

UNIVERSIDAD CATÓLICA DE SANTA MARÍA
FACULTAD DE CIENCIAS E INGENIERIAS FÍSICAS Y FORMALES
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS



DESARROLLO DE UN SISTEMA DE PUBLICIDAD UTILIZANDO TECNOLOGÍAS
MÓVILES, WEARABLES Y COMPUTACIÓN UBICUA

Tesis presentada por los Bachilleres,

NÚÑEZ DEL PRADO MANSILLA, CRISTOPHER

RODRÍGUEZ DELGADO, DIANA LUCÍA

Para optar el Título Profesional de:

INGENIERO DE SISTEMAS

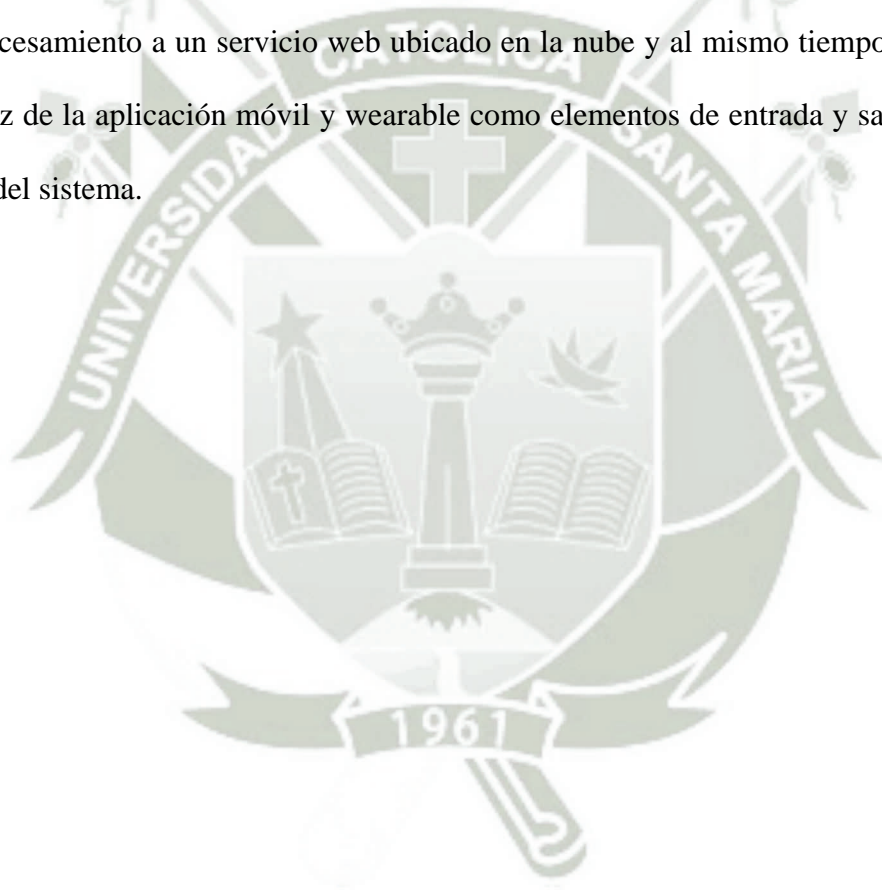
Asesor: FERNÁNDEZ DEL CARPIO, ALVARO

AREQUIPA - PERÚ

2016

Presentación

El siguiente trabajo tiene como objetivo realizar un sistema de publicidad utilizando tecnologías móviles, wearables y computación ubicua. El proyecto se realizó utilizando la metodología ágil programación extrema. Se trabajó bajo el ambiente IOS para la aplicación móvil y WatchOS para la aplicación wearable. Este sistema fue realizado bajo una arquitectura de tres capas, asignando las tareas de procesamiento a un servicio web ubicado en la nube y al mismo tiempo usa la interfaz de la aplicación móvil y wearable como elementos de entrada y salida de datos del sistema.

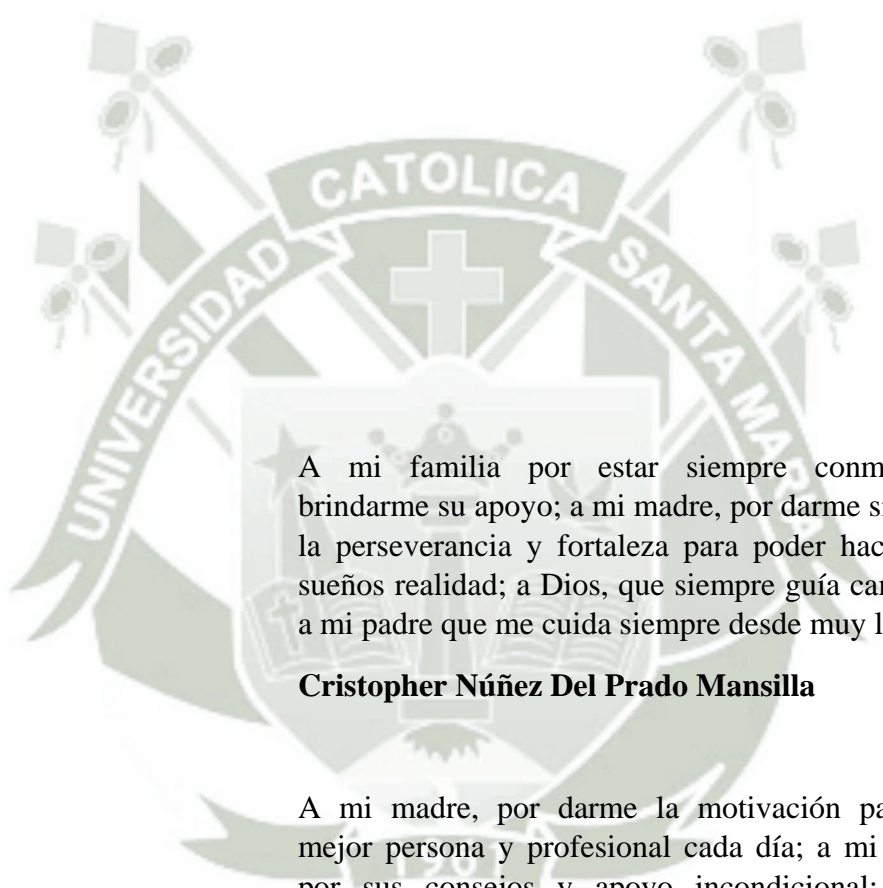


Agradecimientos

A nuestra familia siempre nos apoyó en todo momento y nos brindó su cariño incondicional; a nuestros profesores, quienes nos entregaron el conocimiento y la inspiración para hacer esto posible y a nuestro asesor de tesis, el ingeniero Álvaro Fernández quien nos brindó sus consejos y experiencias durante el desarrollo de este proyecto.



Dedicatoria



A mi familia por estar siempre conmigo y brindarme su apoyo; a mi madre, por darme siempre la perseverancia y fortaleza para poder hacer mis sueños realidad; a Dios, que siempre guía camino y a mi padre que me cuida siempre desde muy lejos.

Cristopher Núñez Del Prado Mansilla

A mi madre, por darme la motivación para ser mejor persona y profesional cada día; a mi padre, por sus consejos y apoyo incondicional; a mi hermana, por estar siempre conmigo y por compartir los buenos momentos; a mis amigos, por enseñarme a soñar sin límites y a Dios, por darme todas estas oportunidades.

Diana Lucía Rodríguez Delgado

Epígrafe

La primera regla de cualquier tecnología utilizada en los negocios, es que la automatización aplicada a una operación eficiente, aumentará la eficiencia. La segunda, es que la automatización aplicada a una operación ineficiente, aumentará la ineficiencia.

Bill Gates

Para saber si lo que estamos buscando automatizar será útil para las demás personas, debemos preguntarnos, si en la actualidad trabaja de manera eficiente y también observar las características que puedan ser mejoradas con la automatización. El éxito radica en siempre tratar de ser mejor que ayer.

Índice o Tabla de contenido

Presentación	2
Agradecimientos	3
Dedicatoria	4
Epígrafe	5
Índice o Tabla de contenido	6
Índice de figuras	9
Índice de tablas	20
Resumen	21
Abstract	22
Introducción	23
Capítulo 1: Descripción del proyecto	25
1.1 Objetivos	25
1.2 Alcances y limitaciones	26
1.2.1 Alcance del sistema.	26
1.2.2 Limitaciones del proyecto	27
1.3 Fundamentos Teóricos	29
1.3.1 Antecedentes del proyecto	29
1.3.2 Bases teóricas del proyecto.....	33
1.4 Metodología programación extrema	38
1.4.1 Fase de planeamiento.....	39

1.4.2 Fase de diseño	40
1.4.3 Fase de codificación	41
1.4.4 Fase de pruebas	41
1.4.5 Fase de implantación	42
1.5 Herramientas	42
1.6 Aspectos relevantes del desarrollo	46
Capítulo 2: Documentación técnica	53
2.1 Plan del proyecto informático	53
2.2 Especificación de requisitos del proyecto de TIC	56
2.3 Especificación de diseño	87
2.4 Documentación técnica de programación	119
2.4.1 Documentación de la aplicación móvil	119
2.4.2 Documentación del servicio web	142
2.5 Pruebas	151
2.5.1 Pruebas de verificación	151
2.5.1.1 Pruebas de ejecución	151
2.5.1.2 Pruebas de rendimiento	175
2.5.1.3 Pruebas de seguridad.....	190
2.5.2 Pruebas de validación	192
2.6 Manuales de usuario	203
Conclusiones	223

Recomendaciones	226
Referencias bibliográficas.....	229
Apéndice A: Plan de proyecto de TIC aprobado	232
Apéndice B: Encuesta de validación de requerimientos a empresas	277
Apéndice C: Documentos y detalles del sistema	286



Índice de figuras

Figura 1: Diagrama del uso de la metodología programación extrema.	39
Figura 2: Diagrama de capas del sistema de promociones.	48
Figura 3: Diagrama de Gantt de los procesos de proyecto	53
Figura 4: Diagrama de Gantt de la fase de elaboración de diseño y arquitectura de software	54
Figura 5: Diagrama de Gantt de la fase de desarrollo del sistema.	55
Figura 6: Diagrama de Gantt de la fase de pruebas del sistema.	55
Figura 7 : Diagrama de Gantt de la fase de estudio de resultados del proyecto. ...	56
Figura 8: Resultados de la primera pregunta de la encuesta realizada a los posibles usuarios	57
Figura 9: Resultados de la segunda pregunta de la encuesta realizada a los posibles usuarios	58
Figura 10 : Resultados de la tercera pregunta de la encuesta realizada a los posibles usuarios.	59
Figura 11: Resultados de la cuarta pregunta de la encuesta realizada a los posibles usuarios	60
Figura 12: Resultados de la quinta pregunta de la encuesta realizada a los posibles usuarios	61
Figura 13 : Resultados de la sexta pregunta de la encuesta realizada a los posibles usuarios	62

Figura 14 : Resultados de la séptima pregunta de la encuesta realizada a los posibles usuarios	63
Figura 15 : Resultados de la octava pregunta de la encuesta realizada a los posibles usuarios	64
Figura 16: Resultados de la novena pregunta de la encuesta realizada a los posibles usuarios	65
Figura 17: Resultados de la décima pregunta de la encuesta realizada a los posibles usuarios	66
Figura 18: Resultados de la décimo primera pregunta de la encuesta realizada a los posibles usuarios	67
Figura 19: Diagrama de casos de uso del sistema de promociones	68
Figura 20 Diagrama de clases de la aplicación	90
Figura 21: Diagrama de clases del web service	92
Figura 22: El patrón Fachada dentro del diagrama de clases del web service.	94
Figura 23: Clases de servicios dentro del diagrama de clases del web service. ...	96
Figura 24: El patrón Observador dentro del diagrama de Clases del web service.	97
Figura 25 : Clases repositorio dentro del diagrama de clases del web service	98
Figura 26 : El patrón Singleton dentro del diagrama de clases del web service ...	99
Figura 27: Diagrama de secuencia de la utilización de un cupón	101
Figura 28: Diagrama de secuencia de la actualización categorías preferidas	103

Figura 29: Diagrama de secuencia de la obtención de promociones	105
Figura 30: Diagrama de secuencia de la actualización de ubicación.	107
Figura 31 : Diagrama de paquetes del sistema de promociones	110
Figura 32: Diagrama de actividades de la utilización de un cupón	111
Figura 33: Diagrama de actividades de la actualización categorías preferidas ..	113
Figura 34: Diagrama de actividades de la obtención promoción	114
Figura 35: Diagrama de actividades de la actualización de ubicación	116
Figura 36: Diagrama de despliegue del sistema de promociones	118
Figura 37 : Código función de conexión de la aplicación móvil con el servicio web	121
Figura 38 : Código función de obtención de ubicación geográfica del usuario..	122
Figura 39 : Código funciones de publicación de promociones usando Facebook y Twitter	124
Figura 40: Código función de generación de código QR para uso de una promoción	125
Figura 41 : Desarrollo de interfaces y vínculos entre las actividades de la aplicación	126
Figura 42: Alertas mostradas dentro de la aplicación cuando se produce un problema	128
Figura 43: Confirmaciones mostradas en la aplicación cuando se realiza una función satisfactoriamente	129

Figura 44: Menú principal de la aplicación móvil	130
Figura 45: Actividad de configuraciones dentro de la aplicación móvil	131
Figura 46: Menú principal de la aplicación wearable	132
Figura 47: Barras de herramientas dentro de la aplicación móvil	133
Figura 48: Actividad de ingreso al sistema dentro de la aplicación móvil	134
Figura 49: Actividades dentro de la aplicación móvil que presentan tablas de datos	135
Figura 50: Actividad de categorías de la aplicación móvil	136
Figura 51: Actividad que muestra una promoción dentro de la aplicación móvil	137
Figura 52: Actividad que muestra de la promoción dentro de la cuponera virtual en la aplicación móvil	138
Figura 53: Actividad de muestra de la promoción dentro de la cuponera virtual en la aplicación wearable	139
Figura 54: Actividad de publicación de promociones usando Facebook o Twitter	140
Figura 55: Generación del código QR de la promoción seleccionada en la aplicación móvil	141
Figura 56: Generación del código QR de la promoción seleccionada en la aplicación wearable	142
Figura 57: Archivo POM del web service	144

Figura 58: Plugin cxf codegen	145
Figura 59: Archivo de configuración de Hibernate	146
Figura 60: El patrón Observador en el web service	147
Figura 61: El patrón Singleton en el web service	148
Figura 62: Interceptor para la autenticación del web service	149
Figura 63: Algoritmo de búsqueda de zona comercial	150
Figura 64: Prueba de iniciar sesión con resultado verdadero	152
Figura 65: Prueba de iniciar sesión con resultado falso	153
Figura 66: Prueba de guardar acceso cuando el usuario está dentro de una zona comercial	154
Figura 67: Prueba de guardar acceso cuando el usuario no está dentro de una zona comercial	155
Figura 68: Prueba de obtener las tiendas según su zona comercial	156
Figura 69: Prueba de obtener los cupones según su zona comercial	157
Figura 70: Prueba de guardar un cupón exitosamente	158
Figura 71: Prueba de guardar un cupón fallidamente	159
Figura 72: Primera prueba de obtener las categorías preferidas	160
Figura 73: Prueba de usar un cupón exitosamente	161
Figura 74: Prueba de usar un cupón fallidamente	162
Figura 75: Segunda prueba de obtener las categorías preferidas	163
Figura 76: Prueba de eliminar un cupón exitosamente	164

Figura 77: Prueba de eliminar un cupón fallidamente	165
Figura 78: Prueba de guardar una categoría exitosamente	166
Figura 79: Prueba de guardar una categoría fallidamente	167
Figura 80: Prueba de eliminar una categoría exitosamente	168
Figura 81: Prueba de eliminar una categoría fallidamente	169
Figura 82: Prueba de obtener las categorías exitosamente	170
Figura 83: Prueba de obtener las categorías fallidamente	171
Figura 84: Prueba de obtener las categorías según preferencias exitosamente ..	172
Figura 85: Prueba de obtener las categorías según preferencias fallidamente	173
Figura 86: Prueba de obtener las categorías exitosamente	174
Figura 87: Prueba de obtener las categorías fallidamente	175
Figura 88: Prueba de uso de memoria RAM por parte de la aplicación wearable	176
Figura 89: Prueba de uso de CPU por parte de la aplicación wearable	177
Figura 90: Prueba integral de recursos de hardware por parte de la aplicación wearable	178
Figura 91: Primera prueba de uso de conectividad de la aplicación móvil.	179
Figura 92: Segunda prueba de uso de conectividad de la aplicación móvil.	180
Figura 93: Medición del ancho de banda utilizado para las pruebas de conectividad	181
Figura 94: Primera prueba de uso de CPU de la aplicación móvil.	182

Figura 95: Segunda prueba de uso de CPU de la aplicación móvil.....	182
Figura 96: Tercera prueba de uso de CPU de la aplicación móvil.....	183
Figura 97: Primera prueba de uso de memoria de la aplicación móvil.....	184
Figura 98: Segunda prueba de uso de memoria de la aplicación móvil.....	185
Figura 99: Tercera prueba de uso de memoria de la aplicación móvil.....	186
Figura 100: Cuarta prueba de uso de memoria de la aplicación móvil.....	187
Figura 101: Medición de velocidad de internet.....	188
Figura 102: Prueba de rendimiento con un hilo.....	188
Figura 103: Prueba de rendimiento con diez hilos.....	189
Figura 104: Primera prueba de seguridad fallida.....	190
Figura 105: Segunda prueba de seguridad fallida.....	191
Figura 106: Prueba de seguridad exitosa.....	192
Figura 107: Resultados de la primera pregunta de la encuesta de validación del sistema.....	194
Figura 108: Resultados de la segunda pregunta de la encuesta de validación del sistema.....	195
Figura 109: Resultados de la tercera pregunta de la encuesta de validación del sistema.....	196
Figura 110: Resultados de la cuarta pregunta de la encuesta de validación del sistema.....	197

Figura 111: Resultados de la quinta pregunta de la encuesta de validación del sistema.....	198
Figura 112: Resultados de la sexta pregunta de la encuesta de validación del sistema.....	199
Figura 113: Resultados de la séptima pregunta de la encuesta de validación del sistema.....	200
Figura 114: Resultados de la octava pregunta de la encuesta de validación del sistema.....	201
Figura 115: Resultados de la novena pregunta de la encuesta de validación del sistema.....	202
Figura 116: Resultados de la décima pregunta de la encuesta de validación del sistema.....	203
Figura 117: Actividad de acceso al sistema dentro de la aplicación móvil	205
Figura 118: Mensaje de datos erróneos para ingresar al sistema	206
Figura 119: Actividad de búsqueda de ubicación geográfica del usuario dentro de la aplicación móvil	207
Figura 120: Alerta cuando no se puede encontrar la posición geográfica del usuario	208
Figura 121: Confirmación de que se encontró la ubicación geográfica del usuario	209
Figura 122: Menú principal de la aplicación	210

Figura 123: Actividad de listado de establecimientos cercanos al usuario.....	211
Figura 124: Actividad de listado de categorías dentro de un establecimiento seleccionado.....	212
Figura 125: Actividad de listado de promociones dentro de un establecimiento	213
Figura 126: Actividad de muestra de una promoción cercana al usuario	214
Figura 127: Actividad de muestra de la cuponera virtual del usuario	215
Figura 128: Actividad de muestra de una promoción dentro de la cuponera virtual	216
Figura 129: Actividades de publicación de una promoción en Facebook y Twitter	217
Figura 130: Actividad de muestra del código QR generado para el uso de una promoción	218
Figura 131: Actividad de configuraciones de la aplicación móvil	219
Figura 132: Actividad de actualización de categorías de preferencias	220
Figura 133: Menú principal de la aplicación wearable.....	221
Figura 134: Actividad de muestra de una promoción dentro de la aplicación wearable	221
Figura 135: Actividad de muestra código QR generado para el uso de una promoción dentro de la aplicación wearable	222

Figura A 1: Diagrama de base de datos sistema de publicidad.....	266
Figura A 2: Diagrama de Gantt de los procesos de proyecto	270
Figura A 3: Diagrama de Gantt de la fase de elaboración de diseño y arquitectura de software	270
Figura A 4: Diagrama de Gantt de la fase de desarrollo del sistema.....	271
Figura A 5: Diagrama de Gantt de la fase de pruebas del sistema	272
Figura A 6: Diagrama de Gantt de la fase de estudio de resultados del proyecto 272	
Figura B 1: Respuestas de la primera pregunta de la encuesta de validación a empresas	278
Figura B 2: Respuestas de la segunda pregunta de la encuesta de validación a empresas	279
Figura B 3: Respuestas de la tercera pregunta de la encuesta de validación a empresas	280
Figura B 4: Respuestas de la cuarta pregunta de la encuesta de validación a empresas	281
Figura B 5: Respuestas de la quinta pregunta de la encuesta de validación a empresas	282
Figura B 6: Respuestas de la sexta pregunta de la encuesta de validación a empresas	283

Figura B 7: Respuestas de la séptima pregunta de la encuesta de validación a
empresas..... 284

Figura B 8: Respuestas de la octava pregunta de la encuesta de validación a
empresas..... 285



Índice de tablas

Tabla 1: Comparativa entre sistemas de publicidad similares	33
Tabla 2: Primer actor de diagrama de casos de uso	69
Tabla 3: Segundo actor de diagrama de casos de uso	69
Tabla 4: Descripción del primer caso de uso	70
Tabla 5: Descripción del segundo caso de uso	71
Tabla 6: Descripción del tercer caso de uso.....	72
Tabla 7: Descripción del cuarto caso de uso	73
Tabla 8: Descripción del quinto caso de uso	74
Tabla 9: Descripción del sexto caso de uso	75
Tabla 10: Descripción del séptimo caso de uso	76
Tabla 11: Descripción del octavo caso de uso	77
Tabla 12: Descripción del noveno caso de uso	78
Tabla 13: Descripción del décimo caso de uso	79
Tabla 14: Descripción del décimo primer caso de uso	80
Tabla 15: Descripción del décimo segundo caso de uso.....	81
Tabla 16: Descripción del décimo tercer caso de uso	82
Tabla 17: Descripción del décimo cuarto caso de uso	83
Tabla 18: Descripción del décimo quinto caso de uso	84
Tabla 19: Descripción del décimo sexto caso de uso	85
Tabla 20: Descripción del décimo séptimo caso de uso	86

Resumen

Actualmente, las empresas realizan publicidad para poder brindar información a los usuarios de los productos que ofrecen. Según las empresas consultadas, la publicidad convencional no genera el impacto deseado en sus clientes y no se refleja de una manera determinante en sus ventas. Se buscó disminuir estos inconvenientes, mediante un sistema de publicidad virtual, la cual trabaja con geolocalización para poder mostrar al usuario promociones de lugares cercanos a él; también se cuenta con gestión de preferencias, con esto se buscó enviar promociones que generen un mayor impacto en el usuario. Se buscó que este sistema sea rápido y al mismo tiempo sólido, por lo que fue realizado con una arquitectura de tres capas, de esta manera, todo el procesamiento de datos es hecho mediante un servicio web ubicado en la nube, mientras la aplicación móvil actúa como entrada y salida de datos. Este software, fue realizado bajo la metodología ágil llamada programación extrema, para agilizar las etapas de desarrollo sin sacrificar el correcto planeamiento del proyecto. El resultado obtenido es el sistema de publicidad con todas las características anteriormente mencionadas, el 93% de los encuestados valoran que la aplicación en general funciona correctamente.

Palabras Clave: Ubicuidad, Publicidad, Ecología, Aplicaciones móviles, Dispositivos Móviles, Dispositivos Wearable.

Abstract

Nowadays, companies advertise to be able to give to their customers, information about the products they offer. According to the companies consulted, conventional advertising does not generate the desired impact on their customers and does not reflected in a determinant way on their sales. We sought to decrease these inconveniences, through a virtual advertising system, which works with geolocation, in order to show promotions from stores close to users; also we implemented preferences management, in order to send promotions to the users trying to generate a greater impact on them. This system is intended to be fast and at the same time solid, so it was done with a three layer architecture, thus, all data processing is done using a web service and mobile applications which only works as a data Input/Output. This software, was developed under the agile methodology called extreme programming, to be able to speed up development stages without sacrificing the correct project planning. The result obtained is the advertising system including all the features mentioned above, 93% of the survey respondents value that the application in global works correctly.

Keywords: Ubiquitous, Advertising, Ecology, Mobile Application, Mobile Devices, Wearables.

Introducción

Actualmente, las empresas tienen un gran déficit en cuanto a la información que poseen de sus clientes y potenciales compradores, por lo cual, no se puede realizar una publicidad adecuada y por ello se pierden muchas oportunidades de ventas y de obtención de clientes; incluso, en ciertas ocasiones logran disgustar a sus propios compradores en vez de fidelizarlos con la empresa que sería lo más óptimo.

Además, las empresas usualmente ofrecen publicidad a sus clientes de la manera convencional para promocionar sus productos o servicios; sin embargo, este tipo de publicidad no es efectivo en relación a sus ventas y al mismo tiempo no ayuda a promover el desarrollo sostenible, debido a que para realizar este tipo de publicidad se utilizan grandes cantidades de materia prima como papel o plástico. Además, en muchas ocasiones la publicidad convencional no alcanza su objetivo trazado, ya que, al no ser personalizada muchas veces los clientes no la toman en cuenta y posteriormente es desechada.

Para menguar estos problemas se realizó el sistema de publicidad, en el cual las empresas pueden ofertar sus productos o servicios utilizando publicidad virtual, para garantizar un correcto uso de recursos ecológicos; y se añadió a esto características que se obtienen gracias al empleo de dispositivos móviles. El éxito de esta aplicación radica en que se desea brindar las promociones al usuario en el

momento y lugar indicado, por lo que se utilizó la herramienta de geolocalización del móvil para poder determinar cuando el usuario se encuentra dentro de una zona comercial de su ciudad, además se conocerán las categorías de productos según las preferencias del usuario para poder enviarle promociones que vayan de acuerdo a sus gustos.

Este sistema está basado en una arquitectura de 3 capas, con lo que se realizaron todos los procesos en un servicio web y de esta manera se logra garantizar un menor uso de los recursos limitados que ofrecen los dispositivos móviles, haciendo que la aplicación móvil sea utilizada como la interfaz principal que tiene este sistema con el usuario. Además Este proyecto se realizó mediante la metodología de desarrollo ágil de programación extrema (XP), gracias a ello se pudo desarrollar el sistema de una manera más óptima.

Además se realizó una aplicación para dispositivos wearable, en este caso relojes inteligentes; con esto se busca que esta aplicación se pueda complementar con la del dispositivo móvil y sirva como una interfaz más rápida al sistema; se muestran los últimos datos vistos en la aplicación móvil y se genera el código QR de la promoción dentro del reloj inteligente y de manera automática, de esta forma, se garantiza un mejor tiempo de respuesta para la satisfacción del cliente con el sistema. Esta herramienta cuenta con muchas características que ayudarán a las personas en diferentes aspectos de la vida cotidiana.

Capítulo 1: Descripción del proyecto

1.1 Objetivos

Objetivo General

Desarrollar un sistema de publicidad utilizando tecnologías móviles, wearables y computación ubicua.

Objetivos Específicos

1. Utilizar la arquitectura de tres capas, para que sea capaz de soportar el sistema de publicidad que se desea desarrollar.
2. Aplicar la metodología ágil de programación extrema (XP) en el desarrollo del sistema de publicidad.
3. Obtener información de los usuarios de la aplicación mediante el uso del sistema y a través de los sensores del dispositivo móvil.
4. Desarrollar una aplicación móvil con interfaz intuitiva, la cual será utilizada por el usuario para poder acceder, de forma sencilla, a toda la información que ofrece el sistema; además se aplicaron métodos de autenticación de datos para proteger la integridad de las transacciones de los usuarios.
5. Desarrollar las pruebas del sistema utilizando pruebas de validación y verificación para garantizar el correcto funcionamiento del sistema.

1.2 Alcances y limitaciones

1.2.1 Alcance del sistema.

Los usuarios de la aplicación, que son los principales actores, obtienen grandes beneficios, ya que reciben publicidad basada en sus gustos, preferencias y necesidades, estas se actualizan conforme el usuario interactúe con la aplicación, de esta manera la asignación de publicidad se puede adaptar fielmente a los verdaderos gustos del usuario, todo esto hace que la aplicación sea personalizada y se beneficie mejor después de cada experiencia que el usuario tiene con la aplicación.

Las empresas obtienen grandes beneficios debido a que la aplicación les permite realizar publicidad sin costo alguno, para que estas puedan mostrar y publicitar sus productos o servicios en oferta, además de obtener un mayor número de clientes e incrementar en sus ventas, ellos pueden disminuir la inversión en publicidad y realizar un mejor tipo de publicidad, una más personalizada a sus clientes gracias a esta herramienta. Este ciclo beneficia tanto a las empresas como al servicio que se busca brindar, dado que al proporcionar un buen servicio a las empresas, más de ellas se afiliarán al servicio, con ello este sistema tendrá más y mejor publicidad para proveer a los usuarios y por lo tanto atraerá a más de ellos, incrementando el alcance de la publicidad.

Dado que el proyecto se basa en publicidad virtual, la sociedad en sí se beneficia, ya que el uso de este tipo específico de publicidad ayuda a contrarrestar la contaminación visual y física, porque se pretende disminuir el uso de recursos naturales para fines publicitarios, al utilizar medios electrónicos para transmitir la publicidad en lugar de utilizar materias primas como papel como se da en el caso de la publicidad convencional.

Se utilizan medios electrónicos para lograr la difusión de las promociones, de este modo, se brinda a las personas publicidad que de verdad requiere en el momento y lugar apropiado para garantizar su efectividad. Además, la publicidad virtual propone el uso de publicidad personalizada a cada usuario, lo que le permitirá a éste, obtener mejores beneficios de las ofertas dado que únicamente obtendrá aquellas que se ajusten a sus preferencias.

1.2.2 Limitaciones del proyecto

Este proyecto es ambicioso en diferentes aspectos, por lo tanto, se deben considerar muchas limitaciones para poder ser desarrollado e implantado. Estas limitaciones son de diferentes tipos, entre ellas se encuentran las limitaciones económicas del proyecto; el sistema utiliza un servicio gratuito en la nube con recursos compartidos, limitando la rapidez de las operaciones y la cantidad de información que se puede almacenar. Para que la herramienta logre soportar un mayor número de usuarios y garantice que las características de la herramienta

funcionan rápida y eficazmente, se deben tomar servicios más especializados, estos conllevan gastos constantes que deben hacerse en los servicios de almacenamiento de datos en la nube y los servidores de aplicaciones, pues estos servicios se tomarán de una empresa externa al proyecto. Algunos de estos servicios pueden ser costosos y la mayoría de ellos se manejan a través de mensualidades o anualidades.

De igual manera se presentan limitaciones tecnológicas, ya que algunas funciones de la aplicación necesitan dispositivos móviles de alta o media gama, pues es necesario realizar la utilización de los componentes que los dispositivo de media a alta gama poseen, tales como los sensores de GPS para efectuar la geolocalización del usuario, tecnología de telefonía móvil 4G, pues esta cuenta con velocidades de transmisión altas, permitiendo así al usuario obtener el máximo provecho de todas las funciones de la aplicación.

Otra de las limitaciones importantes de esta herramienta en desarrollo, es la desconfianza por parte de los usuarios para compartir sus datos personales o datos sobre los productos adquiridos por ellos, a través de una aplicación “desconocida”, los usuarios suelen tener temor a que esta información sea ventilada o que personas ajenas al proyecto puedan utilizar estos datos con otros fines, por lo que se debe concientizar a los usuarios y demostrarles los niveles de seguridad de la información que la herramienta brinda al usuario, para que se sientan seguros de

que los datos extraídos son únicamente para conocer sus preferencias y poder hacer sugerencias de la publicidad que ellos desearían.

1.3 Fundamentos Teóricos

1.3.1 Antecedentes del proyecto

Existen múltiples sistemas de promociones virtuales, cada uno con características similares y diferentes entre sí, a continuación se detallan cinco sistemas con sus respectivas características.

SnipSnap® es un sistema que permite la organización de cupones en paquetes predeterminados o en paquetes creados por el mismo usuario, envía notificaciones en caso el cupón esté cercano a la locación donde el usuario se encuentra y envía alertas cuando el cupón está próximo a expirar. Su servicio genera los datos del cupón a partir del análisis del código de barras y la información del cupón que viene en los cupones impresos. Además, permite reconocer un producto que se desea comprar y realiza una búsqueda de cupones sobre él en las distintas tiendas, es decir, que cuenta con una base de datos de cupones que se actualizan a medida que los usuarios los incluyen en su cuenta para utilizarlos de forma electrónica. SnipSnap (2014). SnipSnap® cuenta con un diseño simple y de fácil comprensión, también permite compartir los cupones con otros usuarios; sin

embargo, se basa en cupones ya impresos, lo cual no ayuda a disminuir el impacto de la publicidad en el medio ambiente.

RetailMeNot®, es un sistema de cupones que se encarga de organizar los mismos según categorías o tiendas, cuenta con características de geolocalización y principalmente muestra las promociones que tienen mayor acogida entre los usuarios; cuenta con la opción de recibir notificaciones sobre las tiendas a las que el usuario se suscribe, no existe límite para dichas suscripciones, esto permite al usuario sincronizar con las nuevas promociones que ofrecen las tiendas en el menor tiempo posible. Sin embargo, no cuenta con las alertas en caso que la promoción esté a punto de caducar. El diseño de su interfaz es simple e intuitivo. Este además ofrece aplicación para el wearable, donde muestra todas las promociones y permite el acceso a su respectivo código de barras. RetailMeNot (2016).

Thecouponsapp®, es un servicio que brinda cupones especiales, estos cupones envían notificaciones cuando el usuario está próximo a alguna promoción. Además, envían alertas cuando la promoción está próxima a expirar. También permite al usuario realizar una suscripción a las tiendas, para obtener promociones de las tiendas a las que el usuario seleccionó, de esta manera el usuario será notificado a penas alguna de las tiendas ingrese una nueva promoción. Thecouponsappcom (2016). Esta aplicación no facilita su uso, en especial para las

personas mayores con problemas de lectura, pues las promociones están en letras pequeñas, lo que provoca que la aplicación se vea desordenada.

Groupon® es considerado un sistema de promociones, se centra principalmente en el descuento de servicios tales como spa y experiencias. Este sistema organiza los cupones que provee por distancia y por fecha de caducidad, además agrupa los cupones según rubros o categorías. No tiene alertas de cupones en específico cuando están a punto de expirar, y usualmente los cupones tienen una duración pequeña. Este sistema, permite comprar directamente a través del sistema aplicando los descuentos correspondientes directamente; además permite compartir con otros usuarios los cupones a través de las redes sociales. Groupon (2016). El diseño es de fácil uso, sin embargo, parte de la información está en letra pequeña, lo cual dificulta su lectura.

Shopular®, es un sistema que intenta impedir la proliferación de promociones inútiles, por ello se enfoca principalmente en enviar publicidad únicamente cuando la promoción es cercana a la localización del usuario. De igual forma que RetailMeNot®, este sistema, permite afiliarse a las tiendas que el usuario desee para obtener mayor cantidad de cupones de ellas. También permite dejar comentarios sobre los cupones, así el usuario podrá saber si el descuento vale la pena. Shopular (2014). El diseño es de fácil uso, pero algunas pantallas se ven

sobrecargadas de información, lo que ocasiona que la aplicación se vea un poco desordenada.

En conclusión, existen muchos sistemas de promociones, cada uno con sus propias características; la mayoría de ellos ofrecen servicios de geolocalización, y son amigables con el medio ambiente, sin embargo, el sistema de promociones que se realizó es organizado según categorías, cuenta con geolocalización, permite compartir con otros usuarios a través de redes sociales ya existentes, muestra alertas de expiración, es amigable con el medio ambiente, tiene una aplicación para el wearable y un diseño que permite al usuario utilizarlo fácilmente. Además, cuenta con una característica que ninguno de los sistemas anteriormente descritos tiene; el sistema realizado se adapta a las categorías que el usuario utiliza comúnmente y le muestra únicamente las promociones que pertenecen dichas categorías.

Tabla 1

Comparativa entre sistemas de publicidad similares

Nombre de la Aplicación	SnipSnap	RetailMeNot	Couponsapp	Groupon	Shopular
Organizado según	Paquetes	Categorías/ Tiendas	Categorías	Categorías	Ubicación
Personalización según uso	NO	NO	NO	NO	NO
Geolocalizado	SI	SI	SI	SI	SI
Compartir con otros usuarios	SI	NO	NO	SI	SI
Alertas de expiración	SI	NO	SI	NO	NO
Amigable con el Medio Ambiente	NO	SI	SI	SI	SI
Aplicación para el wearable	NO	SI	NO	NO	NO
Diseño amigable	SI	SI	NO	SI	SI

Nota: Elaboración Propia

1.3.2 Bases teóricas del proyecto

La publicidad es la manera de ofrecer los productos que vende una empresa a los clientes, su objetivo principal es brindar información a los clientes sobre los productos o servicios que una empresa ofrece, sin embargo, existen bastantes tipos de publicidad que en ocasiones son ignorados por los clientes o hasta rechazados. En estos tiempos se usan dispositivos digitales para poder mostrar promociones, ya que, ese es el destino de la publicidad según Tsang, Ho y Liang (2005). Además, se debe tomar una manera de adicionar nuevas características a la publicidad, según

Tsang, Ho y Liang (2005) "La actitud de los clientes es positiva si se relaciona con la intención de recibir publicidad móvil. La intención es afectada por la asociación del incentivo con la publicidad", por tal motivo se desea que el aplicativo en desarrollo posea estas características adicionales con la finalidad de atraer al cliente sin causarle molestias que muchas veces van de la mano con la publicidad convencional.

Según Vallina-Rodríguez, Shah, Finamore et al. (2012) "Las redes de anuncios son de gran impacto para los usuarios de dispositivos móviles". Por tal motivo, se realizó esta aplicación para los dispositivos móviles con sistema operativo iOS, además funciona en wearables; este sistema se realizó con una arquitectura de tres capas en la cual la capa de interfaz fue desarrollada en aplicaciones para diferentes sistemas operativos, la capa de lógica de negocio es única para todos ellos, eliminando la duplicación de código y posibles conflictos futuros al realizar algún cambio, la capa lógica se encontrará en un servicio web localizado en la nube.

Como nos plantea Mohan, Nath y Riva (2013) la publicidad en dispositivos móviles debe contar con un servidor que se encargará del almacenamiento, procesamiento, asignación y envío de publicidad sin importar el tipo de dispositivo o sistema operativo que lo solicita, en la arquitectura de tres capas planteada, este servidor se encuentra en la capa de lógica de negocio del sistema.

Según Gurumurthy, S., Narayanamoorthy, M., Valarmozhi (2011) “Para que la computación ubicua se una a las expectativas de los usuarios móviles es fundamental que los cambios ocurran de manera en que las personas perciban los roles de los dispositivos, aplicaciones y el ambiente”, por tal motivo se toma en cuenta que la aplicación móvil actúa como la interfaz con el usuario y los roles que se tienen dentro del sistema, el usuario solo puede ver y utilizar las promociones, mientras que las empresas tienen un rol distinto el cual es brindar las promociones para los usuarios.

Según Kurkovsk (2013) "Hoy en día, los teléfonos móviles actúan como puntos finales personalizados para muchos servicios de información". Por tal motivo se utilizarán los dispositivos móviles y wearables como los extractores de los datos de los clientes que usen el aplicativo, la idea es que este sistema se encuentre en cualquier lugar, y funcione también de igual manera en cualquier zona comercial.

Según Kurkovsky, S. (2013) “Una propiedad fundamental de los sistemas ubicuos es la integración del ambiente y rodear al usuario con esta presencia ubicua, los teléfonos móviles pueden ser dispositivos de interfaz importantes que conectan los usuarios al ambiente ubicuo”, por tal motivo este sistema tiene como interfaces gráficas a las aplicaciones móviles y wearable para que el usuario pueda interactuar con el sistema.

Al trabajar con dispositivos móviles y al realizar transacciones de los clientes mediante estos, se debe tener en cuenta la seguridad de los datos con los que se están trabajando en este sistema, por ello se envían estos datos utilizando mecanismos de autenticación de mensajes provistos por estándares ya implementados en los web services. Se sabe que los dispositivos móviles trabajan con todo tipo de datos del usuario los cuales pueden correr bastante peligro en manos equivocadas, en este caso los wearable y móviles son los principales extractores de datos de las preferencias de los usuarios por lo cual se toma cuidado con ellos, Polonetsky, Wolf y Fich (2015) indicaron que "Las organizaciones deben estar preparadas para defender los datos personales de sus clientes de amenazas, como podrían ser hackers o scanners. Dado un alto nivel de conectividad, también es importante que los datos de los wearable sean asegurados adecuadamente tanto dentro del dispositivo como en la nube o cualquier otro lugar donde se encuentre".

Según Thierer, A. (2015) "Los desafíos con la privacidad y de seguridad asociados con el Internet de las cosas y las tecnologías portátiles serán considerables" por tal motivo cada usuario debe estar registrado para poder ingresar al sistema, así ninguna persona puede ver su información sin haber iniciado sesión con su correspondiente usuario.

Según Pegler, Shahabi & Lamano (2014) Se espera que en el futuro los wearables brinden una mayor cantidad de sensores de medición lo cual favorecerá a la obtención de nuevos tipos de datos. Los nuevos sensores y los nuevos tipos de

mediciones por parte de los wearables permitirán en el futuro nuevos tipos de datos que considerar dentro de la asignación de publicidad, lo cual permitirá el estudio de nuevos factores que se deben tomar en cuenta en la asignación de publicidad orientada al usuario y ayudarán a mejorar la experiencia del usuario gracias a la mejor personalización de este servicio.

La publicidad de la forma tradicional tiene naturaleza pasiva, es aquella que aparece con imágenes y texto en periódicos y revistas, además de la publicidad dinámica que engloba las que son transmitidas por emisiones de radio y televisión, Chen y Hsieh (2012). Gracias al uso de las tecnologías de Comunicación y los dispositivos wearables, es posible realizar una publicidad que esté focalizada en el cliente, ya que usualmente estos dispositivos poseen información útil sobre el usuario, de este modo se logra evitar la propagación recurrente y generalizada de publicidad no deseada que puede llegar a hostigar al cliente. La publicidad personalizada se refiere a la asignación de publicidad al usuario basándose en la información recolectada, ofreciendo así un servicio con mayor valor tanto para el cliente como para la organización.

Se concluye indicando que se desarrolló este sistema de publicidad utilizando la computación ubicua para que pueda ser accedido en todos los lugares en lo que se encuentren zonas comerciales, además del uso de tecnologías como los móviles y wearables que son el presente y futuro de la tecnología. Este sistema es

multiplataforma y fue desarrollado en una arquitectura de tres capas, debido a que debe ser escalable y debe tener un tiempo óptimo de respuesta, además ninguna de las capas de la arquitectura fue desarrollada de una manera que afecte el desempeño de otra capa. Se debe tener en cuenta también que se trabaja con datos personales de los clientes, por eso, todos estos datos son protegidos de posibles ataques.

1.4 Metodología programación extrema

Al realizar un proyecto de software es necesario elegir la metodología adecuada para desarrollarlo correctamente en el menor tiempo posible. Por esto se eligió una metodología ágil frente a las metodologías tradicionales como el modelo de cascada, incremental, espiral, entre otros, ya que las metodologías tradicionales son estrictas y rígidas frente a cambios, la participación por parte de los clientes es mínima o casi nula y los grupos de desarrollo son grandes y poco comunicativos. Por el contrario, el atributo más representativo de las metodologías ágiles es la flexibilidad que tiene cuando ocurren de cambios en cualquier etapa del proyecto, los clientes se integran con los encargados del proyecto logrando una mejor comunicación, minimizando errores en requerimientos y maximizando la satisfacción de este, los grupos que integran el proyecto son pequeños, así se coordina de mejor manera el proyecto y la retroalimentación se da de manera más continua y efectiva.

Para el desarrollo del sistema de publicidad utilizando móviles, wearables y computación ubicua, se utiliza la metodología de programación extrema, la cual fue determinada para este proyecto por la cantidad de personas que trabajan en el desarrollo de esta herramienta y las características determinantes de esta metodología, las cuales se adecuan de mejor manera a proyectos como este.

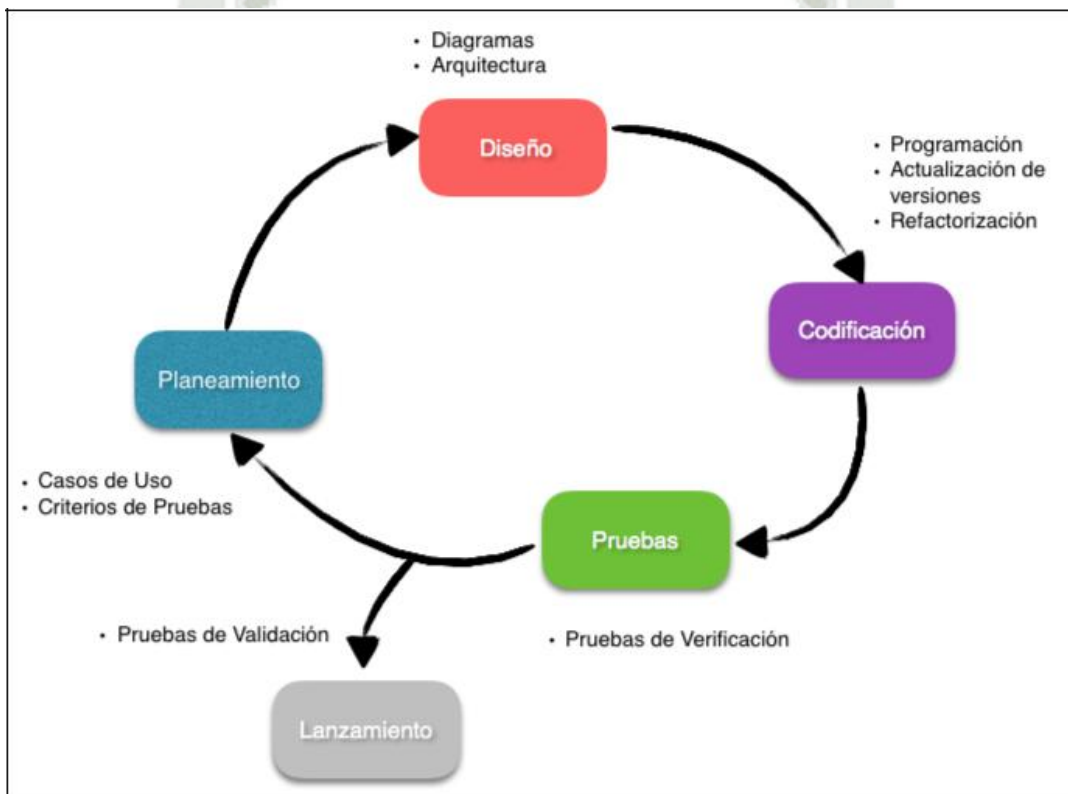


Figura 1: Diagrama del uso de la metodología programación extrema.
Fuente: Elaboración propia

1.4.1 Fase de planeamiento

La metodología de programación extrema inicia después de la fase de extracción de requerimientos, la primera fase de la Programación Extrema

corresponde al Planeamiento donde se organizan los requerimientos extraídos previamente, en casos de uso que contienen el requerimiento, la identidad del cliente que lo requiere y el beneficio que se obtendrá a partir de estos, estos casos de uso son organizadas y se seleccionan las que son más importantes para el producto que se desea desarrollar. En este punto, se realiza la primera retroalimentación, el cliente del producto debe revisar los casos de uso y selecciona cuales van de acuerdo a las necesidades que tenga a fin de certificar si el producto está bien encaminado. Luego se planean las liberaciones, las tareas y requisitos especificados en los casos de uso que se deben desarrollar y principalmente el tiempo que cada una de estas tareas va a necesitar para que sean finalizadas de la forma deseada.

1.4.2 Fase de diseño

Después de realizar los casos de uso se realizó la fase de desarrollo del diseño y arquitectura del software que se está trabajando. Esta fase fue realizada por ambos miembros del grupo para el correcto conocimiento de los detalles de las funciones del sistema, con esto se puede garantizar un correcto entendimiento compartido del código para las personas que lo implementaron. Además gracias a esto se trabajaba un diseño simple del sistema y el uso de estándares de programación para el código, como la implementación de patrones de diseño y la codificación del sistema, la cual fue realizada completamente en idioma inglés.

1.4.3 Fase de codificación

A continuación se realizó la fase de Codificación del software, dentro de esto se realizó actualización de versiones, lo cual corresponde a la entrega de bloques pequeños de código la cual debe ser hecha cada cierto tiempo, en este caso se realizan envíos al repositorio del sistema, este era realizado al menos una vez cada dos días, a menos que se estuviera implementando una funcionalidad extensa y el desarrollo exigiera el uso de más tiempo. Para esto se estuvo trabajando con un repositorio de código en la nube, la herramienta que se usó para la función de almacenamiento de código fue *Github*. Además en la sección del sistema que corresponde al servicio web, se realizó refactorización de código, lo cual también es una característica muy importante de esta metodología de desarrollo, esto se realizaba mediante constante eliminación de código duplicado o ineficiente, con lo cual el equipo de desarrollo mejoraba el diseño del sistema. En esta fase, el código también era evaluado continuamente para ofrecer la mayor calidad posible.

1.4.4 Fase de pruebas

En la fase de Pruebas se realizaron pruebas de verificación continuas al código, con esto se puede lograr asegurar que el código se desarrolla de manera efectiva, todo el tiempo de realizaron pruebas a las funciones nuevas que se implementaron dentro del sistema.

1.4.5 Fase de implantacion

Por último antes de la implantación del sistema, se realizó una prueba de validación de software para garantizar que los usuarios confirman que los requerimientos no funcionales de la aplicación fueron desarrollados de manera óptima. Posteriormente se implantó el sistema en los simuladores que provee el sistema para su uso.

En conclusión, esta metodología es la mejor para el proyecto, ya que permite la flexibilidad necesaria para que el este se adapte de la mejor manera a las necesidades del usuario, dado que el sistema presenta altos riesgos en cuanto a cambios en los requisitos, dado que es un proyecto que tiene como clientes finales al público en general.

1.5 Herramientas

Las herramientas utilizadas dentro de cada fase del ciclo de vida de desarrollo de este sistema serán indicadas a continuación, indicando el uso que se le dio y la fase en la que se utilizó cada uno de estos.

La extracción de requerimientos del sistema se realizó por medio de una encuesta realizada a posibles usuarios, esta encuesta fue no probabilística y se tomó como muestra a personas de diferentes edades que son cercanas al equipo de desarrollo. La extracción de requerimientos se realizó en una encuesta debido a que va dirigida al público en general, por lo tanto se preguntó a los encuestados las

funcionalidades con las que les gustaría que cuenta este sistema, posteriormente las funcionalidades más elegidas por los usuario fueron tomadas como requerimientos para el posterior diseño y desarrollo del sistema.

Se desarrolló una arquitectura de software de tres capas para separar las responsabilidades de cada uno de los componentes del proyecto, de esta manera cada capa es independiente y trabaja de manera autónoma con respecto a las demás, pues se busca que el sistema sea más fácil de mantener y pueda ser escalable para la adición de las funciones que se mencionan anteriormente. Se eligió el modelo de tres capas ya que es altamente flexible, confiable, escalable y es mantenible, también permite que la capa de presentación que es la principal interfaz de interacción con el cliente sea ligera, puesto que la capa de lógica de negocio es la encargada del procesamiento de las solicitudes. Además permite realizar cambios sin necesidad de constantes actualizaciones en la interfaz de usuario. Se eligió esta arquitectura por encima de la de dos capas porque se considera que este sistema tiene altas probabilidades de crecimiento y se necesita un modelo desacoplado que permita realizar estos cambios afectando mínimamente a los usuarios.

Se realizaron diferentes tipos de pruebas, primero se realizaron pruebas unitarias, a cada uno de los componentes o métodos del sistema, para asegurar que los componentes funcionan de forma adecuada, posteriormente se realizaron pruebas de validación que se encarga de asegurar que el sistema satisface al usuario

y finalmente verificación que determina si el software cumple correctamente con los requerimientos especificados.

Previamente se explicó que para aplicar la metodología de Programación Extrema es necesario tener un repositorio del proyecto, para que los miembros del equipo de trabajo puedan almacenar el avance y al mismo tiempo se cuente con una copia de seguridad, para este fin se trabajó con el repositorio en línea *Github*. Se eligió este repositorio de código frente a otros, porque es sólido y seguro, también fue elegido por la experiencia previa en él, de los miembros del equipo de desarrollo de este sistema, además se necesita el repositorio *Github* para actualizar el contenido en el servidor de aplicaciones *Openshift*, este servidor procesa los cambios al repositorio de *Github*, si no se encuentra ningún error, el servidor actualiza automáticamente el servicio. El avance periódico se fue documentando en este repositorio en línea y cada vez que se realizaba una carga de código al proyecto, se especificaron las características de la nueva carga para mantener al tanto el avance del equipo.

Se determinó realizar los elementos de la capa de vista dentro del entorno iOS para la aplicación móvil y WatchOS para la aplicación wearable. Se escogieron estos sistemas por la disponibilidad de simuladores de prueba de ambos dispositivos, la facilidad de uso de los componentes de los dispositivos con los que se está trabajando y la experiencia del equipo de desarrollo dentro de este sistema. La codificación de estas dos aplicaciones se realizó con el lenguaje de

programación Swift por su robustez, la solidez que brinda para el uso de recursos de hardware de los dispositivos con los que trabaja, que se encarga del procesamiento de datos y la obtención de la ubicación del usuario gracias al GPS del teléfono móvil.

Para la implementación del servicio web se aplicaron algunos principios SOLID, porque administran las dependencias del sistema y le permite ser más robusto y más reutilizable. Existen 5 principios: el principio de una única responsabilidad (SRP), el principio de abierto cerrado (OCP), el principio de sustitución Liskov(LSP), el principio de interfaz de segregación(ISP) y el principio de inversión de dependencia (DIP), Martin, R (2002). En el servicio web, las clases se dividen en clases de servicio y clases repositorio, con la finalidad de que cada clase tenga una responsabilidad única siguiendo el primer principio, porque mientras más responsabilidades tenga una clase, son más propensas a cambiar y eso tiende a crear errores. Al utilizar el principio de responsabilidad única, se utiliza también el principio de interfaz de segregación, que indica que no se obligue a utilizar una clase con más métodos de los necesarios, ya que al tener una única responsabilidad, no se tendrán métodos adicionales a los que se necesitan. Siguiendo el principio de abierto cerrado, cada clase tiene su respectiva interfaz, de tal manera las clases están abiertas a extensión pero están cerradas a modificación, también siguen el principio Liskov, ya que al tener las interfaces anteriormente mencionadas, se puede sustituir un objeto por cualquier objeto que

descienda de la misma interfaz. El último principio indica que las abstracciones no deben depender de los detalles, estos deben depender de las abstracciones, es decir que las clases concretas no debería depender de otra clase concreta, sino de su interfaz, en el servicio web siempre se llama a la interfaz del servicio o del repositorio, de esta manera, en caso que se desee cambiar la clase concreta no afecta a la interfaz que realiza la solicitud.

1.6 Aspectos relevantes del desarrollo

Para el desarrollo del sistema de publicidad se utilizó la arquitectura de tres capas, que consta de la capa de presentación, la de lógica de negocio, la de acceso a base de datos. Según Satpute R. et al. (2015) esta arquitectura permite que los sistemas sean escalables, flexibles y reusables, gracias a su separación entre capas se puede realizar cambios sin afectar a las demás capas, esta independencia reduce el consumo de recursos y tiempo, además reduce la complejidad para la implementación del cambio. Se eligió la arquitectura de tres capas porque se planificó que el sistema de promociones en el futuro se expanda y esta arquitectura permite realizar la expansión del sistema de forma más sencilla y rápida, por su modularidad, su capacidad de agregación de capas y su reutilización de código.

En primer lugar, se tiene la capa de presentación que es la que interactúa directamente con el usuario final en el sistema de promociones, esta capa contiene la aplicación del teléfono, con sistema operativo “IOS” y la aplicación del Apple

Watch. Ambos están encargados de la administración de las acciones del usuario en las interfaces tanto en el teléfono como en el reloj inteligente.

En segundo lugar, está la capa de lógica de negocio, en ella se encuentra un único servidor el cual contiene el servicio web “SOAP”, que es el encargado de procesar todas las solicitudes y responderlas. En esta capa también se encuentra el bean de autenticación, este se encarga de verificar que los mensajes enviados al servicio web son confiables.

Por último está la capa de Acceso a la base de datos, esta capa contiene el módulo de acceso a datos, cuya función es administrar los datos que se encuentran en la base de datos, que en el caso del sistema de promociones está basado en el gestor de base de datos *PostgreSQL*. Esta capa también tiene el módulo *hibernate* que es una herramienta de mapeo objeto-relacional para java, como su nombre lo indica, *hibernate* se encarga de relacionar las tablas de la base de datos con los modelos en java correspondientes.

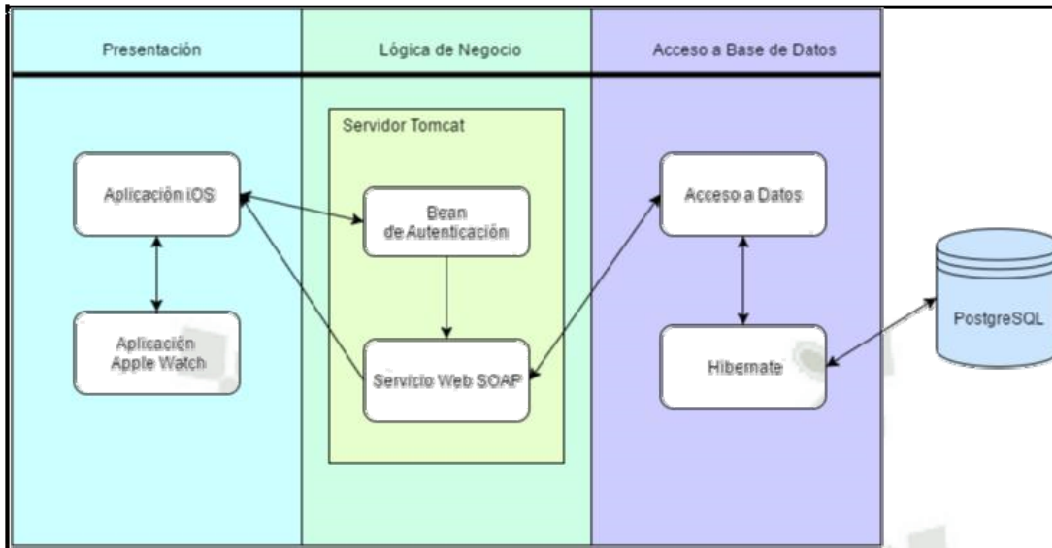


Figura 2: Diagrama de capas del sistema de promociones.

Fuente: Elaboración propia

La fase de extracción de requisitos de este sistema se realizó mediante una encuesta realizada a posibles usuarios de este sistema, de diferentes edades, gustos y lugares de residencia; en dicha encuesta se les preguntó lo que ellos pensaban sobre el sistema que se estaba pensando realizar, además de las características y las funcionalidades que ellos desearían que tuviera este sistema. Posteriormente aquellas funcionalidades más populares entre los encuestados pasaron a ser requerimientos que se plasmaron en el diseño del software.

Se determinaron también requerimientos no funcionales, conociendo el perfil del usuario y las circunstancias en la que se usará este sistema, estos requerimientos también pasaron a ser elementos importantes al realizar el diseño de este sistema.

El diseño del sistema de publicidad está basado en el modelo 4+1 propuesto por Kruchten (1995), este consta de 4 vistas la vista lógica, física, de desarrollo, de procesos y la vista de escenarios adicional. La vista escenarios es adicional porque las 4 vistas deben estar estrechamente relacionadas con esta vista que se encarga de mostrar los intereses de los *stakeholders*. Gracias a este modelo se pudo realizar diagramas que muestran los múltiples enfoques del sistema.

Tomando los requerimientos obtenidos se desarrolla la vista de escenario a través del diagrama de casos de uso, este diagrama se realizó primero, ya que los demás diagramas están basados en él. En los casos de uso se determina el actor del sistema que tiene que realizar cada tipo de acción en el software, las funciones que debían ser desarrolladas en el sistema y la manera en que cada caso de uso trabajaría en su entorno siguiendo el flujo del sistema.

Para la vista de desarrollo, se utilizó el diagrama de paquetes que se divide en las tres capas definidas por la arquitectura. Para la vista lógica se realizaron siete diagramas, el diagrama de clases del servicio web, el diagrama de clase de la aplicación y cuatro diagramas de secuencia que representan procesos importantes del sistema. Para la vista de procesos, se realizaron cuatro diagramas de actividades, que al igual que los diagramas de secuencia, representan procesos importantes del sistema. Para finalizar, en la vista física, se utilizó un diagrama de despliegue que muestre los dispositivos físicos del sistema y cómo se relacionan entre ellos.

La codificación de este proyecto fue hecha utilizando los fundamentos de la metodología de desarrollo de software llamada programación extrema. Durante toda la fase de desarrollo de este sistema se usó un repositorio de código en línea donde los miembros del equipo subían el avance que tenían durante un determinado lapso de tiempo, aproximadamente se realizaban dos actualizaciones de código al día. Cada actualización de código debía ser correctamente documentada para poder indicar al resto del equipo las características más importantes sobre el avance del proyecto.

En la fase de documentación se crearon manuales de usuario para la aplicación móvil, para esto se explicaron todas las funcionalidades que fueron implementadas en este sistema y la manera en la que el usuario podría utilizar cada una de estas, además se incluyeron imágenes de las interfaces de usuario para que de esta manera sea más fácil el uso de este software.

En el sistema se realizaron pruebas de validación y de verificación. Las pruebas de verificación fueron clasificadas en tres tipos, pruebas de ejecución, rendimiento y seguridad. Para las pruebas de ejecución se probó el sistema en diferentes escenarios y con diferentes datos para poder determinar que este trabaja de manera efectiva en cualquier circunstancia.

