aislado, independiente del resto del sistema.

Tabla 30. *Especificación de la prueba de unidad PDU001*

Identificador de la prueba:	EPDU001
Método a probar:	Validación del acceso al sistema
Objetivo de la prueba:	Comprobar que el método de validación del campo de usuario y contraseña cumpla su función, y el usuario tiene que encontrarse registrado, y su contraseña sea la correspondiente.
Datos de Entrada	
Usuario y contraseña	
Resultado Esperado	
1.	El usuario se encuentre registrado
2.	La contraseña sea correcta
3.	Que el usuario tenga el rol correspondiente
Comentarios	
Que no me permita ingresar al sistema si no están llenos los campos.	

Tabla 31. *Especificación de la prueba de unidad PDU002*

Identificador de la Prueba:	PDU002
Método a Probar:	Ingreso datos del cliente.
Objetivo de la Prueba:	Comprobar que los datos ingresados del cliente sean correctos.
Datos De Entrada:	Tipo de usuario o cliente
	Nombre del usuario.
	Apellido del usuario.
	Teléfono.
	Email.
	Clave.
Confirmación de clave.	
Resultado Esperado:	
Guardar la información del cliente correctamente	
Comentarios:	
Ninguna	

Tabla 32. Especificación de la prueba de unidad PDU003

Identificador de la Prueba:	PDU003
Método a Probar:	Ingreso de una nueva ubicación de hotel
Objetivo de la Prueba:	Comprobar que los datos ingresados "coordenadas" del hotel sean correctos.
Datos De Entrada:	Sector del hotel Nombre del hotel Dirección del hotel Longitud "automático" Latitud "automático" Ubicación en el mapa Guardar
Resultado Esperado:	Guardar la ubicación correcta del hotel y se visualice en la App
Comentarios:	Asegurar que se ubique correctamente el hotel en el mapa.

Tabla 33. Especificación de la prueba de unidad PDU004

Identificador de la Prueba:	PDU004
Método a Probar:	Ingreso tipo de usuario
Objetivo de la Prueba:	Comprobar que los datos ingresados sean el correcto.
Datos De Entrada:	Tipo de Rol
Resultado Esperado:	Que el tipo de rol del usuario sea el correcto para una buena administración.
Comentarios:	Ninguno

Tabla 34. Especificación de la prueba de unidad PDU005

Identificador de la prueba:	PDU005
Método a Probar:	Ingreso de actividades de trabajo
Objetivo de la Prueba:	Registrar las actividades y datos reales que se realizan en la orden de trabajo.
Datos de Entrada	
	1.- Detalle de los servicios que se dará
	2.- Diagnostico de los servicios que se dará
	3.- Tipo de usuario
Resultados Esperados	
	1.- Registro correcto de los servicios y ubicación del hotel
	2.- Acceso inmediato a los servicios y ubicación del hotel
Comentarios: Ninguno	

5.04. Especificación de pruebas de aceptación

Esta prueba se elaboran a partir de las necesidades del usuario, durante un lapso de tiempo, define los objetivos de la prueba de un sistema, se debe especificar los aspectos a tratar, se utilizara como referencia por los usuarios del proyecto, en este caso para los turistas que obtengan la App.

Tabla 35. Especificación de prueba de aceptación EPDA001 aprobar App

Identificador de Prueba:	EPA001
Caso de Uso	UCR001
Tipo de usuario	Jefe de Sistemas
Objetivo de la Prueba	Aprobar la App
Secuencia de eventos	
	1.- Ver la necesidad de este sistema para la sociedad.
	2.- Alcance y aceptación del sistema (App)
Resultados Esperados	
	Aprobación por parte del jefe de sistemas y del gerente general de la empresa.
Comentarios	
	Ninguno
Estado:	Aceptado

Tabla 36. Especificación de prueba de aceptación EPDA002 desarrollo App

Identificador de Prueba:	EPA002
Caso de Uso	UCR002
Tipo de usuario	Programador
Objetivo de la Prueba	Desarrollo de la App
Secuencia de eventos	
1.- Aprobación del proyecto	
2.- Escoger las plataformas de programación	
3.- Apoyarse en los requerimientos levantados para la aprobación del sistema.	
4.- Realizar pruebas de implementación	
Resultados Esperados	
Concluir con la App en buen estado sin problemas	
Comentarios	
	Ninguno
Estado:	Aceptado

Tabla 37. Especificación de prueba de aceptación EPDA003 entrega App

Identificador de Prueba:	EPA003
Caso de Uso	UCR003
Tipo de usuario	Programador
Objetivo de la Prueba	Entrega de la App
Secuencia de eventos	
1.- Realizar pruebas local host	
2.- Realizar pruebas con datos reales y con servidor externo.	
3.- Entrega de la App	
Resultados Esperados	
Entrega de la App funcionando correctamente	
Comentarios	
	Ninguno
Estado:	Aceptado

Tabla 38. *Especificación de prueba de aceptación EPDA004 registro de cliente*

Identificador de Prueba:	EPA004
Caso de Uso	UCR004
Tipo de usuario	Vendedor
Objetivo de la Prueba	Registrar al cliente correctamente
Secuencia de eventos	
1.- Tipo de sistema	
2.- Tipo de usuario	
3.- Nombre usuario	
4.- Apellido usuario	
5.- Teléfono	
6.- Email	
7.- Clave	
Resultados Esperados	
Guardar la información del cliente correctamente	
Comentarios	
	Ninguno
Estado:	Aceptado

Tabla 39. *Especificación de prueba de aceptación EPDA005 uso de la App*

Identificador de Prueba:	EPA005
Caso de Uso	UCR005
Tipo de usuario	Usuario/ Cliente
Objetivo de la Prueba	Manipulación de la App
Secuencia de eventos	
1.- Instalación de la App en el dispositivo móvil	
2.- Búsqueda del hotel	
3.- Servicios y ubicación del hotel	
Resultados Esperados	
Acogida del sistema (App) de geo localización hotelera.	
Comentarios	
	Ninguno
Estado:	Aceptado

5.05. Especificación de pruebas de carga

El objetivo de las especificaciones de pruebas de carga es el de identificar el tiempo de respuesta del software para transacciones, casos de negocio bajo o ingreso de datos. Adicional a esto, las pruebas evalúan las características de desenvolvimiento del sistema (tiempos de respuestas, tasas de transacción y otros problemas sensibles con respecto al tiempo).

Es decir las pruebas de carga miden las condiciones en las que el sistema se somete a una transición, su carga de trabajo y así evaluar la habilidad del sistema para continuar su correcto funcionamiento.

Tabla 40. *Especificación de la prueba de carga EPC001.*

Identificador de Prueba:	EPC001
Tipo de prueba	Carga
Objetivo de la prueba:	Determinar si el sistema soporta una carga masiva de consultas o de ingreso de clientes, equipos en la base de datos al mismo tiempo.
Descripción	
Se utiliza el programa "LoadUI web2.0" para la ejecución de esta prueba	
Número de Usuarios:15	
Empezar con 2 usuarios	
Agregar cada 2 seg. 2 usuarios	
Resultados Esperados	
El sistema sea capaz de soportar cargas de información notablemente Considerables.	
Comentarios	
La ejecución de la prueba fue satisfactoria con 2 usuarios.	

La presente imagen nos indicara donde realizaremos la prueba de carga, en este caso el navegador Google Chrome, en la misma pondremos la dirección URL de nuestro proyecto a realizar la respectiva prueba.

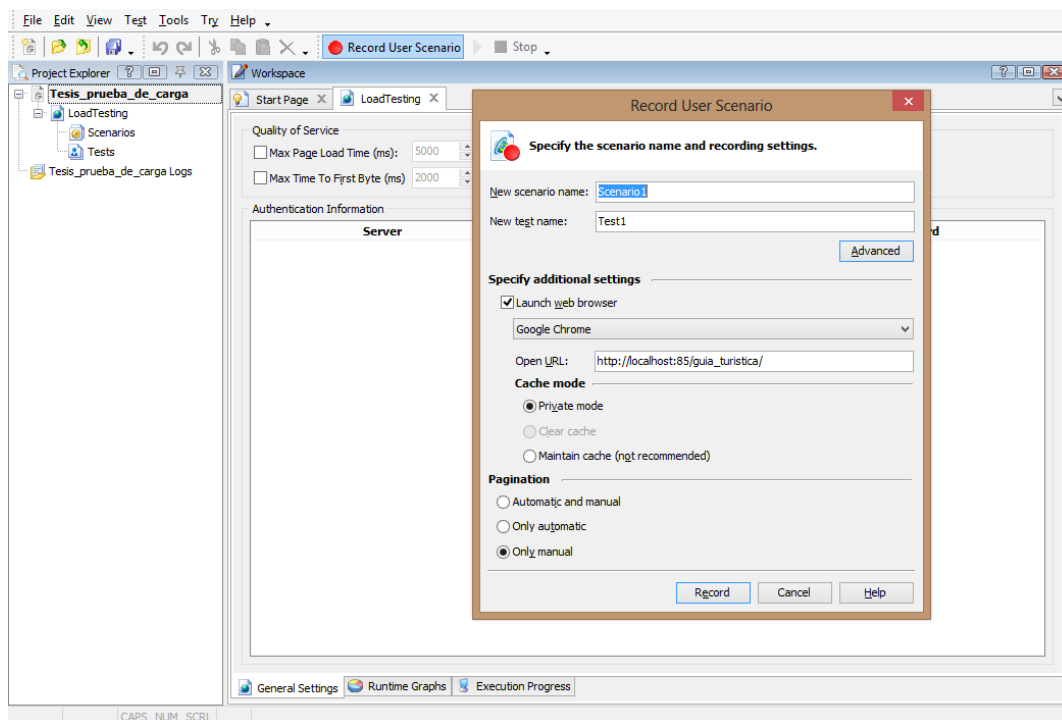


Figura 51. Software para realizar las pruebas de carga

Se procede a realizar las respectivas pruebas en la pantalla principal del sistema donde se ira grabando cada proceso que el usuario realice, y cada uno de los procesos realizados tienen que ser nombrados de manera que se los pueda reconocer para las pruebas futuras a realizarse.



The image shows a login form titled 'INGRESO DE USUARIOS'. It has a blue background. There are two input fields: 'Usuario:' with a user icon and 'Contraseña:' with a lock icon. Below the fields is a button labeled 'Ingresar'.

Figura 52. Inicio de la pruebas de carga

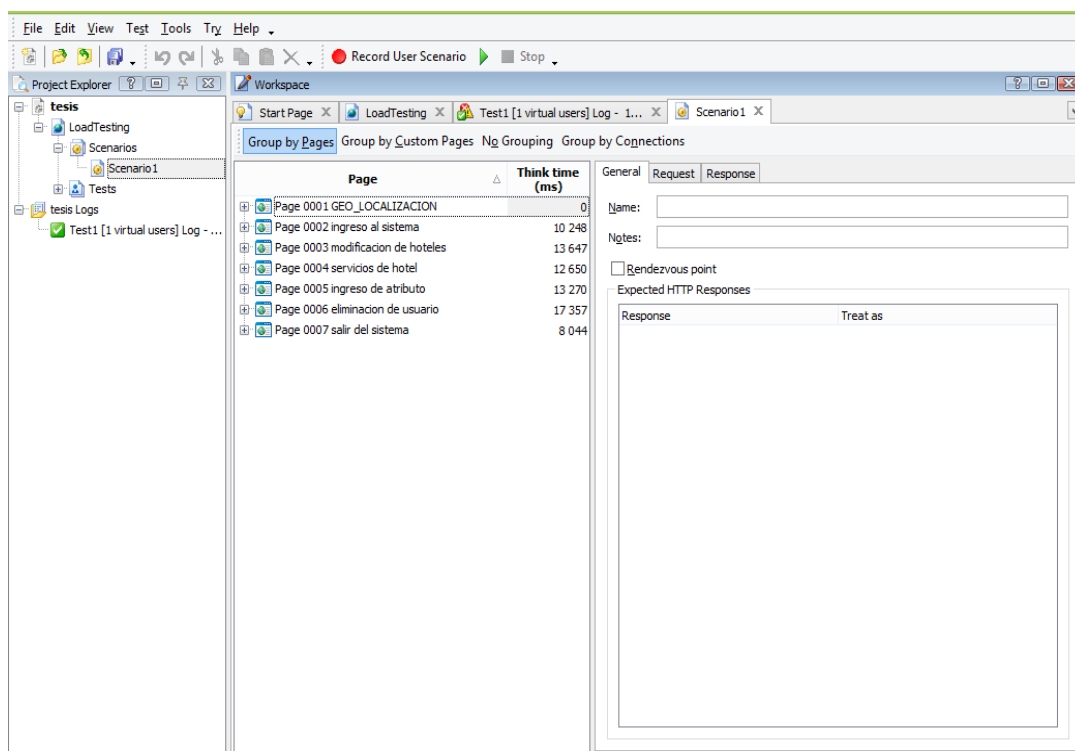


Figura 53. Pruebas realizadas

A continuación podemos apreciar la cantidad de solicitudes que el usuario realiza al servidor, en el mismo podemos visualizar que no existe ningún error en la prueba.

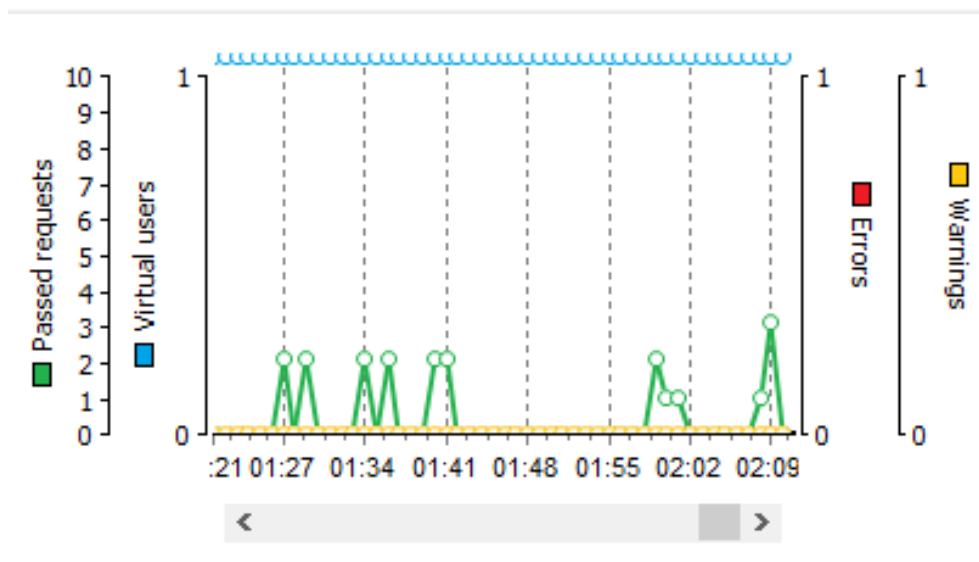


Figura 54. Cantidad de solicitudes realizadas por el usuario

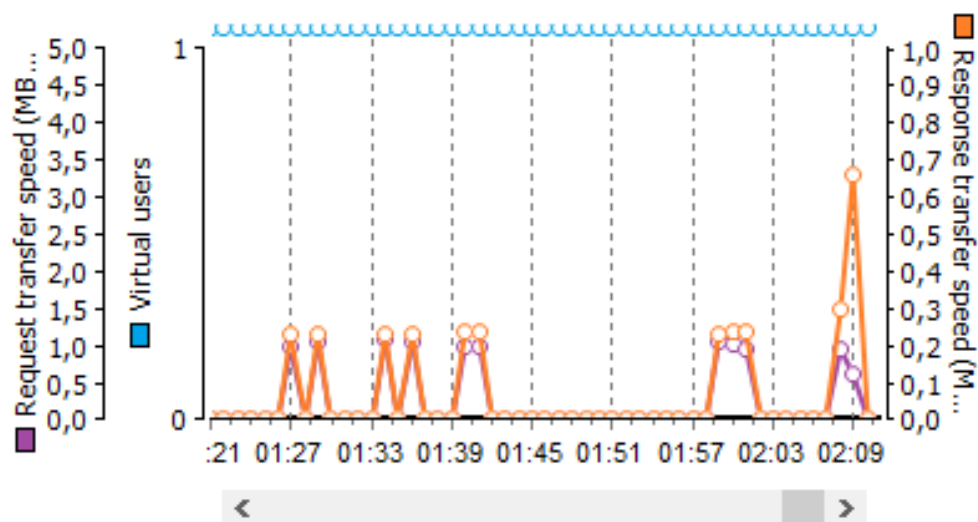


Figura 55. Velocidad de transmisión de solicitudes enviadas al servidor

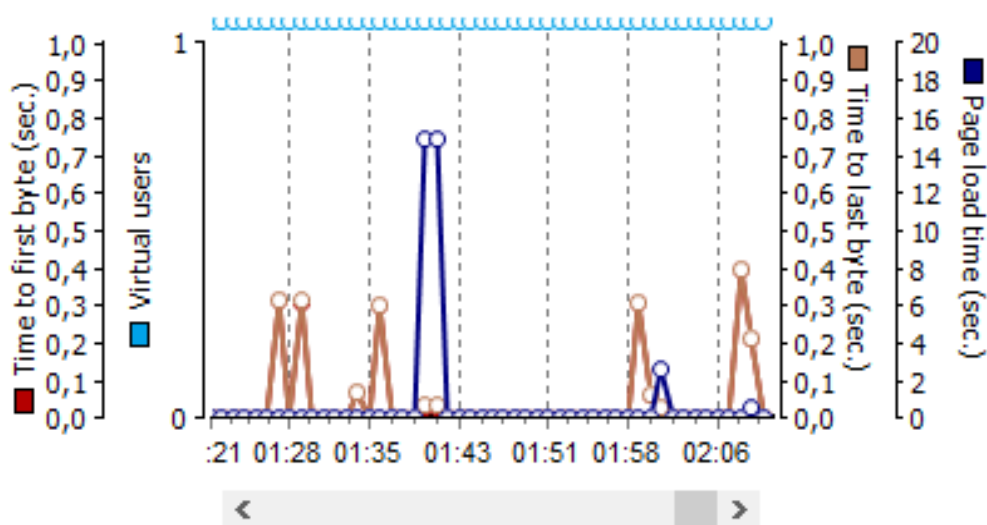


Figura 56. Tiempo de carga de las páginas ejecutadas

En la siguiente figura se representa de forma general el tiempo que tarda la transferencia de una petición, y la velocidad de respuesta, así, determinando la velocidad más alta en la presente prueba de carga.

