

# Trabajo Terminal

Sistema de recomendaciones basado en aprendizaje máquina  
y aplicación móvil conectada con emisores de señal Bluetooth  
Beacons (Sapphire)  
2017 - B034

---

Castro Flores Marcela  
Galindo García Adrián Eduardo  
Murga Dionicio Rubén Adolfo

Escuela Superior de Cómputo, IPN

6 de noviembre de 2018

---

## Índice general

---

<b>1. Introducción</b>	<b>15</b>
1.1. Contexto . . . . .	16
1.2. Problemática . . . . .	16
1.3. Estado del arte . . . . .	17
1.3.1. Tienda departamental Macy's . . . . .	17
1.3.2. Supermercado Carrefour . . . . .	18
1.3.3. Tienda de ropa American Eagle Outfitters . . . . .	18
1.3.4. Aplicación makeitapp . . . . .	19
1.3.5. TT 2015-A056 API para el desarrollo de sistemas de recomendación . . . . .	20
1.3.6. TT 2010-0063 Trabajo Terminal Sistema generador de recomendaciones para una tienda en-línea de videojuegos . . . . .	20
1.4. Solución propuesta . . . . .	21
1.5. Justificación . . . . .	22
1.6. Objetivos . . . . .	23
1.6.1. Objetivo general . . . . .	23
1.6.2. Objetivos específicos . . . . .	23
1.7. Metodología . . . . .	24
1.8. Beneficios esperados . . . . .	24
1.9. Alcance . . . . .	24
1.10. Limitantes . . . . .	25
<b>2. Marco teórico</b>	<b>26</b>
2.1. Aprendizaje máquina . . . . .	26
2.1.1. Antecedentes de la Inteligencia Artificial . . . . .	27
2.1.1.1. Nacimiento [1952-1956] . . . . .	27
2.1.1.2. Primer invierno de la Inteligencia Artificial [1974-1980] . . . . .	27
2.1.1.3. La explosión de los 80's [1980-1987] . . . . .	28
2.1.1.4. Segundo Invierno de la Inteligencia Artificial [1987-1993] . . . . .	28
2.1.1.5. Explosión y adopción comercial [2006-Actual] . . . . .	28
2.1.2. Tipos de aprendizaje . . . . .	29
2.1.2.1. Aprendizaje supervisado . . . . .	29
2.1.3. Aprendizaje no supervisado . . . . .	30
2.1.3.1. Algoritmo de agrupamiento K-means . . . . .	30
2.1.4. Algoritmos de Aprendizaje supervisado . . . . .	31
2.1.4.1. Hiperplano . . . . .	31

2.1.4.2.	Separabilidad lineal . . . . .	32
2.1.4.3.	Red neuronal artificial . . . . .	32
2.1.4.4.	Perceptrón . . . . .	33
2.1.4.5.	Sistemas de recomendación . . . . .	34
2.2.	IoE (Internet of Everything) . . . . .	38
2.2.1.	Antecedentes . . . . .	39
2.2.1.1.	IoT (Internet of Things) . . . . .	39
2.2.1.2.	Sistemas embebidos . . . . .	39
2.2.1.3.	Cómputo ubicuo . . . . .	40
2.2.2.	Tendencias . . . . .	40
2.2.2.1.	IoE e inteligencia artificial . . . . .	40
2.2.2.2.	IoE y conectividad . . . . .	40
2.2.3.	Ventajas . . . . .	41
2.2.4.	Desventajas . . . . .	41
2.2.5.	Ejemplos . . . . .	41
2.3.	Mobile Computing . . . . .	42
2.3.1.	Dispositivos de computación móvil . . . . .	42
2.3.1.1.	Asistente digital personal (PDA) . . . . .	42
2.3.1.2.	Tableta . . . . .	42
2.3.1.3.	PC Ultra-Móvil . . . . .	43
2.3.1.4.	Computadoras corporales . . . . .	43
2.3.1.5.	Teléfono inteligente . . . . .	43
2.3.2.	Antecedentes del teléfono inteligente . . . . .	44
2.3.3.	Sistemas operativos . . . . .	46
2.3.3.1.	Symbian . . . . .	46
2.3.3.2.	Windows . . . . .	46
2.3.3.3.	BlackBerry . . . . .	47
2.3.3.4.	iOS . . . . .	47
2.3.3.5.	Android . . . . .	47
2.3.4.	Limitaciones . . . . .	48
2.4.	Sistema de posicionamiento en interiores . . . . .	48
2.5.	Beacons Estimote . . . . .	49
2.5.0.1.	iBeacon y Eddystone . . . . .	49
2.5.0.2.	Estimote API . . . . .	50
2.6.	Marketing . . . . .	51
2.6.1.	Marketing de proximidad . . . . .	51
2.6.2.	Importancia del marketing de proximidad . . . . .	52
2.6.3.	Estrategias de marketing de proximidad . . . . .	52
2.6.3.1.	Conexión a una red WiFi . . . . .	52
2.6.3.2.	Cupones digitales en el lugar . . . . .	52
2.6.3.3.	Lectura de código QR . . . . .	52
2.6.3.4.	NFC . . . . .	52
2.6.3.5.	Beacons . . . . .	53
2.7.	Servidor Web asincrónico . . . . .	53
<b>3.</b>	<b>Bosquejo general de la solución</b> . . . . .	<b>54</b>
3.1.	Análisis de factibilidad . . . . .	54
3.1.1.	Factibilidad técnica . . . . .	54
3.1.2.	Factibilidad económica . . . . .	56
3.2.	Elección de tecnologías . . . . .	57
3.3.	Ánalisis de riesgos . . . . .	61
3.4.	Arquitectura de la aplicación . . . . .	67

3.5. Diccionario de actores . . . . .	68
3.6. Diagrama de casos de uso general . . . . .	68
3.7. Requerimientos Funcionales (RF) . . . . .	70
3.8. Requerimientos No Funcionales (RNF) . . . . .	77
<b>4. Modelo de datos</b>	<b>78</b>
4.1. Modelo relacional . . . . .	78
4.2. Prototipo 1: Diseño inicial de la base de datos . . . . .	84
4.2.1. Diccionario de datos . . . . .	84
4.2.2. Modelo de la base de datos . . . . .	100
4.3. Prototipo 1.1: Modificaciones al modelo de la base de datos . . . . .	104
4.3.1. Modificaciones realizadas . . . . .	104
4.3.2. Diccionario de datos . . . . .	104
4.3.3. Modelo de la base de datos . . . . .	107
4.4. Prototipo 1.2: Modificaciones al modelo de la base de datos . . . . .	111
4.4.1. Modificaciones realizadas . . . . .	111
4.4.2. Diccionario de datos de entidades modificadas . . . . .	111
4.4.3. Modelo de la base de datos . . . . .	111
4.5. Prototipo 1.3: Modificaciones al modelo de la base de datos . . . . .	115
4.5.1. Modificaciones realizadas . . . . .	115
4.5.2. Diccionario de datos de entidades modificadas . . . . .	115
4.5.3. Modelo de la base de datos . . . . .	115
4.6. Prototipo 1.4: Modificaciones al modelo de la base de datos . . . . .	119
4.6.1. Modificaciones realizadas . . . . .	119
4.6.2. Diccionario de datos de entidades modificadas . . . . .	119
4.6.3. Modelo de la base de datos . . . . .	123
4.7. Prototipo 2: Generador de Registros Artificiales (GRA) . . . . .	127
4.7.1. Análisis . . . . .	127
4.7.2. Diseño . . . . .	129
4.8. Prototipo 2.1: Generador de Registros Artificiales (GRA) . . . . .	141
4.8.1. Modificaciones realizadas . . . . .	141
4.8.2. Análisis . . . . .	141
4.8.3. Diseño . . . . .	144
<b>5. Aplicaciones móviles</b>	<b>148</b>
5.1. Aplicación Interactiva Difusora de Productos (AIDP) . . . . .	148
5.1.1. Prototipo 1: Diseño inicial de la aplicación . . . . .	149
5.1.1.1. Análisis . . . . .	149
5.1.1.2. Diseño . . . . .	151
5.1.2. Prototipo 2: Conexión de la AIDP con Beacon . . . . .	184
5.1.2.1. Análisis . . . . .	184
5.1.2.2. Diseño . . . . .	184
5.1.3. Prototipo 3: Integración de módulos del sistema . . . . .	188
5.1.3.1. Análisis . . . . .	188
5.1.3.2. Diseño . . . . .	199
5.1.4. Prototipo 4: Integración de módulos restantes del sistema . . . . .	245
5.1.4.1. Análisis . . . . .	245
5.1.4.2. Diseño . . . . .	253
5.2. Aplicación Interactiva para el Personal de Ventas (AIPV) . . . . .	258
5.2.1. Prototipo 1: Diseño inicial de la aplicación y descubrimiento de Beacons . . . . .	258
5.2.1.1. Análisis . . . . .	258
5.2.1.2. Diseño . . . . .	263
5.2.2. Prototipo 2: Integración con Apache Kafka . . . . .	274

5.2.2.1. Análisis . . . . .	274
5.2.2.2. Diseño . . . . .	281
5.2.3. Prototipo 3: Integración con servidor (Detalles y estadísticas de clientes y envío de recomendaciones) . . . . .	286
5.2.3.1. Análisis . . . . .	287
5.2.3.2. Diseño . . . . .	291
5.2.4. Prototipo 4: Prototipo final . . . . .	300
5.2.4.1. Análisis . . . . .	300
5.2.4.2. Diseño . . . . .	302
<b>6. Servidores</b>	<b>320</b>
6.1. NodeJS con Express . . . . .	320
6.1.1. Panel de Administración (PA) . . . . .	321
6.1.2. Prototipo 1: Diseño inicial del Panel de Administración . . . . .	321
6.1.2.1. Análisis . . . . .	321
6.1.2.2. Diseño . . . . .	321
6.1.3. Prototipo 2: Monitor geográfico de beacons . . . . .	323
6.1.3.1. Análisis . . . . .	323
6.1.3.2. Diseño . . . . .	325
6.1.4. Prototipo 3: Integración con servicios REST a panel de administración . . . . .	330
6.1.4.1. Análisis . . . . .	330
6.1.4.2. Diseño . . . . .	332
6.1.5. Prototipo 4: Integración de pantalla para ejecutar algoritmo k-means . . . . .	345
6.1.5.1. Análisis . . . . .	345
6.1.5.2. Diseño . . . . .	347
6.1.5.3. Pruebas . . . . .	354
6.1.5.4. Conclusiones . . . . .	358
6.2. Tornado . . . . .	358
6.2.1. Sistema de gestión, procesamiento y proveedor de datos de Retail . . . . .	358
6.2.2. Prototipo 1: Desarrollo de servicios REST . . . . .	358
6.2.2.1. Análisis . . . . .	358
6.2.2.2. Diseño . . . . .	360
6.2.3. Prototipo 2: Integración de algoritmo de agrupamiento K-medias . . . . .	384
6.2.3.1. Análisis . . . . .	384
6.2.3.2. Diseño . . . . .	385
6.2.4. Prototipo 3: Integración de sistema de recomendación basado en filtrado colaborativo primera versión . . . . .	386
6.2.4.1. Análisis . . . . .	387
6.2.4.2. Diseño . . . . .	390
6.2.4.3. Pruebas . . . . .	391
6.2.4.4. Conclusiones . . . . .	398
<b>7. Trabajo a futuro</b>	<b>399</b>
7.1. Aplicación Interactiva Difusora de Productos . . . . .	399
7.2. Aplicación Interactiva para el Personal de Ventas . . . . .	399
7.3. Panel de Administración . . . . .	399
7.4. Sistema de Recomendaciones . . . . .	399
<b>8. Glosario de términos</b>	<b>400</b>
8.0.1. Beacons . . . . .	400
8.0.1.1. Firmware . . . . .	400
8.0.1.2. Ventajas . . . . .	401
8.0.2. Bluetooth Low Energy (BLE) . . . . .	402

---

<b>9. Acrónimos</b>	<b>403</b>
<b>10. Anexos</b>	<b>413</b>

---

## Índice de figuras

---

1.1. Aplicación de Macy's "Walk in and Win"[7]. . . . .	18
1.2. Aplicación "Smart Shopping"de Carrefour trabajando sobre una tablet "Samsung"[9]. . . . .	18
1.3. Tienda de ropa American Eagle Outfitters [11]. . . . .	19
1.4. Metodología de Prototipado Evolutivo [22]. . . . .	24
2.1. Diagrama de componentes básicos del problema de aprendizaje [23]. . . . .	27
2.2. Ejemplo de agrupamiento con K-means [25]. . . . .	31
2.3. Diagrama de una red neuronal. . . . .	33
2.4. Función de clasificación lineal $\theta_1x_1 + \theta_2x_2 = -b$ . . . . .	34
2.5. Proceso de filtrado colaborativo [31]. . . . .	36
3.1. Arquitectura interna del sistema. . . . .	67
3.2. Diagrama de casos de uso general. . . . .	69
4.1. Modelo relacional de base de datos (Visualización completa). . . . .	79
4.2. Modelo relacional de base de datos (Parte 1). . . . .	81
4.3. Modelo relacional de base de datos (Parte 2). . . . .	83
4.4. Prototipo 1: Diagrama del modelo de la base de datos (Visualización completa). . . . .	101
4.5. Prototipo 1: Diagrama del modelo de la base de datos (Parte 1). . . . .	102
4.6. Prototipo 1: Diagrama del modelo de la base de datos (Parte 2). . . . .	103
4.7. Prototipo 1.1: Diagrama del modelo de la base de datos (Visualización completa). . . . .	108
4.8. Prototipo 1.1: Diagrama del modelo de la base de datos (Parte 1). . . . .	109
4.9. Prototipo 1.1: Diagrama del modelo de la base de datos (Parte 2). . . . .	110
4.10. Prototipo 1.2: Diagrama del modelo de la base de datos (Visualización completa). . . . .	112
4.11. Prototipo 1.2: Diagrama del modelo de la base de datos (Parte 1). . . . .	113
4.12. Prototipo 1.2: Diagrama del modelo de la base de datos (Parte 2). . . . .	114
4.13. Prototipo 1.3: Diagrama del modelo de la base de datos (Visualización completa). . . . .	116
4.14. Prototipo 1.3: Diagrama del modelo de la base de datos (Parte 1). . . . .	117
4.15. Prototipo 1.3: Diagrama del modelo de la base de datos (Parte 2). . . . .	118
4.16. Prototipo 1.4: Diagrama del modelo de la base de datos (Visualización completa). . . . .	124
4.17. Prototipo 1.4: Diagrama del modelo de la base de datos (Parte 1). . . . .	125
4.18. Prototipo 1.4: Diagrama del modelo de la base de datos (Parte 2). . . . .	126
4.19. Diagrama de casos de uso módulo Generador de Registros Artificiales. . . . .	127
4.20. Diagrama de clases de módulo Generador de Registros Artificiales. . . . .	128
4.21. Diagrama de secuencia para generar persona. . . . .	129

4.22. Diagrama de secuencia para generar dirección.	130
4.23. Diagrama de secuencia para generar tienda.	131
4.24. Diagrama de secuencia para generar departamento.	132
4.25. Diagrama de secuencia para generar producto.	134
4.26. Diagrama de secuencia para generar imágenes pertenecientes a productos.	135
4.27. Diagrama de secuencia para asignar imágenes a productos.	136
4.28. Diagrama de secuencia para generar empleados.	137
4.29. Diagrama de secuencia para generar clientes.	138
4.30. Diagrama de secuencia para asignar productos favoritos a cliente	140
4.31. UIRA: Menú	141
4.32. Diagrama de casos de uso módulo Generador de Registros Artificiales.	142
4.33. Diagrama de clases de módulo Generador de Registros Artificiales.	143
4.34. Diagrama de secuencia para generar compras a clientes.	145
4.35. Diagrama de secuencia para asignar logros y nivel a clientes.	146
4.36. UIRA: Menú	147
5.1. Arquitectura del módulo AIDP.	149
5.2. Casos de uso de la Aplicación Interactiva Difusora de Productos.	150
5.3. Diagrama de clases (Visualización completa).	151
5.4. Diagrama de clases (Parte 1).	152
5.5. Diagrama de clases (Parte 2).	153
5.6. Diagrama de clases (Parte 3).	154
5.7. Flujo de navegación de la Aplicación Interactiva Difusora de Productos (Visualización completa).	155
5.8. Flujo de navegación de la Aplicación Interactiva Difusora de Productos (Parte 1).	156
5.9. Flujo de navegación de la Aplicación Interactiva Difusora de Productos (Parte 2).	157
5.10. Flujo de navegación de los métodos para iniciar sesión (Visualización completa).	158
5.11. Flujo de navegación de la Aplicación Interactiva Difusora de Productos (Visualización completa).	159
5.12. Flujo de navegación de la Aplicación Interactiva Difusora de Productos (Parte 1).	160
5.13. Flujo de navegación de la Aplicación Interactiva Difusora de Productos (Parte 2).	161
5.14. UIAIDP1 Pantalla de inicio.	162
5.15. UIAIDP2 Log in de la aplicación.	163
5.16. UIAIDP2.1 Log in mediante una cuenta previa.	164
5.17. UIAIDP2.2 Log in mediante una cuenta nueva.	165
5.18. UIAIDP2.3 Log in mediante la creación de una cuenta nueva en Facebook.	166
5.19. UIAIDP2.4 Log in solicitado por la API de Facebook.	167
5.20. UIAIDP3 Visualizar folletos.	168
5.21. UIAIDP3.1 Detectar Beacons.	169
5.22. UIAIDP4 Visualizar menú de opciones.	170
5.23. UIAIDP5 Logros.	171
5.24. UIAIDP6 Promociones.	172
5.25. UIAIDP7 Favoritos.	173
5.26. UIAIDP8 Recomendaciones.	174
5.27. UIAIDP9 Búsqueda de productos.	175
5.28. UIAIDP10 Ver perfil.	176
5.29. UIAIDP10.1 Seleccionar sexo.	177
5.30. UIAIDP10.4 Seleccionar estado civil.	178
5.31. UIAIDP10.2 Seleccionar fecha de nacimiento.	179
5.32. UIAIDP11 Gustos.	180
5.33. UIAIDP12 Beneficios.	181
5.34. UIAIDP13 Proporcionar permisos.	182

5.35. UIAIDP14 Cerrar sesión . . . . .	183
5.36. Diagrama de secuencia para detectar Beacons (Visualización completa). . . . .	184
5.37. Diagrama de secuencia para detectar Beacons (Visualización completa). . . . .	185
5.38. Diagrama de secuencia para detectar Beacons (Parte 1). . . . .	186
5.39. Diagrama de secuencia para detectar Beacons (Parte 2). . . . .	187
5.40. Casos de uso de la Aplicación Interactiva Difusora de Productos. . . . .	189
5.41. Clases POJO (Parte 1). . . . .	190
5.42. Clases POJO (Parte 2). . . . .	191
5.43. Clases POJO (Parte 3). . . . .	192
5.44. Clases POJO (Parte 4). . . . .	192
5.45. Diagrama de clases del prototipo 2 de la AIDP (Visualización completa). . . . .	193
5.46. Diagrama de clases del prototipo 2 de la AIDP (Parte 1). . . . .	193
5.47. Diagrama de clases del prototipo 2 de la AIDP (Parte 2) . . . . .	194
5.48. Diagrama de clases del prototipo 2 de la AIDP (Visualización completa). . . . .	195
5.49. Diagrama de clases del prototipo 2 de la AIDP (Visualización completa). . . . .	196
5.50. Diagrama de clases del prototipo 2 de la AIDP (Visualización completa). . . . .	197
5.51. Diagrama de clases del prototipo 2 de la AIDP (Parte 1). . . . .	198
5.52. Diagrama de clases del prototipo 2 de la AIDP (Parte 2). . . . .	198
5.53. UIAIDP15 Registrar cuenta nueva. . . . .	200
5.54. UIAIDP16 Recuperar contraseña. . . . .	201
5.55. UIAIDP17 Muestra información de productos. . . . .	202
5.56. UIAIDP18 Eliminar producto de favoritos. . . . .	203
5.57. Diagrama de secuencia para iniciar sesión (Visualización completa). . . . .	204
5.58. Diagrama de secuencia para iniciar sesión (Parte 1). . . . .	205
5.59. Diagrama de secuencia para iniciar sesión (Parte 2). . . . .	206
5.60. Diagrama de secuencia para recuperar contraseña (Visualización completa). . . . .	207
5.61. Diagrama de secuencia para recuperar contraseña (Parte 1). . . . .	208
5.62. Diagrama de secuencia para recuperar contraseña (Parte 2). . . . .	209
5.63. Diagrama de secuencia para registrar cuenta nueva (Visualización completa). . . . .	210
5.64. Diagrama de secuencia para registrar cuenta nueva (Parte 1). . . . .	211
5.65. Diagrama de secuencia para registrar cuenta nueva (Parte 2). . . . .	212
5.66. Diagrama de secuencia para consultar folletos (Visualización completa). . . . .	213
5.67. Diagrama de secuencia para consultar folletos (Parte 1). . . . .	214
5.68. Diagrama de secuencia para consultar folletos (Parte 2). . . . .	215
5.69. Diagrama de secuencia para visualizar menú de opciones (Visualización completa). . . . .	216
5.70. Diagrama de secuencia para visualizar menú de opciones (Parte 1). . . . .	217
5.71. Diagrama de secuencia para visualizar menú de opciones (Parte 2). . . . .	218
5.72. Diagrama de secuencia para consultar logros (Visualización completa). . . . .	219
5.73. Diagrama de secuencia para consultar logros (Parte 1). . . . .	220
5.74. Diagrama de secuencia para consultar logros (Parte 2). . . . .	221
5.75. Diagrama de secuencia para consultar promociones (Visualización completa). . . . .	222
5.76. Diagrama de secuencia para visualizar favoritos (Visualización completa). . . . .	223
5.77. Diagrama de secuencia para visualizar favoritos (Parte 1). . . . .	224
5.78. Diagrama de secuencia para visualizar favoritos (Parte 2). . . . .	225
5.79. Diagrama de secuencia para añadir o eliminar productos a favoritos (Visualización completa). . . . .	226
5.80. Diagrama de secuencia para consultar recomendaciones (Visualización completa). . . . .	227
5.81. Diagrama de secuencia para consultar recomendaciones (Parte 1). . . . .	228
5.82. Diagrama de secuencia para consultar recomendaciones (Parte 2). . . . .	229
5.83. Diagrama de secuencia para buscar productos (Visualización completa). . . . .	230
5.84. Diagrama de secuencia para buscar productos (Parte 1). . . . .	231
5.85. Diagrama de secuencia para buscar productos (Parte 2). . . . .	232
5.86. Diagrama de secuencia para consultar datos personales (Visualización completa). . . . .	233

5.87. Diagrama de secuencia para consultar datos personales (Parte 1) . . . . .	234
5.88. Diagrama de secuencia para consultar datos personales (Parte 2) . . . . .	235
5.89. Diagrama de secuencia para actualizar gustos genéricos (Visualización completa). . . . .	236
5.90. Diagrama de secuencia para actualizar gustos genéricos (Parte 1) . . . . .	237
5.91. Diagrama de secuencia para actualizar gustos genéricos (Parte 2) . . . . .	238
5.92. Diagrama de secuencia para consultar beneficios (Visualización completa). . . . .	239
5.93. Diagrama de secuencia para consultar beneficios (Parte 1) . . . . .	240
5.94. Diagrama de secuencia para consultar beneficios (Parte 2) . . . . .	241
5.95. Diagrama de secuencia para actualizar permisos (Visualización completa). . . . .	242
5.96. Diagrama de secuencia para actualizar permisos (Parte 1) . . . . .	243
5.97. Diagrama de secuencia para actualizar permisos (Parte 2) . . . . .	244
5.98. Diagrama de secuencia para cerrar sesión (Visualización completa). . . . .	245
5.99. Casos de uso de la Aplicación Interactiva Difusora de Productos. . . . .	246
5.100Flujo de navegación de la Aplicación Interactiva Difusora de Productos (Visualización completa). . . . .	247
5.101Flujo de navegación de la Aplicación Interactiva Difusora de Productos (Parte 1) . . . . .	248
5.102Flujo de navegación de la Aplicación Interactiva Difusora de Productos (Parte 2) . . . . .	249
5.103Flujo derivado de navegación de la Aplicación Interactiva Difusora de Productos (Visualización completa). . . . .	250
5.104Flujo derivado de navegación de la Aplicación Interactiva Difusora de Productos (Parte 1) . . . . .	251
5.105Flujo derivado de navegación de la Aplicación Interactiva Difusora de Productos (Parte 2) . . . . .	252
5.106UIAIDP19 Solicitar apoyo a vendedor. . . . .	253
5.107Diagrama de secuencia de solicitar apoyo a vendedor (Visualización completa). . . . .	254
5.108Diagrama de secuencia de inicio de sesión con cuenta de Sapphire (Visualización completa). . . . .	255
5.109Diagrama de secuencia de inicio de sesión con cuenta de Sapphire (Parte 1) . . . . .	256
5.110Diagrama de secuencia de inicio de sesión con cuenta de Sapphire (Parte 2) . . . . .	257
5.111Arquitectura del módulo AIPV. . . . .	258
5.112Casos de uso de la AIPV. . . . .	259
5.113Diagrama de clases del prototipo 1 de la AIPV (Visualización completa). . . . .	260
5.114Diagrama de clases del prototipo 1 de la AIPV (Parte 1) . . . . .	261
5.115Diagrama de clases del prototipo 1 de la AIPV (Parte 2) . . . . .	262
5.116Diagrama de clases del prototipo 1 de la AIPV (Parte 3) . . . . .	263
5.117Diagrama de secuencia para el inicio de sesión de un vendedor. . . . .	264
5.118Diagrama de secuencia para ver una introducción a la aplicación. . . . .	265
5.119Diagrama de secuencia para enviar la ubicación de Beacons (Visualización completa). . . . .	265
5.120Diagrama de secuencia para enviar la ubicación de Beacons (Parte 1) . . . . .	266
5.121Diagrama de secuencia para enviar la ubicación de Beacons (Parte 2) . . . . .	267
5.122Flujo de navegación de la AIPV. . . . .	268
5.123UIAPV1 Login. . . . .	268
5.124UIAPV2 Introducción a la aplicación. . . . .	269
5.125UIAPV3 Home: Clientes cercanos. . . . .	270
5.126UIAPV4 Editar perfil. . . . .	271
5.127UIAPV5 Detalles del cliente. . . . .	272
5.128UIAPV6 Enviar recomendaciones al cliente. . . . .	273
5.129UIAPV7 Estadísticas del cliente. . . . .	274
5.130Diagrama de clases del prototipo 2 de la AIPV (Visualización completa). . . . .	275
5.131Diagrama de clases del prototipo 2 de la AIPV (Parte 1) . . . . .	276
5.132Diagrama de clases del prototipo 2 de la AIPV (Parte 2) . . . . .	278
5.133Diagrama de clases del prototipo 2 de la AIPV (Parte 3) . . . . .	280
5.134Diagrama de secuencia para editar perfil de un vendedor. . . . .	281
5.135Diagrama de secuencia para editar perfil de un vendedor (Parte 1) . . . . .	282
5.136Diagrama de secuencia para editar perfil de un vendedor (Parte 2) . . . . .	283

5.137	Diagrama de secuencia para ver a los clientes cercanos en la aplicación.	284
5.138	Diagrama de secuencia para ver a los clientes cercanos en la aplicación (Parte 1).	285
5.139	Diagrama de secuencia para ver a los clientes cercanos en la aplicación (Parte 2).	286
5.140	Diagrama de clases del prototipo 3 de la AIPV (Visualización completa).	287
5.141	Diagrama de clases del prototipo 3 de la AIPV (Parte 1).	288
5.142	Diagrama de clases del prototipo 3 de la AIPV (Parte 2).	290
5.143	Diagrama de clases del prototipo 3 de la AIPV (Parte 3).	291
5.144	Diagrama de secuencia para obtener estadísticas de un cliente.	292
5.145	Diagrama de secuencia para obtener estadísticas de un cliente (Parte 1).	293
5.146	Diagrama de secuencia para obtener estadísticas de un cliente (Parte 2).	294
5.147	Diagrama de secuencia para obtener recomendaciones para un cliente.	295
5.148	Diagrama de secuencia para obtener recomendaciones para un cliente (Parte 1).	296
5.149	Diagrama de secuencia para obtener recomendaciones para un cliente (Parte 2).	297
5.150	Diagrama de secuencia para enviar recomendaciones a un cliente.	298
5.151	Diagrama de secuencia para enviar recomendaciones a un cliente (Parte 1).	298
5.152	Diagrama de secuencia para enviar recomendaciones a un cliente (Parte 2).	299
5.153	Casos de uso de la AIPV.	301
5.154	Diagrama de secuencia para actualizar el token de FCM.	302
5.155	Diagrama de secuencia para actualizar el token de FCM (Parte 1).	303
5.156	Diagrama de secuencia para actualizar el token de FCM (Parte 2).	304
5.157	Diagrama de secuencia para actualizar el token de FCM (Parte 3).	305
5.158	Flujo final de navegación de la AIPV.	306
5.159	UIAPVF1 Login.	307
5.160	UIAPVF2 Introducción a la aplicación.	308
5.161	Introducción a la aplicación - Ver clientes cercanos	309
5.162	Introducción a la aplicación - Perfil de compra de los clientes	309
5.163	Introducción a la aplicación - Más sobre el cliente	310
5.164	Introducción a la aplicación - Editar perfil	310
5.165	UIAPVF3 Home: Clientes cercanos (Vacia).	311
5.166	UIAPVF3-1 Home: Clientes cercanos .	312
5.167	NAPV1 Notificación clientes cercanos.	312
5.168	UIAPVF4 Editar perfil.	313
5.169	NAPV2 Notificación confirmación de actualización de datos.	313
5.170	UIAPVF5 Detalles del cliente.	314
5.171	UIAPVF5-2 Detalles del cliente: Beneficios y recomendaciones.	315
5.172	UIAPVF5-3 Detalles del cliente: Productos favoritos.	316
5.173	NAPV3 Notificación detalles del cliente.	316
5.174	NAPV4 Notificación recomendaciones para el cliente.	316
5.175	UIAPVF6 Enviar recomendaciones al cliente.	317
5.176	NAPV5 Notificación recomendación en proceso de envío.	318
5.177	NAPV6 Notificación recomendación enviada.	318
5.178	UIAPVF7 Estadísticas del cliente.	318
5.179	UIAPVF8 Categorías más compradas.	319
6.1.	Logo de Node.js. [105].	320
6.2.	Logo de Express. [106].	320
6.3.	Arquitectura del módulo PA.	321
6.4.	Diseño general del PA.	322
6.5.	UIPanel1: Diseño computadora.	322
6.6.	UIPanel2: Diseño tableta.	323
6.7.	Casos de uso PA.	324
6.8.	Diagrama de secuencia del monitor geográfico (Visualización completa).	325

6.9. Diagrama de secuencia del monitor geográfico (Visualización parte uno).	326
6.10. Diagrama de secuencia del monitor geográfico (Visualización parte 2).	327
6.11. Diseño monitor geográfico.	328
6.12. UIPanel3: Monitor Beacons y tiendas computadora.	328
6.13. UIPanel4: Monitor Beacons y tiendas tableta.	329
6.14. Flujo de navegación general Panel de Administración.	330
6.15. Diagrama de casos de uso.	331
6.16. Diagrama de secuencia Iniciar sesión.	332
6.17. Diagrama de secuencia Cerrar sesión.	333
6.18. Diagrama de secuencia Visualizar anuncios publicados (Visualización completa).	334
6.19. Diagrama de secuencia Visualizar anuncios publicados (Parte uno).	335
6.20. Diagrama de secuencia Visualizar anuncios publicados (Parte dos).	335
6.21. Diagrama de secuencia Visualizar anuncios publicados (Parte tres).	335
6.22. Diagrama de secuencia Visualizar anuncios publicados (Parte cuatro).	336
6.23. Diagrama de secuencia Visualizar Beacons registrados (Visualización completa).	337
6.24. Diagrama de secuencia Visualizar Beacons registrados (Visualización completa).	338
6.25. Diagrama de secuencia Visualizar Beacons registrados (Parte uno).	339
6.26. Diagrama de secuencia Visualizar Beacons registrados (Parte dos).	339
6.27. UIPanel31: Iniciar sesión.	340
6.28. UIPanel32: Anuncios publicados.	340
6.29. UIPanel33: Información específica de un anuncio.	341
6.30. UIPanel34: Crear nuevo anuncio.	341
6.31. UIPanel35: Visualizar Beacons.	342
6.32. UIPanel36: Información de Beacon específica.	342
6.33. Flujo de navegación general Panel de Administración.	343
6.34. Flujo de navegación de pantalla Beacons registrados.	344
6.35. Flujo de navegación de pantalla Anuncios publicados.	345
6.36. Casos de uso PA.	346
6.37. Diagrama de secuencia Ejecutar algoritmo de agrupamiento (Visualización completa).	348
6.38. Diagrama de secuencia Ejecutar algoritmo de agrupamiento (Parte uno).	349
6.39. Diagrama de secuencia Ejecutar algoritmo de agrupamiento (Parte dos).	350
6.40. UIPanel37: Ejecutar algoritmo de agrupamiento (Parte uno).	351
6.41. UIPanel38: Ejecutar algoritmo de agrupamiento (Parte dos).	351
6.42. Flujo de navegación general Panel de Administración.	352
6.43. Flujo de navegación de pantalla Beacons registrados.	353
6.44. Flujo de navegación de pantalla Anuncios publicados.	354
6.45. Prueba 1.	355
6.46. Prueba 1 características de cada cluster.	355
6.47. Prueba 2.	356
6.48. Prueba 2 características de cada cluster.	356
6.49. Prueba 3.	357
6.50. Prueba 3 características de cada cluster (Parte uno).	357
6.51. Prueba 3 características de cada cluster (Parte dos).	358
6.52. Logo de Tornado. [104]	358
6.53. Diagrama de flujo de algoritmo Filtrado Colaborativo.	388
6.54. Gradiante descendiente.	389
6.55. Prueba 1 con validación.	392
6.56. Prueba 1: J respecto al número de iteraciones.	393
6.57. Prueba 2 con validación.	394
6.58. Prueba 2: J respecto al número de iteraciones.	395
6.59. Prueba 3 con validación.	396
6.60. Prueba 3: J respecto al número de iteraciones.	397

---

8.1. Arquitectura interna del Beacon [99]. . . . .	400
10.1. Gráfica 1. . . . .	413
10.2. Gráfica 2. . . . .	413
10.3. Gráfica 3. . . . .	414
10.4. Gráfica 4. . . . .	414
10.5. Gráfica 5. . . . .	415

---

## Índice de cuadros

---

1.1. Comparación entre sistemas de venta en línea y sistemas de venta tradicional [1][2]. . . . .	16
1.2. Comparación entre sistemas desarrollados para tiendas comerciales. . . . .	20
1.3. Comparación entre Trabajos Terminales de ESCOM. . . . .	21
2.1. Comparación entre Beacons “Estimote” disponibles [72]. . . . .	49
2.2. Cuadro comparativo entre iBeacon y Eddystone [73] [74]. . . . .	50
3.1. Cuadro de costos de equipo. . . . .	56
3.2. Cuadro de costos de servicios. . . . .	57
3.3. Cuadro de salarios para empleados requeridos en el proyecto. . . . .	57
3.4. Cuadro de costo total de la aplicación. . . . .	57
3.5. Comparación de sistemas gestores de bases de datos [61] [62]. . . . .	58
3.6. Comparación de sistemas operativos móviles [63] - [67]. . . . .	59
3.7. Comparación de lenguajes de programación ocupados en servidores [68] - [71]. . . . .	60
3.8. Riesgos de negocio que presenta el proyecto. . . . .	62
3.9. Riesgos de negocio que presenta el proyecto. . . . .	63
3.10. Riesgos de proyecto que presenta el proyecto. . . . .	64
3.11. Riesgos técnicos que presenta el proyecto. . . . .	66
3.12. Requerimientos Funcionales del Generador de Registros Artificiales. . . . .	70
3.13. Requerimientos Funcionales de la Aplicación Interactiva Difusora de Productos. . . . .	73
3.14. Requerimientos Funcionales de la Aplicación Interactiva para el Personal de Ventas. . . . .	74
3.15. Requerimientos Funcionales del Panel de Administración. . . . .	75
3.16. Requerimientos Funcionales del Sistema de Gestión, Procesamiento y Proveedor de datos de Retail. . . . .	77
3.17. Requerimientos No Funcionales del sistema. . . . .	77
4.1. Tabla de diccionario de datos Anuncio. . . . .	84
4.2. Tabla de diccionario de datos Atributo. . . . .	84
4.3. Tabla de diccionario de datos Beneficio. . . . .	85
4.4. Tabla de diccionario de datos Categoría. . . . .	85
4.5. Tabla de diccionario de datos Cliente. . . . .	86
4.6. Tabla de diccionario de datos Compra. . . . .	87
4.7. Tabla de diccionario de datos Departamento_anuncio. . . . .	88
4.8. Tabla de diccionario de datos Departamento_beacon. . . . .	88
4.9. Tabla de diccionario de datos Departamento. . . . .	89

4.10. Tabla de diccionario de datos Dirección_persona. . . . .	89
4.11. Tabla de diccionario de datos Empleado. . . . .	90
4.12. Tabla de diccionario de datos Favorito. . . . .	90
4.13. Tabla de diccionario de datos Imagen_producto. . . . .	91
4.14. Tabla de diccionario de datos Log_inventario. . . . .	91
4.15. Tabla de diccionario de datos Logro_cliente. . . . .	92
4.16. Tabla de diccionario de datos Logro. . . . .	92
4.17. Tabla de diccionario de datos Marca. . . . .	93
4.18. Tabla de diccionario de datos Método_pago_compra. . . . .	93
4.19. Tabla de diccionario de datos Método_pago. . . . .	94
4.20. Tabla de diccionario de datos Nivel_beneficio. . . . .	94
4.21. Tabla de diccionario de datos Nivel. . . . .	95
4.22. Tabla de diccionario de datos Persona. . . . .	96
4.23. Tabla de diccionario de datos Producto_anuncio. . . . .	97
4.24. Tabla de diccionario de datos Producto_atributo. . . . .	97
4.25. Tabla de diccionario de datos Producto_imagen. . . . .	97
4.26. Tabla de diccionario de datos Producto. . . . .	98
4.27. Tabla de diccionario de datos Promoción. . . . .	99
4.28. Tabla de diccionario de datos Tienda. . . . .	100
4.29. Tabla de diccionario de datos Tipo_atributo. . . . .	100
4.30. Tabla de diccionario de datos Empleado del prototipo 1.1 . . . . .	104
4.31. Tabla de diccionario de datos Dirección del prototipo 1.2. . . . .	105
4.32. Tabla de diccionario de datos Promoción del prototipo 1.1 . . . . .	106
4.33. Tabla de diccionario de datos Tienda del prototipo 1.1 . . . . .	107
4.34. Tabla de diccionario de datos Tienda.categoría del prototipo 1.1. . . . .	107
4.35. Tabla de diccionario de datos Dirección del prototipo 1.2. . . . .	111
4.36. Tabla de diccionario de datos Departamento_beacon. . . . .	115
4.37. Tabla de diccionario de datos Persona. . . . .	120
4.38. Tabla de diccionario de datos Cliente. . . . .	122
4.39. Tabla de diccionario de datos Producto. . . . .	123
5.1. Requerimientos añadidos a los Requerimientos Funcionales de la Aplicación Interactiva Difusora de Productos. . . . .	188
5.2. Requerimiento añadido a los Requerimientos Funcionales de la Aplicación Interactiva Difusora de Productos. . . . .	245
5.3. Requerimiento añadido a los Requerimientos Funcionales de la Aplicación Interactiva para el Personal de Ventas. . . . .	300

# CAPÍTULO 1

---

## Introducción

---

En el presente documento se expone el desarrollo del Trabajo Terminal “Sistema de recomendaciones basado en aprendizaje máquina y aplicación móvil conectada con emisores de señal Bluetooth Beacon (Sapphire)” con número 2017-B034. Expone el análisis de las problemáticas que dieron origen a esta idea, así como la solución que fue propuesta y entorno a la cual se desarrolla actualmente el proyecto, teniéndose en cuenta el alcance del proyecto con respecto a los tiempos de ejecución y los riesgos que la realización de este conlleva.

Dentro de la propuesta de solución a los problemas identificados, se presenta la justificación a dicho trabajo y los objetivos que se desean cumplir. Posteriormente, dentro del marco teórico se explican de forma más precisa ciertos subcampos, algoritmos y técnicas de las ciencias de la computación, así como los dispositivos y las bibliotecas que serán utilizadas.

En el apartado del análisis encontramos el análisis de factibilidad con el cual se pretende evaluar si se cuenta con la tecnología necesaria para el desarrollo del sistema y el costo total que este tendría. Así mismo, se analizan los requerimientos tanto funcionales como los no funcionales y de igual forma, son analizados y se priorizados los riesgos que se localizan.

Finalmente, se muestran una serie de diagramas y modelos que explican de manera gráfica el comportamiento del sistema y la arquitectura de este.

Este documento va dirigido a toda la comunidad politécnica y externos a ésta a fin de proveer una nueva fuente de información que permita crear o innovar la manera en que se les proporciona la información publicitaria a los clientes dentro de un enfoque comercial.

Realizado por:  
Castro Flores Marcela  
Galindo García Adrián Eduardo  
Murga Dionicio Rubén Adolfo

Escuela Superior de Cómputo - Instituto Politécnico Nacional

## 1.1. Contexto

En los últimos años, internet se ha poblado con miles de plataformas tales como: Amazon, Apple Music, Mercadolibre, Netflix, Twitter, Spotify, entre muchas otras, que incorporan millones de usuarios, los cuales generan infinidad de datos, mismos que a su vez han ocasionado la implementación y la gran popularidad de sistemas que permitan filtrar, priorizar y otorgar información relevante, mejor conocidos como sistemas de recomendaciones. Este tipo de sistemas realizan una búsqueda a través de un gran volumen de información generada dinámicamente para proporcionar contenidos y servicios personalizados, utilizando algoritmos que en su mayoría provienen del campo del aprendizaje máquina. El aprendizaje máquina es un subcampo de la inteligencia artificial que permite a las computadoras aprender de información y datos proporcionados sin programar explícitamente reglas a seguir. Por ejemplo, un sistema de recomendaciones podría ser utilizado en puntos de venta en línea para ofrecer a los clientes sugerencias sobre lo que les gustaría comprar, basándose en su historial de compra y/o búsqueda de productos.