Las pruebas de rendimiento fueron realizadas basándose en el uso de recursos de hardware por parte del sistema para diferentes escenarios. Se probaron

tareas de bajo y alto costo de recursos y se determinó si estas medidas estaban dentro de los parámetros permitidos para garantizar un buen rendimiento por parte de la aplicación.

Las pruebas de seguridad verificaron que se realiza una correcta autenticación de los mensajes enviados entre la aplicación móvil y el servicio web, de esta manera se evitará que se produzcan ataques “Man in the middle”. Este proceso se realizó para poder garantizar que los datos de las transacciones de los usuarios están seguros y que otras personas no podrán acceder a su información.

Para poder hacer pruebas de validación de requerimientos del sistema se realizó una encuesta, la cual fue hecha a los usuarios que probaron el sistema y se preguntó si la aplicación cuenta con todas las funcionalidades necesarias para su fin y cuáles serían las nuevas funciones que a ellos les gustaría que fueran implementadas en el futuro para brindar una mejor experiencia.

En la fase de implementación, se utilizó la herramienta *Openshift*, que es una plataforma que permite utilizar un servidor compartido de forma gratuita, para el sistema de promociones se utilizó el servidor *Tomcat* versión 7, y el gestor de base de datos *PostgreSQL 9.2*. Para realizar la implementación se creó el dominio de la aplicación llamado *adsws*, el link del servicio web es “<http://webws-adsws.rhcloud.com/services>”, para poner el servicio web ya desarrollado anteriormente en el servidor de *Openshift*, se manda los archivos del servicio web al repositorio de la aplicación creada en este servidor, en caso de realizar cambios,

se actualiza nuevamente en el repositorio, se paraliza el servicio, se procesan los cambios y una vez que termina el servicio retoma su funcionamiento.



Capítulo 2: Documentación técnica

2.1 Plan del proyecto informático

Para la realización del proyecto en desarrollo se realizó un diagrama de Gantt en el cuál se podrá identificar las tareas para realizar y las fechas en las cuales se realizarán cada una de estas, cada tarea cuenta con sub tareas para realizar a continuación se determinarán los principales procesos del proyecto en desarrollo.

- Elaboración de Diseño y Arquitectura de Software
- Desarrollo del Sistema
- Pruebas
- Estudio de Resultados.

Dichas tareas serán mostradas en el siguiente diagrama de Gantt, además de la secuencia en la que se realizó cada proceso. También se determina la duración de cada proceso según el número de dedicación efectiva indicada en horas/hombre. Cabe recalcar que el proyecto será de una duración de 158 días.

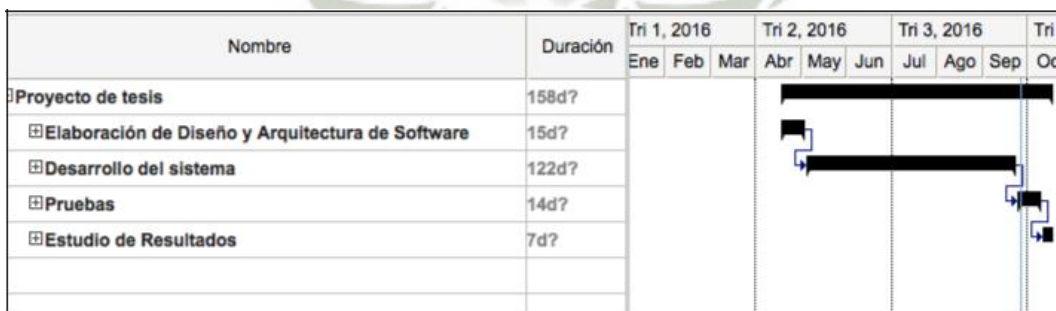


Figura 3: Diagrama de Gantt de los procesos de proyecto

Fuente: Elaboración propia

A continuación se muestra el plan de actividades de cada uno de los procesos determinados para el proyecto, para empezar se muestra el plan de actividades de la fase de Elaboración de Diseño y Arquitectura de Software, en la cual se detallan las subtarefas, además se indica el tiempo de duración de cada una de estas y la secuencia que cada una debe seguir. Con lo que se logra determinar que la fase de Elaboración de Diseño y Arquitectura de Software será de una duración de 15 días.

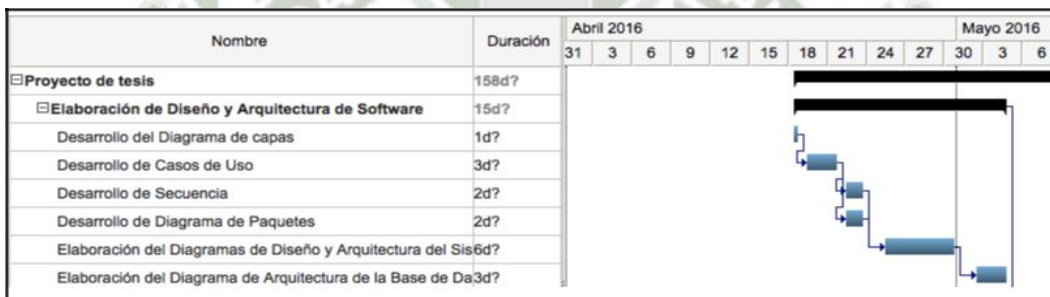


Figura 4: Diagrama de Gantt de la fase de elaboración de diseño y arquitectura de software
Fuente: Elaboración propia

Posteriormente se realizó el planeamiento de la fase de desarrollo del sistema, en la cual se propusieron dos subprocesos principales, el desarrollo de la aplicación móvil y el desarrollo del servicio web, ambos son importantes para el sistema desarrollado debido a que ambos tienen alta prioridad de uso. Se determinó las tareas de cada uno de estos subprocesos, además se asignan las fechas de inicio y fin, por lo que se determina que el desarrollo de sistema tendrá una duración de 17 semanas y 3 días de trabajo.

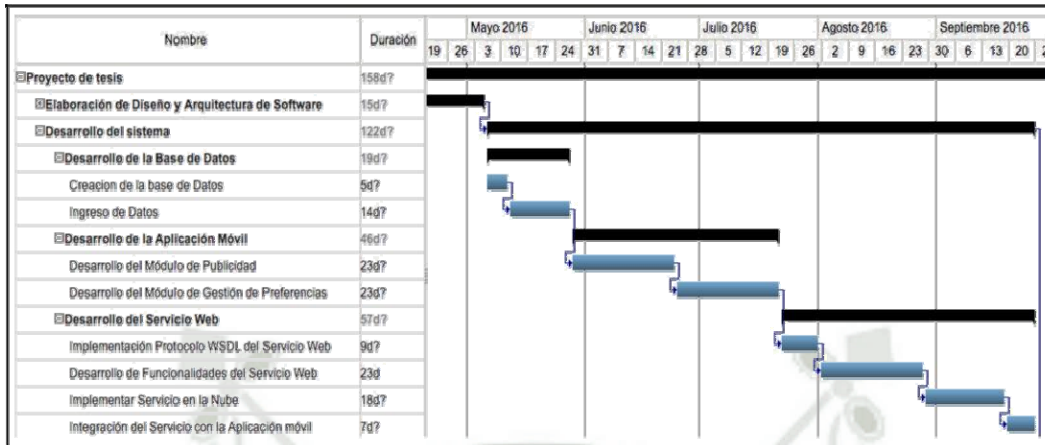


Figura 5: Diagrama de Gantt de la fase de desarrollo del sistema.

Fuente: Elaboración propia

Después de la fase de Desarrollo del sistema, se desarrolla la planeación del cronograma de la fase de pruebas, en la que se encontraron subtareas y se asignan las fechas necesarias para este proceso. Se determinó que la fase de pruebas del proyecto tendrá una duración de 2 semanas.

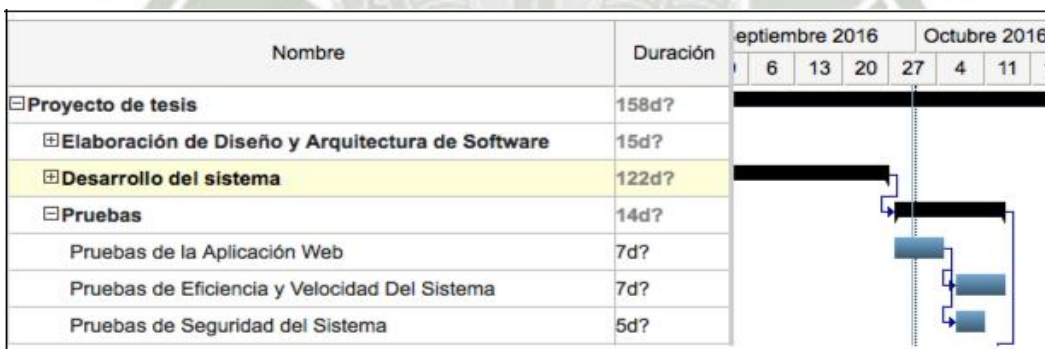


Figura 6: Diagrama de Gantt de la fase de pruebas del sistema.

Fuente: Elaboración propia

Posteriormente se determinó el cronograma para la fase de Estudio de Resultados, se identificaron las sub tareas que tiene este proceso y se indicó la

duración de cada una de estas y las fechas en la que se desarrollaron. Cabe recalcar que la fase de Estudio de Resultados tendrá una duración de una semana.

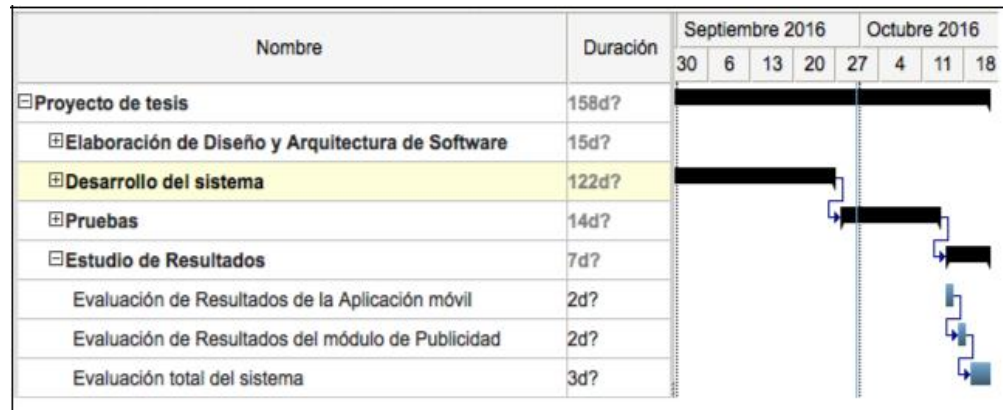


Figura 7 : Diagrama de Gantt de la fase de estudio de resultados del proyecto.
Fuente: Elaboración propia

2.2 Especificación de requisitos del proyecto de TIC

Para realizar la especificación de requisitos de software para este proyecto, se realizó una encuesta a los posibles usuarios debido a que este sistema tiene como usuarios principales al público en general. Esta encuesta lleva como nombre “Encuesta de Usuarios, Sistema de Publicidad en Dispositivos Móviles”.

Para hacer la elección de personas para encuentra se utilizó el método no probabilístico. Por lo tanto, para la realización de esta encuesta se utilizó personas allegadas a los miembros de equipo del proyecto, entre ellos familia, amigos, etc.

El tamaño del universo de la encuesta es de 72 personas. La edad de las personas que fueron encuestadas se dividió en 6 diferentes rangos, estos fueron delimitados según el poder adquisitivo que tienen las personas dentro de cada rango

de edad distinto, dado que esta aplicación se relaciona directamente con esta característica por ser una aplicación que brinda promociones.

Dentro de las respuestas a la primera pregunta se puede observar que el 73.6% de las personas encuestadas se encuentran entre los 19 y 25 años de edad, el 11.1% son personas entre 26 y 30 años, 5.6% corresponde a personas mayores a 45 años, 4.2% son personas entre 31 y 37 años, 2.8% son personas entre 38 y 45 años y por último 2.8% son personas entre 15 y 18 años. Con estos datos se puede observar que la mayoría de personas encuestadas son personas jóvenes.

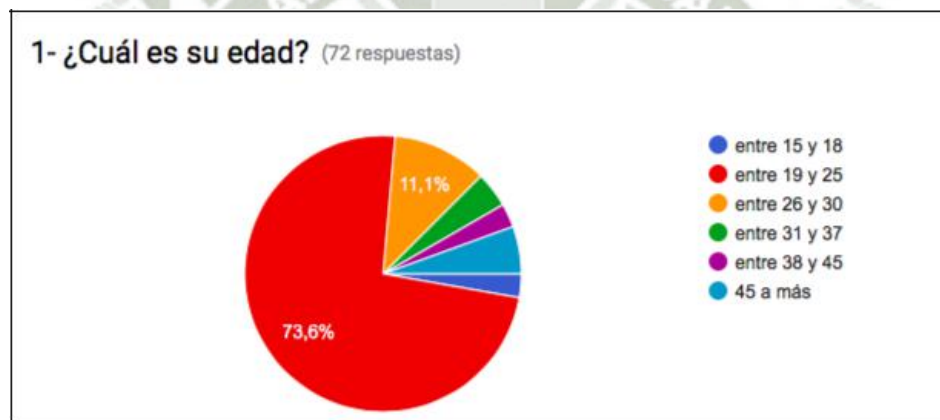


Figura 8: Resultados de la primera pregunta de la encuesta realizada a los posibles usuarios
Fuente: Elaboración propia

El siguiente aspecto a evaluar es la ciudad en la que habitan las personas encuestadas, para poder tener una idea de zonas en donde se podría implantar esta aplicación en el futuro. Bajo este criterio se pudo determinar que el 86.1% de las personas encuestadas son de la ciudad de Arequipa, 6.9% se encuentran en la ciudad de Lima, 2.8 viven en Trujillo, 1.4 son de la ciudad de Cusco y 2.8 son de ciudades que no se encuentran dentro de la opciones de la encuesta.

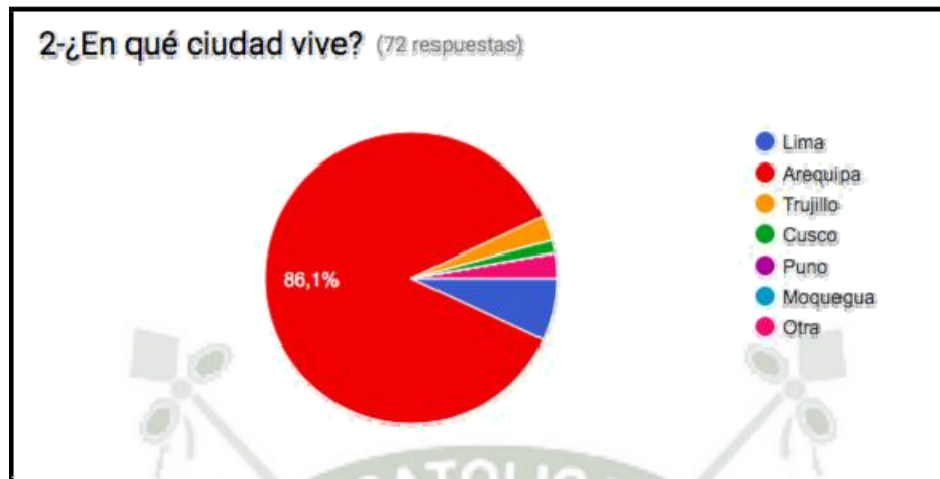


Figura 9: Resultados de la segunda pregunta de la encuesta realizada a los posibles usuarios
Fuente: Elaboración propia

El siguiente aspecto que se desea conocer sobre las personas encuestadas es un factor muy importante para esta aplicación, se busca conocer la frecuencia con la que estas personas acuden a las zonas comerciales de su ciudad en el lapso de un mes. Los resultados nos indican que el 48.6% de las personas encuestadas van a zonas comerciales de su ciudad entre 2 a 5 veces al mes, 29.2 % acuden a zona comerciales entre 5 y 10 veces, 12.5% acuden solo una vez al mes a zonas comerciales de su ciudad y 9.7% de las personas encuestadas van a zonas comerciales más de 10 veces al mes. Con los resultados anteriores se puede determinar que el 87.5% de las personas encuestadas van a zonas comerciales de su ciudad más de dos veces al mes y 38.9% de las personas van más de 5 veces al mes, por lo que se puede afirmar que las personas acuden a zonas comerciales con frecuencia, lo cual es un aspecto positivo para esta aplicación.



Figura 10 : Resultados de la tercera pregunta de la encuesta realizada a los posibles usuarios.

Fuente: Elaboración propia

La siguiente pregunta se relaciona directamente con el problema que se desea solucionar con esta aplicación, en esta pregunta se desea saber si las personas encuestadas reciben folletos de publicidad cuando van a zonas comerciales de su ciudad. El 84.7% de las personas encuestadas dicen que si reciben folletos de publicidad, mientras que el otro 15.3% indicó que no reciben este tipo de información, con esto se puede indicar que al 84.7% de las personas encuestadas les interesa recibir publicidad de los lugares que se encuentran cerca de ellos.

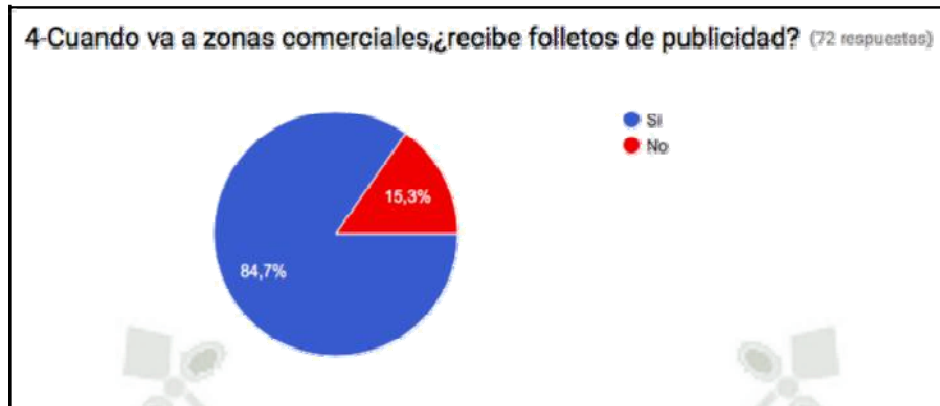


Figura 11: Resultados de la cuarta pregunta de la encuesta realizada a los posibles usuarios
Fuente: Elaboración propia

Siguiendo con la pregunta anterior, se les preguntó a los que respondieron afirmativamente la pregunta anterior, la relevancia de la información de los folletos que reciben dentro de la zona comercial en donde se encuentran. Las respuestas nos indican que el 64.5% de personas que respondieron esta pregunta piensan que esta información es poco relevante, el 21% de personas encuestadas indicaron que la información dentro de estos folletos es relevante, 11.3% de las personas que respondieron esta pregunta piensan que la información de los folletos que reciben es nada relevante para ellos, 1.6% de las personas encuestadas creen que la información dentro de los folletos de publicidad que reciben son muy relevantes y el 1.6% restante de la muestra no precisa una respuesta a esta pregunta. Estas respuestas nos indican que al 64.5% de las personas que reciben folletos en las zonas comerciales de su ciudad no le importa mucho la información que presentan los folletos al ser muy generalizados. Con esto se puede concluir que la publicidad en folletos es muy general por lo que esta tendría mayor eficacia si fuera focalizada

y entregada a los clientes basándose en sus preferencias. Por lo que la función de gestión de preferencias de los usuarios para brindarles promociones personalizadas, es muy importante para implementar dentro de la aplicación que se desarrolló.



Figura 12: Resultados de la quinta pregunta de la encuesta realizada a los posibles usuarios
Fuente: Elaboración propia

La siguiente pregunta ayuda a poder confirmar la conclusión anterior, se le pregunta a las personas encuestadas si les molesta ser impactados por publicidad que muchas veces no es útil para ellos, los resultados indican que el 73.6% de personas si sienten molestia por este tipo de publicidad que no es importante para ellos, el 20.8% indicaron que no les molesta recibir este tipo de promociones que no van de acuerdo a sus preferencias en muchos casos y 5.9% de los encuestados no precisan una respuesta a esta pregunta. Como se deseaba, se logró confirmar las conclusión que se tenían a las respuestas anteriores, la publicidad convencional al

no ser personalizada, puede llegar a molestar a los usuarios, lo cual es algo que no se desea con esta aplicación.



Figura 13 : Resultados de la sexta pregunta de la encuesta realizada a los posibles usuarios
Fuente: Elaboración propia

La siguiente pregunta ayuda a confirmar un requerimiento funcional de la aplicación el cual es determinar las preferencias de los usuarios y enviarles publicidad personalizada. A esta pregunta, el 93.1% de las personas encuestadas les gustaría recibir promociones según sus gustos pero al otro 6.9% de encuestados no les gustaría recibir promociones basándose en sus gustos y preferencias. Con esto se puede confirmar el requerimiento de basar la asignación de promociones en las preferencias de los usuario y hacerla personalizada. Además para las personas que no deseen que esta asignación se base en sus gustos, se podrá desactivar el uso de las preferencias de los usuarios dentro de las configuraciones de la aplicación.

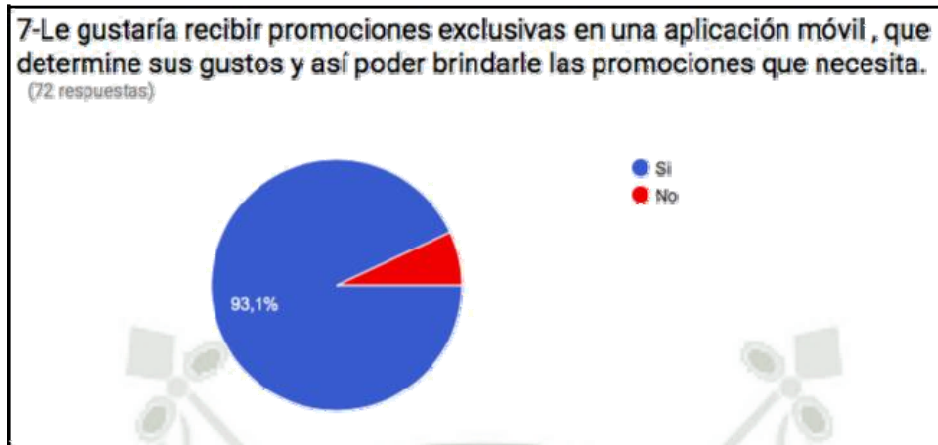


Figura 14 : Resultados de la séptima pregunta de la encuesta realizada a los posibles usuarios
Fuente: Elaboración propia

Con la siguiente pregunta, se desea saber cuáles son las funcionalidades que les gustaría a los usuarios para ser implementadas dentro de la aplicación. Los resultados nos demuestran que 67.1% de los encuestados desean que las promociones pudiesen ser guardadas dentro de una cuponera virtual dentro de la aplicación, el 62.9% de la muestra desea que se emitan alertas a los usuarios cuando una promoción guarda dentro de la cuponera virtual vaya a vencerse pronto. 57.1% de las personas que respondieron esta pregunta desea que se establezca un sistema de beneficios para la aplicación y brindar promociones exclusivas a personas que la usen con mayor frecuencia, 51.4% de las personas desean poder compartir las promociones de la aplicación con sus amigos para que estos también puedan utilizarlas, 34.3% de las personas encuestadas desean que esta aplicación tenga un vínculo con Facebook® y Twitter® para poder compartir las promociones utilizando dichas redes sociales. De las respuestas anteriores se extrajeron

funciones adicionales para desarrollar dentro de la aplicación, todas las funciones previamente descritas serán implementadas, por ser requerimientos confirmados por los usuarios encuestados.

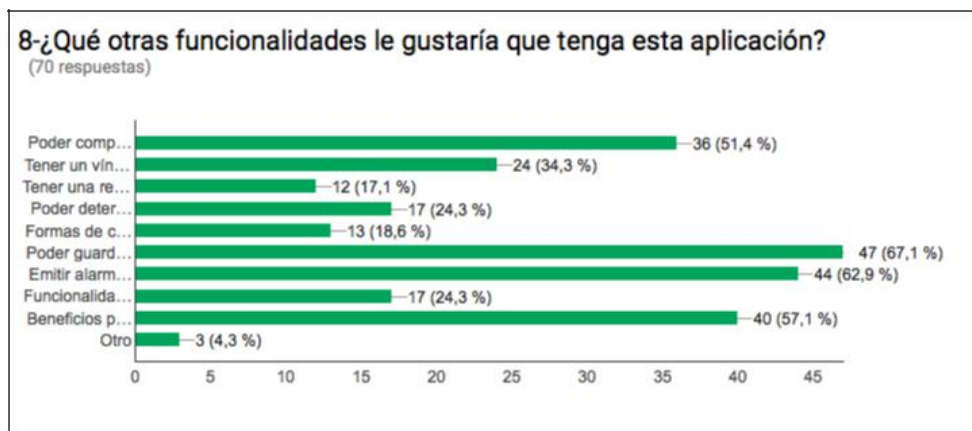


Figura 15 : Resultados de la octava pregunta de la encuesta realizada a los posibles usuarios
Fuente: Elaboración propia

La siguiente pregunta se realiza para poder evaluar uno de los objetivos principales de esta aplicación, el cual es ayudar al cuidado del medio ambiente al reducir el uso de papel en publicidad, a esta pregunta el 93.1% de los encuestados creen que esta aplicación si ayuda a contribuir con el cuidado del medio ambiente, 4.2% piensan que no se ayuda con el cuidado del medio ambiente y 2.8% no precisa una respuesta a esta pregunta. Con los datos previamente mostrados se puede concluir que los usuarios en su mayoría piensan que con esta aplicación se garantiza el desarrollo sostenible de la publicidad, lo cual es uno de los problemas principales que se desea solucionar.

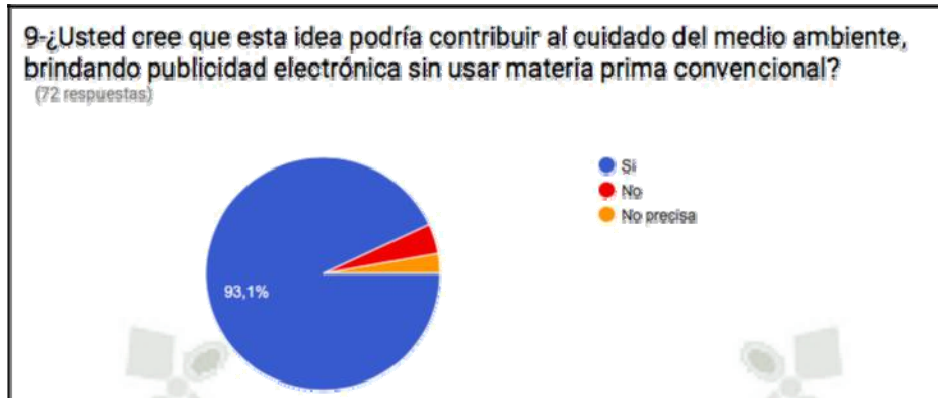


Figura 16: Resultados de la novena pregunta de la encuesta realizada a los posibles usuarios
Fuente: Elaboración propia

En la siguiente pregunta se busca saber la opinión del usuario en cuanto a una comparativa de esta aplicación frente a la publicidad convencional, el usuario debe responder en una escala del 1 al 5, siendo 1 sin importancia y 5 muy importante. Los resultados nos indican que el 48.6% de los encuestados piensan que este método de publicidad es importante frente a la publicidad convencional, el 29.2% de las personas encuestadas piensan que ambas formas de publicidad son similares y que ninguna es de mayor importancia, el 15.3% de los encuestados piensan que esta aplicación tiene mucha más importancia frente a la publicidad convencional y el otro 7% piensa que la publicidad en esta aplicación no es relevante frente a la publicidad convencional. Estos resultados pueden ayudar a confirmar la conclusión anterior, los usuarios en su mayoría creen que esta aplicación es más importante y trabaja de una mejor manera que la publicidad convencional, la cual se busca reducir.



Figura 17: Resultados de la décima pregunta de la encuesta realizada a los posibles usuarios
Fuente: Elaboración propia

Por último se le preguntó a las personas encuestadas la frecuencia con la que ellas piensan que utilizarían esta aplicación, a lo que el 55.6% de los encuestados respondieron que utilizarían esta aplicación frecuentemente, el 38.9% de las personas encuestadas respondieron que utilizarían esta aplicación a veces, 1.4% de los encuestados respondieron que utilizarían esta aplicación siempre, y 4.2% restante respondieron que no la utilizarían con frecuencia. Estos resultados ayudan a confirmar, que esta aplicación será utilizada con frecuencia por los usuarios, ya que, solo puede ser usada cuando estos se encuentran en zonas comerciales de su ciudad.

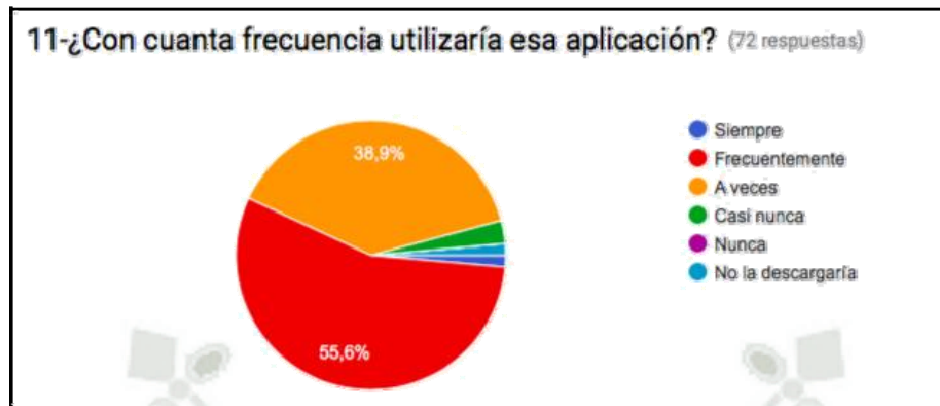


Figura 18: Resultados de la décimo primera pregunta de la encuesta realizada a los posibles usuarios

Fuente: Elaboración propia

Después de realizar la encuesta y de hacer el estudio de resultados, se realizó un diagrama de caso de uso con todos los requerimientos de software que se pudieron determinar gracias a la encuesta, estos requerimientos serán implementados por el equipo de desarrollo del sistema.

En la siguiente imagen se puede observar que el diagrama de casos de uso tiene dos tipos de actores el usuario final y el servicio web. El usuario final, es la persona que utiliza la aplicación móvil y la aplicación para el reloj inteligente, estas son las interfaces gráficas con las que los usuarios finales interactúan para realizar las acciones mostradas en el diagrama que se muestra a continuación. El servicio web es el encargado de realizar la lógica de negocio y es el intermediario entre las interfaces gráficas y la base de datos, en la siguiente imagen se muestran todos los procesos que el servicio web es capaz de realizar.



Figura 19: Diagrama de casos de uso del sistema de promociones
Fuente: Elaboración propia

A continuación se muestran las tablas de los actores que participan dentro de los casos de uso de esta aplicación en la cual se pueden observar mayores detalles y la manera en la que trabajan dentro del sistema. Posteriormente, se pueden ver las tablas de los casos de uso.

Tabla 2

Primer actor de diagrama de casos de uso

Actor	Usuario
Descripción	Es la persona encargada de utilizar este sistema de publicidad y activar las funciones con las que se cuenta para realizar el uso de las promociones que se brindan.
Responsabilidades	Gestionar promociones en la cuponera Solicitar Promociones Iniciar sesión Modificar Ubicación Modificar preferencias

Nota: Elaboración Propia

Tabla 3

Segundo actor de diagrama de casos de uso

Actor	Servicio web
Descripción	Es el servicio que tiene este sistema en la nube para poder realizar las tareas que el usuario le indica desde la aplicación móvil, además envía desde la base de datos todo lo que la aplicación le solicita.
Responsabilidades	Gestionar cupones de usuario Enviar cuponera según geolocalización Autenticar usuario Autenticar mensajes Enviar datos solicitados por el usuario Determinar zona comercial del usuario Guardar datos dentro de la base de datos

Nota: Elaboración Propia

Tabla 4

Descripción del primer caso de uso

Caso de Uso	#1 Iniciar Sesión										
Actor	Usuario										
Descripción	El usuario debe ingresar su nombre de usuario y contraseña dentro de la aplicación, posterior a eso debe presionar el botón Login para que el sistema valide sus datos.										
Flujo básico	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Usuario ingresa al sistema</td> </tr> <tr> <td>2</td> <td>Usuario ingresa nombre y contraseña</td> </tr> <tr> <td>3</td> <td>Los datos se validan</td> </tr> <tr> <td>4</td> <td>Permite acceso al usuario</td> </tr> </tbody> </table>	Paso	Acción	1	Usuario ingresa al sistema	2	Usuario ingresa nombre y contraseña	3	Los datos se validan	4	Permite acceso al usuario
Paso	Acción										
1	Usuario ingresa al sistema										
2	Usuario ingresa nombre y contraseña										
3	Los datos se validan										
4	Permite acceso al usuario										
Flujos alternos	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Usuario ingresa al sistema</td> </tr> <tr> <td>2</td> <td>Usuario ingresa nombre y contraseña</td> </tr> <tr> <td>3</td> <td>Los datos no se pueden validar</td> </tr> <tr> <td>4</td> <td>Se muestra un mensaje de error</td> </tr> </tbody> </table>	Paso	Acción	1	Usuario ingresa al sistema	2	Usuario ingresa nombre y contraseña	3	Los datos no se pueden validar	4	Se muestra un mensaje de error
Paso	Acción										
1	Usuario ingresa al sistema										
2	Usuario ingresa nombre y contraseña										
3	Los datos no se pueden validar										
4	Se muestra un mensaje de error										
Pre-condiciones	El usuario debe estar registrado en el sistema										
Post-condiciones	Los datos del usuario deben ser correctos										
Requerimientos trazados	Enviar datos al servicio web Autenticar Usuario										
Excepciones	El usuario ingresa datos incorrectos.										

Nota: Elaboración Propia

Tabla 5

Descripción del segundo caso de uso

Caso de Uso	#2 Actualizar ubicación										
Actor	Usuario										
Descripción	El usuario selecciona la opción de actualizar ubicación dentro de la aplicación móvil, esta ubicación se envía al servicio web.										
Flujo básico	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Usuario selecciona actualizar ubicación.</td> </tr> <tr> <td>2</td> <td>Se extrae la ubicación</td> </tr> <tr> <td>3</td> <td>Se envía ubicación al servicio web</td> </tr> <tr> <td>4</td> <td>Se actualiza ubicación</td> </tr> </tbody> </table>	Paso	Acción	1	Usuario selecciona actualizar ubicación.	2	Se extrae la ubicación	3	Se envía ubicación al servicio web	4	Se actualiza ubicación
Paso	Acción										
1	Usuario selecciona actualizar ubicación.										
2	Se extrae la ubicación										
3	Se envía ubicación al servicio web										
4	Se actualiza ubicación										
Flujos alternos	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Usuario selecciona actualizar ubicación.</td> </tr> <tr> <td>2</td> <td>Se extrae la ubicación</td> </tr> <tr> <td>3</td> <td>Se envía ubicación al servicio web</td> </tr> <tr> <td>4</td> <td>No se encuentra zona comercial</td> </tr> </tbody> </table>	Paso	Acción	1	Usuario selecciona actualizar ubicación.	2	Se extrae la ubicación	3	Se envía ubicación al servicio web	4	No se encuentra zona comercial
Paso	Acción										
1	Usuario selecciona actualizar ubicación.										
2	Se extrae la ubicación										
3	Se envía ubicación al servicio web										
4	No se encuentra zona comercial										
Pre-condiciones	Debe contar con la función de GPS activada										
Post-condiciones	Se encuentra dentro de una zona comercial										
Requerimientos trazados	Obtener zona comercial Enviar datos al servicio web										
Excepciones	No se puede obtener ubicación El usuario no está en una zona comercial										

Nota: Elaboración Propia

Tabla 6

Descripción del tercer caso de uso

Caso de Uso	#3 Solicitar promociones con preferencias										
Actor	Usuario										
Descripción	El usuario ingresa a las promociones dentro de su zona comercial y tiene la función de uso de preferencias al mismo tiempo, por lo que el servicio web envía solo las promociones que le interesen										
Flujo básico	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Usuario ingresa a la promociones dentro de su misma zona comercial.</td> </tr> <tr> <td>2</td> <td>Se solicita ubicación</td> </tr> <tr> <td>3</td> <td>Se solicita datos al servidor, enviando ubicación del usuario y solicitando que sean promociones basadas en preferencias</td> </tr> <tr> <td>4</td> <td>Se muestra datos recibidos del servicio web, dentro de la aplicación</td> </tr> </tbody> </table>	Paso	Acción	1	Usuario ingresa a la promociones dentro de su misma zona comercial.	2	Se solicita ubicación	3	Se solicita datos al servidor, enviando ubicación del usuario y solicitando que sean promociones basadas en preferencias	4	Se muestra datos recibidos del servicio web, dentro de la aplicación
Paso	Acción										
1	Usuario ingresa a la promociones dentro de su misma zona comercial.										
2	Se solicita ubicación										
3	Se solicita datos al servidor, enviando ubicación del usuario y solicitando que sean promociones basadas en preferencias										
4	Se muestra datos recibidos del servicio web, dentro de la aplicación										
Flujos alternos	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Usuario ingresa a la promociones dentro de su misma zona comercial.</td> </tr> <tr> <td>2</td> <td>Se solicita ubicación</td> </tr> <tr> <td>3</td> <td>Se solicita datos al servidor, enviando ubicación del usuario</td> </tr> <tr> <td>4</td> <td>Se muestra alerta informando que no se cuenta con promociones</td> </tr> </tbody> </table>	Paso	Acción	1	Usuario ingresa a la promociones dentro de su misma zona comercial.	2	Se solicita ubicación	3	Se solicita datos al servidor, enviando ubicación del usuario	4	Se muestra alerta informando que no se cuenta con promociones
Paso	Acción										
1	Usuario ingresa a la promociones dentro de su misma zona comercial.										
2	Se solicita ubicación										
3	Se solicita datos al servidor, enviando ubicación del usuario										
4	Se muestra alerta informando que no se cuenta con promociones										
Pre-condiciones	Debe contar con la ubicación del usuario Debe tener opción de promociones basadas en preferencias activa.										
Post-condiciones	Debe seleccionar alguna de las promociones										
Requerimientos trazados	Solicitar promociones con preferencias Recibir promociones desde el servicio web Utilizar preferencias del usuario										
Excepciones	No se puede obtener ubicación del usuario No se conocen las preferencias del usuario										

Nota: Elaboración Propia

Tabla 7

Descripción del cuarto caso de uso

Caso de Uso	#4 Solicitar promociones generales										
Actor	Usuario										
Descripción	El usuario ingresa a las promociones dentro de su zona comercial y tiene desactivada la función de uso de preferencias al mismo tiempo										
Flujo básico	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Usuario ingresa a la promociones dentro de su misma zona comercial.</td> </tr> <tr> <td>2</td> <td>Se solicita ubicación</td> </tr> <tr> <td>3</td> <td>Se solicita datos al servidor, enviando ubicación del usuario, indicando que se necesita promociones generales.</td> </tr> <tr> <td>4</td> <td>Se muestra datos recibidos del servicio web, dentro de la aplicación</td> </tr> </tbody> </table>	Paso	Acción	1	Usuario ingresa a la promociones dentro de su misma zona comercial.	2	Se solicita ubicación	3	Se solicita datos al servidor, enviando ubicación del usuario, indicando que se necesita promociones generales.	4	Se muestra datos recibidos del servicio web, dentro de la aplicación
Paso	Acción										
1	Usuario ingresa a la promociones dentro de su misma zona comercial.										
2	Se solicita ubicación										
3	Se solicita datos al servidor, enviando ubicación del usuario, indicando que se necesita promociones generales.										
4	Se muestra datos recibidos del servicio web, dentro de la aplicación										
Flujos alternos	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Usuario ingresa a la promociones dentro de su misma zona comercial.</td> </tr> <tr> <td>2</td> <td>Se solicita ubicación</td> </tr> <tr> <td>3</td> <td>Se solicita datos al servidor, enviando ubicación del usuario</td> </tr> <tr> <td>4</td> <td>Se muestra alerta informando que no se cuenta con promociones</td> </tr> </tbody> </table>	Paso	Acción	1	Usuario ingresa a la promociones dentro de su misma zona comercial.	2	Se solicita ubicación	3	Se solicita datos al servidor, enviando ubicación del usuario	4	Se muestra alerta informando que no se cuenta con promociones
Paso	Acción										
1	Usuario ingresa a la promociones dentro de su misma zona comercial.										
2	Se solicita ubicación										
3	Se solicita datos al servidor, enviando ubicación del usuario										
4	Se muestra alerta informando que no se cuenta con promociones										
Pre-condiciones	Debe contar con la ubicación del usuario Debe tener opción de promociones basadas en preferencias desactiva.										
Post-condiciones	Debe seleccionar alguna de las promociones										
Requerimientos trazados	Solicitar promociones sin preferencias Recibir promociones desde el servicio web Utilizar preferencias del usuario										
Excepciones	No se puede obtener ubicación del usuario										

Nota: Elaboración Propia

Tabla 8

Descripción del quinto caso de uso

Caso de Uso	#5 Agregar cupón a la cuponera												
Actor	Usuario												
Descripción	El usuario selecciona una determina promoción y después selecciona la opción de agregarla a la cuponera.												
Flujo básico	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Usuario ingresa a la promociones dentro de su misma zona comercial.</td> </tr> <tr> <td>2</td> <td>Selecciona una promoción</td> </tr> <tr> <td>3</td> <td>Selecciona la opción de agregar cupón dentro de la cuponera virtual</td> </tr> <tr> <td>4</td> <td>Se envía los datos al servicio web</td> </tr> <tr> <td>5</td> <td>Se muestra la confirmación de tarea realizada</td> </tr> </tbody> </table>	Paso	Acción	1	Usuario ingresa a la promociones dentro de su misma zona comercial.	2	Selecciona una promoción	3	Selecciona la opción de agregar cupón dentro de la cuponera virtual	4	Se envía los datos al servicio web	5	Se muestra la confirmación de tarea realizada
Paso	Acción												
1	Usuario ingresa a la promociones dentro de su misma zona comercial.												
2	Selecciona una promoción												
3	Selecciona la opción de agregar cupón dentro de la cuponera virtual												
4	Se envía los datos al servicio web												
5	Se muestra la confirmación de tarea realizada												
Flujos alternos	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Usuario ingresa a la promociones dentro de su misma zona comercial.</td> </tr> <tr> <td>2</td> <td>Selecciona una promoción</td> </tr> <tr> <td>3</td> <td>Selecciona la opción de agregar cupón dentro de la cuponera virtual</td> </tr> <tr> <td>4</td> <td>Se envía los datos al servicio web</td> </tr> <tr> <td>5</td> <td>Se muestra que no fue posible realizar la acción</td> </tr> </tbody> </table>	Paso	Acción	1	Usuario ingresa a la promociones dentro de su misma zona comercial.	2	Selecciona una promoción	3	Selecciona la opción de agregar cupón dentro de la cuponera virtual	4	Se envía los datos al servicio web	5	Se muestra que no fue posible realizar la acción
Paso	Acción												
1	Usuario ingresa a la promociones dentro de su misma zona comercial.												
2	Selecciona una promoción												
3	Selecciona la opción de agregar cupón dentro de la cuponera virtual												
4	Se envía los datos al servicio web												
5	Se muestra que no fue posible realizar la acción												
Pre-condiciones	Debe contar con promociones dentro de su misma zona No ingresar la misma promociones previamente en la cuponera												
Post-condiciones	No ingresar nuevamente dicha promoción en la cuponera												
Requerimientos trazados	Guardar promoción en cuponera Enviar datos del cupón en cuponera Mostrar respuesta del servicio web en una alerta.												
Excepciones	El cupón fue ingresado dentro de la cuponera virtual previamente.												

Nota: Elaboración Propia

Tabla 9

Descripción del sexto caso de uso

Caso de Uso	#6 Eliminar cupón de la cuponera												
Actor	Usuario												
Descripción	El usuario selecciona una promoción dentro de su cuponera y la elimina.												
Flujo básico	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Usuario ingresa a la promociones dentro de su misma zona comercial.</td> </tr> <tr> <td>2</td> <td>Selecciona una promoción</td> </tr> <tr> <td>3</td> <td>Selecciona la opción de eliminar cupón de la cuponera virtual</td> </tr> <tr> <td>4</td> <td>Se envía los datos al servicio web</td> </tr> <tr> <td>5</td> <td>Se muestra la confirmación de tarea realizada</td> </tr> </tbody> </table>	Paso	Acción	1	Usuario ingresa a la promociones dentro de su misma zona comercial.	2	Selecciona una promoción	3	Selecciona la opción de eliminar cupón de la cuponera virtual	4	Se envía los datos al servicio web	5	Se muestra la confirmación de tarea realizada
Paso	Acción												
1	Usuario ingresa a la promociones dentro de su misma zona comercial.												
2	Selecciona una promoción												
3	Selecciona la opción de eliminar cupón de la cuponera virtual												
4	Se envía los datos al servicio web												
5	Se muestra la confirmación de tarea realizada												
Flujos alternos	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Usuario ingresa a la promociones dentro de su misma zona comercial.</td> </tr> <tr> <td>2</td> <td>Selecciona una promoción</td> </tr> <tr> <td>3</td> <td>Selecciona la opción de agregar cupón dentro de la cuponera virtual</td> </tr> <tr> <td>4</td> <td>Se envía los datos al servicio web</td> </tr> <tr> <td>5</td> <td>Se muestra que no fue posible realizar la acción</td> </tr> </tbody> </table>	Paso	Acción	1	Usuario ingresa a la promociones dentro de su misma zona comercial.	2	Selecciona una promoción	3	Selecciona la opción de agregar cupón dentro de la cuponera virtual	4	Se envía los datos al servicio web	5	Se muestra que no fue posible realizar la acción
Paso	Acción												
1	Usuario ingresa a la promociones dentro de su misma zona comercial.												
2	Selecciona una promoción												
3	Selecciona la opción de agregar cupón dentro de la cuponera virtual												
4	Se envía los datos al servicio web												
5	Se muestra que no fue posible realizar la acción												
Pre-condiciones	Debe contar con promociones dentro de su misma zona Debe tener promociones dentro de la cuponera												
Post-condiciones	No aparece la promoción dentro de la cuponera virtual												
Requerimientos trazados	Eliminar promoción de la cuponera Enviar datos del cupón en cuponera Mostrar respuesta del servicio web en una alerta.												
Excepciones	No hay promociones que eliminar No se seleccionó una promoción												

Nota: Elaboración Propia

Tabla 10

Descripción del séptimo caso de uso

Caso de Uso	#7 Utilizar cupón																
Actor	Usuario																
Descripción	El usuario selecciona una promoción dentro de su cuponera y la usa, generando un código QR, posteriormente elimina dicha promoción de la cuponera																
Flujo básico	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Usuario ingresa a la promociones dentro de su misma zona comercial.</td> </tr> <tr> <td>2</td> <td>Selecciona una promoción</td> </tr> <tr> <td>3</td> <td>Selecciona la opción de usar cupón</td> </tr> <tr> <td>4</td> <td>Se envía los datos al servicio web</td> </tr> <tr> <td>5</td> <td>Se genera el código QR de la promoción</td> </tr> <tr> <td>6</td> <td>Se guarda el uso de promoción en la base de datos</td> </tr> <tr> <td>7</td> <td>Se elimina la promoción de la cuponera virtual</td> </tr> </tbody> </table>	Paso	Acción	1	Usuario ingresa a la promociones dentro de su misma zona comercial.	2	Selecciona una promoción	3	Selecciona la opción de usar cupón	4	Se envía los datos al servicio web	5	Se genera el código QR de la promoción	6	Se guarda el uso de promoción en la base de datos	7	Se elimina la promoción de la cuponera virtual
Paso	Acción																
1	Usuario ingresa a la promociones dentro de su misma zona comercial.																
2	Selecciona una promoción																
3	Selecciona la opción de usar cupón																
4	Se envía los datos al servicio web																
5	Se genera el código QR de la promoción																
6	Se guarda el uso de promoción en la base de datos																
7	Se elimina la promoción de la cuponera virtual																
Flujos alternos	No existen flujos alternos																
Pre-condiciones	Debe contar con promociones dentro de su misma zona Debe seleccionar la promoción que desea usar																
Post-condiciones	La promoción desaparece de la cuponera																
Requerimientos trazados	Eliminar promoción de la cuponera Generar código QR Guardar uso de la promoción en la base de datos Enviar datos del cupón en cuponera Mostrar respuesta del servicio web en una alerta.																
Excepciones	No se seleccionó una promoción																

Nota: Elaboración Propia

Tabla 11

Descripción del octavo caso de uso

Caso de Uso	#8 Modificar preferencias														
Actor	Usuario														
Descripción	El usuario selecciona la opción de actualizar preferencias y selecciona las categorías que son de su preferencia y las guarda														
Flujo básico	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Usuario ingresa a las configuraciones</td> </tr> <tr> <td>2</td> <td>Selecciona la opción de actualizar preferencias</td> </tr> <tr> <td>3</td> <td>Selecciona las categorías que son de su preferencia</td> </tr> <tr> <td>4</td> <td>Aprueba la actualización de preferencias</td> </tr> <tr> <td>5</td> <td>Se envía los datos al servicio web</td> </tr> <tr> <td>6</td> <td>Se guardan sus preferencias en la base de datos</td> </tr> </tbody> </table>	Paso	Acción	1	Usuario ingresa a las configuraciones	2	Selecciona la opción de actualizar preferencias	3	Selecciona las categorías que son de su preferencia	4	Aprueba la actualización de preferencias	5	Se envía los datos al servicio web	6	Se guardan sus preferencias en la base de datos
Paso	Acción														
1	Usuario ingresa a las configuraciones														
2	Selecciona la opción de actualizar preferencias														
3	Selecciona las categorías que son de su preferencia														
4	Aprueba la actualización de preferencias														
5	Se envía los datos al servicio web														
6	Se guardan sus preferencias en la base de datos														
Flujos alternos	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Usuario ingresa a las configuraciones</td> </tr> <tr> <td>2</td> <td>Selecciona la opción de actualizar preferencias</td> </tr> <tr> <td>3</td> <td>No se modifican las preferencias</td> </tr> <tr> <td>4</td> <td>Aprueba la actualización de preferencias</td> </tr> <tr> <td>5</td> <td>Se envía los datos al servicio web</td> </tr> <tr> <td>6</td> <td>No se guardan sus preferencias en la base de datos</td> </tr> </tbody> </table>	Paso	Acción	1	Usuario ingresa a las configuraciones	2	Selecciona la opción de actualizar preferencias	3	No se modifican las preferencias	4	Aprueba la actualización de preferencias	5	Se envía los datos al servicio web	6	No se guardan sus preferencias en la base de datos
Paso	Acción														
1	Usuario ingresa a las configuraciones														
2	Selecciona la opción de actualizar preferencias														
3	No se modifican las preferencias														
4	Aprueba la actualización de preferencias														
5	Se envía los datos al servicio web														
6	No se guardan sus preferencias en la base de datos														
Pre-condiciones	Activar opción de usar preferencias Seleccionar categorías de preferencia														
Post-condiciones	Cambian las preferencias del usuario														
Requerimientos trazados	Actualizar preferencias Guardar preferencias en la base de datos Enviar datos al servicio web Mostrar respuesta del servicio web en una alerta.														
Excepciones	No se actualizan las promociones														

Nota: Elaboración Propia

Tabla 12

Descripción del noveno caso de uso

Caso de Uso	#9 Guardar preferencias														
Actor	Servicio web														
Descripción	El servicio recibe las preferencias de un determinado usuario y guarda las nuevas preferencias dentro de la base de datos														
Flujo básico	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Usuario actualiza preferencias en la aplicación</td> </tr> <tr> <td>2</td> <td>Se envían preferencias al servicio web</td> </tr> <tr> <td>3</td> <td>El servicio guarda las nuevas preferencias en la base de datos</td> </tr> <tr> <td>4</td> <td>Se envía un mensaje de confirmación de la acción a la aplicación</td> </tr> <tr> <td>5</td> <td>La aplicación muestra mensaje de confirmación</td> </tr> <tr> <td>6</td> <td>Se muestran las nuevas preferencias del usuario</td> </tr> </tbody> </table>	Paso	Acción	1	Usuario actualiza preferencias en la aplicación	2	Se envían preferencias al servicio web	3	El servicio guarda las nuevas preferencias en la base de datos	4	Se envía un mensaje de confirmación de la acción a la aplicación	5	La aplicación muestra mensaje de confirmación	6	Se muestran las nuevas preferencias del usuario
Paso	Acción														
1	Usuario actualiza preferencias en la aplicación														
2	Se envían preferencias al servicio web														
3	El servicio guarda las nuevas preferencias en la base de datos														
4	Se envía un mensaje de confirmación de la acción a la aplicación														
5	La aplicación muestra mensaje de confirmación														
6	Se muestran las nuevas preferencias del usuario														
Flujos alternos	No existen flujos alternos														
Pre-condiciones	Recibir preferencias de la aplicación														
Post-condiciones	Cambian las preferencias del usuario														
Requerimientos trazados	Guardar preferencias en la base de datos Mostrar nuevas preferencias del usuario Enviar datos al servicio web Mostrar respuesta del servicio web en una alerta.														
Excepciones	No se actualizan las promociones														

Nota: Elaboración Propia

Tabla 13

Descripción del décimo caso de uso

Caso de Uso	#10 Eliminar preferencias														
Actor	Servicio web														
Descripción	El servicio recibe las preferencias de un determinado usuario y elimina las preferencias que fueron removidas por el usuario dentro de la base de datos														
Flujo básico	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Usuario actualiza preferencias en la aplicación</td> </tr> <tr> <td>2</td> <td>Se envían preferencias al servicio web</td> </tr> <tr> <td>3</td> <td>El servicio elimina de la base de datos las preferencias que fueron removidas en la aplicación</td> </tr> <tr> <td>4</td> <td>Se envía un mensaje de confirmación de la acción a la aplicación</td> </tr> <tr> <td>5</td> <td>La aplicación muestra mensaje de confirmación</td> </tr> <tr> <td>6</td> <td>Se muestran las nuevas preferencias del usuario</td> </tr> </tbody> </table>	Paso	Acción	1	Usuario actualiza preferencias en la aplicación	2	Se envían preferencias al servicio web	3	El servicio elimina de la base de datos las preferencias que fueron removidas en la aplicación	4	Se envía un mensaje de confirmación de la acción a la aplicación	5	La aplicación muestra mensaje de confirmación	6	Se muestran las nuevas preferencias del usuario
Paso	Acción														
1	Usuario actualiza preferencias en la aplicación														
2	Se envían preferencias al servicio web														
3	El servicio elimina de la base de datos las preferencias que fueron removidas en la aplicación														
4	Se envía un mensaje de confirmación de la acción a la aplicación														
5	La aplicación muestra mensaje de confirmación														
6	Se muestran las nuevas preferencias del usuario														
Flujos alternos	No existen flujos alternos														
Pre-condiciones	Eliminar una preferencia anteriormente guardada														
Post-condiciones	Cambian las preferencias del usuario														
Requerimientos trazados	Eliminar preferencias en la base de datos Mostrar nuevas preferencias del usuario Mostrar respuesta del servicio web en una alerta.														
Excepciones	No se eliminan preferencias														

Nota: Elaboración Propia

Tabla 14

Descripción del décimo primer caso de uso

Caso de Uso	#11 Guardar ubicación														
Actor	Servicio web														
Descripción	El servicio recibe nueva ubicación de un determinado usuario, determina en qué zona comercial se encuentra y guarda su ubicación dentro de la base de datos.														
Flujo básico	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Usuario actualiza su ubicación</td> </tr> <tr> <td>2</td> <td>Se envían nueva ubicación al servicio web</td> </tr> <tr> <td>3</td> <td>Se determina la zona comercial donde se encuentra el usuario</td> </tr> <tr> <td>4</td> <td>Se guarda la ubicación en la base de datos</td> </tr> <tr> <td>5</td> <td>Se envía confirmación de actualización a la aplicación</td> </tr> <tr> <td>6</td> <td>Se muestran datos de nueva ubicación en la aplicación</td> </tr> </tbody> </table>	Paso	Acción	1	Usuario actualiza su ubicación	2	Se envían nueva ubicación al servicio web	3	Se determina la zona comercial donde se encuentra el usuario	4	Se guarda la ubicación en la base de datos	5	Se envía confirmación de actualización a la aplicación	6	Se muestran datos de nueva ubicación en la aplicación
Paso	Acción														
1	Usuario actualiza su ubicación														
2	Se envían nueva ubicación al servicio web														
3	Se determina la zona comercial donde se encuentra el usuario														
4	Se guarda la ubicación en la base de datos														
5	Se envía confirmación de actualización a la aplicación														
6	Se muestran datos de nueva ubicación en la aplicación														
Flujos alternos	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Usuario actualiza su ubicación</td> </tr> <tr> <td>2</td> <td>Se envían nueva ubicación al servicio web</td> </tr> <tr> <td>3</td> <td>No se puede determinar la zona comercial donde se encuentra el usuario</td> </tr> <tr> <td>4</td> <td>Se envía alerta de no poder actualizar la ubicación</td> </tr> <tr> <td>5</td> <td>Se muestra alerta en la aplicación</td> </tr> </tbody> </table>	Paso	Acción	1	Usuario actualiza su ubicación	2	Se envían nueva ubicación al servicio web	3	No se puede determinar la zona comercial donde se encuentra el usuario	4	Se envía alerta de no poder actualizar la ubicación	5	Se muestra alerta en la aplicación		
Paso	Acción														
1	Usuario actualiza su ubicación														
2	Se envían nueva ubicación al servicio web														
3	No se puede determinar la zona comercial donde se encuentra el usuario														
4	Se envía alerta de no poder actualizar la ubicación														
5	Se muestra alerta en la aplicación														
Pre-condiciones	Actualizar ubicación del usuario														
Post-condiciones	Cambia la zona comercial donde se encuentra el usuario														
Requerimientos trazados	Encontrar ubicación del usuario Actualizar ubicación en la base de datos Enviar datos al servicio web Mostrar respuesta del servicio web en una alerta.														
Excepciones	El usuario no cambia de ubicación														

Nota: Elaboración Propia

Tabla 15

Descripción del décimo segundo caso de uso

Caso de Uso	#12 Determinar ubicación zona comercial														
Actor	Servicio web														
Descripción	El servicio recibe nueva ubicación de un determinado usuario, determina en qué zona comercial se encuentra.														
Flujo básico	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Usuario actualiza su ubicación</td> </tr> <tr> <td>2</td> <td>Se envían nueva ubicación al servicio web</td> </tr> <tr> <td>3</td> <td>Se buscan todas las zonas comerciales</td> </tr> <tr> <td>4</td> <td>Se determinan qué zonas comerciales están más cerca al usuario</td> </tr> <tr> <td>5</td> <td>Se determina si el usuario está dentro del radio de la zona comercial más cercana a él</td> </tr> <tr> <td>6</td> <td>Se guarda la ubicación en la base de datos</td> </tr> </tbody> </table>	Paso	Acción	1	Usuario actualiza su ubicación	2	Se envían nueva ubicación al servicio web	3	Se buscan todas las zonas comerciales	4	Se determinan qué zonas comerciales están más cerca al usuario	5	Se determina si el usuario está dentro del radio de la zona comercial más cercana a él	6	Se guarda la ubicación en la base de datos
Paso	Acción														
1	Usuario actualiza su ubicación														
2	Se envían nueva ubicación al servicio web														
3	Se buscan todas las zonas comerciales														
4	Se determinan qué zonas comerciales están más cerca al usuario														
5	Se determina si el usuario está dentro del radio de la zona comercial más cercana a él														
6	Se guarda la ubicación en la base de datos														
Flujos alternos	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Usuario actualiza su ubicación</td> </tr> <tr> <td>2</td> <td>Se envían nueva ubicación al servicio web</td> </tr> <tr> <td>3</td> <td>Se buscan todas las zonas comerciales</td> </tr> <tr> <td>4</td> <td>Se determinan qué zonas comerciales están más cerca al usuario</td> </tr> <tr> <td>5</td> <td>Se determina que el usuario no está dentro del radio de la zona comercial más cercana a él</td> </tr> <tr> <td>6</td> <td>Se envía un mensaje diciendo que el usuario no se encuentra dentro de una zona comercial</td> </tr> </tbody> </table>	Paso	Acción	1	Usuario actualiza su ubicación	2	Se envían nueva ubicación al servicio web	3	Se buscan todas las zonas comerciales	4	Se determinan qué zonas comerciales están más cerca al usuario	5	Se determina que el usuario no está dentro del radio de la zona comercial más cercana a él	6	Se envía un mensaje diciendo que el usuario no se encuentra dentro de una zona comercial
Paso	Acción														
1	Usuario actualiza su ubicación														
2	Se envían nueva ubicación al servicio web														
3	Se buscan todas las zonas comerciales														
4	Se determinan qué zonas comerciales están más cerca al usuario														
5	Se determina que el usuario no está dentro del radio de la zona comercial más cercana a él														
6	Se envía un mensaje diciendo que el usuario no se encuentra dentro de una zona comercial														
Pre-condiciones	Actualizar ubicación del usuario Usuario debe estar dentro del radio de una zona comercial														
Post-condiciones	Cambia la zona comercial donde se encuentra el usuario														
Requerimientos trazados	Encontrar ubicación del usuario Determinar la zona comercial donde se encuentra el usuario Guarda ubicación en la base de datos Enviar datos al servicio web Mostrar respuesta del servicio web en una alerta.														
Excepciones	El usuario no se encuentra en el radio de una zona comercial														

Nota: Elaboración Propia

Tabla 16

Descripción del décimo tercer caso de uso

Caso de Uso	#13 Enviar tiendas según geolocalización										
Actor	Servicio web										
Descripción	El servicio web recibe una solicitud de recepción de tiendas desde la aplicación. El servicio web busca la ubicación del usuario, y manda la información de las tiendas que se encuentran dentro de la misma zona comercial										
Flujo básico	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>La aplicación solicita tiendas dentro de la misma zona comercial del usuario</td> </tr> <tr> <td>2</td> <td>El servicio web solicita la ubicación del usuario y determina la zona comercial en la que se encuentra</td> </tr> <tr> <td>3</td> <td>El servicio web envía la información de las tiendas que se encuentran en esa misma zona comercial.</td> </tr> <tr> <td>4</td> <td>Se muestra la información de los establecimientos, dentro de la aplicación.</td> </tr> </tbody> </table>	Paso	Acción	1	La aplicación solicita tiendas dentro de la misma zona comercial del usuario	2	El servicio web solicita la ubicación del usuario y determina la zona comercial en la que se encuentra	3	El servicio web envía la información de las tiendas que se encuentran en esa misma zona comercial.	4	Se muestra la información de los establecimientos, dentro de la aplicación.
Paso	Acción										
1	La aplicación solicita tiendas dentro de la misma zona comercial del usuario										
2	El servicio web solicita la ubicación del usuario y determina la zona comercial en la que se encuentra										
3	El servicio web envía la información de las tiendas que se encuentran en esa misma zona comercial.										
4	Se muestra la información de los establecimientos, dentro de la aplicación.										
Flujos alternos	No existen flujos alternos										
Pre-condiciones	Se debe conocer la ubicación del usuario. Deben existir establecimientos dentro de la zona comercial										
Post-condiciones	Se muestran los establecimientos dentro de la zona comercial										
Requerimientos trazados	Determinar tiendas que se encuentran dentro de la misma zona marcial que el usuario Enviar datos de las tiendas desde el servicio web hasta la aplicación										
Excepciones	No existen establecimientos dentro de la zona comercial.										