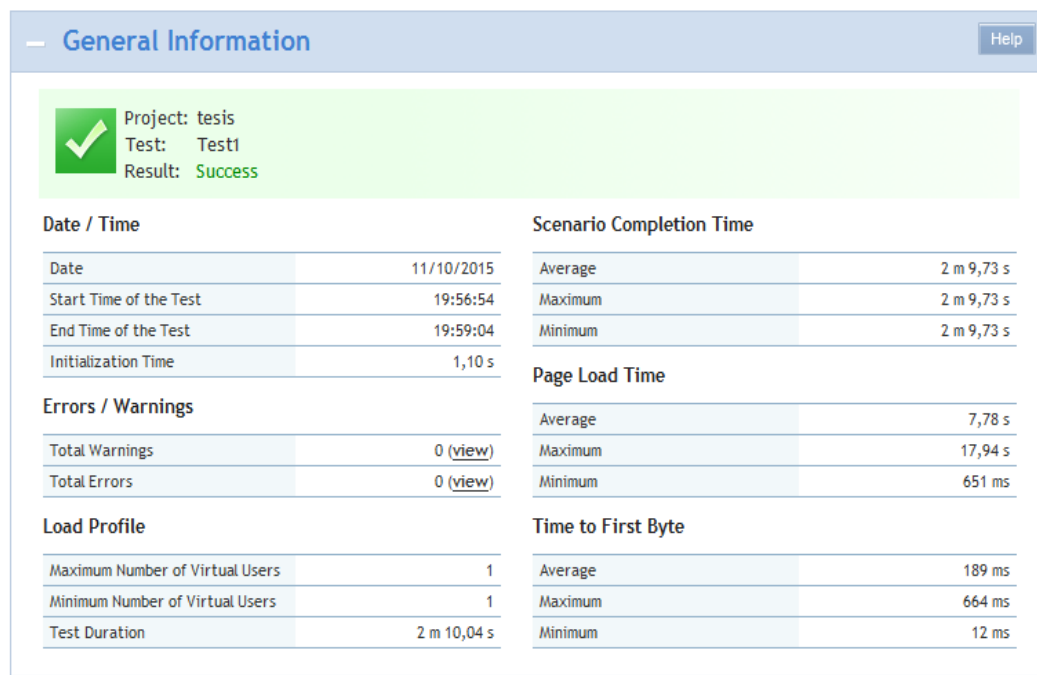


Figura 57. Informe general de las pruebas de carga con dos usuarios

Tomando de la *figura 53*, con sus respectivos nombres de los procedimientos realizados se hará las pruebas con distintos números de usuarios y tiempos, el listado de los procesos pertenece a cada una de las pruebas a realizarse.

Tabla 41. *Especificación de la prueba de carga EPC002*

Identificador de Prueba:	EPC002
Tipo de prueba	Carga
Objetivo de la prueba:	Determinar si el sistema soporta una carga masiva de consultas o de ingreso de clientes, equipos en la base de datos al mismo tiempo.
Descripción	
Se utiliza el programa "LoadUI web2.0" para la ejecución de esta prueba	
Número de Usuarios:15	
Empezar con 2 usuarios	
Agregar cada 2 seg. 2 usuarios	
Resultados Esperados	
El sistema sea capaz de soportar cargas de información notablemente Considerables.	
Comentarios	
La ejecución de la prueba fue satisfactoria con 15 usuarios.	

La grafica de LoadUI Web, muestra la nueva prueba con la creación de 15 usuarios, iniciando 2 usuarios e incrementando 2 cada segundo.

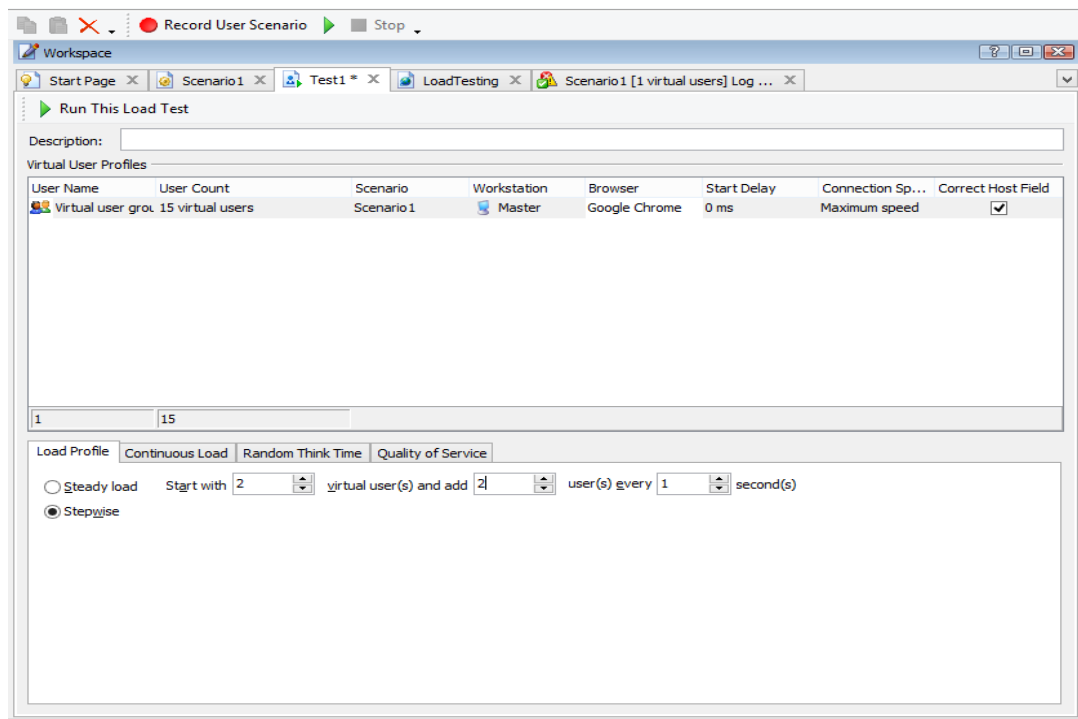


Figura 58. Interfaz de LoadUI Web para 15 usuarios

La siguiente figura muestra la cantidad y calidad de solicitudes que realiza un usuario al servidor.

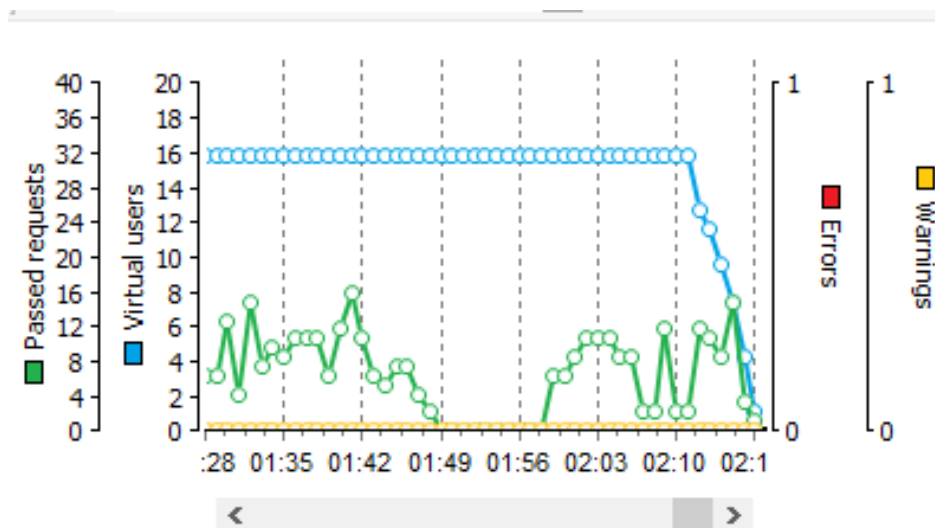


Figura 59. Pruebas de carga con 15 usuarios, solicitudes enviadas por el usuario

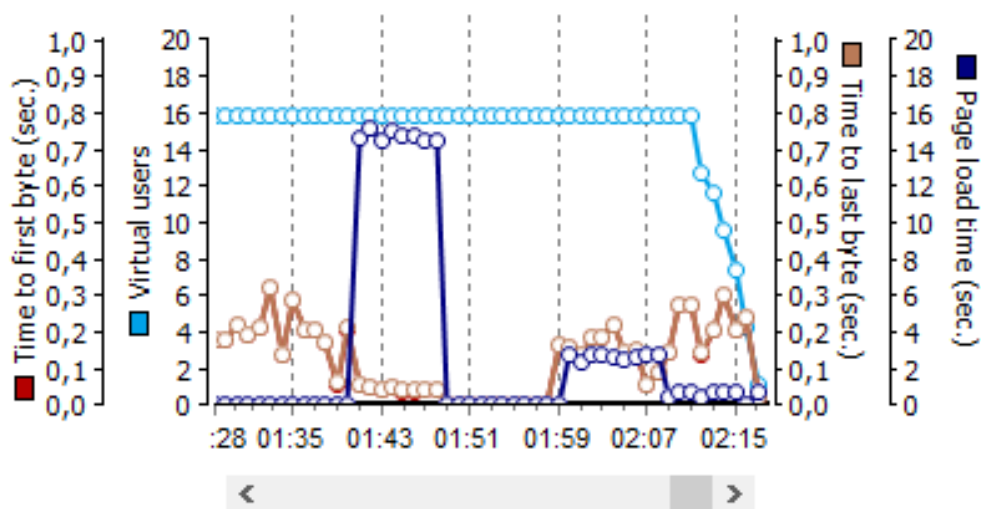


Figura 60. Pruebas de carga con 15 usuarios, tiempo de carga de las paginas

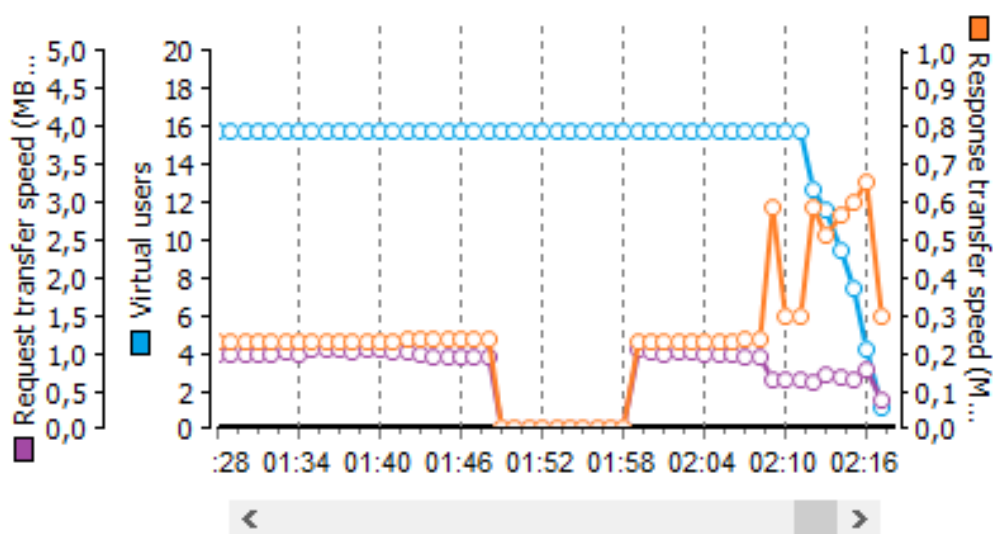


Figura 61. Pruebas de carga con 15 usuarios, transmisión de solicitudes

La siguiente figura representa la información general de la prueba de carga realizada con éxito, sin problemas con 15 usuarios.

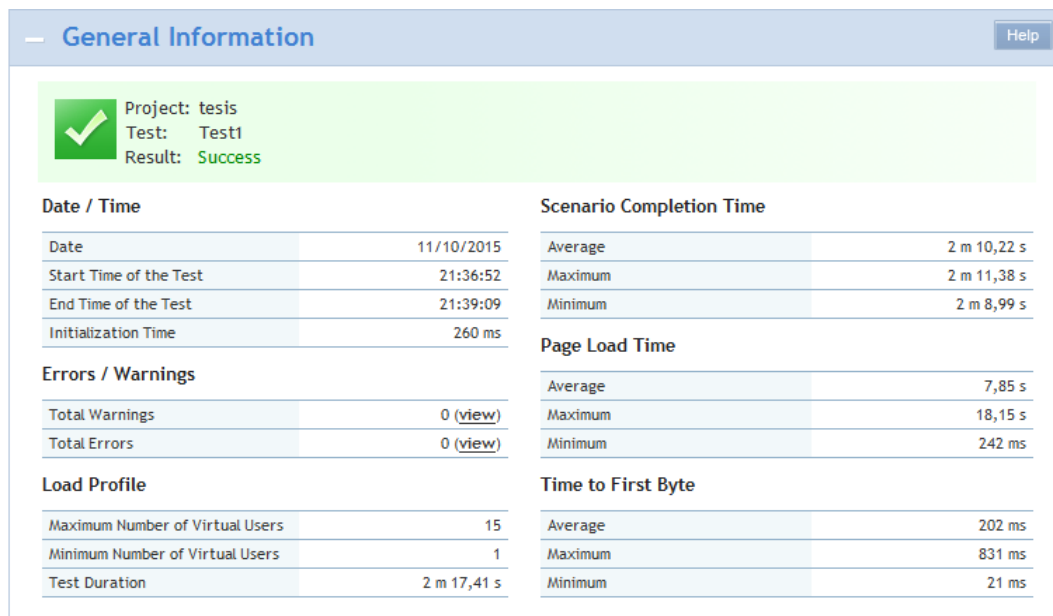


Figura 62. Pruebas de carga con 15 usuarios, información de la prueba generada

5.06. Configuración del ambiente mínima/ideal

La presente aplicación se elaboró conforme a los estándares de programación establecidos para que los procesos se ejecuten sin ningún problema y el tiempo de respuesta sea el mínimo, para que así las acciones que se ejecuten en el sistema no tenga ningún inconveniente en el momento que se necesite, estos pueden ser: guardar, modificar y eliminar.

Características técnicas.

- Wampserver: Servidor Local de la aplicación.
- Está basado en MySQL Workbench 6.0 CE.
- NetBeans IDE 8.0.
- Android Studio.

REQUERIMIENTOS:

Los requerimientos que se detallara a continuación son los mínimos para que el sistema desarrolle con total normalidad sus procesos en cualquier estación de trabajo así como subidos a un servidor

REQUERIMIENTOS DEL HARDWARE

Equipo Servidor:

- Procesador Intel Core2 Dúo de 2 GHz.
- Disco Duro de 100 GB de espacio libre para datos del sistema
- Memoria RAM de 4 GB.

Equipo Cliente:

- Procesador Intel Core2 Dúo de 2 GHz.
- Memoria RAM de 2 GB
- Disco duro de 80 GB suficientes para alojar información, imágenes y complementos del sistema, además para la instalación de los navegadores necesarios.

REQUERIMIENTOS DEL SOFTWARE

Equipo Servidor

Sistemas operativos

- Windows Server 2008 o superior

Manejadores de Base de datos

- MySQL Workbench 6.0 CE.

Equipo Cliente

En este caso el cliente/usuario va hacer uso de la App, por ello debe tener un teléfono inteligente que brinde las siguientes características como mínimo:

- Soporte para GSM 3G.
- Soporte para GSM / GPRS / EDGE (850 / 900 / 1.800 / 1.900 MHz).
- Soporte para HSPA+ (850 / 900 / 1.900 / 2.100 MHz).
- Wi-Fi Direct disponible.
- Procesador de aplicaciones Dual Core disponible.
- Velocidad de CPU de 1,7 GHz.
- S.O. Android Jelly Bean 4.2.2.
- Dimensiones (alto x ancho x prof.) de 124,6 x 61,3 x 8,94 mm.
- Pan de datos.