## 1.2. Problemática

Hoy en día, los sistemas de recomendaciones implementados en las tiendas departamentales se han limitado a otorgar beneficios por medio de ventas en línea y en monederos electrónicos, lo cual no tiene una rentabilidad directa con las ventas físicas, además de que, muchas de las recomendaciones que se ofrecen al público son generalizadas lo cual disminuye el interés de los clientes por los productos recomendados. Tomando esto en cuenta, a continuación en el cuadro 1.1, se presenta una comparación entre los sistemas de venta en línea contra los sistemas de venta tradicional.

Sistemas de venta en línea	Sistemas de venta tradicional
El número de clientes a los cuales se puede dirigir un producto es mayor debido a que virtualmente el cliente tiene acceso a cada uno de ellos y puede filtrarlos para encontrar el de su preferencia.	Se obtienen los productos con una satisfacción inmediata, es decir, no se tiene que esperar un determinado plazo de días para que el producto llegue al consumidor, por lo que tampoco se tiene que pagar un costo extra por envío.
<b>Ventajas</b>	
Las recomendaciones que se le presentan al usuario son basadas en productos relacionados con los que se han visto o comprado con anterioridad, por lo tanto son útiles y acertadas.	Se proporciona a los clientes un mejor asesoramiento sobre las especificaciones de los productos, ya que en las tiendas, el cliente puede preguntar lo que deseé y tener una respuesta inmediata.
Permite dar una retroalimentación sobre el producto por parte de los usuarios que compraron dicho producto.	El cliente puede visualizar, incluso hasta probar los productos que está por comprar.
Por medio de un sitio Web, el cliente tiene acceso a cualquier hora del día para consultar y/o adquirir los diferentes productos propuestos.	Contrariamente a lo anterior, una tienda física funciona en un horario definido por lo cual las ventas se ven restringidas a dicha jornada del día.
<b>Limitantes</b>	
Existen personas que no confían al 100 % en la compra de productos en plataformas en línea ya que temen que al proporcionar sus datos personales estos sean utilizados para otros fines.	Aún y cuando los empleados proporcionan atención al cliente, en ocasiones ofrecen mercancía fuera de su interés, lo cual llega a ser molesto.
En ocasiones el comprador necesita visualizar y/o probar el producto antes de estar seguro de realizar su compra.	No permite visualizar opiniones sobre el producto hechas por compradores anteriores.

Cuadro 1.1: Comparación entre sistemas de venta en línea y sistemas de venta tradicional [1][2].

Por otro lado, en años recientes se han creado también muchas aplicaciones que hacen uso de la geolocalización de los usuarios por medio de Internet o del Sistema de Posicionamiento Global (Global Positioning System, por sus siglas en inglés GPS), y su vez se ha dado el crecimiento de unos pequeños dispositivos conocidos como Beacons, los cuales emiten señales de corto alcance por medio de la tecnología Bluetooth de bajo consumo (Bluetooth Low Energy, por sus siglas en inglés BLE). Debido a su corto alcance (5 a 100 metros en promedio) resultan muy efectivos para dicha geolocalización en proximidad. Un claro ejemplo del uso de estos dispositivos está dentro del marketing de proximidad, tanto por la facilidad de puntos de referencia como por los anuncios de ofertas e información de productos que pueden proveer [3].

### 1.3. Estado del arte

Se realizó una investigación para conocer si en la actualidad se han desarrollado aplicaciones similares tanto dentro como fuera del Instituto Politécnico Nacional (IPN). La búsqueda fue realizada tanto en el sitio oficial de la Escuela Superior de Cómputo (ESCOM) y en su biblioteca como en sitios externos a este, con el fin de conocer los trabajos terminales realizados previamente que fueron enfocados al desarrollo de un software similar o que de igual manera utilizaron Beacons para estos. Para obtener una perspectiva teórica sobre los dispositivos que se plantean utilizar y que ya se han utilizado en las aplicaciones encontradas, podemos encontrar la definición de los Beacons en la sección Glosario de Términos.

Se explican los sistemas de recomendaciones que se han realizado con anterioridad, elaborados para un uso comercial dentro de tiendas departamentales y que a su vez, utilizan dichos dispositivos. De igual manera, se explica el funcionamiento de los Trabajos Terminales (TT) desarrollados dentro de la ESCOM que utilizan Beacons y de igual manera desarrollan sistemas de información. Cabe mencionar que los sistemas de ámbito comercial, se desarrollaron por parte de tiendas departamentales de gran renombre como lo es la tienda Macy's de Nueva York y San Francisco, el supermercado Carrefour, muy conocido por todo Europa, la tienda de ropa American Eagle Outfitters, entre otras, las cuales han obtenido grandes beneficios a partir del uso práctico de los Beacons.

#### 1.3.1. Tienda departamental Macy's

Macy's es el primer minorista que ha hecho uso de los iBeacons de Apple desde el año 2013 a la actualidad, a partir de los cuales, envía diferentes tipos de alertas a los usuarios de iPhone con el fin de informarles acerca de las diferentes ofertas y productos que pueden ser de su interés y que claro, se encuentran a la venta en la tienda. Este funcionamiento es realizado por medio de la aplicación conocida como "Shopkick", misma a la cual son redirigidos los compradores al ingresar a Macy's [5]. Esta aplicación que inició como una prueba en el año 2013, en las sucursales de Macy's Herald Square en Nueva York y en Union Square en San Francisco y que se limitaba a hacer un ping a los iPhones de los clientes al momento en que ellos ingresaban a la tienda, se ha expandido hasta el año pasado en el Black Friday, durante la campaña "Walk in and Win", en el cual los clientes obtenían en la aplicación el típico juego de rasca y gana "pero virtual y durante la jornada de compras alrededor de toda la tienda, ellos debían estar atentos pues podían adquirir grandes beneficios como tarjetas de regalo y reuniones con celebridades [6].

La figura 1.1 muestra la interfaz de usuario que la aplicación de Macy's "Walk in and Win" ofrecía a sus consumidores.

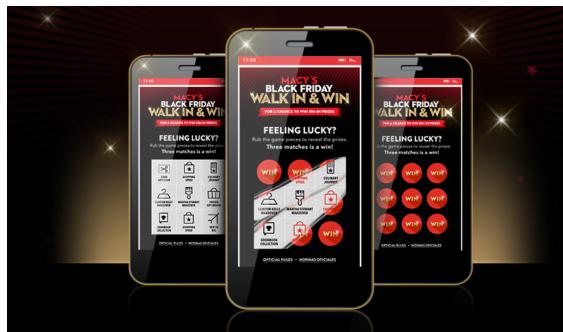


Figura 1.1: Aplicación de Macy's "Walk in and Win"[\[7\]](#).

### 1.3.2. Supermercado Carrefour

En el 2015, el supermercado Carrefour instaló 600 Beacons BLE en sus 28 hipermercados en Rumania para guiar a los compradores a lo largo de toda la tienda y al mismo tiempo, estos recibían promociones y ofertas tanto en sus dispositivos móviles como en las tablets incorporadas al manubrio de los carritos de supermercado. Esta empresa hacía uso de la tecnología Beacon marca Onyx Beacon, misma que se ha encargado de instalar dichos dispositivos a lo largo de autobuses y trolebuses en Bucharest a fin de guiar a los pasajeros con discapacidades visuales. El uso de la aplicación era simple y consistía en indicar la ruta óptima por toda la tienda al cliente, con el fin de que obtuviera todos los artículos que previamente había seleccionado de cada uno de los departamentos disponibles del supermercado. A manera en la que el cliente avanzaba por toda la tienda, este recibía diferentes promociones y notificaciones sobre los productos del departamento en el que se hallaba [\[8\]](#).

La figura 1.2 muestra uno de los carritos del supermercado Carrefour a los cuales se les implantaron las tablets con la aplicación "Smart Shopping" para que el cliente realizará su recorrido dentro de la tienda mientras obtenía a su paso por los diferentes departamentos, promociones y descuentos.

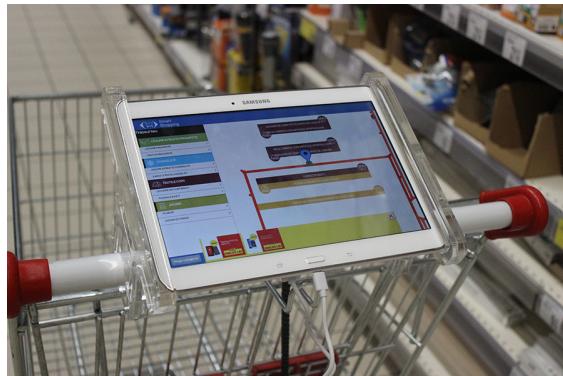


Figura 1.2: Aplicación "Smart Shopping" de Carrefour trabajando sobre una tablet "Samsung"[\[9\]](#).

### 1.3.3. Tienda de ropa American Eagle Outfitters

En el caso de la tienda de ropa American Eagle, tomaron en cuenta que los clientes que entran a los probadores son más propensos a adquirir una prenda de ropa por lo cual, en el año 2014, haciendo uso de la misma aplicación que la tienda Macy's, instaló en 100 de las 929 tiendas de la marca, Beacons en los probadores, mediante los cuales se enviaban ofertas a los usuarios. Para ello, fue importante encontrar que aquellos clientes que recibían una oferta, estaban más predispuestos a probarse otra prenda a diferencia de

---

aquellos que no lo hacían [10].

En la figura 1.3 se presenta una de las sucursales de la tienda de ropa American Eagle Outfitters que instaló los dispositivos Beacons dentro de los probadores de los clientes con el fin de enviarles diferentes descuentos y de este modo, promover el aumento en las ventas de la ropa.



Figura 1.3: Tienda de ropa American Eagle Outfitters [11].

#### 1.3.4. Aplicación makeitapp

Finalmente y una de las aplicaciones que nos pareció muy importante de considerar fue la aplicación makeitapp debido a que esta empresa permite realizar todo el desarrollo de una app personalizada haciendo uso de los Beacons y a su vez, provee de un panel de control al administrador. A primera vista esta aplicación tiene cierta similitud con el TT propuesto ya que realiza el envío de notificaciones que sugieren productos y servicios personalizados, sin embargo, la diferencia que este sistema presenta es que no ofrece al cliente la asistencia directa por parte de un vendedor cuando él/ella ha ingresado a la tienda departamental [12].

Ya que se han descrito con mayor profundidad lo que los sistemas anteriores realizan, se muestra a continuación una comparación entre estos. El cuadro 1.2 se enfoca principalmente en las características que cada uno de los sistemas implementa mismo que a su vez se compara con las características del sistema a desarrollar.

Característica/Sistema	Tienda departamental Macy's	Supermercado Carrefour	Tienda de ropa American Eagle Outfitters	makeitapp	Sapphire
Uso de Beacons	Sí	Sí	Sí	Sí	Sí
Aplicación móvil	Sí	Sí	Sí	Sí	Sí
Atención personalizada	No	No	No	No	Sí
Plataforma Web (Panel de control)	Sí	Sí	Sí	Sí	Sí
Ludificación	Sí	Sí	Sí	Sí	Sí
Recomendaciones personalizadas	No	No	No	No	Sí

Cuadro 1.2: Comparación entre sistemas desarrollados para tiendas comerciales.

Ahora proseguimos con el análisis de los TT's realizados en la ESCOM entre los cuales obtuvimos 2 que son los que más se asemejan al tema que se busca desarrollar.

### 1.3.5. TT 2015-A056 API para el desarrollo de sistemas de recomendación.

Lo que este Trabajo Terminal proponía era la realización de una Interfaz de Aplicación de Programación (Application Programming Interface, por sus siglas en inglés API) que permitía desarrollar diferentes sistemas de recomendación con el fin de brindar diferentes funciones de abstracción, clasificación y análisis de los datos recopilados. Se centraban en el caso de estudio de un restaurante con diferentes platillos en una zona geográfica en particular [13].

### 1.3.6. TT 2010-0063 Trabajo Terminal Sistema generador de recomendaciones para una tienda en-línea de videojuegos.

Este trabajo consistió en una plataforma Web de información, enfocada a la venta de artículos de entretenimiento haciendo uso de un sistema de recomendaciones con el fin de mejorar las compras de sus usuarios [14].

Posteriormente, en el cuadro 1.3 se muestra la comparación entre el TT propuesto y aquellos que se han realizado con anterioridad dentro de la ESCOM.

Característica/Sistema	TT 2015-A056	TT 2010-0063	Sapphire
Uso de Beacons	No	No	Sí
Aplicación móvil	Sí	No	Sí
Atención personalizada	No	No	Sí
Plataforma Web (Panel de control)	Sí	Sí	Sí
Ludificación	No	No	Sí
Recomendaciones personalizadas	Sí	Sí	Sí

Cuadro 1.3: Comparación entre Trabajos Terminales de ESCOM.

## 1.4. Solución propuesta

Considerando todo lo anterior, proponemos como proyecto de titulación el desarrollo de un sistema compuesto por múltiples subsistemas que permitan atacar algunas de las limitantes de los sistemas de venta tradicional, los cuales son:

- Un módulo de gestión, procesamiento y proveedor de datos de Retail que funciona como interfaz de conexión con dos módulos basados en aprendizaje máquina el cual haciendo uso de la información relacionada con las compras anteriormente registradas, las preferencias y características del usuario formulará una serie de recomendaciones de productos, las cuales se presentarán por medio de una aplicación móvil al usuario final. Por otro lado, sirve como servicios de Transferencia de Estado Representacional (Representational State Transfer, por sus siglas en inglés REST) para la gestión de datos del sistema.
- Se propone una aplicación móvil dirigida hacia los clientes en la cual podrán recibir sugerencias en formato de alertas basadas en las preferencias del usuario y recompensas por usar la aplicación, formadas a partir del sistema de recomendaciones anteriormente mencionado.
- Por otro lado, una aplicación hecha específicamente para vendedores, mediante la cual se les proveerá una lista de los clientes dentro de la tienda y se mantendrán las ubicaciones de los Beacons actualizadas; dichos vendedores tendrán la posibilidad de seleccionar un cliente y visualizar productos que podrían ofrecerles personalmente; esto será posible si y sólo si el cliente establece en sus configuraciones ser visible por los vendedores o si solicita atención. Cabe mencionar que de igual manera que en la aplicación del cliente, los productos sugeridos que se le muestren al vendedor para cada uno de los clientes serán obtenidos del sistema de recomendaciones.
- Finalmente, el último módulo considerado es una plataforma Web, la cual fungirá como un panel de administración y herramienta de gestión de información en la que se podrán administrar usuarios, anuncios, promociones, establecimientos, además de ajustar los parámetros del sistema de recomendaciones para mejorar la precisión de las predicciones.

Es importante mencionar que para el desarrollo de la comunicación de la aplicación móvil del cliente se hará uso del Kit de Desarrollo de Software (Software Development Kit, por sus siglas en inglés SDK) de proximidad de Estimote para establecer una comunicación con los Beacons, mismos que nos proporcionarán la ubicación en la que un usuario se estará posicionando dentro de la tienda departamental [15].

Estos sensores serán divididos por departamentos dentro de la tienda y así mismo, se desarrollará un módulo generador de registros artificiales que nos permitirá simular una gran cantidad de datos e ingresarlos al subsistema de recomendaciones para poder verificar la efectividad del mismo.

## 1.5. Justificación

Según datos presentados por el Instituto Nacional de Estadística y Geografía (INEGI) quiénes respecto a lo publicado en el informe “Estadísticas a propósito del día mundial de internet (17 de mayo)” con fecha del 15 de mayo del 2017 informan que el 59.5 % de la población de seis o más años de edad en México se declaró usuario de Internet, mientras que el otro 40.5 % no lo usan y a su vez de ese 59.5 %, el 96.0 % hace uso del mismo de 1 a 7 días de la semana, siendo el otro 4 % la población que no lo utiliza en lo absoluto. Si bien las transacciones electrónicas (compras o pagos vía Internet) no son actividades recurrentes entre los internautas mexicanos, se observa un crecimiento respecto a resultados anteriores, pasando del 12.8 % en el 2015 al 14.7 % en el 2016. Este mismo informe hace mención de más datos relevantes como que el 73.6 % de la población de seis años o más cuenta con un dispositivo móvil y el otro 26.4 % hace referencia a los menores de 6 años que no hacen uso de dicho dispositivo. Asimismo se menciona que tres de cada cuatro de estos usuarios (76.0 %) cuentan con un teléfono inteligente (Smartphone), lo cual significa que este tiene acceso a internet y por lo tanto, el otro 24 % no cuentan con un dicho teléfono celular. Sin embargo sólo el 89 % de los usuarios con un celular inteligente se conectan a internet y el otro 11 % no realizan dicha acción [16].

Por otra parte, debemos considerar los indicadores que las empresas comerciales al por menor (las cuales incluyen a las tiendas de autoservicio y departamentales) están reflejando en estos últimos meses, esto con el fin de conocer los ingresos reales que se están generando por el suministro de bienes y servicios ya que este será el sector al que estaríamos beneficiando; el comunicado de prensa número 296/17 de la INEGI informa que a tasa anual, los ingresos por suministro de bienes y servicios aumentaron 3.4 % mientras que las remuneraciones medias reales se redujeron (-)0.2 % y el personal ocupado (-)0.1 % durante el quinto mes del 2017, esto respecto al mismo mes de un año antes [17].

Los datos presentados anteriormente son importantes para la realización de este proyecto, debido a que este está enfocado a usuarios que cuentan con un dispositivo móvil y que a su vez tienen acceso a una conexión de Internet o a un plan de datos móviles pues deberán obtener la aplicación móvil de nuestro sistema por medio de la cual obtendrán las promociones y los artículos sugeridos, así mismo, es necesario que los clientes cuenten con un Smartphone pues estos son los que tienen una conexión Bluetooth que será la que permitirá realizar la comunicación entre el Beacon y el celular que nos permitirá saber la posición en la que la persona se encuentre.

Una de las múltiples ventajas que este trabajo terminal posee respecto a proyectos que se han desarrollado en el mercado únicamente, será la gran reducción de papel que se obtendrá a partir de la eliminación de folletos promocionales que se reparten comúnmente dentro de las tiendas departamentales. En un informe de la Universidad Tecnológica de Monterrey se menciona que, en México, se talan medio millón de árboles diariamente con el fin de obtener su pulpa virgen para la producción del papel, si se reciclará ese papel se ahorrarían 28 mil litros de agua y 17 árboles. De igual manera el Instituto Nacional de Ecología declara que México ocupa el tercer sitio en índices anuales de deforestación y un cálculo aproximado de este desperdicio es de [18]:

Medida de 1 hoja con norma del Instituto Alemán de Normalización (Deutsches Institut für Normung, por sus siglas en alemán DIN)-A4:  $297mm \times 210mm = 62370mm^2 = 0,062m^2$   
 1 hoja pesa  $80\frac{g}{m^2} \times 0,062m^2 = 4,8g$   
 500 hojas pesarán  $500g \times 4,8g = 2,400g = 2,4kg$

En adición a las ventajas del sistema es importante recalcar que existen sistemas de recomendación en múltiples plataformas como Amazon y MercadoLibre, pero ninguna de ellas te permite tener un contacto directo con el producto de una manera en la que uno como comprador tome la decisión más completa respecto a adquirir o no el artículo.

Por otra parte, se pretende aplicar en el proyecto el término “Gamification/Ludificación” mismo que es utilizado para referirse a brindar ciertas recompensas a un usuario a cambio de que él, en este caso, se registre en nuestra aplicación. Esta actividad no ha sido implementada a la par del uso de los Beacons, si bien el marketing de proximidad permite enviar promociones a todo aquel que se encuentre dentro del radio de emisión de la señal Bluetooth, no envía sugerencias personalizadas, lo cual implica una característica sobresaliente sobre los sistemas ya existentes.

Por lo planteado en este documento, consideramos que esta es una propuesta valiosa como tema de titulación, para el cual, se utilizarán los conocimientos adquiridos en la carrera de Ing. en Sistemas Computacionales de las unidades de aprendizaje: Redes Neuronales, Desarrollo de Aplicaciones Móviles, Sistemas Web, además de los siguientes cursos y libros:

- Machine Learning, Stanford, Andrew Ng. [19].
- Deep Learning A-Z: Hands-On Artificial Neural Networks [20].
- Machine Learning, Tom M. Mitchell [21].

## 1.6. Objetivos

### 1.6.1. Objetivo general

Diseñar e implementar un sistema de recomendaciones basado en aprendizaje máquina que funcione como el núcleo de un conjunto de aplicaciones móviles y una plataforma Web a fin de generar recomendaciones de productos y/o servicios de forma personalizada a clientes de una tienda departamental.

### 1.6.2. Objetivos específicos

1. Desarrollar el prototipo de una aplicación móvil para clientes en la cual recibirán además de las recomendaciones, anuncios, promociones y notificaciones al encontrarse dentro de la tienda departamental y estar dentro del rango de un Beacon.
2. Crear un módulo encargado de generar registros artificiales que nos ayudarán a simular datos con los cuales podremos entrenar y poner a prueba las recomendaciones que el sistema produzca.
3. Desarrollar el prototipo de una aplicación móvil para los vendedores de una tienda departamental, con la que puedan dar una atención personalizada a los clientes ofreciendo recomendaciones de productos que sean de su interés o simplemente asesorar su decisión de compra.
4. Realizar el desarrollo del prototipo de una plataforma en línea para que los administradores de la tienda departamental puedan gestionar anuncios y promociones, además de ver gráficas sobre las características del mercado que se tiene.
5. Implementar el SDK de proximidad de Estimote para comunicar a los Beacons con las aplicaciones móviles y conocer la ubicación del usuario con el objetivo de dar recomendaciones personalizadas dependiendo de la tienda/departamento donde se encuentre.
6. Desarrollar un prototipo de un servidor de servicios REST para realizar las operaciones de lectura y escritura sobre el repositorio de datos y proveer de recomendaciones a las diferentes aplicaciones móviles.

## 1.7. Metodología

La metodología seleccionada y la que el equipo considera como la más acertada para el desarrollo de este proyecto es el “Prototipado Evolutivo”, mismo que se caracteriza por permitir la construcción de un prototipo inicial al cual se le irán añadiendo diferentes componentes a medida que el proyecto avanza. Esta metodología es funcional en escenarios en los cuales no se conocen por completo los requerimientos, provee al equipo con ventajas como la flexibilidad en los mismos y entregas rápidas de productos funcionales.

Se pretende para cada uno de los prototipos previstos, realizar las etapas de análisis, diseño y pruebas, mismos que se irán actualizando con cada nuevo ciclo como se muestra en la figura 1.4 de la metodología y cada nuevo prototipo realizado. Dichos prototipos continuarán indefinidamente hasta que las expectativas del cliente, el objetivo general y los objetivos específicos sean cubiertos de una manera satisfactoria.

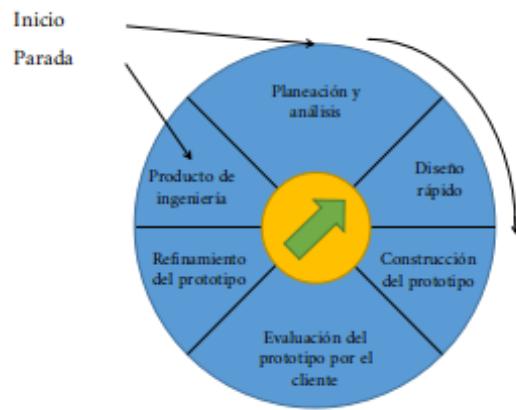


Figura 1.4: Metodología de Prototipado Evolutivo [22].

## 1.8. Beneficios esperados

Como se ha explicado anteriormente, existen sistemas que han sido implementados en diferentes países, sin embargo, es importante recalcar que ninguno de ellos ha sido implementado ni probado en México, por otro lado, el uso y las funciones que los Beacons ofrecen, permiten romper la barrera que existe entre la tecnología y el mundo real. Es por eso que una de las principales metas a realizar es impulsar el uso de esta, así se implementan sistemas probados en países con un desarrollo tecnológico superior, lo que impulsa al desarrollo en el país y de esta forma, como ya se explicó en la justificación, se disminuyen las cantidades de residuos generados por folletos y otras propagandas distribuidas en las tiendas.

## 1.9. Alcance

La implementación de un sistema que permita generar recomendaciones por medio de un esquema de aprendizaje máquina. Dicho sistema debe funcionar sobre aplicaciones móviles y una plataforma Web. La finalidad es generar recomendaciones de productos y/o servicios de forma personalizada a clientes de una tienda departamental.

## 1.10. Limitantes

Entre las principales limitantes a las que el sistema a desarrollar se enfrenta son:

- Si la persona o el negocio no cuenta con servicio de Internet, no existe la comunicación entre los módulos por lo tanto es imposible el envío de recomendaciones a los clientes.
- Debido a los nuevos filtros de seguridad que Facebook ha impuesto, solo nos es posible obtener los datos básicos de un usuario como lo son su nombre, foto de perfil y su correo electrónico, por tal motivo, si dicho usuario no entra a la sección de "Gustos" para marcar las categorías que se relacionen con sus gustos, este no podrá ser clasificado dentro de una categoría y no podrá recibir recomendaciones acertadas.
- El tiempo de recolección de datos puede abarcar más tiempo de lo esperado debido a que cada usuario tiene un manejo diferente de la información dentro de sus redes sociales.
- Debido a no contar con datos reales con los cuales desarrollar este proyecto se desarrolló un generador de registros artificiales para generar datos de prueba. Esto ocasionó que algunos de los productos cuenten con imágenes repetidas, categorías y/o departamentos aleatorios.
- La SDK de Estimote se encuentra en constante actualización (al momento de iniciar el trabajo terminal se encontraba en pruebas beta), por esta razón, pueden ocurrir ciertos fallos o recargas en la pantalla de las aplicaciones móviles, especialmente en la creación de las zonas de proximidad las cuales requieren del Bluetooth del dispositivo.
- La realización de pruebas en dispositivos tales como Huawei P20, Xperia XA Ultra, Samsung Galaxy A5 y Xiaomi Redmi 4, nos permitió visualizar ciertos fallos ocurridos en únicamente algunos dispositivos. Por dicha razón, es importante considerar que dependiendo el hardware en el cual se hayan instalado las aplicaciones móviles, pueden ocurrir fallos referidos a la utilización del Bluetooth o las API's implementadas.

# CAPÍTULO 2

---

## Marco teórico

---

En este capítulo se abarcan los términos y tecnologías que serán utilizadas a lo largo del desarrollo del Trabajo Terminal, a fin de brindar un contexto más amplio para facilitar la comprensión del lector.

### 2.1. Aprendizaje máquina

El aprendizaje máquina es un subcampo de Inteligencia Artificial (Artificial Intelligence, por sus siglas en inglés AI) que proporciona a los sistemas la capacidad de aprender y mejorar automáticamente a partir de la experiencia (registros almacenados en un repositorio de datos, fotos, música) sin estar explícitamente programados para resolver una tarea concreta. Se dice que el aprendizaje máquina hace uso de algoritmos genéricos sobre datos específicos y de esta manera se logra abordar problemas concretos.

Después de una breve introducción pasamos a una definición formal del aprendizaje máquina brindada por Tom M. Mitchell en 1997 “Se dice que un programa de computadora aprende de la experiencia E con respecto a alguna clase de tareas T y la medida de rendimiento P si su desempeño en tareas en T, medido por P, mejora con la experiencia E” [21].

Una vez proporcionada una explicación formal del aprendizaje máquina hecha por uno de los contribuyentes más importantes en el campo, es necesario dar nombres y símbolos a los componentes principales de el problema de aprendizaje. Está la entrada  $x$  (características de los datos de entrada que se usan para tomar una decisión), la función objetivo desconocida  $f : X \mapsto Y$  (fórmula ideal para descripción y/o predicción del problema a resolver), donde  $X$  es el espacio de entrada (conjunto de todas las posibles entradas  $x$ ), e  $Y$  es el espacio de salida (conjunto de todas las salidas posibles). Hay un conjunto de datos  $D$  de ejemplos de entrada-salida (Datos de entrenamiento)  $(x_1, y_1), (x_2, y_2), \dots, (x_n, Y_n)$ , donde  $y_n = f(x_n)$  para  $n = 1, \dots, N$ . Estos ejemplos a menudo se denominan puntos de datos [23].

Otro componente de suma importancia es el algoritmo de aprendizaje  $A$  que hace uso del conjunto de datos  $D$  para elegir una fórmula  $g : X \mapsto Y$  que se aproxima a  $f$ . El algoritmo elige  $g$  de un conjunto de fórmulas candidatas que llamamos conjunto de hipótesis  $H$ . Por ejemplo,  $H$  podría ser el conjunto de todas las fórmulas lineales a partir de las cuales el algoritmo elegiría el mejor ajuste lineal para los datos; La decisión será buena solo en la medida en que  $g$  se aproxime a  $f$ . Para lograrlo, el algoritmo elige  $g$  que mejor se adapte a  $f$  en los datos de ejemplo provistos, con la esperanza de que el algoritmo generalice correctamente [23].

En la figura 2.1 podemos ver la manera en la que los componentes de un problema de aprendizaje básico trabajan en conjunto.

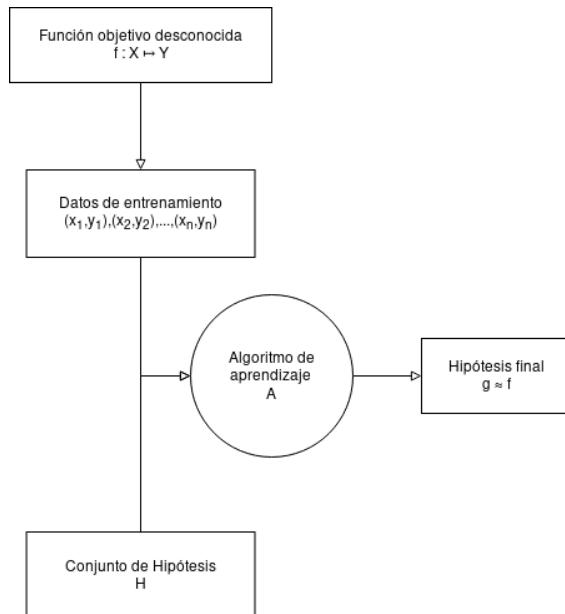


Figura 2.1: Diagrama de componentes básicos del problema de aprendizaje [23].

### 2.1.1. Antecedentes de la Inteligencia Artificial

Al pertenecer el aprendizaje máquina al campo de la inteligencia artificial es pertinente mencionar los antecedentes que dieron paso a esta disciplina que ha estado perdiendo y recuperando importancia desde el primer momento en que fue mencionada.

#### 2.1.1.1. Nacimiento [1952-1956]

1950 - Alan Turing crea la “Prueba de Turing” para determinar si una máquina es realmente inteligente o no. Para pasar la prueba, la máquina debe ser capaz de hacer creer a un ser humano que se trata de otro ser humano en lugar de una computadora [27].

1952 - Arthur Samuel escribe el primer programa de computadora capaz de aprender. El software era un programa que podía jugar a las damas y mejorar con cada juego que jugaba [27].

1956 - Martin Minsky y John McCarthy, con la ayuda de Claude Shannon y Nathan Rochester, tuvieron una conferencia en Dartmouth en 1956, que se considera donde nació el campo de la Inteligencia Artificial. Minsky convenció a los asistentes a adoptar el término “Inteligencia Artificial” como el nombre del nuevo campo [27].

1958 - Frank Rosenblatt diseña el Perceptron, la primera red neuronal artificial [27].

#### 2.1.1.2. Primer invierno de la Inteligencia Artificial [1974-1980]

En la segunda mitad de la década de 1970, el campo sufrió su primer “invierno”. Varias agencias que habían estado financiando la investigación en AI cortaron fondos después de años de altas expectativas y

---

pocos avances reales [27].

1979 - Los estudiantes de la Universidad de Stanford inventan el “Stanford Cart”, un robot móvil capaz de moverse de forma autónoma por una habitación evitando obstáculos [27].

1967 - Se escribe el algoritmo “Vecino más cercano”. Este hito se considera el nacimiento del campo del reconocimiento de patrones en las computadoras [27].

#### 2.1.1.3. La explosión de los 80's [1980-1987]

Los años 80 se conocen como el nacimiento de sistemas expertos, basados en reglas. Estos fueron rápidamente adoptados por el sector corporativo, generando un nuevo interés en Machine Learning [27].

1981 - Gerald DeJong introduce el concepto de “Aprendizaje basado en la explicación” (Explanation-Based Learning, por sus siglas en inglés EBL), en el que una computadora analiza los datos de capacitación y crea reglas generales que permiten descartar los datos menos importantes [27].

1985 - Terry Sejnowski inventa NetTalk, que aprende a pronunciar palabras de la misma manera que un niño aprendería a hacer [27].

#### 2.1.1.4. Segundo Invierno de la Inteligencia Artificial [1987-1993]

A finales de la década de 1980 y principios de la década de 1990, AI experimentó un segundo “invierno”. Esta vez, sus efectos duraron varios años y la reputación del campo no se recuperó por completo hasta principios de la década de 2000 [27].

Década de 1990: el trabajo en Machine Learning se mueve de un enfoque basado en el conocimiento a uno basado en datos. Los científicos comienzan a crear programas que analizan grandes cantidades de datos y extraen conclusiones de los resultados [27].

1997 - La computadora Deep Blue, IBM (International Business Machines), vence al campeón mundial de ajedrez Gary Kaspárov [27].

#### 2.1.1.5. Explosión y adopción comercial [2006-Actual]

El crecimiento en el potencial de cálculo junto con la gran abundancia de datos disponibles han relanzado el campo del aprendizaje automático. Muchas empresas están moviendo sus empresas hacia los datos e incorporando Machine Learning en sus procesos, productos y servicios con el fin de obtener una ventaja sobre su competencia [27].

2006 - Geoffrey Hinton acuñó la frase “Aprendizaje Profundo” (Deep Learning, por sus siglas en inglés DL) para explicar las nuevas arquitecturas de redes neuronales profundas capaces de aprender modelos mucho mejores [27].

2011 - La computadora Watson de IBM supera a sus competidores humanos en Jeopardy, un juego que consiste en responder preguntas en lenguaje natural [27].

2012 - Jeff Dean, en Google, con la asistencia de Andrew Ng (Universidad de Stanford), lidera el proyecto GoogleBrain, que desarrolló una red neuronal profunda utilizando toda la capacidad de la infraestructura de Google para detectar patrones en videos e imágenes [27].

2012 - Geoffrey Hinton lidera el equipo ganador en el concurso de Visión por computadora en Imagenet utilizando una Red Neuronal Profunda (Deep Neuronal Network, por sus siglas en inglés DNN). El equi-

po ganó por un amplio margen, dando lugar a la explosión actual de Machine Learning basada en DNN [27].

2012 - El laboratorio de investigación Google X utiliza GoogleBrain para analizar de forma autónoma videos de YouTube y detectar aquellos que contienen gatos [27].

2014 - Facebook desarrolla DeepFace, un algoritmo basado en DNN capaces de reconocer a las personas con la misma precisión que un ser humano [27].

2014 - Google compra DeepMind, una startup británica de aprendizaje profundo que recientemente ha demostrado las capacidades DNN con un algoritmo capaz de jugar juegos Atari simplemente viendo los píxeles en la pantalla, de la misma manera que lo haría una persona. El algoritmo, después de horas de entrenamiento, era capaz de vencer a expertos humanos en los juegos [27].

2015 - Amazon lanza su propia plataforma de Machine Learning [27].

2015: Microsoft crea el “Kit de herramientas para el aprendizaje de máquinas distribuidas”, que permite la distribución eficiente de problemas de aprendizaje automático en varias computadoras [27].

2015 - Elon Musk y Sam Altman, entre otros, fundan la organización sin fines de lucro OpenAI, dándole mil millones de dólares con el objetivo de garantizar que la inteligencia artificial tenga un impacto positivo en la humanidad [27].

2016 - Google DeepMind vence al jugador profesional de Go, Lee Sedol, de cinco juegos a uno en lo que se considera uno de los juegos de mesa más complejos. Los jugadores de Expert Go confirmaron que el algoritmo era capaz de realizar movimientos “creativos” que nunca habían visto antes [27].

Hoy, estamos experimentando una tercera explosión en inteligencia artificial. Aunque hay escépticos que dicen que no podemos descartar la posibilidad de un tercer invierno, esta vez los avances en el sector se están aplicando a las empresas hasta el punto de crear mercados completamente nuevos y producir cambios significativos en las estrategias de las grandes y pequeñas empresas [27].

La amplia disponibilidad de datos parece ser el combustible detrás de estos motores de algoritmo que, a su vez, están sobrepasando las limitaciones de cálculo que existían antes de la computación distribuida. Todo esto parece indicar que deberíamos seguir teniendo acceso a más y más datos que alimentarán nuestros algoritmos; la comunidad científica no parece quedarse sin ideas para seguir avanzando en el campo. 2017 y los años siguientes prometen ser realmente frenéticos [27].

## 2.1.2. Tipos de aprendizaje

La premisa básica de aprender de datos haciendo uso de un conjunto de observaciones para predecir el resultado de un proceso o descubrir un patrón oculto a simple vista es una premisa muy amplia y difícil de encajar en un marco único. Como resultado, han surgido diferentes paradigmas de aprendizaje para tratar las múltiples situaciones que se pueden presentar. En esta sección, presentamos algunos de estos paradigmas [23].

### 2.1.2.1. Aprendizaje supervisado

La mayoría del aprendizaje automático en la práctica utiliza el aprendizaje supervisado. El aprendizaje supervisado se aplica cuando se tienen variables de entrada ( $X$ ) y una variable de salida ( $Y$ ) ya conocida, dicho paradigma utiliza un algoritmo para aprender la función de mapeo desde la entrada hasta la salida.

$$Y = f(X) \quad (2.1)$$

2.1: Función de mapeo [23].

El objetivo es aproximar la función de mapeo tan bien que cuando tenga datos de entrada nuevos ( $X$ ) pueda predecir las variables de salida ( $Y$ ) para esos datos (Ecuación 2.1).

Se denomina aprendizaje supervisado porque el proceso de aprendizaje de un algoritmo a partir del conjunto de datos de capacitación puede considerarse como un maestro que supervisa el proceso de aprendizaje. Conocemos las respuestas correctas, el algoritmo realiza predicciones de forma iterativa sobre los datos de entrenamiento y es corregido por el profesor. El aprendizaje se detiene cuando el algoritmo alcanza un nivel aceptable de rendimiento [23].

### 2.1.3. Aprendizaje no supervisado

El aprendizaje no supervisado se aplica en casos cuando solo tienes datos de entrada ( $X$ ) y no hay variables de salida correspondientes. El objetivo del aprendizaje no supervisado es modelar la estructura o distribución subyacente en los datos para aprender más sobre los datos. Estos se llaman aprendizaje no supervisado porque a diferencia del paradigma anterior, no hay respuestas correctas y no hay profesor. Los algoritmos se dejan a sus propios dispositivos para descubrir y presentar la estructura interesante en los datos. Los problemas de aprendizaje sin supervisión se pueden agrupar en problemas de agrupamiento y asociación [23].

- Agrupamiento: Un problema de agrupamiento, como lo dice su nombre, consiste en encontrar distintos grupos que contengan características en común dentro de grandes cantidades de datos [23].
- Asociación: Un problema de asociación se basa en encontrar comportamientos que se describen mediante reglas como: si pasa  $X$  entonces tiende a pasar  $Y$ . Estas reglas (llamadas formalmente reglas de asociación) se pueden ver muy reflejadas en las compras en un supermercado; los clientes que suelen comprar el producto  $X$  también compran el producto  $Y$  [23].

#### 2.1.3.1. Algoritmo de agrupamiento K-means

K-means (MacQueen, 1967) es uno de los algoritmos de aprendizaje no supervisado más simples que resuelven el conocido problema de agrupamiento. El procedimiento sigue una forma simple y fácil de clasificar un conjunto de datos dado a través de un cierto número de agrupaciones (suponga  $k$  agrupaciones) fijadas a priori. La idea principal es definir  $k$  centroides, uno para cada grupo. Estos centroides deben colocarse de forma astuta debido a que la ubicación diferente causa un resultado diferente. Entonces, la mejor opción es colocarlos lo más lejos posible unos de otros. El siguiente paso es tomar cada punto que pertenezca a un conjunto de datos determinado y asociarlo al centroide más cercano. Cuando no hay ningún punto pendiente, se completa el primer paso y se realiza una agrupación temprana. En este punto, necesitamos volver a calcular  $k$  nuevos centroides como baricentros de los grupos resultantes del paso anterior. Una vez que tengamos estos  $k$  nuevos centroides, se debe realizar un nuevo enlace entre los mismos puntos de conjunto de datos y el nuevo centroide más cercano. Se ha generado un bucle. Como resultado de este bucle, podemos observar que los  $k$  centroides cambian su ubicación paso a paso hasta que no se realicen más cambios. En otras palabras, los centroides ya no se mueven. Finalmente, este algoritmo apunta a minimizar una función objetivo, en este caso una función de error al cuadrado. La función de error es la siguiente: [24]

$$J = \sum_{j=1}^k \sum_{i=0}^N \|x_i^{(j)} - c_j\|^2 \quad (2.2)$$

## 2.2: Función de error K-means [24].

Donde 2.2 es una medida de distancia elegida entre un punto  $x_i^{(j)}$  de datos y el centro del clúster  $c_j$ , En otras palabras, es un indicador de la distancia de los n puntos de datos desde sus respectivos centros de clúster.