Nota: Elaboración Propia

Tabla 17

Descripción del décimo cuarto caso de uso

Caso de Uso	#14 Enviar promociones										
Actor	Servicio web										
Descripción	El servicio web recibe una solicitud de recepción de promociones desde la aplicación. El servicio web busca las promociones de la tienda en la que se encuentra el usuario y las envía.										
Flujo básico	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>La aplicación solicita tiendas dentro de la misma zona comercial del usuario</td> </tr> <tr> <td>2</td> <td>El servicio web solicita establecimiento en el que se encuentra el usuario</td> </tr> <tr> <td>3</td> <td>El servicio web envía la información de las promociones que solicitó el usuario</td> </tr> <tr> <td>4</td> <td>Se muestra la información de las promociones, dentro de la aplicación</td> </tr> </tbody> </table>	Paso	Acción	1	La aplicación solicita tiendas dentro de la misma zona comercial del usuario	2	El servicio web solicita establecimiento en el que se encuentra el usuario	3	El servicio web envía la información de las promociones que solicitó el usuario	4	Se muestra la información de las promociones, dentro de la aplicación
Paso	Acción										
1	La aplicación solicita tiendas dentro de la misma zona comercial del usuario										
2	El servicio web solicita establecimiento en el que se encuentra el usuario										
3	El servicio web envía la información de las promociones que solicitó el usuario										
4	Se muestra la información de las promociones, dentro de la aplicación										
Flujos alternos	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>La aplicación solicita tiendas dentro de la misma zona comercial del usuario</td> </tr> <tr> <td>2</td> <td>El servicio web solicita establecimiento en el que se encuentra el usuario</td> </tr> <tr> <td>3</td> <td>IL no existir promociones dentro del establecimiento que el usuario seleccionó, el servicio web envía un mensaje indicando que no existen promociones.</td> </tr> <tr> <td>4</td> <td>Se muestra dentro de la aplicación el mensaje recibido desde el servicio web</td> </tr> </tbody> </table>	Paso	Acción	1	La aplicación solicita tiendas dentro de la misma zona comercial del usuario	2	El servicio web solicita establecimiento en el que se encuentra el usuario	3	IL no existir promociones dentro del establecimiento que el usuario seleccionó, el servicio web envía un mensaje indicando que no existen promociones.	4	Se muestra dentro de la aplicación el mensaje recibido desde el servicio web
Paso	Acción										
1	La aplicación solicita tiendas dentro de la misma zona comercial del usuario										
2	El servicio web solicita establecimiento en el que se encuentra el usuario										
3	IL no existir promociones dentro del establecimiento que el usuario seleccionó, el servicio web envía un mensaje indicando que no existen promociones.										
4	Se muestra dentro de la aplicación el mensaje recibido desde el servicio web										
Pre-condiciones	El servicio web debe conocer la tienda seleccionada por el usuario Deben existir promociones dentro del establecimiento que el usuario seleccionó										
Post-condiciones	Se muestran las promociones dentro de la aplicación										
Requerimientos trazados	Enviar establecimiento seleccionado al servicio web Enviar promociones desde el servicio web hacia la aplicación Mostrar datos de las promociones dentro de la aplicación										
Excepciones	No existen promociones dentro del establecimiento seleccionado por el usuario										

Nota: Elaboración Propia

Tabla 18

Descripción del décimo quinto caso de uso

Caso de Uso	#15 Enviar categorías según preferencias										
Actor	Servicio web										
Descripción	El servicio web recibe una solicitud de recepción de categorías desde la aplicación. El servicio web busca las categorías de la tienda que el usuario seleccionó, busca las preferencias del usuario y envía las categorías que al usuario le interesan										
Flujo básico	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>La aplicación solicita categorías dentro del establecimiento seleccionado por el usuario</td> </tr> <tr> <td>2</td> <td>El servicio web busca las preferencias del usuario</td> </tr> <tr> <td>3</td> <td>El servicio web solo envía las categorías del establecimiento que correspondan a las preferencias del usuario</td> </tr> <tr> <td>4</td> <td>Se muestra las categorías dentro de la aplicación</td> </tr> </tbody> </table>	Paso	Acción	1	La aplicación solicita categorías dentro del establecimiento seleccionado por el usuario	2	El servicio web busca las preferencias del usuario	3	El servicio web solo envía las categorías del establecimiento que correspondan a las preferencias del usuario	4	Se muestra las categorías dentro de la aplicación
Paso	Acción										
1	La aplicación solicita categorías dentro del establecimiento seleccionado por el usuario										
2	El servicio web busca las preferencias del usuario										
3	El servicio web solo envía las categorías del establecimiento que correspondan a las preferencias del usuario										
4	Se muestra las categorías dentro de la aplicación										
Flujos alternos	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>La aplicación solicita categorías dentro del establecimiento seleccionado por el usuario</td> </tr> <tr> <td>2</td> <td>El servicio web busca las preferencias del usuario</td> </tr> <tr> <td>3</td> <td>Al no conocer las preferencias del usuario, el servicio web envía todas las categorías de la tienda</td> </tr> <tr> <td>4</td> <td>Se muestra las categorías dentro de la aplicación</td> </tr> </tbody> </table>	Paso	Acción	1	La aplicación solicita categorías dentro del establecimiento seleccionado por el usuario	2	El servicio web busca las preferencias del usuario	3	Al no conocer las preferencias del usuario, el servicio web envía todas las categorías de la tienda	4	Se muestra las categorías dentro de la aplicación
Paso	Acción										
1	La aplicación solicita categorías dentro del establecimiento seleccionado por el usuario										
2	El servicio web busca las preferencias del usuario										
3	Al no conocer las preferencias del usuario, el servicio web envía todas las categorías de la tienda										
4	Se muestra las categorías dentro de la aplicación										
Pre-condiciones	Opción de uso de preferencias seleccionada Conocer las categorías que el usuario prefiere										
Post-condiciones	Se muestran las categorías dentro de la aplicación										
Requerimientos trazados	Buscar preferencias del usuario Enviar categorías que sean de la preferencia del usuario										
Excepciones	No se conocen las preferencias del usuario.										

Nota: Elaboración Propia

Tabla 19

Descripción del décimo sexto caso de uso

Caso de Uso	#16 Enviar categorías generales										
Actor	Servicio web										
Descripción	El servicio web recibe una solicitud de recepción de categorías desde la aplicación. El servicio web busca las categorías de la tienda que el usuario seleccionó y las envía										
Flujo básico	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>La aplicación solicita categorías dentro del establecimiento seleccionado por el usuario</td> </tr> <tr> <td>2</td> <td>El servicio web busca las categorías dentro de la base de datos</td> </tr> <tr> <td>3</td> <td>El servicio web envía las categorías del establecimiento que seleccionó el usuario</td> </tr> <tr> <td>4</td> <td>Se muestra las categorías dentro de la aplicación</td> </tr> </tbody> </table>	Paso	Acción	1	La aplicación solicita categorías dentro del establecimiento seleccionado por el usuario	2	El servicio web busca las categorías dentro de la base de datos	3	El servicio web envía las categorías del establecimiento que seleccionó el usuario	4	Se muestra las categorías dentro de la aplicación
Paso	Acción										
1	La aplicación solicita categorías dentro del establecimiento seleccionado por el usuario										
2	El servicio web busca las categorías dentro de la base de datos										
3	El servicio web envía las categorías del establecimiento que seleccionó el usuario										
4	Se muestra las categorías dentro de la aplicación										
Flujos alternos	No existen flujos alternos										
Pre-condiciones	Opción de uso de preferencias no seleccionada										
Post-condiciones	Se muestran las categorías dentro de la aplicación										
Requerimientos trazados	Enviar categorías sin usar las preferencias del usuario Mostrar categorías dentro de la aplicación										
Excepciones	No existen categorías dentro del establecimiento seleccionado										

Nota: Elaboración Propia

Tabla 20

Descripción del décimo séptimo caso de uso

Caso de Uso	#17 Autenticar usuario										
Actor	Servicio web										
Descripción	El servicio web recibe una solicitud de autenticación del usuario, busca los datos dentro de la base de datos y si estos coinciden permite el acceso al usuario.										
Flujo básico	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>La aplicación envía datos de autenticación al servicio web</td> </tr> <tr> <td>2</td> <td>El usuario busca datos del usuario en la base de datos</td> </tr> <tr> <td>3</td> <td>El servicio web determina que el usuario fue autenticado correctamente</td> </tr> <tr> <td>4</td> <td>Se permite que el usuario acceda al sistema</td> </tr> </tbody> </table>	Paso	Acción	1	La aplicación envía datos de autenticación al servicio web	2	El usuario busca datos del usuario en la base de datos	3	El servicio web determina que el usuario fue autenticado correctamente	4	Se permite que el usuario acceda al sistema
Paso	Acción										
1	La aplicación envía datos de autenticación al servicio web										
2	El usuario busca datos del usuario en la base de datos										
3	El servicio web determina que el usuario fue autenticado correctamente										
4	Se permite que el usuario acceda al sistema										
Flujos alternos	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>La aplicación envía datos de autenticación al servicio web</td> </tr> <tr> <td>2</td> <td>El usuario busca datos del usuario en la base de datos</td> </tr> <tr> <td>3</td> <td>El servicio web determina que el usuario no puede ser autenticado.</td> </tr> <tr> <td>4</td> <td>No se permite que el usuario acceda al sistema</td> </tr> </tbody> </table>	Paso	Acción	1	La aplicación envía datos de autenticación al servicio web	2	El usuario busca datos del usuario en la base de datos	3	El servicio web determina que el usuario no puede ser autenticado.	4	No se permite que el usuario acceda al sistema
Paso	Acción										
1	La aplicación envía datos de autenticación al servicio web										
2	El usuario busca datos del usuario en la base de datos										
3	El servicio web determina que el usuario no puede ser autenticado.										
4	No se permite que el usuario acceda al sistema										
Pre-condiciones	Recibe datos de autenticación del usuario El usuario está registrado en el sistema										
Post-condiciones	El usuario tiene acceso al sistema										
Requerimientos trazados	Autenticación de usuario Búsqueda de usuario en la base de datos Sólo permitir acceso a un usuario autenticado										
Excepciones	El usuario no fue registrado dentro del sistema										

Nota: Elaboración Propia

2.3 Especificación de diseño

Para representar la arquitectura de software usada en la construcción del sistema de publicidad se utilizó el modelo 4+1. Según Kruchten (1995), el modelo 4+1, proporciona 5 vistas, la vista lógica, física, de desarrollo, de procesos y de escenarios. Este modelo permite mostrar múltiples enfoques del sistema para poder abordar el conjunto de intereses de los distintos *stakeholders* y así conseguir representar el sistema de una forma más precisa.

Cada diagrama que se realizó pertenece a una de las vistas anteriormente mencionadas. En la vista lógica se realizaron los diagramas de clases y de secuencia, en la vista de desarrollo, se hizo el diagrama de paquetes, en la vista de procesos, el diagrama de actividad, en la vista física, el diagrama de despliegue y finalmente en la vista de escenarios se realizó el diagrama de casos de uso.

En la vista de escenarios se realiza el diagrama de casos de uso, en él se representan las abstracciones de los procesos del sistema y la forma en la que interactúan con los distintos *stakeholders*. El diagrama y las especificaciones de los casos de uso se encuentran en la sección “Especificación de requisitos del Proyecto de TIC”.

En la vista lógica se busca representar los requisitos funcionales del sistema, es decir que muestra los intereses de los usuarios finales. Primero se muestra el diagrama de clases de la aplicación, como se puede observar, se tiene dos clases

principales en esta aplicación, la clase UIViewController, que es el controlador de vistas que provee IOS para poder controlar las interfaces de usuario; y la clase ServerConnection, se encarga de brindar las funcionalidades de conexión entre la aplicación y el servicio web que se encarga de la parte lógica del sistema.

Se puede observar que la clase ServerConnection presenta herencia en diferentes clases, dentro de ellas se encuentra loginConnection encargada de realizar la validación de datos con el servidor, gpsConnection encargado de enviar los datos de la localización geográfica del usuario a servicio web, storeConnection encargado de recibir las tiendas que se encuentran en la misma zona comercial que el usuario, categoryConnection responsable de recibir las categorías que tiene un determinado establecimiento, couponsConnection encargado de recibir los cupones que tiene un determinado establecimiento, saveCouponConnection encargado de guardar los cupones que el usuario desea dentro de su cuponera virtual y enviarlo al servicio web, deleteCouponConnection el cual sirve para eliminar un determinado cupón de la cuponera del usuario, savePreferenceConnection responsable de guardar las categorías que son de la preferencia del usuario y guardarlas en la base de datos del servicio web, y deletePreferenceConnection el cual se encarga de eliminar las categorías que no son de preferencia del usuario y eliminarlas de la base de datos del servicio web.

Todas las clases de conexión con el servidor nombradas previamente implementan la clase principal serverConnection, la cual cuenta con todos los

requerimientos para realizar esta conexión, sin embargo estas clases desarrollan funciones específicas para funciones determinadas, por tal motivo cuentan con datos específicos para realizar estas funciones de la manera correcta según su protocolo de conexión con el servicio web.

En cuanto a la clase UIViewController, la cual es brindada por biblioteca de IOS, hereda sus atributos y métodos a las clases ViewController, MainMenu, GPSView, StoresView, CategoryView, CouponsView, CouponBookView, CouponBookedShow, CouponUseView y PreferencesManagerView; todas estas clases son controladores de las vistas que posee la aplicación y cuentan con las funciones específicas para realizar cada una de sus tareas, todas estas vistas usan los objetos ServerConnection según la tareas que desean realizar y a veces usan más de un tipo de conexión con el servicio web.

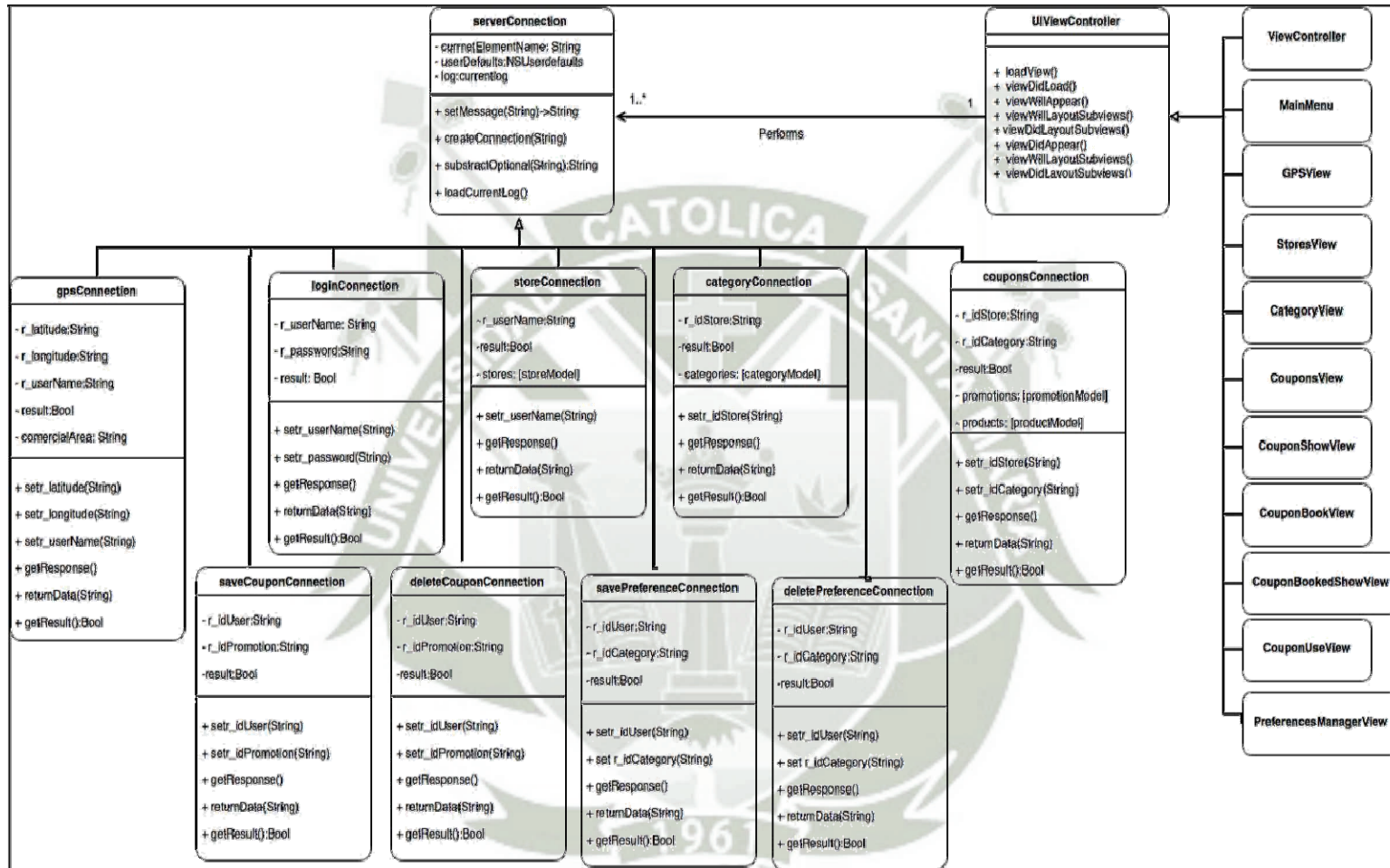


Figura 20 Diagrama de clases de la aplicación

Fuente: Elaboración propia

Luego se muestra el diagrama de clases del web service, este contiene únicamente las clases representativas para el sistema, para poder mostrar un diagrama más sencillo de comprender. Las clases que se han omitido son las que contienen los modelos de cada tabla de la base de datos, estas son necesarias principalmente para la manipulación de datos desde la herramienta ORM “Hibernate”; también se omitieron las clases que se utilizan para la comunicación de datos entre el web service y la aplicación móvil, es decir las clases de solicitud y respuesta de cada operación descrita en el archivo WSDL, estas clases son generadas automáticamente gracias al plugin “cxf-codegen-plugin”; por último se omitieron las clases Adaptadores que se utilizan para mapear los tipos de datos fecha y hora de java con los tipos de datos creado automáticamente por el plugin anteriormente mencionado.

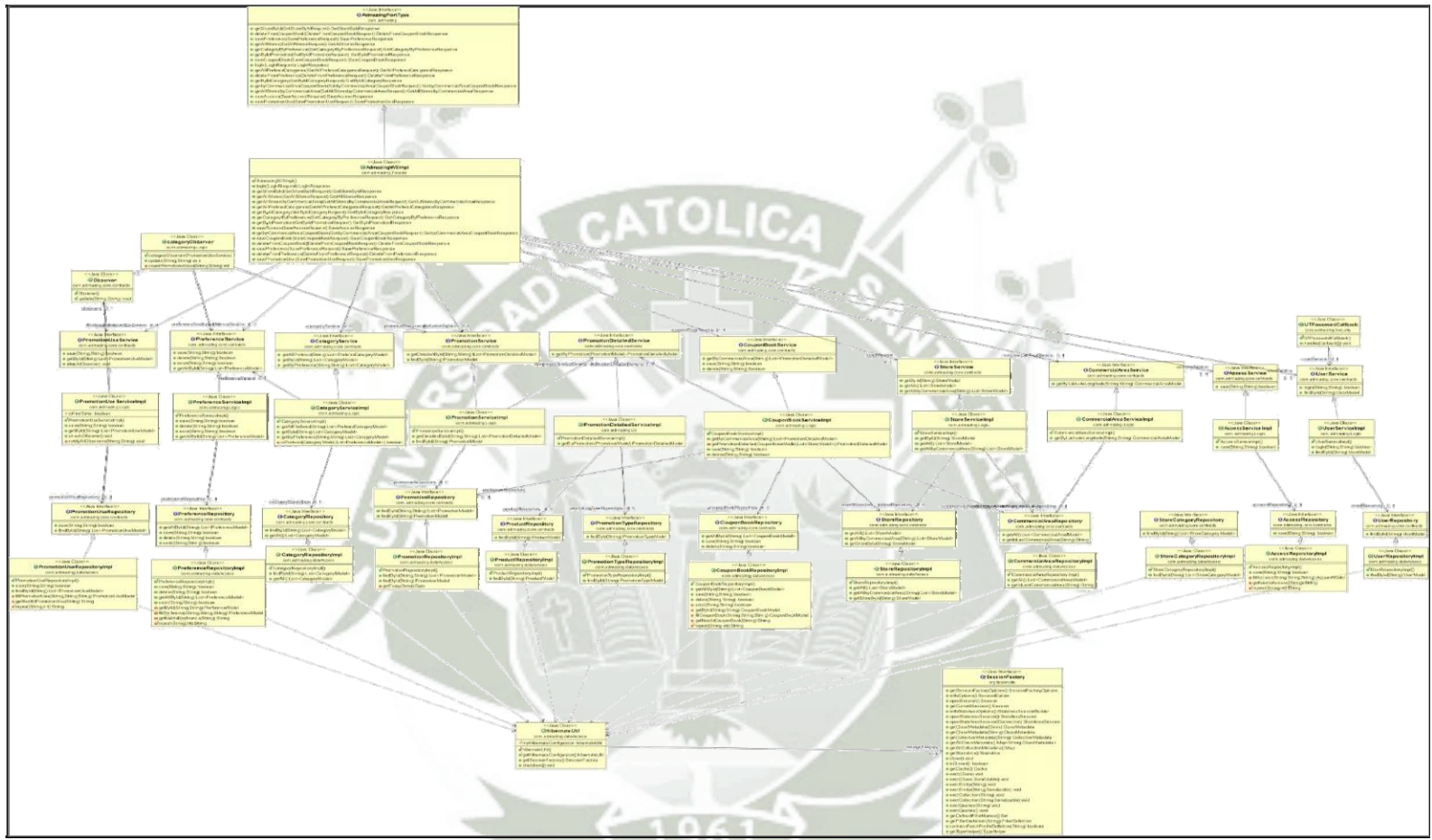


Figura 21: Diagrama de clases del web service
Fuente: Elaboración propia

En la siguiente imagen se muestra un extracto del diagrama de clases, en él se muestra que la interfaz “AdmazingPortType” contiene todas las operaciones que la aplicación puede utilizar. “AdmazingPortType” es la interfaz del patrón fachada y la clase “AmazingWSImpl” es la clase de implementación de dicha interfaz, se utilizó el patrón estructural fachada con la finalidad de reducir la complejidad y la dependencia del sistema, logrando que la aplicación móvil únicamente acceda al servicio Web a través de esta clase y así la aplicación móvil depende únicamente de esta clase.

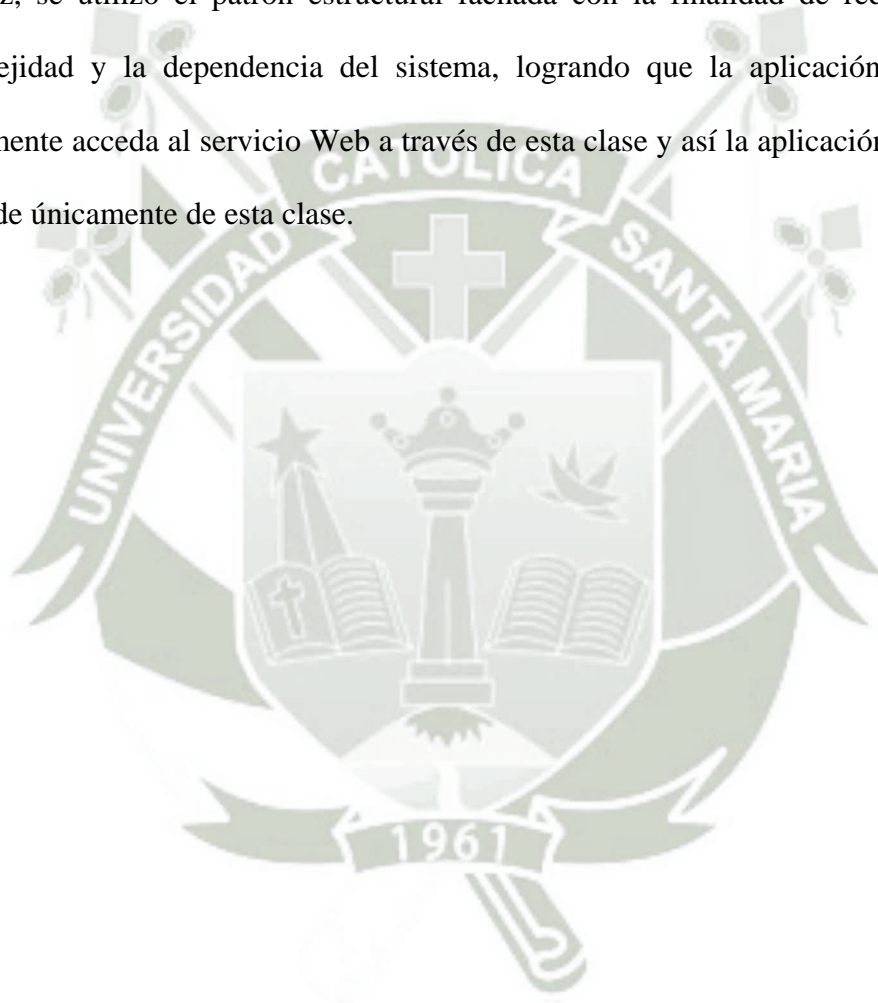




Figura 22: El patrón Fachada dentro del diagrama de clases del web service.

Fuente: Elaboración propia

Cada operación detallada en la clase utilizada como fachada para el servicio web llama a las distintas clases de servicio según su necesidad. Estas clases son el intermediario entre las operaciones que se encuentran en la clase

“AdmazingPortType” y las clases que acceden a la base de datos. Las clases de servicio se encargan de realizar el desarrollo de los algoritmos para responder a las solicitudes descritas en la clase fachada, por ello se crearon únicamente las clases de servicio necesarias para responder dichas solicitudes. En la siguiente imagen se muestran todas las interfaces de los servicios requeridos y sus implementaciones correspondientes.



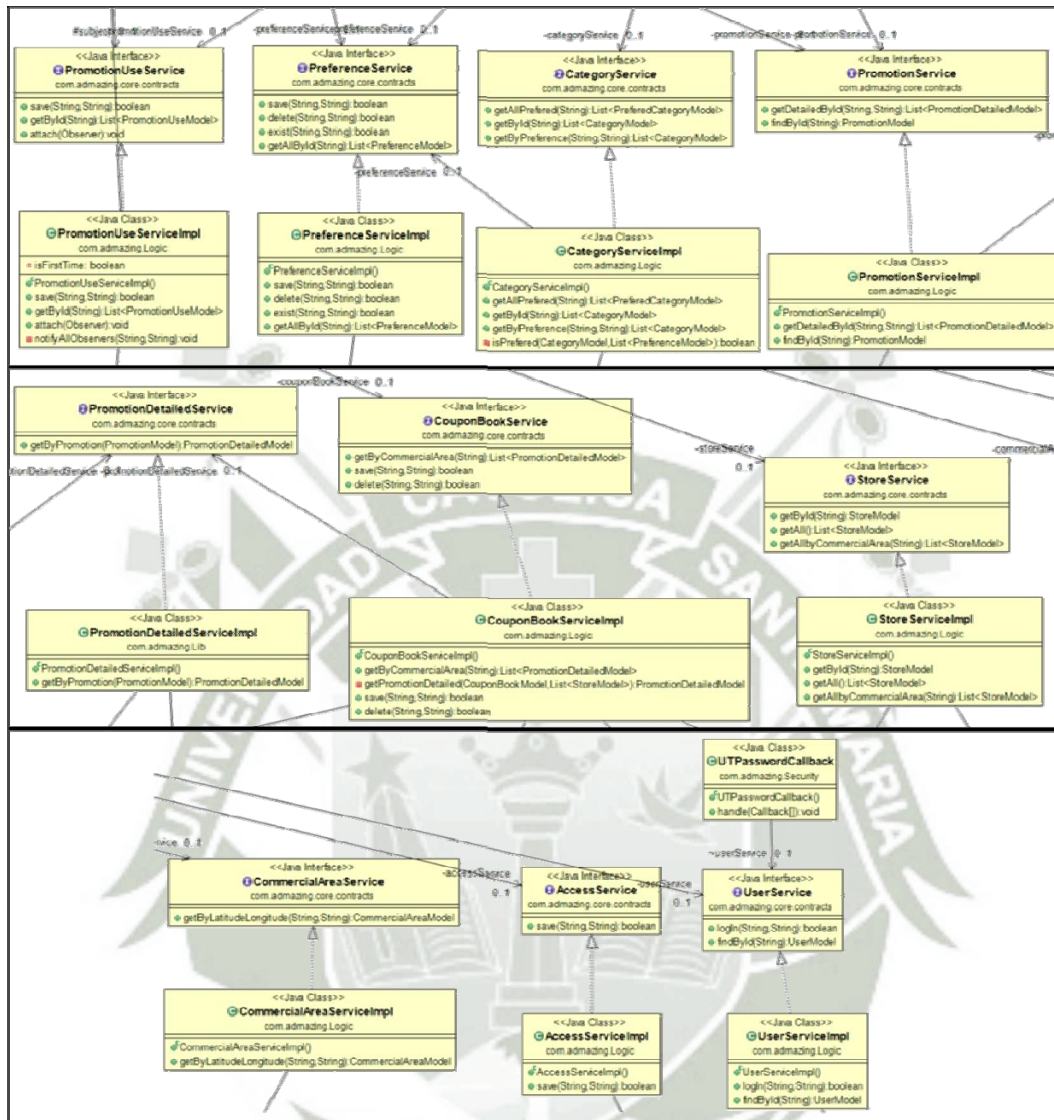


Figura 23: Clases de servicios dentro del diagrama de clases del web service.

Fuente: Elaboración propia

A continuación se muestra otro extracto del diagrama de clases, en él se muestra la utilización del patrón Observador. La clase “Observer” es la interfaz que contiene como observado a la clase “PromotionUseService”, porque se necesita que

cada vez que se guarde un uso de una promoción, se ejecute el algoritmo que se encuentra implementado en “CategoryObserver”. El algoritmo consiste en buscar la categoría a la que pertenece la promoción y calcular las veces que promociones de esta categoría fueron utilizadas, si las veces exceden de 5 se agrega esta categoría a preferencias. A causa de este algoritmo, la clase “CategoryObserver” tiene dependencias con “PreferenceService” y “CategoryService” tal como se puede observar en la siguiente imagen.

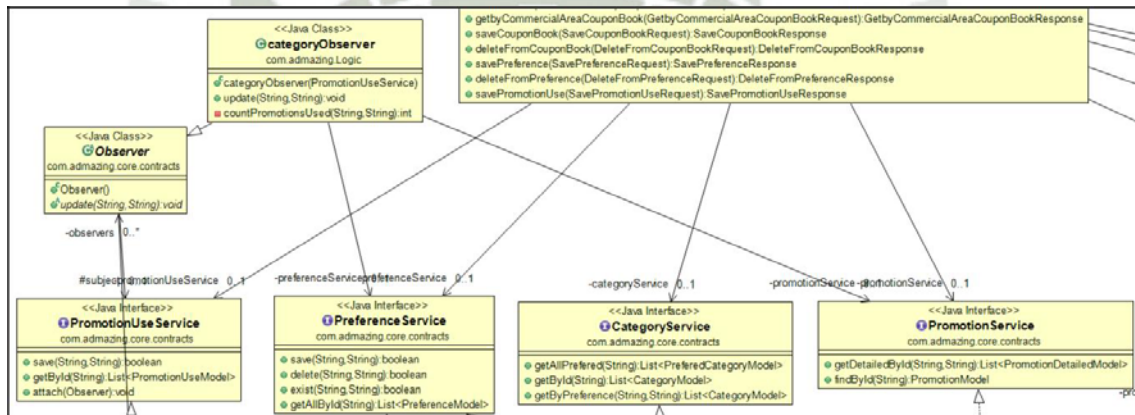


Figura 24: El patrón Observador dentro del diagrama de Clases del web service.

Fuente: Elaboración propia

Las clases repositorio son las encargadas de obtener, agregar, actualizar o eliminar datos de la base de datos a través del ORM “Hibernate”, estos repositorios utilizan los modelos de cada tabla para lograr exitosamente el mapeo entre una clase java y una tabla en la base de datos. A diferencia de las clases de servicio, las clases repositorio están estrechamente ligadas con las tablas de la base de datos, por ello, existe una clase repositorio por cada tabla existente en la base de datos. En la

siguiente imagen se muestran las interfaces e implementaciones de las clases repositorio.

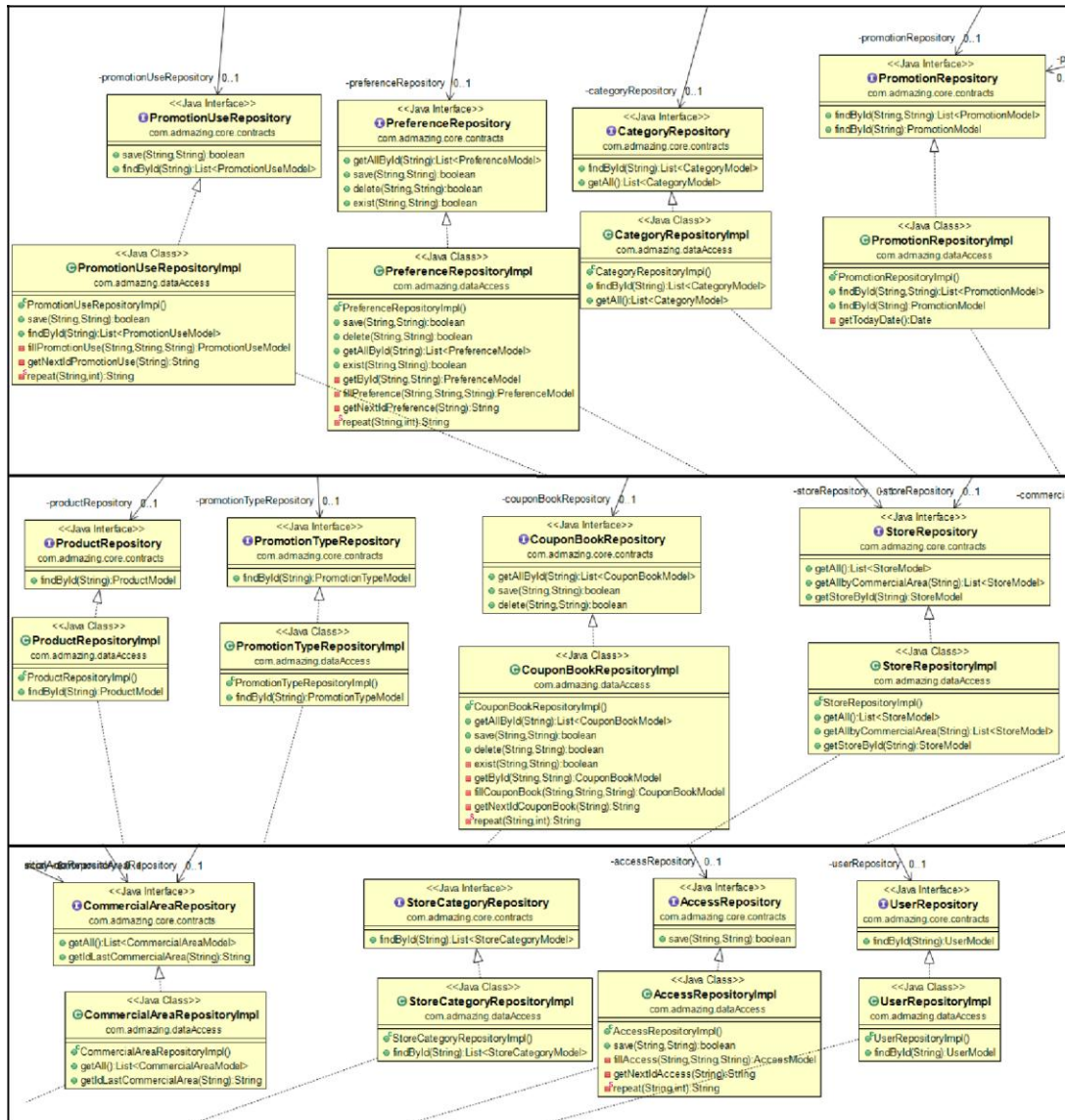


Figura 25 : Clases repositorio dentro del diagrama de clases del web service
Fuente: Elaboración propia

Finalmente, en el último extracto del diagrama de clases, se encuentra la clase “hibernateUtil” que instancia a la clase “SessionFactory”, esta es la encargada de crear conexiones a la base de datos. Se utilizó el patrón “Singleton” en la clase “hibernateUtil”, para crear una única instancia de este, y por consiguiente una única instancia de “SessionFactory”, con el objetivo de que cada una de las clases repositorio, anteriormente mostradas, obtengan conexiones de un único Creador de Conexiones (“SessionFactory”), así se logra que el servicio utilice menos memoria y que sea más rápido respondiendo a las solicitudes de la aplicación móvil.

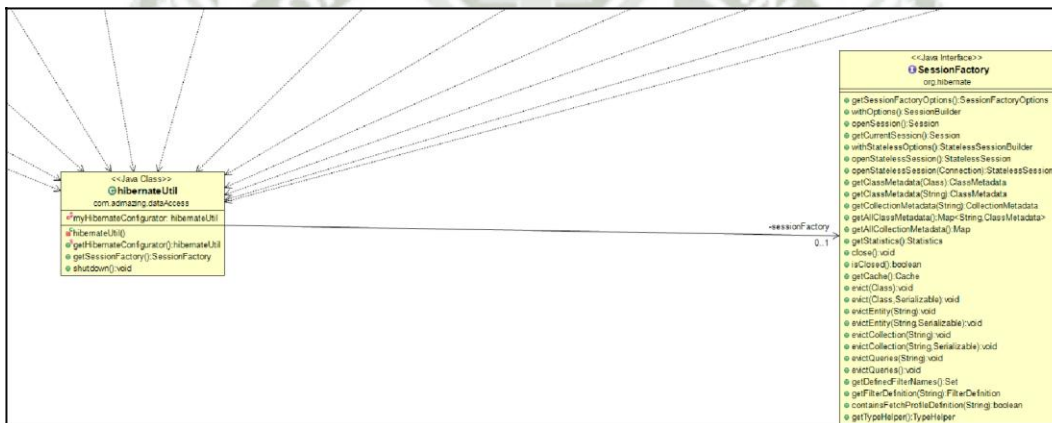


Figura 26 : El patrón Singleton dentro del diagrama de clases del web service
Fuente: Elaboración propia

Posteriormente en la vista, se muestran los diagramas de secuencia, en ellos se observan las interacciones de cuatro situaciones distintas que se efectúan al utilizar el sistema de promociones. Se mostrará desde la petición del usuario hasta

la base de datos, es decir que toma en cuenta tanto la parte de la aplicación móvil como la del servicio web.

La primera situación ilustra la utilización de un cupón, se inicia con la acción del usuario al pulsar el botón de uso de cupón, este es manejado a través del controlador de vista “couponShow”, luego a la conexión de servidor “couponShow” y realiza la solicitud “savePromotion” a la fachada del servidor web “AdmazingPortType”. La fachada primero guarda el uso de la promoción a través de la clase de servicio “PromotionUseService” y esta a su vez llama a “PromotionUseRepository”, la cual se conecta con la base de datos para efectuar el guardado. Luego, si la fachada recibe que el resultado de la acción es verdadero entonces elimina la promoción de la cuponera, llamando a la clase “CouponBookService”, la cual llama a la clase “CouponBookRepository” que se conecta con la base de datos para la eliminación. Si el resultado de la eliminación es correcto, la fachada devuelve verdadero a la aplicación móvil, y en ella se muestra el mensaje de confirmación. Si el resultado es negativo tanto en el guardado como en la eliminación, la fachada devuelve falso a la aplicación y se muestra el mensaje de error.

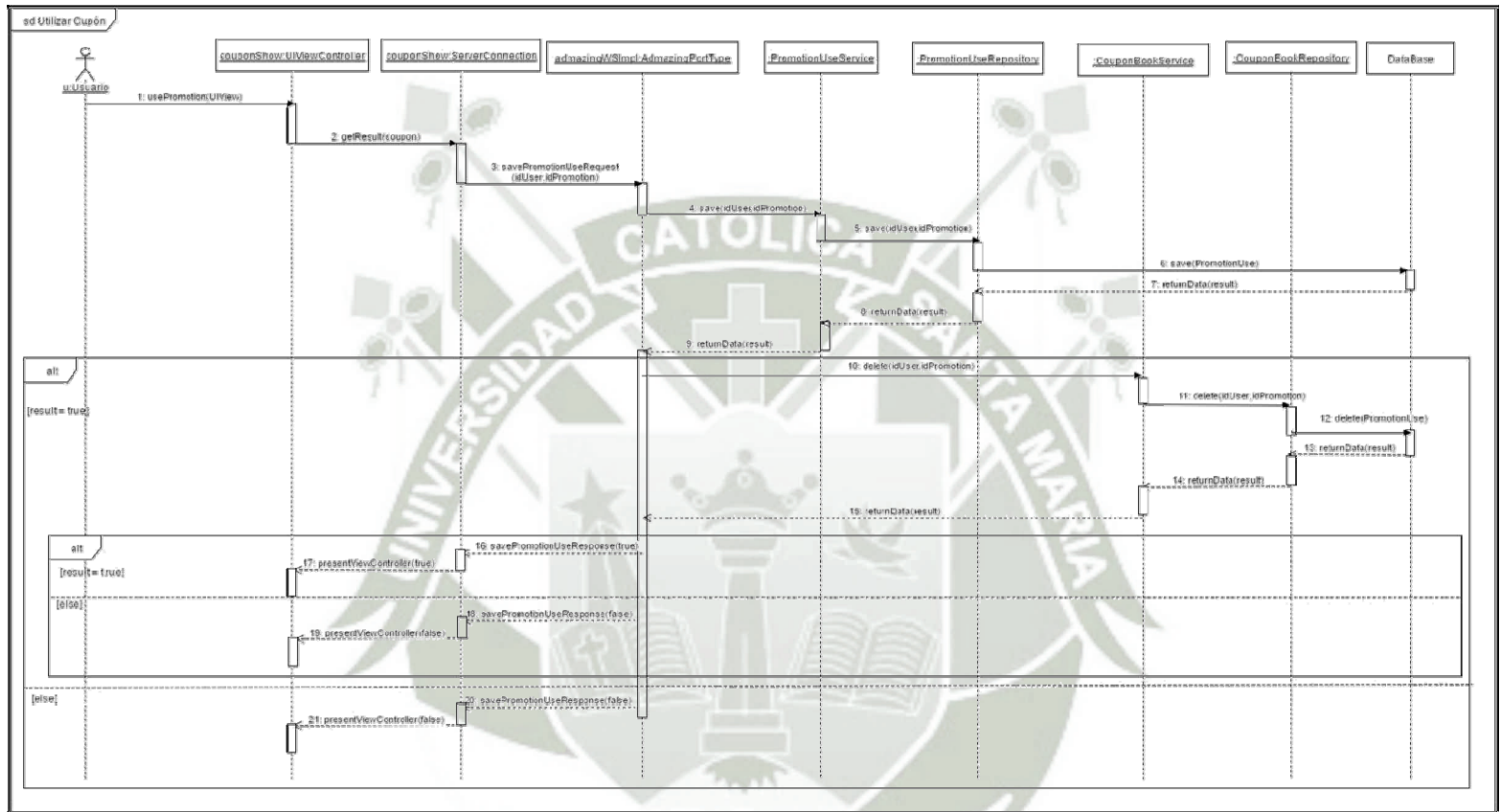


Figura 27: Diagrama de secuencia de la utilización de un cupón

Fuente: Elaboración propia

La segunda situación detalla la actualización categorías preferidas, se inicia con la acción del usuario al pulsar el botón de actualizar categorías, este es manejado a través del controlador de vista “preferencesManager”, este obtiene todas las categorías. Se inicia el bucle que recorre cada categoría. En caso que la categoría esté marcada, se llama a la conexión de servidor “preferencesManager” y se realiza la solicitud “savePreference” a la fachada del servidor web “AdmazingPortType”. La fachada guarda la preferencia a través de la clase de servicio “PreferenceService” y esta a su vez llama a “PreferenceRepository”, la cual se conecta con la base de datos para efectuar el guardado. Si el resultado que recibe la fachada es verdadero, la fachada devuelve verdadero a la aplicación móvil, y obtiene el mensaje de confirmación. Si el resultado es negativo, la fachada devuelve falso a la aplicación y se obtiene el mensaje de error. En caso que la categoría no se encuentre marcada, se llama a la conexión de servidor “preferencesManager” y se realiza la solicitud “deleteFromPreference” a la fachada del servidor web “AdmazingPortType”. La fachada elimina la preferencia a través de la clase de servicio “PreferenceService” y esta a su vez llama a “PreferenceRepository”, la cual se conecta con la base de datos para efectuar la eliminación. Si el resultado que recibe la fachada es verdadero, la fachada devuelve verdadero a la aplicación móvil, y obtiene el mensaje de confirmación. Si el resultado es negativo, la fachada devuelve falso a la aplicación y se muestra el mensaje de error.

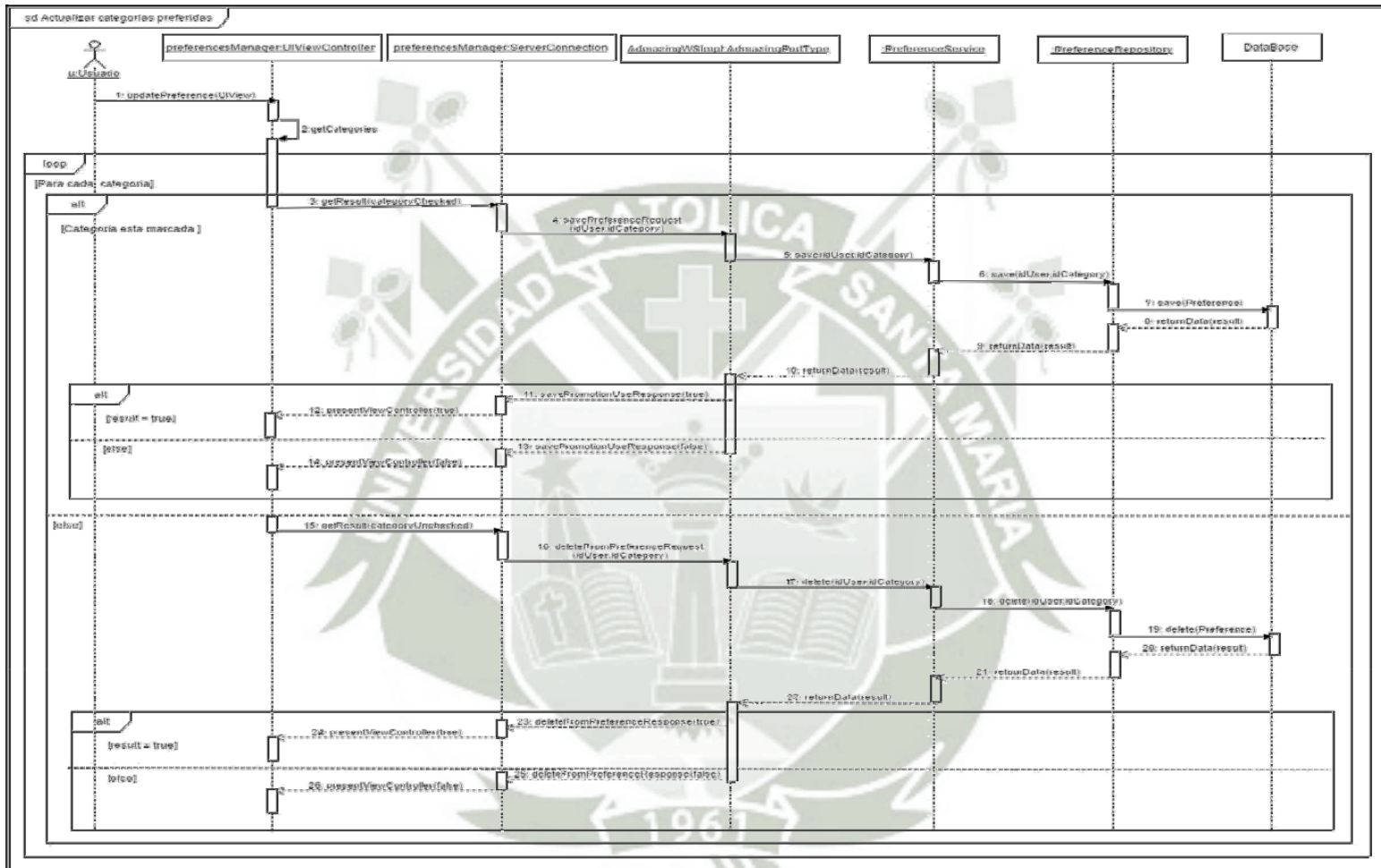


Figura 28: Diagrama de secuencia de la actualización categorías preferidas
Fuente: Elaboración propia

La tercera situación detalla la obtención de promociones, se inicia con la acción del usuario al elegir la categoría dentro de una tienda específica, este es manejado a través del controlador de vista “coupons”, luego llama a la conexión de servidor “coupons” y se realiza la solicitud “getByIdPromotion” a la fachada del servidor web “AdmazingPortType”. La fachada obtiene las promociones a través de la clase de servicio “PromotionService” y esta a su vez llama a “PromotionRepository”, la cual se conecta con la base de datos para efectuar la búsqueda de promociones según la tienda y la categoría seleccionadas, además verifica si las promociones aún son válidas. Si existen promociones, dados los criterios expuestos anteriormente, se inicia el bucle que recorre cada promoción obtenida, se debe obtener el detalle de cada una de las promociones, para lograrlo se llama a la clase de servicio “PromotionDetailedService”, cuya función es obtener los datos del producto mediante la clase “ProductRepository” y el tipo de promoción mediante “PromotionTypeRepository”, ambas clases se conectan con la base de datos para obtener los datos necesarios. Si los datos del producto y el tipo de promoción no están vacíos, se devuelve la promoción detallada en caso contrario se devuelve nada. La fachada devuelve el resultado obtenido de la clase “PromotionDetailedService” a la aplicación móvil, y este muestra las promociones o en su defecto el mensaje de error.

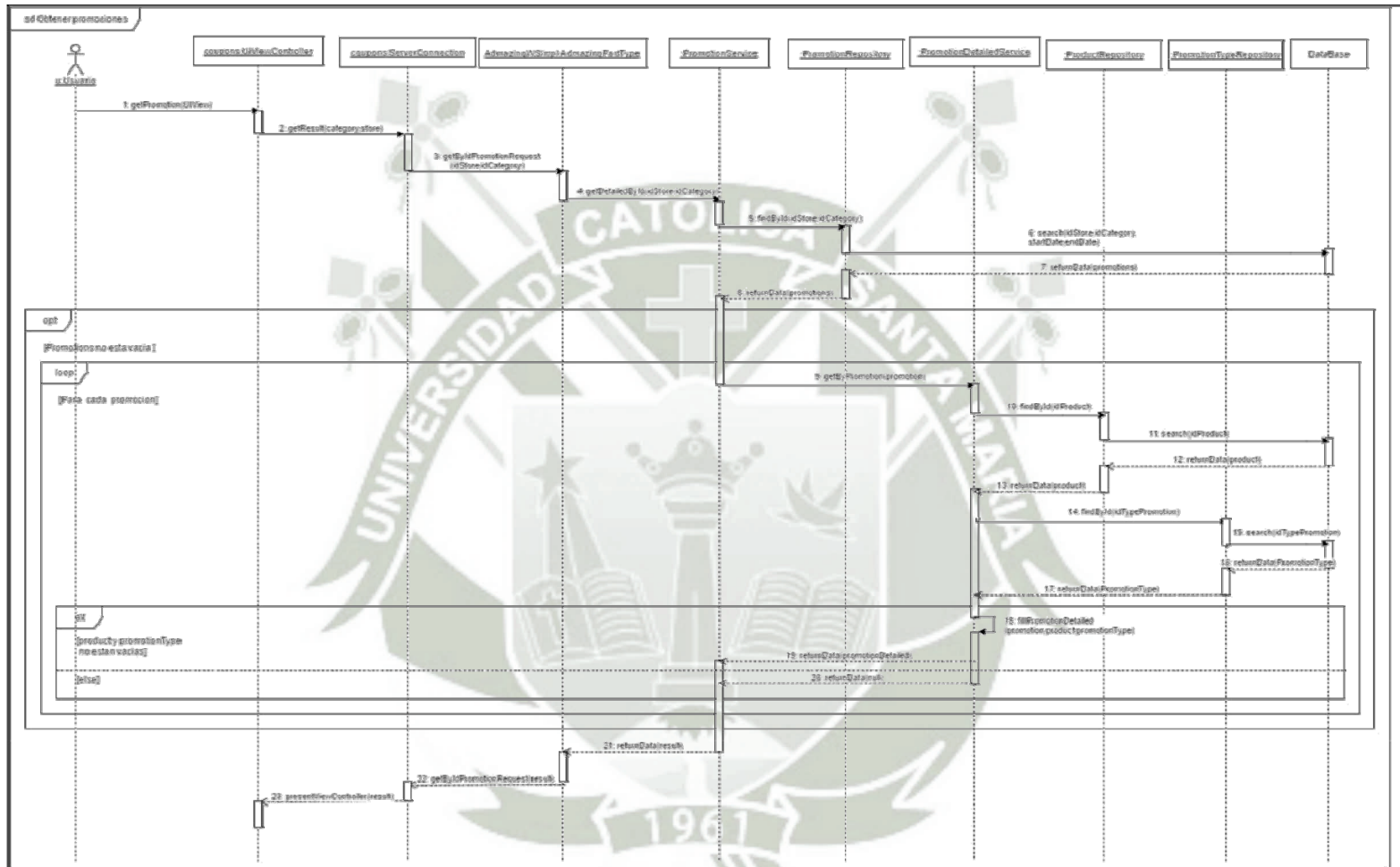


Figura 29: Diagrama de secuencia de la obtención de promociones
Fuente: Elaboración propia

La última situación detalla la actualización de ubicación, se inicia con la acción del usuario al ingresar al sistema o al pulsar el botón de actualizar ubicación, este es manejado a través del controlador de vista “location”, el cual obtiene la ubicación actual del usuario a través de su GPS. Luego se llama a la conexión de servidor “location” y se realiza la solicitud “saveAccessRequest” a la fachada del servidor web “AdmazingPortType”. La fachada obtiene todas las zonas comerciales a través de la clase de servicio “CommercialAreaService” y esta a su vez llama a “CommercialAreaRepository”, la cual se conecta con la base de datos para la obtención. Se inicia el bucle que recorre cada zona comercial, primero se calcula la distancia entre el usuario y el eje de la zona comercial, en caso la distancia hallada sea menor al radio de dicha zona comercial, se devuelve a la fachada los datos de ella. La fachada se encarga de guardar el acceso del usuario a la zona comercial llamando a la clase de servicio “AccessService”, esta a su vez llama a la clase repositorio “AccessRepository”, que se conecta a la base de datos para realizar el guardado. Si la información de la zona comercial que recibe la fachada no está vacía, se envía que el resultado fue correcto al guardar el acceso y se envía los datos de la zona comercial a la aplicación móvil, la cual muestra el mensaje correspondiente. Si la información de la zona comercial está vacía, la fachada devuelve falso a la aplicación y se obtiene el mensaje de error.

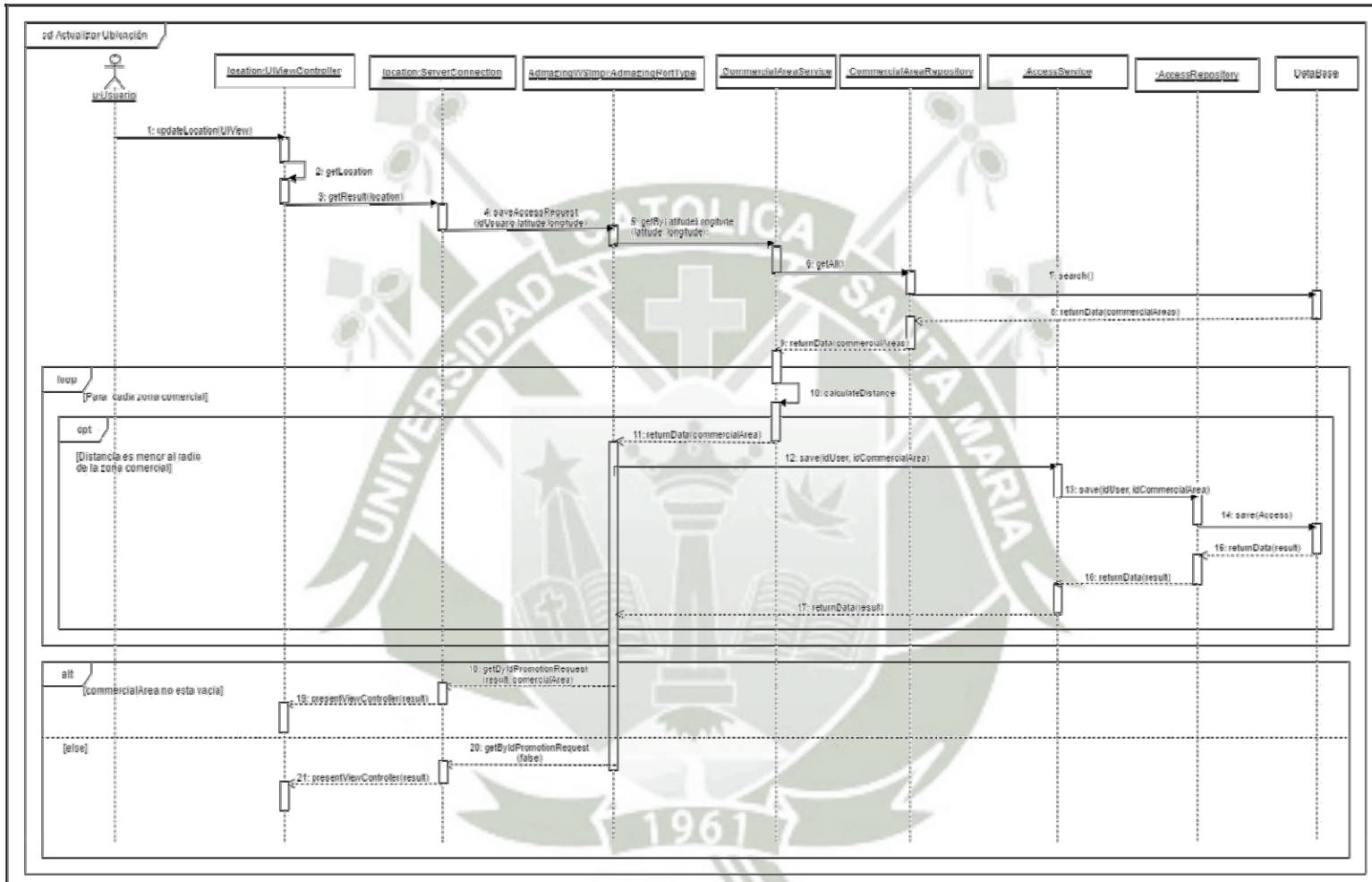


Figura 30: Diagrama de secuencia de la actualización de ubicación.

Fuente: Elaboración propia

En la vista de desarrollo se representan los módulos de software, esta vista busca representar los intereses de los jefes de proyecto y de los programadores. Para esta vista se hizo el diagrama de paquetes, este diagrama proporciona la organización de capas, módulos o paquetes y submódulos del sistema con sus respectivas dependencias. La siguiente imagen muestra que el diagrama de paquetes del sistema, se divide en tres capas, siguiendo la arquitectura propuesta, la parte de presentación, la de lógica de negocio y la capa de acceso a base de datos.

En la capa de presentación hay cinco módulos o paquetes, el paquete de Interfaz de Usuario del teléfono, con sistema operativo “IOS”, el paquete de Interfaz de Usuario del Apple Watch., tiene los dos paquetes de controladores de vistas que realizan las funciones necesarias para manejar las acciones realizadas por el usuario final y también posee el paquete de conexiones con el webservice, que se ocupa de realizar las solicitudes y recibir las respuestas del servicio web.

En la capa de lógica de negocio existen cinco paquetes:

- El paquete de seguridad que contiene las clases para brindar la seguridad del sistema.
- El paquete de facade o fachada que contiene la interfaz e implementación del patrón fachada.