CAPÍTULO VI: ASPECTOS ADMINISTRATIVOS

6.01. Recursos

Tabla 42. *Recursos humanos y económicos*

Recursos Humanos			
HUMANO	NOMBRE	ACTIVIDAD	RESPONSABILIDAD
Tutor	Ing. Roberto Morales	Director del proyecto	Responsable de guiar y perfeccionar el proceso.
Programador	Alumno: Xavier Yanacallo	Desarrollador del Sistema	Desarrollo del Sistema
Recursos Económicos			
DESCRIPCIÓN	CANTIDAD	VALOR UNITARIO	VALOR TOTAL
Computadora	1	1100	1100
Imp. a color	200	0,25	50
Impresiones a b/n	200	0,05	10
Internet	6 Meses	43,20	259,20
Servicios Básicos	3	12,5	35,5
Capacitación	3 meses	30	90
Transporte capacitación	3 meses	6,40	19,20
Transporte tutorías	6 meses	6,40	38,40
Alimentación	3 meses	12	36
VALOR TOTAL			1638,30

Tabla 43. *Detalle de los equipos utilizados*

CANTIDAD	EQUIPO	DESCRIPCION
1	Computador	- Acer Core i5 de 2.5 GHz
		- 6 GB RAM
		- 500 GB Disco Duro
		- DVD ritter
1	Impresora	Epson L555

6.02. Presupuestos

Tabla 44. *Presupuesto del proyecto*

PRESUPUESTO			
DETALLE	VALOR UNITARIO	CANTIDAD	VALOR TOTAL
Computadora	1100	1	1100
Impresiones a color	0,25	200	50
Impresiones a b/n	0,05	200	10
Anillados	2	3,50	7
Empastados	8	1	8
Internet	43,20	6 meses	259,20
Capacitación	30	3 meses	90
Seminario	785,15	1	785,15
Transporte capacitación	6,40	3 meses	19,20
Transporte tutorías	6,40	6 meses	38,40
Alimentación	12	3 meses	36
VALOR TOTAL			2402,95

6.03. Cronograma

El cronograma de actividades es el cual donde se detalla cada uno de los pasos que se desarrolló en el sistema con fecha designada, ello podemos ver en el *Anexo A.04.*

CAPÍTULO VII: CONCLUSIONES Y RECOMENDACIONES

7.01. Conclusiones

Luego de haber realizado el sistema de geo localización hotelera, vemos que la App desarrollada es una herramienta que contribuye al desempeño de la empresa STAFF VACATIONS CIA. LTDA., misma que facilito la información y requerimientos para el desarrollo.

El presente software ayudo a mejorar y sacar a flote las destrezas en la programación de software, tales como el levantamiento de requerimientos, el desarrollo de los caso de uso, modelamiento de base de datos, lenguajes de programación, etc., y de esa manera nos ayudó a mejorar los niveles de conocimiento.

Una vez terminada la App puede ser implementada en cualquier empresa turística del medio, la utilización de esta herramienta facilitara a la comodidad de los turistas que llegan al país, ya que la obtención de esta App es muy fácil y de costo muy bajo.

7.02. Recomendaciones

Primordialmente se recomienda que se dé más énfasis al estudiante en leguajes de programación actuales, para así pueda desarrollar el software con mayor facilidad.

También se recomienda que la parte de ingreso de datos solo sea manipulado por el administrador ya que pueden ingresar datos erróneos o repetitivos y así provocar daños en el funcionamiento de la App. Para salvaguardar los datos de la aplicación se debe realizar Backus, por lo general cada 8 días. Para un mejor desempeño del sistema se debe tener muy en cuanta la parte de soporte técnico al momento de realizar el mantenimiento del sistema, este debe tener conocimientos básicos del tema para no registrar datos incorrectos y provocar daños.

ANEXOS

Anexo A.01. Matriz de análisis de Alternativas

Tabla 45. *Matriz de alternativas*

Variables o estrategias	Necesidad de Recursos			Factibilidad Política	Duración del Proyecto	Puntaje	Prioridad
	Humanos	Técnicos	Financieros				
Una buena acogida por parte de los turistas, a la vez existan buenos ingresos económicos al hotel es necesario guiarse mediante un mapa de ubicación.	4	4	4	4	3	19	MEDIO MEDIO -ALTO
Con los datos correctos de esta App se dará una información confiable a los turistas y así podrán elegir destinos confiables, seguros y en poco tiempo.	5	5	5	3	3	21	ALTO
Analizar, diseñar e implementar un sistema de ubicación que simplifique procesos de búsqueda, control y evaluación de los servicios hoteleros.	4	4	5	3	3	19	MEDIO MEDIO -ALTO
Total	13	13	14	10	9	59	
PUNTUACION							
	0 AL 5	6 AL 10	11 AL 15	16 AL 20	21 AL 25		
	BAJO	MEDIO BAJO	MEDIO	MEDIO ALTO	ALTO		

Anexo A.02. Matriz de Impacto de Objetivos

Tabla 46. *Matriz de impacto*

Objetivos	Factibilidad a lograrse	Impacto de Género	Impacto Ambiental	Relevancia	Sostenibilidad
Una buena acogida por parte de los turistas, a la vez existan buenos ingresos económicos al hotel es necesario guiarse mediante un mapa de ubicación.	_Conocer y guiarse en un mapa del D.M.Q. _Tener una buena guía de ubicación de los hoteles	Intervendrán en el proceso: El desarrollador de la aplicación y los usuarios de la misma.	El impacto ambiental que se da en esta App es de gran beneficio ya que se reduce el costo de impresiones de guías y/o mapas turísticos y por ende la contaminación se reducirá en gran cantidad. También se ahorrará la tala de bosques del cual se contiene la materia prima que en este caso es el papel.	Alto ingreso económico para los hoteles del D.M.Q.	El desarrollador de esta aplicación debe tener un conocimiento exacto de la ubicación de los hoteles.
Con los datos correctos de esta App se dará una información confiable a los turistas y así podrán elegir destinos confiables, seguros y en poco tiempo.	_Acoplarse a un tiempo menor en trasladarse de un lugar a otro.	Intervendrán en el proceso: El desarrollador de la aplicación y los usuarios de la misma.		Con los datos correctos se logrará una planificación adecuada de las actividades a desarrollar por los turistas.	Los usuarios de esta App serán los que asignen el beneficio del mismo y si crean que es conveniente ampliar a los demás sectores del país y a la vez sea un sistema público ya que por el momento es solo para los que estén afiliados a la empresa.
Analizar, diseñar e implementar un sistema de ubicación que simplifique procesos de búsqueda, control y evaluación de los servicios hoteleros.	_Que los hoteles del D.M.Q sean uno de los lugares preferidos, de buen servicio y de fácil ubicación por parte de los turistas.	Intervendrán en el proceso: El desarrollador de la aplicación y los usuarios de la misma.		Conocimiento adecuado de la ubicación de los hoteles del D.M.Q. Con una información confiable para la correcta ubicación.	

Anexo A.03. Especificaciones de estándares de programación

Tabla 47. *Estándares de programación*

Tipo de control	Prefijo	Ejemplo	Descripción de Estándares
Label	lbl	lbl_ingreso	Las tres primeras letras deben ser la abreviatura del objeto tipo label. Ejm: lbl_ingreso
TextBox	txt	txt_nombre	Las tres primeras letras deben ser la abreviatura del objeto tipo TextBox. Ejm: txt_cliente.
Imagen	img	img_nombre	Las tres primeras letras deben ser la abreviatura del objeto tipo Imagen. Ejm: img_Entregar.
Button	btn	btn_nombre	Las tres primeras letras deben ser la abreviatura del objeto tipo Button. Ejm: btn_cam.
GridView	gdv	gdv_nombre	Las tres primeras letras deben ser la abreviatura del objeto tipo GridView. Ejm: gdv_Clientes.
Table	tbl	tbl_nombre	Las tres primeras letras deben ser la abreviatura del objeto tipo Table.
DataSet	dts	dts_nombre	Las tres primeras letras deben ser la abreviatura del objeto tipo DataSet. Ejm dts_cliente.
Representación de los estándares para Android Studio			
TextView	txt	txt_nombre	Las tres primeras letras deben ser la abreviatura del objeto tipo TextView. Ejm txt_llegar
ImageView	image	image_mapa	Los cinco primeras letras deben ser la abreviatura del objeto tipo ImageView. Ejem image_mapa
EditText	etxt	etxt_nombre	Las cuatro primeras letras deben ser la abreviatura del objeto o tipo EditText. Ejem etxt_usuario

Anexo A.04. Cronograma de actividades

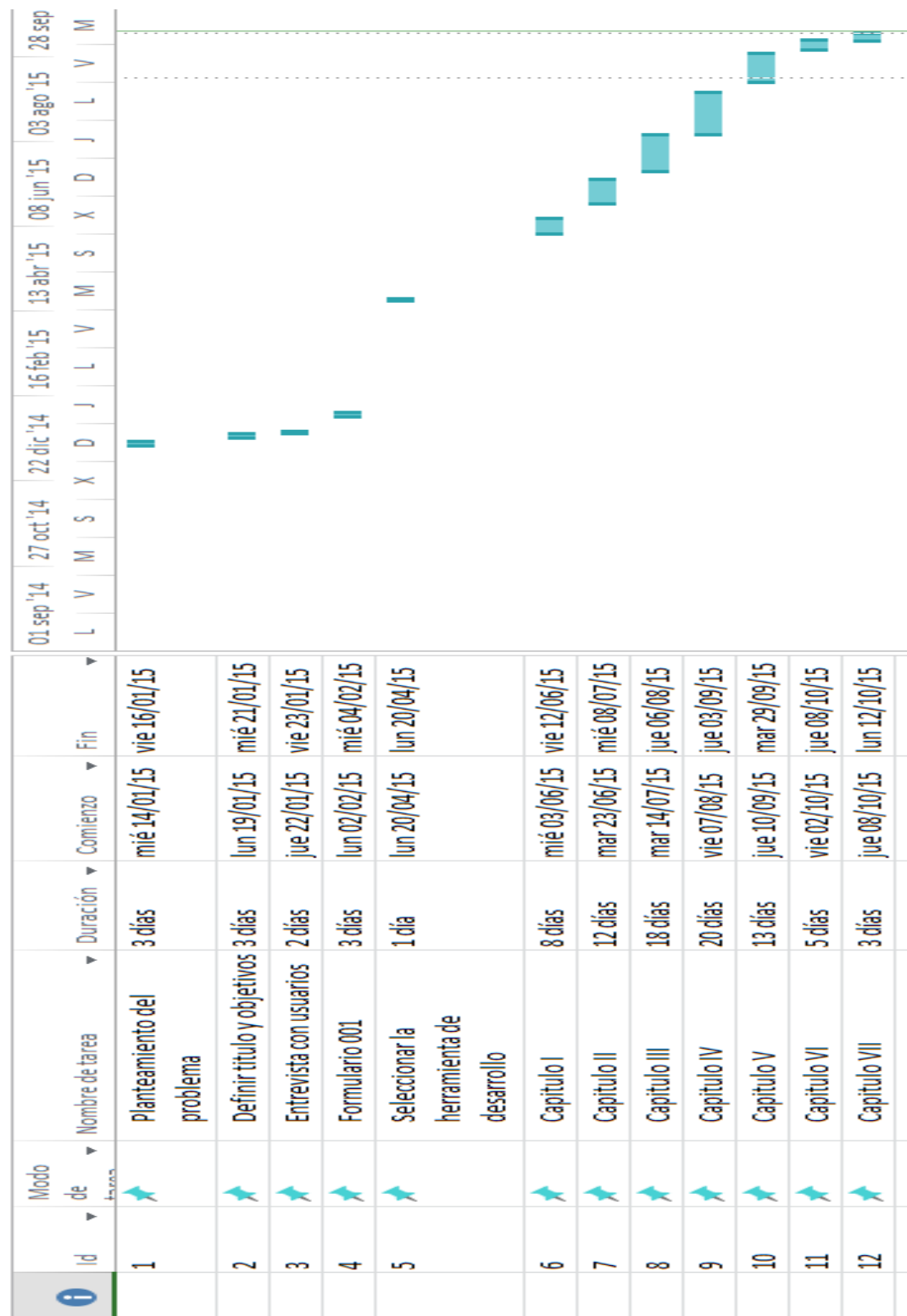


Figura 63. Cronograma de actividades para el desarrollo del sistema

Anexo A.05. Manual de instalación de aplicaciones

Instalación de MySQL Workbench CE



Figura 64. NET descarga del framework



Figura 65. Inicio de instalación de Mysql

Una vez descargada y ejecutando el instalador de MySQL te avisara que es necesario descargar "Microsoft.NET Framework 4". En caso de que no contenga la computadora.

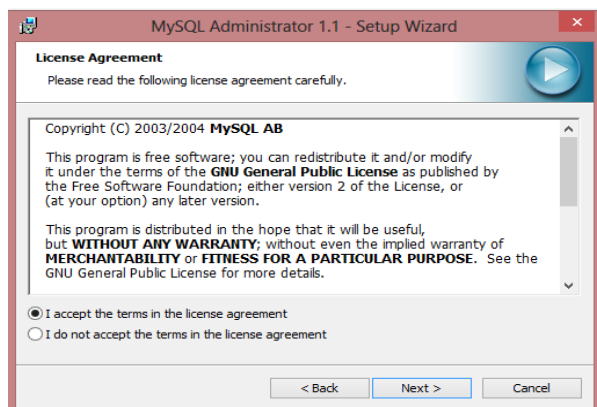


Figura 66. Aceptación de licencia MySQL

En la imagen siguiente se debe elegir la opción instalación completa en la misma se podrá instalar los complementos del motor de base de datos que se va a ocupar.

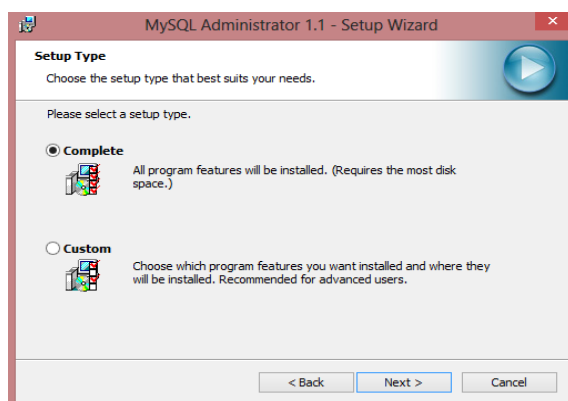


Figura 67. Opción de instalación completa de MySQL

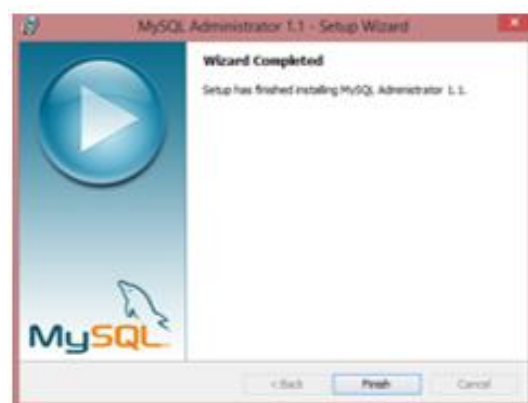


Figura 68. Finalización de la instalación

Nombre	Fecha de modifica...	Tipo	Tamaño
mysql-5.5.28-win32	14/12/2012 10:13	Paquete de Windo...	31.880 KB
mysql-administrator-1.1	22/03/2013 19:48	Paquete de Windo...	5.138 KB
mysql-workbench-gpl-5.2.44-win32	14/12/2012 10:15	Paquete de Windo...	26.955 KB

Figura 69. Instalación de mysql workbench

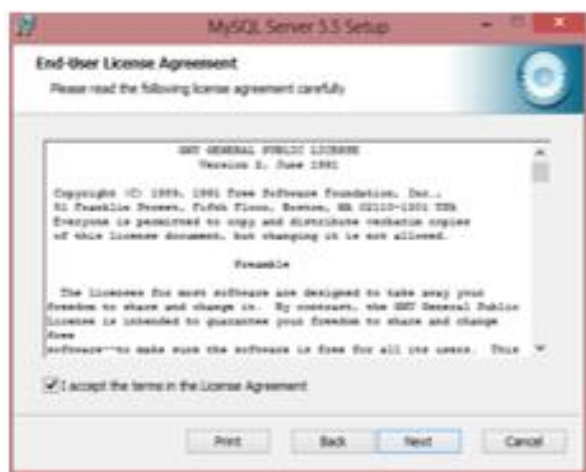


Figura 70. Aceptación de licencia del workbench



Figura 71. Iniciación de la instalación del workbench

A continuación se presentara un cuadro donde elegiremos la opción DEVELOPER MACHINE y damos en siguiente



Figura 72. Opción de instalación developer machine, server machine, dedicated MySQL Server.

Luego elegiremos la primera opción porque ayudara a optimizar al servidor el uso de motor de almacenamiento de datos.