El algoritmo para el agrupamiento K-means es el siguiente (La cuál fue obtenida de [24]):

1. Coloque K puntos en el espacio representado por los objetos que se agrupan. Estos puntos representan los centroides del grupo inicial.
2. Asigne cada objeto al grupo que tenga el centroide más cercano.
3. Cuando todos los objetos hayan sido asignados, vuelva a calcular las posiciones de los K centroides.
4. Repita los pasos 2 y 3 hasta que los centroides ya no se muevan. Esto produce una separación de los objetos en grupos a partir de los cuales se puede calcular la métrica a minimizar.

Finalmente, una representación gráfica de el resultado de ejecutar K-means con  $K = 3$  sobre un conjunto de datos se puede visualizar en la figura 2.2

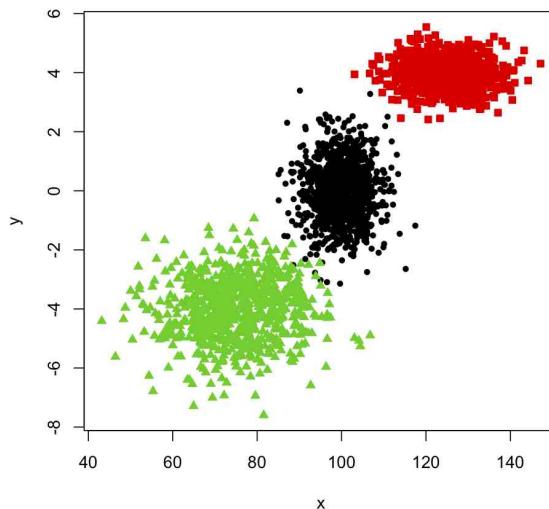


Figura 2.2: Ejemplo de agrupamiento con K-means [25].

### 2.1.4. Algoritmos de Aprendizaje supervisado

En esta sección se explicarán algunos algoritmos pertenecientes al paradigma de Aprendizaje Supervisado pero antes de realizar la explicación es necesario familiarizarse con algunas técnicas y términos.

#### 2.1.4.1. Hiperplano

Un hiperplano es un subespacio lineal de dimensión  $n - 1$  que separa en un espacio de dimensión n en dos mitades. Todo hiperplano en  $\mathbb{R}^n$  se describe mediante la ecuación número 6.1.

$$\sum_{i=1}^n a_i x_i = \Theta \quad (2.3)$$

2.3: Definición matemática de un hiperplano [28].

#### 2.1.4.2. Separabilidad lineal

La separabilidad lineal se define en la ecuación 2.4 de la siguiente manera: Dos conjuntos  $M_1 \in \mathbb{R}^n$  y  $M_2 \in \mathbb{R}^n$  son linealmente separables si existen números  $a_1, a_2, \dots, a_n, \Theta$  tales que

$$\forall x \in M_1 \sum_{i=1}^n a_i x_i > \Theta \text{ y } \forall x \in M_2 \sum_{i=1}^n a_i x_i \leq \Theta \quad (2.4)$$

2.4: Definición matemática de separabilidad lineal [28].

Donde al valor de  $\Theta$  se le denomina umbral.

#### 2.1.4.3. Red neuronal artificial

La definición más simple de una red neuronal, más propiamente llamada Red Neuronal Artificial (Artificial Neuronal Network, por sus siglas en inglés ANN), la proporciona el inventor de uno de los primeros neurocomputadores, el Dr. Robert Hecht-Nielsen. Él define una red neuronal como: "... un sistema informático compuesto por una serie de elementos de procesamiento simples, altamente interconectados, que procesan la información por su respuesta de estado dinámico a las entradas externas." en "Neural Network Primer: Part I" por Maureen Caudill, AI Expert, febrero de 1989. Las RNA son dispositivos de procesamiento (algoritmos o hardware real) que se modelan de forma flexible basadas en la estructura neuronal de la corteza cerebral, pero a escalas mucho más pequeñas. Una ANN grande puede tener cientos o miles de unidades de procesador, mientras que un cerebro tiene miles de millones de neuronas con un aumento correspondiente en la magnitud de su interacción global y comportamiento emergente [29].

#### Elementos de una red neuronal

Las redes neurales se organizan típicamente en capas. Las capas están formadas por una serie de 'nodos' interconectados que contienen una 'función de activación'. Los patrones se presentan en la red a través de la capa de entrada, que se comunica con una o más capas ocultas donde el procesamiento real se realiza a través de un sistema de 'conexiones' ponderadas. Las capas ocultas se vinculan a una capa de salida donde la respuesta se muestra como se muestra como en la figura 2.3 [29].

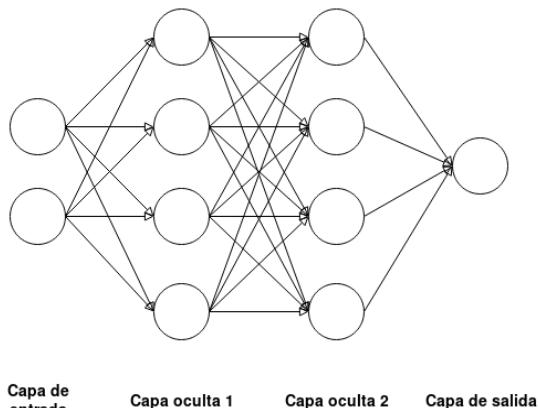


Figura 2.3: Diagrama de una red neuronal.

### Función de activación

Consideremos la neurona que se muestra en la ecuación 2.5.

$$Y = \sum_{i=0}^n (\theta_i x_i) \quad (2.5)$$

2.5: Definición matemática de una neurona. [19]

El valor de Y puede ser cualquier valor desde  $-\infty$  a  $+\infty$ . La neurona realmente no conoce los límites del valor. Entonces, ¿cómo decidimos si la neurona debe ser activada o no? para solucionar este problema decidimos agregar funciones de activación que verifican el valor Y producido por una neurona y toman la decisión de si las conexiones externas deberían considerar esta neurona como activada o no [30]. Algunas de las funciones de activación más utilizadas son las siguientes:

- Función Escalón unitario,
- Función Sigmoide,
- Función Tangente Hiperbólica,
- Función Rectificadora.

#### 2.1.4.4. Perceptrón

Es un algoritmo de clasificación muy simple que equivale a una red neuronal de dos capas (capa de entrada y capa de salida) que usualmente va de la mano de la función de activación escalón unitario (Threshold) o la función signo (Sign). Dentro de esta red neuronal cada nodo representa una neurona artificial y cada enlace simula la sinapsis que ocurre en las neuronas biológicas.

Los elementos de un perceptrón se definen como  $\theta = (\theta_1, \theta_2, \dots, \theta_n) \in \mathbb{R}^n$  que denota el vector de pesos y  $X \in \mathbb{R}^n$  el vector de entradas. Un perceptrón se define como la función de mapeo  $P : X \in \mathbb{R}^n \mapsto \{-1, +1\}$  que se puede escribir como se muestra en la ecuación 2.6.

$$P(x) = \begin{cases} +1, & Si \sum_{i=1}^n \theta_i x_i > \Theta \\ -1 & En otro caso \end{cases} \quad (2.6)$$

2.6: Definición matemática del Perceptrón. [28]

Observe que las variables  $x_i$  corresponden a cada elemento del vector de entradas  $X$  que serán tomadas en cuenta durante la clasificación. Como se puede observar en la fórmula  $\sum_{i=1}^n \theta_i x_i > \Theta$ , todos los puntos por encima del hiperplano  $\sum_{i=1}^n \theta_i x_i = \Theta$  se clasifican como positivos y los restantes como negativos. El algoritmo de aprendizaje más básico del perceptrón se define de la manera siguiente:

Perceptron( $M_+, M_-$ )  
 $w$  = vector aleatorio de números reales  
 Repetir hasta que  $x \in M_+ \cup M_-$  esté clasificado correctamente  
 $\forall x \in M_+$   
     si  $xw \leq \Theta$  entonces  $\theta_+ = x$   
 $\forall x \in M_-$   
     si  $xw > \Theta$  entonces  $\theta_- = x$  [19]

Donde  $M_+$  y  $M_-$  representan a los datos de entrenamiento positivos y negativos respectivamente. El perceptrón deberá asignar la salida de  $+1$  a todo  $x \in M_+$  y  $-1$  a todo  $x \in M_-$ . Si el perceptrón clasifica erróneamente se hace el ajuste del vector de pesos para modificar éste en la dirección correcta. En la figura 2.4 se puede apreciar el resultado final de un proceso de clasificación hecho con un perceptrón.

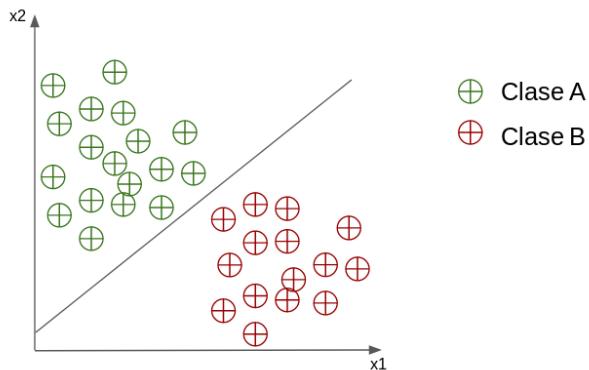


Figura 2.4: Función de clasificación lineal  $\theta_1 x_1 + \theta_2 x_2 = -b$ .

#### 2.1.4.5. Sistemas de recomendación

Un sistema de recomendación se define como una estrategia de toma de decisiones para los usuarios en entornos de información complejos. También ha sido definido desde la perspectiva del comercio electrónico como una herramienta que ayuda a los usuarios a buscar a través de registros de conocimiento que están relacionados con los intereses y preferencias de éstos mismos. Por otro lado, se definió como un medio para ayudar y aumentar el proceso social de usar recomendaciones de otros para tomar decisiones cuando no hay suficiente conocimiento personal o experiencia de las alternativas. Los sistemas de recomendación manejan el problema de la sobrecarga de información que los usuarios normalmente encuentran al proporcionarles recomendaciones de servicio y contenido personalizado y exclusivo. Recientemente, se han desarrollado varios enfoques para construir sistemas de recomendación, que pueden utilizar el filtrado colaborativo, el filtrado basado en contenido o el filtrado híbrido [31]. Cada una de estas técnicas serán explicadas a

---

continuación.

### Enfoque basado en contenido

La técnica basada en contenido es un algoritmo que se enfatiza en el análisis de los atributos de los artículos para generar predicciones. Cuando se recomiendan documentos como sitios Web, publicaciones y noticias, la técnica de filtrado basada en el contenido es la más exitosa. En la técnica de filtrado basada en contenido, la recomendación se basa en los perfiles de usuario que utilizan características extraídas del contenido de los artículos que el usuario ha evaluado en el pasado. Los artículos que están relacionados principalmente con los artículos calificados positivamente se recomiendan para el usuario. El Enfoque Basado en Contenido (Content Based Focus, por sus siglas en inglés CBF) utiliza diferentes tipos de modelos para encontrar similitud entre los documentos a fin de generar recomendaciones significativas. Podría utilizar Modelo de espacio vectorial o modelos probabilísticos como Clasificador bayesiano ingenuo, árboles de decisión o redes neuronales artificiales para modelar la relación entre las diferentes características dentro de un producto/documento. Estas técnicas hacen recomendaciones aprendiendo el modelo subyacente con análisis estadístico o técnicas de aprendizaje automático. La técnica de filtrado basada en contenido no necesita el perfil de otros usuarios, ya que no influyen en la recomendación. Además, si el perfil del usuario cambia, la técnica CBF todavía tiene el potencial de ajustar sus recomendaciones en un período de tiempo muy corto. La principal desventaja de esta técnica es la necesidad de tener un conocimiento profundo y una descripción de las características de los artículos en el perfil [31].

### Enfoque basado en filtrado colaborativo

El filtrado colaborativo es una técnica de predicción para contenido que no puede describirse de manera fácil y adecuada mediante metadatos, como películas o música. La técnica de filtrado colaborativo funciona mediante la creación de una base de datos (matriz Usuario-Artículo) de las preferencias de los artículos por parte de los usuarios. A continuación, compara usuarios con intereses y preferencias relevantes calculando similitudes entre sus perfiles para hacer recomendaciones. Dichos usuarios crean un grupo llamado barrio. Un usuario recibe recomendaciones para aquellos artículos que no ha calificado anteriormente pero que ya fueron calificados positivamente por los usuarios de su vecindario. Las recomendaciones que produce FC pueden ser de predicción o recomendación. La predicción es un valor numérico,  $R_{ij}$ , que expresa la puntuación pronosticada del ítem j para el usuario i, mientras que la Recomendación es una lista de los N artículos principales que más le gustarán al usuario, como se muestra en la figura 2.5. La técnica de filtrado colaborativo puede ser dividido en dos categorías: basado en la memoria y basado en modelos [31].



Figura 2.5: Proceso de filtrado colaborativo [31].

### Filtrado colaborativo: Basado en memoria

Los artículos que ya fueron calificados por el usuario antes desempeñan un papel relevante en la búsqueda de un vecino que comparta gustos con él. Una vez que se encuentra el vecino de un usuario, se pueden usar diferentes algoritmos para combinar las preferencias de los vecinos para generar recomendaciones. Debido a la efectividad de estas técnicas, han logrado un gran éxito en aplicaciones de la vida real. El FC basada en la memoria se puede lograr de dos maneras a través de técnicas basadas en el usuario y en artículo. La técnica de filtrado colaborativo basada en el usuario calcula la similitud entre usuarios al comparar sus calificaciones en el mismo artículo, y luego calcula la calificación pronosticada para un artículo por el usuario activo como un promedio ponderado de las calificaciones del artículo por usuarios similares al usuario activo. Las técnicas de filtrado basadas en artículos calculan predicciones utilizando la similitud entre los artículos y no la similitud entre los usuarios. Construye un modelo de similitudes de artículo recuperando todos los artículos calificados por un usuario activo de la matriz de Usuario-Artículo, determina qué tan similares son los artículos recuperados al artículo destino, luego selecciona los k artículos más similares y sus similitudes correspondientes. La predicción en este enfoque se realiza tomando un promedio ponderado de la calificación de usuarios activos en los artículos similares k. Se utilizan varios tipos de medidas de similitud para calcular la similitud entre el artículo / usuario. Las dos medidas de similitud más populares están basadas en correlación y en coseno. El coeficiente de correlación de Pearson se utilizada para medir el nivel en que dos variables aleatorias cuantitativas se relacionan linealmente entre sí. Este coeficiente se define como se muestra en la ecuación 2.7 [31].

$$s(a, u) = \frac{\sum_{i=1}^n (r_{a,i} - \bar{r}_a)(r_{u,i} - \bar{r}_u)}{\sqrt{\sum_{i=1}^n (r_{a,i} - \bar{r}_a)^2} \sqrt{\sum_{i=1}^n (r_{u,i} - \bar{r}_u)^2}} \quad (2.7)$$

2.7: Definición matemática de coeficiente de correlación de Pearson [31].

De la ecuación anterior,  $s(a, u)$  denota la similitud entre dos usuarios a y u,  $r_{a,i}$  es la calificación otorgada al artículo i por el usuario a,  $\bar{r}_a$  es la calificación promedio dada por el usuario a mientras que n es el número total de elementos en el espacio Usuario-Artículo. Además, la predicción para un artículo se realiza a partir de la combinación ponderada de las calificaciones de los vecinos seleccionados, que se calcula como la desviación ponderada de la media de los vecinos. La fórmula de predicción general es (ecuación 2.8) [31]:

$$p(a, i) = \bar{r}_a + \frac{\sum_{i=1}^n (r_{u,i} - \bar{r}_u) \times s(a, u)}{\sum_{i=1}^n s(a, u)} \quad (2.8)$$

2.8: Definición matemática de fórmula de predicción [31].

La similitud del coseno es diferente de la medida basada en Pearson porque es un modelo vectorial-espacial que se basa en el álgebra lineal más que en el enfoque estadístico. Mide la similitud entre dos vectores n-dimensionales en función del ángulo que los separa. La medida basada en coseno se utiliza ampliamente en los campos de la recuperación de información y extracción de textos para comparar dos documentos de texto, en este caso, los documentos se representan como vectores de términos. La similitud entre dos artículos u y v se puede definir con la ecuación 2.8 [31]:

$$s(\vec{u}, \vec{v}) = \frac{\vec{u} \cdot \vec{v}}{|\vec{u}| * |\vec{v}|} = \frac{\sum_i (r_{u,i} r_{v,i})}{\sqrt{\sum_i r_{u,i}^2} \times \sqrt{\sum_i r_{v,i}^2}} \quad (2.9)$$

2.9: Definición matemática de similitud del coseno [31].

La medida de similitud también se conoce como métrica de similitud, y son métodos utilizados para calcular los puntajes que expresan que tan similares son los usuarios o artículos. Estos puntajes se pueden usar como base para la generación de recomendaciones basadas en usuarios o en artículos. Dependiendo del contexto de uso, las métricas de similitud también pueden denominarse métricas de correlación o métricas de distancia [31].

### Filtrado colaborativo: Basado en modelos

Esta técnica emplea las clasificaciones anteriores para aprender un modelo con el fin de mejorar el rendimiento de la técnica de filtrado colaborativo (Colaborative Filter, por sus siglas en inglés CF). El proceso de construcción del modelo se puede hacer usando técnicas de aprendizaje máquina o minería de datos. Estas técnicas pueden recomendar rápidamente un conjunto de elementos por el hecho de que usan un modelo precalculado y han demostrado producir resultados de recomendación que son similares a las técnicas de recomendación basadas en el vecindario. Entre los ejemplos de estas técnicas se incluyen la técnica de reducción de la dimensionalidad, como la descomposición del valor singular (Singular Value Decomposition, por sus siglas en inglés SVD), la técnica de finalización de la matriz, los métodos semánticos latentes y la regresión y agrupamiento. Las técnicas basadas en modelos analizan la matriz de elementos de usuario para identificar las relaciones entre los elementos; usan estas relaciones para comparar la lista de recomendaciones de N principales. Las técnicas basadas en modelos resuelven los problemas de dispersión asociados con los sistemas de recomendación [31].

El uso de algoritmos de aprendizaje también ha cambiado la forma en que los sistemas de recomendación pasaron de qué consumir a cuándo consumir realmente un producto. Por lo tanto, es muy importante examinar otros algoritmos de aprendizaje utilizados en los sistemas de recomendación basados en modelos [31]:

- Reglas de asociación,
- Agrupamiento,
- Árboles de decisión,
- Redes neuronales artificiales,
- Análisis de enlaces,

- 
- Regresión,
  - Clasificadores Bayesianos.

### Pros y contras del filtrado colaborativo

El filtrado colaborativo tiene algunas ventajas importantes sobre CBF ya que puede funcionar en dominios donde no hay mucho contenido asociado con los elementos y donde el contenido es difícil de analizar para un sistema informático. Además, la técnica CF tiene la capacidad de proporcionar recomendaciones fortuitas, lo que significa que puede recomendar elementos que son relevantes para el usuario, incluso sin que el contenido esté en el perfil del usuario. A pesar del éxito de las técnicas de CF, su uso generalizado ha revelado algunos problemas potenciales, como los siguientes [31].

- **Inicio frío:** Esto se refiere a una situación en la que un sistema de recomendaciones no cuenta con información adecuada sobre un usuario o un elemento para hacer predicciones relevantes. Este es uno de los principales problemas que reduce el rendimiento del sistema de recomendaciones. El perfil de dicho nuevo usuario o artículo estará vacío ya que no ha calificado ningún elemento; por lo tanto, su gusto no es conocido por el sistema [31].
- **Dispersión de datos:** Este es el problema que ocurre como resultado de la falta de información, es decir, cuando solo algunos de los artículos disponibles en una base de datos son calificados por los usuarios. Esto siempre conduce a una matriz de elementos de usuario dispersa, incapacidad para localizar vecinos exitosos y, finalmente, la generación de recomendaciones débiles. Además, la escasez de datos siempre conduce a problemas de cobertura, que es el porcentaje de elementos en el sistema que para el cual se pueden hacer recomendaciones [31].
- **Escalabilidad:** Este es otro problema asociado con los algoritmos de recomendación porque el cálculo normalmente crece linealmente con la cantidad de usuarios y elementos. Una técnica de recomendación que sea eficiente cuando el número de conjunto de datos es limitado puede no ser capaz de generar un número satisfactorio de recomendaciones cuando se aumenta el volumen del conjunto de datos. Por lo tanto, es crucial aplicar técnicas de recomendación que sean capaces de escalar de manera exitosa a medida que aumenta el número de conjuntos de datos en una base de datos. Los métodos utilizados para resolver problemas de escalabilidad y acelerar la generación de recomendaciones se basan en técnicas de reducción de dimensionalidad, como el método de descomposición de valores singulares (SVD), que tiene la capacidad de producir recomendaciones confiables y eficientes [31].

### Filtrado híbrido

La técnica de filtrado híbrido combina diferentes técnicas de recomendación para obtener una mejor optimización del sistema para evitar algunas limitaciones y problemas de los sistemas de recomendación puros. La idea detrás de las técnicas híbridas es que una combinación de algoritmos proporcionará recomendaciones más precisas y efectivas que un solo algoritmo, ya que las desventajas de un algoritmo pueden ser superadas por otro algoritmo. El uso de múltiples técnicas de recomendación puede suprimir las debilidades de una técnica individual en un modelo combinado. La combinación de enfoques se puede realizar de cualquiera de las siguientes maneras: implementación separada de algoritmos y combinación del resultado, utilizando algunos filtros basados en contenido en enfoque colaborativo, utilizando algunos filtros colaborativos en enfoque basado en contenido, creando un sistema de recomendación unificado que brinde juntos ambos enfoques [31].

## 2.2. IoE (Internet of Everything)

El internet de todo (Internet of Everything, por sus siglas en inglés IoE) es un concepto que nace en Cisco a partir del internet de las cosas (Internet of Things, por sus siglas en inglés IoT), nos habla de la conexión inteligente de personas, procesos, datos y cosas. Mientras que en el internet de las cosas todas las comunicaciones se dan de Máquina a Máquina (Machine to Machine, por sus siglas en inglés M2M) en el IoE se expande este concepto incluyendo además las interacciones de Máquina a Persona (Machine to Person, por sus siglas en inglés M2P) y las de persona a persona (Person to Person, por sus siglas en inglés P2P) asistida por tecnología [32]. Para comprender mejor el IoE es necesario hablar del IoT.

### 2.2.1. Antecedentes

El internet de todo tiene su origen en el internet de las cosas, “*consiste en reunir personas, procesos, datos y cosas para conseguir que las conexiones de red sean más pertinentes y valiosas que nunca, convirtiendo la información en acciones que crean nuevas capacidades, experiencias más ricas y oportunidades económicas sin precedentes para las empresas, las personas y los países*” [33].

#### 2.2.1.1. IoT (Internet of Things)

El internet de las cosas, es la red de objetos físicos a los que se tiene acceso a través de internet. Dichos objetos tienen tecnología que permite interactuar tanto interna como externamente con otros objetos sin requerir de la entrada de una persona, un ejemplo de esto son los termostatos inteligentes, dispositivos que se conectan a la calefacción y que autorregulan la temperatura de una casa para optimizar el consumo eléctrico [33].

El IoT se puede interpretar como el Internet que se utiliza a través de diferentes dispositivos, como fotocopiadoras, teléfonos, televisores, entre otros. El objetivo de éste es brindar una experiencia en línea al usuario desarrollando nuevos y mejores gadgets que se encuentren en conexión a través de Internet [33].

El IoE incluye como partes fundamentales a las personas, los procesos, los datos y las cosas, está basado en el IoT que a su vez está basado en un solo pilar, las cosas [33]:

- Personas: Son los puntos finales, conectados a Internet permiten compartir información y actividades, por ejemplo, mediante redes sociales.
- Procesos: Permiten agregar valor mediante la conectividad de datos, cosas y personas.
- Datos: Datos analizados y procesados en información útil para permitir decisiones inteligentes y mecanismos de control.
- Cosas: Objetos físicos, dispositivos u otros elementos que permitan generar o recibir información.

#### 2.2.1.2. Sistemas embebidos

Para definir un sistema embebido es necesario distinguir cuatro características fundamentales: Hardware (embedded hardware), software (embedded software), inteligencia computacional y ejecución de una o varias tareas en tiempo real (el sistema es predecible y determinista). En este sentido, un sistema embebido se define como un dispositivo electrónico que tiene inteligencia computacional, diseñado para cumplir una o varias tareas relacionadas que se determinan desde el diseño y por lo tanto, son predecibles al ejecutarse en tiempo real y que está integrado por componentes de hardware y software. Los sistemas embebidos representan una evolución de los sistemas electrónicos rígidos, y su principal característica es la posibilidad de ser programados para resolver algorítmicamente un problema determinado. La programación de estos dispositivos puede ser realizada por el diseñador del sistema o por el sistema mismo, lo que da la sensación

---

de razonamiento, es decir, de que el sistema puede aprender por sí mismo [34].

Cuando se comunican estos sistemas con Internet, se les cambia de nombre por el de sistemas embebidos "Smart" o dispositivos "Smart" (Smart TV, Smart Phone). Así, adicional a su función original, envían información relacionada al problema que resuelven (la Smart TV, por ejemplo, podría enviar una bitácora con los canales más vistos y el horario en el que es utilizada). La información emitida por estos sistemas es almacenada en servidores, para procesarla y predecir patrones de conducta de los usuarios o avisar de fallas del sistema al proveedor. Los sistemas embebidos son cada vez más usados en todos los ámbitos de nuestras vidas: Desde un sensor de aparcamiento hasta una cafetera que se comunica con tu celular, a un smartwatch [34].

### 2.2.1.3. Cómputo ubicuo

Es entendida como la integración de la informática en el entorno de la persona, de forma que los ordenadores no se perciban como objetos diferenciados, apareciendo en cualquier lugar y en cualquier momento [35].

A diferencia de la computación de escritorio, la computación ubicua puede ocurrir al emplear cualquier dispositivo, en cualquier ubicación y en cualquier formato. El usuario interactúa con la computadora embebida, que puede existir en distintas formas, incluyendo computadoras portátiles, tabletas y terminales en objetos comunes tales como refrigeradores, televisiones o un juego de anteojos. La tecnología subyacente que soporta la computación ubicua incluye el Internet, el middleware, sistemas operativos, código móvil, sensores, microprocesadores, interfaces de usuario, redes, protocolos de comunicación, posicionamiento y ubicación y nuevos materiales [35].

La mejor forma de saber que estamos frente a computación ubicua es que las computadoras están integradas al ambiente de tal forma que nuestra interacción con estas sea casi invisible pero si perceptible, por esto también es llamada Inteligencia Ambiental. Es un ambiente basado en un modelo de interacción en el cual los usuarios están rodeados de un entorno digital consciente de su presencia, sensible al contexto y que puede adaptarse a sus necesidades y hábitos, Además, el cómputo ubicuo es una visión que coloca al ser humano como centro de desarrollo futuro en la sociedad del conocimiento, la información y la tecnología [36].

El papel que juega el Internet en el cómputo ubicuo es primordial: la alta conectividad, la capacidad de interconectar sistemas y acceder a diferentes bases de datos. Los servicios Web nos permiten acceder a todo esto de forma sencilla y estandarizada. Cualquier dispositivo con conectividad a Internet tiene el potencial de realizar peticiones a Web services y de esta forma recibir o actualizar información [36].

## 2.2.2. Tendencias

Algunas de las tendencias del IoE que impactan tanto a empresas como a lo que tecnología se refiere son las siguientes:

### 2.2.2.1. IoE e inteligencia artificial

La inteligencia artificial puede ayudar a reducir grandes cantidades de información para obtener lo que es más significativo, a través del análisis de los datos en las siguientes áreas: preparación de datos, descubrimiento de datos, visualización de datos de transmisión, precisión de datos en series temporales, análisis predictivos y avanzados, datos geoespaciales y localización en tiempo real (datos logísticos) eléctrico [37].

### 2.2.2.2. IoE y conectividad

Los objetos físicos pueden conectarse a sensores por medio de distintas tecnologías, incluyendo el Wi-Fi (Wireless Fidelity), Bluetooth, Wi-Fi de baja potencia, WiMAX (Worldwide Interoperability for Microwave

Access), Ethernet, entre otros nuevos estándares para comunicaciones. A medida que IoE siga expandiéndose asistiremos a un aumento de los dispositivos conectados a la red en distintas áreas, tanto en mercados de negocio como de consumo [38].

### 2.2.3. Ventajas

Algunas de las ventajas del IoE se presentan a continuación [37] [39] :

1. Interconexión entre diferentes objetos del entorno, gracias a ello y a su capacidad de actuar de forma autónoma se genera un entorno más inteligente.
2. Para las empresas, IoE les ayuda a reducir costes, aumentar su productividad y expandirse a nuevos mercados o desarrollar nuevos productos.
3. Velocidad de análisis de datos: La recopilación y análisis de datos se vuelve crucial. Mientras más información se obtenga con IoE será más fácil la toma de decisiones.
4. Ahorro económico: El IoE puede reemplazar ciertas tareas de monitorización que puede realizar un humano.

### 2.2.4. Desventajas

Algunas de las desventajas del IoE son las siguientes [37] [39]:

1. Falta de estandarización: No existen estándares claros, este problema se puede dividir en 3 categorías.
  - a) Plataforma: Interfaces de usuario, herramientas de análisis para la transmisión de datos procedentes de todos los objetos y la escalabilidad.
  - b) Conectividad: Incluye a todos los objetos con los que las personas interactúan
  - c) Aplicaciones: Determina 3 funciones, controlar cosas, recoger datos y analizar datos.
2. Fiabilidad de la información al dejar todo a manos de los objetos.
3. Seguridad: Los riesgos de seguridad son responsabilidad del consumidor, existe un alto riesgo de que la información de los usuarios se ha robado con otros fines.

### 2.2.5. Ejemplos

Actualmente existen muchos ejemplos para describir la función del IoE, un ejemplo es la aplicación Google Maps, en la cuál podemos conocer zonas de embotellamiento vial, para tomar en cuenta vías alternas y llegar a tiempo a nuestro destino. Esto sólo por mencionar una de las utilidades de esta multifuncional aplicación.

IoT o IoE está en un gran crecimiento gracias a compañías como Intel, Google, Cisco , Microsoft Corp. e IBM (International Business Machines Corp.) poniendo todo su empeño detrás, al igual que la cantidad de dispositivos conectados [33].

En 2012 había 8.700 millones de objetos conectados en todo el mundo, que representaban el 0.6 % de las "cosas" en el planeta, según Cisco Systems Inc. en 2013, este número superó los 10 mil millones y Cisco espera que el número de objetos conectados alcance 50 mil millones para 2020 ó el 2.7 % de las "cosas" en el mundo, impulsadas por la reducción del precio por conexión y el rápido crecimiento en el número de conexiones máquina a máquina [40].

Según ABI Research, Bluetooth, Wi-Fi, ZigBee, Cellular, Identificación por Radiofrecuencia (Radio Frequency Identification, por sus siglas en inglés RFID) y muchas otras tecnologías inalámbricas impulsan el crecimiento del IoE [40]. De acuerdo con la firma de investigación Gartner Inc. IoE creará decenas de millones de nuevos objetos y sensores que recopilen datos en tiempo real, así mismo pronostica que las empresas harán un uso intensivo de la tecnología de Tecnologías de la Información (Information Technology, por sus siglas en inglés IT) y que se venderá una amplia gama de productos en diversos mercados. Entre ellos se incluyen dispositivos médicos avanzados, sensores para la automatización de fábricas y aplicaciones de robótica industrial, sensores minúsculos para aumentar la producción agrícola, sensores para el sector de la automoción y sistemas de monitorización de la integridad de infraestructuras para áreas tan diversas como el transporte por carretera y por ferrocarril, la distribución de agua y el transporte de electricidad, en suma, una lista interminable de productos y servicios [33].

## 2.3. Mobile Computing

La Computación Móvil se refiere al uso de computadoras sin la necesidad de estar conectadas a una red cableada, sino a través de redes inalámbricas o satelitales. Se puede definir la Computación Móvil como los dispositivos que hacen uso de la computación para lograr su funcionamiento, de esta forma se desarrollan las computadoras portátiles, los teléfonos celulares, los cuadernos de notas computarizados, las calculadoras de bolsillo, entre otras. La computación móvil se está volviendo día a día un paradigma tecnológico de uso común, el cual está cambiando la forma en que se realizan las actividades laborales, académicas, de investigación y entretenimiento, como en su momento lo hizo la computación como se conoce hasta hoy [44].

El cómputo móvil implica software, hardware y comunicación móvil. La portabilidad es un aspecto importante del mismo ya que se le conoce como la capacidad de utilizar la capacidad informática sin una ubicación predefinida y/o la conexión a una red para captar datos e información [46].

El continuo avance de la tecnología y las necesidades básicas de comunicación, han permitido el crecimiento de la computación móvil, como un elemento en la vida cotidiana, en la agilización y optimización de los procesos de los organizadores e instituciones educativos. En los próximos años, los teléfonos inteligentes, serán quienes dirijan el consumo de dispositivos móviles [45].

### 2.3.1. Dispositivos de computación móvil

Desde la década de 1990 se han creado muchos tipos de computadoras portátiles, entre los más conocidos se encuentran listados a continuación: el asistente personal, teléfono inteligente, tableta, PC ultra móvil y computadoras corporales [46].

#### 2.3.1.1. Asistente digital personal (PDA)

Un asistente digital personal (Personal Digital Assistant, por sus siglas en inglés PDA), también conocido como computadora de bolsillo o asistente de datos personales, es un dispositivo móvil que funciona como un administrador de información personal. Los PDA se consideran, en gran medida, obsoletos con el uso generalizado de los teléfonos inteligentes [47].

La mayoría de los PDA pueden acceder a Internet, intranets o extranets a través de métodos inalámbricos como Wi-Fi o inalámbrico. La mayoría de los PDA usan tecnología de pantalla táctil. El término PDA se utilizó por primera vez el 7 de enero de 1992 por el CEO (Chief Executive Officer) de Apple Computer, John Sculley, en el Consumer Electronics Show en Las Vegas, Nevada, refiriéndose a Apple Newton [48].

#### 2.3.1.2. Tableta

Las tabletas son más grandes que un teléfono celular o un asistente digital personal. Son un tipo de dispositivos móviles integrados en una pantalla táctil plana y operados principalmente tocando la pantalla.

No se coloca ningún teclado físico en ellos. A menudo utiliza un teclado virtual en pantalla, un lápiz óptico pasivo o un bolígrafo digital [46].

Los primeros ejemplos del concepto de tableta se originaron en los siglos XIX y XX principalmente como prototipos e ideas conceptuales. Los primeros dispositivos electrónicos portátiles comerciales basados en el concepto aparecieron a finales del siglo XX. Apple lanzó el iPad con sistema operativo y tecnología de pantalla táctil en 2010 y se convirtió en el primer éxito comercial en todo el mundo [46].

Esto provocó un nuevo mercado para la tableta y después de este éxito muchos otros fabricantes han producido versiones propias, incluyendo Samsung, HTC (High Tech Computer Corporation), Motorola, BlackBerry, Sony, Amazon, HP (Hewlett-Packard), Microsoft, Archos, etc. Entre tabletas, los principales sistemas operativos son iOS (Apple), Android (Google), Windows (Microsoft) y QNX(BlackBerry). En 2012, se informó que el 31 % de los usuarios de Internet en Estados Unidos tenían una tableta, que se utilizaba principalmente para visualizar contenido como videos y noticias [49].

#### 2.3.1.3. PC Ultra-Móvil

Una computadora personal ultra móvil (Ultra Mobile PC, por sus siglas en inglés UMPC) es una versión de formato pequeño de una computadora de bolígrafo, una clase de computadora portátil cuyas especificaciones fueron lanzadas por Microsoft e Intel en la primavera de 2006. Sony con su serie Vaio U había fabricado el primer intento en esta dirección en 2004, que sin embargo solo se vendió en Asia. Los UMPCs son más pequeños que los subportátiles operados como tabletas, con una pantalla de Transistor de Películas Finas (Thin Film Transistor, por sus siglas en inglés TFT) que mide (en diagonal) alrededor de 12.7 a 17.8 cm, y una pantalla táctil o un lápiz óptico. No hay límites definidos entre los subportátiles y las PC ultra móviles [50].

Los UMPC de primera generación eran simples computadoras personales (Personal Computer, por sus siglas en inglés PC) con Linux o una versión adaptada del sistema operativo de la tableta PC de Microsoft. Los UMPC de segunda generación usan menos electricidad y, por lo tanto, se pueden usar por más tiempo (hasta cinco horas) y también son compatibles con Windows Vista [50].

Originalmente con el nombre en código Project Origami, el proyecto se lanzó en 2006 como una colaboración entre Microsoft, Intel, Samsung y algunos otros. A pesar de la predicción de la desaparición de la categoría de dispositivo UMPC según el sitio Web de multimedia estadounidenses "CNET", la categoría UMPC parece seguir existiendo, sin embargo, ha sido suplantada en gran medida por las tabletas [46].

#### 2.3.1.4. Computadoras corporales

Las computadoras portátiles, también conocidas como computadoras corporales, son dispositivos electrónicos en miniatura que lleva el portador debajo, con o encima de la ropa . Una de las principales características de una computadora portátil es la consistencia. Hay una interacción constante entre la computadora y el usuario, es decir, no hay necesidad de encender o apagar el dispositivo. Otra característica es la capacidad de multitarea. No es necesario que el usuario deje de hacer lo que está haciendo para usar el dispositivo. Estos dispositivos pueden ser incorporados por el usuario para actuar como una prótesis [46].

A comienzos del siglo XXI, la "computadora corporal" es un tema de investigación, cuyas áreas de estudio incluyen diseño de interfaces de usuario, realidad virtual y reconocimiento de patrones. El uso de estos dispositivos para aplicaciones específicas o para compensar discapacidades, así como apoyar en la estabilidad de los ancianos va en aumento. La aplicación de las computadoras corporales en el diseño de moda queda ejemplificada en el prototipo de Microsoft del "Vestido Impreso", presentado en el Simposio Internacional de Dispositivos Corporales en junio de 2011 [53].

### 2.3.1.5. Teléfono inteligente

Un teléfono inteligente es un teléfono móvil construido en un sistema operativo móvil, con capacidad informática más avanzada y mejor conectividad que un simple teléfono [46].

Los primeros teléfonos inteligentes eran una combinación de un asistente digital personal (PDA) y las funcionalidades de un teléfono móvil. Algunas funcionalidades se agregaron en modelos posteriores, como reproducir multimedia, cámaras digitales compactas de gama baja, unidades de navegación GPS para formar un dispositivo multiuso, pantallas táctiles de alta resolución y navegadores Web para visualización de sitios Web estándar y páginas optimizadas para dispositivos móviles. Además, la conexión por medio de Wi-Fi proporciona acceso a datos de alta velocidad y banda ancha móvil [46].

Los sistemas operativos móviles (Mobile Operative System, por sus siglas en inglés Mobile OS) más comunes utilizados por los teléfonos inteligentes modernos incluyen Android de Google, iOS de Apple, Symbian de Nokia, BlackBerry OS de RIM, Bada de Samsung y Windows Phone de Microsoft. Tales sistemas operativos son capaces de ajustarse con muchos modelos diferentes de teléfonos [46].

### 2.3.2. Antecedentes del teléfono inteligente

En aproximadamente 45 años desde la primera llamada a través de un teléfono inalámbrico, los teléfonos inteligentes mejor conocidos por su nombre en inglés, smartphones, han cambiado desde su tamaño y aspecto, hasta en sus funcionalidades. El diario periodístico "Gestión de Perú" nos presenta una breve historia y evolución de los smartphones hasta el año 2000 la cual se muestra a continuación [51]:

**13 de abril de 1973: La primera llamada.** Se realizó aquel día a través de un teléfono inalámbrico, cuando el director de Motorola Martin Cooper se comunicó desde una calle de Nueva York con su mayor rival en el sector, Joel Engel de Bell Labs de AT&T.

**1 de agosto de 1994: IBM Simon Personal Communicator.** Creado por IBM, fue el primer celular reconocido como Smartphone. Incluía características de un asistente personal de datos. Simon, fue el primer Smartphone con pantalla táctil. En 1995 se dejó de vender.

**1996: Nokia 9000 communicator.** Mezclaba las capacidades de una agenda digital con las funciones de comunicación de un celular. Podían enviar correos, contaban con agenda para citas, capacidad para grabar contactos y hasta podían mandar documentos por fax.

**1997: Ericsson GS88.** Ericsson describió su modelo GS88 "Penelope" como un "teléfono inteligente", separando las dos palabras en inglés, "Smartphone".

**1999: Benefon Esc.** El segundo fabricante finlandés de teléfonos móviles, después de Nokia, lanzó el Benefon Esc, el primer teléfono con grabadora de mensajes y un receptor satelital tipo GPS.

**2000: Ericsson R380.** Al iniciar el nuevo siglo, Ericsson lanzó el que podría ser considerado el primer Smartphone propiamente dicho. Se trataba del R380, un terminal que costaba cerca de US\$ 700 y el primero en usar el sistema operativo Symbian OS, que contenía un soporte para Bluetooth.

**2000: SPH-M100.** De Samsung, fue el primer celular con reproductor de archivos MP3. Su display era un LCD monocromo con capacidad para cinco líneas de texto y gráficos sencillos. Contaba con 64MB de memoria interna y la reproducción se controlaba desde un mando en el cable de los auriculares.

**Junio 2000: SCH-V200.** Samsung lanzó al mercado de Corea del Sur el modelo SCH-V200, el primer teléfono con cámara fotográfica incorporada. El dispositivo tomaba fotos de 0,35 megapíxeles y tenía capa-

---

ciudad para tomar 20 fotografías.

**Noviembre 2000: Sharp J-SH04.** Sharp Corporation puso a la venta en Japón el J-SH04, un teléfono móvil con una cámara, la cual contaba con un dispositivo de imagen CMOS (Complementary Metal Oxide Semiconductor, por sus siglas en inglés CMOS), (tecnología para evitar menor consumo de energía) con una lente de 0,11 megapíxeles que permitía tomar fotografías y enviarlas a otras personas por correo electrónico.