- El paquete de lógica que contiene todas las clases de servicio, estas se encargan de las funcionalidades básicas del sistema de promociones, aquellas que son indispensables para su correcto funcionamiento.
- El paquete de librería que se encarga de tener las funciones secundarias que no son imprescindibles para el sistema, pero son útiles para las clases que se encuentran en el paquete de lógica.
- El paquete core o núcleo, es el paquete más importante del sistema y el más independiente. En él se tienen dos submódulos, el de modelos que contiene todas las clases que sirven para el manejo de datos que se obtienen desde la base de datos a través de hibernate y el submódulo de contratos que tiene todas las interfaces para las clases que se encuentran en los paquetes acceso a datos, librería y lógica.

En la capa de acceso de datos se tienen dos paquetes: el paquete de acceso a datos contiene todas las clases repositorio, descritas anteriormente en el diagrama de clases del servicio web, y el paquete de hibernate, este contiene los archivos “hbm.xml”, son los archivos de configuración para el mapeo entre los modelos que se encuentran en el paquete core y las tablas en la base de datos.

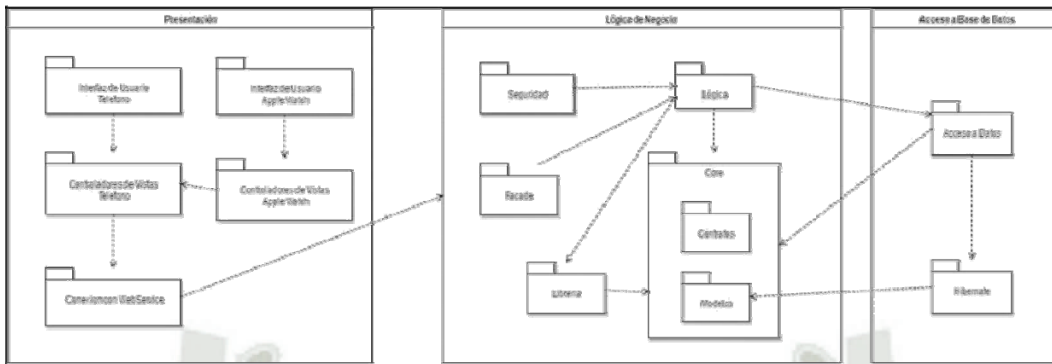


Figura 31 : Diagrama de paquetes del sistema de promociones
Fuente: Elaboración propia

En la vista de procesos se representa la distribución y la integración del sistema, es decir que muestra el flujo de trabajo del sistema de promociones. Para esta vista se desarrollaron diagramas de actividades para diferentes situaciones que ocurren durante el uso de nuestro software, cada una de estas situaciones es independiente y usa diferentes componentes del sistema.

La primera situación en el este diagrama de actividad trata de ilustrar la utilización de promociones por parte del usuario dentro de la aplicación, se puede observar que primero el usuario debe seleccionar la promoción con la que este desea interactuar, después de indicar la promoción seleccionada la aplicación captura el código de la promoción a usar y envía sus datos al servicio web que se encarga de buscar la promoción en la base de datos, evalúa la posibilidad de usar la promoción deseada. En caso que esta sea posible, la realiza, guarda el uso de promoción en la base de datos, elimina la promoción de la cuponera y envía una confirmación a la aplicación, de otra forma envía un mensaje indicando que ésta no puede ser

realizada; en ambos casos la respuesta que brinda el servicio web es mostrada al usuario mediante la aplicación, junto a otros detalles que sean importante mostrar al usuario.

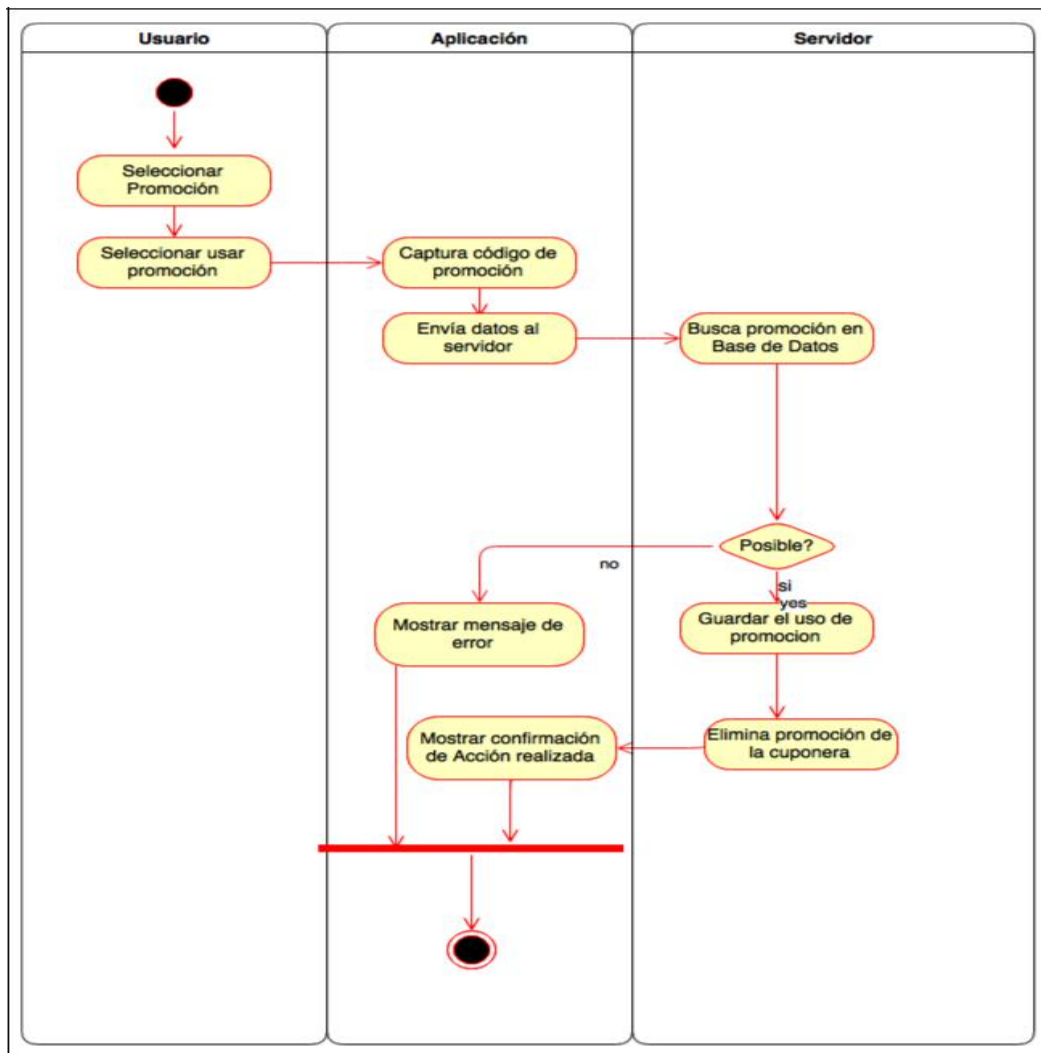


Figura 32: Diagrama de actividades de la utilización de un cupón

Fuente: Elaboración propia

En el segundo diagrama de actividad se ilustra la siguiente situación. Se quiere actualizar las categorías preferidas del usuario, para esto el usuario debe

presionar el botón de “actualizar preferencias” dentro de la aplicación, en consecuencia a esto la aplicación obtiene todas las categorías se identifica si estas son marcadas o no. En caso que las categorías estén marcadas se envían al servidor para ser guardadas, se verifica si esta categoría existe dentro de las preferencias del usuario, si no existe se guarda y envía resultado afirmativo a la aplicación, si existe, solo se envía el resultado negativo. Por el contrario, si las categorías no están marcadas, se envían al servidor para ser eliminadas, de igual manera se verifica si existe la categoría entre las preferencias, pero el resultado es inverso a cuando se guardan, porque si se confirma la existencia se elimina y el resultado que devuelve es afirmativo, si no se confirma que el resultado es negativo. Para ambos casos, tanto si las categorías están marcadas como si no, se muestran los mensajes ya sean de confirmación o de error según la respuesta del servidor.

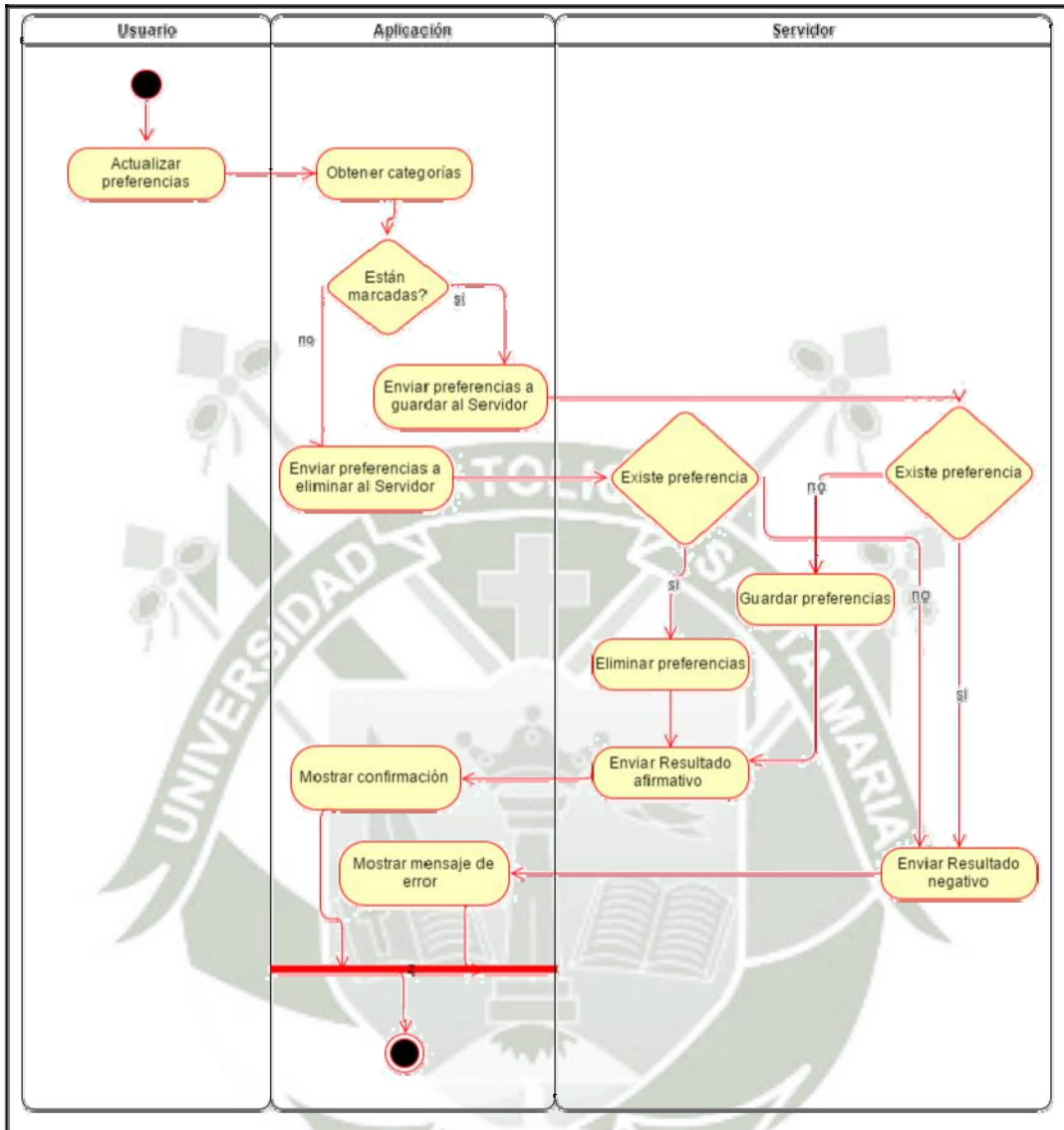


Figura 33: Diagrama de actividades de la actualización categorías preferidas

Fuente: Elaboración propia

En el tercer diagrama actividad se muestra cómo el usuario obtiene las promociones que se encuentran en su misma zona comercial, para esto debe ingresar a la opción de “ver promociones”, a continuación la aplicación envía el código de la categoría y de la tienda al servicio web, el cual se encarga de buscar

las promociones dentro de la base datos, si se determina que hay promociones, estas se envían a la aplicación, en caso contrario se envía un mensaje de falta de promociones de la tienda y categoría seleccionadas. En ambos casos se muestra la respuesta de la acción recibida del servicio web dentro de la aplicación y las promociones, si estas existen.

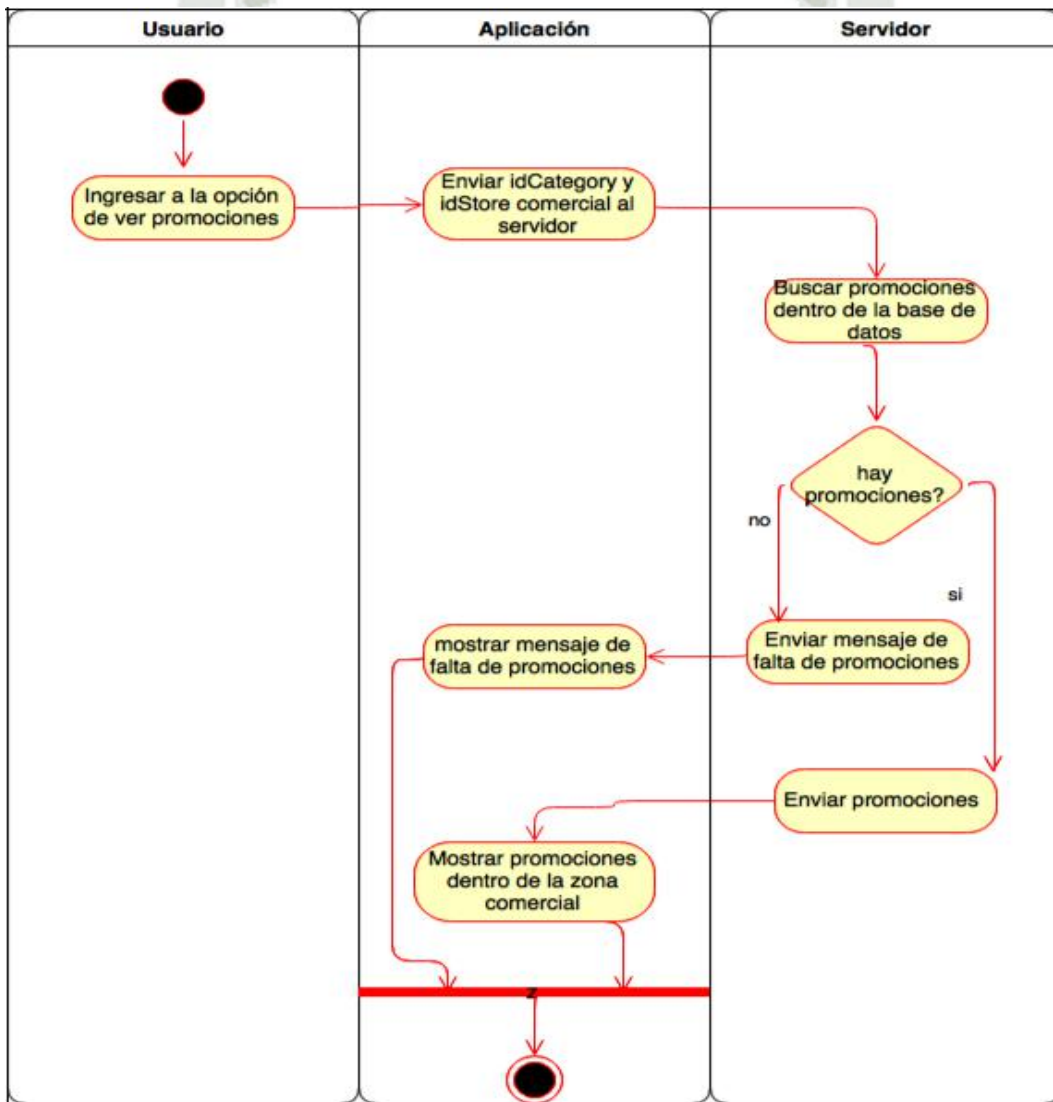


Figura 34: Diagrama de actividades de la obtención promoción

Fuente: Elaboración propia

El último diagrama de actividad ilustra cómo actualizar la posición geográfica del usuario, para esto el usuario debe presionar el botón de “actualizar ubicación” dentro de la aplicación, en consecuencia a esto la localización (latitud y longitud) del usuario es obtenida mediante la aplicación y enviada al servicio web junto con otros datos importantes como su código de usuario, luego el servicio web obtiene la información de todas las zonas comerciales, posteriormente determina la distancia del usuario con respecto a cada una de las zonas comerciales, y si la distancia es menor al radio establecido de dicha zona comercial, actualiza la ubicación del usuario dentro de la base de datos, devuelve que el resultado es afirmativo y los datos de la zona comercial donde el usuario se encuentra. Por último la confirmación del servicio web y los datos de esta zona comercial son mostrados en la aplicación mediante una alerta. En caso que el usuario no se encuentre en ninguna zona comercial, el servicio web devolverá negativo a la aplicación móvil y esta mostrará un mensaje de error.

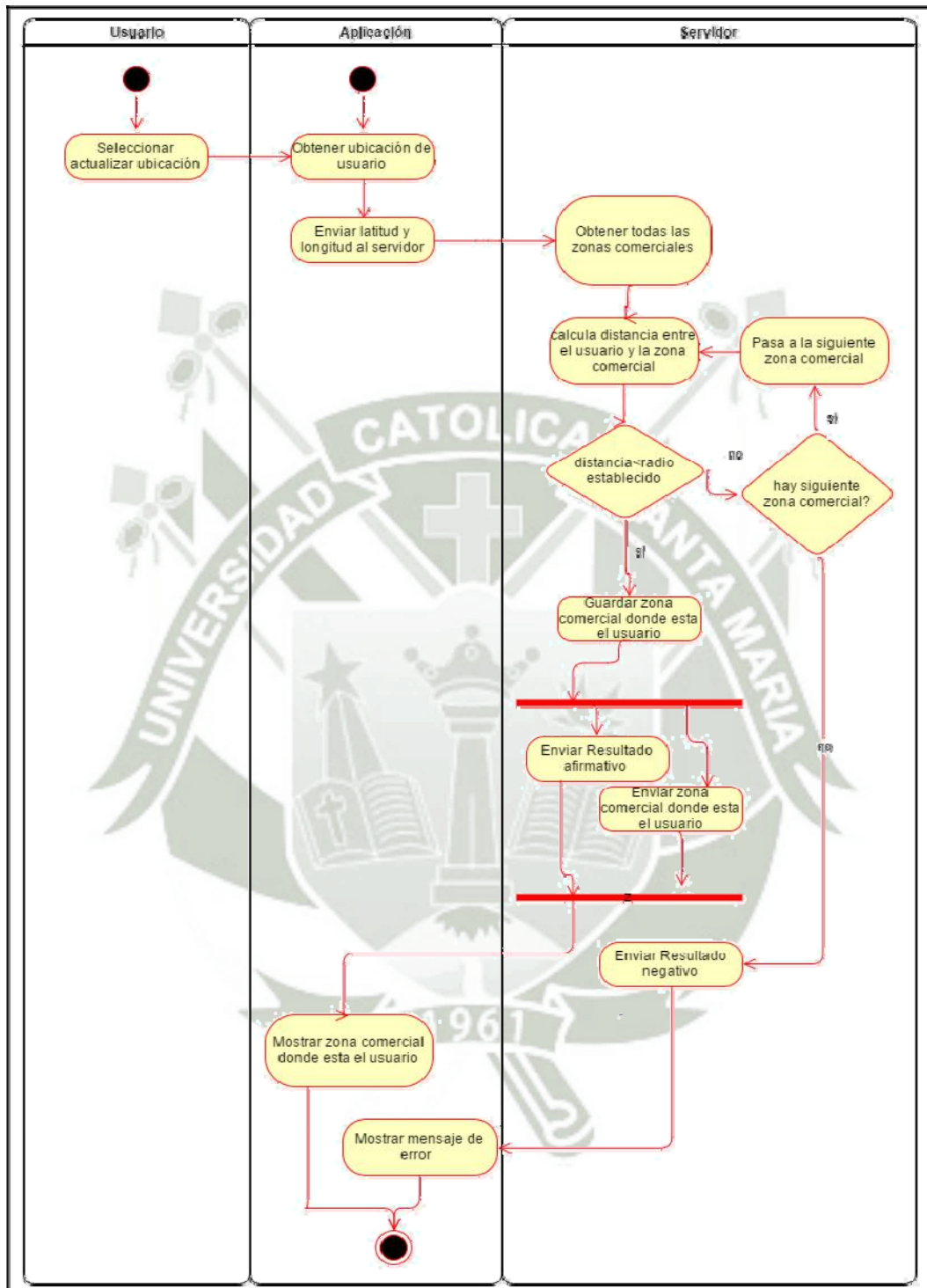


Figura 35: Diagrama de actividades de la actualización de ubicación

Fuente: Elaboración propia

En la vista física se describe como el software está organizado en los nodos físicos, es decir que muestra la distribución del sistema en el mundo real. Para esta vista se realizó el diagrama de despliegue, este consta de 4 nodos: el teléfono móvil con sistema operativo iOS; el reloj de la marca Apple®, en ambos nodos se ejecuta una aplicación nativa ; el servidor de Aplicaciones *Openshift*, que es una plataforma como servicio(PaaS), en ella se encuentra el servicio web “SOAP” y el bean de seguridad que se encarga de la autenticación de los mensajes de solicitud y la base de Datos, en la cual se encuentra el gestor de base de datos *PostgreSQL*. Para la correcta comunicación del Servidor con la aplicación móvil, se tiene una interfaz llamada “AdmazingPortType”, esta es la fachada de acceso a los servicios brindados por el servicio web.

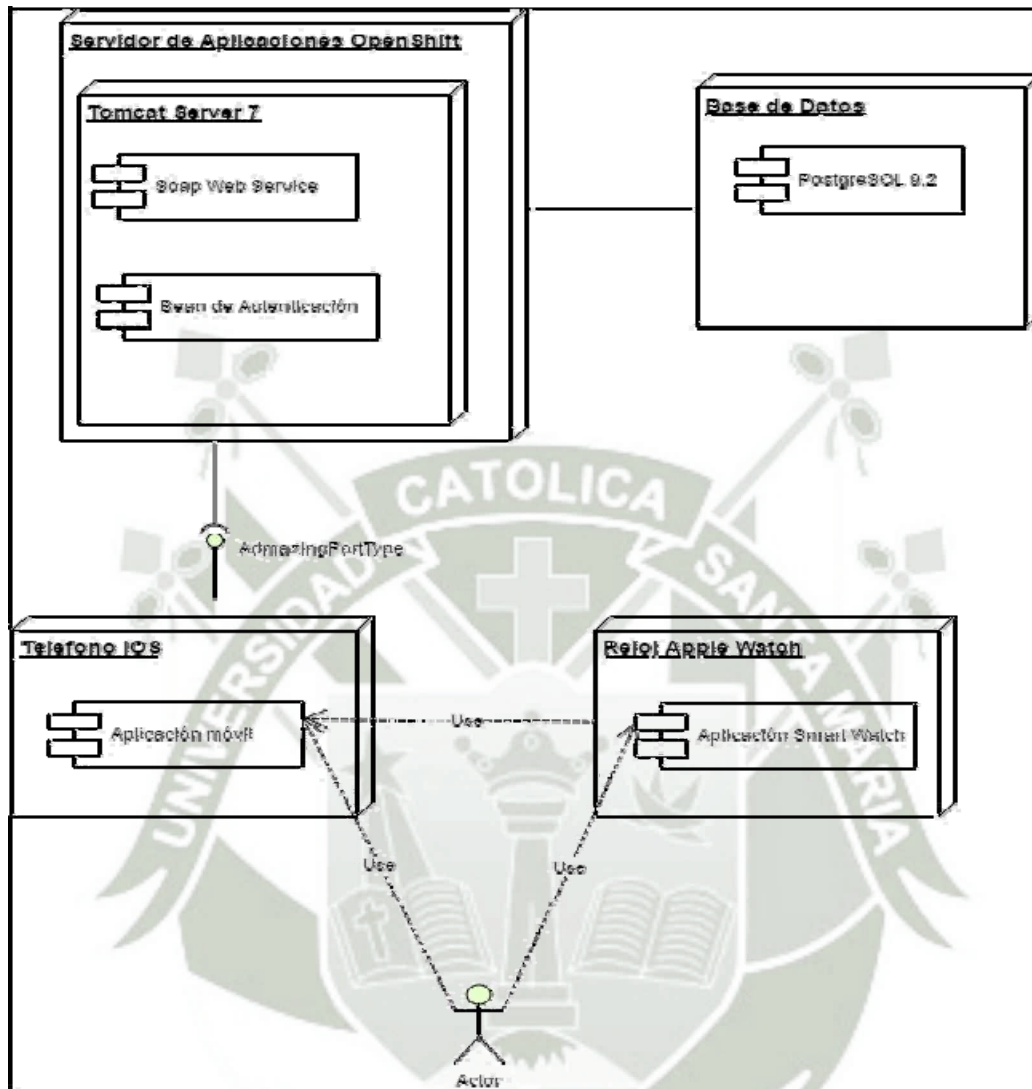


Figura 36: Diagrama de despliegue del sistema de promociones
Fuente: Elaboración propia

2.4 Documentación técnica de programación

2.4.1 Documentación de la aplicación móvil

Para la realización de la aplicación móvil se utilizó el IDE *XCode*, el cual permite realizar aplicaciones para dispositivos con los sistemas operativos IOS y WatchOS, se trabajó con ambos sistemas operativos para el desarrollo de este proyecto. En ambos sistemas operativos se trabajó con el lenguaje de programación Swift.

Para que el sistema tenga una adecuada flexibilidad e independencia de los componentes, se eligió utilizar la arquitectura de tres capas. La aplicación móvil y wearable se encuentran dentro de la capa de vista. Ambas aplicaciones se encargan de la entrada y salida de datos en este sistema, sólo son utilizadas como interfaz del sistema con el usuario y no se encarga de realizar procesamiento de datos, dado que este se realiza completamente dentro del servicio web. La aplicación móvil tiene múltiples funcionalidades importantes para detallar, las cuales serán mostradas a continuación con sus respectivos detalles.

La primera funcionalidad corresponde a la conexión que se realiza entre la aplicación móvil y el servicio web, para esto trabaja con mensajes SOAP y XML como protocolos de conexión entre ambos sistemas. A continuación se muestra la función principal de conexión del sistema, se puede observar que la conexión debe contar con el mensaje SOAP para poder enviar los datos al servicio web, además

se debe ingresar la dirección URL del servicio web; todo esto es ingresado dentro de un objeto NSMutableURLRequest, el provisto por el lenguaje Swift para realizar este tipo de conexiones.

Posteriormente se crea un objeto NSURLConnection para realizar la solicitud síncrona hacia el servicio web, utilizando los datos que se ingresaron dentro del objeto NSMutableURLRequest. La respuesta del servicio web se guarda dentro de una variable del tipo NSMutableData, la cual debe ser codificada al formato String y posteriormente procesada mediante un objeto de la clase NSXMLParser. De esta manera ya se puede guardar estos datos dentro de un arreglo de objetos para mostrarlo al usuario utilizando los componentes visuales que correspondan a la función que se desea realizar.


```

func createConnection(body: String)
{
    let soapMessage = setMessage(body)

    let urlString = "http://webws-adsws.rhcloud.com/services/admazing?wsdl"
    let url = NSURL(string: urlString)

    let theRequest = NSMutableURLRequest(URL: url!)

    let msgLength = soapMessage.characters.count

    theRequest.addValue("text/xml; charset=utf-8", forHTTPHeaderField: "Content-Type")
    theRequest.addValue(String(msgLength), forHTTPHeaderField: "Content-Length")
    theRequest.HTTPMethod = "POST"
    theRequest.HTTPBody = soapMessage.dataUsingEncoding(NSUTF8StringEncoding, allowLossyConversion:
false)
    let response: AutoreleasingUnsafeMutablePointer<NSURLResponse?>=nil
    do{
        let data = try NSURLConnection.sendSynchronousRequest(theRequest, returningResponse: response)
        let encodedData = NSString(data: data, encoding: NSUTF8StringEncoding)
        let xmlParser = NSXMLParser(data: data)
        xmlParser.delegate = self
        xmlParser.parse()
        xmlParser.shouldResolveExternalEntities = true
    }catch let error as NSError
    {
        print(error.localizedDescription)
    }
}
}

```

Figura 37 : Código función de conexión de la aplicación móvil con el servicio web

Fuente: Elaboración propia

A continuación se mostrará la funcionalidad de reconocimiento de la ubicación del usuario utilizando el sistema de geolocalización del dispositivo móvil, esta función tiene como nombre GPS. En la siguiente imagen se puede observar que es necesario crear un objeto de la clase CLLocationManager el cual se encarga de obtener la ubicación del usuario, a continuación se indica que se necesita la aprobación del usuario para poder obtener su ubicación, debido a que se desea determinar si el usuario se encuentra dentro de una zona comercial, luego se solicita la ubicación del usuario. Si la anterior función fue realizada con éxito, se obtiene la latitud y longitud de la posición actual del usuario y ambos datos se

envían al servicio web donde se determina si el usuario está dentro de una zona comercial y devuelve la información de dicha zona. Por última se muestra una confirmación utilizando un objeto de la clase UIAlertController, en la alerta se incluye la información de la zona, si esta fue reconocida por el servicio web. De otra manera se indica al usuario que no se encuentra dentro de una zona comercial y se vuelve a pedir su ubicación.

```
loadCurrentLog()
manager = CLLocationManager()
manager.requestAlwaysAuthorization()
manager.delegate = self
manager.desiredAccuracy = kCLLocationAccuracyBest
manager.requestWhenInUseAuthorization()
manager.requestLocation()
if manager.location != nil{
    r_latitude = String(manager.location!.coordinate.latitude)
    r_longitude = String(manager.location!.coordinate.longitude)

    closeMall = sendToServer()
    if(closeMall != ""){
        fillCurrentLocation()
        saveCurrentLocation()

        let alert = UIAlertController(title: "Suces", message: "Latitud: \(r_latitude),
            Longitud: \(r_longitude), Se encuentra en el centro comercial \(closeMall)",
            preferredStyle: UIAlertControllerStyle.Alert)
        alert.addAction(UIAlertAction(title: "OK", style: UIAlertActionStyle.Default, handler: {
            action in self.performSegueWithIdentifier("goMainMenu", sender: self)}))
        self.presentViewController(alert, animated: true, completion: nil)
        NSUserDefaults().setValue(closeMall, forKey: "closeMall")
    }
    else{
        let alert = UIAlertController(title: "Alert", message: "No se encuentra cerca a una zona
            comercial establecida", preferredStyle: UIAlertControllerStyle.Alert)
        alert.addAction(UIAlertAction(title: "OK", style: UIAlertActionStyle.Default, handler:
            nil))
        self.presentViewController(alert, animated: true, completion: nil)
    }
}
```

Figura 38 : Código función de obtención de ubicación geográfica del usuario

Fuente: Elaboración propia

A continuación se mostrará las características de la funcionalidad que permite a un usuario compartir una promoción con sus amigos utilizando las redes sociales de Facebook® y Twitter®. Para compartir la promoción utilizando

Facebook®, se usa un objeto del tipo `SLComposeViewController` para poder mostrar una vista en la que se carga un objeto del tipo `SLServiceTypeFacebook` con el cual se puede compartir información a través de Facebook®, después se añade el texto que se desea compartir, utilizando la función `setInitialText` y se carga la imagen de la promoción seleccionada por el usuario y luego de tener todos los campos llenos se comparte la promoción, para este se debe tener en cuenta que el usuario ya debe estar autenticado previamente en la aplicación de Facebook® dentro del dispositivo móvil, dado que usa esos datos de autenticación para el servicio con el que se está trabajando.

Para compartir la promoción utilizando Twitter® también se usa un objeto de la clase `SLComposeViewController` para poder mostrar una vista en la que se carga un objeto del tipo `SLServiceTypeTwitter`, en la cual permite al usuario hacer un Tweet con los detalles de la promoción seleccionada anteriormente dentro de esta red social. En la siguiente imagen se puede observar que se debe poner un texto inicial dentro de la publicación, el texto debe tener la longitud de caracteres permitidos por esta red social el cual es 140. Además también se debe añadir la imagen de la promoción a compartir dentro de esta red social. Finalmente con todos los datos previamente explicados se realiza la publicación, para realizar esta operación, el usuario también debe estar autenticado en la aplicación de Twitter® dentro del dispositivo móvil, para poder realizar el Tweet desde dicha cuenta autenticada.

```

@IBAction func shareCouponFacebook(sender: AnyObject) {
    if SLComposeViewController.isAvailableForServiceType(SLServiceTypeFacebook){
        var facebookSheet:SLComposeViewController =
            SLComposeViewController(forServiceType: SLServiceTypeFacebook)
        facebookSheet.setInitialText("Les invito a usar esta promoción en la aplicación Admazing! \n\n
            \(\coupon.promodescription) \n Disponible en la tienda \(\store.nameStore) del centro
            comercial \(\closeMall) \n Precio Original: S/\(\product.price) \n Valido hasta: \(\coupon.
            endDate) ")
        facebookSheet.addImage(self.couponImage.image)
        //--->store name and commercial area left
        self.presentViewController(facebookSheet, animated: true, completion: nil)
    } else {
        var alert = UIAlertController(title:
            "Accounts", message: "Please login to a Facebook account to share.", preferredStyle:
            UIAlertControllerStyle.Alert)
        alert.addAction(UIAlertAction(title: "OK", style: UIAlertActionStyle.Default, handlers: nil))
        self.presentViewController(alert, animated: true, completion: nil)
    }
}

@IBAction func shareCouponTwitter(senders: AnyObject) {
    if SLComposeViewController.isAvailableForServiceType(SLServiceTypeTwitter) {
        var tweetShare:SLComposeViewController =
            SLComposeViewController(forServiceType: SLServiceTypeTwitter)
        tweetShare.setInitialText("Les invito a usar esta promoción en la aplicación Admazing! \n\n \
            (\coupon.promodescription)")
        tweetShare.addImage(self.couponImage.image)

        self.presentViewController(tweetShare, animated: true, completion: nil)
    } else {
        var alert = UIAlertController(title: "Accounts", message: "Please login to a Twitter account
            to tweet.", preferredStyle: UIAlertControllerStyle.Alert)

        alert.addAction(UIAlertAction(title: "OK", style: UIAlertActionStyle.Default, handlers: nil))

        self.presentViewController(alert, animated: true, completion: nil)
    }
}

```

Figura 39 : Código funciones de publicación de promociones usando Facebook y Twitter
Fuente: Elaboración propia

Posteriormente se muestra la funcionalidad de generación del código QR para el uso de la promoción que el usuario desea utilizar, para esto ya se tiene el String que se desea convertir a QR, el String está guardado en la variable MessageQR, a continuación se realiza la conversión a código QR con la función dataUsingEncoding para la cual se usa el filtro CIQRCodeGenerator. Por último, el código QR obtenido se muestra dentro del imageView de la actividad en la que se encuentra el usuario. Este código debe ser leído por un scanner dentro del

establecimiento que tiene la promoción, de esta manera la promoción será usada satisfactoriamente por el usuario.

```
func fit(QRImage) {  
    if messageQR == "" {  
        return  
    }  
    else{  
        let data = messageQR.dataUsingEncoding(NSISOLatin1StringEncoding, allowLossyConversion:  
            false)  
  
        let filter = CIFilter(name: "CIQRCodeGenerator")  
  
        filter!.setValue(data, forKey: "inputMessage")  
        filter!.setValue("Q", forKey: "inputCorrectionLevel")  
  
        qrcodeImage = filter!.outputImage  
        displayQRCodeImage()  
    }  
}  
func displayQRCodeImage() {  
    let scaleX = qrImageView.frame.size.width / qrcodeImage.extent.size.width  
    let scaleY = qrImageView.frame.size.height / qrcodeImage.extent.size.height  
  
    let transformedImage = qrcodeImage.imageByApplyingTransform(CGAffineTransformMakeScale(scaleX,  
        scaleY))  
  
    qrImageView.image = UIImage(CIImage: transformedImage)  
}
```

Figura 40: Código función de generación de código QR para uso de una promoción

Fuente: Elaboración propia

Por último se puede observar la manera en la que se realizaron las actividades de la aplicación, esto corresponde a la parte visual de la aplicación, se puede observar en la siguiente imagen la manera en que se disponen los elementos visuales dentro de cada actividad de la aplicación, de esta manera se pueden tener todos los elementos de entrada y salida de datos de la manera en la que se podrán ver dentro de la aplicación. Además se programó la interacción que se tienen entre las actividades de la aplicación, de esta manera se dispone el flujo con el que el usuario utilizará la aplicación.



Figura 41 : Desarrollo de interfaces y vínculos entre las actividades de la aplicación

Fuente: Elaboración propia

La capa de vista de la arquitectura de tres capas, corresponde directamente con dos elementos muy importantes, la aplicación móvil para teléfonos celulares, la cual fue realizada para el sistema operativo IOS, y la aplicación para wearables, la cual fue realizada el sistema operativo WatchOS. Todos los componentes y funcionalidad que corresponden a estas dos aplicaciones fueron basadas en las recomendaciones de la Guía de interfaz humana de IOS desarrollada por Apple Developers (2016). Esta guía presenta tres principios de diseño en una interfaz IOS, la claridad, la empatía y la profundidad del elemento visual de la aplicación.

En el principio de claridad de la interfaz se tomaron en cuenta diferentes recomendaciones que se brindan, como el tamaño y la legibilidad del texto en los diferentes mensajes que se muestran durante la ejecución de la aplicación, la lucidez y la precisión de los iconos que se utilizaron para cada funcionalidad que se brinda,

la precisión en cuanto a espacios sobre las barras e iconos que se usan dentro de la aplicación según su importancia y funcionalidad, los colores utilizados dentro de la aplicación son los que brinda Apple® según sus estándares para cada tipo de acción, por ejemplo el color azul para acciones que se usó en los botones que denotaban una acción específica o el color rojo para elegir una opción negativa importante como la cancelación del uso de una promoción o cerrar la sesión del usuario, también se tomó en cuenta el cambio o alteración de color de un elemento dentro de una tabla cuando este está seleccionado o un botón que fue activado, para indicarle al usuario la acción que está siendo procesada por el sistema.

En el principio de Empatía con el usuario, se tomaron en cuenta recomendaciones como la fluidez de la aplicación en cuanto a las acciones que desea realizar el usuario dentro de las funcionalidad brindadas, la cantidad de elementos controlados dentro de la interfaz, para asegurar importancia de la información brindada al usuario dentro de cada vista, la facilidad de comprender la interfaz, gracias a la poca complejidad que se buscó implementar para satisfacer todas las necesidades del usuario y la muestra de alertas implementada con un control de errores cuando el usuario se encuentre dentro un escenario anómalo por parte del sistema y no se pierda la fluidez en cuanto al desempeño de la aplicación.

En el principio de la profundidad, se tomó en cuenta recomendaciones como tener diferentes capas y jerarquía dentro de cada vista en la aplicación, por lo que se tienen vistas importantes como el menú principal o la cuponera virtual, la

facilidad de entendimiento del usuario con el uso de herramientas u opciones que sean importantes y el flujo constante de información que se presenta al usuario mediante las alertas que se muestran.

Las características mostradas previamente se aplicaron también en la aplicación wearable en su totalidad y además se agregó el principio de “los tres clics” que se usa en el desarrollo web, para poder garantizar que con solo tres clics por parte del usuario, se podrá completar una funcionalidad del sistema. Al ser la aplicación wearable más simple que la aplicación para el teléfono móvil se puede utilizar este principio y con solo ese número de clics lograr el uso de la promoción que el usuario desee.

A continuación se muestran las alertas que se presentan en esta aplicación para poder comunicar al usuario sobre un suceso inesperado o informarle que las acciones que le ordenó al sistema se realizaron con éxito y de esta manera guiar al usuario.

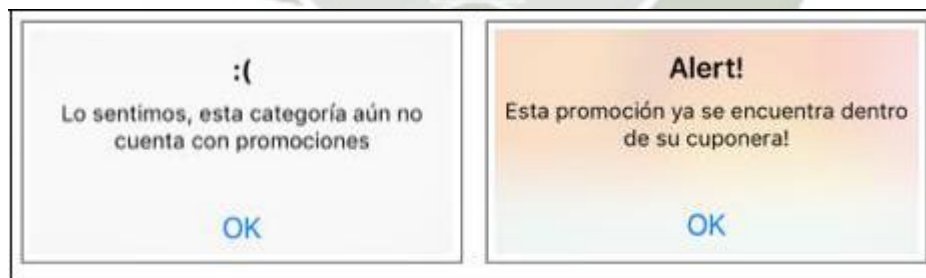


Figura 42: Alertas mostradas dentro de la aplicación cuando se produce un problema

Fuente: Elaboración propia

Las siguientes alertas se usan para poder confirmar al usuario que las acciones que desea realizar se llevaron a cabo con éxito y brindando un mayor detalles sobre estas acciones, por ejemplo en la tercera alerta se puede observar que se indica al usuario la latitud y longitud en la que este se encuentra, además de la zona comercial en la que se encuentra.



Figura 43: Confirmaciones mostradas en la aplicación cuando se realiza una función satisfactoriamente

Fuente: Elaboración propia

Dentro de la aplicación móvil se puede observar dos menús, el primero es el menú principal de la aplicación el cual cuenta con cuatro opciones importantes y el segundo es el menú de configuraciones.

En el menú principal, la primera opción lleva el nombre de la aplicación, dentro de esta se puede acceder a las promociones de las tiendas que se encuentran en la zona comercial donde está el usuario, la segunda opción tiene como nombre

Cupones, dentro de esta opción se puede encontrar la cuponera virtual del usuario y la posición de todas las tiendas que brindan estas promociones; la tercera opción tiene como nombre Ubicación, como dice su nombre, esta opción permite al usuario poder actualizar su ubicación en el caso que este se encuentre dentro de una zona comercial diferente a la programada previamente; y por último se tiene la cuarta opción con el nombre de Configuraciones, dentro de esta opción se tiene todo el centro de configuraciones que permite personalizar las preferencias del usuario.



Figura 44: Menú principal de la aplicación móvil
Fuente: Elaboración propia

El segundo menú se muestra cuando se selecciona la opción configuraciones en el menú principal, dentro de este se puede observar el nombre del usuario, además se pueden observar dos opciones para poder marcarse con un

switch, la primera opción será marcada si el usuario desea usar la publicidad que le presenta la aplicación basada en las categorías que son de su preferencia, y la segunda, si el usuario desea recibir alertas cuando sus cupones están prontos a vencer.



Figura 45: Actividad de configuraciones dentro de la aplicación móvil
Fuente: Elaboración propia

Dentro de la aplicación wearable se encuentra el menú principal, el cual también cuenta con cuatro opciones, la primera llamada GPS, se encarga de actualizar la posición satelital del usuario y hacer esta actualización dentro del web service de este sistema; la segunda opción tiene como nombre Último cupón, en esta opción se puede observar el último cupón visto dentro de la aplicación del teléfono celular y tener un acceso rápido a la visualización de esta; la tercera opción

se llama Usar Último cupón, como su nombre lo dice, esta opción te permite usar directamente el último cupón visto en la aplicación móvil sin una visualización previa por parte del usuario; y por último se tiene la opción de configuraciones, la cual permite al usuario usar las funciones de la aplicación según sus preferencias.



Figura 46: Menú principal de la aplicación wearable

Fuente: Elaboración propia

Esta aplicación solo cuenta con una barra de herramientas, la cual se sitúa en la parte superior de la mayoría de las vistas, sin embargo esta cambia dependiendo de la vista en la que se encuentra. La barra de herramientas cuenta con dos botones y el logo de la aplicación, el símbolo de la izquierda siempre se encargará de navegación a través de todas las vista brindando el diferentes casos el botón home que envía al usuario al menú principal, o en otros casos el botón atrás para poder enviar al usuario a la vista anterior. En la parte derecha de esta barra de herramienta, en todos los casos se presentará el botón que lleva al usuario a la vista de configuraciones de la aplicación.

A continuación se muestran los tres tipos de barras de herramientas, el primero se presenta en vistas cuya posición en navegación se encuentra cercana al menú principal, la segunda opción se presenta en vistas cuya posición de navegación es más profunda y se prioriza volver a la vista anterior en vez de volver al menú principal y perder todos los datos de navegación. La tercera y última opción se presenta solo en la vista en la que se genera el código QR de la promoción, en la cual no se precisa volver a la vista anterior.

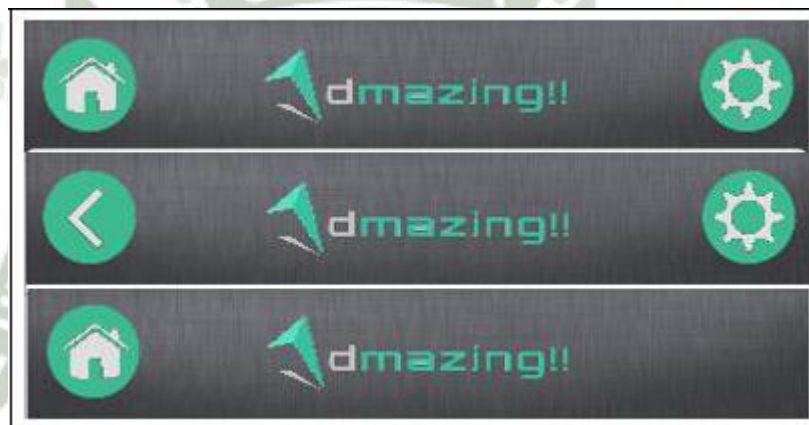


Figura 47: Barras de herramientas dentro de la aplicación móvil
Fuente: Elaboración propia

En la interfaz gráfica de usuario de esta aplicación se presentan diferentes tipos de vistas, a continuación se mostrarán todos estos tipos y sus principales características.

Primero se presentan las vistas de Inicio de Sesión, dentro de la vista de Inicio de sesión sólo se puede observar cuatro componentes: el logo de la aplicación en la parte superior de la vista, una casilla de texto en donde se debe ingresar el

nombre de usuario, otra casilla de texto en la que el usuario debe ingresar su contraseña; y por último un botón que permite realizar la función de autenticación del usuario.

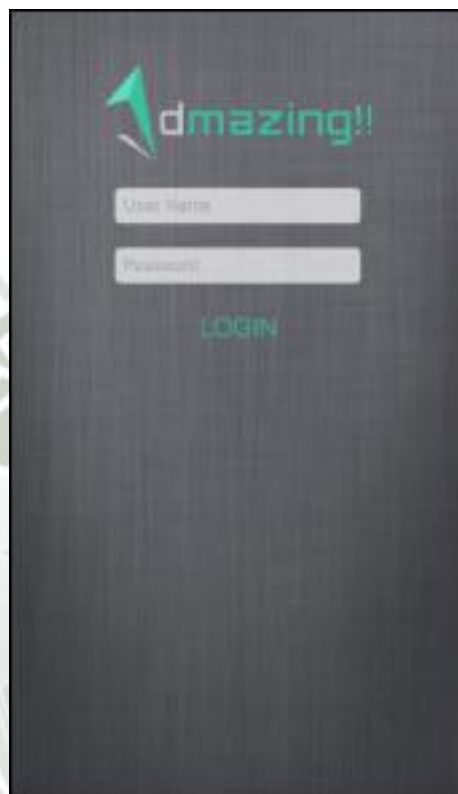


Figura 48: Actividad de ingreso al sistema dentro de la aplicación móvil
Fuente: Elaboración propia

A continuación se muestran las vistas que contienen listados de objetos, estas vistas pueden contener la lista de establecimientos cercanos al usuario o la lista de promociones que tiene cada establecimiento de la zona comercial, en ambos casos se trabaja con el objeto `TableViewController` que provee el entorno gráfico de IOS. Se puede observar que cada celda en ambos casos cuenta con una

imagen de referencia, en el caso de los establecimientos, muestra el logo del establecimiento y en el caso de las promociones, muestra la imagen del producto o servicio que se brinda; cada celda cuenta también con una descripción escrita, en el caso de los establecimientos cuenta con el nombre de este y en caso de las promociones, una descripción sobre las promociones.

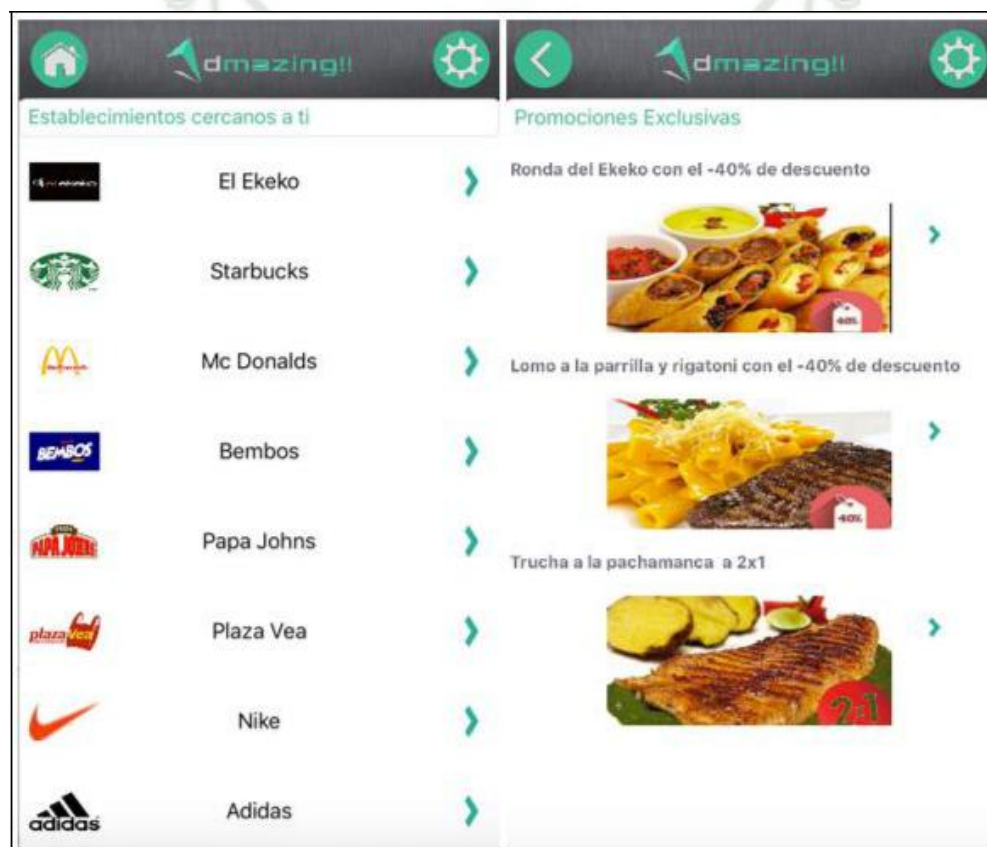


Figura 49: Actividades dentro de la aplicación móvil que presentan tablas de datos

Fuente: Elaboración propia

A continuación se muestra la vista de categorías dentro de un establecimiento, esta vista cuenta con un mapa, que muestra un punto azul que indica la ubicación actual del usuario y un icono de color rojo, la ubicación de la tienda en el mapa para poder guiar al usuario y mostrarle el lugar donde se encuentra el establecimiento a donde desea ir, este mapa se trabaja con el objeto MapKitView, además en la parte inferior se muestra un listado de las categorías que pertenecen al establecimiento seleccionado previamente. Dentro de cada celda de esta tabla se encuentra el icono de las categorías y una pequeña descripción de ella, de igual manera esta vista trabaja con el objeto TableViewController.

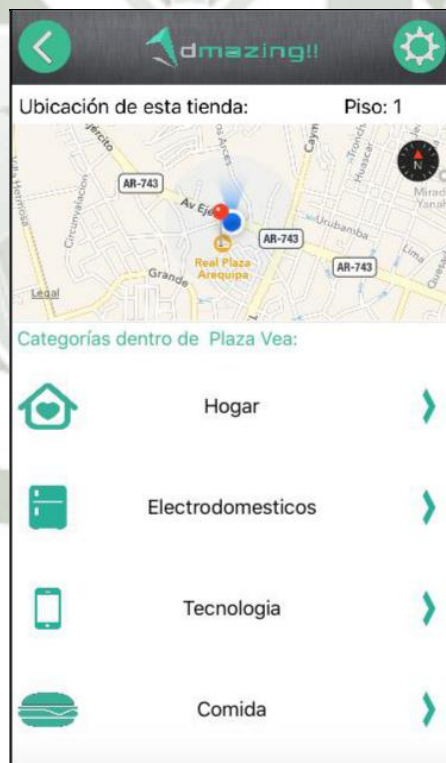


Figura 50: Actividad de categorías de la aplicación móvil
Fuente: Elaboración propia

En la vista de descripción de la promoción se puede observar la imagen de la promoción en la parte superior y un pequeño icono con la descripción del tipo de promoción en la esquina inferior izquierda de la imagen, en el ejemplo se presenta una oferta del 40% de descuento en el producto. Debajo de la imagen, se muestra la descripción de la promoción y la descripción del producto, además se muestran datos importantes para el cliente, por ejemplo el costo real del producto y la fecha de caducidad de la promoción. En la parte inferior se presentan dos botones, el primero para agregar dicha promoción en la cuponera y la segunda para poder acceder a la cuponera virtual.



Figura 51: Actividad que muestra una promoción dentro de la aplicación móvil
Fuente: Elaboración propia

La vista del cupón dentro de la cuponera es similar a la vista anterior, se muestran los datos principales de la promoción y del producto relacionado a ella, sin embargo se cambian las opciones de la parte inferior, en esta vista se presentan opciones para usar y descartar la promoción de la cuponera, además se añaden dos botones en la parte inferior para poder compartir esta promoción utilizando las redes sociales de Facebook® y Twitter®.



Figura 52: Actividad que muestra de la promoción dentro de la cuponera virtual en la aplicación móvil

Fuente: Elaboración propia

De igual manera en la aplicación para el wearable, se muestra la descripción de la promoción, la descripción del producto o servicio y una opción para poder utilizar la promoción seleccionada, además de la función de navegación para poder regresar a la vista anterior.



Figura 53: Actividad de muestra de la promoción dentro de la cuponera virtual en la aplicación wearable
Fuente: Elaboración propia

A continuación se muestra las ventanas que se crean para compartir la promoción a través de las redes sociales de Facebook® y Twitter® respectivamente, en ambos casos se debe usar el objeto o SLComposeViewController que provee el entorno gráfico de IOS.



Figura 54: Actividad de publicación de promociones usando Facebook o Twitter
Fuente: Elaboración propia

La siguiente vista corresponde a la que contiene el código QR generado por el sistema, cuenta con un mensaje en el que pide al usuario escanear dicho código por el encargado de la tienda para poder utilizar la promoción, la siguiente imagen que contiene el código QR y en la parte inferior contiene dos opciones, la primera para poder confirmar el uso del cupón y la segunda para poder cancelar el uso de la promoción.



Figura 55: Generación del código QR de la promoción seleccionada en la aplicación móvil
Fuente: Elaboración propia

En la aplicación para el wearable, se tiene la vista correspondiente al uso de promoción, en ella se muestra el código QR generado por la aplicación, el mensaje para su uso y la opción para poder confirmar el uso de la promoción, tal como se puede observar en la siguiente imagen.



Figura 56: Generación del código QR de la promoción seleccionada en la aplicación wearable
Fuente: Elaboración propia

2.4.2 Documentación del servicio web

Para la realización del servicio Web del sistema de publicidad, se eligió utilizar la tecnología SOAP, SOAP es un protocolo que permite la transferencia de estructuras de Datos a través de XML, gracias a esto el servicio puede ser accedido a través de cualquier dispositivo. El principal beneficio que ofrece SOAP, es que tiene ya implementado el manejo de errores, por ello en caso que exista un error, el servicio web retorna información sobre el error facilitando su corrección. Además SOAP brinda múltiples estándares, que facilitan la creación de algunas funciones comunes en los servicios web, tales como WS-Addressing, WS-Policy, WS-Security, WS-Federation, WS-ReliableMessaging, WS-Coordination, WS-AtomicTransaction, etc. Oracle (2011). En el desarrollo del sistema de

promociones, se utilizó el estándar WS-Security para realizar la autenticación del mensaje.

Para la implementación del servicio web SOAP se utilizó el lenguaje Java porque es orientado a objetos, es muy flexible, tiene una gran cantidad de librerías disponibles que permiten que la codificación sea más sencilla y rápida. Se utilizó Eclipse como ambiente de desarrollo o IDE, porque es altamente utilizado por desarrolladores del lenguaje java, brinda múltiples herramientas para que la codificación y la detección de errores sean más sencillas y también porque está basado en patrones.

Para el manejo del proyecto en eclipse se utilizó *Maven*. *Maven* es un software para la administración de proyectos que se basa en el POM, este es un archivo que administra el proyecto. Este software ayuda a agilizar la instalación de los plugin o dependencias necesarias en un solo archivo y administrar sus respectivas propiedades. Apache (2016). De esta manera se logra eliminar redundancias en las instalaciones. En la siguiente imagen se muestra un extracto de las dependencias que se encuentran dentro del proyecto.

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <project xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
3   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
4   <modelVersion>4.0.0</modelVersion>
5   <groupId>com.admazing</groupId>
6   <artifactId>webws</artifactId>
7   <version>1.0</version>
8   <packaging>war</packaging>
9   <name>Admazing WS</name>
10  <description>AdmazingWS</description>
11  <properties>
12    <hibernate.version>4.2.15.Final</hibernate.version>
13    <cxf.version>3.0.2</cxf.version>
14  </properties>
15  <dependencies>
16    <dependency>
17      <groupId>org.apache.cxf</groupId>
18      <artifactId>cxf-rt-frontend-jaxws</artifactId>
19      <version>3.0.2</version>
20      <scope>compile</scope>
21    </dependency>
22    <dependency>
23      <groupId>org.apache.cxf</groupId>
24      <artifactId>cxf-rt-transports-http</artifactId>
25      <version>3.0.2</version>
26      <scope>compile</scope>
27    </dependency>
28    <dependency>
29      <groupId>org.apache.cxf</groupId>
30      <artifactId>cxf-rt-transports-http-jetty</artifactId>
31      <version>3.0.2</version>
32      <scope>compile</scope>
33    </dependency>
34    <dependency>
35      <groupId>commons-codec</groupId>
36      <artifactId>commons-codec</artifactId>
37      <version>1.10</version>
38      <scope>compile</scope>
39    </dependency>
40    <dependency>
41      <groupId>org.springframework</groupId>
42      <artifactId>spring-web</artifactId>
43      <version>4.3.10.RELEASE</version>
44      <scope>compile</scope>
45    </dependency>
46  </dependencies>
47 </project>

```

Overview Dependencies Dependency Hierarchy Effective POM pom.xml

Figura 57: Archivo POM del web service

Fuente: Elaboración propia

Además se utilizó CXF, que es un framework de la compañía Apache que ayuda a desarrollar servicios tanto en REST como en SOAP, estos servicios pueden utilizar múltiples protocolos como XML/HTTP, CORBA. Apache (2015). En el

caso del sistema de promociones, se utilizó el plugin CXF codegen que permite la creación automática de las clases necesarias a partir del WSDL, el archivo contiene todas las estructuras de datos y la especificación de las operaciones que el servicio web realiza.

```
<plugin>
  <groupId>org.apache.cxf</groupId>
  <artifactId>cxf-codegen-plugin</artifactId>
  <version>${cxf.version}</version>
  <executions>
    <execution>
      <id>generate-sources</id>
      <phase>generate-sources</phase>
      <configuration>
        <sourceRoot>${basedir}/src/generated</sourceRoot>

        <defaultOptions>
          <validateWsdL>true</validateWsdL>
        </defaultOptions>
        <wsdlOptions>

          <wsdlOption>
            <bindingFiles>
              <bindingFile>${basedir}/src/main/resources/bindings.xjb</bindingFile>
            </bindingFiles>
            <wsdl>${basedir}/src/main/webapp/WEB-INF/wsdl/Admazing.wsdl</wsdl>
          </wsdlOption>
        </wsdlOptions>
      </configuration>
    </execution>
  </executions>
</plugin>
```

Figura 58: Plugin cxf codegen
Fuente: Elaboración propia

Para la gestión de la base de datos se utilizó *PostgreSQL*, porque es software libre y su uso e instalación son más sencillos que las herramientas que ofrece Microsoft®. Además el servidor de Aplicaciones que se utilizó llamado *OpenShift*, no permite el uso de SQL Server. Como herramienta para el mapeo objeto-relacional (ORM) se utilizó *Hibernate*, un ORM permite a los desarrolladores transformar los datos que se encuentran en la base de datos como objetos relacionales, a objetos de clases de Java en el caso de *Hibernate*. Redhat (2016).

En la siguiente imagen se muestra el documento de configuración del servicio Web, en él se detallan el driver del gestor de base de datos, en este caso el driver de *PostgreSQL*, el nombre de usuario, la contraseña, el URL de conexión con la base de Datos y los mapeos correspondientes de las tablas de la base de datos.