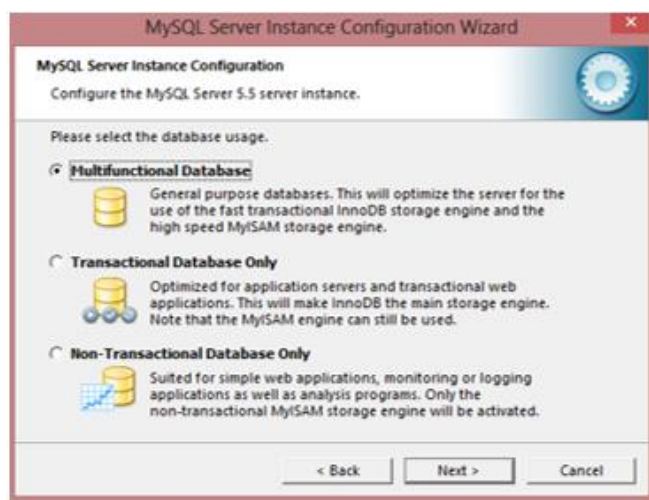


Figura 73. Opción multifuncional Data base

En las siguientes imágenes se muestra las opciones de configuración, pero si se deja de manera original no pasará nada y después en las imágenes referentes a los

usuarios se debe ingresar una contraseña para el administrador y añadir un usuario si es necesario.



Figura 74. Agregar usuarios y contraseñas



Figura 75. Asignar contraseña al usuario creado

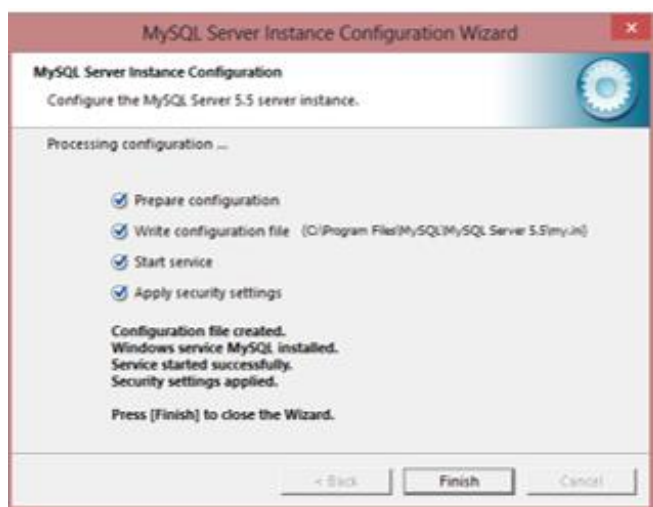


Figura 76. Final de la instalación de MySQL

Instalación de Netbeans y JDK

Vamos a utilizar esta herramienta para programar en PHP ya que nos ayudara con mayor facilidad una programación en tres capas.

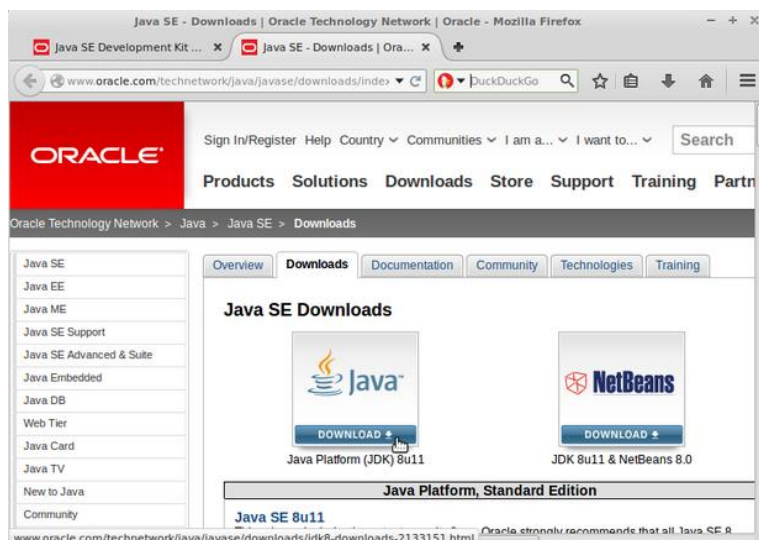


Figura 77. Descargar e instalar el java Development Kit (JDK)

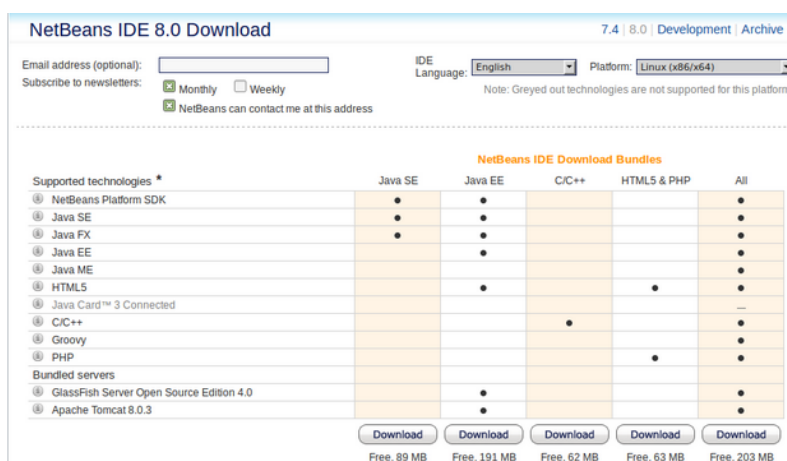


Figura 78. Descargar Netbeans

Luego de haber terminado la descarga aparecerá la siguiente ventana para continuar la instalación en la máquina.



Figura 79. Comienzo de la instalación

Si el JDK es el correcto aparecerá la siguiente página:

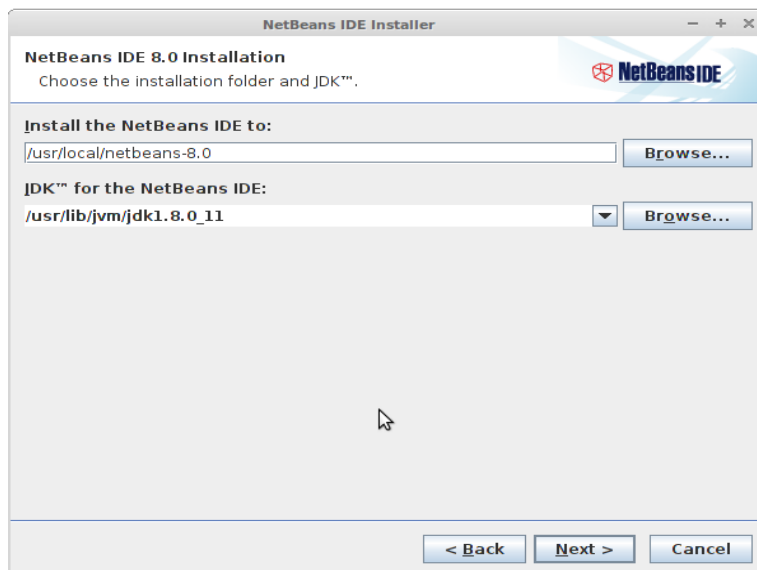


Figura 80. Elegir la carpeta de instalación y JDK

En caso de que aparezca un error será porque falta el JDK y se procederá a la descarga. Se nos recomendara instalar el Glassfish y Apache Tomcat, pondremos siguiente e instalaremos.

Luego de ellos se instalara y podrán ejecutarlo con el comando "Netbeans" y también se le podrá encontrar en el menú de la aplicación.

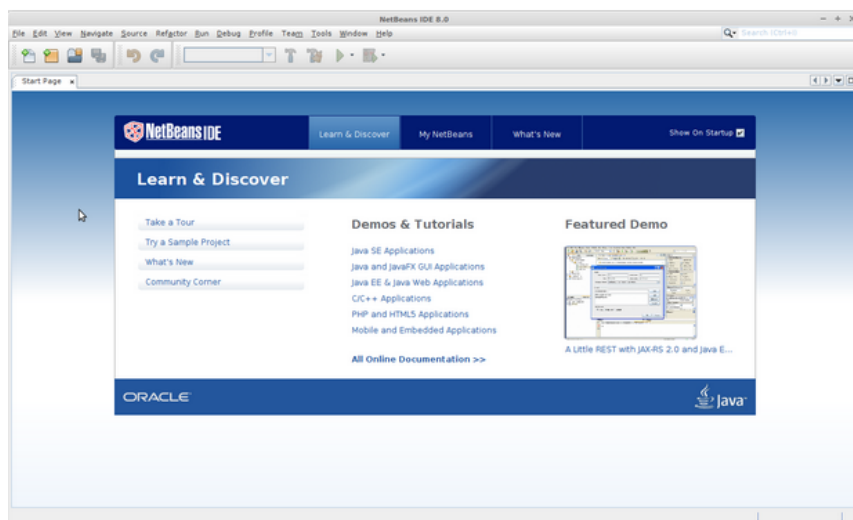


Figura 81. Utilización de Netbeans

Instalación de Wampserver

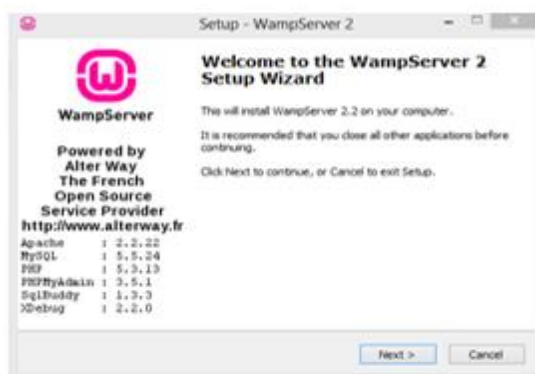


Figura 82. Inicio de instalación

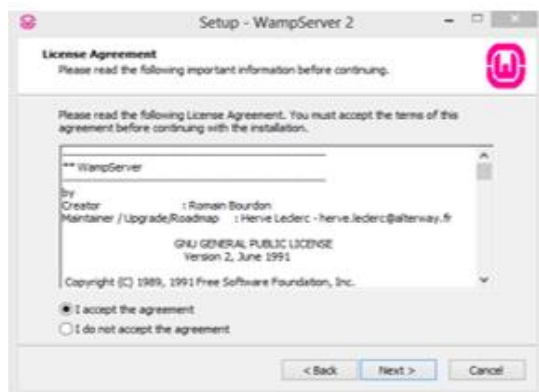


Figura 83. Términos y condiciones de la instalación

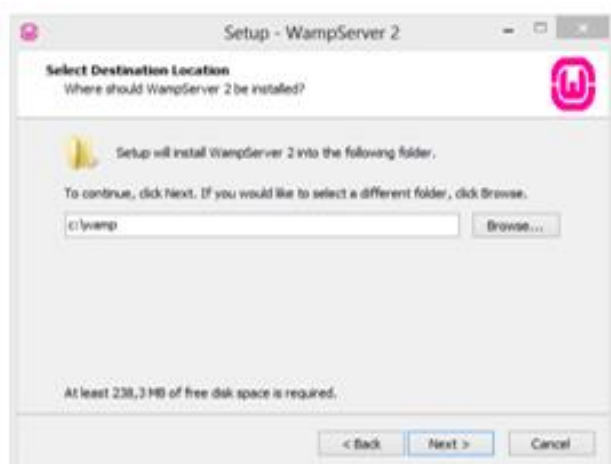


Figura 84. Seleccionamos la dirección donde se instalará

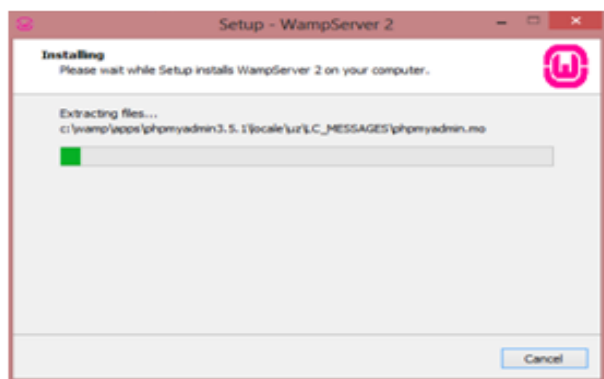


Figura 85. Progreso de instalación

Antes de terminar la instalación nos aparecerá una sugerencia para utilizar el tipo de navegador que deseamos para que funcione el Wampserver.

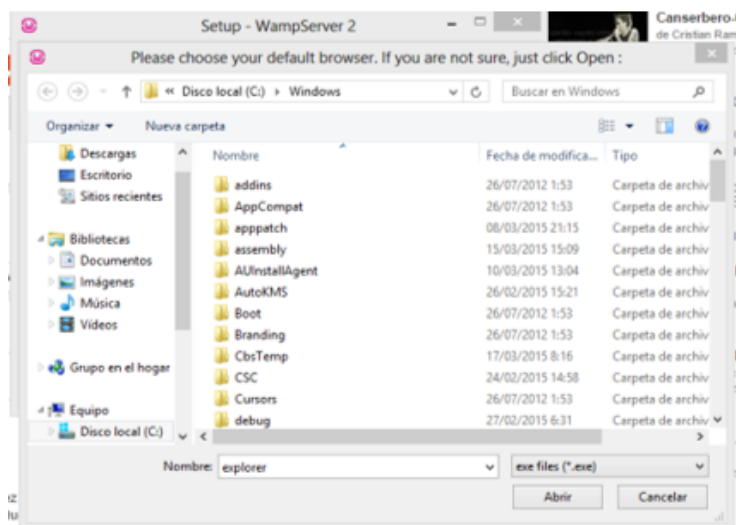


Figura 86. Selección de navegador

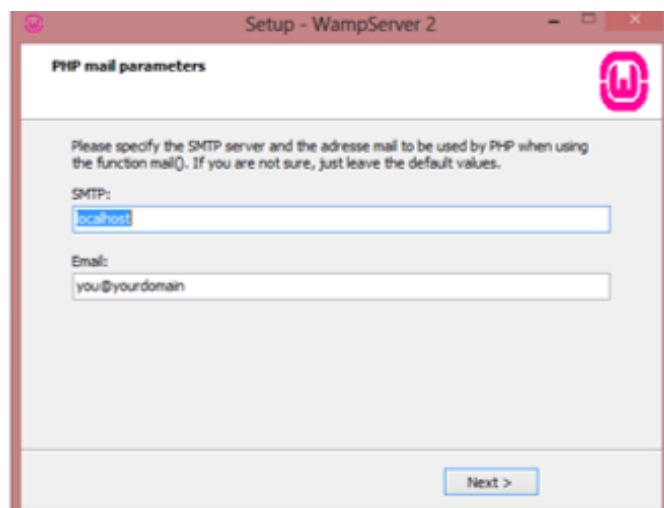


Figura 87. Instalación por defecto

Antes de terminar la instalación nos aparecerá si queremos iniciar el Wampserver, seleccionaremos la opción que nos indica y finalizaremos.

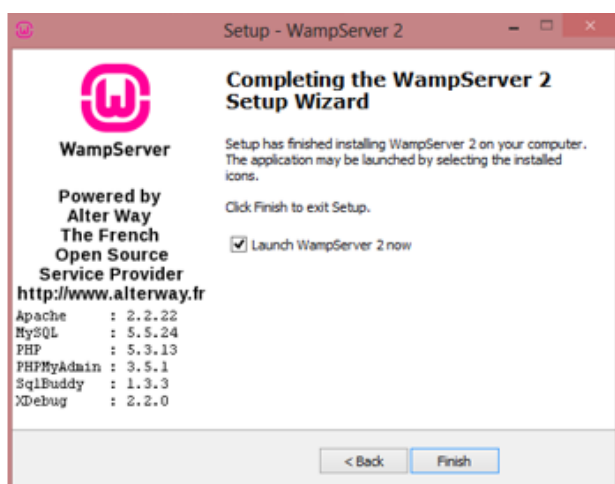


Figura 88. Finalización de la instalación

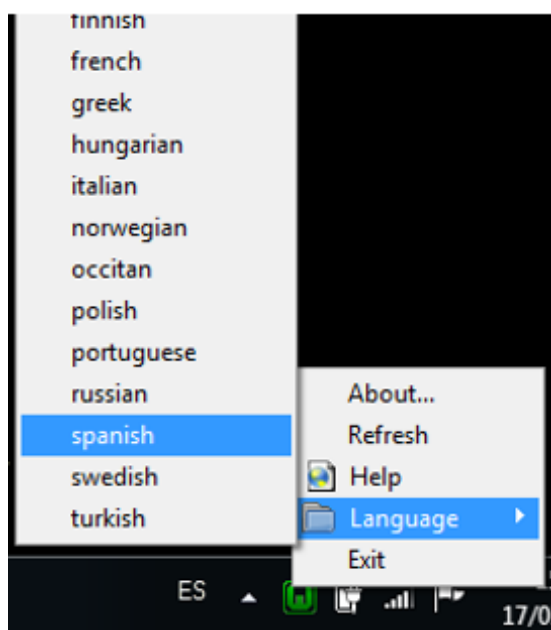


Figura 89. Seleccionaremos el idioma

Instalación del Android Studio

Para una correcta instalación es recomendable proceder a la página principal de Android Studio (<https://developer.android.com/sdk/index.html>) y descargar la aplicación.