**2001: Sony Ericsson T68.** Fue el primer teléfono móvil en el mundo con pantalla de color.

**Noviembre 2002: Sanyo SCP-5300.** Fue el primer dispositivo con cámara fotográfica que llegó a los Estados Unidos. Las novedades en este teléfono con forma de "Concha" eran los ajustes personalizables de la cámara, como uso de flash, control de balance de blancos, auto disparador, zoom digital y filtros como sepia, blanco y negro, negativo, etc.

**Octubre 2003: Nokia N-Gage.** Nokia presentó, una apuesta de la empresa con sede en Finlandia por combinar un teléfono móvil con una videoconsola portátil. Si bien fue un fracaso comercial, logró abrir un hueco en la historia de la telefonía móvil dado su carácter único.

**2003: Palmone Treo 600.** La empresa Palm lanzó el Treo 600 que contaba con cuatribanda Sistema Global para comunicaciones Móviles(Global System for Mobile communications, por sus siglas en inglés GSM), pantalla de color, navegador de cinco direcciones, teclado retroiluminado y cámara digital.

**9 de enero del 2007: iPhone.** Steve Jobs presentó el iPhone de Apple, seis meses después se empieza a vender con mucho éxito en Estados Unidos. Además de llamadas telefónicas, recibir y enviar mensajes y tomar fotos, daba información de la bolsa, tenía correo electrónico, calendario, estado del tiempo, entre otras cosas. Marcó el hito, pues los otros fabricantes comenzaron a copiar las características del iPhone.

**22 de octubre del 2008: HTC Dream.** HTC lanzó el Dream, comercializado también como T-Mobile G1 y denominado popularmente Google Phone o Gphone. Según su fabricante, fue el primer dispositivo móvil de comunicación en incorporar el sistema operativo móvil de Google denominado Android.

**2009: OMNIA HD i8910.** El primer teléfono celular del mundo capaz de grabar video en alta definición. También poseía soporte DivX, puerto HDMI, conector de 3.5mm, Wi-Fi con DLNA (Digital Living Network Alliance) y GPS.

**2010: Nokia N8.** Fue el teléfono celular insignia de Nokia, con una pantalla touchscreen capacitiva de 3.5 pulgadas, cámara de 12 megapíxeles con flash Xenon y captura de video a 720p y 25 cuadros por segundo, salida interfaz Multimedia de Alta Definición (High-Definition Multimedia Interface, por sus siglas en inglés HDMI), radio FM con transmisor, Wi-Fi, GPS y 16GB de memoria interna entre otras características. Fue el primer teléfono inteligente de Nokia en correr el sistema operativo Symbian 3.

**2010: Sony Ericsson Z1010.** Fue el primer celular con dos cámaras, dos pantallas y una velocidad que multiplica por dos su rendimiento. Su pantalla de color ofrecía una estupenda visibilidad bajo luz solar directa.

**2011: iPhone 4S.** Poseía la misma pantalla de 3.5 pulgadas a 640x960 píxeles de resolución, pero con un procesador dual-core A5 que le proveía el doble de velocidad y le incorpora una cámara de 8 megapíxeles con captura de video 1080p y corre con el renovado iOS5.

**2011: LG Optimus 3D.** Fue el primer teléfono celular capaz de leer gráficos 3D. Posee una pantalla LCD capacitiva de 4.3 pulgadas a 480x800 píxeles de resolución, con tecnología 3D sin necesidad de anteojos. Las dos cámaras de 5 megapíxeles en la parte posterior graban video 3D estereoscópico a resolución HD, que

---

se puede publicar en un canal especial de YouTube.

**2011: Motorola Atrix.** Es un teléfono inteligente con sistema operativo Android y un procesador NVIDIA Tegra 2 de doble núcleo. Posee una pantalla de 5 megapíxeles con captura de video HD, Wi-Fi, GPS, 1GB de RAM y una batería de 1930mAh.

**2015: Fujitsu Arrows NX F-04G.** Es el primer terminal con reconocimiento del iris, un sistema que se utiliza actualmente para desbloquear el teléfono como para aceptar pagos electrónicos. El dispositivo fue presentado por la operadora DoCoMo.

### 2.3.3. Sistemas operativos

Un sistema operativo móvil o Mobile OS es un conjunto de programas de bajo nivel que permite la abstracción de las peculiaridades del hardware específico del teléfono móvil y provee servicios a las aplicaciones móviles, que se ejecutan sobre él. Al igual que los PC que utilizan Windows, Linux o Mac OS, los dispositivos móviles tienen sus sistemas operativos, como hemos hecho mención anteriormente, tanto para los teléfonos inteligentes como las tabletas los sistemas operativos más utilizados son Android, iOS, Windows Phone, BlackBerry OS, entre otros. Los sistemas operativos móviles son mucho más simples y están más orientados a la conectividad inalámbrica, los formatos multimedia para móviles y las diferentes maneras de introducir información en ellos [52].

#### 2.3.3.1. Symbian

Symbian era un sistema operativo móvil diseñado para teléfonos inteligentes desarrollado originalmente por Symbian Ltd. pero actualmente mantenido por Accenture. La plataforma Symbian es la sucesora de Symbian OS y Nokia Series 60. La última versión, Symbian 3, se lanzó oficialmente en el cuarto trimestre de 2010 y se utilizó por primera vez en el Nokia N8. El primer teléfono Symbian, el teléfono inteligente con pantalla táctil Ericsson R380 fue lanzado en 2000 y fue el primero dispositivo que se comercializó como un “teléfono inteligente” [46].

En 2010, Nokia lanzó el teléfono inteligente Nokia N8 con una pantalla táctil capacitiva sin aguja, el primer dispositivo en usar el Symbian 3. Algunas estimaciones indican que la cantidad de dispositivos móviles con el sistema operativo Symbian hasta el final del segundo trimestre de 2010 fue de 385 millones. Symbian fue la plataforma de teléfonos inteligentes número uno por cuota de mercado desde 1996 hasta 2011 cuando cayó al segundo lugar detrás del sistema operativo Android de Google [46].

En febrero de 2011, Nokia anunció que reemplazaría a Symbian con Windows Phone como sistema operativo en todos sus futuros teléfonos inteligentes. Esta transición se completó en octubre de 2011, cuando Nokia anunció su primera línea de Teléfonos inteligentes Windows Phone 7.5, Nokia Lumia 710 y Nokia Lumia 800 [46].

#### 2.3.3.2. Windows

Windows Phone (abreviado WP) es un sistema operativo móvil desarrollado por Microsoft, como sucesor de Windows Mobile. A diferencia de su predecesor fue enfocado en el mercado de consumo en lugar del mercado empresarial. Con Windows Phone, Microsoft ofreció una nueva interfaz de usuario que integró varios de sus servicios activos. Compitió directamente contra Android de Google e iOS de Apple. Su última versión fue Windows Phone 8.1, lanzado el 14 de abril de 2014 [54].

Debido a la evidente fragmentación de sus sistemas operativos, Microsoft anunció en enero de 2015 que daba de baja a Windows Phone, para enfocarse en un único sistema más versátil denominado Windows 10 Mobile, disponible para todo tipo de plataformas (teléfonos inteligentes, tabletas y computadoras). Sin embargo el 8 de octubre de 2017, el ejecutivo de Microsoft, Joe Belfiore, reveló que la compañía ya no

desarrollaría nuevas funciones o hardware para teléfonos con Windows, debido a su baja participación en el mercado y la consiguiente falta de software de terceros para la plataforma. Microsoft abandonó en gran parte su negocio móvil, despidiendo a la mayoría de los empleados de Microsoft Mobile en 2016, en su lugar se centró en proporcionar aplicaciones y servicios compatibles con Android e iOS [55].

#### 2.3.3.3. BlackBerry

Blackberry OS, es un sistema operativo incluido teléfonos móviles de la compañía Canadiense Research In Motion (RIM, viene incorporado en los móviles que también llevan nombre Blackberry, seguido por el modelo correspondiente. Su nivel de seguridad es lo que hace a estos de teléfonos móviles los preferidos por los profesionales y empresarios, así como la privacidad y salvaguarda de sus datos privados, aunque en los últimos años han ganado lugar entre el público masivo, dada la incorporación de herramientas de entretenimiento exclusivas [56].

La versión más actual del sistema operativo Blackberry OS, es la versión 7.0, aunque no es apta para todos sus terminales, solo los modelos más avanzados tienen soporte y actualización automática a esta última versión. Además, el sistema operativo Blackberry cuenta con un completo sistema de entretenimiento en el que se incluyen Blackberry Music y Blackberry Messenger dos aplicaciones esenciales para los usuarios de este sistema operativo [56].

#### 2.3.3.4. iOS

iOS (previamente iPhone OS) es un sistema operativo móvil desarrollado y distribuido por Apple Inc. Originalmente lanzado en 2007 para iPhone y iPod Touch, se ha ampliado para admitir otros dispositivos de Apple como el iPad y Apple TV. A diferencia de Windows CE de Windows (Windows Phone) y Android de Google, Apple no licencia iOS para instalación en hardware que no sea de Apple. A partir de 2012, la App Store de Apple contenía más de 700,000 aplicaciones, que colectivamente se han descargado más de 30 mil millones de veces [46].

La interfaz de usuario de iOS se basa en el concepto de manipulación directa, utilizando gestos multitáctiles. El control de la interfaz tiene elementos que consisten de pantallas deslizables, interruptores y botones. La respuesta a la entrada del usuario es inmediata y proporciona una interfaz fluida. La interacción con el OS incluye gestos como deslizar, tocar, pellizcar y pellizcar hacia atrás, todos los cuales tienen definiciones específicas dentro del contexto del sistema operativo iOS y su interfaz multitáctil. Acelerómetros internos son usados por algunas aplicaciones para responder al movimiento del dispositivo o rotándolo en tres dimensiones (un resultado común es cambiar de modo vertical a horizontal). iOS se deriva de OS X, con el que comparte la fundación Darwin y, por lo tanto, es un sistema operativo Unix. iOS es la versión móvil de Apple del sistema operativo OS X utilizado en las computadoras Apple. En iOS, hay cuatro capas de abstracción: la capa Core OS, la capa Core Services, la capa Media y la capa Cocoa Touch [46].

#### 2.3.3.5. Android

Android es un sistema operativo basado en Linux diseñado principalmente para dispositivos móviles con pantalla táctil como teléfonos inteligentes y tabletas, desarrolladas por Google junto con Open Handset Alliance. Desarrollado inicialmente por Android Inc, a quien Google respaldó financieramente y luego compró en 2005. Android se dio a conocer en 2007 junto con la fundación de Open Handset Alliance, un consorcio de 86 empresas de hardware, software y telecomunicaciones dedicadas al avance de estándares abiertos para dispositivos móviles [46].

Google publicó el código de Android como código abierto, bajo la Licencia Apache. El proyecto de código abierto de Android (AOSP), liderado por Google, tiene la tarea de mantener y desarrollar aún más para Android. Adicionalmente, Android tiene una gran comunidad de desarrolladores que escriben aplicaciones ("aplicaciones") que extienden la funcionalidad de los dispositivos. Los desarrolladores escriben

principalmente en una versión personalizada de Java y las aplicaciones se pueden descargar de tiendas en línea tales como Google Play (anteriormente Android Market), la tienda de aplicaciones dirigida por Google o sitios de terceros. En junio de 2012, hubo más de 600,000 aplicaciones disponibles para Android y la cantidad estimada de aplicaciones descargadas de Google Play fue de 20 mil millones [46].

El primer teléfono con Android se vendió en octubre de 2008 y para finales de 2010, Android se había convertido en la plataforma de teléfonos inteligentes líder en el mundo. Tenía una cuota de mercado mundial de teléfonos inteligentes del 59 % al comienzo de 2012. Para mayo de 2017 Android cuenta con el 85 % del mercado mundial de teléfonos inteligentes, siguiéndole iOS con el 14.7%, Windows Phone con el 0.1 % y el resto de sistemas operativos igualmente con 0.1 % [57].

#### 2.3.4. Limitaciones

Las siguientes son algunas de las limitaciones que presenta el cómputo móvil, las cuales son descritas en el artículo "Mobile Computing: Principles, Devices and Operating Systems" [46].

- **Estándares de seguridad:** Cuando se trabaja en dispositivos móviles, en ocasiones se depende de las redes públicas, que requieren un uso cuidadoso de VPN. La seguridad es una preocupación importante al respecto de los estándares de computación móvil. Se podría atacar fácilmente la VPN a través de una gran cantidad de redes interconectadas a través de la línea [46].
- **Seguridad en teléfonos inteligentes:** Son los objetivos preferidos de los ataques. Estos ataques explotan las debilidades relacionadas con los teléfonos inteligentes que pueden provenir de medios de telecomunicación inalámbrica como redes Wi-Fi(Wireless Fidelity) y Sistema Global para comunicaciones Móviles (Global System for Mobile communications, por sus siglas en inglés GSM). También hay ataques que explotan las vulnerabilidades del software tanto del navegador Web como del sistema operativo. Finalmente, existen formas de software malicioso que se basan en el conocimiento débil de los usuarios promedio [46].
- **Consumo de energía:** Cuando una toma de corriente o un generador portátil no está disponible, las computadoras móviles deben depender completamente de la energía de la batería. En combinación con el tamaño compacto de muchos dispositivos móviles, esto a menudo significa que se deben usar baterías inusualmente caras para obtener la duración de la batería necesaria [46].
- **Interferencias de transmisión:** El clima, el terreno y el rango desde el punto de señal más cercano pueden interferir con la recepción de la señal. La recepción en túneles, algunos edificios y áreas rurales a menudo es pobre [46].
- **Interfaz humana con el dispositivo:** Las pantallas y los teclados tienden a ser pequeños, lo que puede dificultar su uso. Los métodos de entrada alternativos, como reconocimiento de voz o escritura a mano, requieren capacitación [46].

#### 2.4. Sistema de posicionamiento en interiores

Muchas cosas que resultaban útiles hace más de 30 años hoy en día pueden estar obsoletas, como en el caso de los mapas, esto cambió radicalmente con los avances tecnológicos a inicio de los 2000 con la popularización de la navegación por satélite (Sistemas globales de navegación por satélite, GNSS, por sus siglas en inglés) con el ya famoso GPS, que nos permite determinar la posición de un objeto en toda la tierra [41].

Con estos navegadores es más rápido y eficiente llegar a cualquier sitio. Con la llegada del GPS a los dispositivos móviles conocer la ubicación de lugares, ha significado un gran impacto tanto para los usuarios

como para las empresas de logística y mensajería [43].

Sin embargo, existen dificultades para determinar la ubicación de una persona en espacios interiores ya que la señal de los satélites no es capaz de llegar con la intensidad necesaria a estos y si llegara se necesitarían tener los mapas de los edificios de forma pública y en un formato adecuado para que un dispositivo móvil fuera capaz de guiarnos por ellos [41].

Un sistema de posicionamiento en interiores (Indoor Positioning System, por sus siglas en inglés IPS) es una red de dispositivos utilizados para localizar inalámbricamente objetos o personas dentro de un edificio. En lugar de usar los satélites, un IPS se basa en anclajes de proximidad que localizan etiquetas o proporcionan contexto ambiental a los sensores [42].

La red de dispositivos se puede crear mediante balizas Bluetooth, de ultrasonidos, de banda ultra ancha (Ultra Wide Band, por sus siglas en inglés UWB), de infrarrojos y usando otros tipos de sensores dedicados. En cuanto a las aplicaciones, la herramienta puede mejorar la navegación en hospitales, para ayudar a los usuarios a encontrar más fácilmente las áreas a las que desean llegar. También puede emplearse en tiendas de retail, terminales de autobuses, aeropuertos y para temas de logística en una fábrica o en grandes superficies [43].

## 2.5. Beacons Estimote

Hoy en día, la marca “Estimote”, productora de estos dispositivos, proporciona 5 diferentes tipos de Beacons con una variedad de características que los diferencia entre sí. En el cuadro 2.1 se observa una comparación entre las características que dichos Beacons de la marca “Estimote” ofrecen.

Característica	Location UWB Beacon	Location Beacon	Proximity Beacon	Sticker Beacon	Video Beacon
Batería	5 años	5 años	2 años	1 año	Carga por USB
Distancia	200 m.	200 m.	70 m.	7 m.	10 m.
Ancho	27 mm.	24 mm.	17 mm.	6 mm.	14 mm.
Sensores integrados	Temperatura Movimiento Luz ambiente Presión	Temperatura Movimiento Luz ambiente Presión Magnetómetro	Temperatura Movimiento	Temperatura Movimiento	-
Localización interior	Automapeo	Mapeo manual	-	Rastreo Nearable	-

Cuadro 2.1: Comparación entre Beacons “Estimote” disponibles [72].

### 2.5.0.1. iBeacon y Eddystone

Actualmente existen dos diferentes protocolos que permiten a los dispositivos móviles recibir señales emitidas por los Beacons, estas tecnologías conocidas como iBeacon y Eddystone fueron desarrolladas por Apple en el 2013 y Google en el 2015, respectivamente. Estos protocolos tienen como funcionalidad el poder definir los datos que los Beacons emitirán mediante el Bluetooth y el formato que tendrán dichos datos, sin embargo estos protocolos no son las únicas opciones para emitir la publicidad, los Beacons desarrollados por Estimote como ya se explicará más a fondo posteriormente, tienen la capacidad de transmitir paquetes

de información adicionales tales como la batería de los dispositivos y algunos otros datos de telemetría.

A continuación en el cuadro 2.2 se muestra una comparativa entre las características que las dos tecnologías mencionadas anteriormente ofrecen.

Característica	iBeacon	Eddystone
Compatibilidad	Soporte iOS y Android	Soporte iOS y Android
Transmisión de paquetes	Transmite 1 tipo de trama: UUID.	Transmite 3 tipos de trama: URL, UID y TLM.
Interacción con aplicaciones	Al trabajar únicamente con UUID, necesita una aplicación móvil para interactuar.	Puede interactuar a través del envío de URLs a navegadores Beacon, es decir, sin una aplicación móvil.
Perfil	Es un software propiedad de Apple.	Es un software de código libre. Su especificación fue publicada en GitHub con la licencia Apache v2.0 para que desarrolladores puedan contribuir a él.
Facilidad de implementación	Su implementación es simple lo cual facilita su integración en los sistemas.	Es flexible pero requiere una codificación más complicada para realizar su integración en el sistema debido a que envía mayor información.

Cuadro 2.2: Cuadro comparativo entre iBeacon y Eddystone [73] [74].

Las opciones que cada uno de ellos provee para la transmisión de paquetes son importantes debido a que estas contienen la publicidad o información a emitir por parte de los Beacons.

Para el caso de iBeacon, solo se maneja una forma de transmisión conocida como Identificador Único Universal (Universally Unique Identifier, por sus siglas en inglés UUID), misma que es un sistema identificador estándar que permite generar un número de identificación único para un dispositivo o en este caso un Beacon, y a su vez, distinguir los Beacons que son externos a la red que no se está controlando [75].

Por otro lado, en el caso de Eddystone las opciones provistas son un poco mayores, se tienen:

- Eddystone-UID: El identificador único (Unique ID, por sus siglas en inglés UID), es una característica de Eddystone que identifica cada Beacon. Se comporta de manera similar a UUID, sin embargo este divide su trama en 2 partes diferentes un nombre y una instancia [76].
- Eddystone-URL: Es un formato de broadcast Beacon que envía una URL (Uniform Ressource Locator) al dispositivo de un usuario. Este señal puede ser obtenida por un Widget de Google Chrome o una "Aplicación Física" Web [78].
- Eddystone-TLM: Este tipo de trama permite transmitir diferentes datos de telemetría (Telemetry, por sus siglas en inglés TLM) como la batería, el tiempo de actividad del dispositivo, entre otros [79].

#### 2.5.0.2. Estimote API

Las API's de Estimote permiten la interacción con el hardware (Beacons), su función es “consumir” los paquetes de Bluetooth enviados por los Beacons vía broadcast con la finalidad de obtener los movimientos de las personas u objetos sin la necesidad de una red WI-FI.

Las diferentes API's que Estimote proporciona son [80]:

- Proximity: Permite a la aplicación móvil detectar las áreas de interés cercanas al dispositivo. Por ejemplo si el Beacon ha sido colocado en una puerta principal, los teléfonos celulares ahora detectarán que están próximos a dicha puerta.
- Indoor Location: Replica la tecnología GPS pero al interior de los establecimientos en los cuales no hay una cobertura satelital, por lo tanto al usar un conjunto de este tipo de Beacons, ellos automáticamente mapearán y crearán un plano.
- Nearables: Estos objetos simplemente advierten su estado y presencia al Bluetooth del dispositivo móvil.
- Display(Mirror): Esta API permite obtener el control inalámbricamente de los displays cercanos. En un caso hipotético, se podría caminar por el aeropuerto y al pasar frente una televisión, se obtendría la puerta y un mapa para llegar a ella en el celular.

Sin embargo, también existen API's a una mayor escala que permiten hacer uso de una gran variedad de Beacons distribuidos a lo largo de diferentes punto geográficos, estas son enlistadas a continuación [80]:

- Beacon Health Check: Los Beacons envían una señal broadcast con datos de telemetría tales como su nivel de batería o la última vez en la que cambiaron su posición. Dicha información es subida a Estimote Cloud mediante los SDK's de Estimote, de esta manera se notifica al usuario si alguno de los Beacons ha sido apagado.
- Remote Fleet Management: Esta API permite a los usuarios actualizar el firmware o alguna otra característica mediante el Estimote Cloud y estas a su vez se propagarán automáticamente a los Beacons registrados cuando estos se encuentren activos.
- Bulk Updater and Deployment Tools: Esta API permite configurar eficientemente los Beacons cuando se cuenta con una gran cantidad de ellos.
- Analytics: Permite monitorear cuáles son las áreas y los objetos que más han interactuado con un Beacon en particular.

## 2.6. Marketing

El marketing es el estudio del potencial interés que tienen los vendedores y compradores por un producto o servicio.

Ese proceso se desenvuelve en las fases de investigación del mercado en el cual se plantea vender el producto, la promoción, la venta y la distribución de ese producto o servicio, mismo que en cada caso puede abarcar desde una distribución local hasta la distribución en el país y fuera de él [81].

### 2.6.1. Marketing de proximidad

Se conoce como marketing de proximidad a aquellos sistemas que utilizan tecnologías de localización para comunicarse directamente con los consumidores por medio de sus dispositivos móviles. Entre los usos del marketing de proximidad se encuentran la distribución de multimedia en conciertos, información de aplicaciones sociales y anuncios locales. Este tipo de marketing no se ve limitado únicamente a teléfonos celulares, puede ser implementado tanto en laptops como en tablets que tienen su GPS activo y que de igual

manera, pueden ser conectados a tecnologías de proximidad como: NFC (Near Field Communication), Geofencing y Wi-Fi Hotspot [82].

Lugares como centros comerciales, tiendas departamentales y grandes puntos de venta donde los consumidores generalmente planean pasar al menos una hora o dos, son áreas privilegiadas para una campaña de marketing de proximidad, debido a que los clientes se encuentran haciendo compras activamente, lo que permite que sean más abiertos a información, promociones y sugerencias de productos [83].

### 2.6.2. Importancia del marketing de proximidad

Según un análisis de The Competitive Intelligence Unit, en México existen 112.8 millones de líneas móviles activas de las cuales el 85 % corresponden a smartphones. Con la cantidad mencionada se puede afirmar que los teléfonos se están convirtiendo en una necesidad básica del siglo XXI, lo cual hace a casi todos los consumidores con un teléfono inteligente potencialmente susceptibles a una campaña de marketing de proximidad, en particular los jóvenes que son más propensos a hacer uso de dichos dispositivos durante sus compras [84].

Entre las principales ventajas que el marketing de proximidad ofrece a los clientes y consumidores se encuentran [85]:

- Acceso inmediato a la información de los productos/servicios que satisfacerán la necesidad, provista por parte de la empresa.
- Recepción de promociones, cupones de descuento y reducciones en futuras compras.
- Mediante el acceso a diferentes motores de búsqueda, se obtiene una mayor facilidad para conocer la calidad de un servicio o producto.
- Interacción en tiempo real mediante la geolocalización de los clientes, mismo que promueve el aspecto lúdico y atractivo lo cual incita a una participación por parte del consumidor.

### 2.6.3. Estrategias de marketing de proximidad

#### 2.6.3.1. Conexión a una red WiFi

En los centros comerciales, es un acto común que un usuario se conecte a una red WiFi de forma gratuita para evitar el consumo de datos, esto da oportunidad al marketing de proximidad para enviar mensajes mientras el usuario permanezca conectado a la red. Otra opción es que para poder usar la red de forma gratuita el usuario complete un registro de esta forma se adquiere más información del usuario [58].

#### 2.6.3.2. Cupones digitales en el lugar

El envío de cupones digitales es una práctica común en el marketing de proximidad, ya sea por Bluetooth o Wi-fi. Puede ser una herramienta poderosa para aprovechar los tiempos de alto tráfico, probar el interés en un nuevo producto o estimular las ventas en un día lento [58].

#### 2.6.3.3. Lectura de código QR

El código QR (Quick Response barcode) es un código de barras bidimensional que actualmente se ha adaptado como una forma más de conectar con los clientes. Esta estrategia funciona únicamente si los mismos eligen escanear el código con su smartphone, pero requiere menos tiempo y atención que la señalización digital. Los compradores pueden escanear sobre la marcha y conectarse con el contenido en el acto

---

o marcarlo y volver a él más tarde [58].

#### 2.6.3.4. NFC

NFC son las siglas de Near Field Communication, una tecnología de comunicación inalámbrica que permite hacer pagos con un smartphone, pero también permite interacción entre clientes y marcas. Tiene muy corto alcance, es decir, se hace necesario que los terminales estén prácticamente en contacto [59].

La ventaja sobre el código QR es la comodidad y la velocidad, el NFC elimina los pasos de tener que realizar una foto; sólo es necesario pasar el dispositivo por encima del NFC y ya se cuenta con la información en pantalla por lo que hace falta únicamente imaginarse las campañas relacionadas con las ofertas y descuentos, con entradas a eventos o las promociones de puntos [60].

#### 2.6.3.5. Beacons

Un Smartphone que cuenta con una aplicación rastreadora de señales de Beacon abre un mundo de posibilidades para los compradores, cada vez que un usuario entra a una tienda que cuente con estos dispositivos, puede recibir notificaciones o información de todo tipo: cupones de descuentos, catálogos, ofertas e información adicional que pudiera serle de utilidad al momento de hacer compras en el lugar [58].

## 2.7. Servidor Web asíncrono

En las aplicaciones tradicionales basadas en Web, la entrada de un usuario desencadena una serie de solicitudes de recursos. Una vez que el servidor ha respondido a las solicitudes, no se produce ninguna otra comunicación hasta la siguiente entrada del usuario. Tal comunicación entre el cliente y el servidor se conoce como comunicación síncrona [86].

Un ejemplo de comunicación síncrona tradicional que pasa entre un navegador y un servidor Web [86]:

1. El usuario hace clic en un control de la interfaz de usuario (User Interface, por sus siglas en inglés UI) en una aplicación Web basada en navegador.
2. El navegador convierte la acción del usuario en una o más solicitudes del Protocolo de Transferencia de HiperTexto (HyperText Transfer Protocol, por sus siglas en inglés HTTP) y las pasa al servidor de aplicaciones Web.
3. El servidor de aplicaciones responde a las solicitudes del usuario devolviendo los datos solicitados al usuario. En este punto, la aplicación se actualiza y el ciclo de comunicación síncrono está completo. Un nuevo bucle de comunicación síncrona comenzará cuando el usuario haga clic en un control de UI en su navegador.

La comunicación síncrona está limitada debido a las fallas en las actualizaciones de la aplicación que se presentan al usuario a intervalos regulares. Incluso si una aplicación síncrona está diseñada para que actualice automáticamente la información del servidor de aplicaciones a intervalos regulares (por ejemplo, cada 12 segundos), aún habrá períodos de demora consistentes entre las actualizaciones de datos. Para muchas aplicaciones, dichos retrasos de actualización no presentan un problema porque los datos que administran no cambian con frecuencia. Sin embargo, algunos tipos de aplicaciones, por ejemplo, las aplicaciones de negociación de acciones, se basan en información continuamente actualizada para proporcionar una funcionalidad óptima y facilidad de uso a sus usuarios [86].

Las aplicaciones basadas en la Web abordan este problema al confiar en la comunicación asíncrona. Las aplicaciones asíncronas entregan datos de aplicaciones actualizados continuamente a los usuarios. Esto se logra separando las solicitudes de los clientes de las actualizaciones de las aplicaciones. Múltiples comu-

---

nicaciones asíncronas entre el cliente y el servidor pueden ocurrir simultáneamente o en paralelo [86].

Si bien la comunicación asíncrona ofrece un gran valor para los usuarios, presenta un serio desafío para los proveedores de herramientas de prueba de software que tienen dificultades para emularla con los scripts de prueba tradicionales [86].

# CAPÍTULO 3

---

## Bosquejo general de la solución

---

Este capítulo presenta la explicación de los análisis de factibilidad tanto técnica como económica del proyecto, la arquitectura general del sistema, las arquitecturas de ambas aplicaciones móviles, los componentes o subsistemas que la conforman y los requerimientos funcionales y no funcionales del sistema.

### 3.1. Análisis de factibilidad

El análisis de factibilidad permite evaluar y determinar si se cuenta o no con los elementos técnicos y económicos necesarios para continuar con el desarrollo de este trabajo y de esta manera tomar la decisión de proceder o no con el proyecto planteado [87]. A continuación se muestran los dos análisis realizados.

#### 3.1.1. Factibilidad técnica

La factibilidad técnica nos permite conocer las herramientas tecnológicas con las que ya se cuenta y las que son necesarias para el cumplimiento de los requerimientos solicitados para el trabajo terminal [88]. Como se ha explicado con anterioridad, entre los módulos del proyecto propuesto se encuentra el desarrollo del sistema de recomendaciones, así como aplicaciones móviles que requieren de ciertos componentes de hardware y software de un rendimiento superior al promedio con los cuales se pueda reducir los tiempos de ejecución durante cada prueba que se realice. Actualmente el equipo cuenta con el hardware presentado en la parte inferior.

##### 1. Hardware

Computadora 1:

- Procesador Intel Core -i5-8250U
- Memoria RAM (Random Access Memory) 8 GB (GigaByte) 3.4 GHz (GigaHertz)
- Tarjeta NVIDIA GEFORCE 930mx
- Disco Duro 1TB (TeraByte)

Computadora 2:

- 
- Procesador AMD (Advanced Micro Devices) Rysen 7 1700
  - Memoria RAM 16 GB 3.2 GHz
  - Tarjeta GTX (Game Tested eXtreme) 1060 Strix DDR5X (Double Data Rate 5 X)
  - Disco Duro 1TB

Computadora 3:

- Procesador Intel Core i5 doble núcleo 2.5 GHz
- Memoria RAM 4GB
- Tarjeta HD (High Definition) Graphics 4000 de Intel
- Disco Duro 500 GB

## 2. Software

Modelo de datos

- Postgresql, versión: 9.6.8

Panel de Administración:

- Node.js , versión: 8.9.4
- Express, versión: 4.15.5
- Pug, versión: 2.0.0
- Angular JS (Angular JavaScript), versión: 1.6.9
- Angular Resource, versión: 1.6.9
- Ngmap, versión: 1.18.4

Módulo de Gestión, Procesamiento y Proveedor de datos Retail

- Python, versión: 3.6.4
- Tornado, versión: 5.0
- queries, versión: 2.0.0

Aplicación Interactiva Difusora de Productos

- Java, versión:1.8.0\_161
- Android Icecream Sandwich, versión:  $\geq$  4.0
- Facebook-android-sdk, versión: 4.5
- Estimote:proximity-sdk, versión:0.4.1

Aplicación Interactiva para el Personal de Ventas

- openjdk, versión: 1.8.0\_152
- Android Jelly Bean, versión:  $\geq$  4.3
- Estimote:proximity-sdk, versión:0.4.1
- gson, versión: 2.8.2
- retrofit, versión: 2.3.0
- butterKnine, versión: 8.8.1
- playServices, versión: 11.8.0

### 3.1.2. Factibilidad económica

Por otro lado, la factibilidad económica nos permitirá estimar tanto los costos de producción del proyecto, que serán divididos en 3 partes, los costos de equipo, los costos de servicios y los sueldos de los empleados que en este caso, son los miembros del equipo; y por otra parte, también se estima el precio en el que finalmente se vendería este sistema [88].

El cuadro 3.1 muestra los costos de equipo necesarios para la realización del sistema propuesto. Para este análisis de costo de equipo, se tomarán en cuenta las 3 computadoras con las que se disponen actualmente, así mismo, se analizará la devaluación de cada uno con respecto al tiempo de desarrollo del proyecto mismo que se ha fijado a 10 meses. Se tomará en cuenta que según como hace mención la Ley de Impuesto Sobre la Renta (LISR) en sus artículos 33,34 y 35, la depreciación para equipos de cómputo por año equivale al 30 % del precio inicial [89].

Equipo	Características	Valor de adquisición	Depreciación anual	Depreciación a 10 meses	Valor depreciado
ASUS VivoBook S	Procesador: Intel Core i5-8250U RAM: 8 GB Disco Duro: 1TB	\$15,200.00	\$3,040.00	\$2,533.33	\$12,666.67
Computadora de escritorio	Procesador: AMD Rysen 7 1700 RAM: 16 GB Disco Duro: 1TB	\$37,200.00	\$7,440.00	\$6,200.00	\$31,000.00
MacBook Pro	Procesador: Intel Core -i5 doble núcleo a 2.5 GHz RAM: 4 GB Disco Duro: 500 GB	\$18,000.00	\$3,600.00	\$3,000.00	\$15,000.00
Beacons	Tipo: Proximidad Duración de batería: 2 años Rango de alcance: 70 metros	\$1,103,89 × 3 \$367.96 c/u	\$39.796	\$33.16	\$1,004.40 ×3
			TOTAL	\$12,432.81	\$59,671.07

Cuadro 3.1: Cuadro de costos de equipo.

El análisis de los costos de servicios abarca como su nombre lo dice, los servicios necesarios y entre ellos algunos que son vitales para concretar de manera adecuada cada prototipo. Se tomarán en cuenta 3 servicios principalmente; luz, internet y agua, mismos que se muestran en el cuadro 3.2 y para ello, se obtienen las tarifas de industria de las páginas del Sistema de Aguas de la Ciudad de México (SACMEX), la Comisión Federal de Electricidad (CFE) y Teléfonos Mexicanos (TELME) [90] - [92]. En el caso del cobro de la luz, se obtienen los siguientes datos:

Una computadora consume en promedio .525 kW/h (kiloWatt por hora), tomando en cuenta que son 8 hrs laborales y a su vez son 5 días a la semana de trabajo durante el mes se obtiene:  $0.525\text{ kW/h} \times 8\text{ h} \times 20\text{ días} = 84\text{ kW/h consumidos en el mes}$ . Según la información encontrada en la página de la CFE, en la categoría tarifaría de "Negocio", la tarifa para el consumo de más de 25kW al mes es de \$3.6 por lo tanto el total a pagar al mes por dicho consumo equivale a  $84\text{ kW} \times \$3.6 = \$302.4$  al mes por computadora.

Servicio	Costo Mensual	Costo a 10 meses
Luz	\$907.20	\$9072.00
Internet	\$404.84 con impuestos	\$4,048.40
Agua	$\$412.37 \times 2 / \$206.185 \text{ mes}$	\$2061.85
Total	-	\$15,182.25

Cuadro 3.2: Cuadro de costos de servicios.

En adición, en el cuadro 3.3 se muestran los empleados necesarios para el desarrollo del trabajo terminal, los sueldos y el periodo de empleo correspondientes a cada uno de ellos. Como apoyo para obtener los sueldos mensuales, se consultó un sitio Web en el cual se proporcionan el sueldo más alto, medio y más bajo por profesión, y se tomó como referencia el sueldo medio en México [93].

Cantidad	Empleado	Sueldo mensual	Sueldo mensual general	Periodo de empleo	Sueldo total
2	Analista	\$7,475.91	\$14,951.83	3 meses	\$44,855.49
1	Diseñador	\$7,611.75	\$7,611.75	2 meses	\$15,223.50
2	Programador back-end	\$4,500.00	\$9,000.00	6 meses	\$54,000.00
2	Programador front-end	\$4,747.41	\$9,494.83	6 meses	\$56,968.98
2	Desarrollador Android	\$3,489.75	\$6,979.50	6 meses	\$41,877.00
2	Tester	\$5,425.25	\$10,850.50	7 meses	\$75,953.50
				TOTAL	\$288,878.47

Cuadro 3.3: Cuadro de salarios para empleados requeridos en el proyecto.

Finalmente, a partir de toda la información presentada anteriormente, en el cuadro 3.4 se muestra el costo total de la aplicación.

Análisis	Costo
Costo de equipo	\$12,432.81
Costo de servicio	\$15,182.25
Sueldos	\$288,878.47
TOTAL	\$316,493.53

Cuadro 3.4: Cuadro de costo total de la aplicación.

### 3.2. Elección de tecnologías

La explicación de la elección de las tecnologías se encuentra dividida en 3 secciones: gestor de base de datos, sistema operativo para las aplicaciones interactivas y lenguajes de programación para el desarrollo de los servidores.

El cuadro 3.5 muestra de forma comparativa, 3 posibles opciones de sistemas gestores de bases de datos relacionales.

<b>SQL Server</b>	<b>Mysql</b>	<b>PostgreSQL</b>
Desarrollado por Microsoft.	Desarrollado por Oracle.	Desarrollado por PostgreSQL Global Development Group.
Licencia comercial.	Licencia de código abierto.	Licencia de código abierto.
Utiliza un motor que es un poco más lento y pesado en recursos, pero cumple totalmente con las características de los parámetros que permiten clasificar las transacciones de los sistemas de gestión de bases de datos. Atomicidad, consistencia, aislamiento y durabilidad.	Compatible con una amplia gama de motores e interfaces; una de las bases de datos más maduras en el mercado.	Se adhiere bien a los estándares actuales de SQL, como resultado es más fácil de aprender.
Sistema de informes extremadamente completo / personalización de almacenamiento.	Ligero.	Implementa funciones avanzadas de análisis de negocios y de ubicación.
Alto grado de control sobre transacciones y procedimientos.	Una de las herramientas de base de datos más populares; fácil de encontrar soporte en línea.	Gran variedad de datos y tipos de caracteres.
Puede ajustar las funciones de seguridad, como quién puede ejecutar cada procedimiento almacenado, quién puede acceder a los datos, etc.	Seguridad limitada en comparación con algunos otros sistemas de bases de datos.	La replicación de datos está pobremente implementada.
Los cambios de esquema no bloquean las tablas.	Experimenta una degradación significativa del rendimiento a gran escala.	No es adecuado para proyectos de baja concurrencia.

Cuadro 3.5: Comparación de sistemas gestores de bases de datos [61] [62].

Para el desarrollo del proyecto se decidió elegir PostgreSQL por 2 aspectos importantes, en primera, cuenta con una gran cantidad de tipos de datos, entre los cuales se encuentran arreglos, JSON (JavaScript Object Notation) y uno en especial de tipo geométrico, Point, para guardar coordenadas de tipo (x,y) como números de coma flotante, lo cual nos ayuda bastante para guardar la geolocalización de los Beacons disponibles.

Por otro lado, posee una gran escalabilidad. Es capaz de ajustarse al número de Unidades de Procesamiento Central (Central Processing Unit, por sus siglas en inglés CPU) y a la cantidad de memoria que posee el sistema de forma óptima, haciéndole capaz de soportar una mayor cantidad de peticiones simultáneas de manera correcta [61].

Para la elección del sistema operativo móvil, en el cuadro 3.6 se muestran algunas ventajas y desventajas del desarrollo de aplicaciones móviles para dispositivos con sistemas operativo iOS y Android.

<b>iOS</b>	<b>Android</b>
Lenguaje: Objective-C / Swift.	Lenguaje: Java.
Última versión: iOS 11.3.1 - 24 de abril de 2018.	Última versión: Android 8.1.0 "Oreo" - 5 de febrero de 2018.
Fuerte cuota de mercado, especialmente en los EE. UU.	Mayor penetración de mercado.
La audiencia es más valiosa.	Proceso de revisión de aplicaciones cortas. El proceso de aprobación y publicación de aplicaciones tarda en promedio 4 horas.
El sistema operativo iOS es más férreo y rígido que Android, al tener más controlado el firmware permite que este sea más seguro que su rival.	Android actualmente posee la mayor plataforma global compartida.
Los usuarios de iOS suelen ser un poco más jóvenes, gastar más dinero en compras desde una aplicación y, por lo general, descargan aplicaciones de negocios, educación y estilo de vida de la tienda de aplicaciones. Gastan un promedio de 4 veces más en una aplicación que los usuarios de Android.	Los usuarios de Android son mucho más conscientes de la relación precio-calidad y descargan aplicaciones que van desde herramientas hasta entretenimiento y categorías de comunicación. Además, prefieren los anuncios integrados en la aplicación que pagar un determinado precio por la misma.
iOS cuenta con una tasa de fragmentación realmente baja. Los datos revelan que casi el 90 % de los usuarios ya tiene iOS 10, es decir, 9 de cada 10 dispositivos que permite el sistema operativo ya poseen la última edición.	Al contar con un gran número de fabricantes que usan el sistema operativo en sus dispositivos, se cuenta con distintas formas y tamaños así como grandes diferencias a nivel de desempeño y tamaños de pantalla. Debido a esto hay muchas versiones activas y para desarrollar una aplicación que sea compatible con todos estos dispositivos es una tarea complicada. La fragmentación da lugar a que muchas marcas personalicen sus propias versiones para ofrecer contenido exclusivo.

Cuadro 3.6: Comparación de sistemas operativos móviles [63] - [67].