```

1 <?xml version='1.0' encoding='UTF-8'?>
2 <!DOCTYPE hibernate-configuration PUBLIC
3     "-//Hibernate/Hibernate Configuration DTD 3.0//EN"
4     "http://www.hibernate.org/dtd/hibernate-configuration-3.0.dtd">
5
6 <hibernate-configuration>
7     <session-factory>
8
9         <property name="hibernate.connection.driver_class">org.postgresql.Driver</property>
10        <property name="hibernate.connection.username">adminsrgtmqd</property>
11        <property name="hibernate.connection.password">6p3ceu-LgxzV</property>
12        <property name="hibernate.connection.url">jdbc:postgresql://127.7.250.130:5432/webws</property>
13
14        <property name="hibernate.dialect">org.hibernate.dialect.PostgreSQLDialect</property>
15        <property name="show_sql">>true</property>
16        <property name="validationQuery">SELECT version()</property>
17        <property name="current_session_context_class">thread</property>
18        <property name="cache.provider_class">org.hibernate.cache.internal.NoCacheProvider</property>
19        <mapping resource="hibernate/User.hbm.xml"></mapping>
20        <mapping resource="hibernate/Store.hbm.xml"></mapping>
21        <mapping resource="hibernate/Category.hbm.xml"></mapping>
22        <mapping resource="hibernate/StoreCategory.hbm.xml"></mapping>
23        <mapping resource="hibernate/Product.hbm.xml"></mapping>
24        <mapping resource="hibernate/Promotion.hbm.xml"></mapping>
25        <mapping resource="hibernate/PromotionType.hbm.xml"></mapping>
26        <mapping resource="hibernate/CommercialArea.hbm.xml"></mapping>
27        <mapping resource="hibernate/Access.hbm.xml"></mapping>
28        <mapping resource="hibernate/CouponBook.hbm.xml"></mapping>
29        <mapping resource="hibernate/Preference.hbm.xml"></mapping>
30        <mapping resource="hibernate/PromotionUse.hbm.xml"></mapping>
31
32     </session-factory>
33 </hibernate-configuration>
34

```

Figura 59: Archivo de configuración de Hibernate
Fuente: Elaboración propia

A continuación se muestran extractos del código más relevantes para el proyecto. En primer lugar se muestra la codificación del patrón observador, este se observador llama a la función “update” a penas el observable sufre una modificación, en este caso el cambio se da cuando el usuario utiliza un cupón. La función “update” se encarga de obtener la categoría a la que pertenece el cupón usado. Si el usuario ha utilizado cupones de esta misma categoría un número de

veces igual o mayor que 5, se agrega la categoría a las preferencias del usuario. Esto permite personalizar las categorías del usuario a medida que utiliza el sistema. Además este patrón da la flexibilidad en caso que se desee cambiar este número máximo o poner otras restricciones como tiempo o lugar. Se utilizó este patrón de comportamiento porque en el futuro se desea agregar otros ámbitos en los que se basarán las personalizaciones, para lograr una personalización de promociones más exacta a los gustos del usuario. Esto se lograría agregando otros observadores en el futuro sin tener que realizar cambios mayores en las clases ya propuestas actualmente.

```
1 package com.admazing.Logic;
2
3*import java.util.List;
11
12 public class categoryObserver extends Observer{
13
14     private PreferenceService preferenceService;
15     private PromotionService promotionService;
16
17     public categoryObserver(PromotionUseService subject){
18         preferenceService = new PreferenceServiceImpl();
19         promotionService= new PromotionServiceImpl();
20
21         Observer.subject=subject;
22         Observer.subject.attach(this);
23     }
24
25     @Override
26     public void update(String idUser,String idPromotion) {
27         PromotionModel promotionUsed=promotionService.findById(idPromotion);
28         if(!preferenceService.exist(idUser, promotionUsed.getIdCategory())){
29             if(countPromotionsUsed(idUser,promotionUsed.getIdCategory())>=5){
30                 preferenceService.save(idUser,promotionUsed.getIdCategory());
31             }
32         }
33     }
34
35     private int countPromotionsUsed (String idUser, String idCategory){
36         int nPromotionUsed=0;
37         List<PromotionUseModel> promotionsUse = Observer.subject.getId(idUser);
38         for(PromotionUseModel promotionUse : promotionsUse){
39             PromotionModel promotion=promotionService.findById(promotionUse.getIdPromotion());
40             if(promotion.getIdCategory().compareTo(idCategory)==0){
41                 nPromotionUsed++;
42             }
43         }
44         return nPromotionUsed;
45     }
}
```

Figura 60: El patrón Observador en el web service

Fuente: Elaboración propia

En segundo lugar se tiene el patrón Singleton que se utilizó para asegurar una única instancia en la creación de la clase hibernateUtil, ya que esta es la encargada de crear los objeto de la clase sessionFactory, este objeto es el que utiliza la herramienta hibernate para crear sesiones de conexión con la base de datos, por lo mismo este objeto ocupa mucha memoria y ralentiza el funcionamiento del sistema si se crea constantemente. Por esta razón se debe crear un único objeto de esta clase para lograr que el sistema se comunique con la base de datos más eficientemente.

```
1 package com.admazing.dataAccess;
2
3 import org.hibernate.SessionFactory;
4
5 public class hibernateUtil {
6
7     private SessionFactory sessionFactory;
8     private static hibernateUtil myHibernateConfigurator;
9     private hibernateUtil(){
10
11         try {
12             sessionFactory = new Configuration().configure().buildSessionFactory();
13         } catch (Throwable ex) {
14             System.err.println("Initial SessionFactory creation failed." + ex);
15             throw new ExceptionInInitializerError(ex);
16         }
17     }
18     public static hibernateUtil getHibernateConfigurator(){
19         if(myHibernateConfigurator==null){
20             myHibernateConfigurator= new hibernateUtil();
21         }
22         return myHibernateConfigurator;
23     }
24     public SessionFactory getSessionFactory() {
25         return sessionFactory;
26     }
27     public void shutdown() {
28         getSessionFactory().close();
29     }
30 }
```

Figura 61: El patrón Singleton en el web service

Fuente: Elaboración propia

En tercer lugar, se muestra el extracto del bean creado para interceptar todas las solicitudes al servicio web, y de esta manera comprobar que el mensaje es realmente del remitente permitido. En la siguiente imagen se muestra el interceptor “<jaxws:inInterceptors>”, se muestra la obtención de los datos que se envían en la cabecera de la solicitud y se envía a la clase correspondiente que verifica que la contraseña enviada corresponde al nombre del usuario. En caso de aprobar la verificación, el mensaje sigue su curso usual, caso contrario envía un mensaje de error.

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <beans xmlns="http://www.springframework.org/schema/beans"
3   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:jaxws="http://cxf.apache.org/jaxws"
4   xmlns:cxf="http://cxf.apache.org/core" xmlns:soap="http://cxf.apache.org/bindings/soap"
5   xsi:schemaLocation="http://cxf.apache.org/core http://cxf.apache.org/schemas/core.xsd http://www
6
7
8   <jaxws:endpoint id="admazingWs" address="/admazing"
9     wsdlLocation="WEB-INF/wsdl/Admazing.wsdl" implementor="com.admazing.Facade.AdmazingWSImpl"
10    serviceName="admazing:AdmazingService"
11    xmlns:admazing="http://admazing.com/">
12
13   <jaxws:inInterceptors>
14     <bean class="org.apache.cxf.ws.security.wss4j.WSS4JInInterceptor">
15       <constructor-arg>
16         <map>
17           <entry key="action" value="UsernameToken" />
18           <entry key="passwordType" value="PasswordText" />
19           <entry key="passwordCallbackRef" value-ref="myPasswordCallback" />
20         </map>
21       </constructor-arg>
22     </bean>
23   </jaxws:inInterceptors>
24 </jaxws:endpoint>
25
26 <bean id="myPasswordCallback"
27   class="com.admazing.Security.UTPasswordCallback"/>
28
29 <cxf:bus>
30   <cxf:features>
31     <cxf:logging />
32   </cxf:features>
33 </cxf:bus>
34 </beans>

```

Figura 62: Interceptor para la autenticación del web service

Fuente: Elaboración propia

En cuarto lugar, se muestra el algoritmo que determina la zona comercial donde el usuario se encuentra. La función `getByLatitudeLongitude` recibe la latitud y longitud del usuario, y compara si la distancia entre el usuario y una zona comercial es menor al radio establecido para la zona comercial. En caso que sea menor se devuelve el nombre de la zona comercial, en caso contrario se devuelve el valor nulo, para que la aplicación se encargue de mostrar al usuario el mensaje de error correspondiente.

```

1 package com.admazing.Logic;
2
3 import java.util.List;
4
5
6
7
8
9
10 public class CommercialAreaServiceImpl implements CommercialAreaService {
11
12
13     private CommercialAreaRepository commercialAreaRepository= new CommercialAreaRepositoryImpl();
14
15     @Override
16     public CommercialAreaModel getByLatitudeLongitude(String latitude, String longitude) {
17         List<CommercialAreaModel> commercialAreas=commercialAreaRepository.getAll();
18         double latitudeSearched =Double.parseDouble(latitude);
19         double longitudeSearched =Double.parseDouble(longitude);
20         CommercialAreaModel commercialArea = null;
21         for (CommercialAreaModel c : commercialAreas) {
22             double currentLatitude=Double.parseDouble(c.getLatitude());
23             double currentLongitude=Double.parseDouble(c.getLongitude());
24             double currentDistance=Double.parseDouble(c.getDistance());
25             double distance=Math.sqrt(Math.pow((latitudeSearched-currentLatitude),2)
26                 +Math.pow((longitudeSearched-currentLongitude),2));
27             if(distance<=currentDistance){
28                 commercialArea= c;
29                 break;
30             }
31         }
32         return commercialArea;
33     }
34 }

```

Figura 63: Algoritmo de búsqueda de zona comercial

Fuente: Elaboración propia

2.5 Pruebas

2.5.1 Pruebas de verificación

2.5.1.1 Pruebas de ejecución

Para las pruebas del servicio web se utilizó la herramienta “SoapUI”, en la versión 5.2.1, la herramienta permite crear solicitudes SOAP y analizar si la respuesta es la correcta. Para ello se crearon aserciones que verifican si la respuesta es la esperada, en caso que la respuesta es la esperada el icono se mostrará verde, caso contrario será rojo. Además verifica si el esquema de la respuesta es correcto con respecto al documento WSDL proporcionado.

La primera prueba consta de dos solicitudes de la función para el inicio de sesión, en la tabla usuario se tiene un registro con el código de usuario “cnpm18@hotmail.com” y la contraseña “cris12345”. Como se muestra en la siguiente imagen la respuesta del inicio de sesión fue verdadero por consiguiente cumple con la respuesta esperada descrita en el recuadro más pequeño con título “Contains Assertion”.

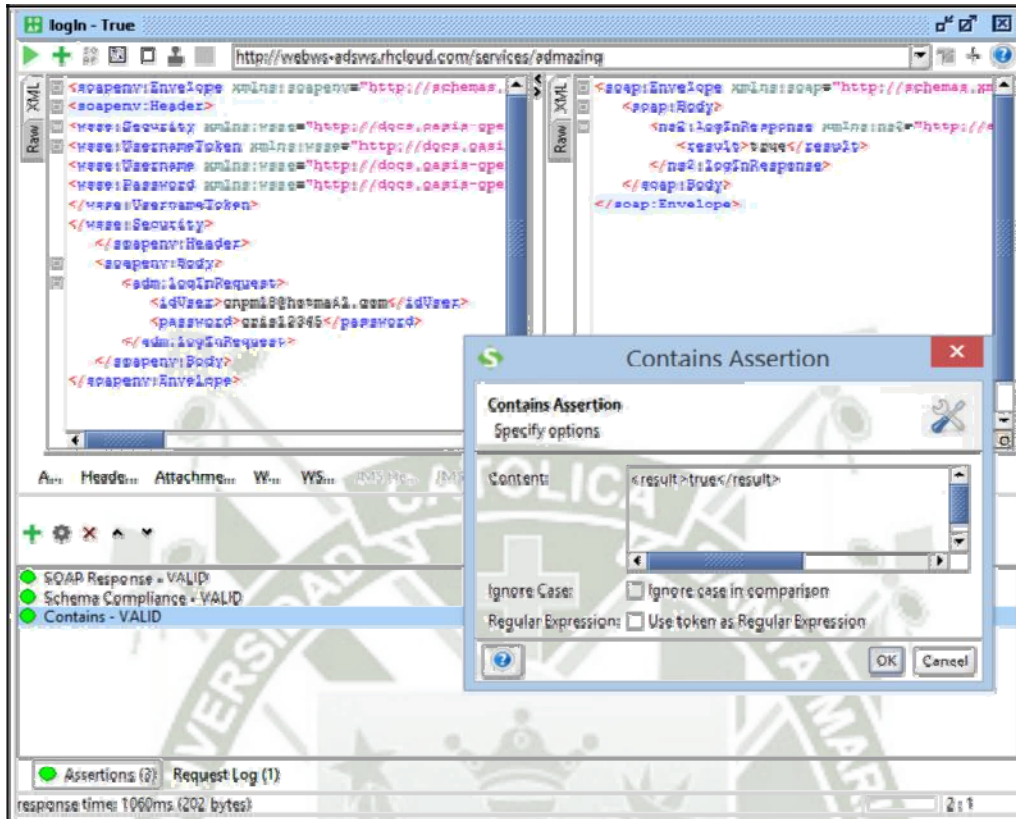


Figura 64: Prueba de iniciar sesión con resultado verdadero
Fuente: Elaboración propia

La siguiente imagen muestra un inicio de sesión inválido por lo cual la respuesta esperada debe contener falso como respuesta.

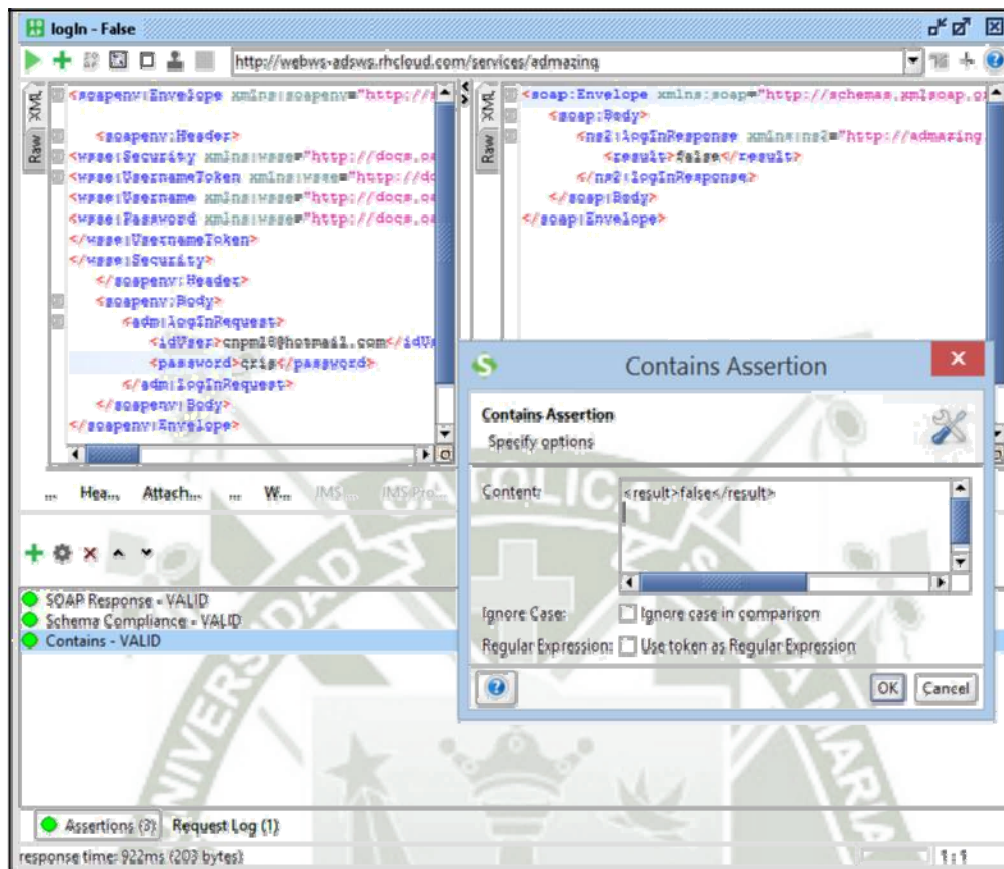


Figura 65: Prueba de iniciar sesión con resultado falso
Fuente: Elaboración propia

La segunda prueba consta de dos solicitudes a la función que se encarga del guardado de acceso, primero se muestra la situación en la que el usuario se encuentre dentro de una zona comercial guardada, el resultado que devuelve es verdadero y devuelve el nombre de la zona comercial en la que se encuentra, en este caso es el “Parque Lambramani Arequipa”.

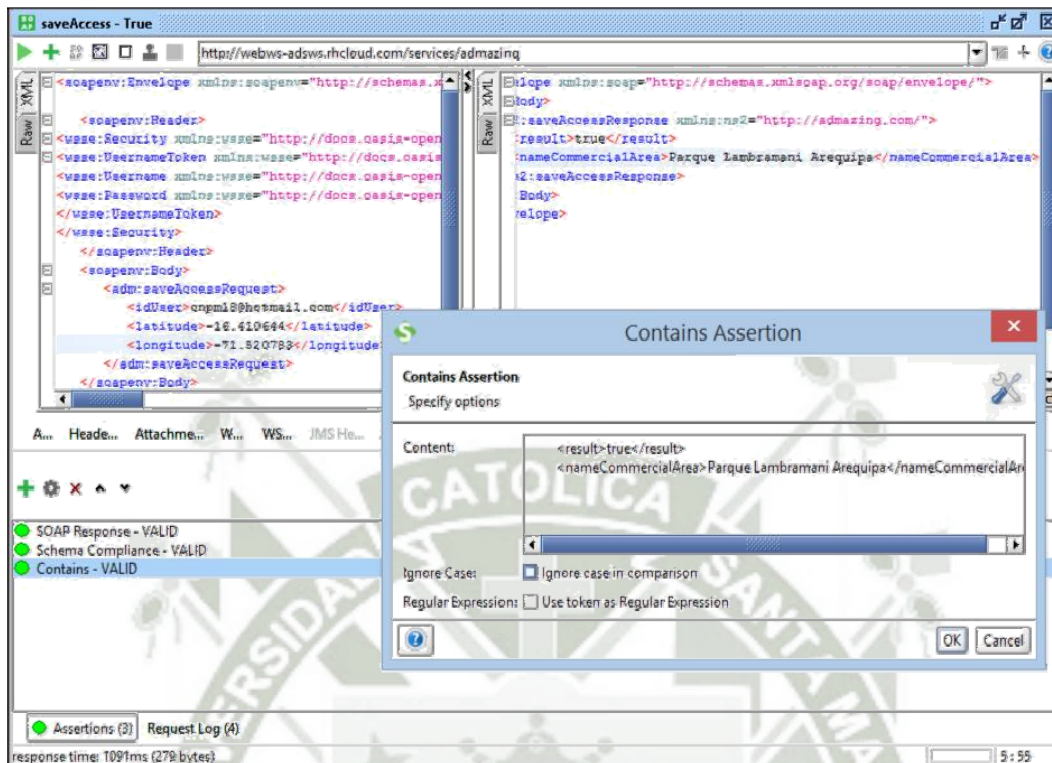


Figura 66: Prueba de guardar acceso cuando el usuario está dentro de una zona comercial
Fuente: Elaboración propia

En la siguiente imagen se muestra el resultado de la función en caso que el usuario no se encuentre en ninguna zona comercial, devuelve el resultado falso.

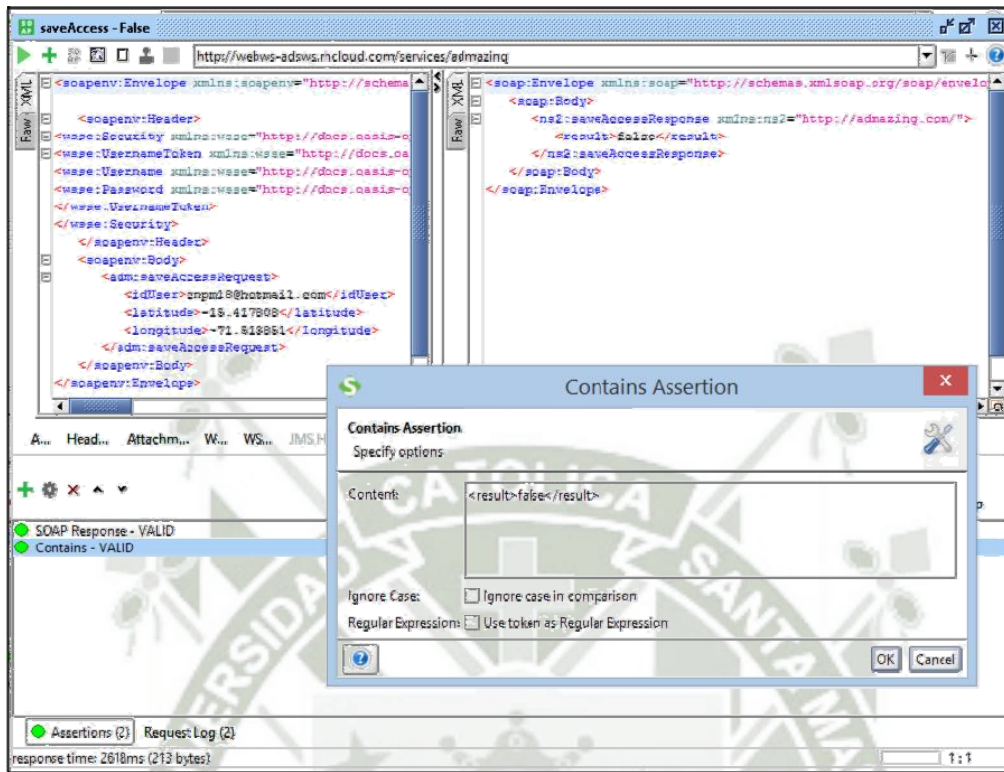


Figura 67: Prueba de guardar acceso cuando el usuario no está dentro de una zona comercial
Fuente: Elaboración propia

La tercera prueba consta de una solicitud de la función para la obtención de todas las tiendas de la zona comercial donde se encuentra el usuario para ello se envía el código del usuario y recibe los datos completos de cada tienda, en este caso como la zona es el “Parque Lambramani Arequipa” las tiendas son Paris, La Ibérica, Cinemark, Chilis, entre otros.

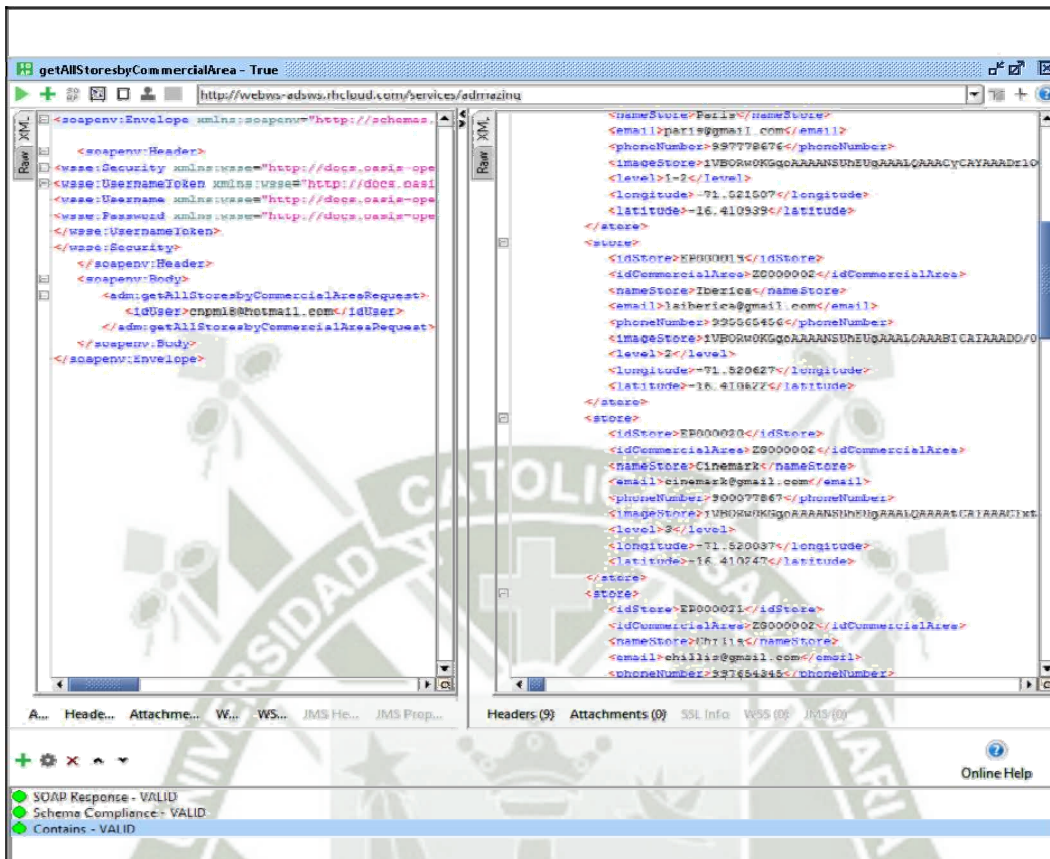


Figura 68: Prueba de obtener las tiendas según su zona comercial
Fuente: Elaboración propia

La cuarta prueba consta de una solicitud de la función que obtiene los cupones guardados en la cuponera que aplican a la zona comercial donde se encuentra el usuario para ello se envía el código del usuario y recibe los datos completos de cada cupón, en este caso solo se encuentra un cupón para un chocolate fondant con 40% de descuento.

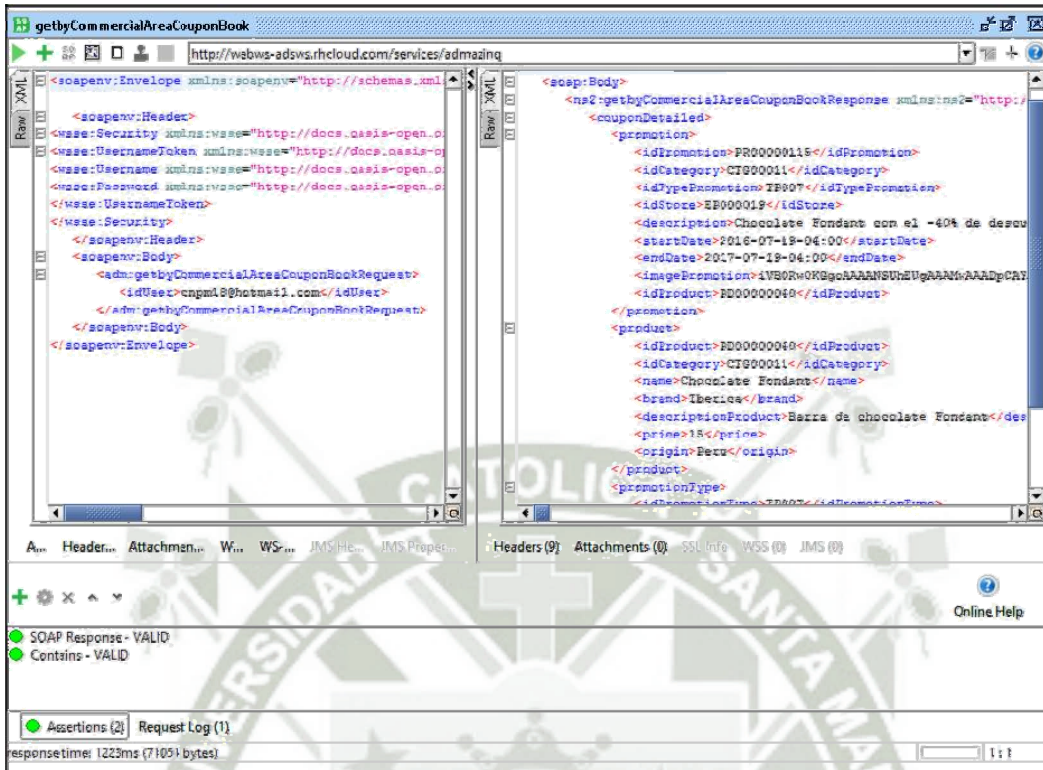


Figura 69: Prueba de obtener los cupones según su zona comercial
Fuente: Elaboración propia

La quinta prueba consta de dos solicitudes de la función para el guardado de un cupón en la cuponera para ello se envía el código del usuario y el código del cupón que se desea guardar y recibe verdadero o falso dependiendo si la operación fue exitosa o no. En la siguiente imagen se observa que el cupón de código “PR00000027” se guarda exitosamente en la cuponera del usuario `cnpm18@hotmail.com`, ya que el resultado es verdadero.

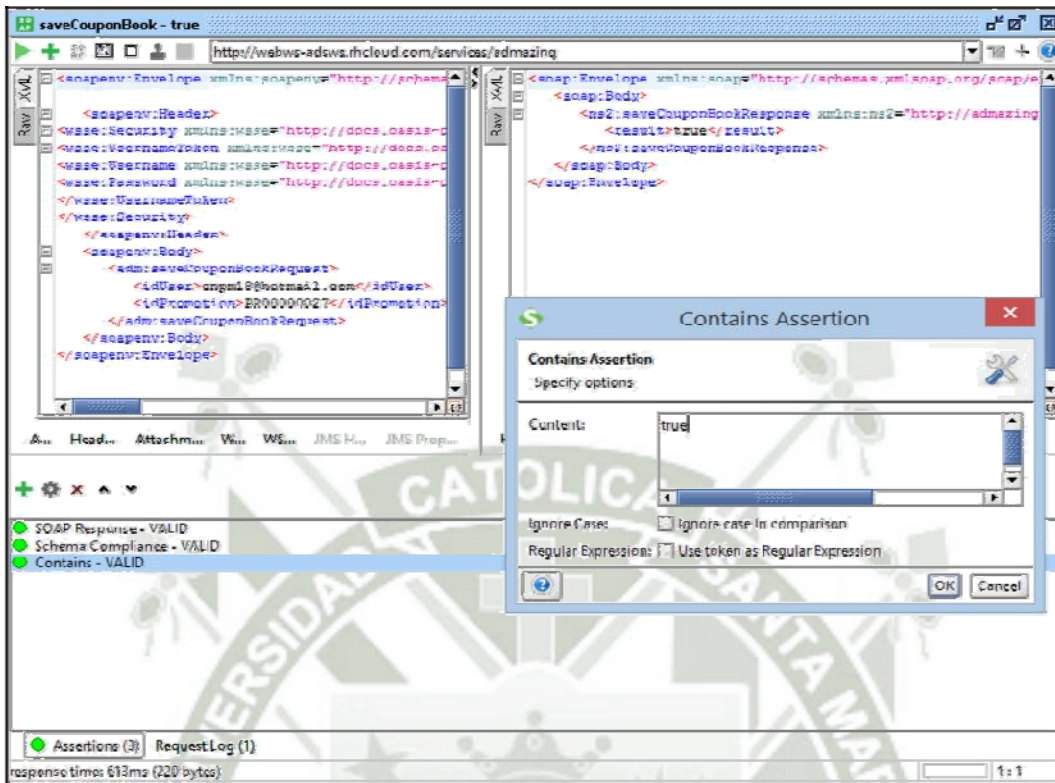


Figura 70: Prueba de guardar un cupón exitosamente

Fuente: Elaboración propia

En la siguiente imagen se envía la misma solicitud con los mismos datos, sin embargo el resultado es falso porque el cupón que se desea agregar ya se encuentra en la cuponera actualmente.

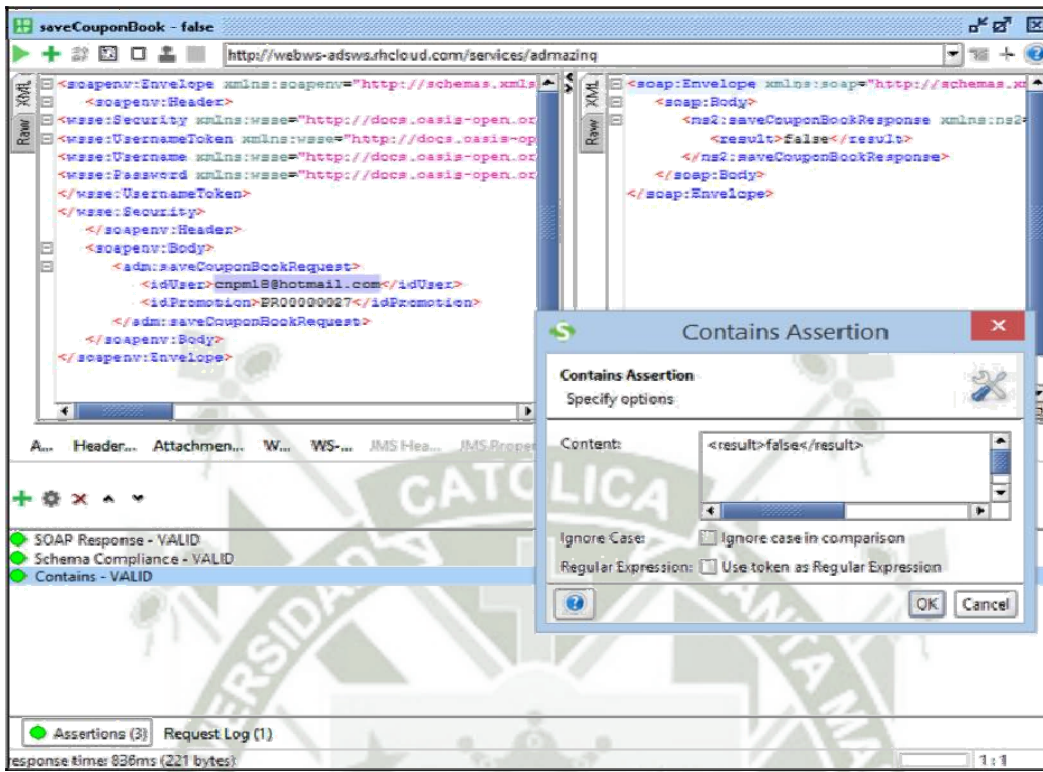


Figura 71: Prueba de guardar un cupón fallidamente

Fuente: Elaboración propia

La sexta prueba consta de cuatro solicitudes para observar el cambio en las preferencias del usuario. Primero se muestra las categorías que el usuario tiene, antes del uso de un cupón. En la siguiente imagen se observa que la categoría dos llamada electrodomésticos no se encuentra entre las preferencias seleccionadas del usuario.

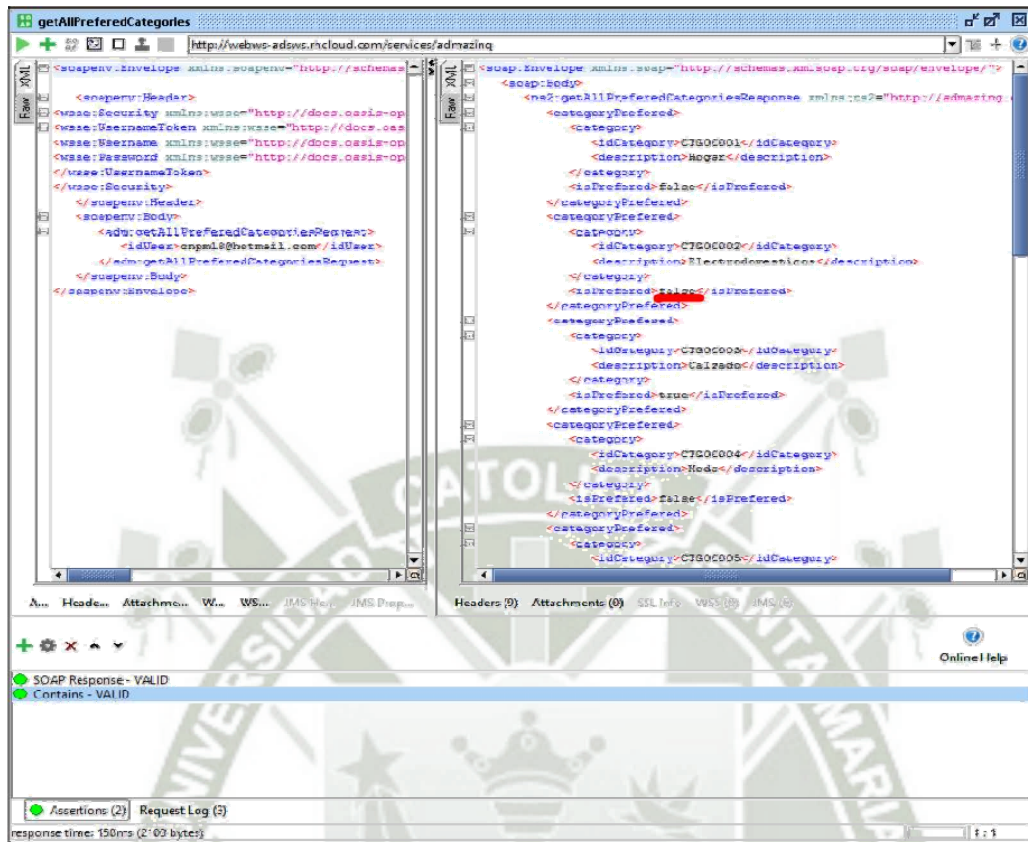


Figura 72: Primera prueba de obtener las categorías preferidas
Fuente: Elaboración propia

A continuación se muestra que el usuario `cnpm18@hotmail.com` utiliza el cupón de código “PR00000027” que se guardó en la anterior prueba, se llama a la función “savePromotionUse” incluyendo estos datos, se elimina de la cuponera y recibe verdadero o falso dependiendo si la operación fue exitosa o no. En la siguiente imagen se observa que el cupón fue usado exitosamente, ya que el resultado es verdadero.

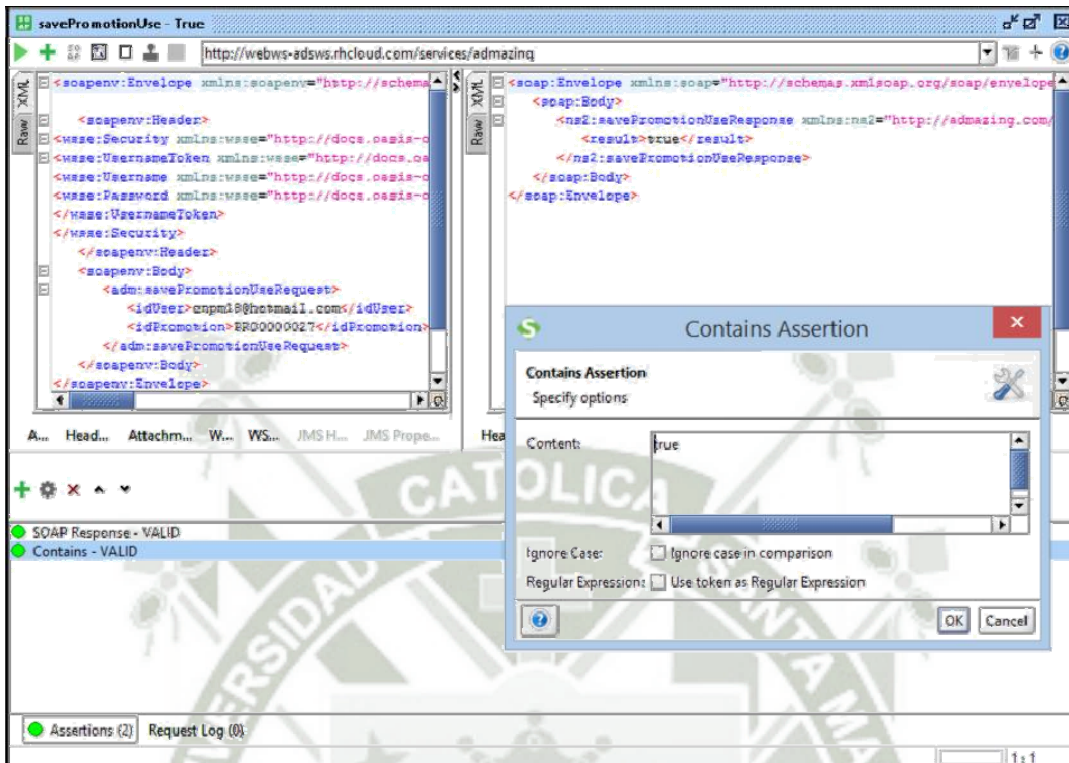


Figura 73: Prueba de usar un cupón exitosamente
Fuente: Elaboración propia

En la siguiente imagen se envía la misma solicitud con los mismos datos, sin embargo el resultado es falso porque el cupón que se desea usar ya no se encuentra en la cuponera actualmente.

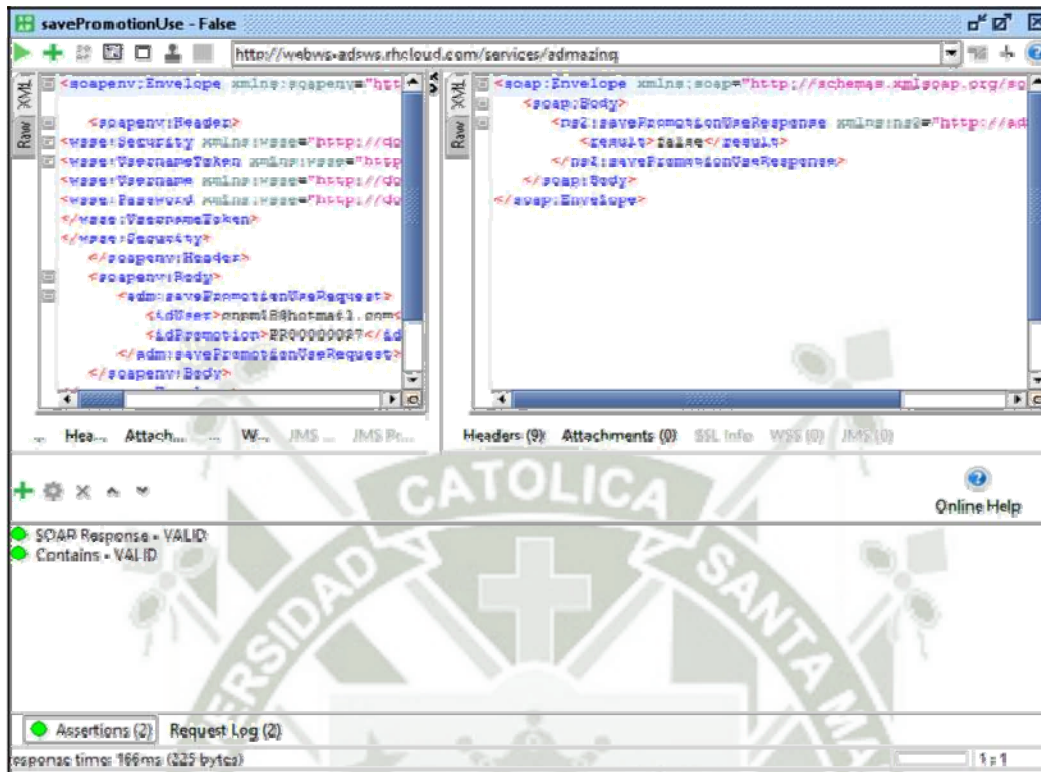


Figura 74: Prueba de usar un cupón fallidamente
Fuente: Elaboración propia

Dado que este cupón pertenece a la categoría dos, y ha sido usado constantemente, cuando se utiliza nuevamente este cupón, la categoría es agregada automáticamente a sus preferencias, tal como se muestra en la imagen.

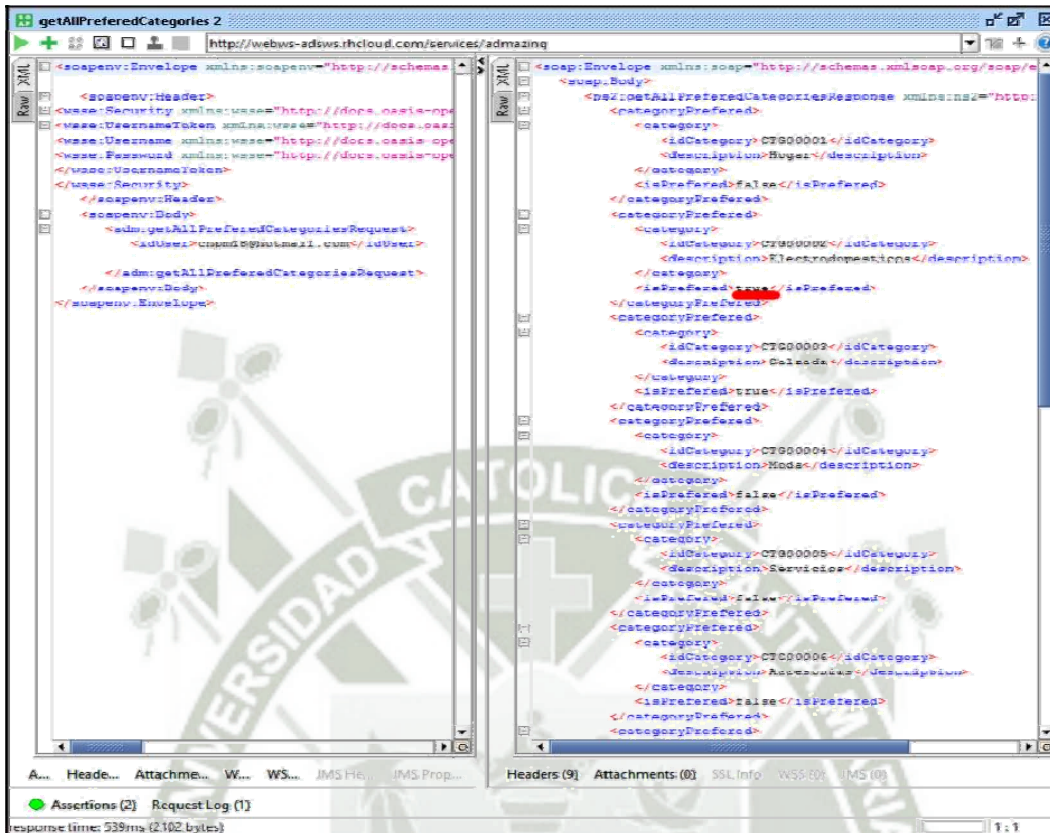


Figura 75: Segunda prueba de obtener las categorías preferidas
Fuente: Elaboración propia

La séptima prueba consta de dos solicitudes a la función de eliminación de un cupón en la cuponera para ello se envía el código del usuario y el código del cupón que se desea eliminar y recibe verdadero o falso dependiendo si la operación fue exitosa o no. Para esta prueba se utilizará el cupón de código “PR00000027”, que fue agregado anteriormente a la cuponera. En la siguiente imagen se observa que el cupón se elimina exitosamente de la cuponera del usuario `cnpm18@hotmail.com`, ya que el resultado es verdadero.

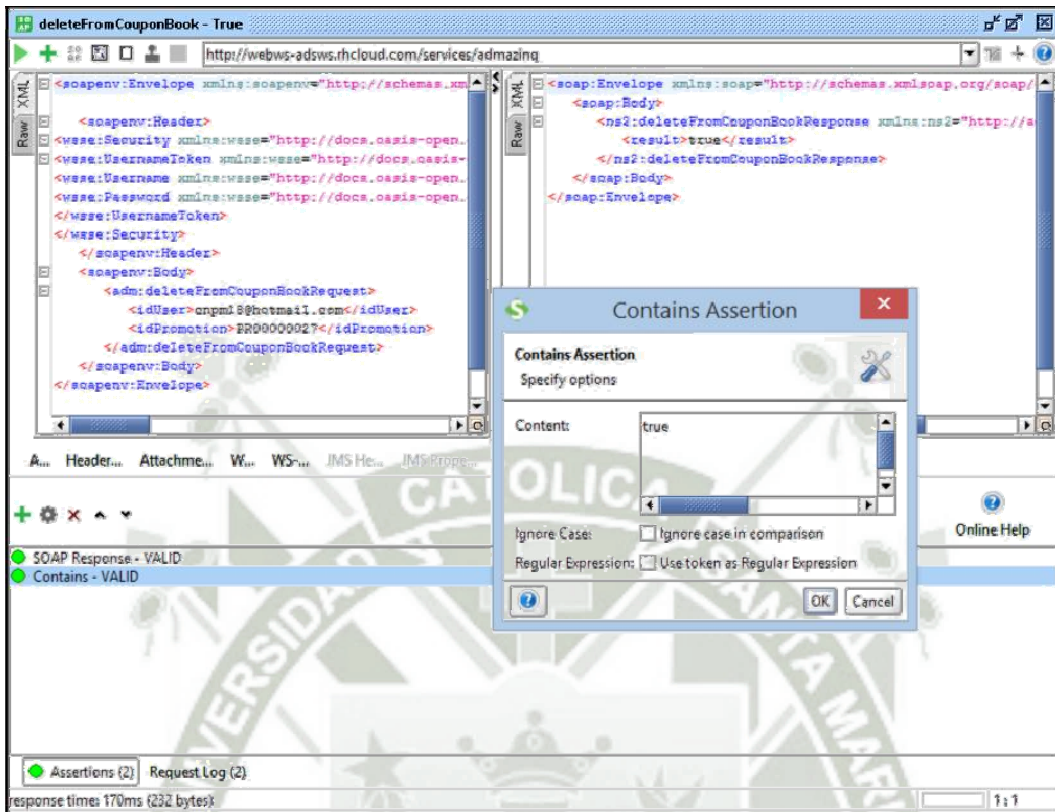


Figura 76: Prueba de eliminar un cupón exitosamente
Fuente: Elaboración propia

En la siguiente imagen se envía la misma solicitud con los mismos datos, sin embargo el resultado es falso porque el cupón que se desea eliminar ya no se encuentra en la cuponera actualmente.

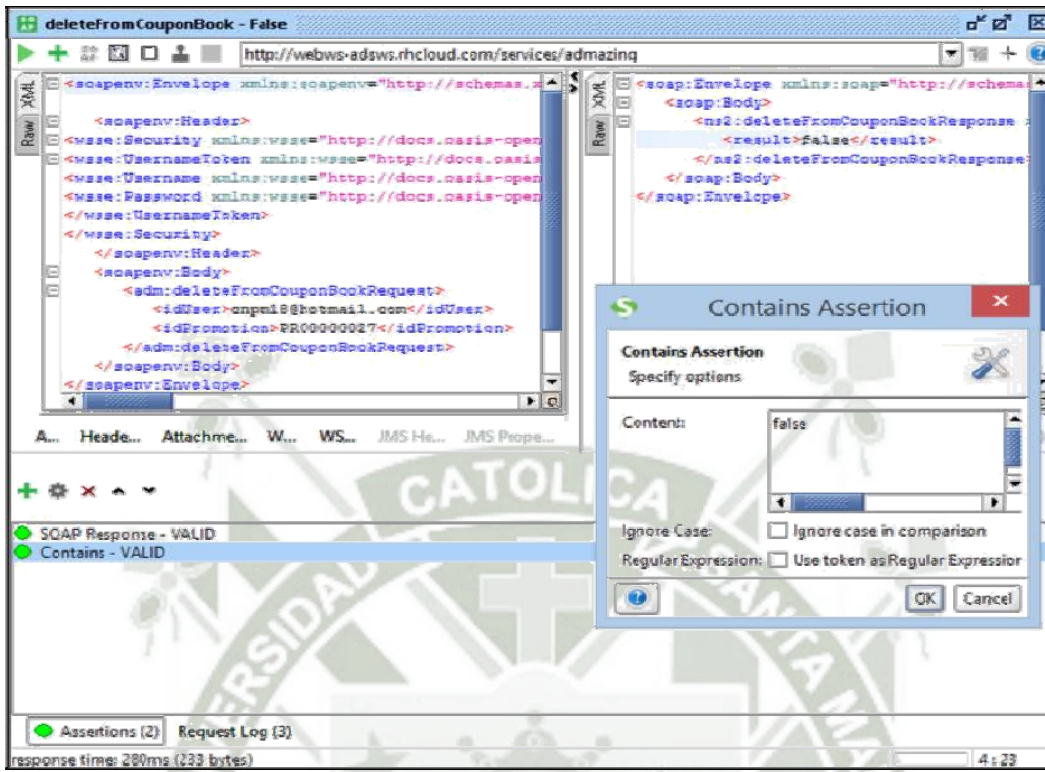


Figura 77: Prueba de eliminar un cupón fallidamente
Fuente: Elaboración propia

La octava prueba consta de cuatro solicitudes, esta muestra la gestión de preferencias para ello se envía el código del usuario y el código de la categoría que se desea guardar o eliminar y recibe verdadero o falso dependiendo si la operación fue exitosa o no. Para esta prueba se utilizará la categoría de código “CTG00009”.

En la siguiente imagen se observa que se guarda exitosamente la categoría entre las preferencias del usuario cnpm18@hotmail.com, ya que el resultado es verdadero.

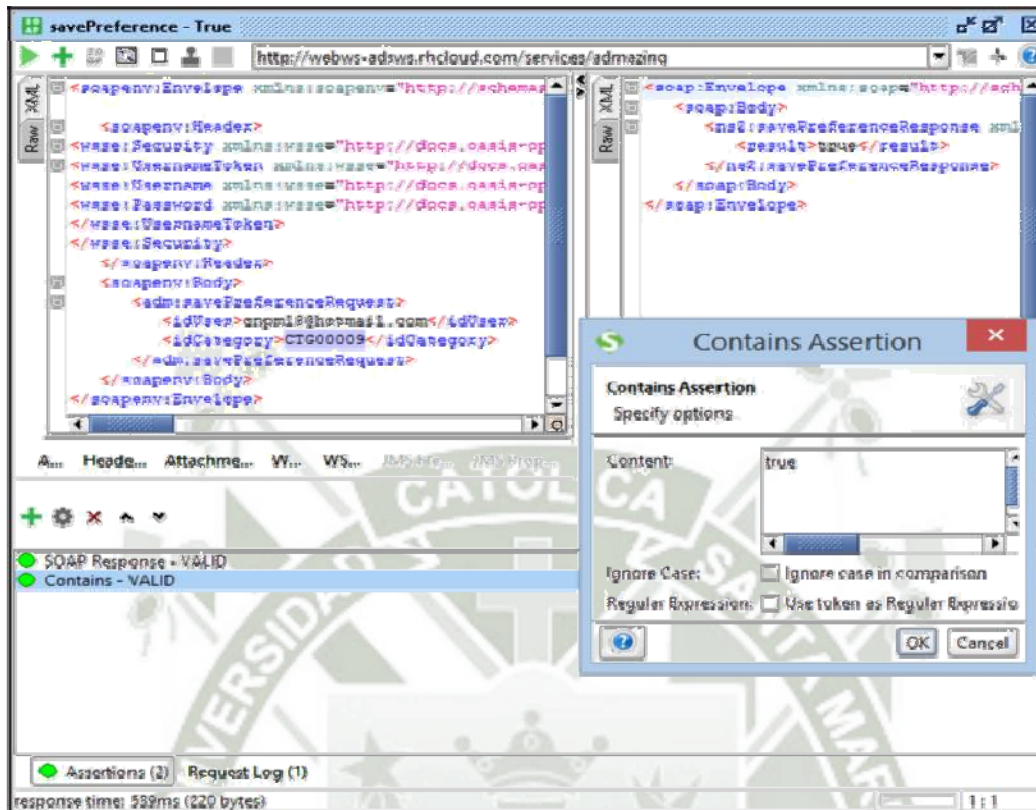


Figura 78: Prueba de guardar una categoría exitosamente

Fuente: Elaboración propia

En la siguiente imagen se envía la misma solicitud con los mismos datos, sin embargo el resultado es falso porque la categoría que se desea guardar ya está entre las preferencias.

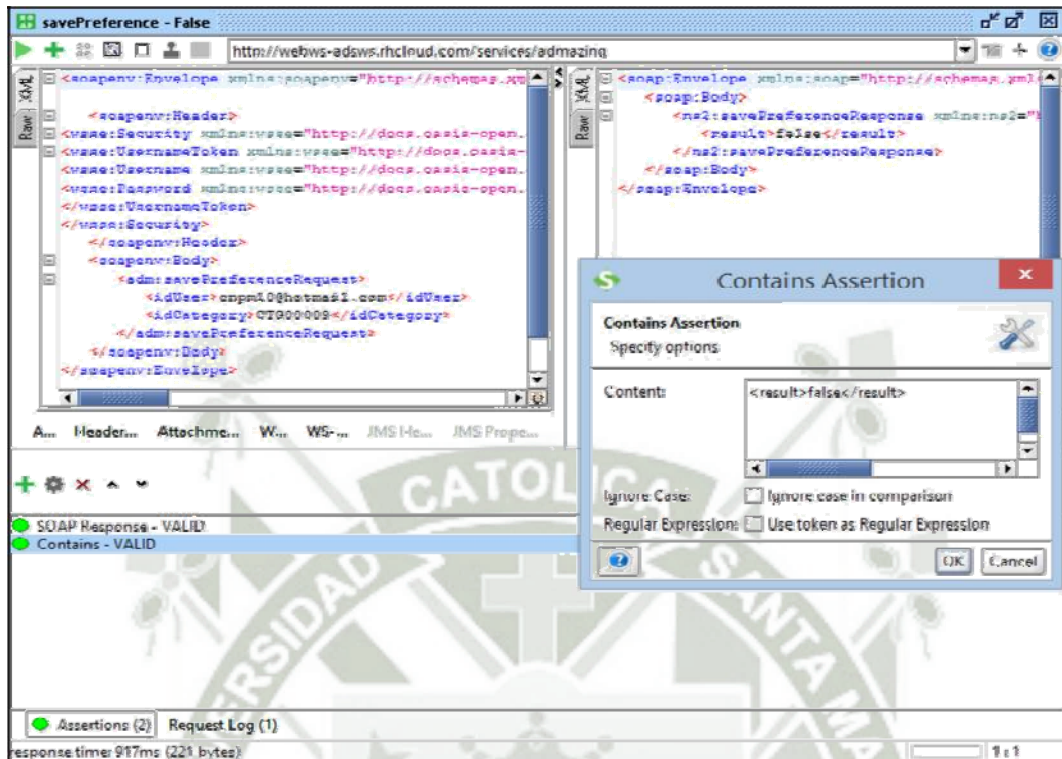


Figura 79: Prueba de guardar una categoría fallidamente
Fuente: Elaboración propia

A continuación se harán las pruebas para eliminar una preferencia de la categoría guardada anteriormente. En la siguiente imagen se observa que se eliminó exitosamente la categoría entre las preferencias del usuario cnpm18@hotmail.com, ya que el resultado es verdadero.

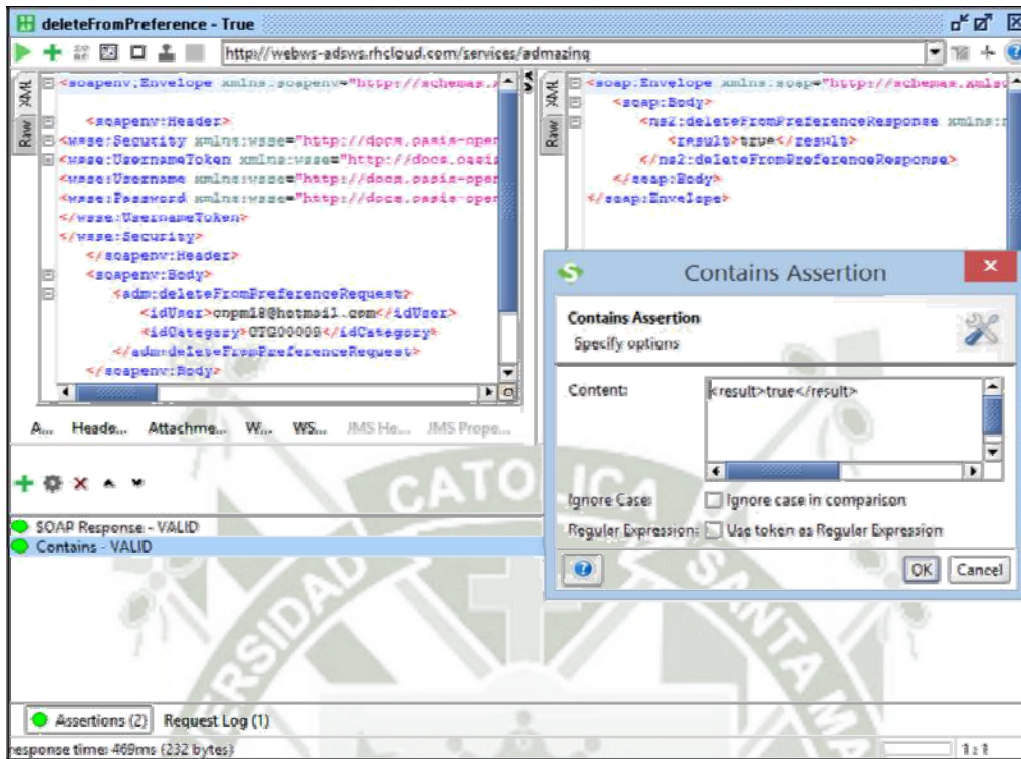


Figura 80: Prueba de eliminar una categoría exitosamente
Fuente: Elaboración propia

En la siguiente imagen se envía la misma solicitud con los mismos datos, sin embargo el resultado es falso porque la categoría que se desea eliminar ya no se encuentra en las preferencias.

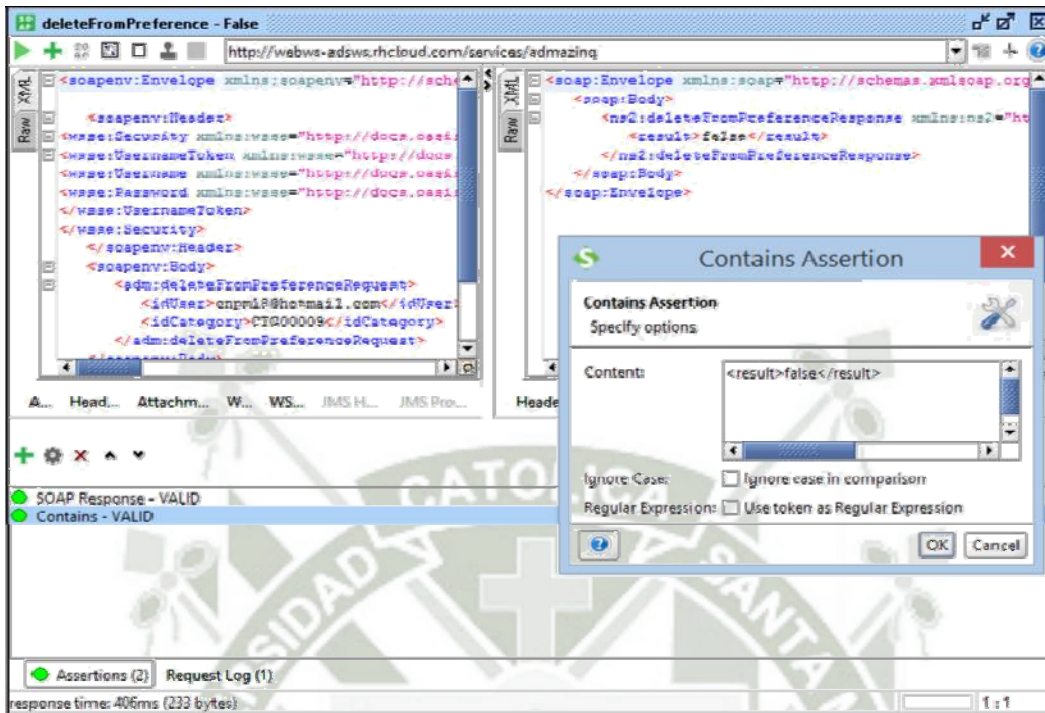


Figura 81: Prueba de eliminar una categoría fallidamente

Fuente: Elaboración propia

La novena prueba consta de cuatro solicitudes para la obtención de categorías para ello se envía el código de la tienda que el usuario seleccionó y recibe las categorías que pertenecen a esa tienda, para esta prueba se utilizará la tienda de código “EP000002”. Si el usuario desea recibir todas las promociones sin que se ajusten a sus preferencias, se llama a la función “getByIdCategory”. En la siguiente imagen se observa que se obtiene exitosamente las categorías.