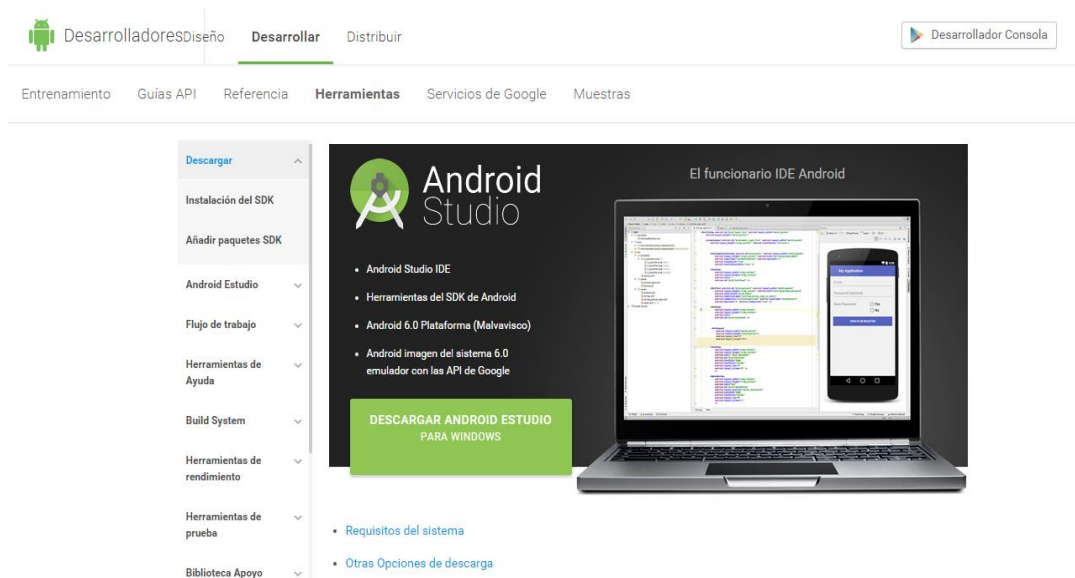


Figura 90. Descarga de Android Studio

Una vez terminada la descarga del instalador procedemos a instalarlo.



Figura 91. Inicio de la instalación

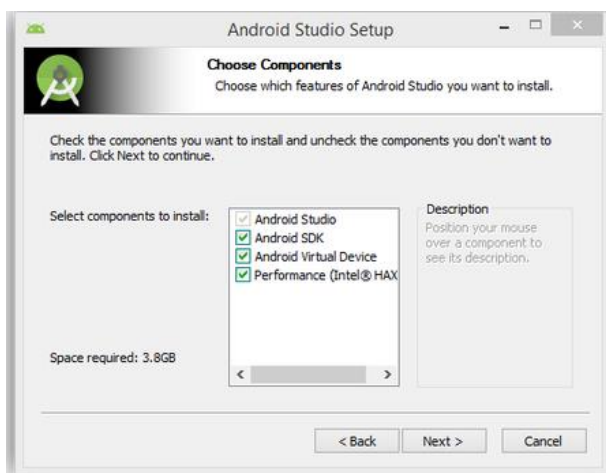


Figura 92. Seleccionamos los componentes que queremos instalar (seleccionar todas las funciones disponibles)

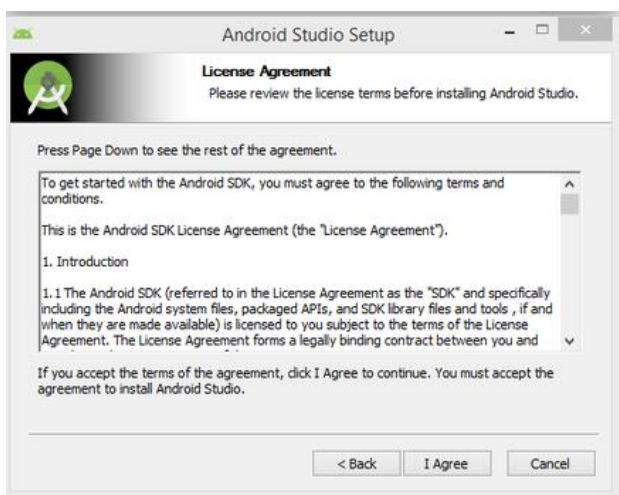


Figura 93. Aceptamos los términos de uso

En el siguiente paso debemos la ruta donde instalaremos el android studio. Debemos elegir una ruta para el programa y otra para el SDK, con bastante espacio ya que la descarga y las actualizaciones de estos necesitan bastante espacio disponible.

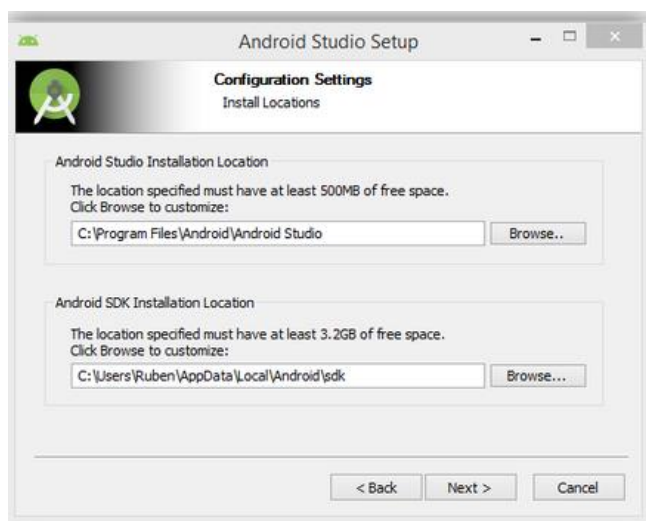


Figura 94. Selección de ruta de instalación

En el siguiente paso elegiremos la cantidad de memoria RAM para el uso de máquinas virtuales y emuladores de Android, con mayor memoria mejor rendimiento tendrá la aplicación pero es preferible dejar con la que nos recomienda el sistema ya que en la actualidad ningún smartphone cuenta con más de 2GB de memoria.



Figura 95: Selección de memoria

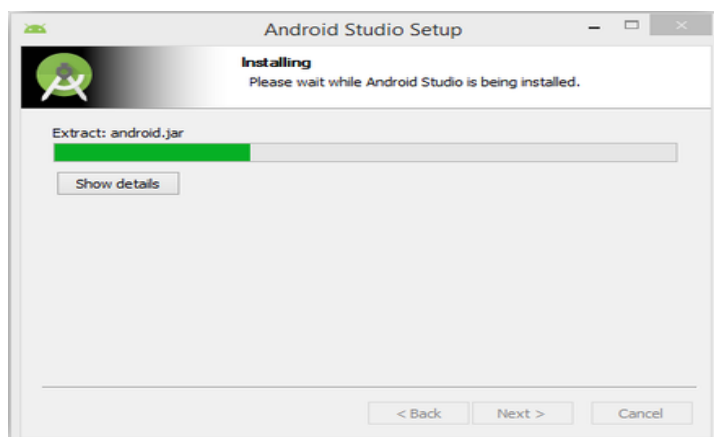


Figura 96: Proceso de instalación

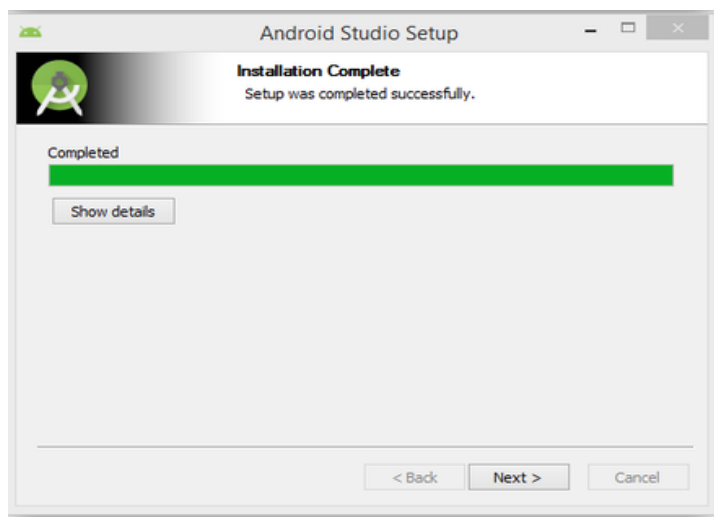


Figura 97. Instalación completa (siguiente)

Una vez terminada la instalación Android se conectara automáticamente a internet para descargar los elementos del SDK necesarios para un correcto funcionamiento.

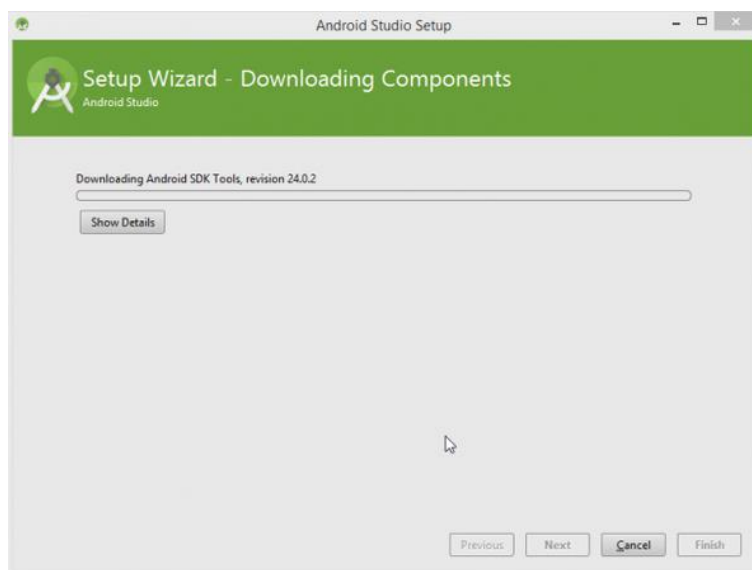


Figura 98. Descarga de componentes SDK

Una vez terminada la descarga estará listo para utilizar la herramienta de programación Android Studio.

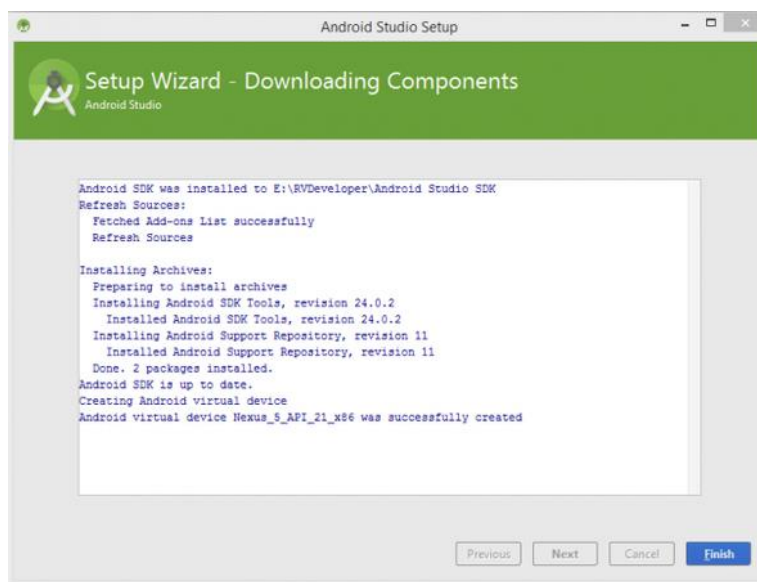


Figura 99. Instalación completa

Anexo A.06. Manual de administrador y usuario

Manual para el administrador

Este manual tiene como objetivo identificar la metodología correcta para utilizar el sistema, ya que veremos los métodos más importantes para poder utilizar de mejor manera y comprensible sea el administrador como el usuario. Entre los detalles mas importantes veremos:

- Ingreso al sistema.
- Registro de usuario.
- Ingreso de un hotel.
- Servicios de un hotel.
- Ingreso de sectores.

Funciones básicas del sistema.

Botón de ingreso el cual permitirá ingresar al sistema.

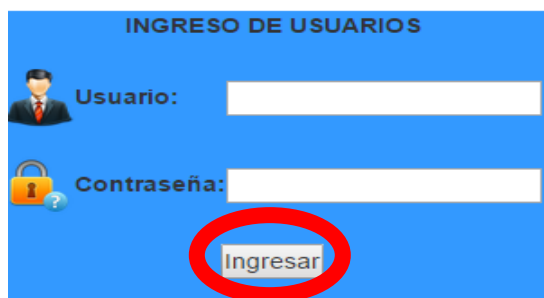


Figura 100. Ingreso al sistema

Este botón permitirá registrar nuevo dato en el sistema

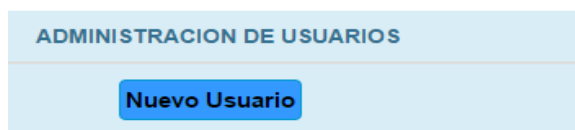


Figura 101. Añadir nuevo

Este botón permitirá modificar un dato no deseado



Figura 102. Botón modificar

Este botón procederá a eliminar un dato no deseado del sistema

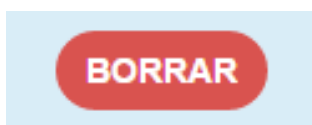


Figura 103. Botón eliminar

Después de haber visto las principales funciones que va a cumplir el sistema procederemos a ingresar al sistema, en este caso se ingresara como administrador ya que es único autorizado para el ingreso, modificación y eliminación de datos.

SISTEMA DE GEO LOCALIZACION HOTELERA

UBICACION DE HOTELES

00:18:19

Un formulario de login con un fondo azul. En la parte superior, el título "INGRESO DE USUARIOS" en letras blancas. Debajo, hay dos campos de texto blancos. El primero está precedido por un ícono de usuario y el texto "Usuario:". El segundo está precedido por un ícono de candado y el texto "Contraseña:". Debajo de los campos, hay un botón rectangular con el texto "Ingresar" en letras blancas.

Figura 104. Ingreso al sistema

Luego de haber ingresado al sistema se procede a realizar los procesos que el administrador crea pertinente, entre ellos podemos destacar los siguientes:

Ingresa un nuevo sector del D.M.Q. para la ubicación de nuevos hoteles.



Figura 105. Clic en el botón de nuevo sector

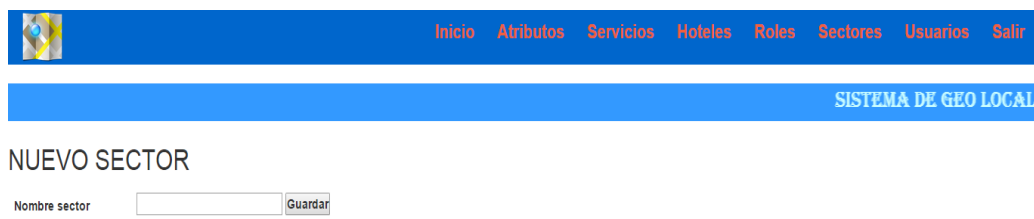


Figura 106. Ingresar nuevo sector

Una vez ingresado los sectores, vamos a ingresar los hoteles dependiendo del sector donde se encuentren, llenando el nombre y ubicamos en el mapa respectivo para obtener las coordenadas y estas utilizarlas en la App.

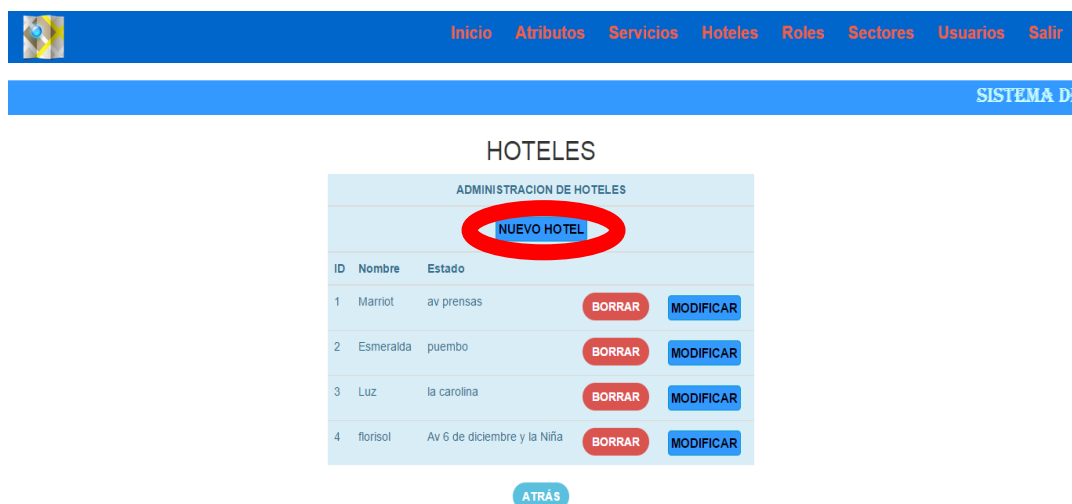


Figura 107. Clic nuevo hotel

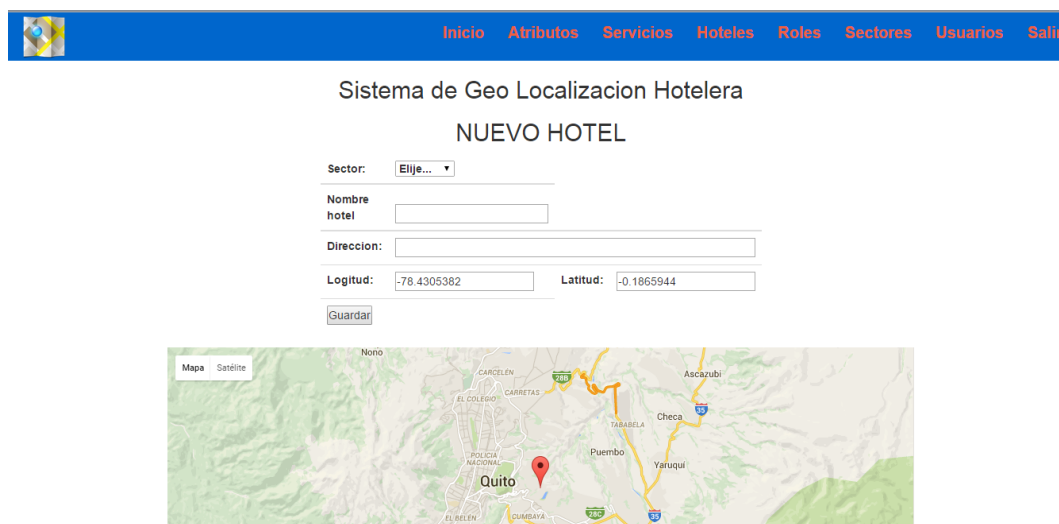


Figura 108. Ingreso de todos los datos del nuevo hotel

Luego de haber ingresado los hoteles respectivos se procederá a ingresar los servicios que el hotel va a dar a los usuarios de la App.