La razón principal por la que se optó a usar Android es que cuenta con el 85 % del mercado mundial de teléfonos inteligentes, siguiéndole iOS con el 14.7 % y el 0.3 % restante pertenece a otros [57].

Finalmente, en el cuadro 3.7 se comparan 3 de los lenguajes de programación más ocupados en servidores.

PHP	Python	Java
PHP es un lenguaje orientado a objetos, se basa en objetos a los que se adjuntan ciertas propiedades, parámetros y métodos. El usuario escribe comandos a los objetos en la interfaz, a los cuales reaccionan los objetos, comenzando ciertas acciones.	Python se atribuye a los lenguajes orientados a aspectos: significa que la escritura del código en Python se reduce a escribir sus módulos individuales, la conexión entre ellos es posteriormente determinada y controlada por las acciones del usuario en la parte frontend.	Lenguaje multi-paradigmático. Combina las características de: <b>orientada a objetos; orientada a componentes</b> : lenguajes basados en la creación de componentes; <b>imperativo</b> : una clase de lenguajes cuya característica es la ejecución estrictamente secuencial de las instrucciones; <b>estructurales</b> : consta de tres estructuras básicas de control: secuencia, bifurcación y ciclo; <b>reflexivo</b> : una clase de idiomas que puede rastrear y cambiar su propia estructura.
En PHP, las bibliotecas y archivos adicionales se cargan manualmente, lo que se considera una característica del lenguaje muy desagradable.	En Python, el proceso de cargar archivos y bibliotecas adicionales se reduce a un simple conjunto de acciones: mover el archivo a la carpeta del programa y escribir algunas líneas de código.	Las aplicaciones se pueden compilar en el bytecode intermedio que luego puede ser interpretado por la máquina virtual Java de la plataforma específica.
PHP no tiene reglas como lenguajes compilados o estándares estrictos como se ve con Python, sino más bien estándares disponibles de la comunidad de desarrolladores.	El código de Python se asemeja al pseudocódigo al igual que todos los lenguajes de scripting. El diseño elegante y las reglas de sintaxis de este lenguaje de programación lo hace bastante legible.	Java también puede requerir muchos recursos, por lo tanto, más memoria, por ejemplo, en comparación con otros lenguajes.
El manejo de errores es tradicionalmente pobre. Los parámetros de configuración global pueden cambiar la semántica del lenguaje, lo que complica la implementación y la portabilidad.	Sintaxis legible y organizada. Ofrece prototipado rápido y capacidades semánticas dinámicas.	Java es independiente de la plataforma, gracias a maquina virtual Java Virtual Machine (JVM, por sus siglas en inglés), una máquina de computación abstracta que convierte el código fuente Java a código máquina.
Generalmente se considera menos seguro que los otros lenguajes de programación.	Es notoriamente difícil de escalar a través de múltiples núcleos en una sola máquina. Debido a las limitaciones del Global Interpreter Lock (GIL). Sin embargo, es muy adecuado para aplicaciones que se escalan horizontalmente a través de servidores sin estado, es una buena solución para aplicaciones que aprovechan la nube.	Java no cambia mucho entre versiones, es fácil de mantener y avanzar compatible con sus versiones futuras.

Cuadro 3.7: Comparación de lenguajes de programación ocupados en servidores [68] - [71].

Para los servidores se eligió el lenguaje Python debido a que existe una gran cantidad de comunidad que ha generado librerías y contenido de forma pública, por mencionar algunas:

- **Numpy**: es una librería de computación numérica que permite manejar los vectores de una forma muy fácil y eficiente.
- **Scikit-learn**: una librería de aprendizaje maquina en Python que se usará como parte importante en el módulo de recomendaciones para el algoritmo de agrupamiento k-means.
- **Tensorflow y Theano**: Para hacer el procesamiento rápidamente de los algoritmos dentro de la unidad de procesamiento de gráficos y trabajar con grandes cantidades de datos.

### 3.3. Análisis de riesgos

En los cuadros mostrados en la parte inferior se enlistan todos los riesgos potenciales presentes dentro del proyecto que podrían retrasar el tiempo de entrega o afectar directamente la elaboración de alguno de los prototipos planteados. Entre los riegos que se presentan, se consideran riesgos de tipo técnico, de negocio y del proyecto y de igual manera se enlistan las probabilidad de que estos sucedan utilizando por convención los siguientes valores [94]:

- Muy Bajo ( 10 %)
- Bajo (< 10 % – > 25 %)
- Medio (25 % – 50 %)
- Alto (50 % – 75 %)
- Muy Alto ( 75 %)

En el cuadro 3.8 y 3.9 se muestran los riesgos de negocio, estos se refieren a los peligros que corre el proyecto con respecto al entorno tanto económico como social en el que se desarrollará.

<b>Tipo de riesgo</b>	<b>Riesgo</b>	<b>Probabilidad</b>	<b>Impacto</b>	<b>Plan de acción</b>
Riesgo de negocio	Competencia respecto a sistemas desarrollados y probados en tiendas departamentales.	25 %	Bajo	Introducir el producto poco a poco en el mercado a pesar de que la respuesta del público no sea inmediata.
Riesgo de negocio	Precio superior respecto a sistemas similares.	10 %	Bajo	Divulgación de los elementos extras con los que cuenta el sistema.
Riesgo de negocio	Precio de venta inferior al requerido para recompensar costos de desarrollo.	15 %	Bajo	Realizar un estudio de mercado con el fin de comprobar si realmente está correcto el precio de venta de la competencia y el nuestro.
Riesgo de negocio	Dificultad para la venta del proyecto.	70.8 %	Alto	Promover los servicios que el sistema ofrece a las diferentes tiendas departamentales y plazas comerciales.
Riesgo de negocio	Poco conocimiento por parte del público sobre los beneficios que los dispositivos Beacon proveen.	29.2 %	Medio	Realizar campañas publicitarias acerca de las ventajas que el uso de los Beacons proporciona.
Riesgo de negocio	Falta de conocimiento de estrategias de venta y promoción del sistema.	60 %	Alto	Contratar o solicitar ayuda de especialistas de ventas.

Cuadro 3.8: Riesgos de negocio que presenta el proyecto.

<b>Tipo de riesgo</b>	<b>Riesgo</b>	<b>Probabilidad</b>	<b>Impacto</b>	<b>Plan de acción</b>
Riesgo de negocio	Poca disposición por parte de los clientes para hacer uso de su información pública de Facebook.	46.2 %	Medio	Distribución de información para hacer del conocimiento de los usuarios la forma en la que sus datos serán utilizados.
Riesgo de negocio	Poca disposición por parte de los clientes para compartir su historial de compras de una tienda departamental.	40 %	Medio	Distribución de información para hacer del conocimiento de los usuarios la forma en la que sus datos serán utilizados.
Riesgo de negocio	Permisos denegados por parte de los clientes para compartir su ubicación a los vendedores dentro de una tienda departamental.	53.8 %	Alto	Dar a conocer el motivo por el cual su ubicación será utilizada únicamente dentro de la tienda.

Cuadro 3.9: Riesgos de negocio que presenta el proyecto.

Por otro lado, el cuadro 3.10 presenta los riesgos de proyecto, los cuales se refieren a los diferentes peligros que pueden encontrar los integrantes del equipo de análisis y desarrollo internamente.

<b>Tipo de riesgo</b>	<b>Riesgo</b>	<b>Probabilidad</b>	<b>Impacto</b>	<b>Plan de acción</b>
Riesgo de proyecto	Pérdida de información y documentación del sistema.	5 %	Bajo	Realizar respaldos diarios de toda la información y actualizaciones hechas al proyecto y a la documentación de este.
Riesgo de proyecto	Incumplimiento de los requerimientos establecidos.	30 %	Medio	Cumplir en tiempo y forma en la medida de lo posible con las actividades planificadas en el cronograma.
Riesgo de proyecto	Falta de presupuesto para adquirir más dispositivos Beacon.	10 %	Bajo	Contabilizar el número de Beacons requeridos para obtener el correcto desempeño de la aplicación y tener en cuenta un poco más del presupuesto que se enfoca a estos dispositivos.
Riesgo de proyecto	No disponibilidad del personal requerido para el desarrollo del proyecto.	30 %	Medio	Anticipar el trabajo en los tiempos libres con el fin de evitar un retraso en caso de que se presente una situación en la cual algún integrante no pueda presentarse para una reunión de trabajo.
Riesgo de proyecto	Necesidad de escalabilidad del sistema.	75 %	Muy Alto	Contemplar el uso de software como bases de datos relacionales orientadas a objetos, que permitan una posterior escalabilidad del sistema.

Cuadro 3.10: Riesgos de proyecto que presenta el proyecto.

---

Finalmente, el cuadro 3.11 muestra los riesgos técnicos, referidos a aquellos que se presentan en el equipo de hardware y/o software utilizados para la elaboración del sistema.

<b>Tipo de riesgo</b>	<b>Riesgo</b>	<b>Probabilidad</b>	<b>Impacto</b>	<b>Plan de acción</b>
Riesgo técnico	Adquisición de licencias de desarrollo no contempladas.	10 %	Muy bajo	Contemplar el presupuesto requerido para la adquisición de licencias que se llegaran a necesitar.
Riesgo técnico	Falla o baja batería de dispositivos Beacon.	20 %	Bajo	Contemplar desde un inicio un porcentaje del presupuesto a fin de solicitar posteriormente más dispositivos. De igual manera, se deberá contemplar el tiempo que tarda el envío de dichos componentes.
Riesgo técnico	Pérdida parcial o total de la información personal de los usuarios.	15 %	Bajo	Respaldar cada 3 días los datos e información recopilada durante las jornadas.
Riesgo técnico	Falla o pérdida de alguna de las computadoras contempladas para el desarrollo del sistema.	50 %	Medio	Respaldar continuamente la información almacenada en cada una de los equipos de cómputo. Así mismo, conocer los tiempos de disponibilidad de los laboratorios de la escuela en caso de requerir su utilización.

Cuadro 3.11: Riesgos técnicos que presenta el proyecto.

### 3.4. Arquitectura de la aplicación

La figura 3.1 muestra la arquitectura general de la aplicación. En ella se pueden observar los diferentes módulos a desarrollar iniciando por las aplicaciones móviles para el cliente y el vendedor de las tiendas que se conectan a los Servicios REST del módulo de Gestión, Procesamiento y Proveedor de datos de Retail; de igual manera, de lado izquierdo se observa el panel de administración que hace uso de dichos servicios REST y estos a su vez realizan la conexión con el repositorio de datos de donde el módulo Generador de Registros Artificiales obtendrá la información.

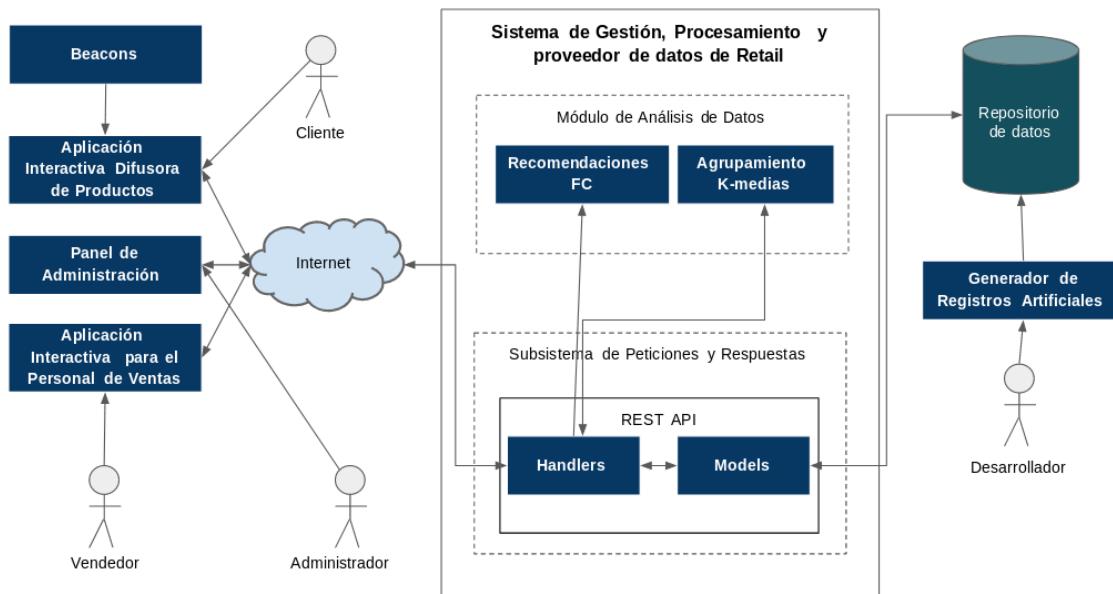


Figura 3.1: Arquitectura interna del sistema.

- **Panel de Administración.** Panel de administración enfocado para los usuarios administradores con el objetivo de proporcionar funcionalidades como: Alta de promociones y folletos, monitorear geográficamente Beacons y tiendas, actualizar información referente a éstos y ejecutar el algoritmo de K-medias para mejorar el clustering de clientes.
- **Aplicación Interactiva Difusora de Productos.** Prototipo de una aplicación móvil para los clientes en la cual se recibirán además de recomendaciones, anuncios, promociones y notificaciones.
- **Aplicación Interactiva para el Personal de Ventas.** Prototipo de una aplicación móvil para los vendedores de la tienda con la que podrán dar una atención personalizada a los clientes ofreciendo recomendaciones de productos que sean de su interés o simplemente asesorarlos en su decisión de compra.
- **Módulo de Gestión, Procesamiento y Proveedor de datos de Retail.**
  - **Subsistema de peticiones y respuestas.**
  - **REST API.** Módulo encargado de manejar las peticiones y respuestas de las diferentes aplicaciones cliente.
  - **Módulo de análisis de datos.**
    - **Algoritmo de agrupamiento K-medias.** Módulo que agrupa a los usuarios automáticamente dependiendo de sus características y gustos.

- **Algoritmo de recomendaciones CF.** Módulo encargado de dar las recomendaciones basado en filtrado colaborativo sobre los usuarios al que cada usuario pertenece.
- **Repositorio de datos.** Repositorio de datos PostgreSQL que almacena los datos del sistema.
- **Generador de registros artificiales.** Módulo encargado de insertar registros ficticios en las diferentes entidades dentro del repositorio de datos del sistema.

### 3.5. Diccionario de actores

A continuación se describen los diferentes tipos de usuarios que se encuentran dentro del funcionamiento del sistema.

- Usuario cliente: Se refiere a todo cliente o consumidor de una tienda departamental o comercio el cual interactuará directamente con la aplicación móvil difusora de productos.
- Usuario vendedor: Se refiere a los encargados de proporcionar información y atender presencialmente a los clientes de un establecimiento.
- Usuario administrador: Se define como usuario administrador a aquellas personas encargadas de seleccionar los productos que se encuentran con promociones y a su vez divulgarlos por medio de un folleto informativo.
- Usuario desarrollador: Se define a este usuario como los integrantes del equipo Sapphire, encargado de ejecutar el generador de registros artificiales y de evaluar las pruebas dentro del guión de pruebas.

### 3.6. Diagrama de casos de uso general

La siguiente figura 3.2, muestra el diagrama de casos de uso en el cual se observa de forma general las acciones que tendrán permitidas los actores previamente definidos.

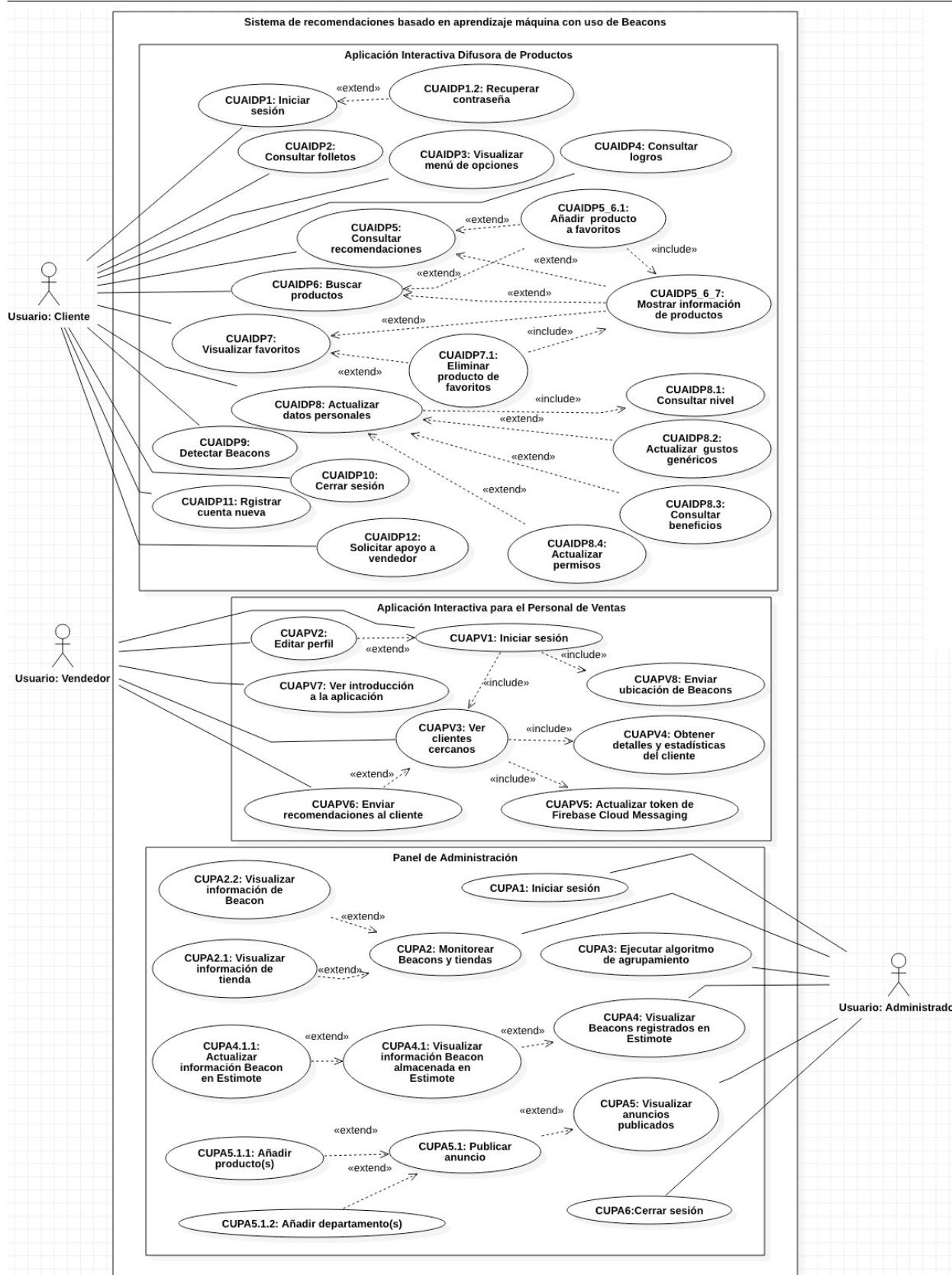


Figura 3.2: Diagrama de casos de uso general.

### 3.7. Requerimientos Funcionales (RF)

A continuación se muestran del cuadro 3.12 al cuadro 3.16, los requerimientos funcionales que están compuestos tanto por las aplicaciones móviles como por los servidores y los cuales consisten en las funcionalidades requeridas por cada uno de los módulos mencionados con el fin de que el proyecto en su totalidad funcione adecuadamente.

#### Requerimientos Funcionales del Generador de Registros Artificiales (RFGRA)

Id	Nombre	Descripción
RFGRA1	Menú principal	Debe existir un menú principal que le permita al usuario desarrollador seleccionar el tipo de acción a realizar.
RFGRA2	Generar persona	Se debe permitir crear automáticamente personas ficticias y registrarlas en el repositorio del sistema.
RFGRA3	Generar dirección	El módulo generador de registros artificiales debe permitir crear automáticamente direcciones ficticias que existan en dentro del territorio de la República Mexicana para posteriormente registrarlas en el repositorio de datos.
RFGRA4	Generar tienda	El módulo generador de registros artificiales permitirá insertar en el repositorio de datos tiendas ficticias.
RFGRA5	Generar departamento	Se debe permitir generar automáticamente departamento asociados a una tienda.
RFGRA6	Generar productos	Dentro del módulo generador de registros artificiales debe existir la opción para crear automáticamente productos ficticios e insertarlos en el repositorio de datos del sistema.
RFGRA7	Generar imágenes pertenecientes a productos	El módulo debe contener la funcionalidad para crear imágenes ficticias de productos que serán agregadas al repositorio de datos con el fin de hacer los registros persistentes.
RFGRA8	Asignar imágenes a productos	El módulo debe poder asignar automáticamente imágenes a productos que estén registrados en el repositorio de datos y así mismo, almacenarlas en dicho repositorio.
RFGRA9	Generar empleados	Se debe permitir generar automáticamente y almacenar en el repositorio de datos, empleados que trabajen en alguna tienda registrada en el sistema con respecto a las personas que existen en el repositorio de datos.
RFGRA10	Generar clientes	Debe existir la posibilidad de generar clientes ficticios con respecto a las personas ya registradas en el repositorio de datos y a su vez, registrar a dichos clientes en el repositorio.
RFGRA11	Asignar productos favoritos a clientes	El módulo en cuestión debe permitir generar registros ficticios de productos añadidos a favoritos por un usuario, mismos que se almacenarán en el repositorio de datos.
RFGRA12	Generar compras a clientes	El Generador de Registros Artificiales debe permitir generar compras ficticias de productos a un usuario cliente, mismos que se almacenarán en el repositorio de datos.
RFGRA13	Generar asignar logros y nivel a clientes	Se debe permitir generar logros y asignar un nivel dependiendo de los movimientos que haya tenido el usuario cliente en el sistema

Cuadro 3.12: Requerimientos Funcionales del Generador de Registros Artificiales.

#### Requerimientos Funcionales de la Aplicación Interactiva Difusora de Productos (RFAIDP)

<b>Id</b>	<b>Nombre</b>	<b>Descripción</b>
RFAIDP1	Iniciar sesión	La aplicación controlará el acceso del cliente a esta mediante el inicio de sesión con la API de Facebook.
RFAIDP2	Consultar folletos	La aplicación mostrará al cliente en la pantalla principal, después de iniciar sesión, los productos que se encuentran actualmente con descuento o las promociones actuales en las tiendas. Dichas promociones serán generados respecto a las diferentes celebraciones del mes (Día de la madre, Navidad, etc.) o por un motivo en particular como su cumpleaños.
RFAIDP3	Visualizar menú de opciones	<p>El cliente podrá visualizar dentro de la aplicación móvil las diferentes opciones que este le ofrece mediante la apertura de un menú lateral. Entre dichas opciones se encuentran las siguientes:</p> <ul style="list-style-type: none"> <li>• Inicio,</li> <li>• Logros,</li> <li>• Favoritos,</li> <li>• Recomendaciones,</li> <li>• Búsqueda de productos,</li> <li>• Ver Perfil,</li> <li>• Cerrar Sesión.</li> </ul> <p>Cada una de dichas opciones deberá direccionar al usuario a su pantalla correspondiente.</p>
RFAIDP4	Consultar logros	<p>Será la pantalla a la cual direccione la opción “Logros” del menú lateral. Un logro será por ejemplo los siguientes:</p> <ul style="list-style-type: none"> <li>• Comprar con tarjeta de crédito.</li> <li>• Comprar un producto en una tienda más de 7 veces.</li> <li>• Realizar una compra igual o mayor a \$3000.00.</li> </ul> <p>Entre otros que serán posteriormente especificados por cada tienda en particular y por los cuales, dicho usuario obtendrá una recompensa por su logro que pueden ser descuentos en sus próximas visitas o meses sin intereses entre otros.</p>
RFAIDP5	Añadir producto a favoritos	El cliente tendrá la opción de almacenar aquellos productos recomendados y/o encontrados que resulten ser de su interés cuando los busque en la sección del menú “Búsqueda de productos”.
RFAIDP6	Visualizar favoritos	El cliente tendrá la oportunidad de visualizar todos los productos que ha añadido previamente a la sección de “Favoritos”.
RFAIDP7	Consultar recomendaciones	El cliente podrá observar al presionar la opción del menú lateral “Recomendaciones”, un listado de productos recomendados específicamente para él, mismos que han sido generadas mediante el algoritmo de recomendación basado en filtrado colaborativo.
RFAIDP8	Buscar productos	El cliente tendrá una sección llamada “Búsqueda de productos” en el menú lateral, en la cuál tendrá la posibilidad de buscar por nombre de producto, cualquier producto que requiera o desee.

Id	Nombre	Descripción
RFAIDP9	Actualizar datos personales	<p>El cliente podrá consultar sus datos personales en la sección “Ver Perfil”, misma que se ha obtenido por default desde su cuenta de Facebook, dicha información involucra su nombre, foto de perfil y correo electrónico. Sin embargo, deberá acompletar su información personal con el fin de categorizarlo correctamente dentro de un grupo con características similares, para ello, deberá incluir:</p> <ul style="list-style-type: none"> <li>• Sexo,</li> <li>• Estado civil,</li> <li>• Fecha de nacimiento.</li> </ul> <p>Esta sección aparecerá en la sección “Ver Perfil” del menú lateral.</p>
RFAIDP10	Consultar nivel	<p>Según los logros que el cliente vaya obteniendo, su nivel irá incrementando dentro de la aplicación, mismo que podrá consultar en su perfil.</p>
RFAIDP11	Actualizar gustos genéricos	<p>Para proporcionar al usuario recomendaciones más acertadas de acuerdo a sus gustos y preferencias, deberá seleccionar de una lista aquellos que se asemejen más a los suyos, entre estos están:</p> <ul style="list-style-type: none"> <li>• Hogar,</li> <li>• Computadoras,</li> <li>• Celulares y tablets,</li> <li>• TV y video,</li> <li>• Cámaras,</li> <li>• Audio,</li> <li>• Videojuegos,</li> <li>• Drones y radio control,</li> <li>• Wearables,</li> <li>• Instrumentos musicales,</li> <li>• Autos, motos y GPS,</li> <li>• Hogar inteligente,</li> <li>• Salud y belleza,</li> <li>• Películas y series.</li> </ul> <p>Esta sección aparecerá en la sección “Ver Perfil” del menú lateral, dentro de una pestaña en la parte superior de esta pantalla.</p>
RFAIDP12	Consultar beneficios	<p>Según el nivel en el que se encuentre el cliente a partir de sus compras, podrá obtener y consultar diferentes beneficios tales como envíos gratis y descuentos en particulares. Esta sección aparecerá en la sección “Ver Perfil” del menú lateral, dentro de una pestaña en la parte superior de esta pantalla.</p>

Id	Nombre	Descripción
RFAIDP13	Actualizar permisos	<p>El cliente tendrá la posibilidad de permitir mostrar a los vendedores parte de su información personal como lo es:</p> <ul style="list-style-type: none"> <li>• Edad</li> <li>• Estado civil</li> <li>• Número de hijos</li> <li>• Productos seleccionados como favoritos</li> <li>• Compras realizadas recientemente</li> </ul> <p>Y de igual manera, permitir a los vendedores acercarse a ellos para ofrecerles distintos productos que puedan ser de su interés. Deberá marcar la casilla correspondiente a los permisos que desee en la sección de "Ver Perfil", en la pestaña de "Permisos".</p>
RFAIDP14	Detectar Beacons	El dispositivo móvil del cliente requiere contar con un BLE que sea capaz de detectar los Beacons que se encontrarán en ciertas zonas de las tiendas. Al captar la señal de alguno de ellos, obtendrá una pequeña alerta con información sobre el piso al que acaba de ingresar, por ejemplo.
RFAIDP15	Cerrar sesión	El cliente podrá cerrar su sesión de Facebook con la cual ingresó a la aplicación móvil con la finalidad de ingresar con cuentas alternativas.
RFAIDP16	Registrar cuenta nueva	<p>El cliente podrá crear una cuenta nueva a partir del ingreso de datos solicitados en el formulario como:</p> <ul style="list-style-type: none"> <li>• Nombre,</li> <li>• Apellido paterno,</li> <li>• Apellido materno,</li> <li>• Correo electrónico,</li> <li>• Contraseña,</li> <li>• Estado civil,</li> <li>• Fecha de nacimiento.</li> </ul>
RFAIDP17	Recuperar contraseña	En caso de olvidar su contraseña, el cliente podrá solicitar que le sea enviada al correo electrónico que introduzca.
RFAIDP18	Eliminar producto de favoritos	En caso de que el cliente haya añadido un producto a la sección de favoritos y no lo desee más, tendrá la opción de eliminar dicho producto de esta sección al presionar sobre algún producto en particular.
RFAIDP19	Mostrar información de productos	Si el cliente presiona sobre un artículo en particular se desplegará una nueva ventana con la información de dicho producto e imágenes de este.
RFAIDP20	Solicitar apoyo a vendedor	Si el cliente presiona sobre la opción "Solicitar apoyo" localizada en el menú de opciones, un vendedor podrá acercarse a él para proveer la ayuda que él requiera.

Cuadro 3.13: Requerimientos Funcionales de la Aplicación Interactiva Difusora de Productos.

#### Requerimientos Funcionales de la Aplicación Interactiva para el Personal de Ventas (RFAIPV)

<b>Id</b>	<b>Nombre</b>	<b>Descripción</b>
RFAPV1	Iniciar sesión	La aplicación controlará el acceso del usuario vendedor por medio del nombre de usuario del vendedor y su respectiva contraseña.
RFAPV2	Enviar de ubicación de beacons	La aplicación deberá detectar y enviar al Sistema de Gestión, Procesamiento y Proveedor de Datos Retail las ubicaciones de los diferentes beacons registrados alrededor del centro comercial.
RFAPV3	Editar perfil del vendedor	La aplicación permitirá al usuario vendedor editar sus datos personales básicos (nombre, apellidos, email y contraseña).
RFAPV4	Ver clientes cercanos	La aplicación permitirá al usuario vendedor visualizar un listado de los clientes cercanos dentro de un comercio o establecimiento con la finalidad de brindarle apoyo personalizado en caso de ser requerido por un usuario cliente o enviar recomendaciones personalizadas del departamento en el que se encuentre el cliente.
RFAIPV5	Obtener detalles y estadísticas de un cliente	La aplicación permitirá al vendedor visualizar detalles de un cliente: sexo, edad, número de hijos, beneficios por su nivel, 5 productos favoritos; así como estadísticas de compra: total de compras realizadas en los últimos 6 meses y las categorías de productos más comprados. Estos datos se mostrarán únicamente si el cliente otorga el permiso de visualización para cada uno de ellos.
RFAIPV6	Enviar recomendaciones a clientes	La aplicación permitirá al vendedor enviar recomendaciones de productos del departamento en el que los clientes se encuentren.
RFAIPV7	Actualizar token de Firebase Cloud Messaging (FCM)	La aplicación permitirá actualizar el token de FCM para suscribirse a las notificaciones de cada uno de los departamentos a los que el vendedor se dirija, es decir, cada que el vendedor cambie de departamento de igual forma lo hará el token.

Cuadro 3.14: Requerimientos Funcionales de la Aplicación Interactiva para el Personal de Ventas.

#### Requerimientos Funcionales del Panel de Administración (RFPA)

<b>Id</b>	<b>Nombre</b>	<b>Descripción</b>
RFPA1	Iniciar sesión	El panel de administración controlará el acceso de un usuario administrador al panel de administración por medio de un nombre de usuario y su respectiva contraseña.
RFPA2	Monitorear Beacons y tiendas	El panel de administración deberá tener un apartado donde se permita plasmar en un mapa la geolocalización de los Beacon y tiendas que se encuentran disponibles por toda la República Mexicana. Así mismo, se mostrará la información básica de dichos Beacons como su id, nombre, batería, departamento al que pertenece; de igual manera, para las tiendas departamentales se mostrará su nombre y su dirección.
RFPA3	Visualizar información de anuncios publicados	El panel de administración deberá tener un apartado donde se permita visualizar la información de los anuncios publicados en el sistema. Por otra parte, se debe permitir registrar nuevos anuncios con sus respectivos productos y departamentos asociados.
RFPA4	Visualizar información de Beacons	El panel de administración deberá contar con un apartado donde se permita ver la información de los Beacon registrados en el sistema. Por otra parte, se debe permitir modificar algunos campos de cada Beacon.

<b>Id</b>	<b>Nombre</b>	<b>Descripción</b>
RFP45	Ejecutar algoritmo de agrupamiento	El panel de administración deberá contar con un apartado en el cuál se permita actualizar los grupos de personas formados anteriormente, además de mostrar la información más relevante de cada grupo.

Cuadro 3.15: Requerimientos Funcionales del Panel de Administración.

**Requerimientos Funcionales del Sistema de Gestión, Procesamiento y Proveedor de datos de Retail (RFGPPR)**

<b>Id</b>	<b>Nombre</b>	<b>Descripción</b>
RFGPPR1	Conectar con módulos basados en aprendizaje máquina	El SGPPDR (Sistema de Gestión, Procesamiento y Proveedor de Datos Retail) se encargará de conectarse con los dos módulos basados en aprendizaje máquina, con el fin de formular una serie de recomendaciones de productos a partir de los ya comprados previamente por el usuario y los comprados por usuarios cliente con características similares a las del usuario cliente en cuestión.
RFGPPR2	Generar módulo clustering	El módulo interno llamado módulo de análisis de datos podrá conformar diferentes grupos de usuarios cliente con base en los gustos y características de éstos para posteriormente almacenar dichas agrupaciones en el repositorio de datos.
RFGPPR3	Generar módulo de recomendaciones basado en filtrado colaborativo	El módulo tomará los registros almacenados previamente por el módulo de clustering y realizará las recomendaciones basado en las compras de los usuarios cliente pertenecientes (también llamado vecindario) al mismo grupo en el que ellos se encuentran.
RFGPPR4	Ejecutar algoritmo de agrupamiento	Debe ser posible ejecutar el algoritmo de agrupamiento sobre los usuarios ya existentes y/o usuarios nuevos.
RFGPPR5	Obtener clusters y características	El sistema debe tener la funcionalidad necesaria para proveer los clusters y sus características principales.
RFGPPR6	Registrar folletos	El SGPPDR debe tener la funcionalidad necesaria para poder hacer el registro de folletos nuevos al sistema.
RFGPPR7	Obtener folletos: Administrador	El SGPPDR debe mostrar los folletos registrados en el sistema.
RFGPPR8	Proveer Beacons registrados	Debe existir la funcionalidad para proveer los Beacons registrados en sistema a la aplicación cliente Panel de Administración.
RFGPPR9	Proveer establecimientos registrados	Debe existir la funcionalidad para proveer los establecimientos registrados en el repositorio de datos a la aplicación cliente Panel de Administración.
RFGPPR10	Proveer departamentos dentro de tienda	El sistema debe tener la funcionalidad necesaria para proveer los departamentos asignados a una tienda o establecimiento.
RFGPPR11	Insertar/Actualizar Beacons del sistema	Debe existir la opción para insertar nuevos Beacons al sistema, por otra parte, si un Beacon ya está registrado se debe poder actualizar su posición actual.
RFGPPR12	Obtener atributos clave-valor de Beacons	Debe ser posible obtener los atributos clave-valor (attachments) de los Beacons registrados desde Estimote Cloud para con ellos se puedan crear las zonas de proximidad.

<b>Id</b>	<b>Nombre</b>	<b>Descripción</b>
RFGPPR13	Proveer token de autenticación: Administrador	El sistema recibirá los datos de autenticación de un usuario administrador para posteriormente devolver un token de autenticación al panel de administración.
RFGPPR14	Proveer token de autenticación: Vendedor	El sistema recibirá los datos de autenticación de un usuario vendedor para posteriormente devolver un token de autenticación a la aplicación interactiva para el personal de ventas.
RFGPPR15	Actualizar token de Firebase Cloud Messaging (FCM) para vendedor	Un usuario vendedor debe poder actualizar su token de Firebase Cloud Messaging para las notificaciones push.
RFGPPR16	Actualizar datos básicos del perfil del vendedor	Debe existir la posibilidad de modificar los datos personales o profesionales de los empleados registrados en el sistema.
RFGPPR17	Obtener clientes cercanos en departamento	Cuando un usuario cliente se encuentre dentro de un departamento asignado a un Beacon el usuario vendedor debe ser capaz de obtener la información de todos los usuarios cliente dentro del área de proximidad del Beacon.
RFGPPR18	Obtener detalles y estadísticas de cliente	El sistema debe devolver al vendedor cuando lo solicite, los detalles y estadísticas a las que el cliente haya otorgado permisos.
RFGPPR19	Obtener recomendaciones por departamento	El sistema debe permitirle a un vendedor ver las recomendaciones de un cliente cuando se encuentre dentro del mismo departamento que el usuario vendedor.
RFGPPR20	Enviar recomendaciones a cliente	El sistema debe permitirle a un vendedor enviar una recomendación de un producto a un cliente por medio de una notificación push usando FCM.
RFGPPR21	Insertar/Actualizar Beacons de desde la AIPV	El sistema debe permitirle a un vendedor actualizar la ubicación de un Beacon para mantener el control de los mismos.
RFGPPR22	Registrar clientes	Debe existir la funcionalidad necesaria para poder hacer el registro de clientes nuevos al sistema.
RFGPPR23	Obtener información del cliente	Un usuario cliente debe tener la posibilidad de obtener su información personal en cualquier momento.
RFGPPR24	Proveer de productos	El SGPPDR recibirá peticiones de productos por parte de ambas aplicaciones móviles para posteriormente devolver los productos incidentes a dicha petición.
RFGPPR25	Proveer de folletos	El módulo deberá proveer a la aplicación interactiva difusora de productos los folletos relevantes en el momento requerido.
RFGPPR26	Agregar producto a favoritos	El módulo debe permitir a un cliente agregar un producto a sus productos favoritos.
RFGPPR27	Actualizar token de Firebase Cloud Messaging para cliente	Un usuario cliente debe poder actualizar su token de Firebase Cloud Messaging para las notificaciones push.
RFGPPR28	Publicar ubicación de cliente en Kafka	Cuando un usuario cliente entre a una zona de proximidad asignada a un Beacon su posición se debe publicar en el sistema.
RFGPPR29	Obtener recomendaciones globales	El sistema debe permitirle a un usuario cliente ver sus recomendaciones de todos los productos registrados en el repositorio de datos.

Id	Nombre	Descripción
----	--------	-------------

Cuadro 3.16: Requerimientos Funcionales del Sistema de Gestión, Procesamiento y Proveedor de datos de Retail.

### 3.8. Requerimientos No Funcionales (RNF)

Los requerimientos no funcionales corresponden a aquellos criterios que son necesarios para medir el desempeño del sistema tanto internamente como con la interacción de este con el usuario. En la parte inferior se muestra en el cuadro 3.17 los requerimientos no funcionales del sistema.

Id	Atributo	Necesidad/Estrategia
RNF1	Eficiencia	<ul style="list-style-type: none"> <li>• Los datos modificados en el repositorio de datos deben ser actualizados para todos los usuarios que acceden al sistema en cuanto actualicen o cambien de pantalla.</li> </ul>
RNF2	Usabilidad	<ul style="list-style-type: none"> <li>• El sistema debe contar con manuales de usuario estructurados adecuadamente.</li> <li>• El sistema debe proporcionar mensajes de error que sean informativos y orientados a usuario final.</li> <li>• El sistema debe contar con un módulo de ayuda en línea.</li> <li>• La aplicación Web debe poseer un diseño “responsive” a fin de garantizar la adecuada visualización en múltiples computadores personales, dispositivos tableta y teléfonos inteligentes.</li> </ul>
RNF4	Portabilidad	<ul style="list-style-type: none"> <li>• Las aplicaciones móviles funcionarán únicamente en la plataforma Android.</li> <li>• La aplicación móvil debe funcionar a partir de la versión 4.3 de Android.</li> <li>• El panel de administración debe funcionar en las plataformas Windows, Linux y MacOs.</li> <li>• El panel de administración será implementado para navegadores Web únicamente con HTML5 y Javascript.</li> </ul>
RNF5	Diseño responsivo	<ul style="list-style-type: none"> <li>• EL Panel de Administración debe poseer un diseño “responsive” a fin de garantizar la adecuada visualización en múltiples computadores personales y dispositivos tableta.</li> </ul>

Cuadro 3.17: Requerimientos No Funcionales del sistema.

# CAPÍTULO 4

---

## Modelo de datos

---

Este capítulo muestra el repositorio de datos generado para todo el sistema, esta base de datos estará almacenada en el gestor de base de datos PostgreSQL y como se observa en la figura 4.1, contiene un total de 27 entidades. De igual manera, se presenta el diccionario de datos con los campos que incorporan cada una de dichas entidades y sus respectivas descripciones.

*Nota: Durante el desarrollo de los prototipos se utilizaron diagramas auxiliares con el único objetivo de mostrar los cambios en el modelo de una manera que la visualización sea más evidente. Estos diagramas comienzan en la figura 4.4 hasta la figura 4.12*

### 4.1. Modelo relacional

En esta sección se encuentra el modelo relacional oficial del TT (figura 4.1) que refleja el resultado final del trabajo realizado en los múltiples prototipos que se encuentran en el capítulo en curso. Para una mejor visualización el diagrama se ha dividido en 2 partes.