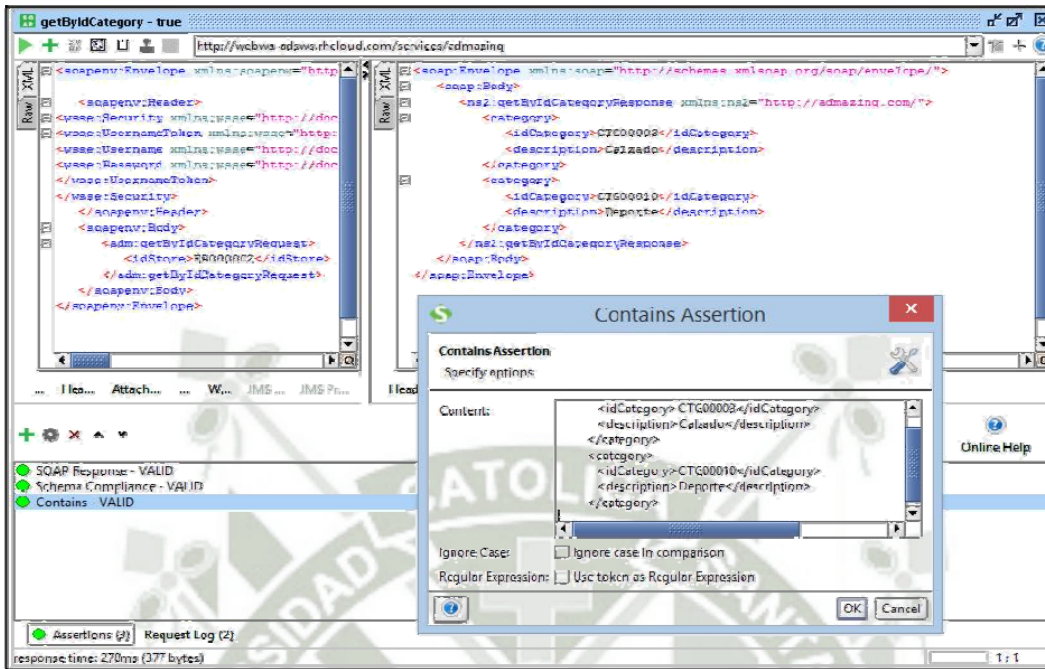


Figura 82: Prueba de obtener las categorías exitosamente
Fuente: Elaboración propia

En la siguiente imagen se envía la solicitud a la misma función pero a una tienda que no existe, por ello devuelve vacía la respuesta.

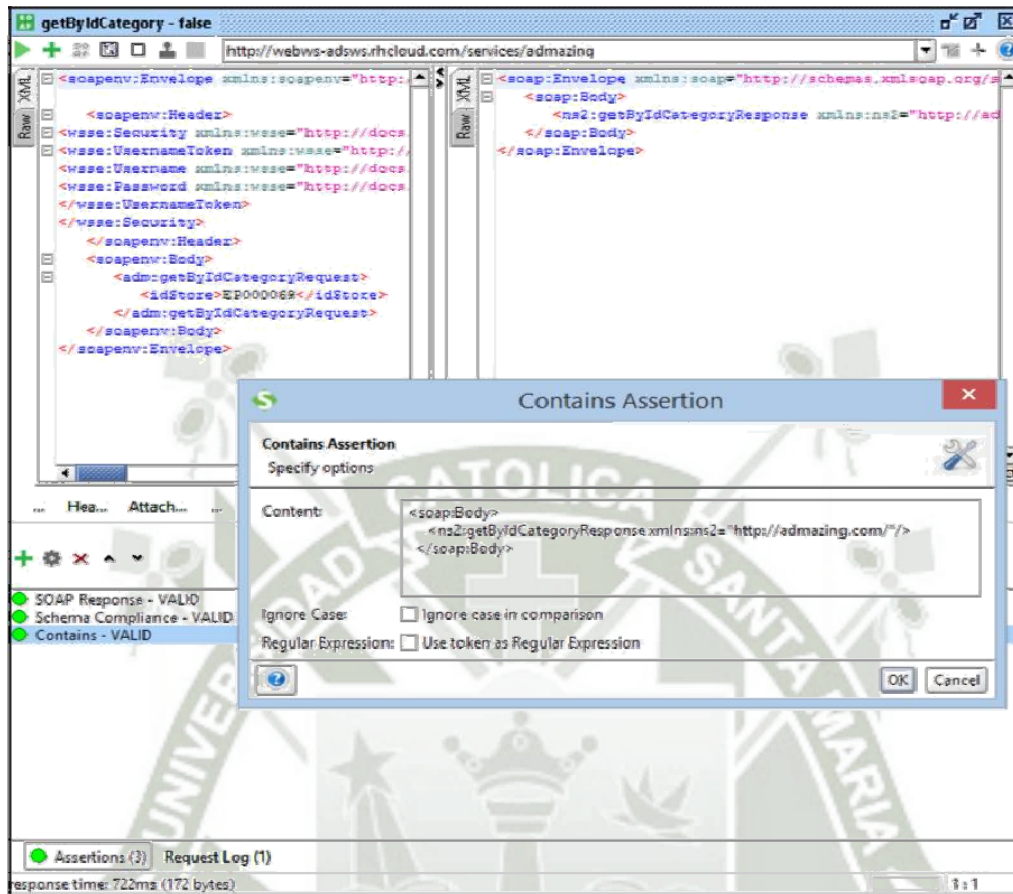


Figura 83: Prueba de obtener las categorías fallidamente

Fuente: Elaboración propia

En caso que el usuario desee que sus promociones se ajusten a sus preferencias se llama a la función “getCategoryByPreference”, para la tienda de código “EP000002” que se utilizó en la primera solicitud, además en la solicitud se debe enviar el código del usuario en este caso es “cnpm18@hotmail.com”. En la siguiente imagen se observa que se obtiene exitosamente las categorías, solo la categoría 3 porque la categoría 10 no está en las preferencias.

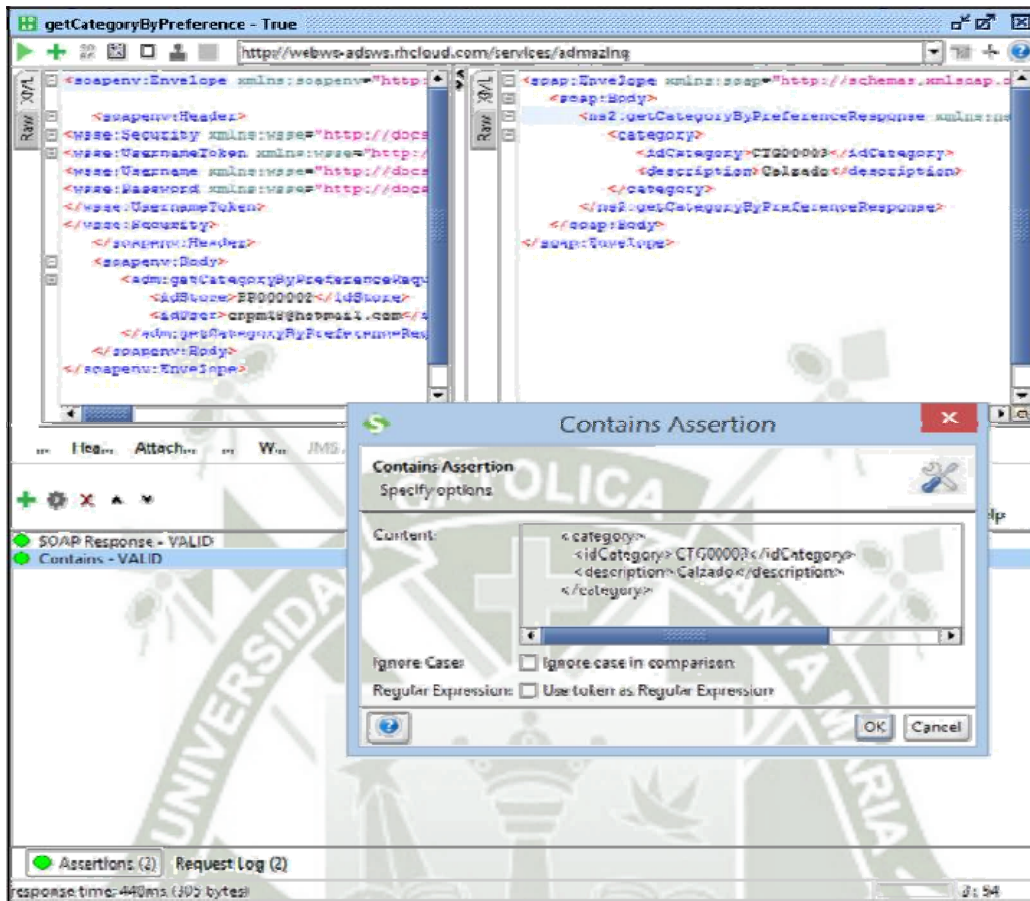


Figura 84: Prueba de obtener las categorías según preferencias exitosamente

Fuente: Elaboración propia

En la siguiente imagen se envía la solicitud a la misma función pero con un usuario que no tiene ninguna preferencia, por ello devuelve vacía la respuesta.

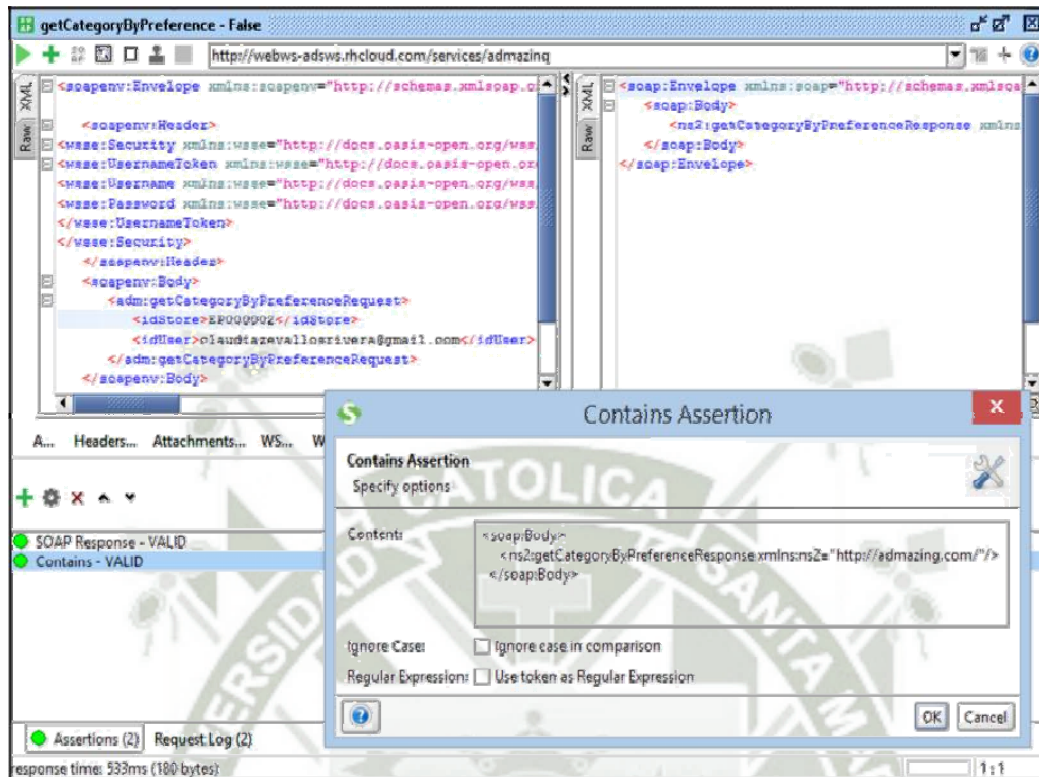


Figura 85: Prueba de obtener las categorías según preferencias fallidamente
Fuente: Elaboración propia

La última prueba consta de dos solicitudes para la obtención de promociones para ello se envía el código de la tienda y el código de la categoría que el usuario seleccionó y recibe las promociones que incluye la información del producto y el tipo de promoción, para esta prueba se utilizará la tienda de código “EP000004” y la categoría de código “CTG00009”. Para ello se llama a la función “getByIdPromotion”. En la siguiente imagen se observa que se obtiene exitosamente las promociones.

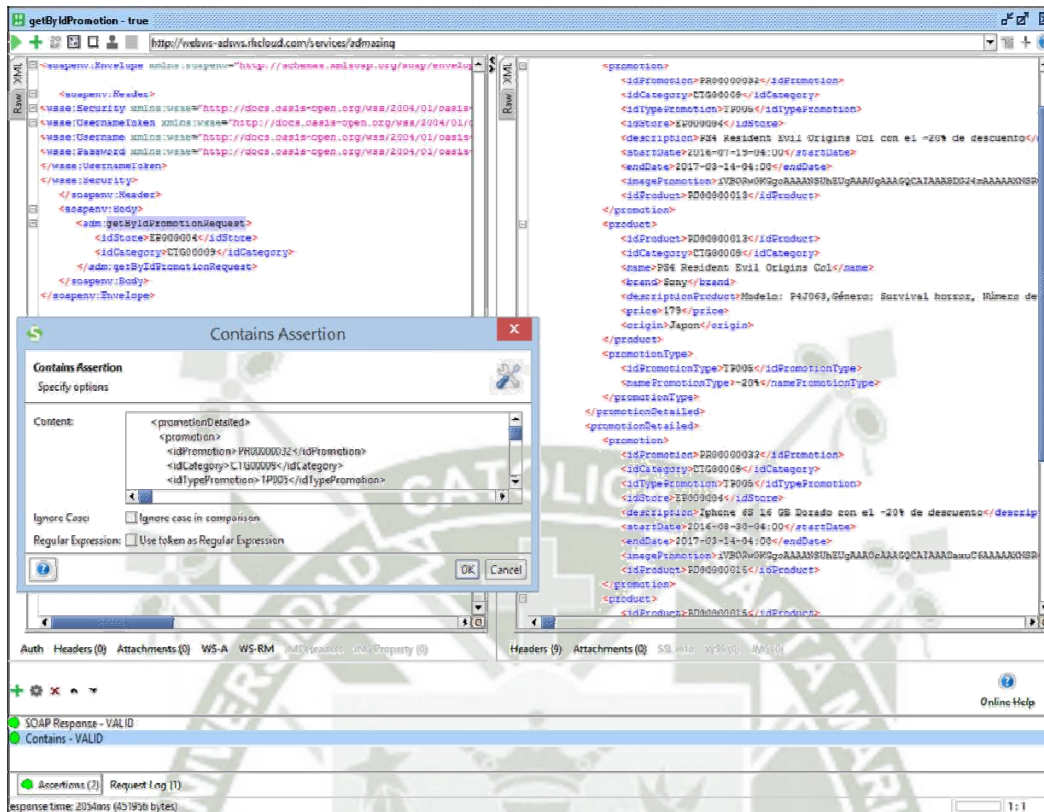


Figura 86: Prueba de obtener las categorías exitosamente

Fuente: Elaboración propia

En la siguiente imagen se envía la solicitud a la misma función pero a la tienda de código “EP000003” y la categoría de código “CTG00009”. En esta tienda no existe la categoría nueva, por ello devuelve vacía la respuesta.

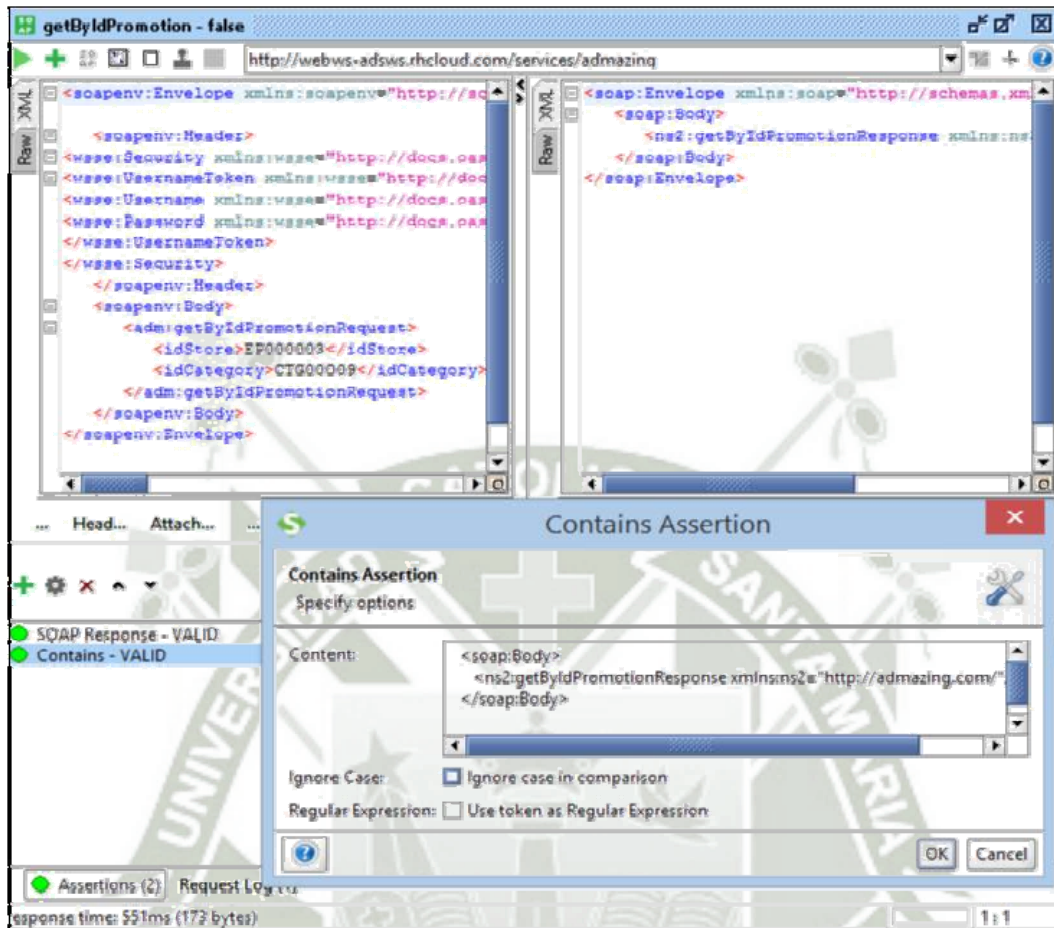


Figura 87: Prueba de obtener las categorías fallidamente
Fuente: Elaboración propia

2.5.1.2 Pruebas de rendimiento

En esta sección se prueba que el rendimiento de la aplicación móvil y la aplicación del Apple Watch en diferentes escenarios. Esta prueba consiste en la medición de uso de recursos del móvil y el wearable; tales como CPU, memoria y conectividad.

Se comienza haciendo la medición de rendimiento de la aplicación del wearable. A continuación se puede observar la utilización de la memoria del reloj en el escenario en que se usa más recursos del wearable, el cual se presenta cuando en la aplicación se está mostrando una promoción. Se puede observar que la aplicación consume 14.3 MB de memoria, lo cual corresponde un 0.17% del total de la memoria que provee este dispositivo, esto indica que esta aplicación wearable es muy ligera en cuanto a consumo de recursos de memoria del dispositivo que la alberga.

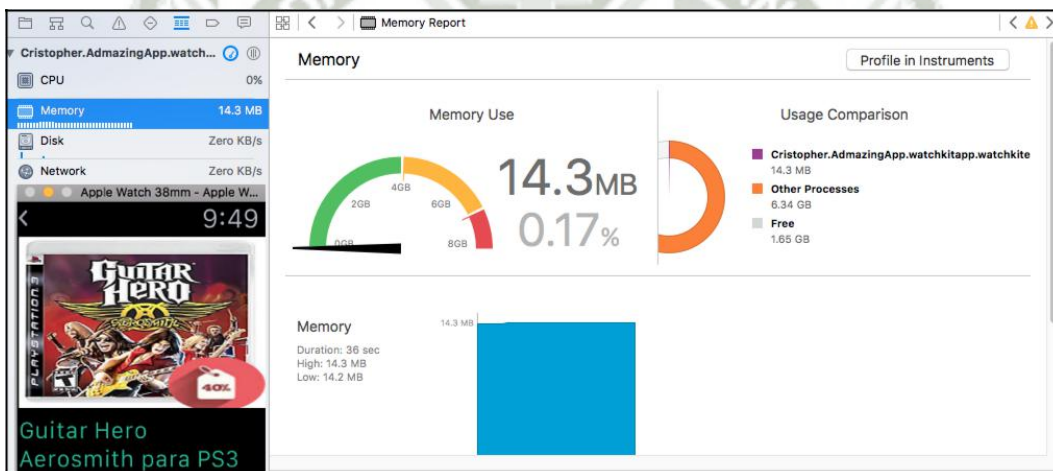


Figura 88: Prueba de uso de memoria RAM por parte de la aplicación wearable
Fuente: Elaboración propia

La siguiente medición se realiza con respecto al consumo de CPU por parte de la aplicación wearable, esta prueba en un escenario distinto que es la generación del código QR para el uso de una promoción, se puede observar en la siguiente imagen que el consumo del CPU es nulo en ese punto, con una duración de uso de

55 segundos como se puede apreciar que esta aplicación no realiza gasto de CPU lo cual nos indica que esta aplicación no genera una carga al CPU del dispositivo wearable con el que se esté trabajando.

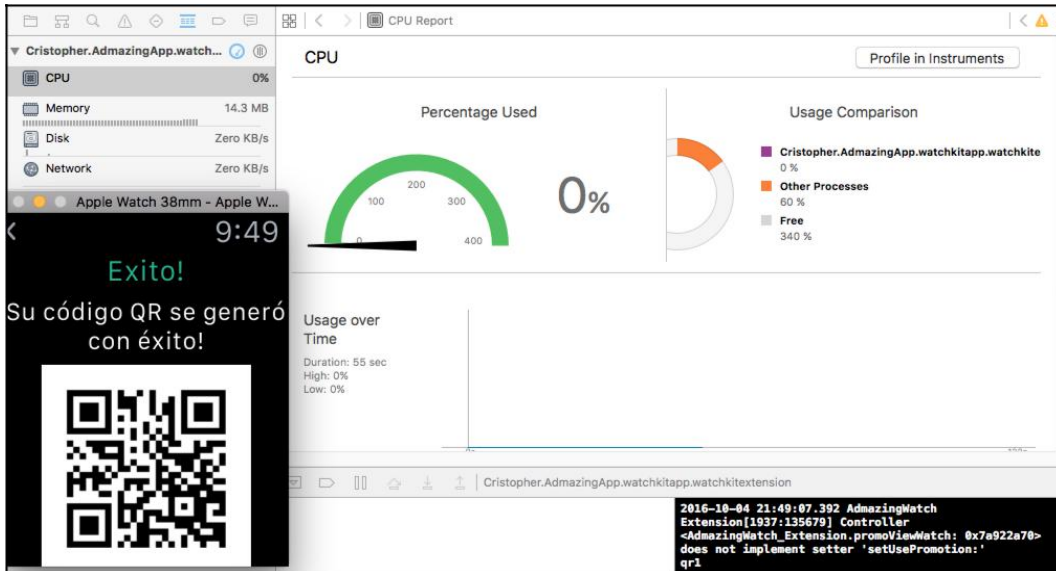


Figura 89: Prueba de uso de CPU por parte de la aplicación wearable
Fuente: Elaboración propia

En conclusión a la prueba de rendimiento de la aplicación wearable se puede indicar que esta no realiza uso de CPU ni conectividad, esta se presenta solo en casos en los que se soliciten datos desde el servidor. Adicionalmente se puede resaltar que el mayor uso de recursos de hardware se presenta con el uso de memoria, lo cual se indica que es reducido frente a otras tareas que se usan dentro del dispositivo. Con esto se puede concluir que la aplicación wearable tiene un rendimiento óptimo en cuanto a uso de recursos del dispositivo wearable con el que trabaja.



Figura 90: Prueba integral de recursos de hardware por parte de la aplicación wearable
Fuente: Elaboración propia

A continuación se realizan las pruebas de uso de recursos para la aplicación móvil, la primera prueba está basada en el uso de conexión para la extracción de datos desde el servicio web.

Como se puede ver a continuación esta prueba se realiza en escenarios distintos, el primero fue hecho cuando se reciben los datos de los establecimientos que se encuentran en la misma zona comercial que el usuario, además dichos datos contienen los logos de los establecimientos. Se puede observar que se está recibiendo 0.2 MB/s y se envían 1.3 KB/s, lo cual nos indica que en ese momento no es mucho el costo de conectividad que se tiene por la aplicación. Además se puede ver que en total hasta ese momento solo se recibieron 2.4 MB desde el servicio web.

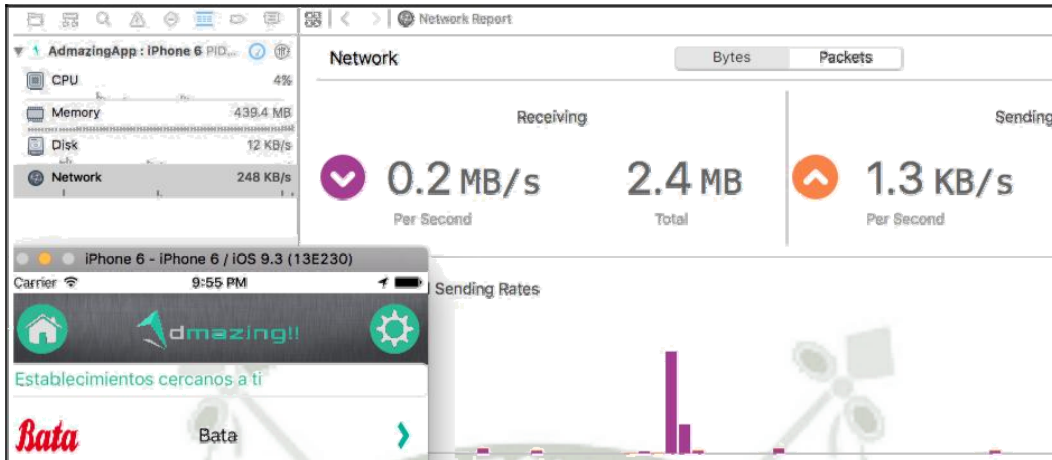


Figura 91: Primera prueba de uso de conectividad de la aplicación móvil.

Fuente: Elaboración propia

Posteriormente se realiza la prueba de recepción de las promociones de un determinado establecimiento y se necesita indicar su ubicación dentro del mapa, por lo que se necesita recibir los datos de su ubicación también, hasta este punto se puede observar que se recibieron 4.1 MB/s y se enviaron 39.5KB desde la aplicación, lo que nos indica que aún el costo de conectividad no es de gran significancia a pesar de todos los datos y la imágenes que se transmitieron desde la aplicación y se recibieron desde el servicio web .

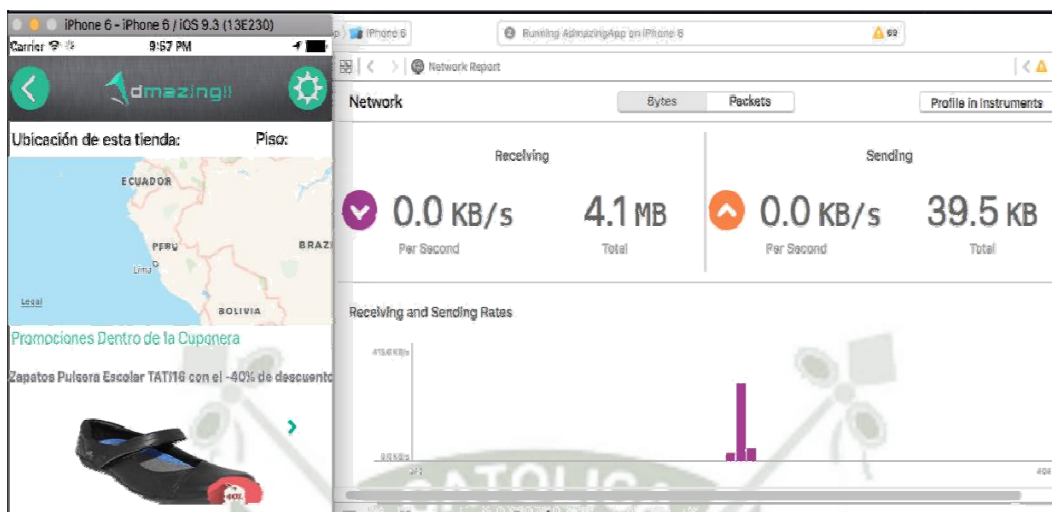


Figura 92: Segunda prueba de uso de conectividad de la aplicación móvil.

Fuente: Elaboración propia

Las pruebas anteriormente mostradas se presentan en un escenario donde se cuenta con 3.05Mbps en velocidad de descarga y 0.54 Mbps en velocidad de carga, el tiempo de respuesta desde el servicio web puede variar dependiendo de la velocidad de conexión con la que cuenta el usuario. A continuación se pueden ver las medidas del ancho de banda que se utilizaron al realizar las pruebas anteriores.



Figura 93: Medición del ancho de banda utilizado para las pruebas de conectividad
Fuente: Elaboración propia

La siguiente prueba realizada para evaluar el rendimiento de la aplicación es la evaluación del uso de CPU del dispositivo móvil.

Esta prueba también es realizada en diferentes escenarios, el primero se presenta cuando se está solicitando la ubicación del usuario, como se puede apreciar esta funcionalidad utiliza el 7% de la capacidad del CPU, lo cual comparado con 66% del uso de otro procesos que nos muestra el gráfico, es muy reducido para la tarea que se está realizando, por lo que se determina que en este punto el uso del CPU por la aplicación móvil es óptimo.

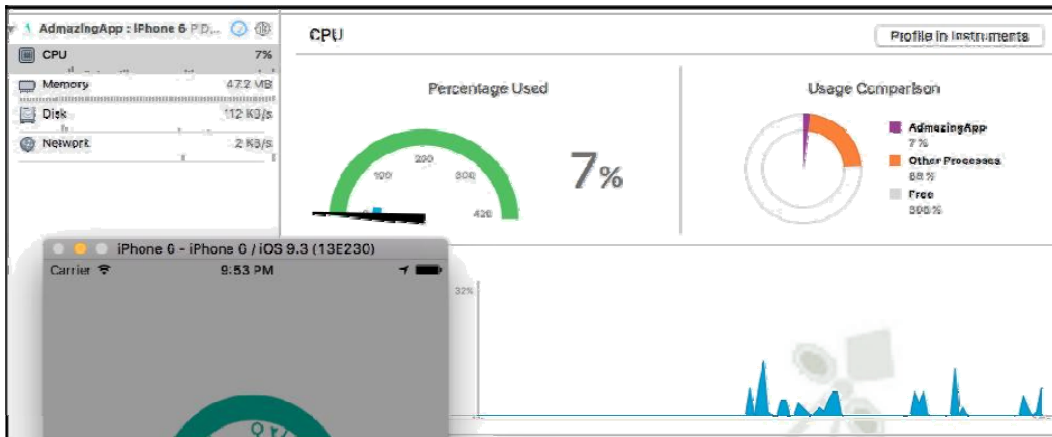


Figura 94: Primera prueba de uso de CPU de la aplicación móvil.
Fuente: Elaboración propia

El siguiente escenario se presenta cuando se muestra la ubicación de un establecimiento y de todas las promociones que este provee. Se puede observar que en esta situación el uso del CPU se redujo a 4% del uso total de este recurso. Esta disminución nos ayuda a validar que este software sigue siendo ligero en cuanto al uso de CPU del dispositivo que lo utiliza.

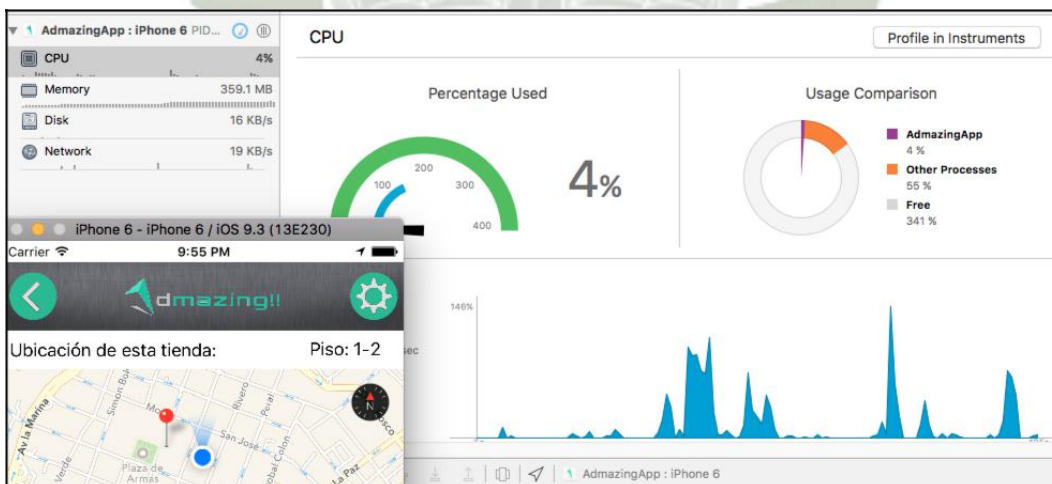


Figura 95: Segunda prueba de uso de CPU de la aplicación móvil.
Fuente: Elaboración propia

El tercer escenario se da dentro de la cuponera, cuando se actualiza la ubicación de los establecimientos que proveen las promociones guardadas en la cuponera del usuario, al hacer todas estas actualizaciones de ubicación dentro del mapa el uso del CPU presenta su punto crítico, mostrando un 35% del uso de este recurso, lo cual se encuentra dentro de los parámetros dados por Apple para el uso de este recurso. Esto indica que la aplicación del sistema de promociones conserva un buen rendimiento en cuanto al uso de CPU del dispositivo IOS con el que se está trabajando.

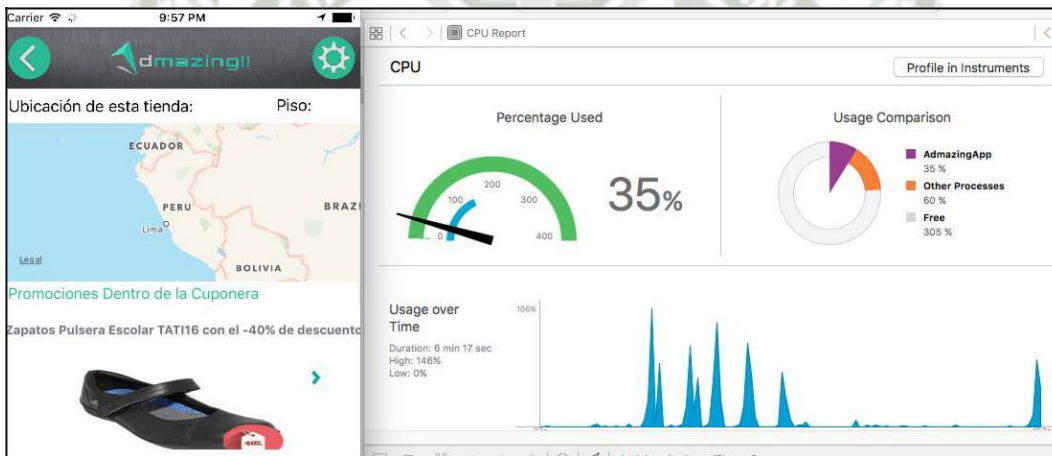


Figura 96: Tercera prueba de uso de CPU de la aplicación móvil.

Fuente: Elaboración propia

Por último se realiza la prueba de uso de memoria por parte de la aplicación, esta se realiza en cuatro escenarios diferentes. El primero, se da cuando se muestran todos los establecimientos de la zona comercial donde se encuentra el usuario. Se puede observar que en ese momento se utiliza 51.3 MB de memoria lo cual

corresponde a 0.63% de la memoria, esto nos asegura que el uso de este recurso hasta ese momento es el correcto para la aplicación.

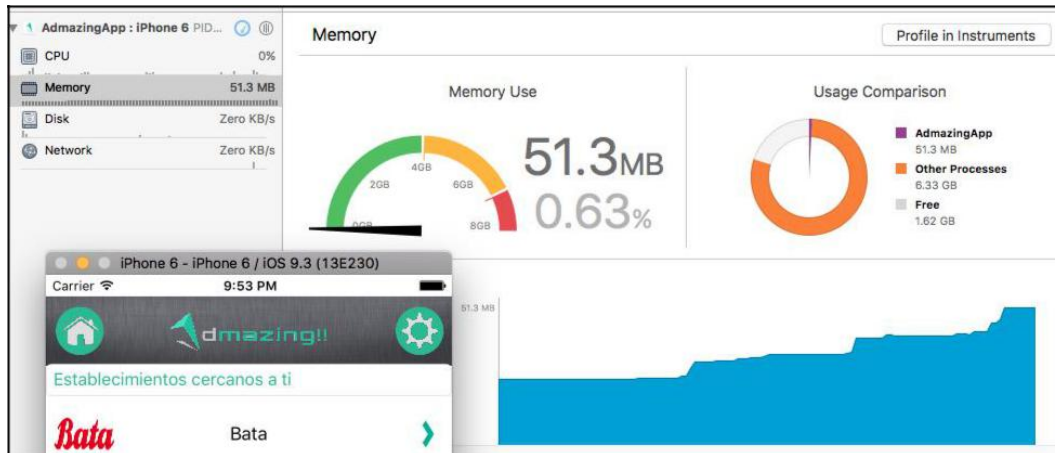


Figura 97: Primera prueba de uso de memoria de la aplicación móvil.

Fuente: Elaboración propia

El siguiente escenario que se evalúa es cuando se muestra la ubicación de la tienda que seleccionó el usuario y las categorías que ésta presenta, todos estos datos son recibidos desde el servicio web, por lo tanto el uso de la memoria por parte de la aplicación incrementa a 155MB lo cual equivale a 1.9% del uso de la memoria del dispositivo móvil que la contiene. Esta medida aún es positiva y confirma que en este proceso aún no se usa excesiva memoria lo cual ayuda a confirmar que el rendimiento de la aplicación en cuanto a uso de memoria es positivo.

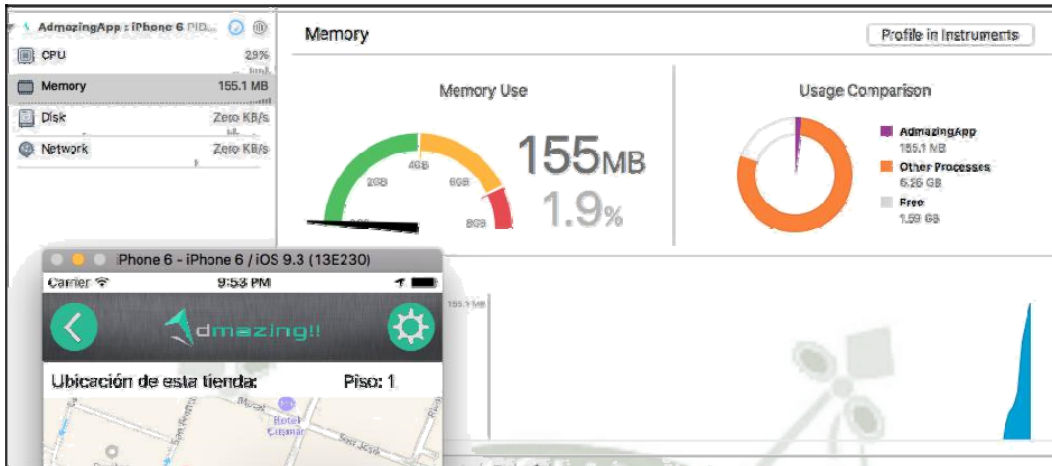


Figura 98: Segunda prueba de uso de memoria de la aplicación móvil.

Fuente: Elaboración propia

La tercera situación se presenta dentro de la cuponera, al momento de actualizar la posición del establecimiento seleccionado en el mapa, al cambiar diferentes ubicaciones rápidamente se puede observar que la aplicación usa mayor memoria, incrementando este uso a 319MB lo que equivale a 3.9% del uso de este recurso, esto aún se encuentra dentro un correcto uso de memoria por parte de la aplicación.

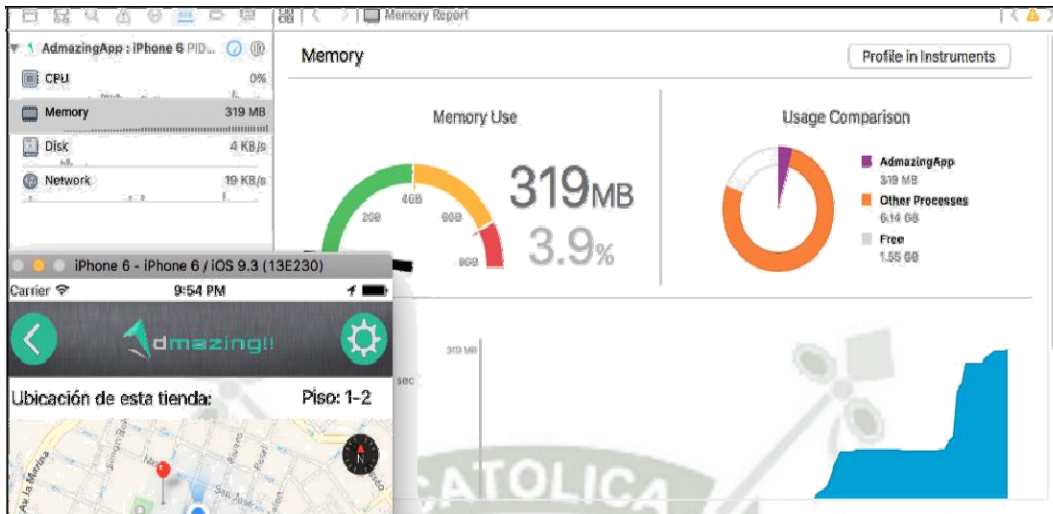


Figura 99: Tercera prueba de uso de memoria de la aplicación móvil.

Fuente: Elaboración propia

El último escenario corresponde al uso del servicio de Facebook® para poder compartir un cupón mediante esta red social, se puede observar que el uso de la memoria asciende a 671 MB lo cual corresponde al 8.2% de la memoria que provee el dispositivo móvil, lo cual es alarmante, sin embargo se encuentra aún dentro de los parámetros de uso de memoria que brinda Apple®, y este incremento se produce porque el software está utilizando un servicio externo a la aplicación móvil, esto nos ayuda a comprobar que pese a este incremento, la aplicación sigue teniendo un buen rendimiento en cuanto a uso de memoria.

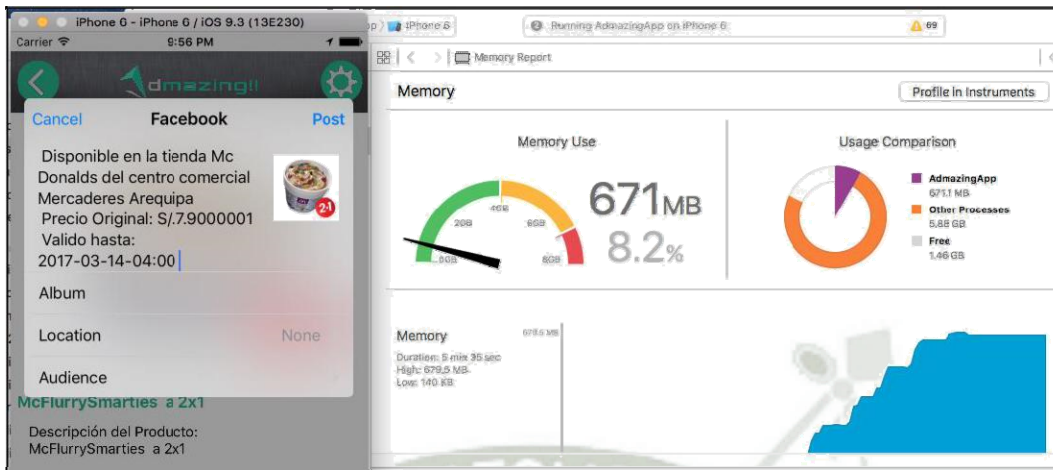


Figura 100: Cuarta prueba de uso de memoria de la aplicación móvil.

Fuente: Elaboración propia

En conclusión, se puede afirmar que el balance de la aplicación en cuanto al rendimiento y el uso de recursos de hardware que esta presenta es positivo, la aplicación móvil y la aplicación del wearable utilizan los recursos de hardware de manera medida pero eficiente, y todos estos datos están dentro de los parámetros que brinda XCode, el IDE de Apple® para desarrollo de aplicaciones en IOS y WatchOS.

Para medir el rendimiento dentro del Servicio Web para cada una de las funciones, se utilizó la misma herramienta “SoapUI”, esta se encarga de realizar múltiples solicitudes de las funciones, obtener el tiempo de respuesta y calcula el promedio de todos ellos. La velocidad con la que fue realiza esta prueba se indica en la siguiente imagen, esta fue extraída de la herramienta que brinda “Movistar”, que es el proveedor de internet.

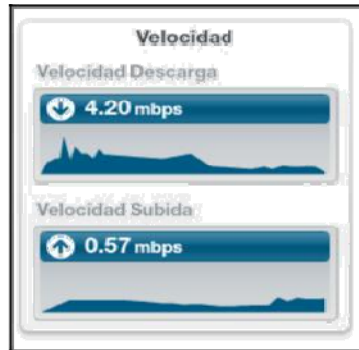


Figura 101: Medición de velocidad de internet
Fuente: Elaboración propia

En la siguiente imagen se observa que los promedios de todas las funciones son menores a los 0.2 segundos, excepto las siguientes funciones: “getAllStoresbyCommercialArea”, “getbyCommercialAreaCouponBook” y “getByIdPromotion” que tienen un promedio de 1.2 segundos, 0.4 segundos y 0.9 segundos respectivamente.

Test Step	min	max	avg	last	cnt	tps	bytes	bps	err	rat
login - True	145	531	192.22	145	9	0.13	1818	27	0	0
login - False	146	153	148.22	146	9	0.13	1827	28	0	0
saveAccess - True	151	156	152.66	152	9	0.13	2511	38	0	0
saveAccess - False	147	264	161.33	148	9	0.13	1917	29	0	0
getAllStoresbyCommercialArea	755	2747	1,240.11	755	9	0.13	4392684	67584	0	0
getbyCommercialAreaCouponBook	224	1707	400.77	226	9	0.13	639459	9838	0	0
saveCouponBook - true	150	312	172.22	312	9	0.13	1980	30	0	0
saveCouponBook - false	148	153	150.44	153	9	0.13	1999	30	0	0
getAllReferencCategories	147	154	148.22	148	9	0.13	18936	291	0	0
savePromotionUse - True	164	225	196.22	225	9	0.13	2016	31	0	0
savePromotionUse - False	152	191	163.33	156	9	0.13	2025	31	0	0
getAllReferencCategories 2	147	152	149.22	152	9	0.13	18929	291	0	0
saveCouponBook - true 2	150	438	184.11	150	9	0.13	1980	30	0	0
deleteFromCouponBook - True	148	154	150	148	9	0.13	2088	32	0	0
deleteFromCouponBook - False	146	154	148.22	149	9	0.13	2097	32	0	0
savePreference - True	148	268	167.33	148	9	0.13	1980	30	0	0
savePreference - False	146	264	172.22	147	9	0.13	1989	30	0	0
deleteFromPreference - True	148	259	168.88	150	9	0.13	2088	32	0	0
deleteFromPreference - False	146	261	160.22	147	9	0.13	2097	32	0	0
getByIdCategory - true	147	266	163	147	9	0.13	3393	52	0	0
getByIdCategory - false	146	255	177	147	9	0.13	1548	23	0	0
getCategoryByPreference - True	147	249	161.22	152	9	0.13	2745	42	0	0
getCategoryByPreference - False	147	297	179.55	148	9	0.13	1620	24	0	0
getByIdPromotion - true	590	1916	936.55	590	9	0.13	4067604	62583	0	0
getByIdPromotion - false	147	251	196.55	148	9	0.13	1557	23	0	0
deleteFromPreference - True 2	147	255	161.77	147	9	0.13	2090	32	0	0
Test Case:	4979	11962	6,365.66	5236	9	0.13	9180967	141256	0	0

Figura 102: Prueba de rendimiento con un hilo
Fuente: Elaboración propia

A continuación se muestra la misma prueba pero con diez hilos corriendo al mismo tiempo, en la imagen se observa que los promedios de todas las funciones son menores a los 0.4 segundos, excepto las siguientes funciones: “getAllStoresbyCommercialArea”, “getByCommercialAreaCouponBook”, y “getByIdPromotion” que tienen un promedio de 3.8 segundos, 0.76 segundos y 2.54 segundos respectivamente.

Test Step	min	max	avg	last	cnt	tps	bytes	bps	err	rat
login - True	156	719	335.83	276	36	0.54	7272	109	0	0
login - False	154	872	324.97	213	36	0.54	7308	109	0	0
saveAccess - True	164	1295	387.94	227	36	0.54	10644	151	0	0
saveAccess - False	151	980	305.69	830	36	0.54	7668	115	0	0
getAllStoresbyCommercialArea	1109	20632	3.823,44	1487	36	0.54	17570736	264416	0	0
getByCommercialAreaCouponBook	254	2924	787.5	556	36	0.54	2557836	38492	0	0
saveCouponBook - true	147	1186	356.83	173	36	0.54	7936	119	0	0
saveCouponBook - false	148	1502	254.22	175	36	0.54	7936	119	0	0
getAllPreferencesCategories	150	748	270.77	596	36	0.54	75708	1139	0	0
savePromotionUse - True	155	2203	402.33	177	36	0.54	8081	121	0	0
savePromotionUse - False	157	1644	348.69	525	36	0.54	8091	121	0	0
getByIdPreferencesCategories 2	148	3903	385.16	206	36	0.54	75702	1139	0	0
saveCouponBook - true 2	148	1607	344.19	235	36	0.54	7936	119	0	0
deleteFromCouponBook - True	149	2591	394.16	1814	36	0.54	8367	125	0	0
deleteFromCouponBook - False	148	1756	292.66	175	36	0.54	8379	126	0	0
savePreference - True	148	2700	359.63	180	36	0.54	7934	119	0	0
savePreference - False	151	428	222.86	180	36	0.54	7946	119	0	0
deleteFromPreference - True	149	1151	269.13	205	36	0.54	8361	125	0	0
deleteFromPreference - False	148	1119	321.36	224	36	0.54	8383	126	0	0
getByIdCategory - true	150	750	258.58	206	36	0.54	13572	204	0	0
getByIdCategory - false	147	1681	307.89	542	36	0.54	6192	83	0	0
getCategoryByPreference - True	148	893	255.58	184	36	0.54	10980	165	0	0
getCategoryByPreference - False	149	1220	233.63	199	36	0.54	6480	97	0	0
getByIdPromotion - true	1192	8137	2.538,81	1857	36	0.54	16279416	248480	0	0
getByIdPromotion - false	152	1123	237.33	350	36	0.54	6228	93	0	0
deleteFromPreference - True 2	150	7027	498.36	7027	36	0.54	8377	126	0	0
Test Case:	6522	68839	14.516,75	18817	36	0.54	36723871	552845	0	0

Figura 103: Prueba de rendimiento con diez hilos
Fuente: Elaboración propia

Según Nielsen, J. (1993) un segundo es el tiempo límite para que los usuarios no crean que existe un retraso en la respuesta del sistema, y diez segundos es el tiempo máximo de espera para que el usuario siga enfocado en la actividad. Por estas razones se puede concluir que el servicio web tiene generalmente un buen rendimiento comparando los promedios de respuesta según los hilos, ya que la mayoría de funciones demora menos de 0.4 segundos. Las funciones que envían imágenes son las que demoran más, ya que estas imágenes son pesadas y toman

más tiempo de envío, estas funciones demoran como máximo un promedio de 3.8 segundos y por ende no sobrepasan el límite de atención del usuario.

2.5.1.3 Pruebas de seguridad

El servicio web verifica que los mensajes que se reciben son emitidos desde una fuente confiable, en este caso desde la aplicación móvil. Esto se asegura agregando en la cabecera de la solicitud los datos del usuario y su respectiva contraseña. Para probar su correcto funcionamiento, se realizan tres diferentes pruebas. En la siguiente imagen, se observa la respuesta de la primera prueba que muestra cuando la solicitud no tiene la información de seguridad en la cabecera.

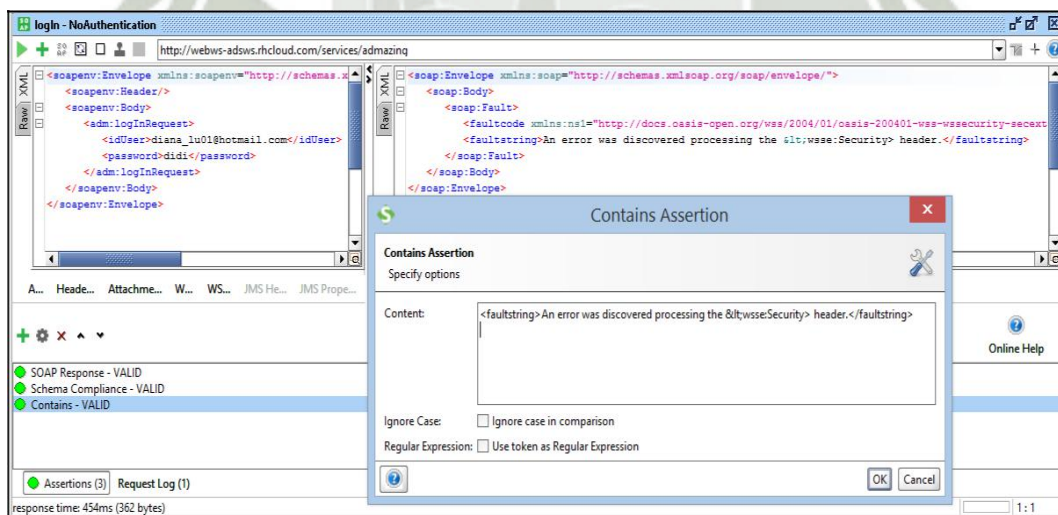


Figura 104: Primera prueba de seguridad fallida
Fuente: Elaboración propia

En la siguiente imagen, se observa la respuesta en caso que la solicitud tenga la información de seguridad en la cabecera, pero el código del usuario y/o de la contraseña no son correctos, como se observa el sistema responde que el token de seguridad no fue autorizado.

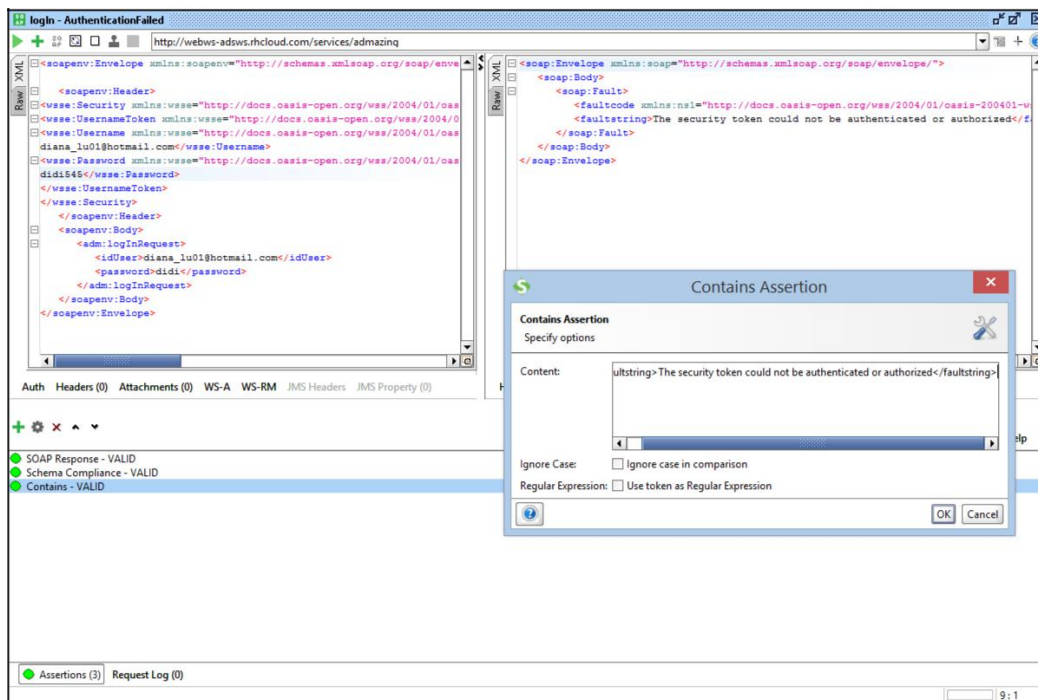


Figura 105: Segunda prueba de seguridad fallida
Fuente: Elaboración propia

En la siguiente imagen se observa la respuesta en caso que la solicitud tenga la información de seguridad en la cabecera y tanto el código de usuario como la contraseña son correctos. Esta función sigue su camino usual que es la función de inicio de sesión y devuelve verdadero al proceso de inicio de sesión.

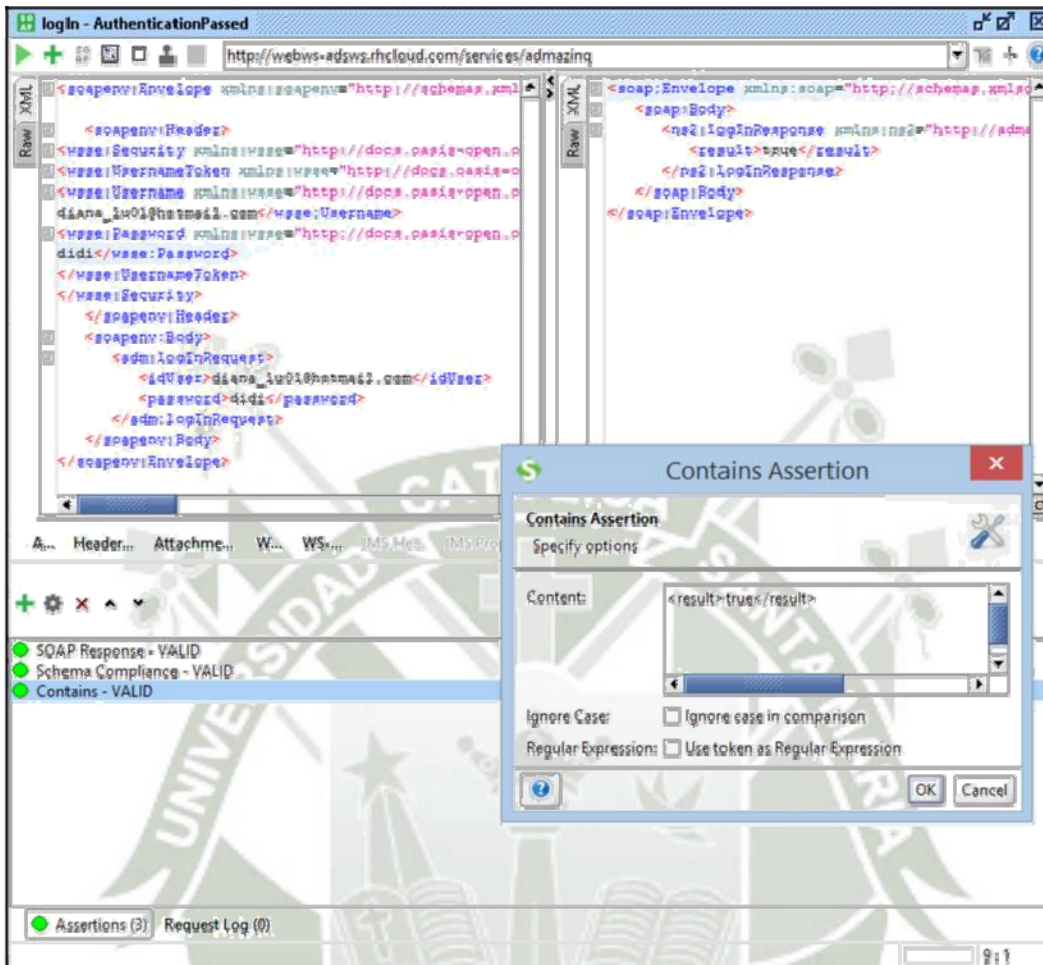


Figura 106: Prueba de seguridad exitosa
Fuente: Elaboración propia

2.5.2 Pruebas de validación

Para realizar las pruebas de validación de los requerimientos que se desarrollaron en este proyecto se realizó una encuesta a personas que previamente utilizaron la aplicación y pudieron probar todas sus funcionalidades. Con esta prueba se buscó validar el sistema de publicidad que se está trabajando, se

realizaron preguntas al usuario sobre requerimientos no funcionales que se buscan cubrir, entre ellos seguridad, tiempo de respuesta, eficiencia, buen rendimiento, facilidad de uso, entre otras. Para esta encuesta la muestra que se toma es de 71 personas escogidas de forma no probabilística, entre ellos personas cercanas al equipo de desarrollo de la aplicación como familiares, amigos y compañeros de trabajo.