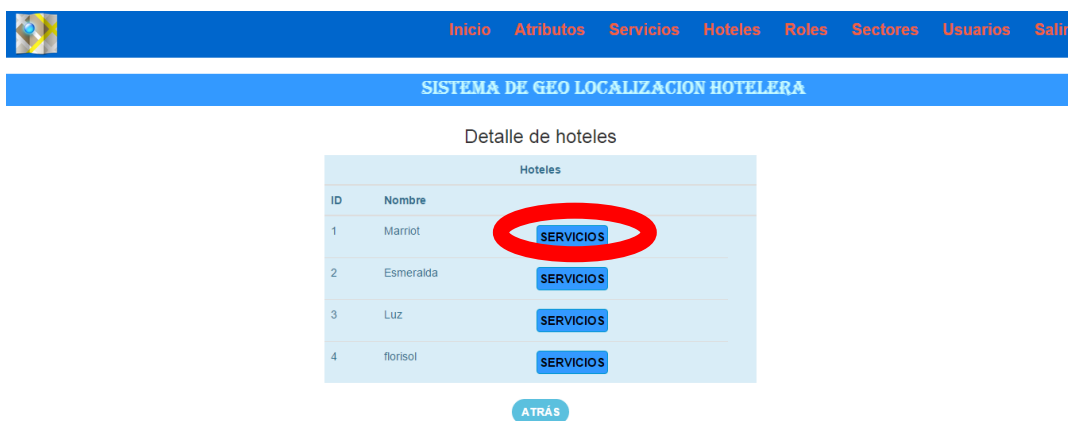


Figura 109. Clic en servicio

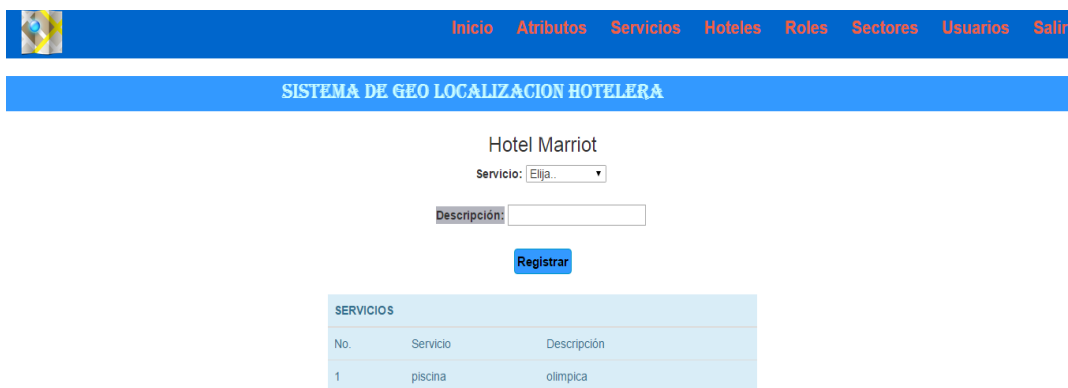


Figura 110. Ingreso detallado de los servicios

Manual de usuario

Como primer paso obtenemos la APK y procedemos a instalar en el Smartphone

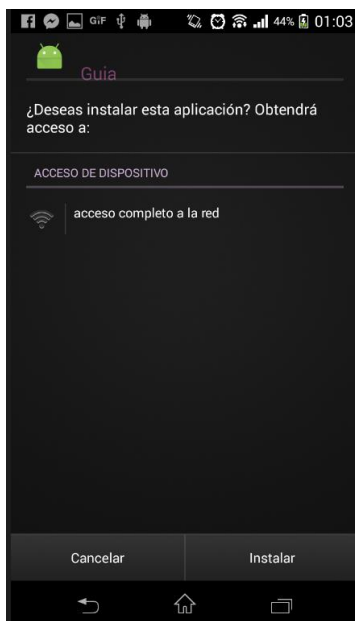


Figura 111. Instalar Apk

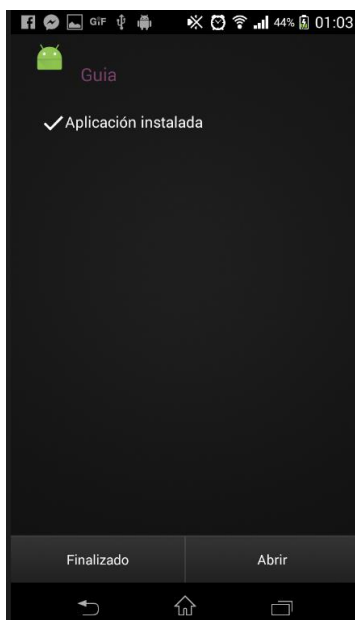


Figura 112. Finaliza la instalación

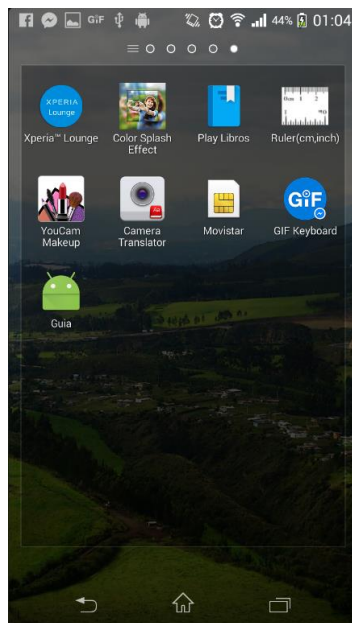


Figura 113. Abrimos la App instalada

Si ya se ha registrado ingresamos el usuario y contraseña caso contrario procedemos a registrarnos.

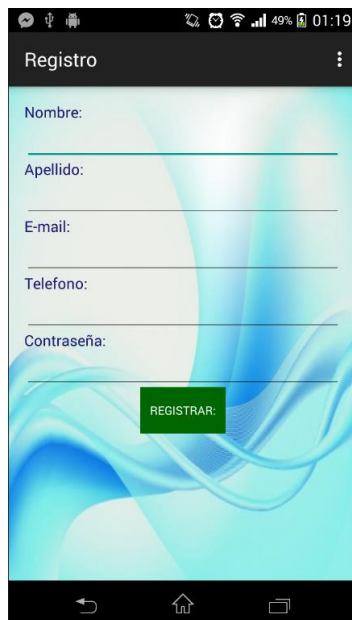


Figura 114. Nuevo registro

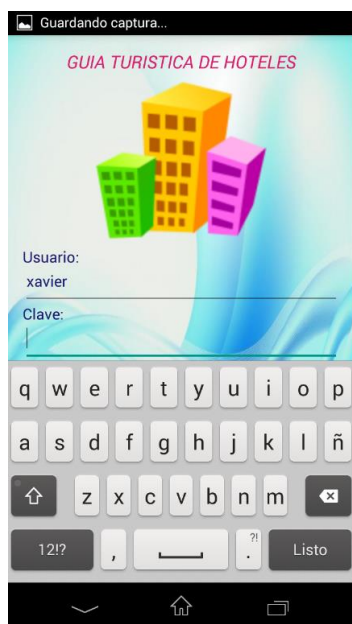


Figura 115. Inicio de sesión

Una vez iniciado la sesión nos aparecerá en el teléfono los sectores de la ciudad de Quito y elegimos cualquiera de ellos para que se nos visualice los hoteles del sector.



Figura 116. Sectores de Quito



Figura 117. Hoteles de un sector

También se puede hacer búsqueda de hoteles por medio de servicios y luego proceder a ubicación del hotel mediante el mapa.

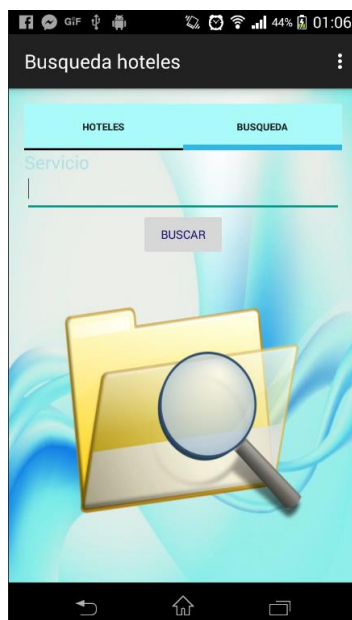


Figura 118. Búsqueda de hoteles

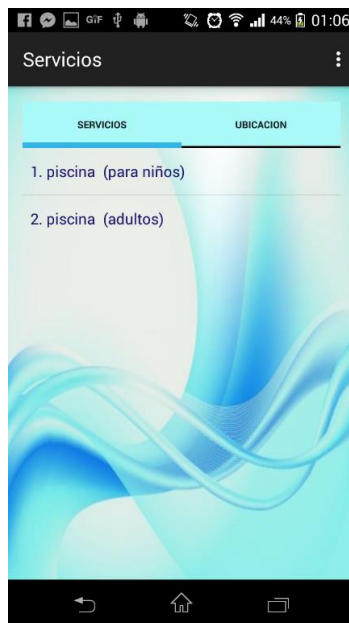


Figura 119. Servicios de un hotel



Figura 120. Como llegar al hotel

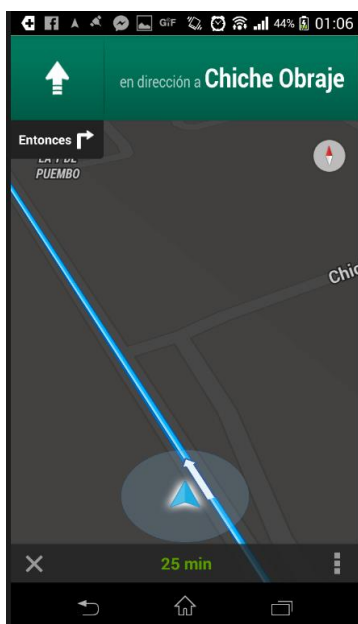


Figura 121. Mapa de ubicación

Anexo A.07. Manual técnico

El siguiente manual tiene la finalidad proporcionar al lector la respectiva lógica y secuencia en la que se desarrolló el sistema, sea este el código de programación y las plataformas de programación entre las cuales se utilizó: Netbeans. MySQL y Android Studio.

Cabe recalcar que la programación se realizó en tres capas mediante PHP utilizando Netbeans para la parte administrativa y para la parte del usuario (App) se utilizó Android Studio.

Contenido

Desarrollo de la interfaz de ingreso al sistema administrativo.

SISTEMA DE GEO LOCALIZACION HOTELERA

UBICACION DE HOTELES

02:14:41

Figura 122. Interfaz de ingreso al sistema

```
?php
this->load->view('header_inicio');
>



Figura 123. Código para ingreso al sistema



MEJORAMIENTO DE UN SISTEMA PARA RASTREO DE HOTELES  
DEFINIDOS EN EL DISTRITO METROPOLITANO DE QUITO MEDIANTE  
UNA APP PARA DISPOSITIVOS MÓVILES EN LA EMPRESA STAFF  
VACATION CIA. LTDA.


```



Figura 124. Interfaz de usuario

```

<td><?php echo $cont; ?></td>
<td><?php echo $value->nombre_usu; ?></td>
<td><?php echo $value->rol_usu; ?></td>
<td>
    <?php echo form_open('usuarios/eliminar_usuario', 'id="form" . $value->id_usu . '''); ?>
    <input name="id_usu" value="<?php echo $value->id_usu; ?>" type="hidden">
    <button type="button" onclick="confirmar_eliminacion(<?php echo $value->id_usu; ?>);" class="btn bt
    <?php echo form_close(); ?>
</td>
<td>
    <?php echo form_open('usuarios/modificar_usuario'); ?>
    <input name="id_usu" value="<?php echo $value->id_usu; ?>" type="hidden">
    <button type="submit" class="btn btn-default">Modificar</button>
    <?php echo form_close(); ?>
</td>
</tr>
<?php
$cont++;
}
?>
</table>

<form>
    <input type="button" value="atrás" name="volver atrás2" class="btn btn-info" onclick="history.back()" />
</form>

</center>

```

Figura 125. Código para visualizar los usuarios



[Inicio](#)
[Atributos](#)
[Servicios](#)
[Hoteles](#)
[Roles](#)
[Sectores](#)
[Usuarios](#)
[Salir](#)

SISTEMA DE GEO LOCALIZACION HOTELERA

NUEVO USUARIO

Datos de Usuario

Tipo:

Elija Usuario ▾

Usuario:

Clave:

Confirmación:

Guardar

Figura 126. Interfaz de ingreso nuevo usuario

```

<center>
<table style="width: 40%;" class="table table-responsive">
  <thead>
    <th colspan="3">Datos de Usuario</th>
  </thead>
  <tbody>
    <tr>
      <td>Tipo: </td>
      <td>
        <select name="tipo" required="true" >
          <option value="">Elija Usuario</option>
          <option value="1">Administrador</option>
          <option value="2">Normal</option>
        </select>
      </td>
      <td> </td>
    </tr>
    <tr>
      <td>Usuario: </td>
      <td><input name="usuario" id="usuario" required="true"></td>
      <td><strong id="mensaje_usuario"></strong></td>
    </tr>
    <tr>
      <td>Clave: </td>
      <td><input name="clave" id="clave" type="password" required="true"></td>
      <td></td>
    </tr>
  </tbody>
</table>

```

Figura 127. Código para ingresar nuevo usuario parte 1

```

<?php echo form_close(); ?>
<script>
    $(function() {
        $('#usuario').keyup(function() {
            var texto = $('#usuario').val();
            var tam = texto.length;
            if (tam < 6) {
                $('#mensaje_usuario').text('Usuario debe tener como mínimo 6 caracteres');
                $('#btn_enviar').attr('disabled', 'disabled');
            } else {
                $('#mensaje_usuario').text('Usuario Correcto');
                $('#btn_enviar').removeAttr('disabled');
            }
        });

        $('#clave').keyup(function() {
            var clave_1 = $('#clave').val();
            var clave_2 = $('#clave_2').val();
            if (clave_1 == clave_2) {
                $('#mensaje_clave').removeAttr('style');
                $('#mensaje_clave').attr('style', 'color: green');
                $('#mensaje_clave').text('Confirmacion Correcta');
                $('#btn_enviar').removeAttr('disabled');
            } else {
                $('#mensaje_clave').removeAttr('style');
                $('#mensaje_clave').attr('style', 'color: red');
                $('#mensaje_clave').text('Confirmacion Incorrecta');
                $('#btn_enviar').attr('disabled', 'disabled');
            }
        });
    });
  
```

Figura 128. Código de validación de contraseña para ingreso de usuario parte 2

```

    $('#clave_2').keyup(function() {
        var clave_1 = $('#clave').val();
        var clave_2 = $('#clave_2').val();
        if (clave_1 == clave_2) {
            $('#mensaje_clave').removeAttr('style');
            $('#mensaje_clave').attr('style', 'color: green');
            $('#mensaje_clave').text('Confirmacion Correcta');
            $('#btn_enviar').removeAttr('disabled');
        } else {
            $('#mensaje_clave').removeAttr('style');
            $('#mensaje_clave').attr('style', 'color: red');
            $('#mensaje_clave').text('Confirmacion Incorrecta');
            $('#btn_enviar').attr('disabled', 'disabled');
        }
    });
  
```

Figura 129. Código de validación de contraseña para ingreso de usuario parte 3

```
function borrar_usuario($id_usu) {
    $this->db->where('id_usu',$id_usu);
    $query= $this->db->delete('usuarios');
    if($query!=null){
        return 1;
    }else{
        return 0;
    }
}

function consultar_datos_de_usuario($id_usu) { ...13 lines }

function modificar_datos_usuario($id_usu,$nombre,$clave) {
    $this->db->where('id_usu',$id_usu);
    $this->db->set('nombre_usu',$nombre);
    $this->db->set('clave_usu',md5($clave));
    $query = $this->db->update('usuarios');
    echo $this->db->last_query();
    if($query!=null){
        return 1;
    }else{
        return 0;
    }
}
```