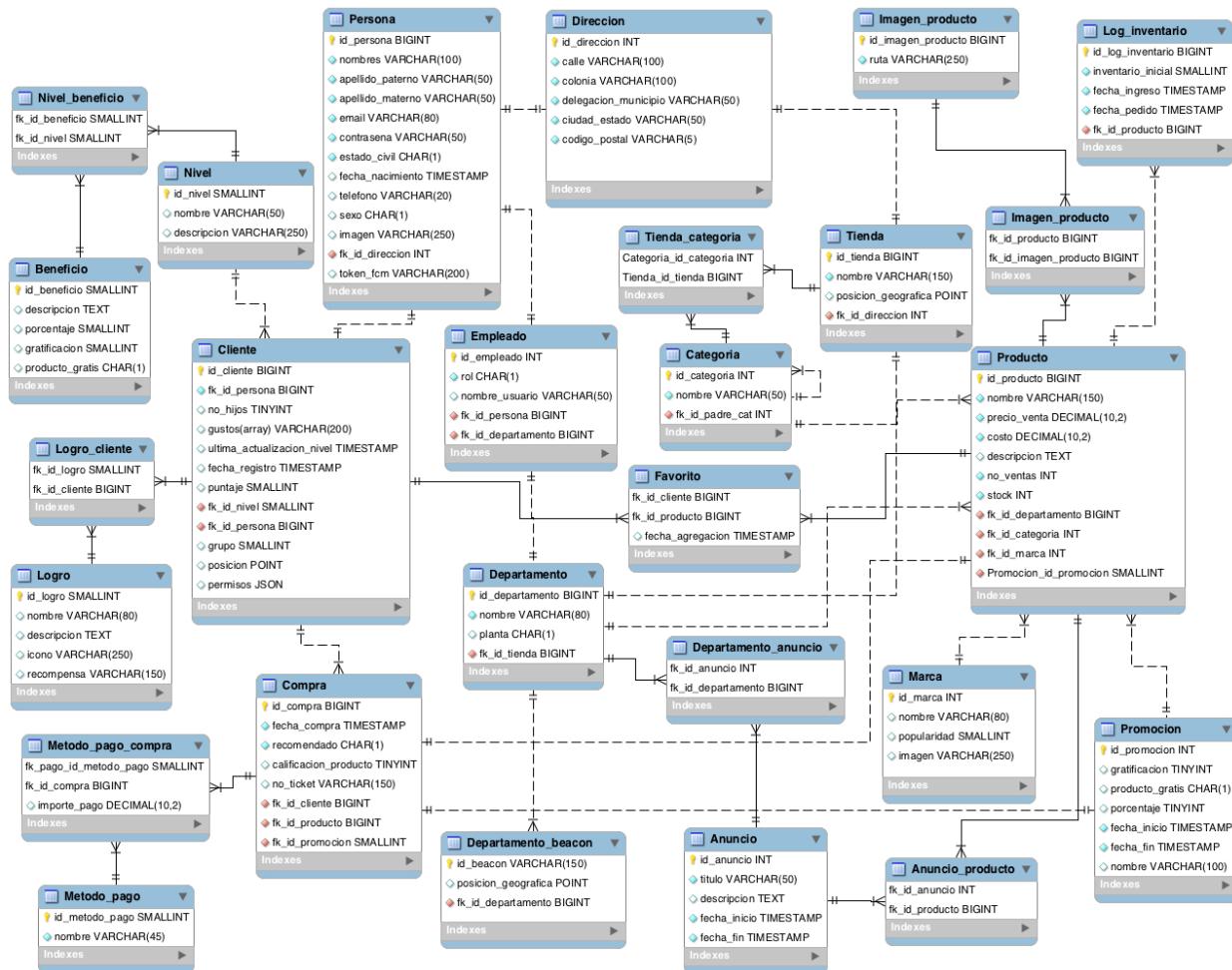


Figura 4.1: Modelo relacional de base de datos (Visualización completa).

---

La figura 4.2 muestra la parte 1 del diagrama.

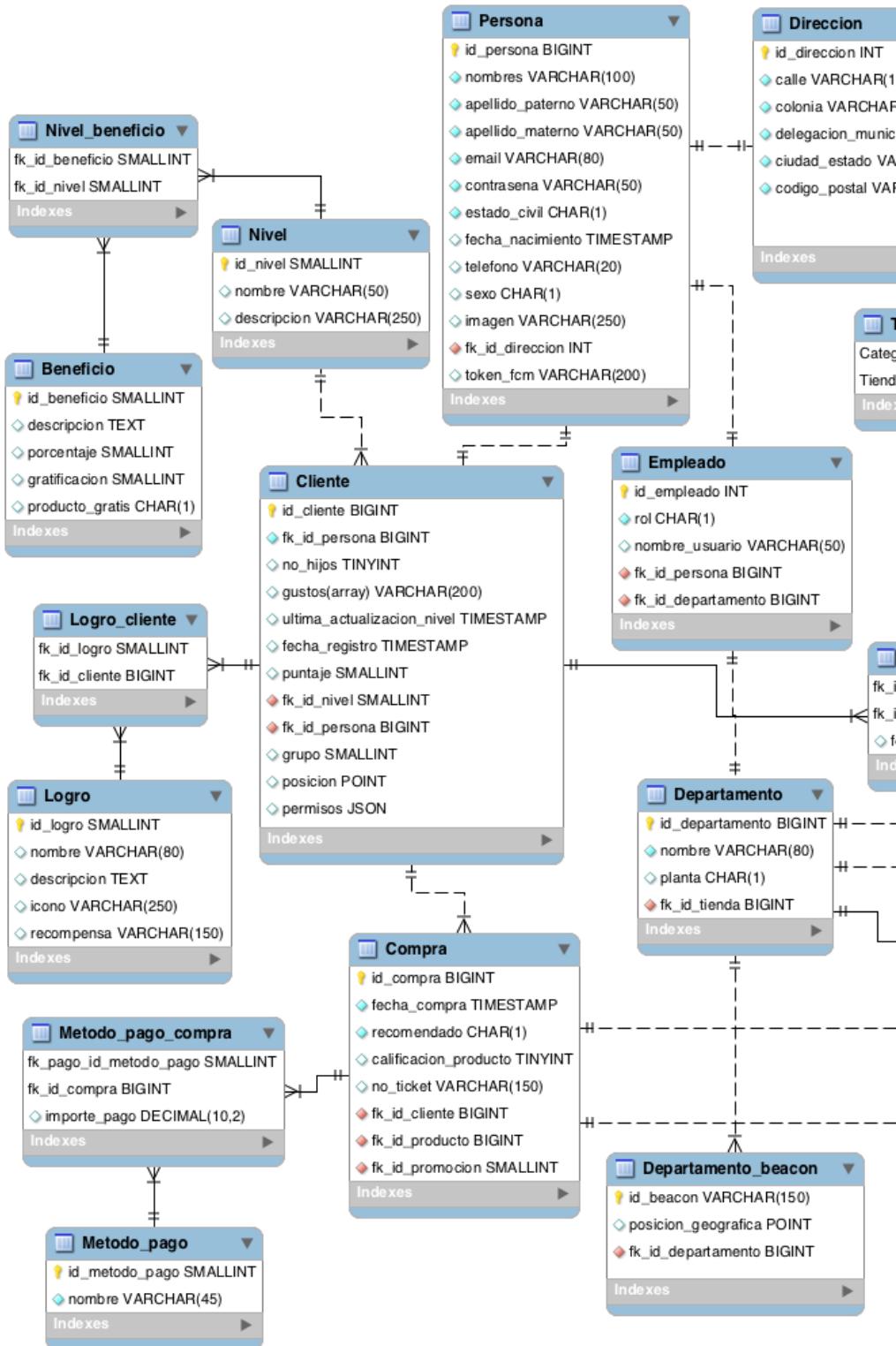


Figura 4.2: Modelo relacional de base de datos (Parte 1).

La figura 4.3 muestra la parte 2 del diagrama.

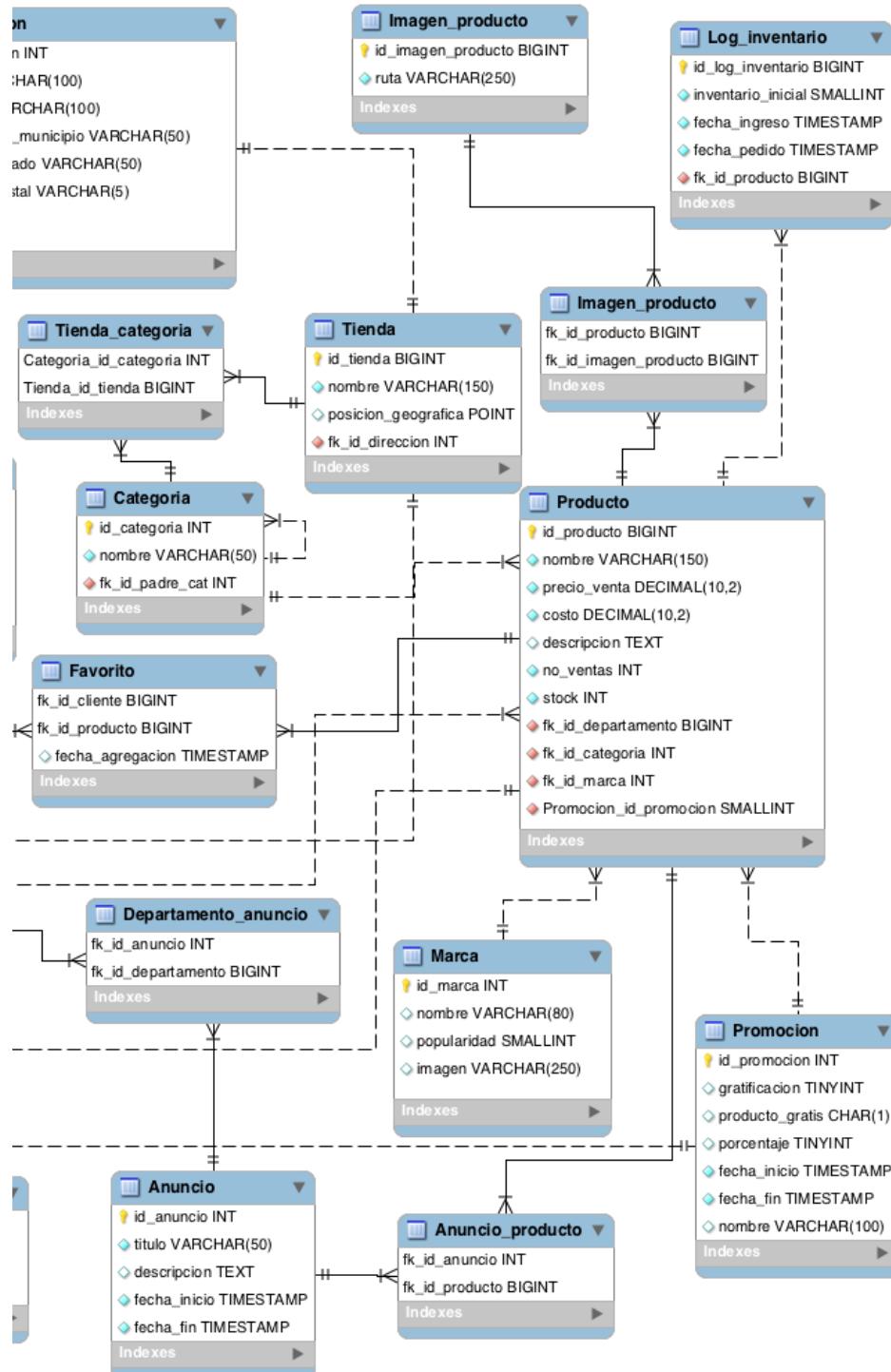


Figura 4.3: Modelo relacional de base de datos (Parte 2).

## 4.2. Prototipo 1: Diseño inicial de la base de datos

### 4.2.1. Diccionario de datos

Las tablas presentadas a continuación muestran el diccionario de datos con la información de cada una de las entidades creadas en la base de datos, los campos que cada una de ellas contiene, su tipo y una pequeña descripción de lo que a cada campo se refiere con el fin de facilitar la comprensión de dicha base.

**Tabla de diccionario de datos Anuncio**

El cuadro 4.1 indica la descripción correspondiente de la entidad Anuncio.

Nombre de la relación:		Anuncio			
Objetivo:		Tabla que almacenará la información sobre los anuncios.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_anuncio	integer	10	PK	X	Identificador del anuncio.
titulo	varchar	50		X	Título del anuncio.
descripcion	text				Breve descripción sobre el anuncio.
fecha_inicio	timestamp			X	Fecha a partir de la cual el anuncio es válido.
fecha_fin	timestamp			X	Fecha a partir de la cual el anuncio deja de ser válido.
ruta_imagen	varchar	250			Almacena la ruta donde está almacenada la imagen.

Cuadro 4.1: Tabla de diccionario de datos Anuncio.

**Tabla de diccionario de datos Atributo**

El cuadro 4.2 indica la descripción correspondiente de la entidad Atributo.

Nombre de la relación:		Atributo			
Objetivo:		Tabla que almacenará los valores de los tipos de atributos que puede tener un producto.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_atributo	integer	10	PK	X	Identificador del atributo.
fk_tipo_atributo	integer	10	FK	X	Identificador del tipo de atributo.
valor	varchar	100		X	Valor específico del tipo de atributo.

Cuadro 4.2: Tabla de diccionario de datos Atributo.

### Tabla de diccionario de datos Beneficio

El cuadro 4.3 indica la descripción correspondiente de la entidad Beneficio.

Nombre de la relación:		Beneficio			
Objetivo:		Tabla que almacenará las marcas que existen en la tienda.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_beneficio	smallint	5	PK	X	Identificador del beneficio.
descripcion	text			X	Descripción del beneficio.
porcentaje	smallint	5		X	Porcentaje de descuento.
gratificacion	smallint	5		X	Porcentaje del precio del producto que se otorgará a un cliente como recompensa de su compra.
producto_gratis	char	1		X	Indica si el beneficio incluye un producto gratis.

Cuadro 4.3: Tabla de diccionario de datos Beneficio.

### Tabla de diccionario de datos Categoría

El cuadro 4.4 indica la descripción correspondiente de la entidad Categoría.

Nombre de la relación:		Categoría			
Objetivo:		Tabla que almacenará los distintos tipos de categorías tanto para productos como para tiendas.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_categoria	integer	10	PK	X	Identificador de la categoría.
fk_id_padre_cat	integer	10	FK	X	Identificador de la categoría padre.
nombre	varchar	50		X	Nombre de la categoría.

Cuadro 4.4: Tabla de diccionario de datos Categoría.

### Tabla de diccionario de datos Cliente

El cuadro 4.5 indica la descripción correspondiente de la entidad Cliente.

Nombre de la relación:		Cliente			
Objetivo:		Tabla que almacenará los datos específicos de un cliente.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_cliente	bigint	20	PK	X	Identificador del cliente.
fk_id_persona	bigint	20	FK	X	Identificador de la persona.
fk_id_nivel	smallint	5	FK	X	Identificador del nivel del cliente.
no_hijo	tinyInt	2			Número de hijos que tiene el cliente.
gustos	varchar[]	100		X	Gustos de la persona almacenados en un arreglo de longitud 100 cada elemento.
ultima_actualizac	timestamp			X	Fecha de la última actualización del nivel del cliente.
fecha_registro	timestamp			X	Fecha en la que se registró el cliente en Sapphire.
puntaje	smallint	2		X	Puntaje que el cliente tiene actualmente, este puntaje se obtiene por las compras o acciones que realice.

Cuadro 4.5: Tabla de diccionario de datos Cliente.

#### Tabla de diccionario de datos Compra

El cuadro 4.6 indica la descripción correspondiente de la entidad Compra.

Nombre de la relación:		Compra			
Objetivo:		Tabla que almacenará los registros del inventario de un producto.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_compra	bigint	20	PK	X	Identificador de la compra.
fk_id_producto	bigint	20	FK	X	Identificador del producto.
fk_id_cliente	bigint	20	FK	X	Identificador del cliente.
fk_id_promocion	smallint	5	FK		Identificador de una posible promoción.
fecha_compra	timestamp			X	Fecha en que se realizó la compra.
recomendando	char	1		X	Indica si el producto comprado ha sido recomendado por el sistema o no.
calificacion_producto	smallint	5			Calificación que el cliente otorga al producto comprado.
no_ticket	varchar	150			Número de ticket generado a partir de las iniciales de la tienda, la fecha de compra en formato Ymd, identificador del cliente y un número de 8 dígitos aleatorio.

Cuadro 4.6: Tabla de diccionario de datos Compra.

#### Tabla de diccionario de datos Departamento\_anuncio

El cuadro 4.7 indica la descripción correspondiente de la entidad Departamento\_anuncio.

Nombre de la relación:		Departamento_anuncio			
Objetivo:		Tabla que almacenará los anuncios de los departamentos.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
fk_id_departame	bigint	20	FK	X	Identificador del departamento.
fk_id_anuncio	smallint	5	FK	X	Identificador del anuncio.

Cuadro 4.7: Tabla de diccionario de datos Departamento\_anuncio.

#### Tabla de diccionario de datos Departamento\_beacon

El cuadro 4.36 indica la descripción correspondiente de la entidad Departamento\_beacon.

Nombre de la relación:		Departamento.beacon			
Objetivo:		Tabla que almacenará los beacons asociados a un departamento.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
fk_id_departame	bigint	20	PK	X	Identificador del departamento.
id_beacon	varchar	50		X	Identificador único del Beacon.
posicion_geografi	point			X	Coordenada geográfica en la que se encuentra el Beacon físicamente.

Cuadro 4.8: Tabla de diccionario de datos Departamento\_beacon.

#### Tabla de diccionario de datos Departamento

El cuadro 4.9 indica la descripción correspondiente de la entidad Departamento.

Nombre de la relación:		Departamento			
Objetivo:		Tabla que almacenará los datos específicos de un departamento dentro de la tienda.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_departamento	bigint	20	PK	X	Identificador del departamento.
fk_id_tienda	bigint	20	FK	X	Identificador de la tienda a la que pertenece el departamento.
nombre	varchar	80		X	Nombre del departamento.
planta	char	1		X	Número de planta en la que se encuentra el departamento.

Cuadro 4.9: Tabla de diccionario de datos Departamento.

#### Tabla de diccionario de datos Dirección\_persona

El cuadro 4.10 indica la descripción correspondiente de la entidad Dirección\_persona.

Nombre de la relación:		Dirección_persona			
Objetivo:		Tabla que almacenará la dirección de una persona.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_direccion	bigint	20	PK	X	Identificador de la dirección del cliente.
calle	varchar	90		X	Nombre de la calle en la que vive el cliente.
colonia	varchar	90		X	Nombre de la colonia en la que vive el cliente.
delegacion	varchar	90			Nombre de la delegación en la que vive el cliente.
ciudad	varchar	70		X	Nombre de la ciudad en la que vive el cliente.
estado	varchar	70		X	Nombre del estado en la que vive el cliente.

Cuadro 4.10: Tabla de diccionario de datos Dirección\_persona.

#### Tabla de diccionario de datos Empleado

El cuadro 4.11 indica la descripción correspondiente de la entidad Empleado.

Nombre de la relación:		Empleado			
Objetivo:		Tabla que almacenará los datos específicos de un empleado.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_empleado	integer	10	PK	X	Identificador del empleado.
fk_id_persona	bigint	20	FK	X	Identificador de la persona.
fk_id_departamento	bigint	20	FK	X	Identificador del departamento al que pertenezca el empleado.
rol	char	1		X	Tipo de rol que tiene el empleado (vendedor/administrador).

Cuadro 4.11: Tabla de diccionario de datos Empleado.

#### Tabla de diccionario de datos Favorito

El cuadro 4.12 indica la descripción correspondiente de la entidad Favorito.

Nombre de la relación:		Favorito			
Objetivo:		Tabla que almacenará la relación entre un cliente y sus productos favoritos.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
fk_id_cliente	bigint	20	PK, FK	X	Identificador del cliente.
fk_id_producto	bigint	20	PK, FK	X	Identificador del producto.
fecha_agregacion	timestamp			X	Fecha en la que se agregó el producto a favoritos.

Cuadro 4.12: Tabla de diccionario de datos Favorito.

#### Tabla de diccionario de datos Imagen\_producto

El cuadro 4.13 indica la descripción correspondiente de la entidad Imagen\_producto.

Nombre de la relación:		Imagen_producto			
Objetivo:		Tabla que almacenará las rutas de las imágenes de productos.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_imagen_produ	bigint	20	PK	X	Identificador de la imagen del producto.
ruta	varchar	250		X	Ruta en el servidor de la ubicación de la imagen.

Cuadro 4.13: Tabla de diccionario de datos Imagen\_producto.

#### Tabla de diccionario de datos Log\_inventario

El cuadro 4.14 indica la descripción correspondiente de la entidad Log\_inventario.

Nombre de la relación:		Log_inventario			
Objetivo:		Tabla que almacenará los registros del inventario de un producto.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_log_inventario	bigint	20	PK	X	Identificador del registro en inventario.
fk_id_producto	bigint	20	FK	X	Identificador del producto.
inventario_inicial	smallint	5		X	Valor inicial del total de stock en inventario del producto.
fecha_ingreso	timestamp			X	Fecha de ingreso de nuevo stock del producto.
fecha_pedido	timestamp			X	Fecha en la que se realizó el pedido.

Cuadro 4.14: Tabla de diccionario de datos Log\_inventario.

#### Tabla de diccionario de datos Logro\_cliente

El cuadro 4.15 indica la descripción correspondiente de la entidad Logro\_cliente.

Nombre de la relación:		Logro_cliente			
Objetivo:		Tabla que relaciona los logros con los clientes.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
fk_id_cliente	bigint	20	PK, FK	X	Identificador del cliente.
fk_id_logro	smallint	5	PK, FK	X	Identificador del logro.

Cuadro 4.15: Tabla de diccionario de datos Logro\_cliente.

#### Tabla de diccionario de datos Logro

El cuadro 4.16 indica la descripción correspondiente de la entidad Logro.

Nombre de la relación:		Logro			
Objetivo:		Tabla que almacenará los logros que tendrán disponibles los clientes.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_logro	smallint	5	PK	X	Identificador del logro.
nombre	varchar	80		X	Nombre del logro.
descripcion	text			X	Descripción del logro.
icono	varchar	250		X	Ruta en la cual se encuentra almacenado el icono del logro.
recompensa	varchar	150		X	Recompensa que incluye el logro.

Cuadro 4.16: Tabla de diccionario de datos Logro.

#### Tabla de diccionario de datos Marca

El cuadro 4.17 indica la descripción correspondiente de la entidad Marca.

Nombre de la relación:		Marca			
Objetivo:		Tabla que almacenará las marcas que existen en la tienda.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_marca	integer	10	PK	X	Identificador de la marca del producto.
nombre	varchar	80		X	Nombre de la marca.
popularidad	smallint	5		X	Entero que representa la popularidad de la marca basado en sus ventas.
imagen	varchar	250		X	Ruta en la cuál se encuentra almacenada la imagen de la marca.

Cuadro 4.17: Tabla de diccionario de datos Marca.

#### Tabla de diccionario de datos Método\_pago\_compra

El cuadro 4.18 indica la descripción correspondiente de la entidad Método\_pago\_compra.

Nombre de la relación:		Método_pago_compra			
Objetivo:		Tabla que almacenará la relación entre una compra y el método de pago utilizado en la misma.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
fk_id_metodo_pag	smallint	5	PK, FK	X	Identificador del método de pago.
fk_id_compra	bigint	20	PK, FK	X	Identificador de la compra.
importe_pago	numeric	10,2		X	Importe que se pago con este método para la compra.

Cuadro 4.18: Tabla de diccionario de datos Método\_pago\_compra.

#### Tabla de diccionario de datos Método\_pago

El cuadro 4.19 indica la descripción correspondiente de la entidad Método\_pago.

Nombre de la relación:		Método_pago			
Objetivo:		Tabla que almacenará los métodos de pago.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_metodo_pago	smallint	5	PK	X	Identificador del método de pago.
nombre	varchar	100		X	Nombre del método de pago.

Cuadro 4.19: Tabla de diccionario de datos Método\_pago.

#### Tabla de diccionario de datos Nivel\_beneficio

El cuadro 4.20 indica la descripción correspondiente de la entidad Nivel\_beneficio.

Nombre de la relación:		Nivel_beneficio			
Objetivo:		Tabla que relaciona los beneficios con un nivel.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
fk_id_nivel	smallint	5	PK, FK	X	Identificador del nivel.
fk_id_beneficio	smallint	5	PK, FK	X	Identificador del beneficio.

Cuadro 4.20: Tabla de diccionario de datos Nivel\_beneficio.

#### Tabla de diccionario de datos Nivel

El cuadro 4.21 indica la descripción correspondiente de la entidad Nivel.

Nombre de la relación:		Nivel			
Objetivo:		Tabla que almacenará los niveles de beneficios disponibles para los clientes.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_nivel	smallint	5	PK	X	Identificador del nivel.
nombre	varchar	100		X	Nombre del nivel.
descripcion	varchar	250		X	Descripción breve del nivel.

Cuadro 4.21: Tabla de diccionario de datos Nivel.

#### Tabla de diccionario de datos Persona

El cuadro 4.22 indica la descripción correspondiente de la entidad Persona.

Nombre de la relación:		Persona			
Objetivo:		Tabla que almacenará los datos generales de una persona.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_persona	bigint	20	PK	X	Identificador de la persona.
nombre	varchar	100		X	Nombre(s) de la persona.
apellido_paterno	varchar	50		X	Apellido paterno de la persona.
apellido_materno	varchar	50		X	Apellido materno de la persona.
email	varchar	80		X	Email de la persona, se conforma de la siguiente expresión regular: ^[_a-zA-Z0-9-]+(.[_a-zA-Z0-9-]+)*@[a-zA-Z0-9-]+\\$. [a-zA-Z0-9-] +)*(. [a-zA-Z]{2,4})\$ .
contrasena	varchar	50		X	Contraseña de la persona.
estado_civil	char	1			Estado civil actual de la persona.
fecha_nacimiento	datetime				Fecha de nacimiento de la persona.
fk_id_direccion	integer	10	FK	X	Identificador de la dirección de la persona.
telefono	varchar	20			Teléfono de contacto principal de la persona.
sexo	tinyint	1		X	Sexo masculino(0) o femenino(1) de la persona.
imagen	varchar	250			Url que apunta a la dirección donde se encuentra almacenada una foto de la persona.

Cuadro 4.22: Tabla de diccionario de datos Persona.

#### Tabla de diccionario de datos Producto\_anuncio

El cuadro 4.23 indica la descripción correspondiente de la entidad Producto\_anuncio.

Nombre de la relación:		Producto_anuncio			
Objetivo:		Tabla que almacenará los anuncios de los productos.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
fk_id_producto	bigint	20	FK	X	Identificador del producto.
fk_id_anuncio	smallint	5	FK	X	Identificador del anuncio.

Cuadro 4.23: Tabla de diccionario de datos Producto\_anuncio.

#### Tabla de diccionario de datos Producto\_atributo

El cuadro 4.24 indica la descripción correspondiente de la entidad Producto\_atributo.

Nombre de la relación:		Producto_atributo			
Objetivo:		Tabla que almacenará la relación entre un producto y sus atributos.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
fk_id_producto	bigint	20	PK, FK	X	Identificador del producto.
fk_id_atributo	integer	10	PK, FK	X	Identificador del atributo.

Cuadro 4.24: Tabla de diccionario de datos Producto\_atributo.

#### Tabla de diccionario de datos Producto\_imagen

El cuadro 4.25 indica la descripción correspondiente de la entidad Producto\_imagen.

Nombre de la relación:		Producto_imagen			
Objetivo:		Tabla que almacenará la relación entre un producto y sus imágenes.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
fk_id_imagen	bigint	20	PK, FK	X	Identificador de la imagen del producto.
fk_id_producto	bigint	20	PK, FK	X	Identificador del producto.

Cuadro 4.25: Tabla de diccionario de datos Producto\_imagen.

#### Tabla de diccionario de datos Producto

El cuadro 4.26 indica la descripción correspondiente de la entidad Producto.

Nombre de la relación:		Producto			
Objetivo:		Tabla que almacenará la información general de un producto.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_producto	bigint	20	PK	X	Identificador del producto.
nombre	varchar	150		X	Nombre del producto.
precio_venta	decimal	10, 2		X	Precio de venta del producto.
costo	decimal	10, 2		X	Costo del producto.
descripcion	text			X	Breve descripción del producto.
fk_id_departamento	bigint	20	FK	X	Identificador del departamento al que pertenece el producto.
fk_id_categoria	bigint	20	FK	X	Identificador de la categoría a la que pertenece el producto.
fk_id_marca	integer	10	FK	X	Identificador de la marca de producto.
fk_id_promocion	smallint	5	FK		Identificador de una posible promoción que pueda tener el producto.
stock	integer	10		X	Número actual de productos disponibles para venta.
no_ventas	integer	10		X	Número de veces que se ha vendido este producto.

Cuadro 4.26: Tabla de diccionario de datos Producto.

#### Tabla de diccionario de datos Promoción

El cuadro 4.27 indica la descripción correspondiente de la entidad Promoción.

Nombre de la relación:		Promoción			
Objetivo:		Tabla que almacenará la información sobre las promociones que se relacionan con los productos y/o compras.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_promocion	smallint	5	PK	X	Identificador de la promoción.
gratificacion	smallint	5			Bonificación que tiene el producto o que se puede obtener al comprarlo.
producto_gratis	smallint	5			Indicada si el producto o la compra del mismo se puede obtener de forma gratuita.
porcentaje	smallint	5			Porcentaje de descuento que puede tener el producto.
fecha_inicio	timestamp			X	Fecha a partir de la cual la promoción es válida.
fecha_fin	timestamp			X	Fecha a partir de la cual la promoción deja de ser válida.

Cuadro 4.27: Tabla de diccionario de datos Promoción.

#### Tabla de diccionario de datos Tienda

El cuadro 4.28 indica la descripción correspondiente de la entidad Tienda.

Nombre de la relación:		Tienda			
Objetivo:		Tabla que almacenará los datos básicos de una tienda departamental.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_tienda	bigint	20	PK	X	Identificador de la tienda.
fk_id_categoria	bigint	20	FK	X	Identificador de la categoría de la tienda.
nombre	varchar	150		X	Nombre de la tienda.
direccion	text				Dirección de la tienda.
posicion_geografi	point			X	Coordinada geográfica en la que se encuentra la tienda.

Cuadro 4.28: Tabla de diccionario de datos Tienda.

#### Tabla de diccionario de datos Tipo\_atributo

El cuadro 4.29 indica la descripción correspondiente de la entidad Tipo\_atributo.

Nombre de la relación:		Tipo_atributo			
Objetivo:		Tabla que almacenará los tipos de atributos que puede tener un producto.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_tipo_atributo	integer	10	PK	X	Identificador del tipo de atributo.
nombre	varchar	100		X	Nombre del tipo de atributo.

Cuadro 4.29: Tabla de diccionario de datos Tipo\_atributo.

#### 4.2.2. Modelo de la base de datos

La figura 4.4 mostrada en la parte inferior, presenta el de la base de datos que esta compuesto por cada una de las entidades definidas anteriormente y que permite visualizar adecuadamente la conexión entre cada una de ellas. Este diagrama se ha dividido en dos secciones a fin de poder obtener una mejor visualización de los datos, las figuras posteriores 4.5 y 4.6 son dichas divisiones de la figura 4.4.

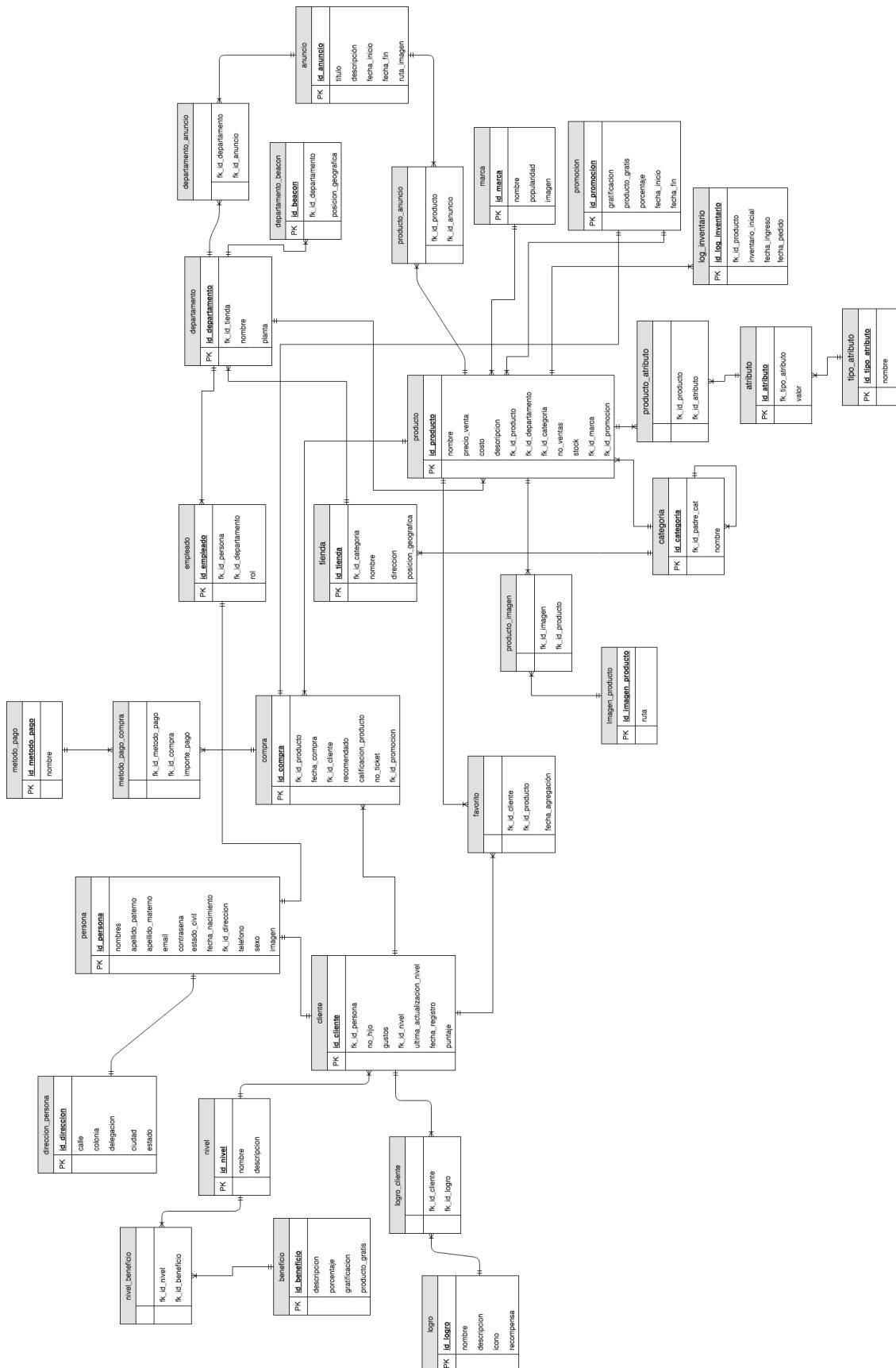


Figura 4.4: Prototipo 1: Diagrama del modelo de la base de datos (Visualización completa).

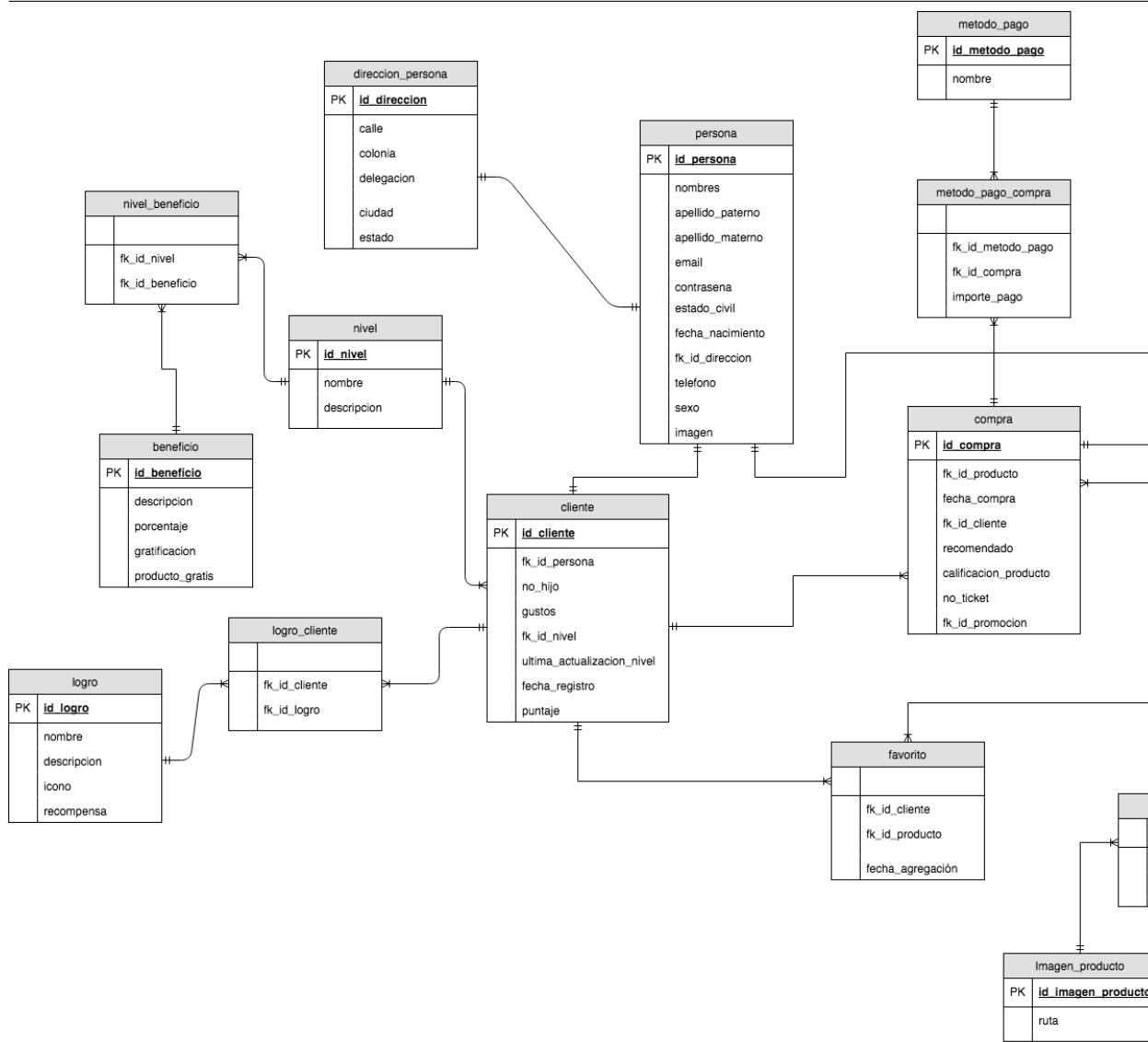


Figura 4.5: Prototipo 1: Diagrama del modelo de la base de datos (Parte 1).

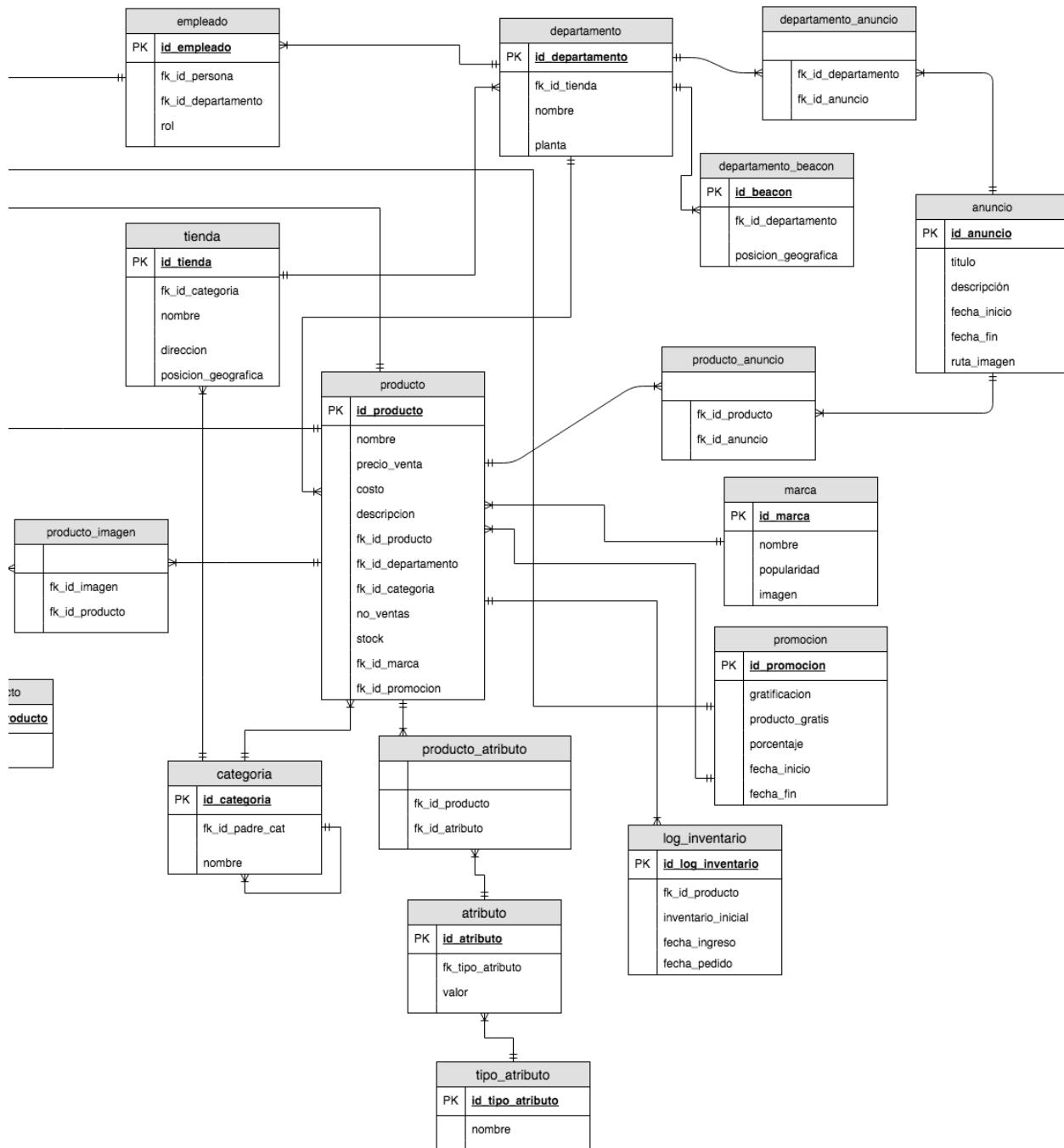


Figura 4.6: Prototipo 1: Diagrama del modelo de la base de datos (Parte 2).