La primera pregunta se realizó para conocer las edades de las personas encuestadas, se establecieron 6 rangos de edad según el poder consumo que se cuenta en cada rango de edad. Se puede observar que el 40.8% se encuentra entre 19 y 25 años, el 16.9% de las personas encuestadas se encuentran entre 25 y 30 años, por otra parte el 15.5% se encuentra entre 38 y 45 años, otro 14.1% de los encuestados tienen de 45 años a más, 9.9% tiene entre 31 y 37 años y por último 2.8% de los encuestados tiene entre 15 y 18 años. Con los presentes datos se busca tener variedad de opiniones sobre la aplicación, con las diferencias de edad en la muestra se podrán lograr diferentes tipos de opiniones.

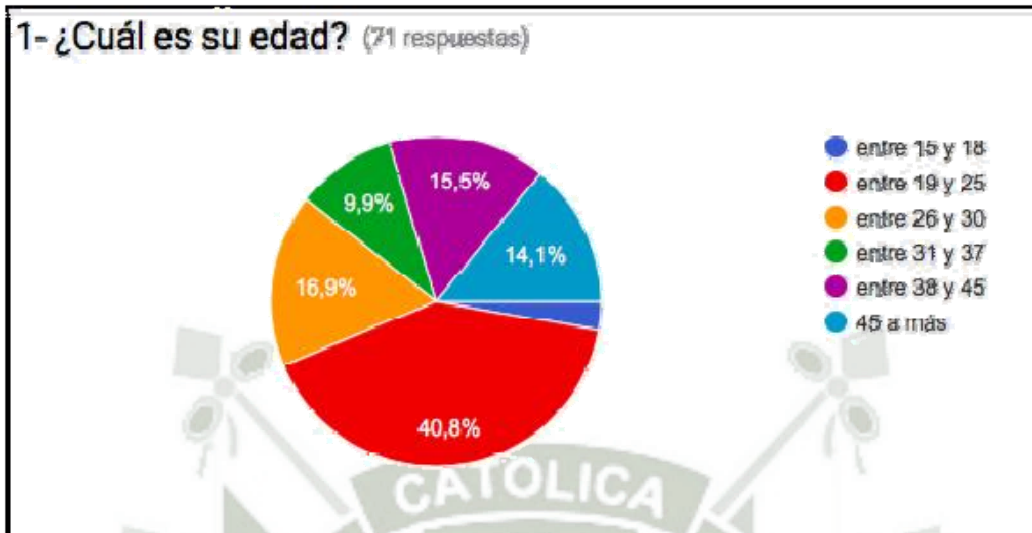


Figura 107: Resultados de la primera pregunta de la encuesta de validación del sistema
Fuente: Elaboración propia

La siguiente pregunta se realizó para poder segmentar uno de los requerimientos más importantes en los usuarios de la aplicación, el cual es tener un teléfono con internet móvil, ya que, la aplicación trabaja casi en su totalidad enviando y recibiendo datos al servicio web. Para esto el 82.9% de las personas encuestadas indicaron que si cuentan con este servicio. Con estos datos se pudo comprobar que más del 80% de los usuarios cuentan con este requerimiento primordial para el uso de la aplicación, lo cual es óptimo.

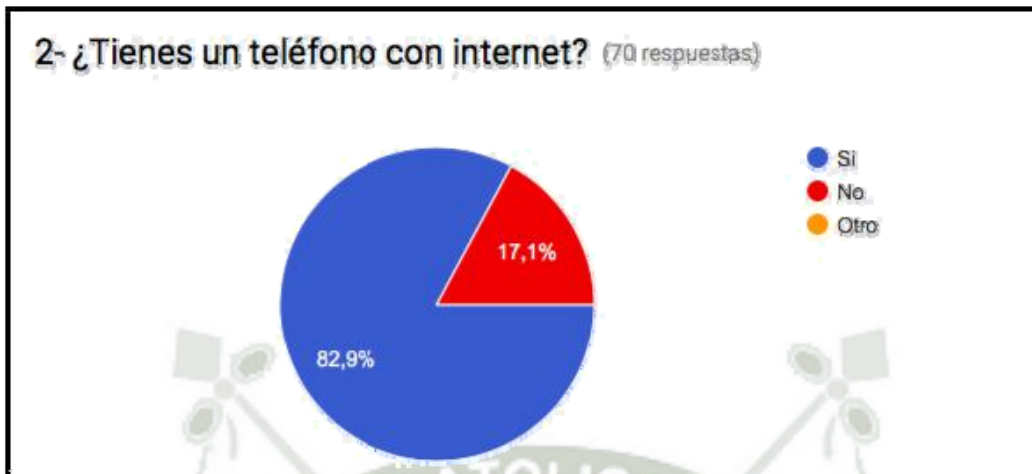


Figura 108: Resultados de la segunda pregunta de la encuesta de validación del sistema
Fuente: Elaboración propia

La tercera pregunta se realiza para comprobar un requerimiento no funcional, el cual es la facilidad de uso de la aplicación, los resultados indican que el 60.6% de los encuestados piensan que la aplicación es relativamente fácil de utilizar y 33.8% de los encuestados indicaron que la aplicación es muy fácil de utilizar; mientras que 4.2% de los encuestados se mostraron indiferentes a esta pregunta y el restante 1.4% de los encuestados indicaron que el uso de la aplicación era relativamente difícil. Estos resultados ayudan a comprobar que se logró implementar uno de los requerimientos no funcionales más importantes, el cual es la facilidad de uso.

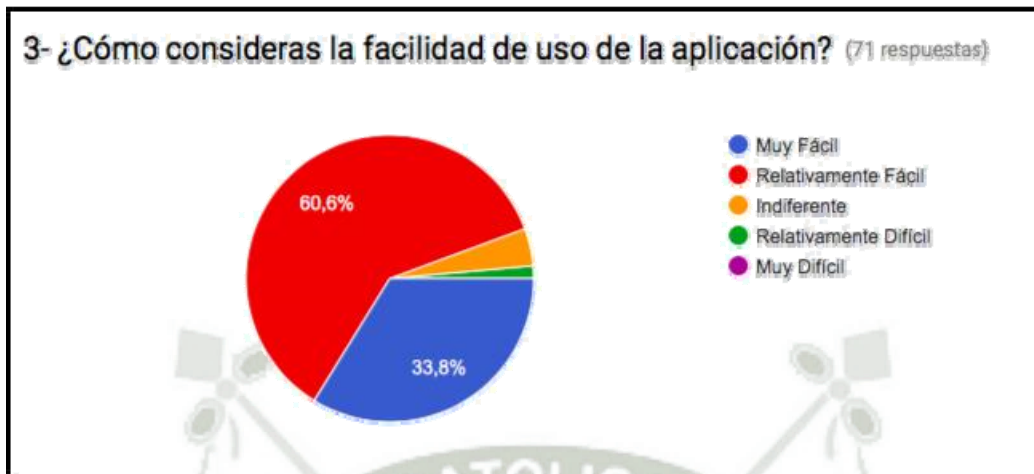


Figura 109: Resultados de la tercera pregunta de la encuesta de validación del sistema
Fuente: Elaboración propia

La siguiente pregunta se realizó para poder garantizar otro requerimiento no funcional, el cual es la tolerancia a fallos, para esto se le preguntó a los usuarios si tuvieron fallos durante el uso de la aplicación, a esto el 94.4% de los encuestados respondió que no tuvieron ningún fallo, mientras que el otro 5.6% de las personas encuestadas indicaron que sí tuvieron un fallo durante el uso de la aplicación. Con estos resultados se puede garantizar que la tolerancia a fallos también fue cubierta para la mayoría de usuarios y se desean conocer los fallos que tuvieron las personas que respondieron de manera afirmativa a esta pregunta.

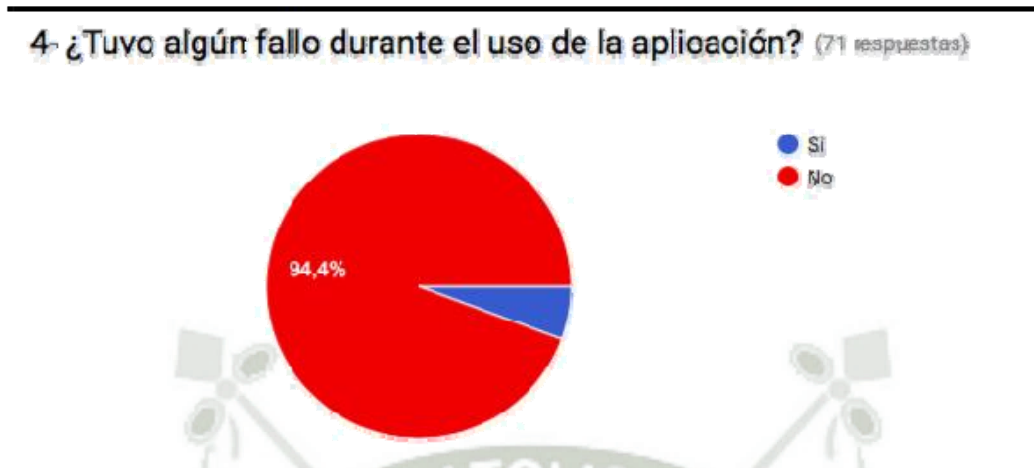


Figura 110: Resultados de la cuarta pregunta de la encuesta de validación del sistema

Fuente: Elaboración propia

La siguiente pregunta fue respondida por el 5.6% de las personas que indicaron un fallo dentro de la pregunta anterior, dentro de las respuestas se puede observar que el 50% se refiere a la calidad de la señal, para lo que se indica que la calidad del servicio de la aplicación es directamente proporcional a la calidad de la señal de internet, debido a que en su mayoría se usa conexiones de envío y recepción de datos con el servicio web. El otro 50% de las personas que respondieron esta pregunta indica el cierre de la aplicación y problemas con el flujo de la aplicación, para esto se desarrolló un sistema de alertas en todos los escenarios de la aplicación para que no se detenga la ejecución del sistema y trabaje de manera óptima.

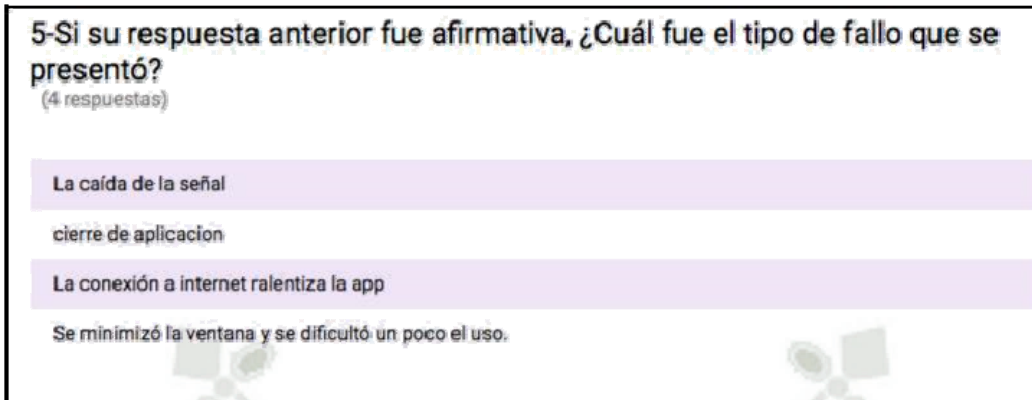


Figura 111: Resultados de la quinta pregunta de la encuesta de validación del sistema
Fuente: Elaboración propia

La siguiente pregunta es para poder garantizar la correcta señalización de escenarios dentro de la aplicación para la facilidad de uso por parte del usuario y la continuidad en el flujo de la aplicación. Los resultados indican que el 66.2% de los encuestados piensan que se muestran apropiados mensajes de error y alertas, otro 22.5% de los encuestados piensan que en su totalidad, la aplicación muestra apropiada señalización para los usuario, 9.9% de las personas encuestadas se muestran indiferentes a esta pregunta, y el restante 1.4% de los encuestados indicaron que no están de acuerdo a los mensajes de alerta y error que presenta la aplicación.

Se infiere que se implementaron los mensajes en la aplicación de manera positiva para la calidad de uso del usuario ya que los resultados indican que más del 80% de los usuarios encuestados piensan que se tiene buena señalización de errores y alertas en la aplicación frente a 1.4% que piensan lo contrario.



Figura 112: Resultados de la sexta pregunta de la encuesta de validación del sistema

Fuente: Elaboración propia

Con la siguiente pregunta se desea conocer si los usuarios encuestados creen que sus datos personales están seguros frente a amenazas externas, se puede conocer que se tienen mecanismos de seguridad como contraseñas y autenticación de mensajes con el servidor lo cual se explicó a los encuestados. Las respuestas indican que el 60.6% de los encuestados piensan que sus datos personales están seguros en esta aplicación, 19.7% de las personas encuestadas están completamente seguras que los datos que tiene la aplicación están seguros, 18.3% de los encuestados se muestran indiferentes a esta pregunta, mientras que el 1.4% de los usuarios encuestados piensan que sus datos no están seguros dentro de esta aplicación. Los resultados anteriores permiten afirmar que la seguridad de datos no es un problema en esta aplicación y que los usuarios saben que sus datos se encuentran seguros, gracias a los mecanismos de seguridad implementados, esto ayuda a poder garantizar el requerimiento de seguridad para el sistema.

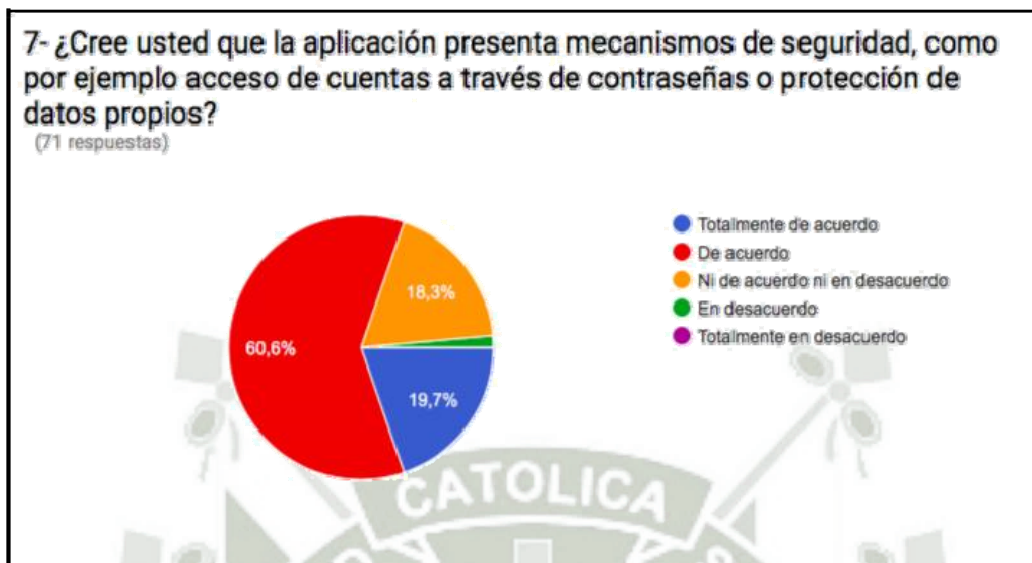


Figura 113: Resultados de la séptima pregunta de la encuesta de validación del sistema
Fuente: Elaboración propia

La siguiente pregunta de la encuesta indaga si los usuarios creen que la aplicación cuenta con las funciones necesarias para el servicio que se desea ofrecer, para esto el 57.7% piensa que faltan implementar algunas funciones, mientras que 40.8% piensa que se tienen todas las funciones implementadas, y 1.4% de los usuarios encuestados piensa que faltan muchas funciones para implementar.

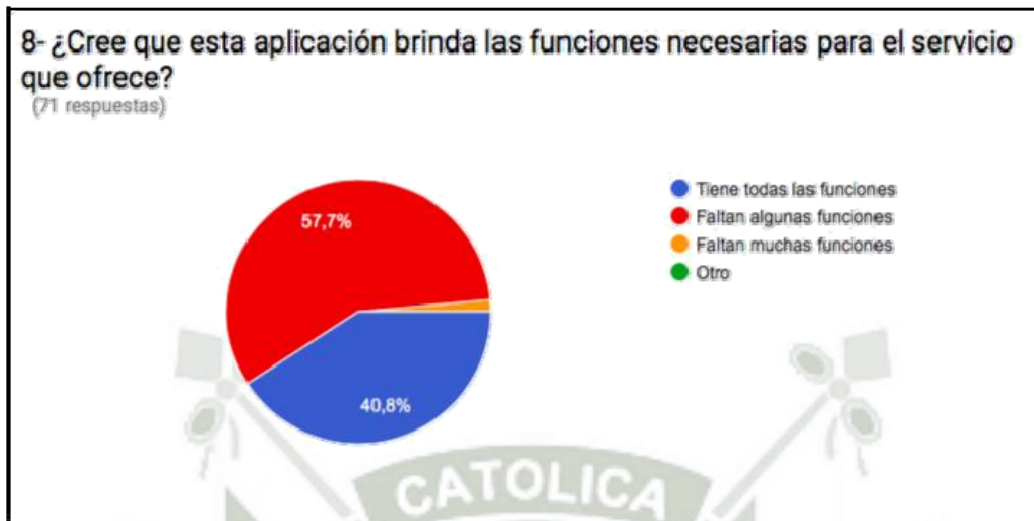


Figura 114: Resultados de la octava pregunta de la encuesta de validación del sistema
Fuente: Elaboración propia

Con la siguiente pregunta se desea conocer qué otras funciones los usuarios desearían que la aplicación brinde, los resultados indican que 54.9% desean que el sistema tenga la capacidad de emitir alarmas cuando una promoción está cerca de su salida de vigencia, 38% desea que el sistema pueda determinar por geolocalización a los amigos del usuario que se encuentren dentro de la misma zona comercial que ellos, el 31% de los encuestados desea que se brinden beneficios a las personas que usen el sistema con frecuencia, 26.8% de los usuarios desean que exista una manera de comunicación con los amigos que se encuentren en la misma zona comercial, mientras que 16.9% de los encuestados desea que la aplicación emita un balance económico de las compras del usuario mensualmente. Todas estas funcionalidades indicadas por el usuario dan un panorama de lo que estos desean

para la aplicación en el futuro y estas funciones se desarrollarán posteriormente por el equipo de desarrollo de la aplicación.

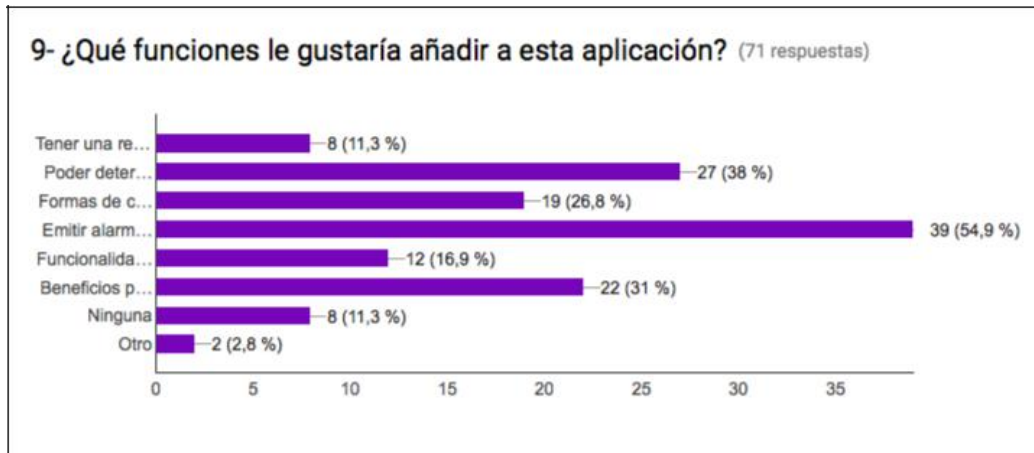


Figura 115: Resultados de la novena pregunta de la encuesta de validación del sistema
Fuente: Elaboración propia

Por último se desea que las personas que usaron la aplicación puedan brindar un balance global de la aplicación, las respuestas están en una escala del 1 al 5, siendo 1 Pésima a 5 que es Excelente. Los resultados indican que el 63.4% de los usuarios indican que la aplicación es buena, el 29.6% de los encuestados indica que la aplicación es excelente y cumple con sus objetivos de manera óptima y el 7% restante piensan que la aplicación es normal, presentando buenas características, pero sin dar una calificación negativa. Los resultados indican que los usuarios dieron una buena calificación a la aplicación e indicaron que esta cumple con sus objetivos principales, gracias a las funciones que brinda y la

capacidad de gestión de promociones y usuarios con la que se encuentra gracias a la calidad de servicio que se desea brindar.

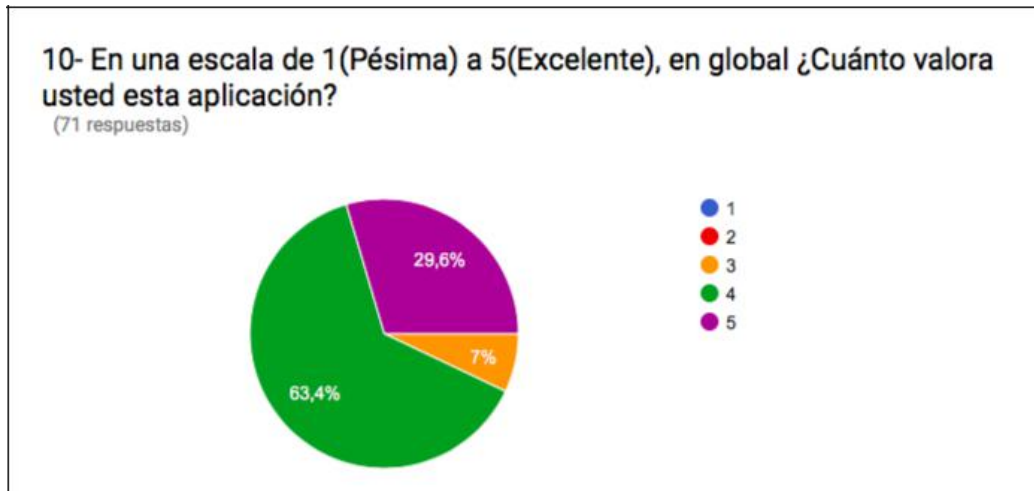


Figura 116: Resultados de la décima pregunta de la encuesta de validación del sistema
Fuente: Elaboración propia

2.6 Manuales de usuario

Para poder utilizar esta aplicación se necesita que el usuario cuente con un teléfono móvil con sistema operativo IOS 9 en adelante, ya que se necesitan las funciones de geolocalización y facilidad de conexión con el servicio web. Es muy importante que el usuario cuente con una señal de internet móvil cuando use la aplicación, debido a que la aplicación se conecta constantemente con el servicio web para obtener los datos que se muestran. En cuanto al wearable, el usuario debe contar con un AppleWatch con sistema operativo WatchOS 2 en adelante, para

poder utilizar con precisión las funciones que brinda el sistema para este dispositivo.

Por el momento esta aplicación solo es para los dos sistemas operativos indicados previamente, IOS en el caso de dispositivos celulares y WatchOS para los relojes inteligentes, aún no se encuentra disponible para celulares con sistema operativo Android ni relojes con el sistema Android Wear.

La primera vista que presenta esta aplicación corresponde a la pantalla de registro, en la cual el usuario debe ingresar su nombre de usuario y contraseña, después de ingresar los datos, el usuario debe presionar el botón “LOGIN” para que estos datos sean autenticados por el sistema y le brinde el acceso.

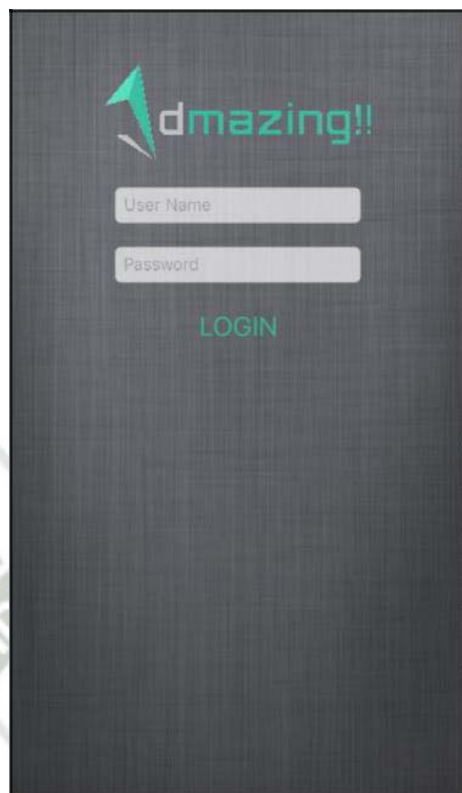


Figura 117: Actividad de acceso al sistema dentro de la aplicación móvil

Fuente: Elaboración propia

Si en la anterior vista el usuario ingresa datos incorrectos la aplicación muestra una alerta que indica que los datos ingresados por el usuario son incorrectos y no pueden ser autenticados por el servicio web, para esto el usuario deberá presionar “OK” dentro del mensaje y volver a ingresar sus datos.



Figura 118: Mensaje de datos erróneos para ingresar al sistema
Fuente: Elaboración propia

Después de autenticar al usuario, el sistema necesita conocer su ubicación, para esto se muestra la siguiente vista, en la que el usuario debe presionar el botón de establecer ubicación, después de eso el sistema pedirá la ubicación del usuario, la cual será extraída gracias a la función de GPS del móvil.



Figura 119: Actividad de búsqueda de ubicación geográfica del usuario dentro de la aplicación móvil

Fuente: Elaboración propia

Si no se puede acceder a la ubicación del usuario se mostrará la siguiente alerta en la que se indica al usuario que no se puede encontrar su señal de GPS, este problema se da en el caso que la función de GPS del móvil está desactivada o si no se cuenta con internet móvil dentro del dispositivo.



Figura 120: Alerta cuando no se puede encontrar la posición geográfica del usuario

Fuente: Elaboración propia

Si la ubicación del usuario se pudo extraer de manera satisfactoria, se mostrará un mensaje en el que indica la latitud y longitud del usuario y la zona comercial cercana a este, se mostrará el mensaje de confirmación de la siguiente manera.

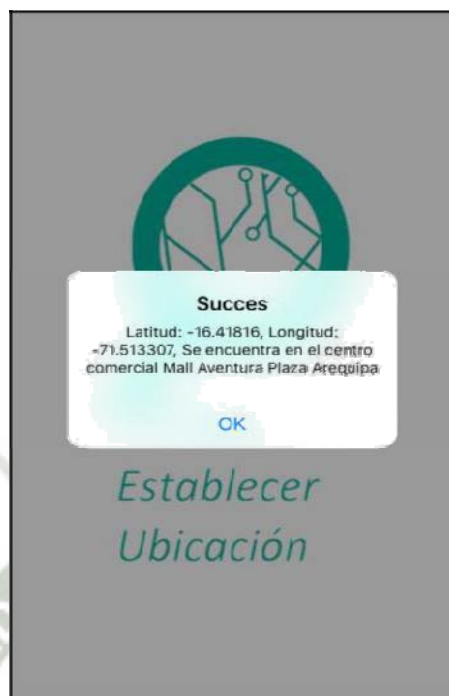


Figura 121: Confirmación de que se encontró la ubicación geográfica del usuario
Fuente: Elaboración propia

Posteriormente el usuario ingresa al menú principal que brinda cuatro opciones, la primera corresponde a la función en la que se muestran los establecimientos de la zona comercial en la que se encuentra el usuario, la segunda opción corresponde a la cuponera virtual en la que se encuentran las promociones que el usuario guardó previamente, en la tercera opción se puede actualizar la ubicación del usuario, esta se usa cuando un usuario se trasladó a otra zona comercial y por último se tiene la opción de configuraciones de la aplicación.



Figura 122: Menú principal de la aplicación
Fuente: Elaboración propia

Al elegir la primera opción se puede ver la pantalla en la que se muestra los establecimientos que se encuentran en la misma zona comercial que el usuario, estos se presentan en una tabla cada uno con su nombre y su respectivo logo, el usuario debe seleccionar la opción del establecimiento del cual desea conocer sus promociones.

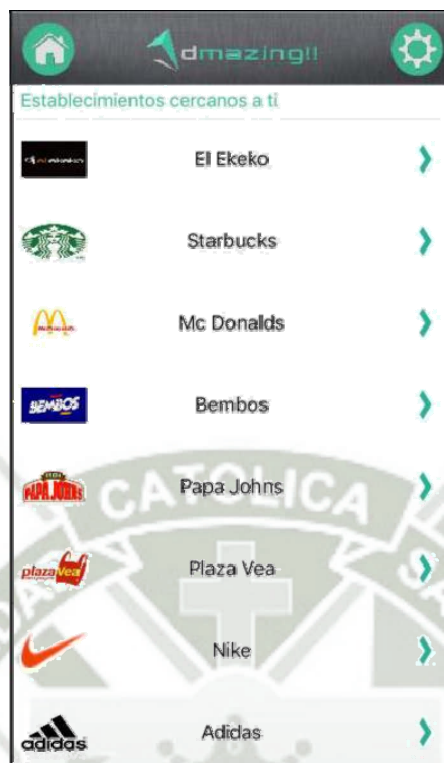


Figura 123: Actividad de listado de establecimientos cercanos al usuario
Fuente: Elaboración propia

A continuación, se puede ver la ubicación de la tienda seleccionada en el mapa y en la parte inferior de la pantalla se puede observar las categorías de productos o servicios con las que cuenta dicho establecimiento, el usuario debe seleccionar la categoría a la que desea acceder para poder ver las promociones.



Figura 124: Actividad de listado de categorías dentro de un establecimiento seleccionado
Fuente: Elaboración propia

Posteriormente se pueden observar las promociones que se encuentran en la categoría que fue seleccionada por el usuario, estas promociones se presentan con una pequeña descripción y una imagen de referencia del producto o servicio que se está promocionando, para acceder a ella el usuario debe seleccionarla.



Figura 125: Actividad de listado de promociones dentro de un establecimiento
Fuente: Elaboración propia

Después de acceder a la promoción que desea ver el usuario se muestra los detalles de esta promoción con información adicional como la descripción del producto, el precio original del producto y la fecha de vencimiento de la promoción, además en la parte inferior se presentan dos opciones, con el primer botón se añade dicha promoción a la cuponera virtual del usuario y con la segunda opción se accede a la cuponera virtual mediante un atajo sin necesidad de ir al menú principal.



Figura 126: Actividad de muestra de una promoción cercana al usuario
Fuente: Elaboración propia

Al ingresar a la cuponera virtual de la aplicación, se muestra de la misma manera, un mapa en el que se indica el punto en el que se encuentra el establecimiento y el lugar en donde se encuentra el usuario, en la parte inferior de esta vista se puede observar el conjunto de promociones que fueron agregadas en la misma zona comercial en la que se encuentra el usuario. Cuando el usuario selecciona una de estas promociones, se actualiza el mapa con la localización del establecimiento que tiene dicha promoción, si el usuario desea acceder a dicha promoción, debe seleccionar dos veces la promoción deseada.

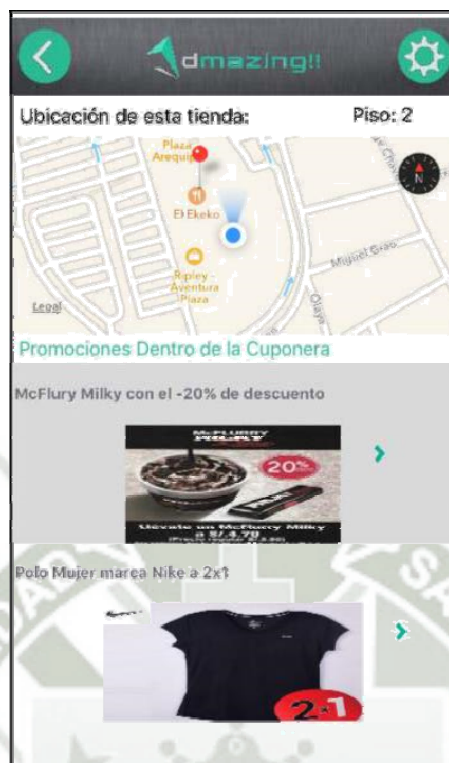


Figura 127: Actividad de muestra de la cuponera virtual del usuario
Fuente: Elaboración propia

Luego se muestra la promoción seleccionada, lista para su uso, se muestran los detalles de la promoción, además de los detalles del producto o servicio que se está promocionando. En la parte inferior de esta vista, se puede observar los botones para compartir esta promoción en las redes sociales de Facebook® y Twitter®. Además se muestra la opción para poder usar la promoción y el botón que nos permite borrarla de la cuponera virtual.



Figura 128: Actividad de muestra de una promoción dentro de la cuponera virtual
Fuente: Elaboración propia

Si se desea compartir la promoción seleccionada, simplemente debe seleccionar la red social con la que se desea realizar esta acción, y a continuación se mostrará una de las ventanas que se muestran en la parte inferior, luego el usuario debe indicar los detalles de la promoción que desea compartir y seleccionar la opción “Post” que se encuentra en la esquina superior izquierda de la pantalla.

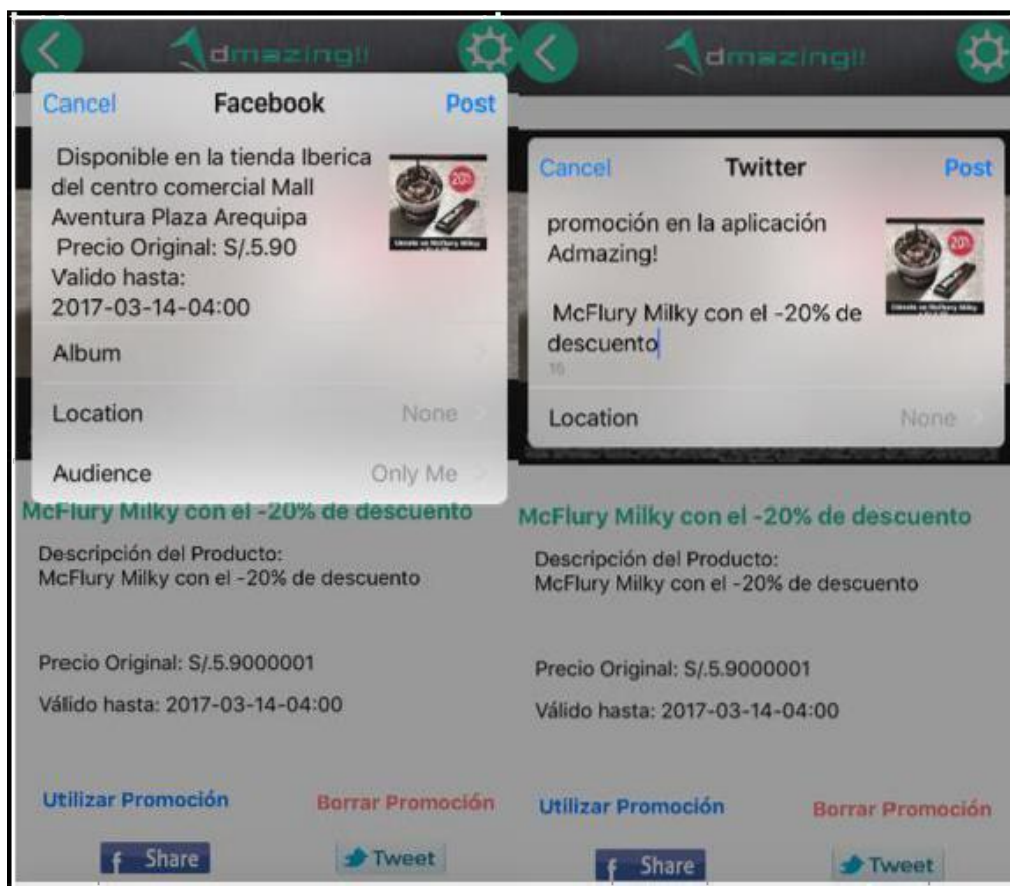


Figura 129: Actividades de publicación de una promoción en Facebook y Twitter
Fuente: Elaboración propia

Quando se selecciona la opción de usar promoción se muestra la siguiente pantalla, la cual corresponde al código QR generado por la aplicación el cual manda la acción de usar el cupón al software que tiene el establecimiento para usar la promoción de la aplicación, después de esto el usuario debe presionar el botón “Listo” para confirmar esta acción.



Figura 130: Actividad de muestra del código QR generado para el uso de una promoción
Fuente: Elaboración propia

La última opción del menú principal es la vista de configuraciones de la aplicación, dentro de esta se muestra el nombre del usuario que ha ingresado al sistema, la opción para definir si se muestran las promociones basándose en las preferencias del usuario, la opción para el uso de alertas cuando una promoción dentro de la cuponera virtual del usuario está próxima a su vencimiento, la opción de actualización de la ubicación del usuario, la opción de actualización de categorías preferidas por el usuario y por último la opción de cerrar la sesión del usuario.



Figura 131: Actividad de configuraciones de la aplicación móvil

Fuente: Elaboración propia

Para hacer la actualización de las preferencias del usuario se ingresa en la siguiente vista, dentro de esta se muestran las categorías con las que se cuenta y si el switch de una categoría está marcado significa que dicha categoría es una preferencia, en este caso se debe marcar las que se desean considerar como preferencias y desmarcar las que ya no se desean como preferencias, después de eso el usuario debe presionar el botón aceptar en la parte inferior para confirmar los cambios.

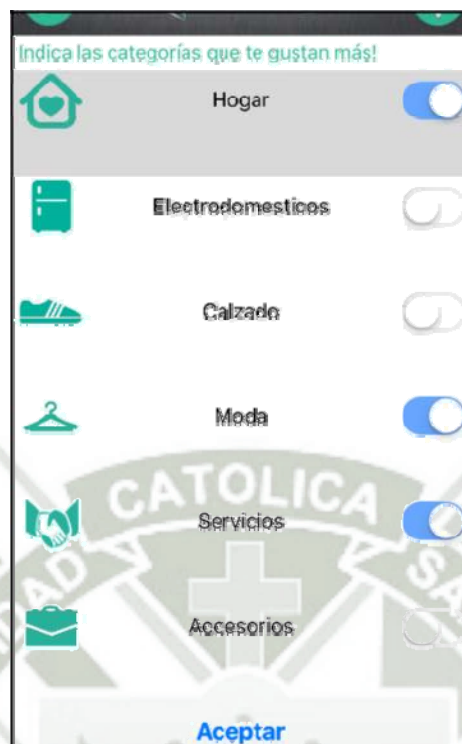


Figura 132: Actividad de actualización de categorías de preferencias
Fuente: Elaboración propia

Por parte de la aplicación wearable, esta es más sencilla, también se cuenta con un menú principal con cuatro opciones, la primera opción actualiza la ubicación del usuario dentro del sistema, la segunda muestra el último cupón visto en la aplicación móvil, la tercera opción genera el código QR de la promoción sin hacer una previa visualización, y la última opción corresponde a las configuraciones de la aplicación,



Figura 133: Menú principal de la aplicación wearable
Fuente: Elaboración propia

Dentro de la opción de último cupón se muestra la siguiente vista, en ella se detalla la promoción y el producto o servicio que se está promocionando, también muestra el botón para utilizar el cupón.



Figura 134: Actividad de muestra de una promoción dentro de la aplicación wearable
Fuente: Elaboración propia

Cuando el usuario indica que desea usar la promoción, la aplicación genera el código QR, lo muestra en la siguiente vista y simplemente el usuario debe hacer que el establecimiento lea este código para hacer efectiva la promoción.



Figura 135: Actividad de muestra código QR generado para el uso de una promoción dentro de la aplicación wearable
Fuente: Elaboración propia

Conclusiones

Después del desarrollo y pruebas del sistema de publicidad virtual utilizando dispositivos móviles y wearable se puede concluir los siguientes puntos.

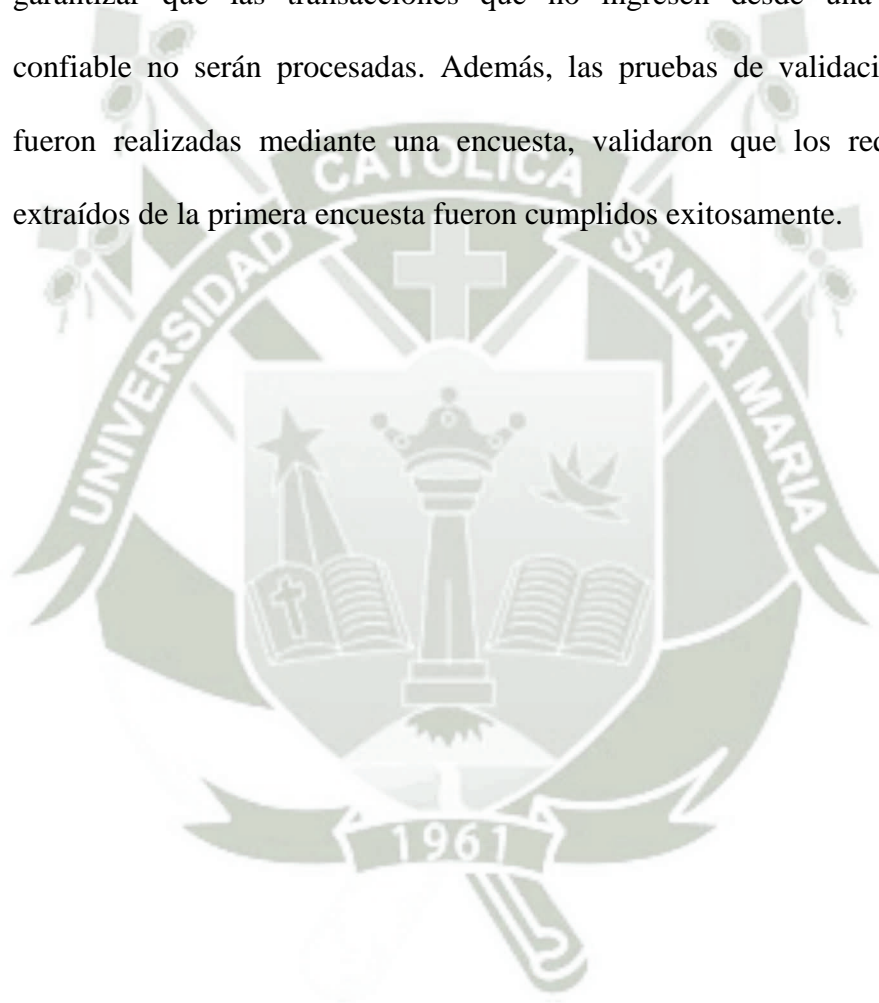
1. El uso de una arquitectura de tres capas para este sistema fue óptimo, dado que cada uno de los elementos del sistema cubría una funcionalidad específica para garantizar independencia entre capas, además brinda la posibilidad de mejorar en aspectos tales como la escalabilidad, flexibilidad, mantenibilidad y la portabilidad que serán necesarios para el futuro.
2. Aplicar una metodología ágil de desarrollo de software fue óptimo para este proyecto, dado que se contaba con un lapso corto de tiempo para su desarrollo y se necesitaba una correcta administración de recursos del proyecto. Se eligió la metodología de programación extrema porque fue la mejor para este proyecto, por el número de integrantes del equipo de desarrollo, por el número de requerimientos que se desarrollaron para este sistema, la alta flexibilidad que brinda y la comunicación que se tenía sobre el sistema que se estaba desarrollando.
3. Dentro del análisis de sistemas de publicidad similares a esta se observó que varias funcionalidades son iguales, sin embargo, el sistema de promociones desarrollado- cuenta con el manejo de preferencias o personalización de publicidad según el uso del usuario, para brindar al usuario una mejor

manera de utilizar sus promociones y garantizar una mejor experiencia con este sistema. Además los usuarios se sintieron a gusto con las funcionalidades del software, dado que estas se basaron en las funciones que ellos deseaban que fueran implementadas dentro de este sistema, estas funcionalidades fueron extraídas de una encuesta que les fue hecha en una fase previa al desarrollo.

4. Las aplicaciones para iPhones y Apple Watches fueron desarrolladas con una interfaz de usuario intuitiva, lo cual es comprobado gracias a que el 94.4% de los encuestados afirman que es relativamente fácil o muy fácil el uso de las aplicaciones, lo cual fue logrado al usar las recomendaciones de la Guía de interfaz humana de IOS desarrollada por Apple Developers (2016). Además, el 80.3% de los encuestados indicaron que la aplicación presenta mecanismos de seguridad, ya que se les mostró los mecanismos de seguridad que se implementaron para el cuidado de sus datos personales, entre ellos, la implementación de la autenticación de usuario y la autenticación de mensajes entre la aplicación móvil y el servicio web.

5. Se obtuvieron los resultados esperados después de realizar los tres tipos de pruebas de verificación. En las pruebas de ejecución, todas las aserciones recibe las respuestas adecuada de acuerdo al contenido de las solicitudes enviadas; en las pruebas de rendimiento, todas las medidas de rendimiento del sistema en cuanto al uso de recursos de hardware se

encontraron dentro de los límites establecidos por Apple Developers (2016) y las medidas de rendimiento correspondientes al tiempo de respuesta del servicio web están dentro de los límites de tiempo establecidos por Nielsen, J. (1993) y con las pruebas de seguridad se logró garantizar que las transacciones que no ingresen desde una fuente confiable no serán procesadas. Además, las pruebas de validación que fueron realizadas mediante una encuesta, validaron que los requisitos extraídos de la primera encuesta fueron cumplidos exitosamente.



Recomendaciones

Dentro de las recomendaciones que se pueden plantear al finalizar este proyecto es el cambio de publicidad tradicional por este nuevo sistema de publicidad virtual, además que se cuentan con funciones especiales que no brinda la publicidad tradicional como la geolocalización y la gestión de preferencias del usuario para la correcta asignación de publicidad a los clientes.

Otro de los planes más importantes a futuro es poder realizar esta aplicación para el sistema operativo Android para la aplicación móvil y además trabajar con una aplicación para relojes inteligentes con el sistema operativo Android Wear. Con esto se desea llegar a un mayor número de personas las cuales cuentan con este tipo de dispositivos.

Para complementar el proyecto se necesita la implementación del sistema para que las empresas puedan subir las promociones que desean mostrar dentro de esta aplicación y además poder asignarles detalles importantes como la ubicación de los establecimientos que aceptan las promociones. Además del sistema que se encargará de leer los códigos QR que genera esta aplicación y hacer que las promociones se efectúen dentro del sistema de cada establecimiento.

Además el objetivo a largo plazo es dar a las empresas asociadas a este sistema, la información sobre las tendencias de mercado en el menor rango de tiempo posible, esta información estará basada en los datos recolectados a diario

por la aplicación. Esta información es crucial para la evolución de empresas, ya que de esta manera se puede realizar la toma de decisiones fundamentada en información verídica y actualizada. Se desea que el alcance de este proyecto sea global, puesto que las problemáticas que se han planteado con anterioridad tienen un impacto a nivel mundial, y con ello se desea lograr adjuntar al servicio una mayor cantidad de entidades comerciales con el objetivo de brindar al usuario publicidad lo más variada y completa posible.

Otra de las características que se trabajará en el futuro es la inclusión de inteligencia de negocios dentro de este sistema para poder trabajar la gestión de preferencia de una manera más óptima, utilizando inteligencia artificial para la gestión de preferencias y una mejor asignación de promociones a cada usuario basándose en preferencias y patrones de compra de usuarios que cuenten con un perfil parecido al suyo. De esta manera se podrá garantizar una mejor experiencia para el usuario y una manera más avanzada para el proceso de gestión de los datos del sistema, los cuales deberán estar guardados dentro de un data warehouse.

Dentro de las características que desean los usuarios para este sistema es poder indicarles cuáles de sus amigos se encuentran dentro de la misma zona comercial para poder programar una reunión además de un sistema de comunicación utilizando chat. Esta es una de las funcionalidades que se desean

trabajar para futuro, de tal manera que se tiene un característica que pueda atraer más usuarios hacia la aplicación.

Por último se desea implementar un sistema de recompensas, de esta manera se podrá asignar promociones especiales a clientes que usen con mayor frecuencia esta aplicación, de esta manera se podrá fidelizar más clientes al sistema.

Se recomienda implementar las funcionalidades expuestas previamente para poder atraer más usuarios a este sistema y al mismo tiempo fidelizarlos. De esta manera además se podrá expandir este sistema e implantarlo en un mayor número de ciudades alrededor del mundo, lo cual es óptimo dado que con este sistema se desea crear un ambiente de desarrollo sostenible de la publicidad y se conoce que esto es de interés global.

Referencias bibliográficas

- Apache. (2015). Apache CXF: An Open-Source Services Framework. Retrieved November 08, 2016, from <https://cxf.apache.org/>.
- Apache. (2016). Maven – Welcome to Apache Maven. Retrieved November 08, 2016, from <https://maven.apache.org/index.html>.
- Apple Developers (2016). IOS Human Interface Guidelines. Retrieved September 29, 2016, from <https://developer.apple.com/ios/human-interface-guidelines/>.
- Chen, P & Hsieh, H. (2012). Personalized mobile advertising: Its key attributes, trends and social impact. *Technological Forecasting and Social Change*, 79(3), 543–557.
- Finamore, A., Shah, J., Vallina-Rodriguez, N. (2012). Breaking for Commercials: Characterizing Mobile Advertising. *Internet measurement conference*, 1, 343-356.
- Groupon. (2016). Groupon Press. Retrieved November 01, 2016, from <https://www.groupon.com/press/about-groupon>.

- Gurumurthy, S., Narayanamoorthy, M., Valarmozhi, A. (2011). An Application Model for Pervasive Computing. the 6th annual international conference on Mobile computing and networking, 1, 266-274.
- Tsang, M., Ho, S., Liang, T. (2005). Consumer Attitudes Toward Mobile Advertising: An Empirical Study. International Journal of Electronic Commerce, 8, 65–78.
- Kurkovsky, S. (2013). Pervasive Computing: Past, Present and Future. ITI 5th International Conference, 5, 65 - 71.
- Martin, R. (2002). The Principles of OOD. Retrieved November 12, 2016, from <http://butunclebob.com/ArticleS.UncleBob.PrinciplesOfOod>
- Mohan, P., Nath, S., Riva, O. (2013). Prefetching mobile ads: Can advertising systems afford it? 8th ACM European Conference on Computer Systems, 1, 267-280.
- Nielsen, J. (1993). Chapter 5 – Usability Heuristics. In Usability engineering (1st ed.). Boston: Academic Press.
- Oracle. (2011). What is SOAP? - Oracle GlassFish Server Message Queue 4.5 Developer's Guide for Java Clients. Retrieved November 07, 2016, from https://docs.oracle.com/cd/E18930_01/html/821-2440/aeqey.html.

- Pegler, M, Shahabi, K & Lamano, J. (2014). The Wearable Future. PricewaterhouseCoopers.
- Redhat. (2016). Hibernate ORM. Retrieved November 09, 2016, from <http://hibernate.org/orm/>.
- RetailMeNot. (2016). RetailMeNot.com Coupon Codes and Discounts. Retrieved November 01, 2016, from <https://www.retailmenot.com/>.
- Shopular. (2014). Shopular | Mobile Coupons Always With You, Always Saving. Retrieved November 01, 2016, from <http://www.shopular.com/>.
- SnipSnap. (2014). SnipSnap - Never Forget A Coupon. Retrieved November 01, 2016, from <http://www.snipsnap.it/>.
- Thecouponsappcom. (2016). TheCouponsAppcom. Retrieved November 01, 2016, from <http://www.thecouponsapp.com/>.
- Thierer, A. (2015). The Internet of Things and Wearable Technology. Richmond Journal of Law and Technology, 21, 118-123.

Apéndice A

PLAN DE PROYECTO DE TIC APROBADO



UNIVERSIDAD CATOLICA DE SANTA MARIA
FACULTAD DE CIENCIAS E INGENIERIAS FISICAS Y FORMALES
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS

Desarrollo de un sistema de publicidad utilizando tecnologías móviles, wearables y computación ubicua.

**Proyecto de Tesis presentado para optar el:
Título profesional de Ingeniero de Sistemas**

Cristopher Núñez Del Prado Mansilla

Diana Lucía Rodríguez Delgado

Asesor: Álvaro Fernández del Carpio

Arequipa, 2015

1. Planteamiento del problema

1.1. Caracterización del Problema

Actualmente las empresas tienen un gran déficit en cuanto a la información que poseen de sus clientes y potenciales compradores, por lo cual no se puede realizar una publicidad adecuada y por lo tanto se pierden muchas oportunidades de ventas y de obtención de clientes, incluso en ciertas ocasiones logran disgustar a sus propios clientes en vez de fidelizarlos con la empresa que sería lo más óptimo.

Existe un gran problema para las empresas al realizar publicidad en distintas zonas comerciales, estos entregan publicidad física, que en muchos casos no es efectiva para los planes que ellos tienen en promocionar sus productos o servicios debido a que este tipo de publicidad es muy general y no es muy interesante para el público, según Tsang, Ho y Liang (2005) la publicidad física será reemplazada por publicidad virtual, con este cambio se logrará agregar características para la personalización de esta.

Otro de los problemas que se buscan solucionar con este proyecto es la contaminación visual que existe en ciertas zonas comerciales de las ciudades, estas zonas suelen estar cubiertas de publicidad, que en muchos casos, en lugar de incitar a los clientes a comprar los productos expuestos, los hostigan y perturban. Además estas empresas adquieren una imagen poco ecológica ya que para elaborar tantos afiches publicitarios invierten muchos recursos.

Los problemas planteados previamente serán disminuidos con la herramienta que se viene desarrollando, por medio de esta se desea poder realizar publicidad virtual para reemplazar a la publicidad física convencional, de esta manera se pueden aprovechar las nuevas características que ofrece este tipo de publicidad, tales como la asignación personalizada de publicidad para poder mostrar a los usuarios lo que les podría interesar según sus gustos. Según Kurkovsk (2013) "Hoy en día, los teléfonos móviles actúan como puntos finales personalizados para muchos servicios de información" por tal motivo se desea utilizar los dispositivos móviles como la interfaz principal con la cual el usuario interactuará con la herramienta, además que con ellos se podrán extraer datos de los usuarios para poder retroalimentar el sistema utilizando los sensores que brinda el móvil, como sensores de geolocalización, entre otros.

Esta herramienta es realizada para poder crear un nuevo enfoque de publicidad y poder reinventarla brindando publicidad a los usuarios sobre productos o servicios que ellos deseen, en el momento y lugar correcto con lo que se podría aumentar el porcentaje de ventas de los establecimientos que se encuentran cercanos a él. Se predice que el porcentaje aumentaría gracias al incremento del alcance de la publicidad, y para demostrar este cambio en el alcance se contabilizará cada vez que un cliente observe una promoción a través de nuestro servicio.

La importancia de esta herramienta que estamos desarrollando es alta, debido a que ayudará a cubrir un problema muy importante a nivel internacional, la contaminación, la cual se verá reducida debido a que se aspira a la eliminación de la entrega de publicidad utilizando papel impreso o grandes *banners* publicitarios, se desea que toda esa publicidad sea electrónica dentro de los dispositivos electrónicos de las personas. Al reducir la producción de afiches y *banners* publicitarios se utilizarán menos recursos naturales alrededor del mundo y se logrará disminuir el problema de la contaminación ambiental.

Con los motivos previamente expuestos se concluye que la herramienta que se está desarrollando beneficiará mucho a las personas a nivel global dado que cubre problemas económicos, ecológicos y sociales, además logra un desarrollo sostenible de la publicidad. La herramienta contará con muchas características que ayudarán a las personas en diferentes aspectos de la vida cotidiana. Esta herramienta utiliza la metodología de desarrollo ágil de programación extrema (XP) y la arquitectura de tres capas, por medio de ambas se podrá desarrollar de una manera más óptima y garantizar a las personas que la usan, una mejor experiencia al momento de hacer compras, gestionar sus transacciones y controlar su economía.

1.2. Línea y Sub-línea de Investigación a la que corresponde el Problema

- Línea: Inteligencia Artificial
- Sublínea: Computación Ubicua

1.3. Palabras Clave

Ubicuidad, Publicidad, Ecología, Aplicaciones móviles, Wearable, Ubiquitous, Advertising, Ecology, Mobile Application.

2. Objetivos del proyecto

2.1. General

Desarrollar un Sistema de Publicidad utilizando tecnologías móviles, wearables y computación ubicua.

2.2. Específicos

Desarrollar una arquitectura de tres capas, que sea capaz de soportar el sistema de publicidad que se desea desarrollar, orientado a la producción de software de calidad.

Aplicar la metodología ágil de programación extrema (XP) en el desarrollo del sistema de publicidad.

Obtener información de los usuarios de la aplicación mediante el uso del sistema y a través de los sensores del dispositivo móvil.

Desarrollar una aplicación móvil con interfaz intuitiva, la cual será utilizada por el usuario para poder acceder, de forma sencilla, a toda la información que ofrece el sistema; además se aplicaron métodos

de autenticación de datos para proteger la integridad de las transacciones de los usuarios.

Desarrollar las pruebas del sistema utilizando pruebas de validación y verificación para garantizar el correcto funcionamiento del sistema.

3. Fundamentos teóricos

3.1. Estado del arte

La publicidad es la manera de ofrecer los productos que vende una empresa a los clientes, el objetivo principal de la publicidad es brindarle información a los clientes sobre los productos o servicios que una empresa ofrece, sin embargo, existen bastantes tipos de publicidad que en ocasiones son ignorados por los clientes o hasta rechazados. En estos tiempos de dispositivos digitales y móviles, se opta por utilizar publicidad utilizando estos medios electrónicos, ya que, ese es el destino de la publicidad según Tsang, Ho y Liang (2005), sin embargo, se sabe que los clientes toman una mala actitud a la publicidad si esta no es aprobada previamente por ellos mismos antes de recibirla, por tal motivo se debe tener un permiso del usuario para poder enviarle la publicidad mediante este servicio, por eso, la herramienta realiza esta petición de permiso antes de instalarse en el dispositivo del cliente. Además se debería tomar una manera de adicionar nuevas

características a la publicidad, según Tsang, Ho y Liang (2005) "La actitud de los clientes es positiva si se relaciona con la intención de recibir publicidad móvil. La intención es afectada por la asociación del incentivo con la publicidad", por tal motivo se desea que el aplicativo en desarrollo posea estas características adicionales con la finalidad de atraer al cliente sin causarle las molestias comunes que van de la mano con la publicidad.

Según Vallina-Rodriguez, Shah, Finamore et al. (2012) "Las redes de anuncios son de gran impacto para los usuarios, especialmente en Android, sino también en el iPhone y el iPad, el ecosistema del anuncio para aplicaciones móviles está dominado principalmente por los servicios de Google®". Por tal motivo se desea que el aplicativo en desarrollo sea destinado para los dispositivos móviles con sistema operativo Android o iOS, además se desea que funcione en wearables como relojes o lentes inteligentes, por tal motivo se plantea realizar este aplicativo con una arquitectura de tres capas en la cual la capa de interfaz pueda ser desarrollada en aplicaciones para diferentes sistemas operativos, la capa de lógica de negocio será única para todos ellos, eliminando la duplicación de código y posibles conflictos futuros al realizar algún cambio, esta estará localizada en forma de web services y finalmente la capa de Base de Datos.

Según Gurumurthy, S., Narayanamoorthy, M., Valarmozhi (2011) "Para que la computación ubicua se una a las expectativas de los usuarios móviles es

fundamental que los cambios ocurran de manera en que las personas perciban los roles de los dispositivos, aplicaciones y el ambiente”, por tal motivo esta aplicación móvil será siempre la interfaz con el usuario y se tomará en cuenta los roles que se tienen dentro del sistema, el usuario solo podrá ver y utilizar las promociones, un rol distinto tendrán las empresas que se encargaran de brindar las promociones para los usuarios.

Según Kurkovsk (2013) "Hoy en día, los teléfonos móviles actúan como puntos finales personalizados para muchos servicios de información". Por tal motivo se utilizarán los dispositivos móviles como los extractores de los datos de los clientes que usen el aplicativo, además también de los wearables, la idea es que este sistema se encuentre en cualquier lugar, y funcione también de igual manera en cualquier zona comercial.

Como nos plantea Mohan, Nath y Riva (2013) la publicidad en dispositivos móviles debe contar con un servidor que se encargará del almacenamiento, procesamiento, asignación y envío de publicidad sin importar el tipo de dispositivo o sistema operativo que lo solicita, en la arquitectura de tres capas planteada, este servidor se encuentra en la capa de lógica de negocio del sistema.

Según Kurkovsky, S. (2013) “Una propiedad fundamental de los sistemas ubicuos es la integración del ambiente y rodear al usuario con esta presencia ubicua, los teléfonos móviles pueden ser dispositivos de interfaz importantes que conectan

los usuarios al ambiente ubicuo” por tal motivo este sistema tendrá como interfaz a los usuarios principalmente aplicaciones móviles realizadas para los sistemas operativos de Android y iOS, además de sus correspondientes aplicaciones para relojes inteligentes.

Al trabajar con dispositivos móviles y al realizar transacciones de los clientes mediante estos, se debe tener en cuenta la seguridad de los datos con los que se están trabajando en este aplicativo, por ello se debe enviar los datos con los que se está trabajando, utilizando mecanismos de encriptación provistos por estándares ya implementados en los web services, que se usará en el desarrollo del sistema, Alomair y Poovendran (2011) indicaron que "Al tomar ventaja del hecho de que el mensaje a ser autenticado debe ser cifrado, proponemos códigos de autenticación demostrablemente seguro que son más eficientes que cualquier código de autenticación de mensaje en la literatura". Sabemos que los dispositivos wearables trabajan con todo tipo de datos del usuario los cuales pueden correr bastante peligro en manos equivocadas, en este caso los wearables y móviles están siendo los principales extractores de datos de las preferencias de los usuarios por lo cual se debe tener especial cuidado con ellos, Polonetsky, Wolf y Fich (2015) indicaron que "Las organizaciones deben estar preparadas para defender los datos personales de sus clientes de amenazas, como personal interno o amenazas externas, como podrían ser hackers o scanners. Dado un alto nivel de conectividad, también es

importante que los datos del wearable sean asegurados adecuadamente tanto dentro del dispositivo como en la nube o cualquier otro lugar donde se encuentre".