Figura 130. Método para borrar y modificar usuario

```
function registrar_nuevo_usuario($tipo,$usuario,$clave_encryptada) {
    $this->db->set('rol_usu',$tipo);
    $this->db->set('nombre_usu',$usuario);
    $this->db->set('clave_usu',$clave_encryptada);
    $this->db->set('estado_usu','A');
    $query = $this->db->insert('usuarios');
    // echo $this->db->last_query();
    if($query!=null){
        return 1;
    }else{
        return 0;
    }
}
```

Figura 131. Método para registrar nuevo usuario



Figura 132. Interfaz de hoteles

```
<?php
    $this->load->view('header');
?>
<center>
<h2>HOTELES</h2>
</center>
<?php if( isset($hoteles) && $hoteles!=null){?>
<center>
<table style="width: 40%;" class="table alert-info">
<thead>
<th colspan="5" style="text-align: center;">ADMINISTRACION DE HOTELES</th>
</thead>
<thead>
<th colspan="5" style="text-align: center;">
<?php echo form_open('hoteles/nuevo_hotel');?>
<button type="submit" class="btn btn-info">Nuevo Hotel</button>
<?php echo form_close(); ?>
</th>
</thead>
<thead>
<th>ID</th>
<th>Nombre</th>
<th>Estado</th>
<th></th>
<th></th>
</thead>

<?php
$cont=1;
```

Figura 133. Código para visualizar los hoteles parte 1

```

foreach ($hoteles as $value) { ?>
<tr>
<td><?php echo $count;?></td>
<td><?php echo $value->nombre_hotel;?></td>
<td><?php echo $value->direccion_fisica;?></td>
<td>
<?php if($rol=="1"){?>
<?php echo form_open('hoteles/eliminar_hotel','id="form_'.$value->id_hotel.'");?>
<input name="id_hotel" value="<?php echo $value->id_hotel;?>" type="hidden">
<button type="button" onclick="confirmar_eliminacion(<?php echo $value->id_hotel;?>);" class="btn btn-danger">Eliminar
<?php echo form_close(); ?>
</td>
<td>
<?php echo form_open('hoteles/modificar_hotel');?>
<input name="id_hotel" value="<?php echo $value->id_hotel;?>" type="hidden">
<button type="submit" class="btn btn-warning">Modificar</button>
<?php echo form_close(); ?>
</td>
<?php ?>
</tr>
<?php $count++;}


} else{ ?>
<h2>No se encontraron sectores activos</h2>
<?php ?>
</table>
<form>
<input type="button" value="atrás" name="volver atrás2" class="btn btn-info" onclick="history.back()" />
</form>
</center>

```

Figura 134. Código para visualizar los hoteles parte 2

```
function confirmar_eliminacion(i){
    var confirmar=confirm('¿ Está seguro de eliminar este Hotel ?');
    if(confirmar){
        $(function (){
            $('#form_'+i).submit();
        });
    }
}
```

Figura 135. Código para confirmar eliminar un hotel



[Inicio](#)
[Atributos](#)
[Servicios](#)
[Hoteles](#)
[Roles](#)
[Sectores](#)
[Usuarios](#)
[Salir](#)

Sistema de Geo Localizacion Hotelera

NUEVO HOTEL

Sector:

Nombre hotel:

Direccion:

Logitud: Latitud:




Figura 136. Interfaz de ingreso nuevo hotel

```

function modificar_datos_hotel($id_hotel,$data) {
    $this->db->where('id_hotel',$id_hotel);
    $this->db->set('nombre_hotel',$data['nombre_hotel']);
    $query = $this->db->update('hoteles');
    if($query!=null){
        $data['id_hotel'] = $id_hotel;
        $this->actualizar_direccion($data);
        return 1;
    }else{
        return 0;
    }
}

function registrar_hotel($data) {
    $this->db->set('nombre_hotel',$data['nombre_hotel']);
    $this->db->set('id_sector',$data['sectore']);
    $this->db->set('estado_hotel','A');
    $query = $this->db->insert('hoteles');
    if($query!=null){
        $q = $this->db->query('select last_insert_id() as last;');
        foreach ($q->result() as $row) {
            $lastid = $row->last;
            $data['id_hotel'] = $lastid;
        }
        $this->registra_direccion($data);
        return 1;
    }else{
        return 0;
    }
}

```

Figura 137. Código para modificar e ingresar nuevo hotel


```
function verificar_existencia_hotel($nombre) {
    $this->db->where('nombre_hotel',$nombre);
    $this->db->select('*');
    $query= $this->db->get('hoteles');
    // echo $this->db->last_query();
    if($query->num_rows()>0){
        return true;
    }else{
        return false;
    }
}

function borrar_hotel($id_hotel) {
    $this->db->where('id_hotel',$id_hotel);
    $this->db->set('estado_hotel','I');
    $query= $this->db->update('hoteles');
    if($query!=null){
        return 1;
    }else{
        return 0;
    }
}
```

Figura 138. Código para verificar si el hotel existe al momento de ingresar nuevo y borrar hotel

A continuación se dará a conocer el manual técnico para android es decir para la parte del usuario, el mismo que está compuesto de dos formas de programación: unos que es para el diseño de la interfaz y la otra en Java para hacer los diferentes métodos que se requiera.

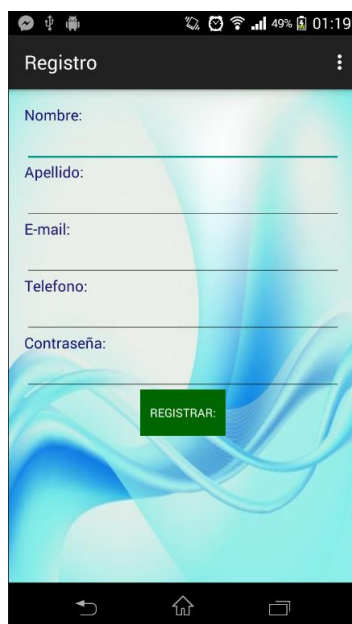


Figura 139. Interfaz de registro nuevo usuario en Androi

```

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:text="Nombre:"
    android:textColor="#1A176F"
    android:id="@+id/lbl_nombre" />

<EditText
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:inputType="textPersonName"
    android:text=""
    android:ems="10"
    android:id="@+id/edit_nombre" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:text="Apellido:"
    android:textColor="#1A176F"
    android:id="@+id/lbl_apellido" />

<EditText
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:inputType="textPersonName"
    android:text=""
  
```

Figura 140. Código de interfaz, registro de nuevo usuario en Android

```
public class Registro_usuarios extends ActionBarActivity {

    EditText nombres, apellidos, email, telefono, clave;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_registro_usuarios);
    }

    public void recogerDatos(View v){
        nombres=(EditText)findViewById(R.id.edit_nombre);
        apellidos=(EditText)findViewById(R.id.edit_apellido);
        email=(EditText)findViewById(R.id.edit_email);
        telefono=(EditText)findViewById(R.id.edit_telefono);
        clave=(EditText)findViewById(R.id.edit_contrasena);

        String nombres_cam =nombres.getText().toString();
        String apellidos_cam =apellidos.getText().toString();
        String email_cam =email.getText().toString();
        String telefono_cam =telefono.getText().toString();
        String contrasena_cam =clave.getText().toString();

        this.registrar_usuario(nombres_cam,apellidos_cam,email_cam,telefono_cam,contrasena_cam);
    }
}
```

Figura 141. Código Java de registro nuevo usuario en Android parte 1

```
public void registrar_usuario(final String nombres_cam, final String apellidos_cam, final String email_cam,
                             final String telefono_cam, final String contrasena_cam){

    Thread nt = new Thread() {
        @Override
        public void run() {
            String url="http://katonproyect.tk/geo_localizacion/hoteles_remoto/registrar_usuario_remoto";
            final String Resultado=sendByPost(url,nombres_cam,apellidos_cam,email_cam,telefono_cam,contrasena_cam);
            try {
                runOnUiThread(new Runnable() {
                    @Override
                    public void run() {
                        if (Resultado.equals("1")){
                            mostrarMensaje("Registro Exitoso");
                            SharedPreferences prefs =
                                getSharedPreferences("DatosUsuario", Context.MODE_PRIVATE);

                            SharedPreferences.Editor editor = prefs.edit();
                            editor.putString("Nombres", nombres_cam);
                            editor.putString("Apellidos", apellidos_cam);
                            editor.putString("Email", email_cam);
                            editor.putString("Telefono", telefono_cam);
                            editor.putString("Clave", contrasena_cam);
                            editor.commit();
                            ir_a_perfil();
                        }
                        else {
                            mostrarMensaje("No Registrado");
                        }
                    }
                });
            }
        }
    };
}
```

Figura 142. Código Java de registro nuevo usuario en Android parte 2

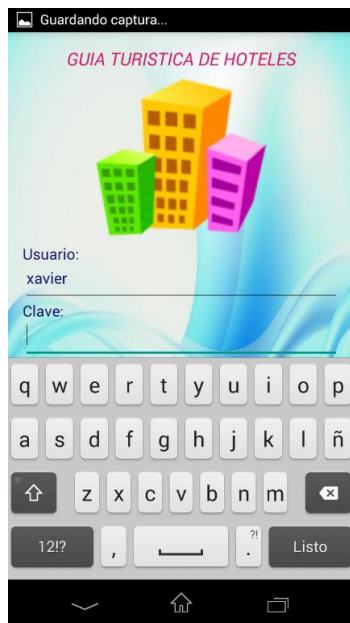


Figura 143. Interfaz de ingreso al sistema en android

```

<EditText
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/txt_usuario" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:text="Clave:"
    android:textColor="#1A176F"
    android:id="@+id/textView2" />

<EditText
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:inputType="textPassword"
    android:ems="10"
    android:id="@+id/txt_clave" />

<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Ingresar"
    android:id="@+id/btnIngresar"
    android:layout_gravity="center_horizontal"
    android:onClick="login"/>
  
```

Figura 144. Código de interfaz para el ingreso al sistema de android

```
public String[] obtDatosJSON(String response){
    String texto="";
    String[] usuario=null;
    try {
        JSONArray json= new JSONArray(response);
        usuario = new String[5];
        usuario[0]=json.getJSONObject(0).getString("nombre_usu");
        usuario[1]=json.getJSONObject(0).getString("apellido_usu");
        usuario[2]=json.getJSONObject(0).getString("telf_usu");
        usuario[3]=json.getJSONObject(0).getString("email_usu");
        usuario[4]=json.getJSONObject(0).getString("clave_usu");

    } catch (Exception e) {
        mostrarMensaje(response);
        mostrarMensaje(e+" Error de Coneccion !!!");
    }
    return usuario;
}
```

Figura 145. Código Java para el ingreso al sistema de android parte 1

```
public void login(String[]usuario){
    if(usuario!=null){
        mostrarMensaje("Bienvenido Usuario");
        SharedPreferences prefs =
            getSharedPreferences("DatosUsuario", Context.MODE_PRIVATE);
        SharedPreferences.Editor editor = prefs.edit();
        editor.putString("Nombre", usuario[0]);
        editor.putString("Apellido", usuario[1]);
        editor.putString("Email", usuario[2]);
        editor.putString("Telefono", usuario[3]);
        editor.putString("Clave", usuario[4]);
        editor.commit();
        MostrarSectores();
    }else{
        mostrarMensaje("Credenciales Erroneas");
    }
}
```

Figura 146. Código Java para el ingreso al sistema de android parte 2

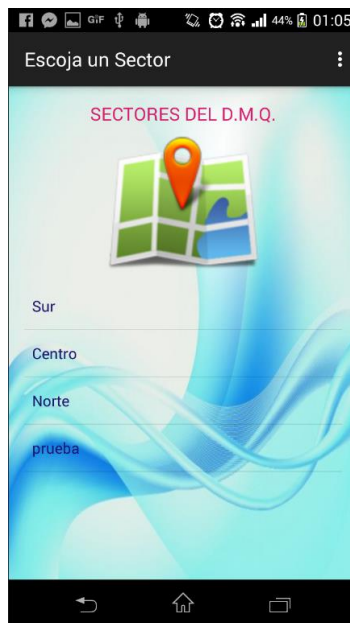


Figura 147. Interfaz de los sectores de Quito, android

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceLarge"
    android:text="SECTORES DEL D.M.Q."
    android:id="@+id/Titulo_sectores"
    android:layout_gravity="center_horizontal"
    android:textColor="#D71D6E"
    android:layout_weight="0.03" />

<ImageView
    android:layout_width="252dp"
    android:layout_height="wrap_content"
    android:id="@+id/imgsectores"
    android:layout_gravity="center_horizontal"
    android:layout_weight="0.51"
    android:src="@drawable/serctor1" />

<ListView
    android:layout_width="match_parent"
    android:layout_height="261dp"
    android:id="@+id/listSectores"
    android:cacheColorHint="@color/link_text_material_light"
    android:layout_gravity="center_horizontal" />

</LinearLayout>
</RelativeLayout>
```

Figura 148. Código de interfaz de los sectores en android

```
public void mostrarMensaje(String msj){
    Toast.makeText(getApplicationContext(), msj,
        Toast.LENGTH_SHORT).show();
}

private void cargarSectores(String[] Sectores){
    try {
        listViewSectores = (ListView) findViewById(R.id.listViewSectores);
        listViewSectores.setAdapter(new ArrayAdapter<String>(this, android.R.layout.simple_list_item_1, Sectores));
        listViewSectores.setOnItemClickListener(new AdapterView.OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
                mostrarMensaje(listViewSectores.getItemAtPosition(position).toString());
                iniciar_actividad(listViewSectores.getItemAtPosition(position).toString());
            }
        });
    } catch (Exception ex){
    }
}
```

Figura 149. Código Java para la carga de sectores en android

```
public void obtener_datosDB(){
    Thread nt = new Thread() {
        @Override
        public void run() {
            final String Resultado=sendByPost("http://katonproyect.tk/geo_localizacion/hoteles_remoto/consultar_sectores");
            try {
                runOnUiThread(new Runnable() {
                    @Override
                    public void run() { cargarSectores(obtDatosJSON(Resultado)); }
                });
            } catch (Exception e) {
                mostrarMensaje("Error " + e);
            }
        }
    };
    nt.start();
}
```

Figura 150. Código Java para obtener datos de los sectores en android



Figura 151. Interfaz de la ubicación de los hoteles, android

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceLarge"
    android:textColor="#C726A7"
    android:text="¿Como desea llegar al hotel?"
    android:id="@+id/textllegar"
    android:layout_gravity="center_horizontal"
    android:layout_weight="0.07" />

<ImageView
    android:layout_width="250dp"
    android:layout_height="10dp"
    android:id="@+id/imageView5"
    android:background="@drawable/ubicacion"
    android:layout_gravity="center_horizontal"
    android:layout_weight="0.53" />

<ImageView
    android:layout_width="100dp"
    android:layout_height="50dp"
    android:id="@+id/imageView3"
    android:background="@drawable/car"
    android:layout_gravity="center_horizontal" />
```

Figura 152. Código de interfaz para la ubicación de los hoteles parte 1


```

<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Conduciendo "
    android:id="@+id/btn_cond"
    android:onClick="Mostrar_mapa_conduciendo"
    android:layout_gravity="center_horizontal" />

<ImageView
    android:layout_width="80dp"
    android:layout_height="50dp"
    android:id="@+id/imageView6"
    android:background="@drawable/caminando"
    android:layout_gravity="center_horizontal"
    android:layout_weight="0.07" />

<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Caminando"
    android:id="@+id/btn_cam"
    android:layout_gravity="center_horizontal" />
  
```

Figura 153. Código de interfaz para la ubicación de los hoteles parte 2

```

public void obtener_datos_de_las_cordenadas(final String metodo) {
    Thread nt = new Thread() {
        @Override
        public void run() {
            final String Resultado = sendByPost("http://katonproyect.tk/geo_localizacion/hoteles_remoto/consultar_coordenadas_por_h");
            try {
                runOnUiThread(new Runnable() {
                    @Override
                    public void run() {
                        recibir_coordenadas(obtDatosJSON_cordenadas(Resultado), metodo);
                    }
                });
            } catch (Exception e) {
                mostrarMensaje("Error " + e);
            }
        }
    };
    nt.start();
}
  
```

Figura 154. Código Java para la obtención de datos (coordenadas), android

```
public void recibir_coordenadas (String coord, String metodo){
    try {
        String[] coordenadas=new String[2];
        coordenadas[0]=coord.split(",")[0];
        coordenadas[1]=coord.split(",")[1];
        Uri gmmIntentUri = Uri.parse("google.navigation:q="+coord.split(",")[1]+" "+coord.split(",")[0]+"&mode="+metodo);
        Intent mapIntent = new Intent(Intent.ACTION_VIEW, gmmIntentUri);
        mapIntent.setPackage("com.google.android.apps.maps");
        startActivity(mapIntent);
    }catch (Exception e){
        mostrarMensaje("intent: "+e);
    }
}
```

Figura 155. Código Java para recibir coordenadas y mostrar el mapa, android

```
public void cerrarSession(){
    SharedPreferences prefs =
        getSharedPreferences("DatosUsuario", Context.MODE_PRIVATE);
    SharedPreferences.Editor editor = prefs.edit();
    editor.putString("Nombre", "");
    editor.putString("Apellido", "");
    editor.putString("Email", "");
    editor.putString("Telefono", "");
    editor.putString("Clave", "");
    editor.commit();
    redireccionar_login();
}

public void redireccionar_login(){
    Intent intent = new Intent(this, Inicio.class);
    startActivity(intent);
}
```