## 4.3. Prototipo 1.1: Modificaciones al modelo de la base de datos

A continuación se muestran enlistados los cambios realizados al modelo de la base de datos anterior, de igual manera se muestra el diccionario de datos para las entidades que fueron modificadas y posteriormente se observa el diagrama con las correcciones correspondientes.

### 4.3.1. Modificaciones realizadas

- Se cambió la entidad **direccion\_persona** a **direccion**.
- Dentro de la entidad **direccion** se realizaron las siguientes modificaciones:
  - Se borró el campo calle.
  - Se cambió de nombre el campo **delegacion** a **delegacion\_municipio**.
  - Se cambió de nombre el campo **estado** a **ciudad\_estado**.
  - Se agregó el campo **codigo\_postal** de tipo *varchar(7)*.
- Se agregó llave foranea **fk\_id\_direccion** a la entidad **tienda** con referencia **direccion(id\_direccion)**.
- Se agregó campo **nombre** a la entidad **promocion** con tipo de dato *varchar(100)*.
- Se modificó la cardinalidad de **tienda-categoría** a muchos a muchos.
- Se eliminó el campo **fk.id\_producto** de la entidad **producto**.
- Se agregó el campo **nombre\_usuario** a la entidad **empleado**

### 4.3.2. Diccionario de datos

#### Tabla de diccionario de datos Empleado

El cuadro 4.30 indica la descripción correspondiente de la entidad Empleado del prototipo 1.1.

Nombre de la relación:		Empleado			
Objetivo:		Tabla que almacenará los datos específicos de un empleado.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_empleado	integer	10	PK	X	Identificador del empleado.
fk_id_persona	bigint	20	FK	X	Identificador de la persona.
fk_id_departamento	bigint	20	FK	X	Identificador del departamento al que pertenece el empleado.
rol	char	1		X	Tipo de rol que tiene el empleado (vendedor/administrador).
nombre_usuario	varchar	50		X	Nombre de usuario con el que se identifica el empleado.

Cuadro 4.30: Tabla de diccionario de datos Empleado del prototipo 1.1 .

### Tabla de diccionario de datos Dirección

El cuadro 4.35 indica la descripción correspondiente de la entidad Dirección del prototipo 1.2.

Nombre de la relación:		Dirección			
Objetivo:		Tabla que almacenará la dirección de una persona o tienda.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_direccion	bigint	20	PK	X	Identificador de la dirección del cliente.
colonia	varchar	100		X	Nombre de las colonias dentro pertenecientes a un código postal.
delegacion_municipio	varchar	50			Nombre de la delegación perteneciente a la Ciudad de México o municipio perteneciente a un estado.
ciudad_estado	varchar	50		X	Nombre de la ciudad(Ciudad de México) o estado perteneciente a la República Mexicana.
codigo_postal	varchar	5		X	Código postal dentro de la República Mexicana.
calle	varchar	100		X	Nombre de la calle perteneciente a la dirección.

Cuadro 4.31: Tabla de diccionario de datos Dirección del prototipo 1.2.

### Tabla de diccionario de datos Promoción

El cuadro 4.32 indica la descripción correspondiente de la entidad Promoción del prototipo 1.1.

Nombre de la relación:		Promoción			
Objetivo:		Tabla que almacenará la información sobre las promociones que se relacionan con los productos y/o compras.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_promocion	integer	10	PK	X	Identificador de la promoción.
nombre	varchar	100		X	Nombre de la promoción.
gratificacion	smallint	5			Bonificación que tiene el producto o que se puede obtener al comprarlo.
producto_gratis	smallint	5			Indicada si el producto o la compra del mismo se puede obtener de forma gratuita.
porcentaje	smallint	5			Porcentaje de descuento que puede tener el producto.
fecha_inicio	timestamp			X	Fecha a partir de la cual la promoción es válida.
fecha_fin	timestamp			X	Fecha a partir de la cual la promoción deja de ser válida.

Cuadro 4.32: Tabla de diccionario de datos Promoción del prototipo 1.1 .

#### Tabla de diccionario de datos Tienda

El cuadro 4.33 indica la descripción correspondiente de la entidad Tienda del prototipo 1.1.

Nombre de la relación:		Tienda			
Objetivo:		Tabla que almacenará los datos básicos de una tienda departamental.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_tienda	bigint	20	PK	X	Identificador de la tienda.
nombre	varchar	150		X	Nombre de la tienda.
fk_id_direccion	bigint	20		X	Identificador de la dirección de la tienda.
posicion_geografi	point			X	Coordenada geográfica en la que se encuentra la tienda.

Cuadro 4.33: Tabla de diccionario de datos Tienda del prototipo 1.1 .

#### Tabla de diccionario de datos Tienda\_categoria

El cuadro 4.34 indica la descripción correspondiente de la entidad Tienda\_categoria del prototipo 1.1.

Nombre de la relación:		Tienda_categoria			
Objetivo:		Tabla que almacenará la relación entre las tiendas y sus categorías.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
fk_id_tienda	bigint	20	PK, FK	X	Identificador de la tienda.
fk_id_categoria	bigint	20	PK, FK	X	Identificador de la categoría.

Cuadro 4.34: Tabla de diccionario de datos Tienda\_categoria del prototipo 1.1.

#### 4.3.3. Modelo de la base de datos

En la figura 4.7, se puede observar las modificaciones realizadas al modelo de la base de datos, marcadas en color verde para diferenciar las entidades agregadas y en letras negritas los campos modificados o agregados. Este diagrama se ha dividido en dos secciones a fin de poder obtener una mejor visualización de los datos, las figuras posteriores 4.8 y 4.9 son dichas divisiones de la figura 4.7.

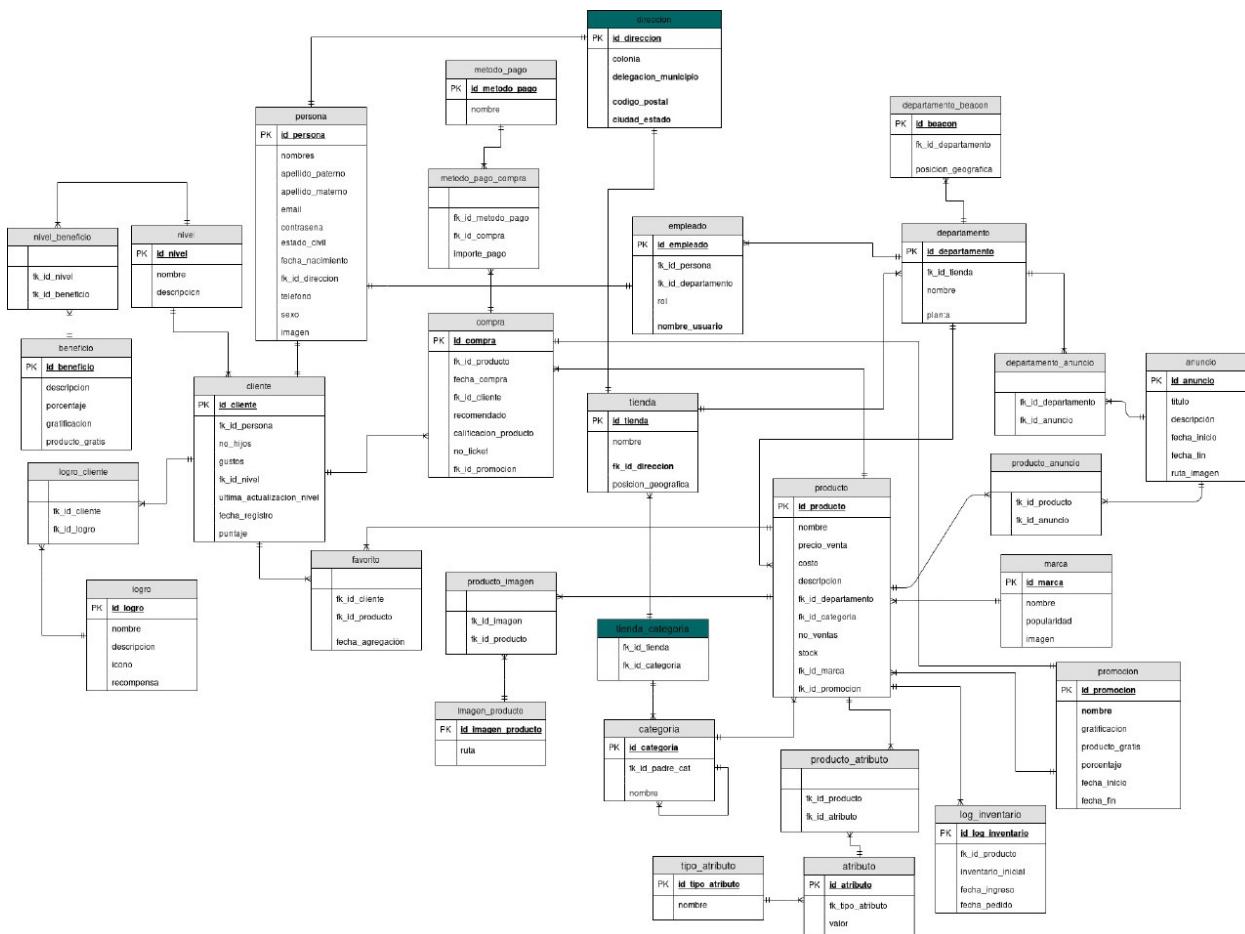


Figura 4.7: Prototipo 1.1: Diagrama del modelo de la base de datos (Visualización completa).

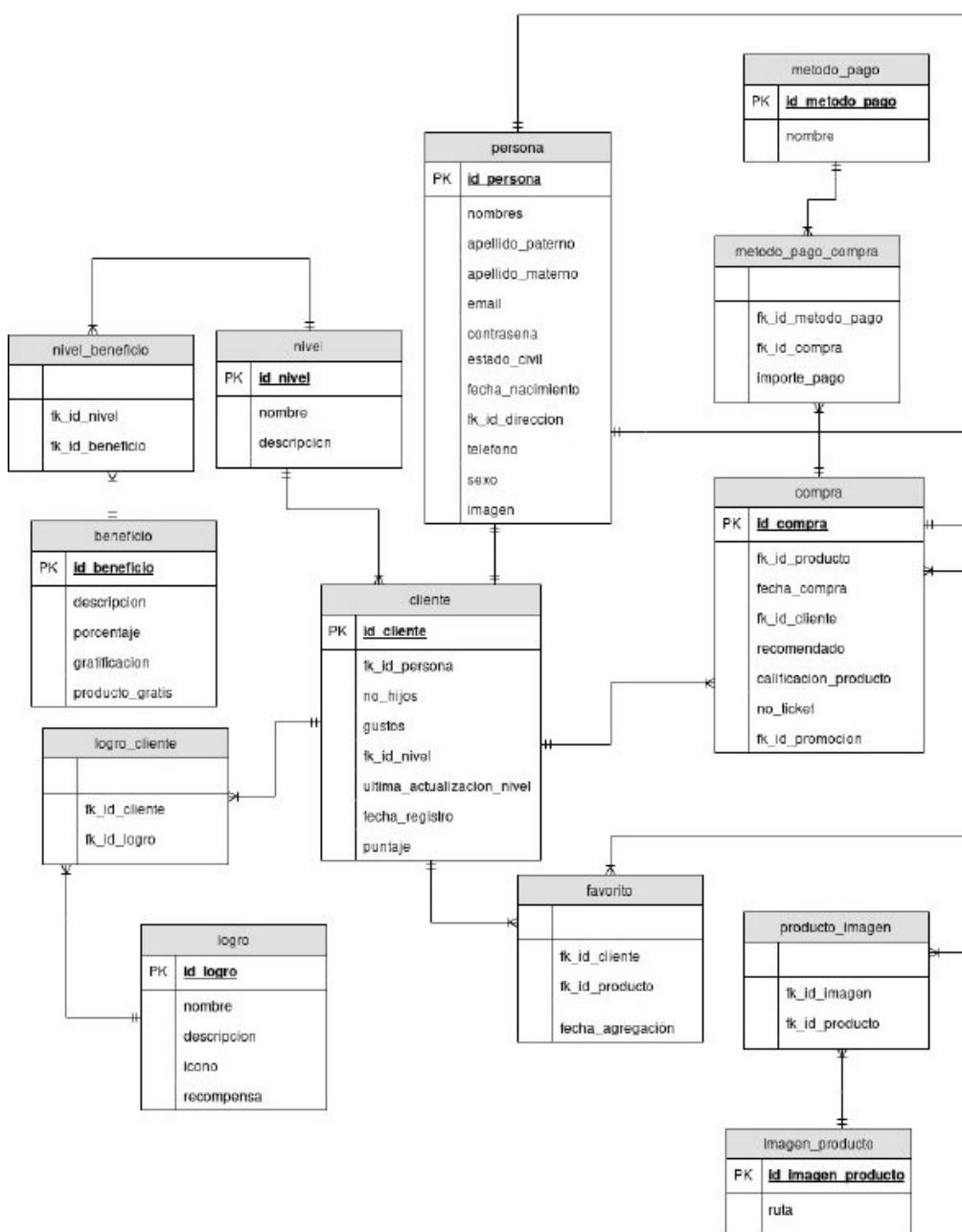


Figura 4.8: Prototipo 1.1: Diagrama del modelo de la base de datos (Parte 1).

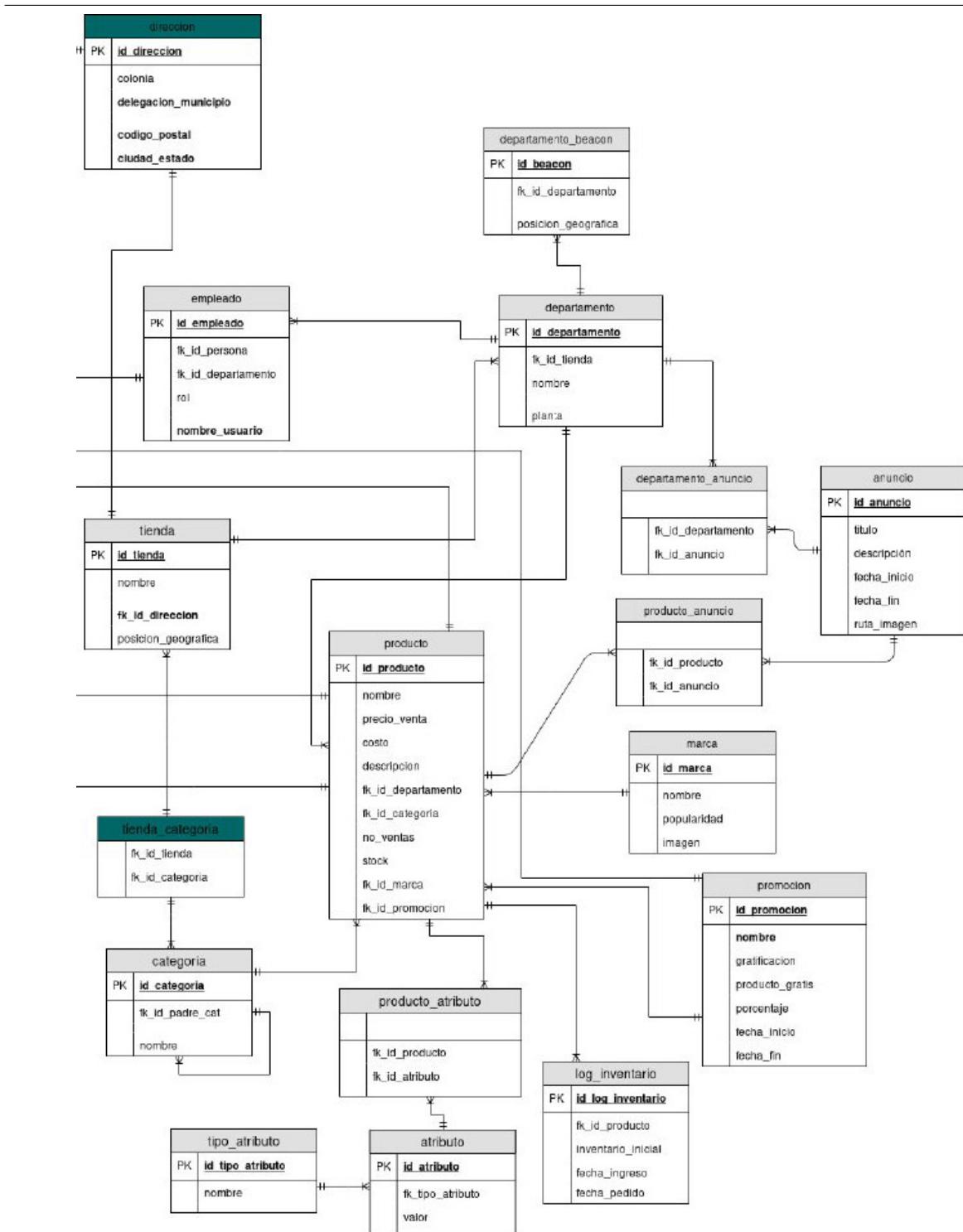


Figura 4.9: Prototipo 1.1: Diagrama del modelo de la base de datos (Parte 2).

## 4.4. Prototipo 1.2: Modificaciones al modelo de la base de datos

En la parte inferior se muestran los cambios realizados al modelo de la base de datos del prototipo 1.1 y de igual manera se muestra el diagrama con dichas modificaciones marcadas sobre él en color rojo las tablas eliminadas y en letras negritas los campos agregados sobre las entidades modificadas.

### 4.4.1. Modificaciones realizadas

- Se eliminó la entidad **producto\_atributo**.
- Se eliminó la entidad **tipo\_atributo**.
- Se eliminó la entidad **atributo**.
- Se agregó el campo **calle** a la entidad **direccion**

### 4.4.2. Diccionario de datos de entidades modificadas

**Tabla de diccionario de datos Dirección**

El cuadro 4.35 indica la descripción correspondiente de la entidad Dirección del prototipo 1.2.

Nombre de la relación:		Dirección			
Objetivo:		Tabla que almacenará la dirección de una persona o tienda.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_direccion	bigint	20	PK	X	Identificador de la dirección del cliente.
colonia	varchar	100		X	Nombre de las colonias dentro pertenecientes a un código postal.
delegacion_municipio	varchar	50			Nombre de la delegación perteneciente a la Ciudad de México o municipio perteneciente a un estado.
ciudad_estado	varchar	50		X	Nombre de la ciudad(Ciudad de México) o estado perteneciente a la República Mexicana.
codigo_postal	varchar	5		X	Código postal dentro de la República Mexicana.
calle	varchar	100		X	Nombre de la calle perteneciente a la dirección.

Cuadro 4.35: Tabla de diccionario de datos Dirección del prototipo 1.2.

### 4.4.3. Modelo de la base de datos

En la figura 4.10, se puede observar las modificaciones realizadas al modelo de la base de datos, marcadas nuevamente en color verde para diferenciarlas. Este diagrama se ha dividido en 2 secciones a fin de poder obtener una mejor visualización de los datos, las figuras posteriores 4.11 y 4.12 son dichas divisiones de la figura 4.10.

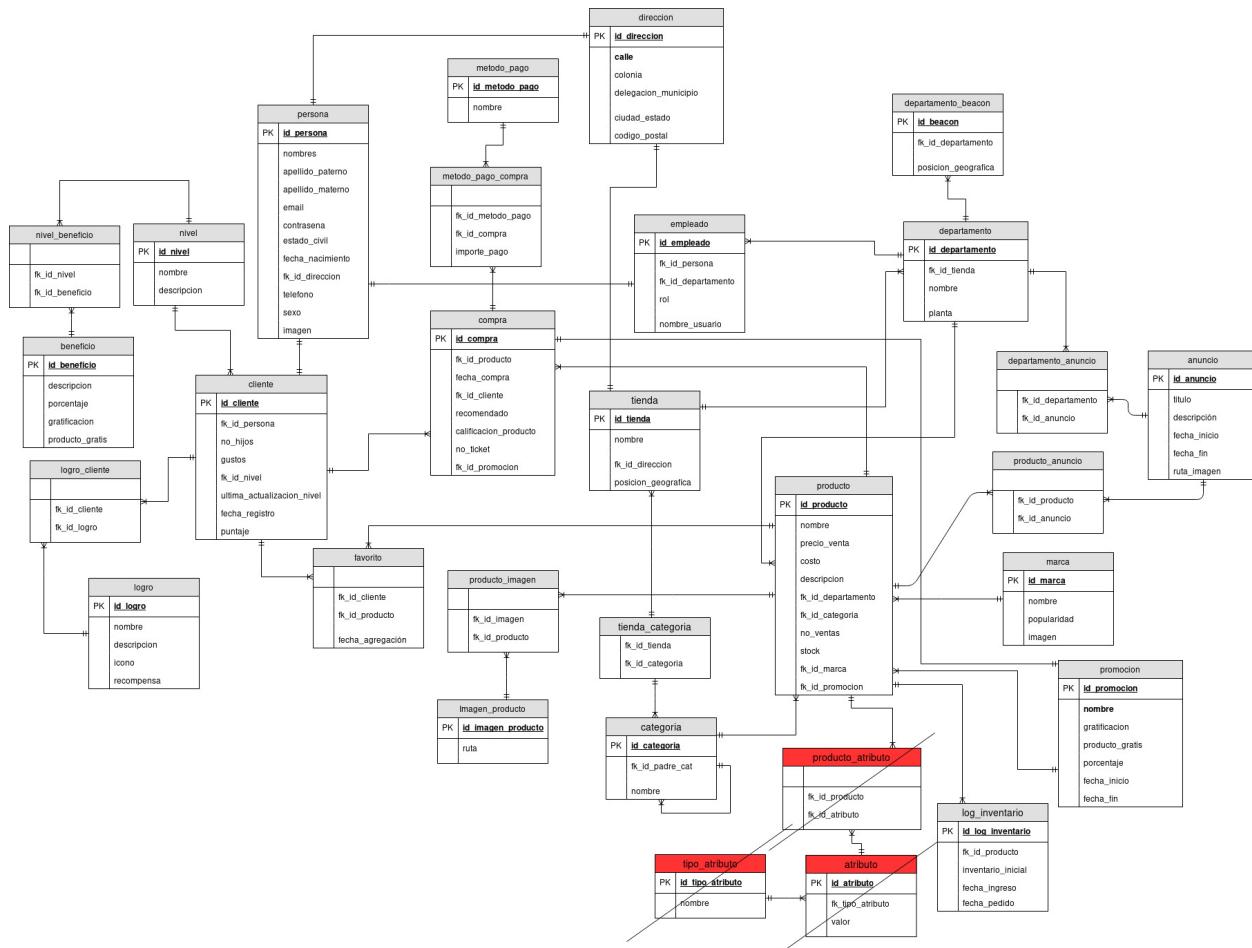


Figura 4.10: Prototipo 1.2: Diagrama del modelo de la base de datos (Visualización completa).

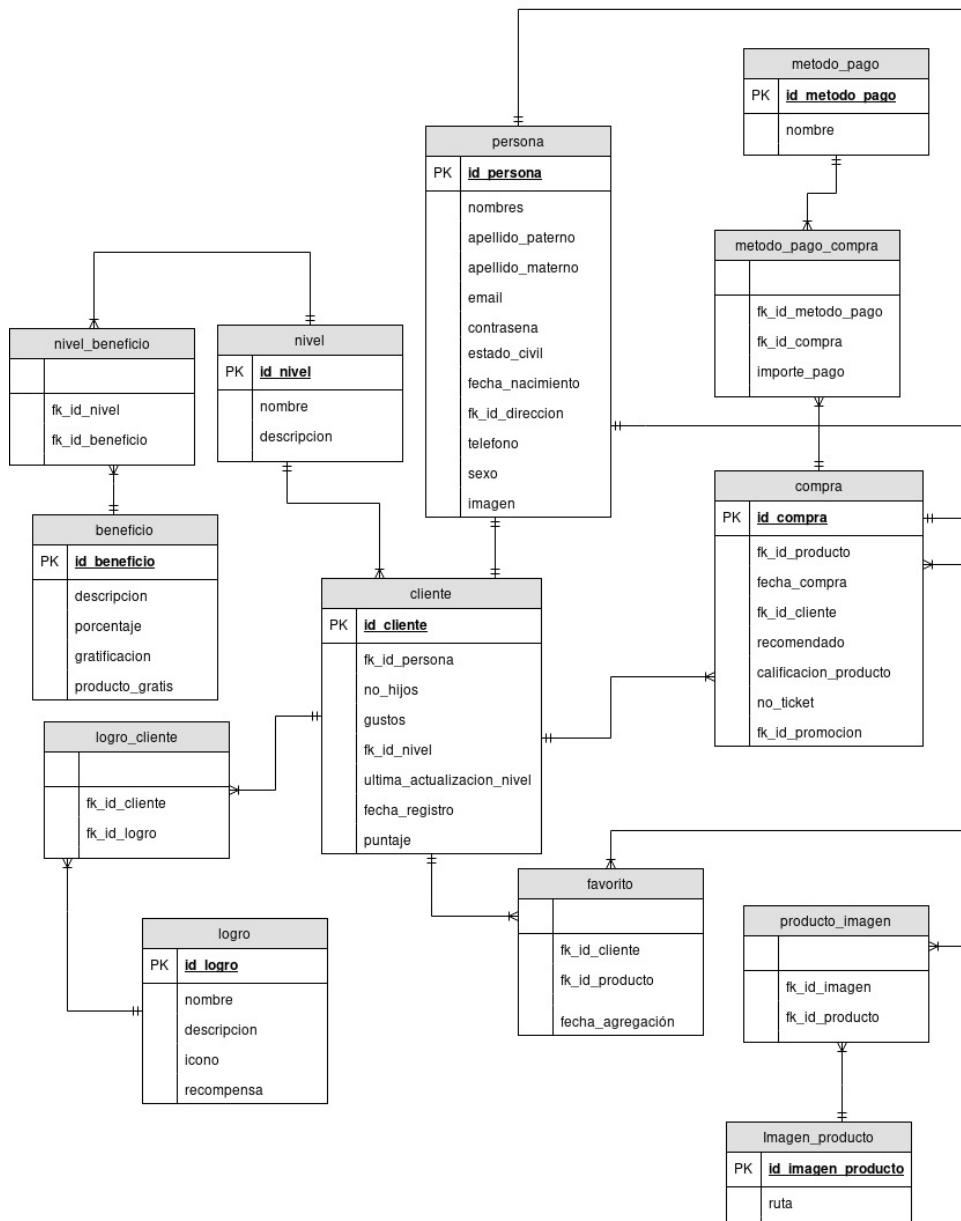


Figura 4.11: Prototipo 1.2: Diagrama del modelo de la base de datos (Parte 1).

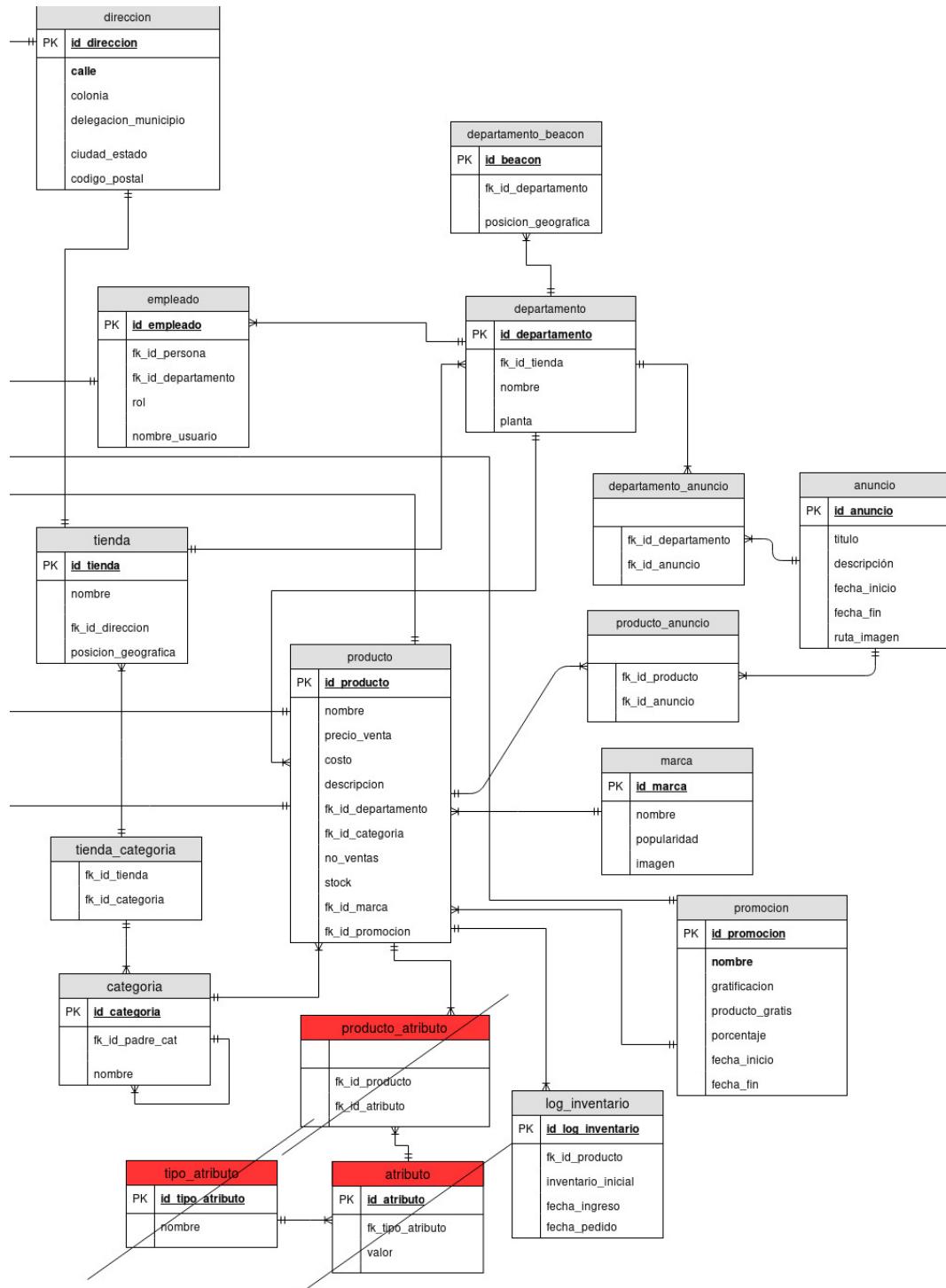


Figura 4.12: Prototipo 1.2: Diagrama del modelo de la base de datos (Parte 2).

## 4.5. Prototipo 1.3: Modificaciones al modelo de la base de datos

En la parte inferior se muestran los cambios realizados al modelo de la base de datos del prototipo 1.2 y de igual manera se muestra el diagrama con dichas modificaciones marcadas sobre él en color rojo las tablas eliminadas y verde con letras negritas los campos agregados sobre las entidades modificadas.

### 4.5.1. Modificaciones realizadas

- Se agregó el campo **fecha\_sincronizacion** a la entidad **departamento\_beacon**

### 4.5.2. Diccionario de datos de entidades modificadas

**Tabla de diccionario de datos Departamento\_beacon**

El cuadro 4.36 indica los cambios hechos en la descripción correspondiente de la entidad Departamento\_beacon.

Nombre de la relación:		Departamento.beacon			
Objetivo:		Tabla que almacenará los beacons asociados a un departamento.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
fk_id_departame	bigint	20	PK	X	Identificador del departamento.
id_beacon	varchar	50		X	Identificador único del Beacon.
posicion_geografi	point			X	Coordenada geográfica en la que se encuentra el Beacon físicamente.
fecha_sincronizac	timestamp			X	Última fecha en la que se hizo la sincronización de los datos del Beacon.

Cuadro 4.36: Tabla de diccionario de datos Departamento.beacon.

### 4.5.3. Modelo de la base de datos

En la figura 4.13, se puede observar las modificaciones realizadas al modelo de la base de datos, marcadas nuevamente en color verde para diferenciarlas. Este diagrama se ha dividido en 2 secciones a fin de poder obtener una mejor visualización de los datos, las figuras posteriores 4.14 y 4.15 son dichas divisiones de la figura 4.13.

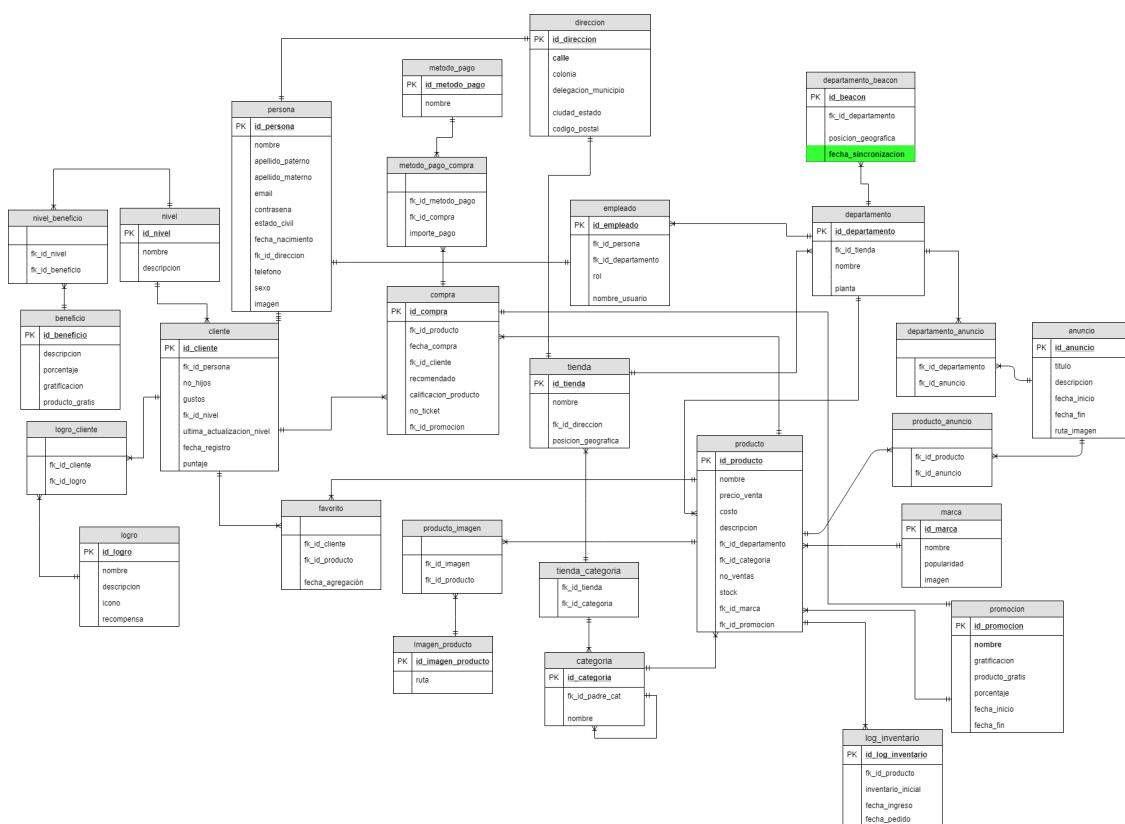


Figura 4.13: Prototipo 1.3: Diagrama del modelo de la base de datos (Visualización completa).

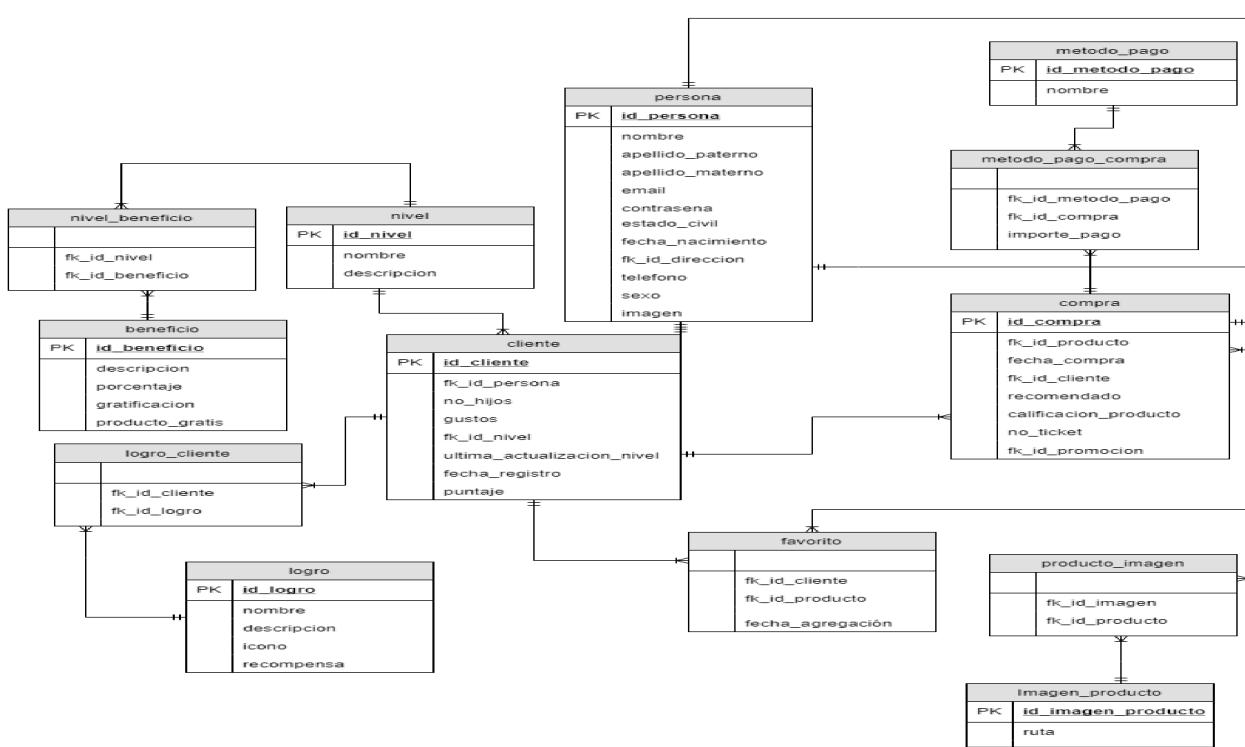


Figura 4.14: Prototipo 1.3: Diagrama del modelo de la base de datos (Parte 1).

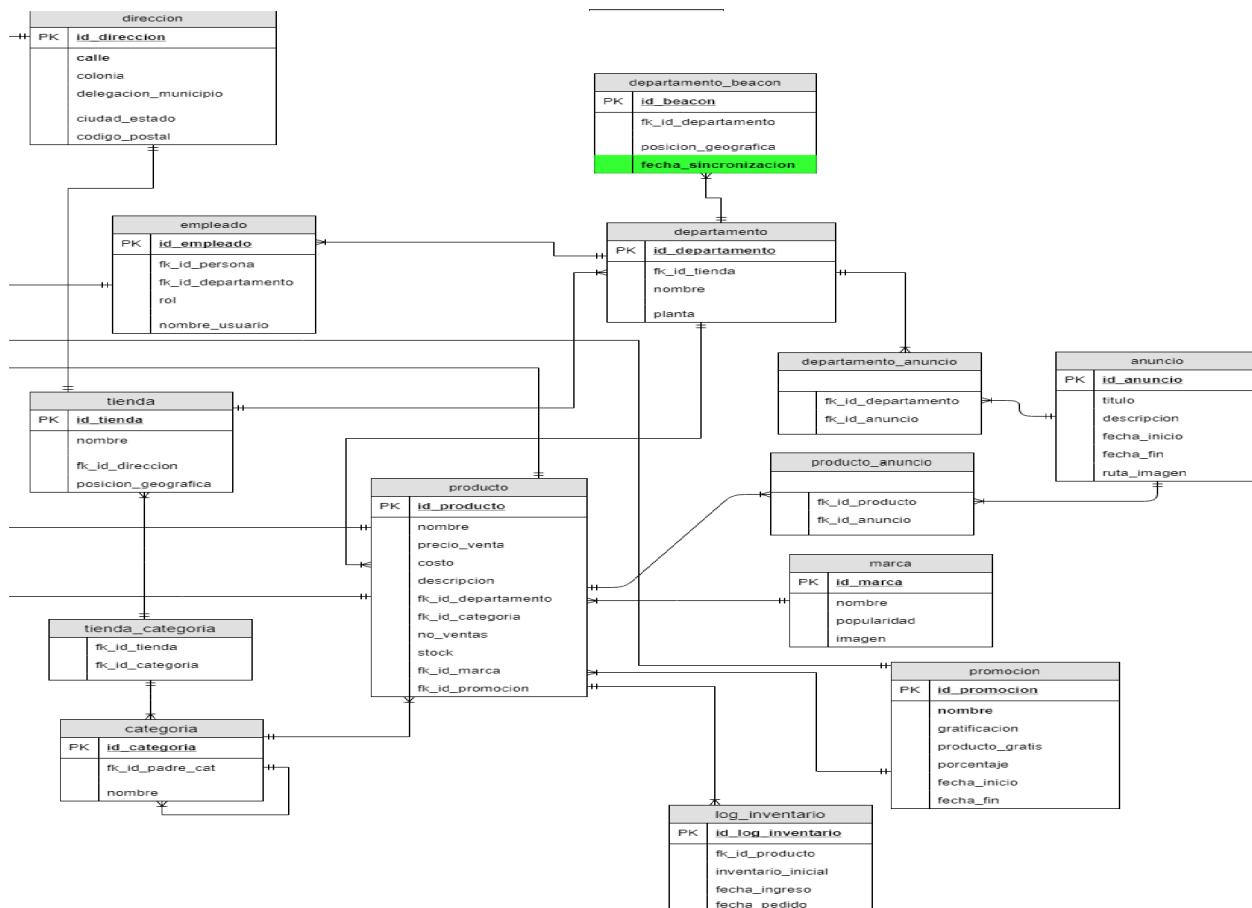


Figura 4.15: Prototipo 1.3: Diagrama del modelo de la base de datos (Parte 2).