Según Thierer, A. (2015) "Los desafíos con la privacidad y de seguridad asociados con el internet de las cosas y las tecnologías portátiles serán considerables" por tal motivo cada usuario debe estar registrado para poder ingresar al sistema, por tal motivo ninguna persona podrá ver su información sin haber iniciado sesión con su correspondiente usuario.

Por tal motivo se concluye indicando que se desarrollará este sistema de publicidad utilizando la computación ubicua para que pueda ser accedido en todos los lugares en lo que se encuentren zonas comerciales, además del uso de tecnologías como los móviles y wearables que son el presente y futuro de la tecnología. Este aplicativo será multiplataforma y será desarrollado en una arquitectura de tres capas, debido a que debe ser escalable y debe tener un tiempo óptimo de respuesta, por tal motivo ninguna de las capas de la arquitectura debe estar desarrollada de una manera que afecte el desempeño de otra capa. Se debe tener en cuenta también que se trabajará con datos de compras de los clientes, por tal motivo, todos estos datos deben ser protegidos tanto de atacantes internos como de atacantes externos.

Las ventajas de utilizar wearables en la publicidad son variadas, los wearables permiten la entrega de publicidad de una manera mejor y con una mayor

relevancia al usuario, la ventaja más significativa al utilizar wearables en la publicidad móvil es la capacidad de utilizar los datos de los sensores que estos poseen, por ejemplo el sensor de geolocalización para conocer la ubicación del usuario y poder enviar publicidad de la zona en la que se encuentra, estos datos son extraídos del usuario por el wearable en todo momento y existen una gran gama de datos que se pueden extraer del usuario los cuales son muy importantes en este campo, estos datos ayudan a la efectividad de búsqueda de nuevos factores o comportamientos del usuario para la correcta asignación de publicidad. Los wearables hacen que la publicidad se base en la actividad del usuario y la integra estrechamente con otros contenidos y experiencias.

En el futuro se espera que la publicidad sea dirigida, optimizada y sea entregada en el momento y lugar perfecto para el usuario siguiendo distintos tipos de características del usuario como necesidades o emocionales.

Según Pegler, Shahabi & Lamano (2014), de igual manera se espera que en el futuro los wearables permitan una mayor cantidad de sensores de medición lo cual favorecerá a la obtención de nuevos tipos de datos. Los nuevos sensores y los nuevos tipos de mediciones por parte de los wearables permitirán en el futuro nuevos tipos de datos que considerarán dentro de la asignación de publicidad, lo cual permitirá el estudio de nuevos factores que se deben tomar en cuenta en la

asignación de publicidad orientada al usuario y ayudarán a mejorar la experiencia del usuario gracias a la mejor personalización de este servicio.

3.2. Bases Teóricas del proyecto

Los wearables son dispositivos cuya funcionalidad principal es la de procesar información móvil para así personalizarla a cada usuario, que puede ser aplicada en múltiples áreas como son la monitorización de la salud, juegos inmersivos, estadísticas del estado físico entre otras. Esta tecnología emergente está basada en la percepción, el procesamiento, almacenamiento, transmisión, lo cual permite obtener información y aplicaciones más adaptables a la circunstancia y al usuario. Sazonov y Neuman (2014).

La publicidad de la forma tradicional tiene naturaleza pasiva, es aquella que aparece con imágenes y texto en periódicos y revistas, además de la publicidad dinámica que engloba las que son transmitidas por emisiones de radio y televisión Chen y Hsieh (2012). Gracias al uso de las tecnologías de Comunicación y los dispositivos wearables, es posible realizar una publicidad que esté focalizada en el cliente, ya que usualmente estos dispositivos poseen información útil sobre el usuario, de este modo se logra evitar la propagación recurrente y generalizada de publicidad no deseada que puede llegar a hostigar al cliente. La publicidad personalizada se refiere a cuando la publicidad que se proporciona al cliente está

de acuerdo a la información recolectada del mismo ofreciendo así un servicio con mayor valor tanto para el cliente como para la organización.

Existen diferentes aplicaciones para gestión de promociones y cupones, como por ejemplo MyCouponGenie® (2016) que es una aplicación multiplataforma que provee cupones según los gustos del cliente, Theouponsapp® (2016) que es una aplicación que provee códigos de promociones, están agrupadas por tiendas, brinda recordatorios y permite guardar las tiendas favoritas y por último Groupon (2016) que es el sistema de publicidad más importante en el mundo la cual tiene funcionalidades parecidas a las que se desean desarrollar en esta, este también cuenta con aplicaciones para los móviles de plataformas Android y iOS, provee cupones ordenados por tipo de producto y por fecha de caducidad. Sin embargo no utiliza la geolocalización para brindar promociones solo de los lugares que se encuentran cerca del usuario, tampoco tiene la capacidad de segmentar la publicidad la publicidad en diferentes categorías, como tiendas o tipos de productos según las preferencias de configuración del usuario, además este sistema será un sistema que migre al mercado de los wearables para poder mejorar la calidad del servicio.

Por su parte este sistema contará con todas estas características, las promociones se agruparán por rubros y tiendas. Además permitirá al usuario obtener las promociones que le interesan según el lugar en el que se encuentra

evitando de esta manera sobrecargar al usuario de promociones que no le serán útiles en ese momento.

4. Presentación del proyecto

4.1. Justificación

Ese proyecto es desarrollado para poder menguar las problemáticas planteadas previamente, un aporte importante de este proyecto es que será una aplicación que contribuirá a contrarrestar un problema global crítico que cada vez más va tomando fuerza alrededor del mundo, el cual es la contaminación ambiental. Mediante el sistema se desea menguar de forma gradual es el desperdicio de materiales en la publicidad convencional como el papel o el plástico en banners y afiches publicitarios, ya que se considera que el problema de la contaminación ambiental está incrementando de manera alarmante, además se busca traspasar la posta al nuevo paradigma de publicidad que se presenta con esta aplicación.

También se desea poder realizar la gestión de esta publicidad en cupones dentro de la aplicación de manera sencilla y rápida, también ayudará a los usuarios a gestionar cada una de las transacciones de compra que se realicen a través de la aplicación, estas serán utilizadas para efectuar una retroalimentación en este sistema. Todas las funciones que serán incluidas en la aplicación son pensadas con el objetivo de satisfacer a los usuarios a poder administrar su economía más fácilmente.

Finalmente, la idea de diseñar este sistema es el poder crear una aplicación que pueda ayudar a las personas a encontrar la publicidad personalizada, debido a que esta es una de las mayores causas del poco alcance de la publicidad convencional y vendría a ser uno de los factores de diferenciación de este sistema frente a otras similares. Esta característica se logrará gracias a la extracción de datos del usuario conseguida mediante la aplicación móvil o el wearable, como por ejemplo su ubicación geográfica, de esta manera se podrá brindar publicidad de establecimientos que se encuentren cerca al usuario, en el futuro se desearía tomar en cuenta otro tipo de características como la hora del día o el clima para poder asignar otro tipo de promociones según estas características, como por ejemplo bebidas calientes para los días con bajas temperaturas u ofertas que solo se presenten en una determinada hora del día.

4.2. Resumen del Proyecto

4.2.1. Descripción del Proyecto a medio y largo plazo.

El resultado que se obtiene del proyecto es un sistema de publicidad basado en la arquitectura de tres capas creada que proporcionará el soporte del proyecto, la capa de lógica de datos será la encargada de comunicar el sistema de datos, con la interfaz de usuario que es la aplicación móvil. El usuario tendrá únicamente acceso a la publicidad personalizada proporcionada a través de la aplicación móvil, más

no tendrá acceso al sistema Web, esta aplicación está basada en la geolocalización y publicidad focalizada en los gustos de cada usuario.

Uno de los principales beneficios de la aplicación para la sociedad es la disminución de contaminación visual y física, ya que el sistema pertenece al área de la publicidad virtual, la cual evita el desperdicio en impresiones, tales como afiches, folletos, panfletos entre otros, que usualmente terminan en basureros o aún peor en el suelo. El uso de publicidad virtual permitirá brindar una imagen mejor, más ordenada y más limpia de las ciudades, que es aspecto a tener en cuenta dado que el Perú es considerado un país turístico por excelencia. La utilidad que presenta para el usuario, es una publicidad más centrada en sus gustos y en la zona comercial donde se encuentra, esto permite que el usuario pueda acceder más rápidamente a aquello que realmente le interese, eliminando publicidad genérica y que el usuario ahorre tiempo en la búsqueda o que evite perder aquellas ofertas que le sean muy útiles.

El alcance del proyecto a medio plazo se define en la creación del sistema de publicidad focalizada que está basada en la geolocalización y las preferencias de los usuarios, en conjunto con el sistema de datos y la aplicación móvil, es decir que está altamente centrado en el usuario de la aplicación, especialmente en mejorar su experiencia con la publicidad focalizada.

El alcance a largo plazo se centra más en las empresas asociadas al sistema, las cuales son parte del proyecto y los beneficios que estas pueden obtener al ser parte del servicio que se brindará, el objetivo a largo plazo es dar a las empresas información sobre las tendencias de mercado en el menor rango de tiempo posible, esta información estará basada en los datos recolectados a diario por la aplicación. Esta información es crucial para la evolución de empresas, ya que de esta manera se puede realizar la toma de decisiones fundamentada en información verídica y actualizada. Se desea que el alcance de este proyecto sea global, puesto que las problemáticas que se han planteado con anterioridad tienen un impacto a nivel mundial, y con ello se desea lograr adjuntar al servicio una mayor cantidad de entidades comerciales con el objetivo de brindar al usuario publicidad más variada y completa.

4.2.2 Usuarios del Proyecto.

Los *stakeholders* principales del sistema son los usuarios finales, aquellos que utilizan la aplicación. Si bien la aplicación está destinada a ser gratuita en las plataformas de Android y IOS, el público objetivo de esta aplicación serán aquellos con poder adquisitivo de regular a alto, ya que son estos a los que principalmente les interesa realizar compras en lugares que congregan múltiples tiendas tales como los centros comerciales.

Los usuarios finales son considerados los beneficiarios ya que ellos, son los principales receptores del servicio de publicidad focalizada y geolocalizada mediante la cual el beneficiario logrará obtener la publicidad en el lugar que se encuentre y cuando este lo desee, también podrá gestionar la publicidad y cupones que tienen en la aplicación basados en los gustos, preferencias o necesidades del usuario.

Sin embargo, a largo plazo se considera que las empresas serán las principales beneficiarias del proyecto, dado que se les proporcionará información sobre el sector del mercado al cual pertenecen, movimientos y las tendencias actualizadas constantemente de sus consumidores. Además las empresas podrán promocionar sus productos o servicios en la aplicación, la cual hará de mediadora entre las empresas y los usuarios, ambas partes se verán beneficiadas por la aplicación. Por un lado las empresas reciben beneficios de la publicidad más acertada hacia sus usuarios e información que les permite estar a la altura de las necesidades de los clientes, y por el otro lado los usuarios recibirán un servicio de mejor calidad por parte de las empresas y mejores publicidades por parte de la aplicación.

4.2.3 Beneficios.

Dado que el proyecto se basa en publicidad virtual, la sociedad en sí se verá beneficiada ya que el uso de este tipo específico de publicidad ayuda a contrarrestar

la contaminación visual y física, porque se pretende disminuir la deforestación, al utilizar medios electrónicos para transmitir la publicidad en lugar de utilizar materias primas como papel como se da en el caso de la publicidad física. Además, la publicidad virtual propone información sobre publicidad personalizada a cada usuario lo que le permitirá a este obtener mejores beneficios de las ofertas, ya que únicamente obtendrá aquellas que se ajusten a sus preferencias.

Asimismo, las empresas se verán beneficiadas, debido a que la distribución de la publicidad dentro de la aplicación será de manera gratuita, y de esta manera podrán incrementar ventas, por medio de la aplicación las empresas afiliadas lograrán reducir su inversión en publicidad e incrementar sus ganancias. Este ciclo beneficia tanto a las empresas como al servicio que se busca brindar pues al proporcionar un buen servicio a las empresas, más de ellas se afiliarán al servicio, con ello el servicio tendrá más y mejor publicidad para proveer a los usuarios y por lo tanto atraerá a más de ellos, incrementando el alcance de la publicidad.

4.2.4 Localización.

El núcleo de la gestión del proyecto una vez culminado, se centrará en Arequipa, ya que el proyecto está destinado a focalizarse inicialmente en centros comerciales de Arequipa, así la gestión de publicidad podrá hacerse de forma más sencilla. Los servidores necesarios para el funcionamiento de esta aplicación serán tomados mediante un servicio de la compañía Amazon®, con el fin de asegurar un

alto nivel de confiabilidad y seguridad de los datos de los usuarios que se transmiten a través del mismo, esta empresa es reconocida por brindar seguridad y actualización rápida en contra de las posibles fallas o posibles ataques de hackers informáticos.

El equipo de este proyecto trabajará en Arequipa, haciendo gestión remota a los servidores de Amazon® ubicados en Estados Unidos, se desarrollará la aplicación, la gestión y el mantenimiento de esta desde Arequipa, en el futuro con la expansión de la aplicación se espera tener un equipo de gestión en cada lugar para una mejor administración de los nuevos establecimientos en los distintos lugares en los cuales estas se afilien, además de tener atención a los usuarios, en caso de reclamos, sugerencias entre otros. Así poder brindarles un mejor servicio y tengan un mayor nivel de aceptación.

4.2.5 Análisis del futuro del Proyecto.

Los usuarios de la aplicación que son los principales actores dentro de esta obtienen grandes beneficios, ya que reciben publicidad basada en sus gustos, preferencias y necesidades basados en factores específicos, estos se actualizan conforme el usuario interactúe con la aplicación, así poder adaptarse más fielmente a los verdaderos gustos del usuario, todo esto hará que la aplicación sea cada vez más personalizada para el usuario y este se beneficie mejor después de cada experiencia con la aplicación.

Las empresas obtendrán grandes beneficios debido a que la aplicación les permitirá realizar publicidad sin costo alguno, para que estos puedan mostrar y publicitar sus productos o servicios en oferta y puedan obtener un mayor número de clientes, además del incremento en sus ventas, con esto ellos podrían disminuir la inversión en publicidad al poder realizar una mejor publicidad y una más personalizada a sus clientes gracias a esta herramienta.

La sociedad será altamente beneficiada por esta aplicación, ya que la aplicación permite menguar el uso de recursos naturales para fines publicitarios. Se utilizan medios electrónicos para lograr la difusión de publicidad, de este modo se brinda a las personas publicidad que de verdad requiere en el momento y lugar apropiado para garantizar su efectividad.

4.2.6. Riesgos que debemos afrontar.

Este proyecto es ambicioso en diferentes aspectos, por lo tanto se deben considerar muchos riesgos para poder ser desarrollado e implantado, estos riesgos son de diferentes tipos, entre ellos se encuentran los riesgos económicos del proyecto los cuales se dan debido a los gastos constantes que deben hacerse en los servicios de almacenamiento de datos en la nube y los servidores de aplicaciones pues estos servicios se tomarán de una empresa externa al proyecto, algunos de estos servicios pueden ser costosos, sin embargo deben ser tomados con el fin de garantizar el correcto funcionamiento de la herramienta. La mayoría de estos

servicios se manejan a través de mensualidades o anualidades. Para la implantación se deben tomar servicios más especializados para que la herramienta logre soportar un mayor número de usuarios, además de garantizar óptimas características de la herramienta.

De igual manera se presenta el riesgo tecnológico, ya que algunas funciones de la aplicación necesitan dispositivos móviles de alta o media gama, pues es necesario realizar la utilización de los componentes que los dispositivos de media a alta gama poseen, tales como los sensores de GPS para efectuar la geolocalización del usuario, tecnología de telefonía móvil 4G, pues esta cuenta con velocidades de transmisión altas permitiendo así al usuario obtener el máximo provecho de todas las funciones de la aplicación.

Otro de los riesgos importantes de esta herramienta en desarrollo, es la desconfianza por parte de los usuarios para compartir sus datos personales o datos sobre los productos adquiridos por ellos, a través de una aplicación “desconocida”, este comportamiento es usual en especial entre los usuarios latinoamericanos que es el mercado al cual se apunta inicialmente, los usuarios suelen tener temor a que esta información sea ventilada o que personas ajenas al proyecto puedan utilizar estos datos con otros fines, por lo que se debe concientizar a los usuarios y demostrarles los niveles de seguridad de la información que la herramienta brinda al usuario, para que se sientan seguros de que los datos extraídos son únicamente

para conocer sus preferencias y poder hacer sugerencias de la publicidad que ellos desearían.

La competencia es importante y es un riesgo a tomar en cuenta, debido a que grandes empresas como Google® brindan servicios de almacenamiento de cupones de descuento, sin embargo estos no son personalizados según las preferencias del usuario, el cual es considerado el valor agregado más importante de esta herramienta, además que Google® con su herramienta Google Wallet tiene como objetivo el ayudar a gestionar la economía del usuario que es un enfoque distinto al servicio que se pretende brindar.

5. Plan de implantación del proyecto

5.1. Definición del Proyecto.

5.1.1. Aspectos Técnicos

El proyecto " Desarrollo de un sistema de publicidad utilizando tecnologías móviles, wearables y computación ubicua ", requiere primordialmente del uso de tecnologías móviles y wearables, para que el sistema sea escalable y de alta disponibilidad, se propone el uso de la arquitectura de tres capas, puesto que esta permite la cohesión de los componentes sin que estos resulten acoplados entre sí. En la capa de presentación se encuentra la aplicación móvil, la cual será la interfaz que el usuario utilizará para poder obtener su publicidad focalizada, esta interfaz

estará conectada a un sistema de Servicios Web, esta es la capa de lógica de datos, capaz de soportar y procesar los pedidos de la aplicación y que podrá conectarse con la capa de datos más eficientemente y así extraer los datos necesarios para responder a la petición realizada desde el dispositivo del usuario. Como se menciona anteriormente es necesario un sistema de almacenamiento que sea capaz de proveer información a las empresas sobre las tendencias del mercado, ya que provee la posibilidad de adquirir información de los datos eficaz y eficientemente.

Siempre que exista una transmisión entre dispositivos es indispensable la seguridad en dicha transmisión, por ello se protege la integridad de la información que el usuario provee mediante métodos de encriptación, así se protege la confidencialidad de datos del usuario que son exigidos no sólo por la ética en el proyecto, sino también por la ley peruana 30096 "Ley de Delitos Informáticos". Congreso de la República del Perú, (2013).

5.1.2. Aspectos Económicos

La publicidad física que realiza cada establecimiento en la zona comercial en un mes es aproximadamente un mínimo de 400 dólares americanos y si esta publicidad es realizada además por internet o aplicaciones móviles, esta cifra podría triplicarse, en muchos casos esta publicidad no tiene la efectividad que se desea, porque es enviada al público en general, siendo la mínima parte de ellos, el mercado objetivo de cada publicidad específica. Con la herramienta se busca poder

reducir esta inversión de publicidad por parte de las empresas, gracias a que la publicidad se brindará por un servicio gratuito. Además la proveerá a los usuarios con alta probabilidad de interés en sus productos o servicios para lograr así, garantizar un cliente potencial para el establecimiento.

A largo plazo se desea brindar un servicio de asesores de publicidad gratuito a los establecimientos para poder determinar la calidad de su publicidad y asesoría para poder tener una mejor llegada al cliente. Este servicio está valorado aproximadamente en un mínimo de 300 dólares americanos al mes, sin embargo, es una inversión que los establecimientos afiliados evitarían, al contrario de la publicidad convencional.

Los usuarios de la aplicación ganarán ofertas cada vez que estén en una zona comercial, estas ofertas garantizarán un ahorro de al menos el 25% de las compras que realicen en total, con lo que podrán poder adquirir nuevos productos de los establecimientos a los que estos acuden.

En cuanto al cuidado del medio ambiente, lo que provee esta aplicación es invaluable económicamente, pues se lograría evitar una inversión de recursos como papel, plástico, entre otros en ámbitos de publicidad. Con lo anteriormente mencionado se garantiza un desarrollo sostenible de la aplicación.

5.1.3. Aspectos Comerciales

El proyecto busca desarrollar una herramienta que cuenta con diferentes funciones que brindarán al usuario una opción diferente en sus hábitos de compra y ahorro, cada una de estas funciones es atractiva, ya que, brinda un diferente beneficio al usuario, y al mismo tiempo logra que la herramienta tenga más aceptación entre los usuarios.

Dentro de las funciones de la aplicación está la de proporcionar publicidad a los usuarios, esta será segmentada de la manera en que el usuario lo prefiera, según tiendas o categorías de productos, solo se presentará publicidad de productos o servicios que se encuentren en la misma zona comercial en la que él se encuentra. La aplicación brindará publicidad personalizada en el usuario, es decir que esta se basará en compras anteriores del usuario, patrones de compras descubiertos a partir de un conjunto de usuarios con características, preferencias y necesidades similares, de esta manera se busca fidelizarlo a la aplicación y que tenga satisfacción con la publicidad brindada, a largo plazo se espera que también contemple aspectos como si se encuentra solo comprando o acompañado, qué tipo de productos le gustan a su acompañante, si desea comer algo dentro de la zona comercial, si existen promociones grupales, etc.

Se contará con una cuponera para poder guardar las promociones especiales para cada usuario, con esta se espera poder terminar con los cupones físicos de los usuarios y evitar el gasto de papel. También se procederá a una buena gestión de

estos cupones debido a que la aplicación mandará una alerta si algún cupón vencerá pronto.

5.1.4. Recursos del Proyecto

Para este proyecto se deben contar con diferentes servicios de almacenamiento, procesamiento de datos, hosting, etc.

Se debe contar con un servicio de almacenamiento de datos en la nube, el cual será tomado de la empresa Amazon® para que se consiga guardar todos los datos de los usuarios y las transacciones que los mismos realizan, para poder determinar la publicidad que se asignará para cada uno de ellos. Se debe contar también con un servidor de procesamiento de datos para que pueda recibir las solicitudes de los dispositivos móviles según el usuario que las realice, consulte lo necesario en el servidor de la base de datos y pueda enviar la respuesta correcta a los usuarios, estas respuestas deben ser lo más rápidas posibles, para lo cual se contratará el servidor de aplicaciones, este proveerá de igual manera la seguridad indispensable que el proyecto requiere.

Se deben tomar en cuenta también que es necesario colocar la aplicación en las plataformas móviles *Playstore* y *Appstore* para que la aplicación logre el alcance deseado, además se debe contar con una licencia especial de desarrollador para el caso del desarrollo de la aplicación en iOS.

También son indispensables los dispositivos móviles con diferentes sistemas operativos como celulares inteligentes y relojes inteligentes, para poder realizar las pruebas del correcto funcionamiento del sistema en diferentes escenarios, además de las pruebas de integración del sistema.

El proyecto a largo plazo requerirá profesionales en publicidad para poder desarrollar y validar los banners de publicidad que se pondrán en la aplicación, de esta manera obtener opiniones profesionales para averiguar si los anuncios serán aceptados por los clientes y poder contactar a las empresas en caso necesiten algún tipo de asesoría con el desarrollo de sus anuncios. De igual manera se requerirán profesionales en seguridad de la información para poder verificar la integridad de los datos y transmisiones, de los datos personales de los usuarios y las transacciones que serán realizadas con la aplicación, se necesitan métodos de encriptación de datos, métodos seguros de procesamiento de datos y pruebas de penetración regulares para encontrar vulnerabilidades en el sistema y que no sea fácil de invadir por una persona o entidad externa.

6. Metodología a emplear

Al realizar un proyecto de software es necesario elegir la metodología adecuada para desarrollarlo correctamente en el menor tiempo posible. Se eligió una metodología ágil frente a las metodologías tradicionales como el modelo de cascada, incremental, espiral, entre otros, ya que las metodologías tradicionales son

rígidas y estrictas frente a cambios, la participación por parte de los clientes es mínima o casi nula, los grupos de desarrollo son grandes y poco comunicativos, se realiza mínima o ninguna retroalimentación. Por el contrario, el atributo más representativo de las metodologías ágiles es la flexibilidad que tiene cuando ocurren de cambios en cualquier etapa del proyecto, los clientes se integran con los encargados del proyecto logrando una mejor comunicación, minimizando errores en requerimientos y maximizando la satisfacción de este, los grupos que integran el proyecto son pequeños, así se coordina de mejor manera el proyecto y la retroalimentación se da de manera más continua y efectiva.

Para el desarrollo del sistema de publicidad utilizando tecnologías móviles, wearables y computación ubicua, se utiliza la metodología de programación extrema, la cual fue determinada para este proyecto por la cantidad de personas que trabajan en el desarrollo de esta herramienta y las características determinantes de esta metodología, las cuales se adecuan de mejor manera proyectos como este. La metodología de programación extrema inicia con la extracción de requerimientos de los *stakeholders*, los cuales son organizados en historias de usuarios que contienen el requerimiento, la identidad del cliente que lo requiere y el beneficio que se obtendrá a partir de estos, estas historias de usuario son organizadas y se seleccionan las que son más importantes para el producto que se desea desarrollar. En este punto, se realiza la primera retroalimentación, el cliente del producto debe revisar las historias de usuario y selecciona cuales van de acuerdo a las necesidades

que tenga a fin de certificar si el producto está bien encaminado. Luego se planean las liberaciones, las tareas y requisitos especificados en las historias de usuario que se deben desarrollar y principalmente el tiempo que cada una de estas tareas va a necesitar para que sean finalizadas de la forma deseada.

Después de realizar el mapa de impacto se realizó el desarrollo del diseño y arquitectura del software que se está trabajando. Esta etapa de desarrollo fue realizada por ambos miembros del grupo para el correcto conocimiento de los detalles de las funciones del sistema, con esto se puede garantizar un correcto entendimiento compartido del código para las personas que lo implementaron. Además gracias a esto se trabajaba un diseño simple del sistema y el uso de estándares de programación para el código, como la implementación de patrones de diseño y la codificación del sistema, la cual fue realizada completamente en idioma inglés.

Existen varios motivos por los que se eligió la programación extrema para el desarrollo de este sistema, uno de ellos es la programación en pares, con esto se logró que ambos miembros del equipo de desarrollo conocieran más sobre el sistema y no existiese problema cuando se unió el servicio web a la aplicación móvil. Otra de las características más importantes que tiene esta metodología es la realización de pruebas continuas al código, con esto se puede lograr asegurar que

el código se desarrolla de manera efectiva, todo el tiempo de realizaron pruebas a las funciones nuevas que se implementaron dentro del sistema.

Otra característica importante de la programación extrema es la entrega de bloques pequeños de código la cual debe ser hecha cada cierto tiempo, en este caso se realizan envíos al repositorio del sistema, este era realizado al menos una vez cada dos días, a menos que se estuviera implementando una funcionalidad extensa y el desarrollo exigiera el uso de más tiempo. Para esto se estuvo trabajando con un repositorio de código en la nube, la herramienta que se usó para la función de almacenamiento de código es *GitHub*. Con esto también se lograba una integración de código continua realizado de manera eficaz.

En la sección del sistema que corresponde al servicio web, se realizó refactorización de código, lo cual también es una característica muy importante de esta metodología de desarrollo, esto se realizaba mediante constante eliminación de código duplicado o ineficiente, con lo cual el equipo de desarrollo mejoraba el diseño del sistema. En esta fase el código también era evaluado continuamente para ofrecer la mayor calidad posible.

En conclusión, esta metodología es la mejor para el proyecto, ya que permite la flexibilidad necesaria para que el este se adapte de la mejor manera a las necesidades del usuario, ya que el sistema presenta altos riesgos en cuanto a

cambios en los requisitos, dado que es un proyecto que tiene como clientes finales al público en general.

Se desarrollará una arquitectura de software de tres capas para separar las responsabilidades de cada uno de los componentes del proyecto, de esta manera cada capa será independiente y trabajará de manera autónoma con respecto a las demás, pues se busca que el sistema sea más fácil de mantener y pueda ser escalable para la adición de funciones que se mencionan anteriormente en la parte de largo plazo. Se eligió el modelo de tres capas ya que es altamente flexible, confiable, escalable y es mantenible, también permite que la capa de presentación que es la principal interfaz de interacción con el cliente sea ligera, puesto que la capa de lógica de negocio es la encargada del procesamiento de las solicitudes. Además permite realizar cambios sin necesidad de constantes actualizaciones en la interfaz de usuario y tiene alta tolerancia a fallos. Se elige esta arquitectura por encima de la arquitectura de dos capas porque se considera que este sistema tiene altas probabilidades de crecimiento y se necesita un modelo desacoplado que permita realizar estos cambios afectando mínimamente a los usuarios.

Para la realización de la interfaz de la aplicación que se está desarrollando se utilizará plantillas y patrones de color de aplicaciones que tengan el mismo fin para poder, de esta manera, garantizar que el usuario se sienta familiarizado a

primera vista con la aplicación que se está desarrollando y que la interfaz sea intuitiva.

Se realizarán diferentes tipos de pruebas, se realizarán pruebas unitarias, a cada uno de los componentes o métodos del sistema, para asegurar que los componentes funcionan de forma adecuada, posteriormente se realizarán pruebas de integración en donde se harán pruebas conjuntas de los componentes para poder probar su correcto performance de manera integrada. Después de estas pruebas se realizarán pruebas de sistema, para realizar la validación que se encarga de asegurar que el sistema satisface al usuario y finalmente verificación que determina si el software cumple con correctamente con los requerimientos especificados.

a) Análisis:

El sistema de publicidad tiene una alta viabilidad considerando las necesidades del mercado, pues actualmente la publicidad online se encuentra en muchas páginas, aplicaciones, entre ellos está Google®, Facebook®, etc. Además, se considera que tiene una alta rentabilidad, no basado en las descargas de la aplicación, ya que esta será gratuita, sino en el beneficio económico que se obtendrá de las empresas que publiquen sus productos en la aplicación.

Para poder extraer los requisitos se utilizarán las entrevistas con usuarios, principalmente utilizando prototipos o mock ups, porque se desea que los usuarios aporten sus ideas y entiendan de forma simple y rápida el funcionamiento de la

aplicación, se prevé que las entrevistas serán de baja duración a los usuarios que forman parte de las metodologías ágiles de desarrollo de software, en este caso programación extrema (XP) y las historias de usuario así se podrá asociar las necesidades encontradas en las historias de los distintos *stakeholders* mediante los "impactos" del sistema.

Para lograr un buen modelado funcional se utilizará 2 tipos de modelos con el objetivo de mostrar los procesos que se dan sin la aplicación y con ella, así se podrá comparar los procesos y cómo el proyecto aporta en su optimización. El primer modelo es el BPMN (*Business Process Model and Notation*), este método permite realizar el modelado del flujo de trabajo de la realidad actual y el modelo de ejecución o procesos, el cual permite realizar el modelado de los procesos que la aplicación realizará una vez implementada.

b) Diseño:

Para realizar el diseño de la arquitectura del sistema se utiliza el diagrama de componentes, porque muestra una vista general del sistema y permite implementar un sistema que no sea dependiente de los otros componentes logrando que, si ocurre algún error, este no se propague ni afecte a otras partes del sistema.

Para el diseño detallado se utiliza el clásico diagrama de clases ya que este es el que permite mostrar más detalladamente lo que se implementará, ya que muestra las clases con cada uno de los métodos y atributos que el sistema debe

implementar, además que permite que la nomenclatura sea conocida por ambos miembros que desarrollaran el sistema.

En lo que respecta al diseño de datos se utiliza un diagrama de Entidad-Relación el cual será implementado en el servidor de base de datos con la finalidad de atender las consultas que serán realizadas desde el servidor de aplicaciones

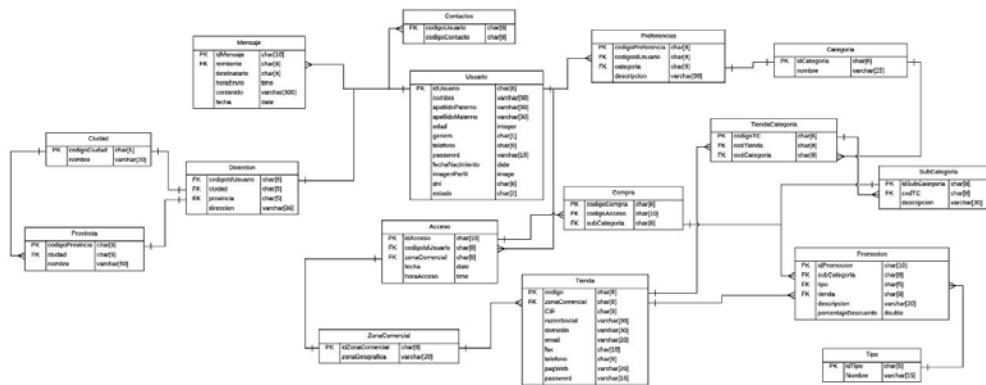


Figura A 1: Diagrama de base de datos sistema de publicidad
Fuente: Elaboración propia

c) Implementación:

En la etapa de implementación, usualmente debe centrarse en metodologías para lograr hacer entender al usuario el funcionamiento del sistema, sin embargo el sistema de publicidad está orientado al público en general y realizar la validación con el usuario es más compleja, por ello el proyecto se centrará en hacer que la interfaz de la aplicación para wearables y móviles sea lo más intuitiva posible, esta es una característica crucial para la aplicación porque pretende conseguir que el

usuario se vuelva más afín a la aplicación, la utilice más seguido y se familiarice con ella. Se debe tomar en consideración que el sistema de publicidad está orientado al público en general, por lo cual no se podrá hacer la metodología de capacitaciones que es usual en el ámbito empresarial.

La implantación del sistema se divide en 3 partes ya que se debe implantar la aplicación móvil, el web service y la base de datos. La primera es implantar la base de datos al servidor de datos que se tomará de un servicio contratado, una vez esta se termine de implantar, se deben realizar las pruebas correspondientes de conectividad.

La segunda parte es implantar el web service en el servidor de aplicaciones que de igual manera será contratado, una vez se logre adaptar la aplicación al nuevo servidor, es necesario llevar a cabo las pruebas de conectividad entre componentes dentro de la aplicación y entre los componentes del web service y el servidor de almacenamiento. Finalmente debe implantarse la aplicación para dispositivos móviles y wearables en las tiendas de aplicaciones para los sistemas operativos de dispositivos móviles, en el caso del sistema operativo Android se subirá a la plataforma *PlayStore* y para el sistema operativo IOS se subirá a la plataforma *AppStore*.

d) Pruebas:

Las pruebas son una de las partes más importantes en este sistema, debido a que se tiene que tener un sistema robusto, escalabilidad y confiabilidad para el usuario.

Para este sistema se utilizará la metodología de Testing de Caja Negra con el fin de asegurar su correcto funcionamiento una vez finalizada su implementación, estas pruebas serán basadas en los modelos especificados de las funcionalidades más cruciales, de esta manera se podrá garantizar el cumplimiento de los requerimientos funcionales de la aplicación.

Una vez implementado un requerimiento, se realizarán las pruebas unitarias a cada uno de los componentes de este, así se podrá garantizar su correcto funcionamiento de cada uno de estos de manera independiente al resto del sistema. Una vez realizadas con éxito todas las pruebas unitarias y concluidos los componentes que sean altamente relacionados con el procesamiento principal del sistema, se comunicarán y posteriormente evaluarán de manera integrada, para poder garantizar que trabajan en conjunto de manera óptima. Al terminar las pruebas, se debe realizar una validación que integre el sistema como una unidad y evalúe los requerimientos funcionales y no funcionales del sistema y que garantice que este cuenta con un buen performance.

Se deben hacer pruebas de hacer pruebas de verificación para contrastar cada una de las funciones del sistema con los requerimientos detallados en el Mapa

de Impacto del sistema y de esta forma garantizar que el sistema trabaja tal como fue planificado previamente. Para finalizar deben realizarse pruebas de validación, con el fin de determinar que el sistema funciona de manera óptima, si lo que se desarrolla es correcto según las opiniones de los usuarios, es decir que busca asegurar la aceptación del sistema por parte de los clientes.

7. Plan de trabajo

Para la realización del proyecto en desarrollo se realizará un diagrama de Gantt en el cuál se podrá identificar las tareas para realizar y las fechas en las cuales se realizarán cada una de estas, cada tarea cuenta con sub tareas para realizar a continuación se determinará los principales procesos del proyecto en desarrollo.

- Elaboración de Diseño y Arquitectura de Software
- Desarrollo del Sistema
- Pruebas
- Estudio de Resultados.

Dichas tareas serán mostradas en el siguiente diagrama de Gantt, además de la secuencia en la que se realizará cada proceso. También se determina la duración de cada proceso según el número de dedicación efectiva indicada en horas/hombre. Cabe recalcar que el proyecto será de una duración de 158 días, lo cual equivale a 7 semanas y cuatro días de trabajo.

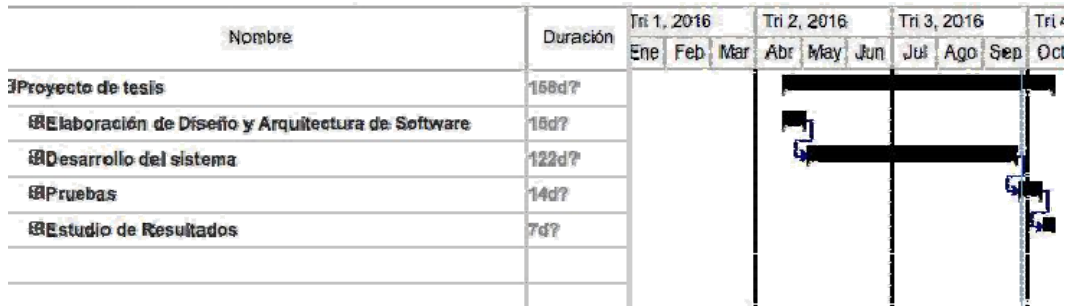


Figura A 2: Diagrama de Gantt de los procesos de proyecto
Fuente: Elaboración propia

A continuación se realizará el plan de actividades de cada uno de los procesos determinados para el proyecto, para empezar se mostrará el plan de actividades de la fase de Elaboración de Diseño y Arquitectura de Software, en la cual se identifican nuevas tareas, además se indica el tiempo de duración de cada una de estas y además la secuencia que cada una debe seguir. Con lo que se logra determinar que la fase de Elaboración de Diseño y Arquitectura de Software será de una duración de 2 semanas y 1 día de trabajo.

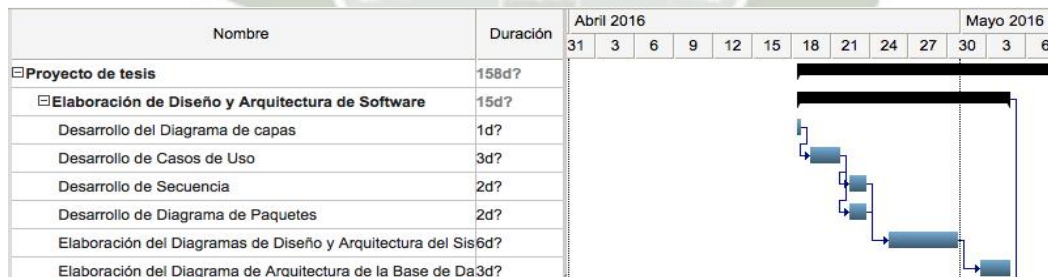


Figura A 3: Diagrama de Gantt de la fase de elaboración de diseño y arquitectura de software
Fuente: Elaboración propia

Posteriormente se realiza el planeamiento de la fase de desarrollo del sistema, en la cual se encontraron dos subprocesos principales, los cuales son el desarrollo de la aplicación móvil y el desarrollo del servicio web, los cuales son importantes para el sistema en desarrollo debido a que ambos tienen alta prioridad de uso. Se debe determinar las tareas de cada uno de estos subprocesos, además se asignan las fechas de inicio y fin, por lo que se determina que el desarrollo de Sistema tendrá una duración de 17 semanas y 4 días de trabajo.

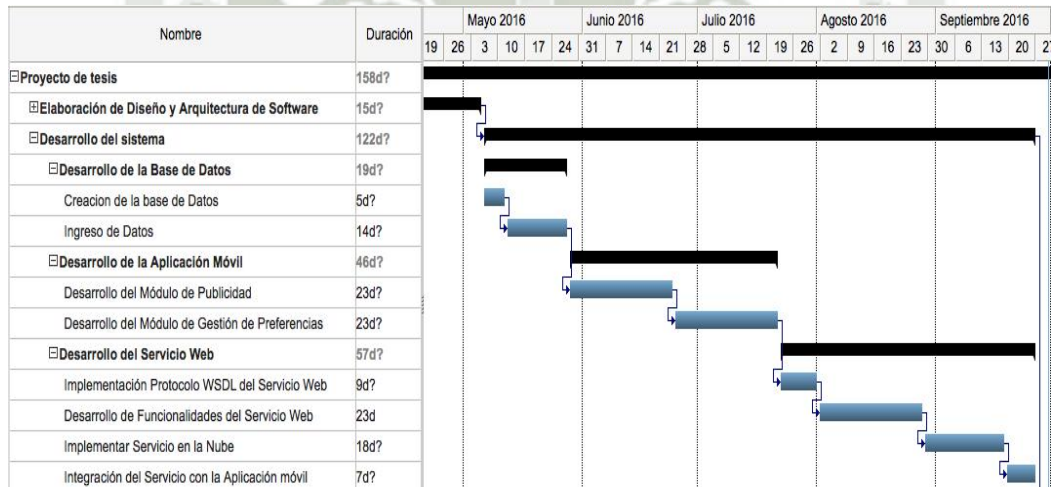


Figura A 4: Diagrama de Gantt de la fase de desarrollo del sistema

Fuente: Elaboración propia

Después de la fase de Desarrollo del sistema, se desarrolla la planeación del cronograma de la fase de pruebas, en la que se encontraron subtareas y se asignan las fechas necesarias para este proceso. Se determinó que la fase de pruebas del proyecto tendrá una duración de 2 semanas.



Figura A 5: Diagrama de Gantt de la fase de pruebas del sistema

Fuente: Elaboración propia

Posteriormente se determinó el cronograma para la fase de Estudio de Resultados, se identificó las sub tareas que tiene este proceso y se indicó la duración de cada una de estas y las fechas en la que se desarrollarán. Cabe recalcar que la fase de Estudio de Resultados tendrá una duración de 1 semana.

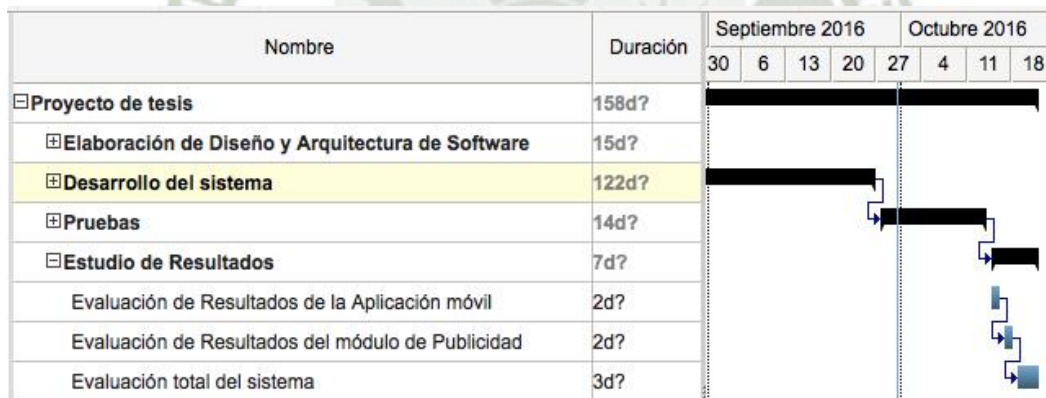


Figura A 6: Diagrama de Gantt de la fase de estudio de resultados del proyecto

Fuente: Elaboración propia

8. Referencias

- Alomair, B., Poovendran, R. (2011). Efficient Authentication for Mobile and Pervasive Computing. 12th International Conference on Information and Communications Security, 13, 469 - 481.
- Chen, P & Hsieh, H. (2012). Personalized mobile advertising: Its key attributes, trends and social impact. *Technological Forecasting and Social Change*, 79(3), 543–557.
- Congreso de la República del Perú. (2013). Ley 30096, Delitos Informáticos. En *Archivo Digital de la Legislación del Perú*. Perú.
- Finamore, A., Shah, J., Vallina-Rodriguez, N. (2012). Breaking for Commercials: Characterizing Mobile Advertising. *Internet measurement conference*, 1, 343-356
- Groupon. (2016). Groupon Press. Retrieved 20 April, 2016, from <https://www.groupon.com/press/about-groupon>
- Gurumurthy, S., Narayanamoorthy, M., Valarmozhi, A. (2011). An Application Model for Pervasive Computing. the 6th annual international conference on Mobile computing and networking, 1, 266-274

- Tsang, M., Ho, S., Liang, T. (2005). Consumer Attitudes Toward Mobile Advertising: An Empirical Study. *International Journal of Electronic Commerce*, 8, 65–78.
- Kurkovsky, S. (2013). Pervasive Computing: Past, Present and Future. *ITI 5th International Conference*, 5, 65 - 71.
- Mohan, P., Nath, S., Riva, O. (2013). Prefetching mobile ads: Can advertising systems afford it? *8th ACM European Conference on Computer Systems*, 1, 267-280.
- Mycoupongeniecom. (2016). My Coupon Genie. Retrieved 20 April, 2016, from https://www.mycoupongenie.com/page/how_it_works.
- Pegler, M, Shahabi, K & Lamano, J. (2014). The Wearable Future. PricewaterhouseCoopers.
- Sazonov E., Neuman M. (2014) *Wearable Sensors: Fundamentals, Implementation and Applications*, Elsevier, 1-5.
- Thecouponsappcom. (2016). TheCouponsAppcom. Retrieved 20 April, 2016, from <http://www.thecouponsapp.com/>
- Thierer, A. (2015). The Internet of Things and Wearable Technology. *Richmond Journal of Law and Technology*, 21, 118-123.

9. Temario del informe final

Cubierta

Cubierta Interna

Copia del Dictamen Aprobatorio del Borrador de

Tesis Presentación

Agradecimientos

Dedicatoria

Epígrafe

Índice o Tabla de Contenidos

Índice de Figuras.

Resumen y Abstract

Introducción

Desarrollo del Tema

Capítulo 1: Descripción del Proyecto

1.1. Objetivos

Objetivo General:

Objetivos Específicos:

1.2. Alcances y Limitaciones

1.3. Fundamentos Teóricos

1.3.1. Estado del arte (Antecedentes del proyecto)

1.3.2. Bases Teóricas del proyecto

1.4. Técnicas y Herramientas

1.5. Aspectos Relevantes del Desarrollo

Capítulo 2: Documentación Técnica

2.1 Plan del Proyecto Informático.

2.2. Especificación de requisitos del software (Elicitación y Análisis)

2.3. Especificación de diseño y arquitectura de software

2.4. Documentación técnica de programación

2.4.1 Documentación de la Aplicación Móvil

2.4.2 Documentación del servicio web

2.5. Pruebas de Ejecución.

2.5.1 Pruebas Unitarias

2.5.2 Pruebas Integradas

2.5.3 Pruebas de Sistema

2.6. Manuales de Usuario

2.6.1 Manual de Usuario de la Aplicación Móvil

2.6.2 Manual de Usuario del servicio web

Conclusiones

Recomendaciones

Referencias Bibliográficas

Apéndice(s)

Apéndice B

ENCUESTA DE VALIDACIÓN DE REQUERIMIENTOS A EMPRESAS

Para esta encuesta se necesitó a 12 empresas de la ciudad de Arequipa, las cuales se encontraron en dos diferentes zonas comerciales de la ciudad, Real Plaza Arequipa y la zona comercial de la Calle Mercaderes en el centro histórico de la ciudad.

En la primera pregunta se busca conocer el nombre de la empresa que se está encuestando, como se puede observar a continuación, las tiendas encuestadas son de diferentes zonas comerciales y vende productos de diferentes rubros, de esta manera se tendrá opiniones de diferentes tipos de establecimiento sobre el servicio que se planea brindar.

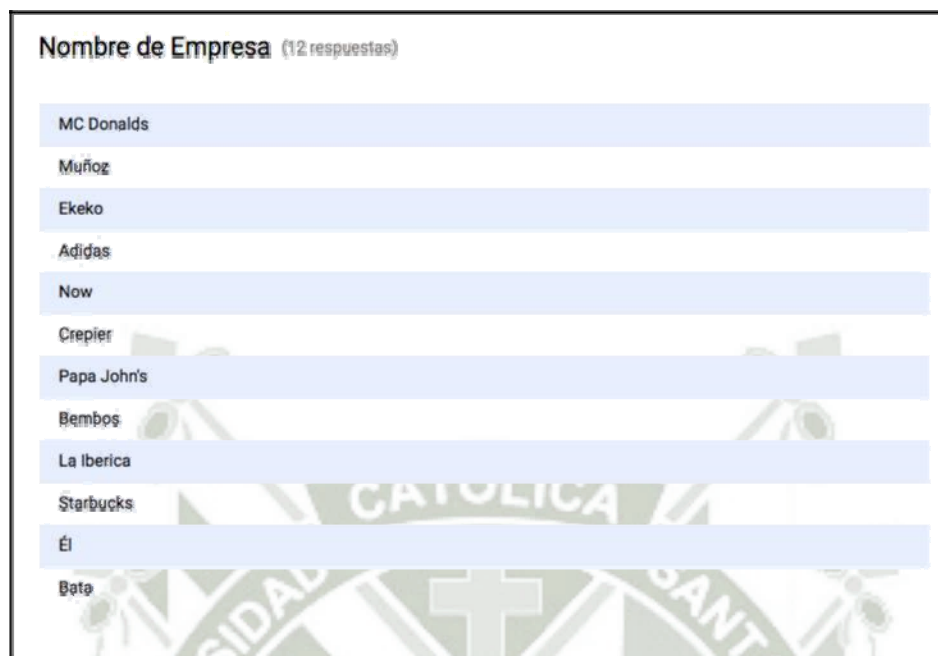


Figura B 1: Respuestas de la primera pregunta de la encuesta de validación a empresas
Fuente: Elaboración propia

A continuación, se busca conocer si estas empresas usan publicidad convencional y brindan folletos a sus clientes para ofrecer sus productos o servicios, se puede ver que el 75% de las empresas brindan folletos a sus clientes y el otro 25% de las empresas no usan este método de publicidad convencional, por lo que se puede concluir que la gran mayoría de estas empresas en la muestra publicitan sus productos mediante medios físicos.

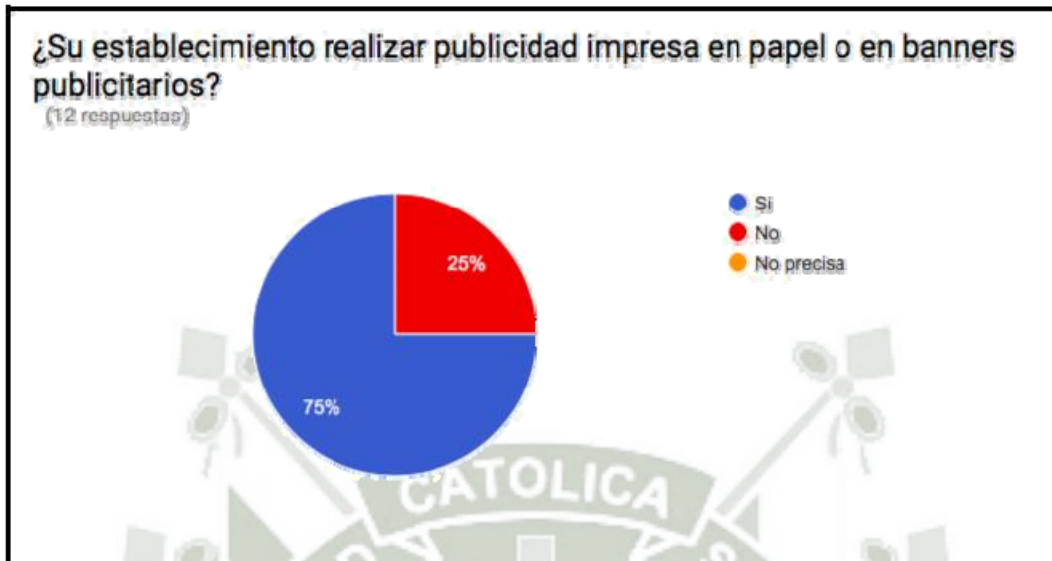


Figura B 2: Respuestas de la segunda pregunta de la encuesta de validación a empresas
Fuente: Elaboración propia

En la siguiente pregunta se busca conocer el valor aproximado de folletos que ofrece una empresa al día para poder conocer la cantidad aproximada de promociones que incluirían estas dentro de la aplicación, se puede observar que el 50% ofrece menos de 50 folletos al día a sus usuarios, el 23% brinda de 50 a 200 folletos al día a sus posibles compradores, el 16.7% de estas empresas dan de 200 a 500 folletos al día a sus clientes y el 8.3% brinda más de 700 folletos publicitarios a sus posibles clientes, esta información nos ayuda a conocer en promedio una empresa brinda alrededor de 1500 folletos al mes, si toda esta publicidad fuera virtual en vez de convencional se reduciría el uso de papel y esto beneficiaría de gran manera al medio ambiente.

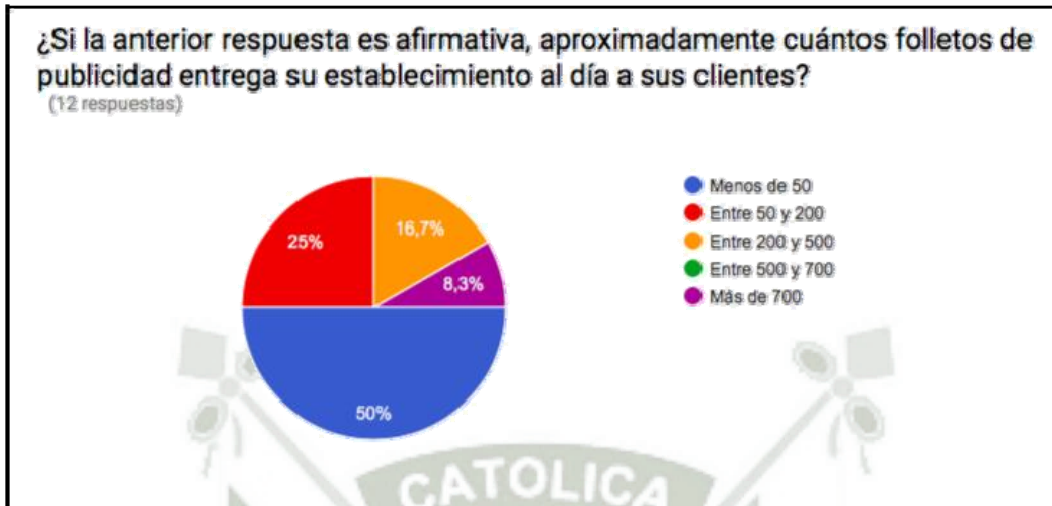


Figura B 3: Respuestas de la tercera pregunta de la encuesta de validación a empresas
Fuente: Elaboración propia

Para poder evaluar la eficacia de la publicidad convencional, se le preguntó al administrador de cada establecimiento, el porcentaje de eficacia de los folletos entregados reflejada en las ventas de su establecimiento. Los datos indicaron que el 50% de las empresas encuestadas creen que la eficacia de la publicidad convencional está entre 20 y 40% para su empresa, un 16.7% indicó que la eficacia de los folletos que ellos entregan está entre 80 y 100% reflejado en sus ventas, otro 16.7% indicó que la eficacia de la publicidad que realizan es de menos de 20% en cuanto a sus ventas, el 8.3% restante piensa que está entre 40 y 80% de eficacia reflejado en sus ventas. Esto nos indica que la eficacia de la publicidad convencional en su mayoría es baja para los administradores de los establecimientos en Arequipa.



Figura B 4: Respuestas de la cuarta pregunta de la encuesta de validación a empresas
Fuente: Elaboración propia

La siguiente pregunta ayuda a tener idea de lo que piensan los establecimientos en cuanto a nuestro servicio, se les preguntó si creían que este servicio ayudaría a mejorar efectivamente sus ventas, el 50% estuvo completamente de acuerdo en esto y el otro 50% se mostró simplemente de acuerdo. Esto ayuda a confirmar de manera definitiva que las empresas que fueron encuestadas confían en que esta aplicación puede ayudarlas a ganar más ventas y un mayor número de clientes.



Figura B 5: Respuestas de la quinta pregunta de la encuesta de validación a empresas
Fuente: Elaboración propia

La siguiente pregunta ayuda a conocer si los establecimientos desearían cambiar la publicidad tradicional por la virtual, los resultados nos indican que el 41.7% le gustaría cambiar la publicidad tradicional y usar la publicidad virtual, sin embargo el otro 41.7% no quisiera que exista este reemplazo y el 16.7 restante si quisiera que se realice este cambio completamente. Esto indica que los establecimientos desean que exista el reemplazo de publicidad tradicional por publicidad virtual, y otra gran mayoría desea que ambos tipos de publicidad trabajen al mismo tiempo, con esto se logra saber que los establecimientos se muestran abierto a este nuevo tipo de publicidad sin embargo algunos aún desean seguir con la publicidad convencional para no arriesgarse.



Figura B 6: Respuestas de la sexta pregunta de la encuesta de validación a empresas
Fuente: Elaboración propia

La siguiente pregunta ayuda a conocer si los establecimientos estarían satisfechos con el servicio que se planea brindar, para estos el 50% de los establecimientos encuestados indicaron que se encontrarían completamente satisfechos con el servicio, el 33.3% les gustaría probar el servicio y el 16.7% de los establecimientos encuestados les gustaría probar el servicio previamente para expresar su opinión. Estos datos nos indican que los establecimientos si estarían abiertos a usar este servicio para que les ayude a mejorar sus ventas y al mismo tiempo disminuir el impacto negativo causado al medio ambiente.

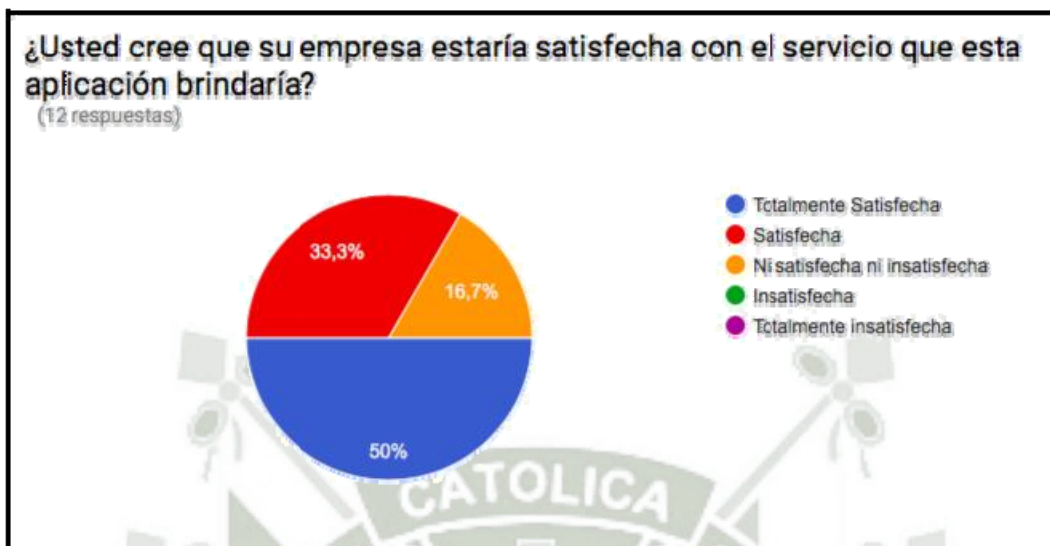


Figura B 7: Respuestas de la séptima pregunta de la encuesta de validación a empresas
Fuente: Elaboración propia

La siguiente pregunta es para poder buscar socios de la aplicación y saber qué empresas se encontrarían dispuestas a trabajar definitivamente con la aplicación, los resultados indican que el 100% de empresas encuestadas se encuentran dispuestas a ser socios y trabajar directamente con la aplicación, brindando su publicidad para que pueda ser mostrada dentro del servicio y pueda trabajar aumentando sus ventas.



Figura B 8: Respuestas de la octava pregunta de la encuesta de validación a empresas

Fuente: Elaboración propia

Apéndice C

DOCUMENTOS Y DETALLES DEL SISTEMA

Para mayor información sobre el diagrama de Gantt, ingrese en el siguiente [link:](https://drive.google.com/open?id=0B89A5S7AoRZ1TjJLeIB6Y0trVWc)
<https://drive.google.com/open?id=0B89A5S7AoRZ1TjJLeIB6Y0trVWc>, luego haga clic en la parte superior que indica “Abrir con Gantter for Google Drive”.

Se utilizó dos repositorios *Github* para el manejo de versiones de la aplicación y el sistema web por separado, para mayor información de la aplicación móvil consulte el siguiente [link:](https://github.com/cnpm18/AdmazingApp)
<https://github.com/cnpm18/AdmazingApp> y para mayor información sobre el web service consulte el siguiente [link:](https://github.com/dianalu01/WebWS)
<https://github.com/dianalu01/WebWS>.

Para mayor detalle de las encuestas realizadas a los usuarios antes del desarrollo del sistema consulte el siguiente [link:](https://docs.google.com/spreadsheets/d/1sZqIEMRzyMJmDPKtJfjGlaSHCcj6QfOSJQjVSfkbnEs/edit?usp=sharing)
<https://docs.google.com/spreadsheets/d/1sZqIEMRzyMJmDPKtJfjGlaSHCcj6QfOSJQjVSfkbnEs/edit?usp=sharing>, para mayores detalles de las encuestas al realizar la validación con el usuario una vez implementado el sistema, consulte el siguiente [link:](https://drive.google.com/open?id=1DevrTVw9VnGtQ74RBwGgm3fGkSQ8G2w0GstH-tsD89A)
<https://drive.google.com/open?id=1DevrTVw9VnGtQ74RBwGgm3fGkSQ8G2w0GstH-tsD89A> y finalmente, para detalles de las encuestas realizadas a empresas que se explicaron anteriormente en el apéndice B, consulte el siguiente [link:](#)

https://drive.google.com/open?id=1tkkZDqEjA_an-zLyra7YMfv7QWcJGqIT247yXnx0bI4.