Figura 156. Código Java para cerrar sesión en android y re direccionar al login

Script de la base de datos.

```
CREATE DATABASE IF NOT EXISTS `geo_localizacion` /*!40100 DEFAULT  
CHARACTER SET utf8 */;
```

```
USE `geo_localizacion`;
```

```
-- MySQL dump 10.13 Distrib 5.6.13, for Win32 (x86)
```

```
--
```

```
-- Host: localhost Database: geo_localizacion
```

```
-----
```

```
-- Server version 5.6.14
```

```
/*!40101 SET
```

```
@OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
```

```
/*!40101 SET
```

```
@OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
```

```
/*!40101 SET
```

```
@OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
```

```
/*!40101 SET NAMES utf8 */;
```

```
/*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
```

```
/*!40103 SET TIME_ZONE='+00:00' */;
```

```
/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS,
```

```
UNIQUE_CHECKS=0 */;
```

```
/*!40014 SET
```

```
@OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
```

```
FOREIGN_KEY_CHECKS=0 */;
```

```
/*!40101 SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;

/*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;

--

-- Table structure for table `atributos`

--

DROP TABLE IF EXISTS `atributos`;

/*!40101 SET @saved_cs_client = @@character_set_client */;

/*!40101 SET character_set_client = utf8 */;

CREATE TABLE `atributos` (
  `id_atributo` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `descripcion_atributo` varchar(100) NOT NULL,
  `estado_atributo` varchar(45) DEFAULT NULL,
  PRIMARY KEY (`id_atributo`)
) ENGINE=InnoDB AUTO_INCREMENT=3 DEFAULT CHARSET=utf8;

/*!40101 SET character_set_client = @saved_cs_client */;

--

-- Dumping data for table `atributos`

--

LOCK TABLES `atributos` WRITE;

/*!40000 ALTER TABLE `atributos` DISABLE KEYS */;
```

```
INSERT INTO `atributos` VALUES (1,'piscina','A'),(2,'cocina','A');

/*!40000 ALTER TABLE `atributos` ENABLE KEYS */;

UNLOCK TABLES;

--

-- Table structure for table `detalle_hotel`

--

DROP TABLE IF EXISTS `detalle_hotel`;

/*!40101 SET @saved_cs_client = @@character_set_client */;

/*!40101 SET character_set_client = utf8 */;

CREATE TABLE `detalle_hotel` (

  `id_detalleHotel` int(10) unsigned NOT NULL AUTO_INCREMENT,

  `id_atributo` int(10) unsigned NOT NULL,

  `id_hotel` int(10) unsigned NOT NULL,

  `descripcion_detalle` varchar(200) NOT NULL,

  `estado_detalleHotel` set('A','T') NOT NULL,

  PRIMARY KEY (`id_detalleHotel`),

  KEY `fk_detalle_hotel_idx` (`id_hotel`),

  KEY `fk_detalle_atributo_idx` (`id_atributo`),

  CONSTRAINT `fk_detalle_atributo` FOREIGN KEY (`id_atributo`)

REFERENCES `atributos` (`id_atributo`) ON DELETE NO ACTION ON UPDATE

NO ACTION,

  CONSTRAINT `fk_detalle_hotel` FOREIGN KEY (`id_hotel`) REFERENCES

`hoteles` (`id_hotel`) ON DELETE NO ACTION ON UPDATE NO ACTION

) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=utf8;
```

```
/*!40101 SET character_set_client = @saved_cs_client */;

--

-- Dumping data for table `detalle_hotel`

--

LOCK TABLES `detalle_hotel` WRITE;

/*!40000 ALTER TABLE `detalle_hotel` DISABLE KEYS */;

INSERT INTO `detalle_hotel` VALUES (1,1,1,'olimpica','A'),(2,1,20,'para
niños','A'),(3,1,20,'adultos','A');

/*!40000 ALTER TABLE `detalle_hotel` ENABLE KEYS */;

UNLOCK TABLES;

--

-- Table structure for table `direccion`

--

DROP TABLE IF EXISTS `direccion`;

/*!40101 SET @saved_cs_client = @@character_set_client */;

/*!40101 SET character_set_client = utf8 */;

CREATE TABLE `direccion` (

  `id_direccion` int(10) unsigned NOT NULL AUTO_INCREMENT,

  `id_hotel` int(10) unsigned NOT NULL,

  `direccion_fisica` varchar(80) NOT NULL,

  `direccion_matematica` varchar(45) DEFAULT NULL,

  `cordenada_x` varchar(30) NOT NULL,

  `cordenada_y` varchar(30) NOT NULL,
```

```
`cordenada_z` int(11) DEFAULT '16',  
  
PRIMARY KEY (`id_direccion`),  
  
KEY `fk_direccion_hotel_idx` (`id_hotel`),  
  
CONSTRAINT `fk_direccion_hotel` FOREIGN KEY (`id_hotel`) REFERENCES  
`hoteles` (`id_hotel`) ON DELETE NO ACTION ON UPDATE NO ACTION  
) ENGINE=InnoDB AUTO_INCREMENT=11 DEFAULT CHARSET=utf8;  
  
/*!40101 SET character_set_client = @saved_cs_client */;  
  
--  
  
-- Dumping data for table `direccion`  
  
--  
  
LOCK TABLES `direccion` WRITE;  
  
/*!40000 ALTER TABLE `direccion` DISABLE KEYS */;  
  
INSERT INTO `direccion` VALUES (1,1,'av prensas',-78.4305382,-0.1865944,'-  
78.4305382',-0.1865944,11),(6,16,'Condado',-78.4305382,-0.1865944,'-  
78.4305382',-0.1865944,11),(7,17,'centro historico',-78.51688387260742,-  
0.21680661478610905',-78.51688387260742',-0.21680661478610905',11),(8,18,'El  
ejido',-78.49615576134033,-0.20891024594496815',-78.49615576134033',-  
0.20891024594496815',11),(9,19,'condado',-78.48418238029785,-  
0.14994485469965663',-78.48418238029785',-  
0.14994485469965663',11),(10,20,'Av 6 de diciembre y la niña',-  
78.48521234855957,-0.19987662347298296',-78.48521234855957',-  
0.19987662347298296',11);  
  
/*!40000 ALTER TABLE `direccion` ENABLE KEYS */;
```

UNLOCK TABLES;

--

-- Table structure for table `hoteles`

--

DROP TABLE IF EXISTS `hoteles`;

/*!40101 SET @saved_cs_client = @@character_set_client */;

/*!40101 SET character_set_client = utf8 */;

CREATE TABLE `hoteles` (

 `id_hotel` int(10) unsigned NOT NULL AUTO_INCREMENT,

 `id_sector` int(10) unsigned NOT NULL,

 `nombre_hotel` varchar(50) NOT NULL,

 `estado_hotel` set('A','I') NOT NULL,

 PRIMARY KEY (`id_hotel`),

 KEY `fk_hoteles_sectores_idx` (`id_sector`),

 CONSTRAINT `fk_hoteles_sectores` FOREIGN KEY (`id_sector`)

REFERENCES `sectores` (`id_sector`) ON DELETE NO ACTION ON UPDATE

NO ACTION

) ENGINE=InnoDB AUTO_INCREMENT=21 DEFAULT CHARSET=utf8;

/*!40101 SET character_set_client = @saved_cs_client */;

--

-- Dumping data for table `hoteles`

--


```
LOCK TABLES `hoteles` WRITE;

/*!40000 ALTER TABLE `hoteles` DISABLE KEYS */;

INSERT INTO `hoteles` VALUES

(1,1,'Marriot','A'),(16,3,'Esmeralda','A'),(17,1,'pelaverde','T'),(18,2,'Luz','A'),(19,3,'condado','T'),(20,2,'florisol','A');

/*!40000 ALTER TABLE `hoteles` ENABLE KEYS */;

UNLOCK TABLES;

--

-- Table structure for table `roles`

--

DROP TABLE IF EXISTS `roles`;

/*!40101 SET @saved_cs_client = @@character_set_client */;

/*!40101 SET character_set_client = utf8 */;

CREATE TABLE `roles` (

  `id_rol` int(10) unsigned NOT NULL AUTO_INCREMENT,

  `descripcion_rol` varchar(25) DEFAULT NULL,

  `estado_rol` varchar(10) DEFAULT NULL,

  PRIMARY KEY (`id_rol`)

) ENGINE=InnoDB AUTO_INCREMENT=16 DEFAULT CHARSET=utf8;

/*!40101 SET character_set_client = @saved_cs_client */;

--

-- Dumping data for table `roles`

--
```

```
LOCK TABLES `roles` WRITE;

/*!40000 ALTER TABLE `roles` DISABLE KEYS */;

INSERT INTO `roles` VALUES

(1,'Administrador','A'),(2,'Usuario','A'),(3,'nuevo','T'),(4,"','T'),(5,"','T'),(6,'nuevo','T'),(7,'
nuevo','T'),(8,'nuevo','T'),(9,'nuevo2','T'),(10,'nuevo2','T'),(11,'nuevo2','T'),(12,'test2','T'),(
13,'test3','T'),(14,'nuevo3','T'),(15,'Usuario2','T');

/*!40000 ALTER TABLE `roles` ENABLE KEYS */;

UNLOCK TABLES;

--

-- Table structure for table `sectores`

--

DROP TABLE IF EXISTS `sectores`;

/*!40101 SET @saved_cs_client = @@character_set_client */;

/*!40101 SET character_set_client = utf8 */;

CREATE TABLE `sectores` (

  `id_sector` int(10) unsigned NOT NULL AUTO_INCREMENT,

  `nombre_sector` varchar(100) NOT NULL,

  `estado_sector` set('A','T') NOT NULL,

  PRIMARY KEY (`id_sector`)

) ENGINE=InnoDB AUTO_INCREMENT=17 DEFAULT CHARSET=utf8;

/*!40101 SET character_set_client = @saved_cs_client */;

--

-- Dumping data for table `sectores`
```

```
--

LOCK TABLES `sectores` WRITE;

/*!40000 ALTER TABLE `sectores` DISABLE KEYS */;

INSERT INTO `sectores` VALUES

(1,'Sur','A'),(2,'Centro','A'),(3,'Norte','A'),(10,'Sector X','T'),(11,'Sector

X','T'),(12,'Sector X','T'),(13,'Sector X','T'),(14,',','T'),(15,'Sector X','T'),(16,'prueba','A');

/*!40000 ALTER TABLE `sectores` ENABLE KEYS */;

UNLOCK TABLES;

--

-- Table structure for table `sesiones`

--

DROP TABLE IF EXISTS `sesiones`;

/*!40101 SET @saved_cs_client = @@character_set_client */;

/*!40101 SET character_set_client = utf8 */;

CREATE TABLE `sesiones` (

  `id_sesiones` int(10) unsigned NOT NULL AUTO_INCREMENT,

  `descripcion_sesiones` varchar(200) NOT NULL,

  PRIMARY KEY (`id_sesiones`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

/*!40101 SET character_set_client = @saved_cs_client */;

--

-- Dumping data for table `sesiones`

--

LOCK TABLES `sesiones` WRITE;
```

```
/*!40000 ALTER TABLE `sesiones` DISABLE KEYS */;

/*!40000 ALTER TABLE `sesiones` ENABLE KEYS */;

UNLOCK TABLES;

--

-- Table structure for table `usuarios`

--

DROP TABLE IF EXISTS `usuarios`;

/*!40101 SET @saved_cs_client = @@character_set_client */;

/*!40101 SET character_set_client = utf8 */;

CREATE TABLE `usuarios` (

  `id_usu` int(10) unsigned NOT NULL AUTO_INCREMENT,

  `rol_usu` int(10) unsigned NOT NULL,

  `nombre_usu` varchar(20) NOT NULL,

  `apellido_usu` varchar(20) NOT NULL,

  `telf_usu` varchar(20) DEFAULT NULL,

  `email_usu` varchar(50) DEFAULT NULL,

  `clave_usu` varchar(200) NOT NULL,

  `estado_usu` set('A','T') NOT NULL,

  PRIMARY KEY (`id_usu`),

  KEY `fk_usuario_rol_idx` (`rol_usu`),

  CONSTRAINT `fk_usuario_rol` FOREIGN KEY (`rol_usu`) REFERENCES

`roles` (`id_rol`) ON DELETE NO ACTION ON UPDATE NO ACTION

) ENGINE=InnoDB AUTO_INCREMENT=10 DEFAULT CHARSET=utf8;

/*!40101 SET character_set_client = @saved_cs_client */;
```

```
--  
  
-- Dumping data for table `usuarios`  
  
--  
  
LOCK TABLES `usuarios` WRITE;  
  
/*!40000 ALTER TABLE `usuarios` DISABLE KEYS */;  
  
INSERT INTO `usuarios` VALUES  
  
(1,1,'xavier',' ',NULL,NULL,'81dc9bdb52d04dc20036dbd8313ed055','A'),(3,2,'santy',  
  
' ',NULL,NULL,'123','A'),(5,2,'andrea',' ',NULL,NULL,'121212','A'),(6,2,'vinicio',' ',N  
ULL,NULL,'202cb962ac59075b964b07152d234b70','A'),(7,2,'Pamelas',' ',NULL,NU  
LL,'4d62ba1451229575f5496c95184d955d','A'),(9,2,'GAby','Paez','90943','jaksja@g  
mail.com','6fb42da0e32e07b61c9f0251fe627a9c','A');  
  
/*!40000 ALTER TABLE `usuarios` ENABLE KEYS */;  
  
UNLOCK TABLES;  
  
--  
  
-- Temporary table structure for view `vihotel`  
  
--  
  
DROP TABLE IF EXISTS `vihotel`;  
  
/*!50001 DROP VIEW IF EXISTS `vihotel` */;  
  
SET @saved_cs_client = @@character_set_client;  
  
SET character_set_client = utf8;  
  
/*!50001 CREATE TABLE `vihotel` (  
  
  `cod_sec` tinyint NOT NULL,  
  
  `nombre_sector` tinyint NOT NULL,  
  
  `estado_sector` tinyint NOT NULL,
```

```
`id_hotel` tinyint NOT NULL,  
  
`id_sector` tinyint NOT NULL,  
  
`nombre_hotel` tinyint NOT NULL,  
  
`estado_hotel` tinyint NOT NULL  
  
) ENGINE=MyISAM *;/  
  
SET character_set_client = @saved_cs_client;  
  
--  
  
-- Dumping events for database 'geo_localizacion'  
  
--  
  
--  
  
-- Dumping routines for database 'geo_localizacion'  
  
--  
  
--  
  
-- Final view structure for view `vihotel`  
  
--  
  
/!*!50001 DROP TABLE IF EXISTS `vihotel`*/;  
  
/!*!50001 DROP VIEW IF EXISTS `vihotel`*/;  
  
/!*!50001 SET @saved_cs_client      = @@character_set_client */;  
  
/!*!50001 SET @saved_cs_results    = @@character_set_results */;  
  
/!*!50001 SET @saved_col_connection = @@collation_connection */;  
  
/!*!50001 SET character_set_client  = utf8 */;  
  
/!*!50001 SET character_set_results = utf8 */;  
  
/!*!50001 SET collation_connection  = utf8_general_ci */;  
  
/!*!50001 CREATE ALGORITHM=UNDEFINED */
```

```
/*!50013 DEFINER=`root`@`localhost` SQL SECURITY DEFINER */

/*!50001 VIEW `vihotel` AS select `s`.`id_sector` AS `cod_sec`,`s`.`nombre_sector`
AS `nombre_sector`,`s`.`estado_sector` AS `estado_sector`,`h`.`id_hotel` AS
`id_hotel`,`h`.`id_sector` AS `id_sector`,`h`.`nombre_hotel` AS
`nombre_hotel`,`h`.`estado_hotel` AS `estado_hotel` from (`sectores` `s` join
`hoteles` `h`) where (`h`.`id_sector` = `s`.`id_sector`) */;

/*!50001 SET character_set_client    = @saved_cs_client */;

/*!50001 SET character_set_results   = @saved_cs_results */;

/*!50001 SET collation_connection    = @saved_col_connection */;

/*!40103 SET TIME_ZONE=@OLD_TIME_ZONE */;

/*!40101 SET SQL_MODE=@OLD_SQL_MODE */;

/*!40014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;

/*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;

/*!40101 SET

CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;

/*!40101 SET

CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;

/*!40101 SET

COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;

/*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;

-- Dump completed on 2015-09-19 13:09:39
```

Bibliografía.

- Marcelo Coba. (06-01-2015). App móviles. STAFF VACATIONS CIA.LTDA, 1, 1. 06-01-2015, De Presidente de la empresa Base de datos.
- <http://users.dcc.uchile.cl/~psalinas/uml/casosuso.html>
- <https://docs.kde.org/stable4/es/kdesdk/umbrello/uml-elements.html>
- <http://www.javeriana.edu.co/biblos/tesis/ingenieria/Tesis236.pdf>
- <http://delta.cs.cinvestav.mx/~pmalvarez/tesis-hermes.pdf>
- <http://eprints.ucm.es/9085/1/Memoria.pdf>
- <http://lignux.com/instalar-el-ultimo-netbeans-y-jdk/>
- <http://www.adslzone.net/software/android-en-el-pc/como-instalar-android-studio-en-windows/>