## 4.6. Prototipo 1.4: Modificaciones al modelo de la base de datos

En la parte inferior se muestran los cambios realizados al modelo de la base de datos del prototipo 1.3 y de igual manera se muestra el diagrama con dichas modificaciones marcadas en fondo verde y letras negritas con los campos agregados sobre las entidades modificadas.

### 4.6.1. Modificaciones realizadas

- Se agregó el campo **token\_fcm** a la entidad **persona**.
- Se agregó el campo **grupo** a la entidad **cliente**.
- Se agregó el campo **posicion** a la entidad **cliente**.
- Se agregó el campo **permisos** a la entidad **cliente**.
- Se agregó el campo **thetas** a la entidad **cliente**.
- Se agregó el campo **x** a la entidad **producto**.
- Se agregó el campo **mu** a la entidad **producto**.

### 4.6.2. Diccionario de datos de entidades modificadas

#### Tabla de diccionario de datos Persona

El cuadro 4.37 indica la descripción correspondiente de la entidad Persona.

Nombre de la relación:		Persona			
Objetivo:		Tabla que almacenará los datos generales de una persona.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_persona	bigint	20	PK	X	Identificador de la persona.
nombre	varchar	100		X	Nombre(s) de la persona.
apellido_paterno	varchar	50		X	Apellido paterno de la persona.
apellido_materno	varchar	50		X	Apellido materno de la persona.
email	varchar	80		X	Email de la persona, se conforma de la siguiente expresión regular: ^[_a-zA-Z0-9-]+(.[_a-zA-Z0-9-]+)*@[a-zA-Z0-9-]+\\$. [a-zA-Z0-9-] +)*(. [a-zA-Z]{2,4})\$ .
contrasena	varchar	50		X	Contraseña de la persona.
estado_civil	char	1			Estado civil actual de la persona.
fecha_nacimiento	datetime				Fecha de nacimiento de la persona.
fk_id_direccion	integer	10	FK	X	Identificador de la dirección de la persona.
telefono	varchar	20			Teléfono de contacto principal de la persona.
sexo	tinyint	1		X	Sexo masculino(0) o femenino(1) de la persona.
imagen	varchar	250			Url que apunta a la dirección donde se encuentra almacenada una foto de la persona.
token_fcm	varchar	200			Token de registro generado por Firebase Cloud Messaging para la instancia de las aplicaciones cliente.

Cuadro 4.37: Tabla de diccionario de datos Persona.

---

### Tabla de diccionario de datos Cliente

El cuadro 4.38 indica la descripción correspondiente a las modificaciones de la entidad Cliente.

Nombre de la relación:		Cliente			
Objetivo:		Tabla que almacenará los datos específicos de un cliente.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_cliente	bigint	20	PK	X	Identificador del cliente.
fk_id_persona	bigint	20	FK	X	Identificador de la persona.
fk_id_nivel	smallint	5	FK	X	Identificador del nivel del cliente.
no_hijo	tinyInt	2			Número de hijos que tiene el cliente.
gustos	varchar[]	100		X	Gustos de la persona almacenados en un arreglo de longitud 100 cada elemento.
ultima_actualizac	timestamp			X	Fecha de la última actualización del nivel del cliente.
fecha_registro	timestamp			X	Fecha en la que se registro el cliente en Sapphire.
puntaje	smallint	2		X	Puntaje que el cliente tiene actualmente, este puntaje se obtiene por las compras o acciones que realice.
grupo	smallint	2			Grupo al que pertenece el cliente.
posicion	point				Coordenada en que se encuentra el cliente dentro de la gráfica de clusters.
permisos	json			X	Lista de permisos que puede otorgar el cliente al sistema para acceder a su información: edad, estado civil, número de hijos, histórico de compras, productos favoritos y recibir recompensas presenciales. Valor default : '{"edad": false,"estado_civil": false,"hijos": false,"compras":false,"favoritos": false,"recom_presenciales": false}'.
thetas	double[]				Vector de pesos de un usuario el cuál permite hacer la predicción de una calificación sobre un producto.

Cuadro 4.38: Tabla de diccionario de datos Cliente.

### Tabla de diccionario de datos Producto

El cuadro 4.39 indica la descripción correspondiente de la entidad Producto.

Nombre de la relación:		Producto			
Objetivo:		Tabla que almacenará la información general de un producto.			
Campo	Tipo de dato	Tamaño	Constraint	No nulo	Descripción
id_producto	bigint	20	PK	X	Identificador del producto.
nombre	varchar	150		X	Nombre del producto.
precio_venta	decimal	10, 2		X	Precio de venta del producto.
costo	decimal	10, 2		X	Costo del producto.
descripcion	text			X	Breve descripción del producto.
fk_id_departamento	bigint	20	FK	X	Identificador del departamento al que pertenece el producto.
fk_id_categoria	bigint	20	FK	X	Identificador de la categoría a la que pertenece el producto.
fk_id_marca	integer	10	FK	X	Identificador de la marca de producto.
fk_id_promocion	smallint	5	FK		Identificador de una posible promoción que pueda tener el producto.
stock	integer	10		X	Número actual de productos disponibles para venta.
no_ventas	integer	10		X	Número de veces que se ha vendido este producto.
x	double[]				Vector de pesos de un producto el cuál representa las características del mismo.
mu	double				Calificación media de un producto.

Cuadro 4.39: Tabla de diccionario de datos Producto.

#### 4.6.3. Modelo de la base de datos

En la figura 4.16, se puede observar las modificaciones realizadas al modelo de la base de datos, marcadas nuevamente en color verde para diferenciarlas. Para una mejor visualización del modelo, se dividió en las figuras 4.17 y 4.18 mostradas a continuación.

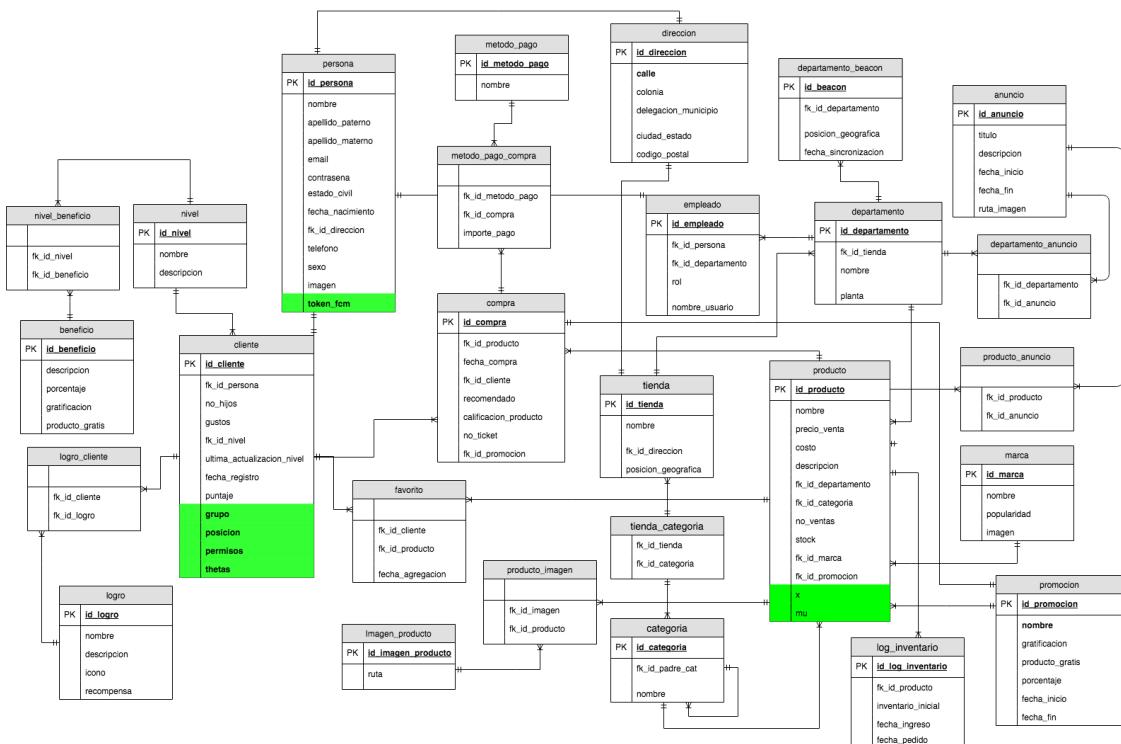


Figura 4.16: Prototipo 1.4: Diagrama del modelo de la base de datos (Visualización completa).

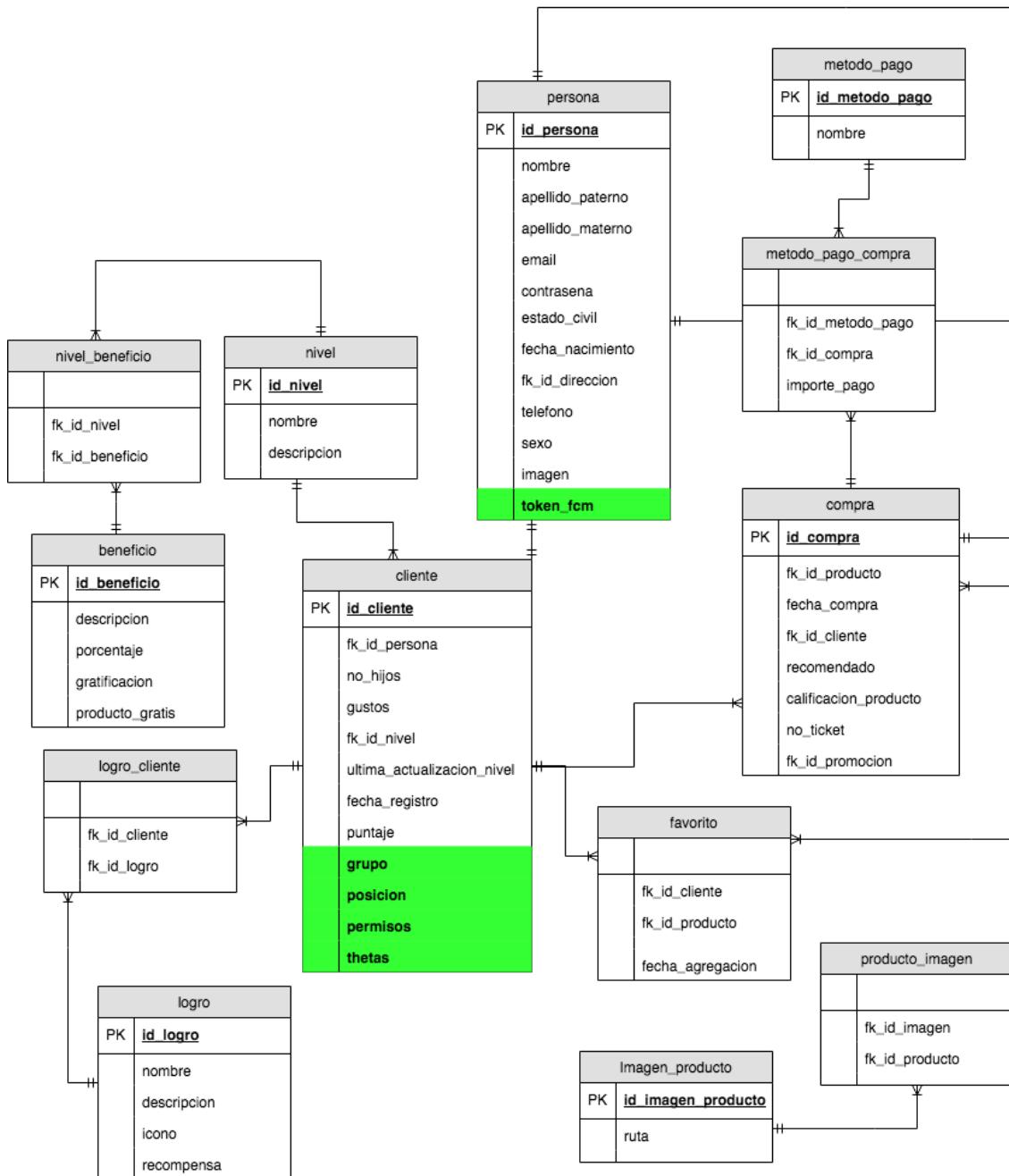


Figura 4.17: Prototipo 1.4: Diagrama del modelo de la base de datos (Parte 1).

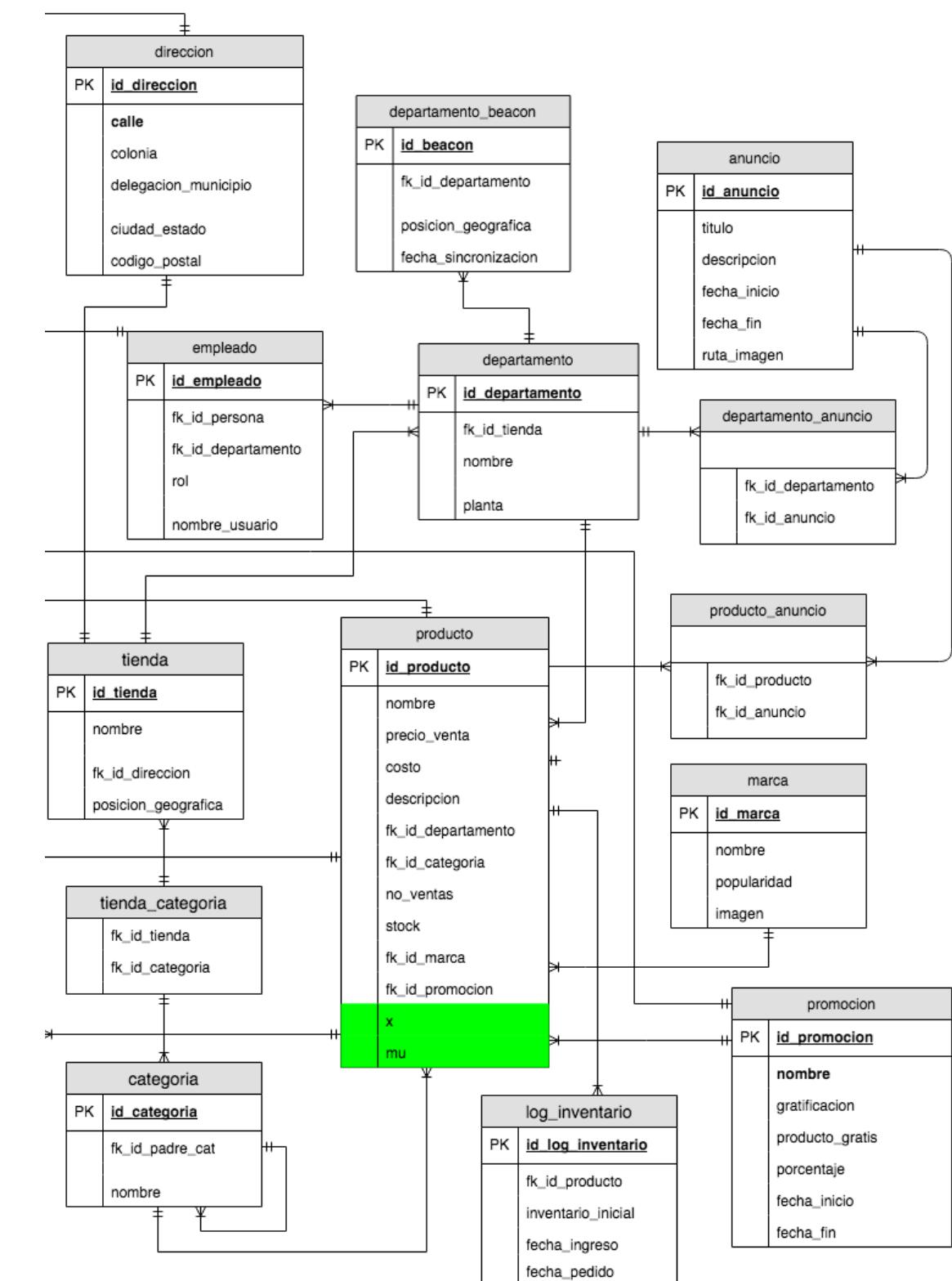


Figura 4.18: Prototipo 1.4: Diagrama del modelo de la base de datos (Parte 2).

## 4.7. Prototipo 2: Generador de Registros Artificiales (GRA)

En la sección actual se encuentra el análisis, diseño y pruebas del prototipo del módulo Generador de Registros Artificiales; este se encarga de insertar registros en múltiples tablas de la base de datos del sistema.

### 4.7.1. Análisis

#### Diagrama de casos de uso

En la figura 4.19 se muestra el diagrama de casos de uso del módulo Generador de Registros Artificiales.

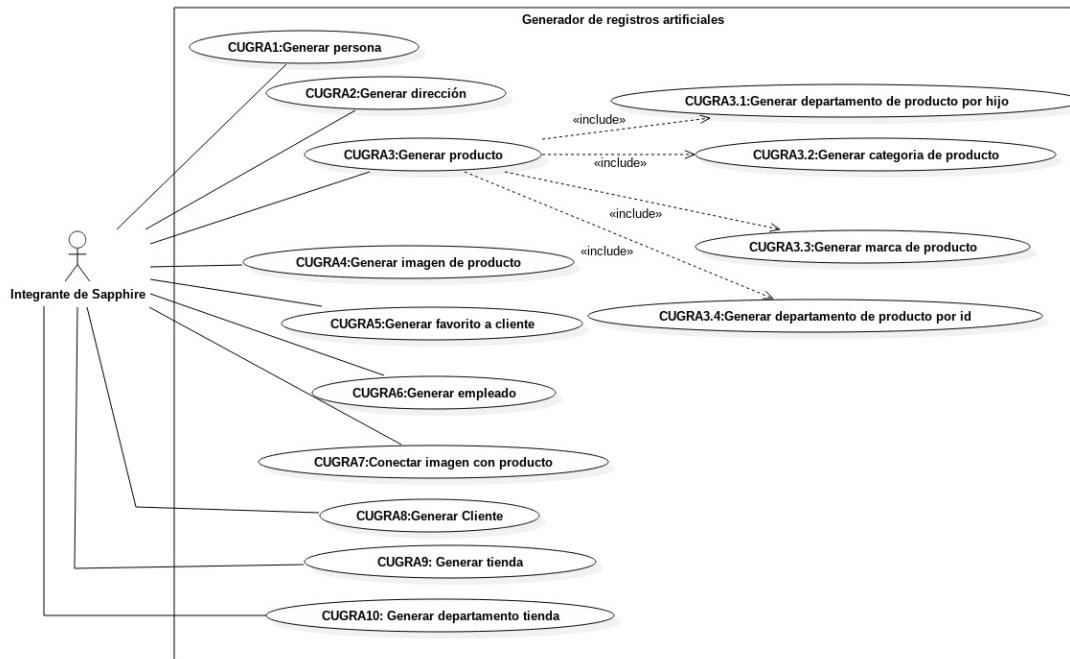


Figura 4.19: Diagrama de casos de uso módulo Generador de Registros Artificiales.

#### Diagrama de clases

La descripción de los elementos en el diagrama de clases (figura 4.20) es la siguiente:

- **Generator:** Clase encargada de la interacción con el usuario, muestra el menú y lee la opción ingresada por el usuario.
- **FakerGenerator:** Clase encargada de tener la lógica para generar los registros artificiales.
- **MysqlConnection:** Clase encargada de manejar la comunicación entre el módulo GRA y el gestor de base de datos MySQL.
- **PsqlConnection:** Clase encargada de manejar la comunicación entre el módulo GRA y el gestor de base de datos PostgreSQL.

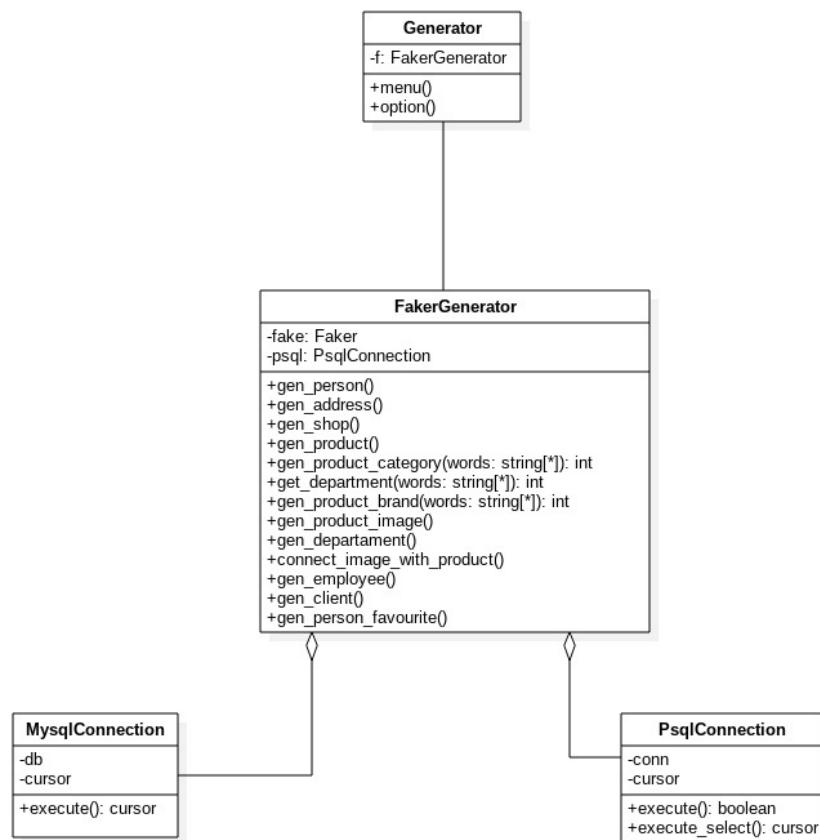


Figura 4.20: Diagrama de clases de módulo Generador de Registros Artificiales.

#### 4.7.2. Diseño

##### Diagramas de secuencia

Durante este apartado se realizará una pequeña explicación de cada método dentro de la clase FakerGenerator, además, dentro del mismo se muestran los diagramas de secuencia de dichos métodos separados dentro de sus respectivos apartados.

##### Generar persona

Método encargado de cubrir el requerimiento funcional **RFGRA2** que permite realizar el registro de personas ficticias en el repositorio de datos. En la figura 4.21 se muestra su diagrama de secuencia.

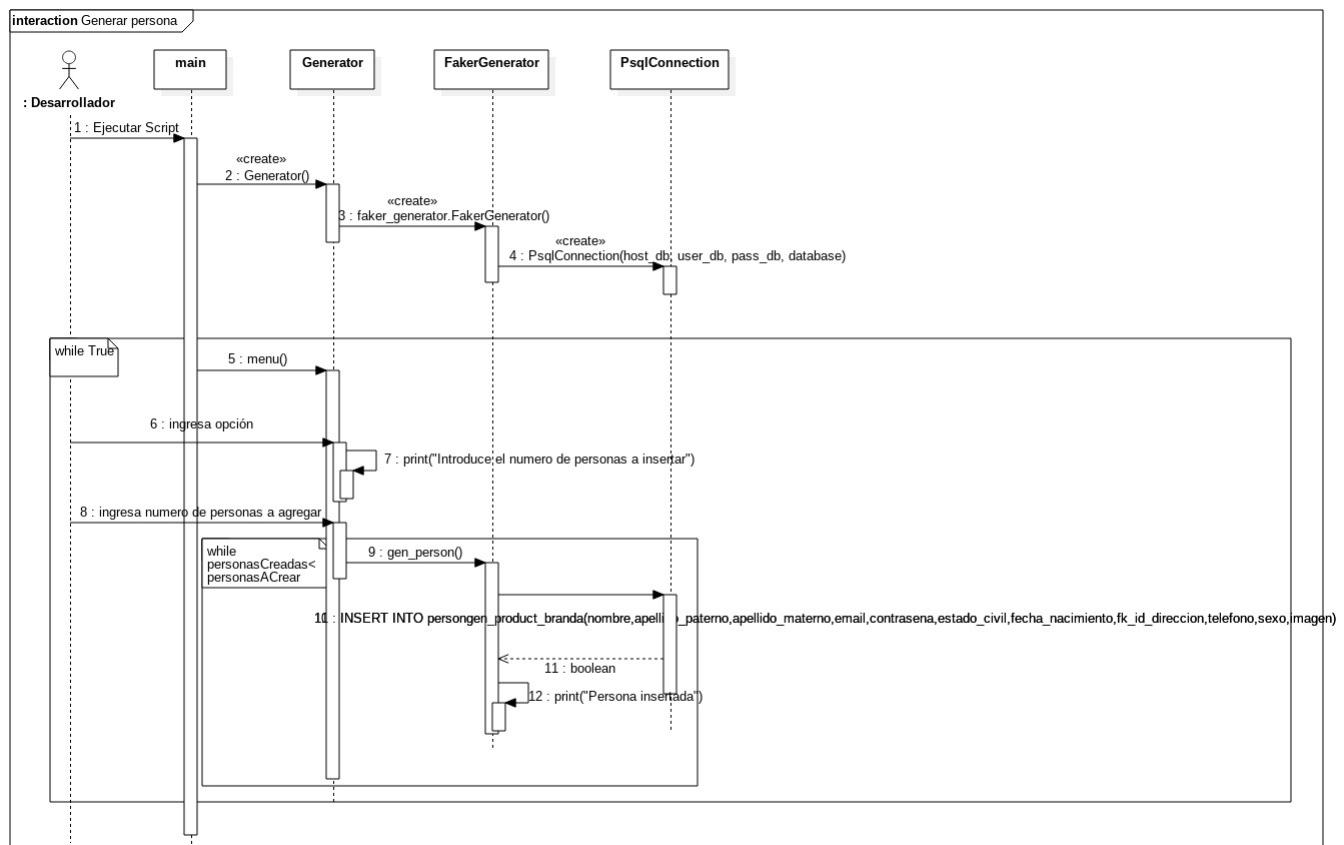


Figura 4.21: Diagrama de secuencia para generar persona.

##### Generar dirección

En el diagrama de secuencia que se muestra en la figura 4.22 se muestra la interacción del método encargado de generar direcciones ficticias que satisfacen el requerimiento funcional **RFGRA3**.

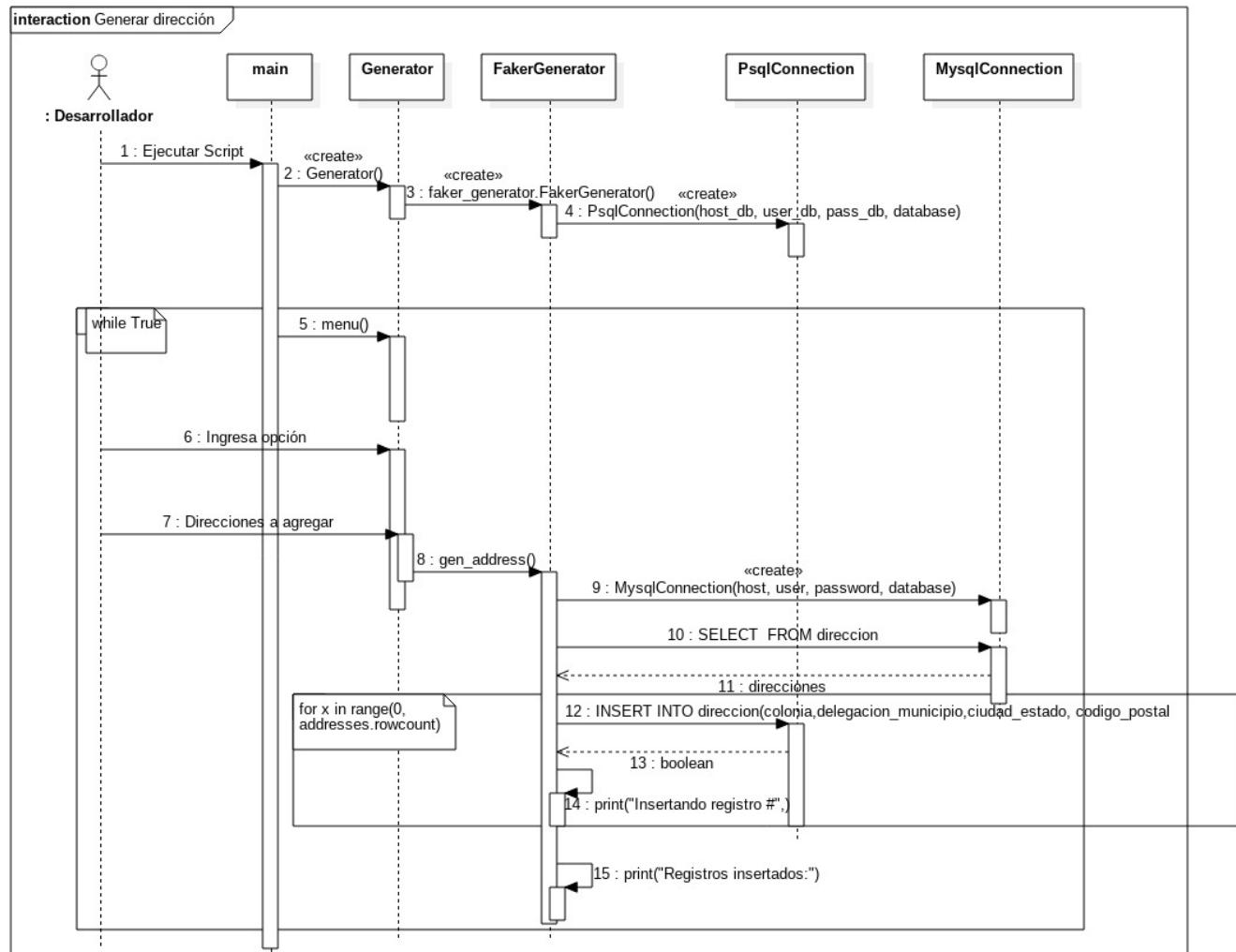


Figura 4.22: Diagrama de secuencia para generar dirección.

### Generar tienda

Método encargado de cubrir el requerimiento funcional **RFGRA4** que permite generar establecimientos comerciales ficticios y posteriormente registrarlos en el repositorio de datos. La figura 4.23 se muestra su diagrama de secuencia.

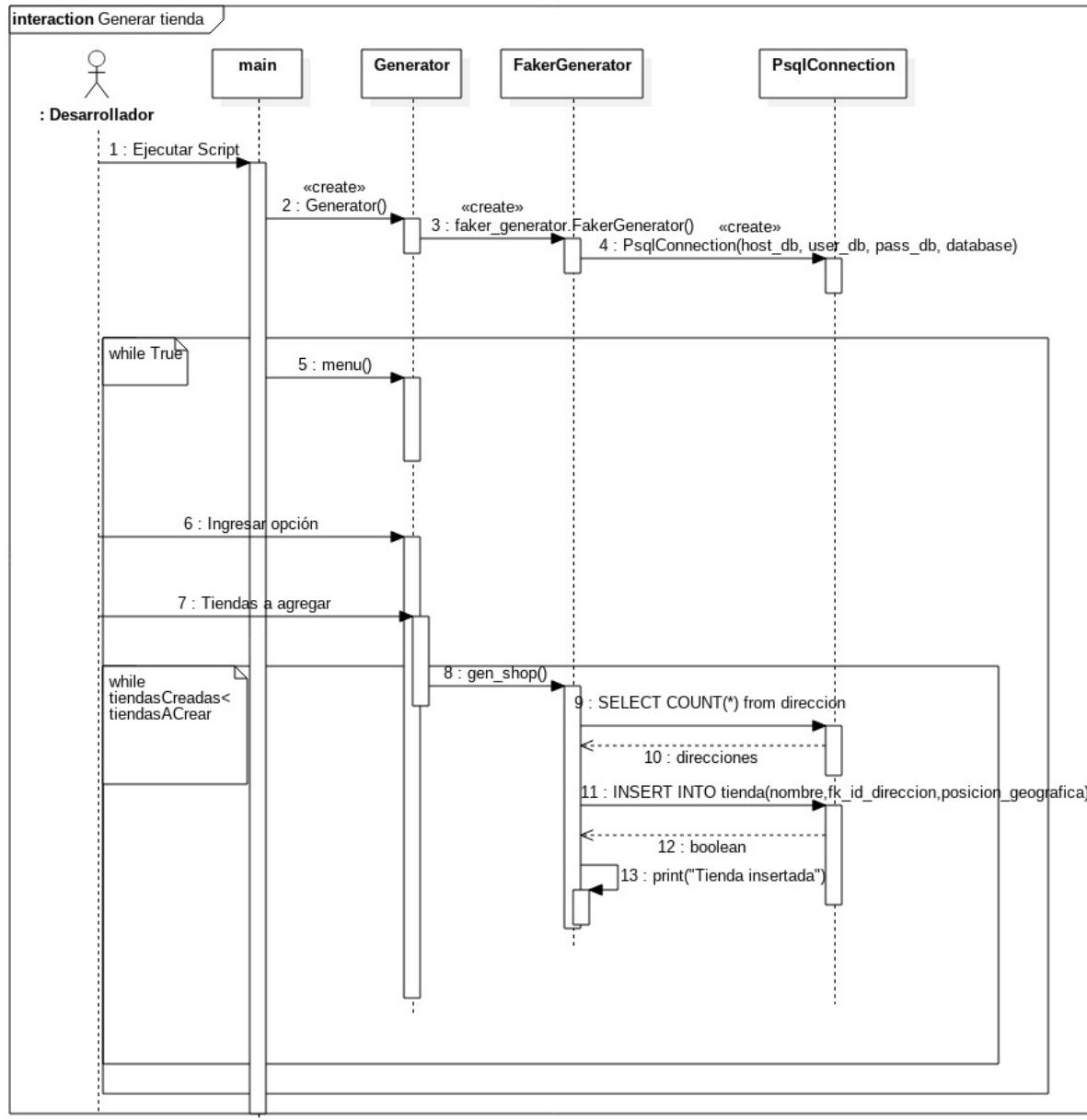


Figura 4.23: Diagrama de secuencia para generar tienda.

### Generar departamento

En el diagrama de secuencia que se muestra en la figura 4.24 se muestra la interacción del método encargado de generar direcciones ficticias que satisfacen el requerimiento funcional **RFGRA5**.

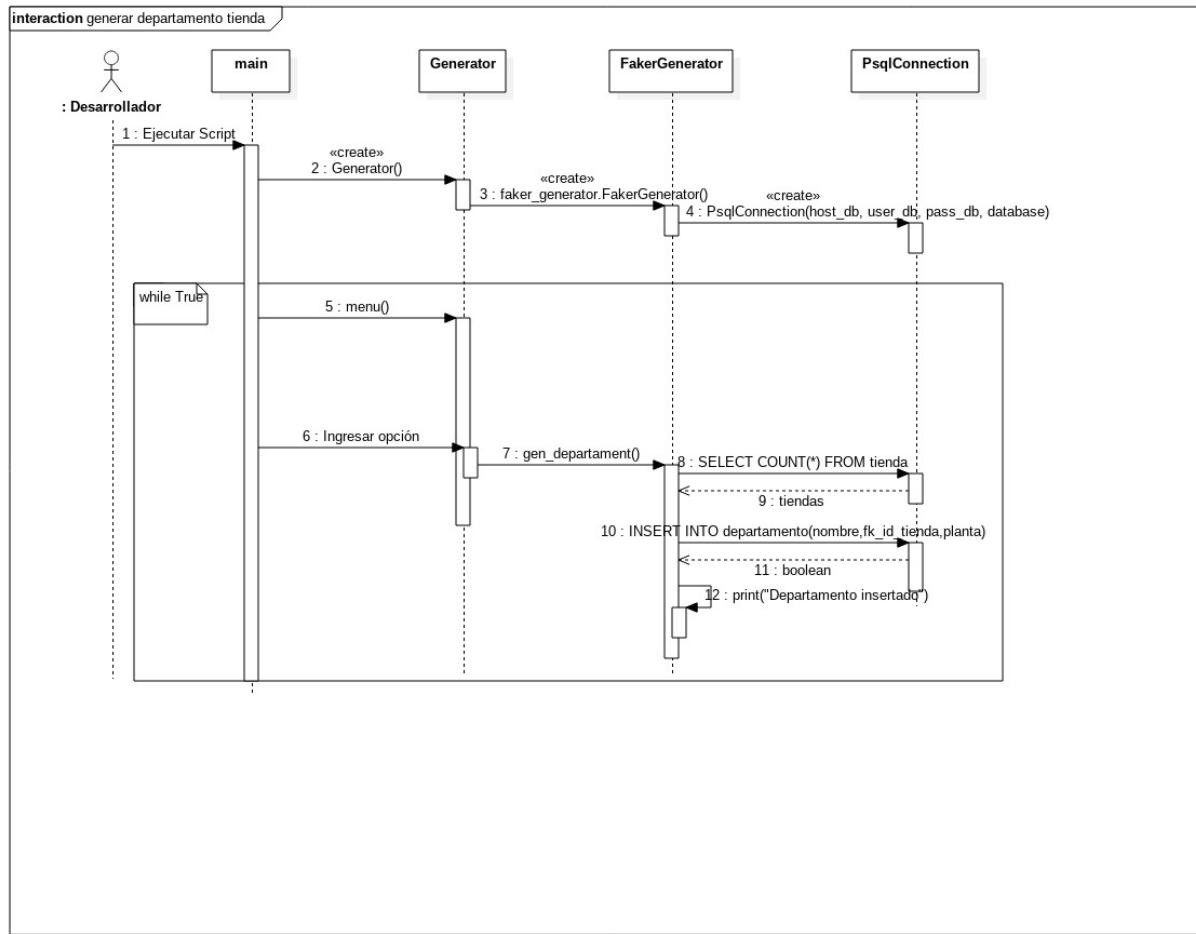


Figura 4.24: Diagrama de secuencia para generar departamento.

### Generar producto

En el siguiente diagrama (Figura 4.25) se muestra la interacción del método creado con el objetivo de insertar productos en el repositorio de datos que satisface el requerimiento funcional **RFGRA6**. Cabe mencionar que este método hace uso de los casos de uso **CUGRA3.1,CUGRA3.2,CUGRA3.3,CUGRA3.4** y por lo tanto se encuentran inmersos en el diagrama de secuencia.

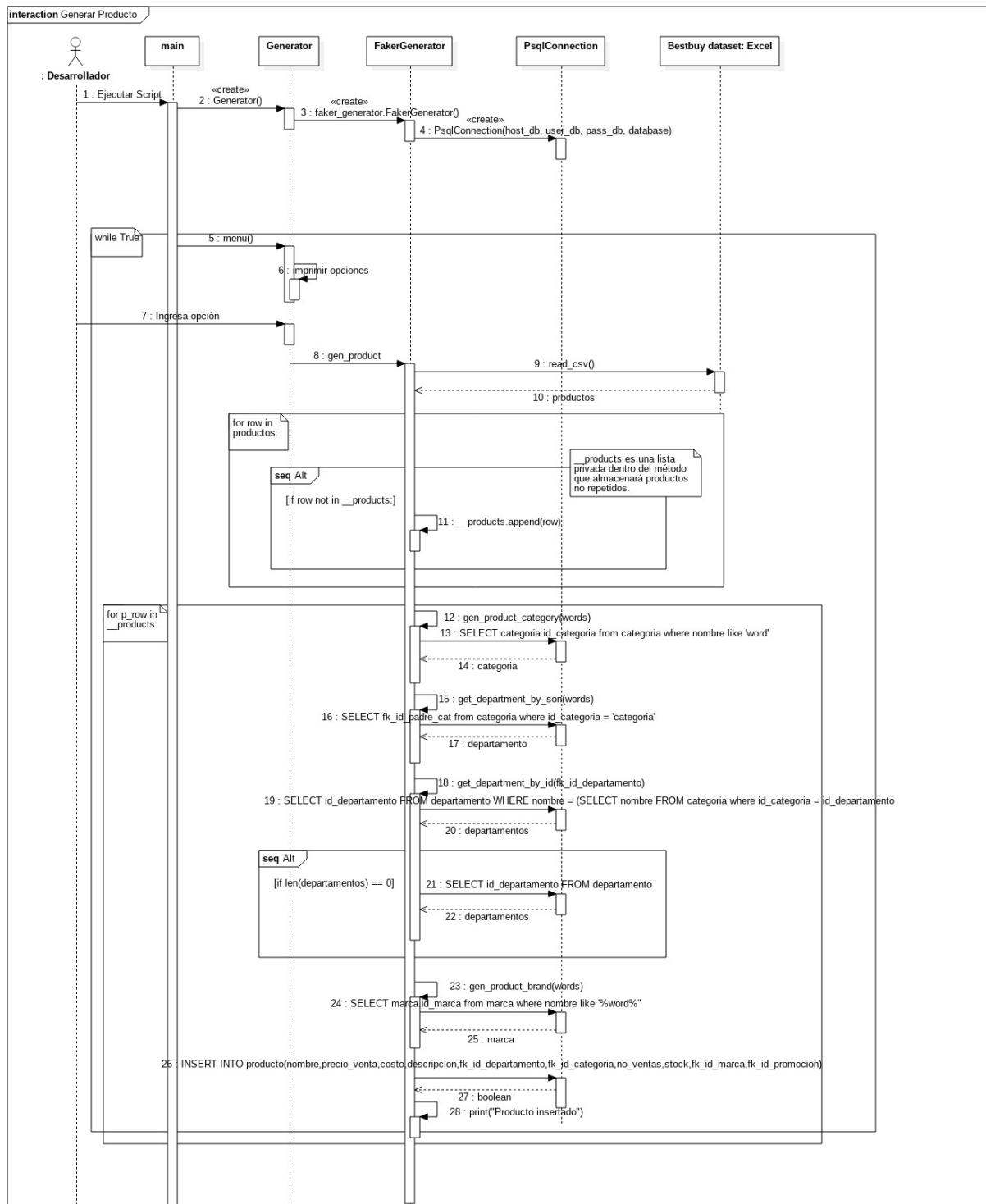


Figura 4.25: Diagrama de secuencia para generar producto.

### Generar imágenes pertenecientes a productos

En el diagrama de secuencia que se muestra en la figura 4.26 se muestra la interacción del método encargado de insertar imágenes en el repositorio de datos que satisfacen el requerimiento funcional RFGRA7.

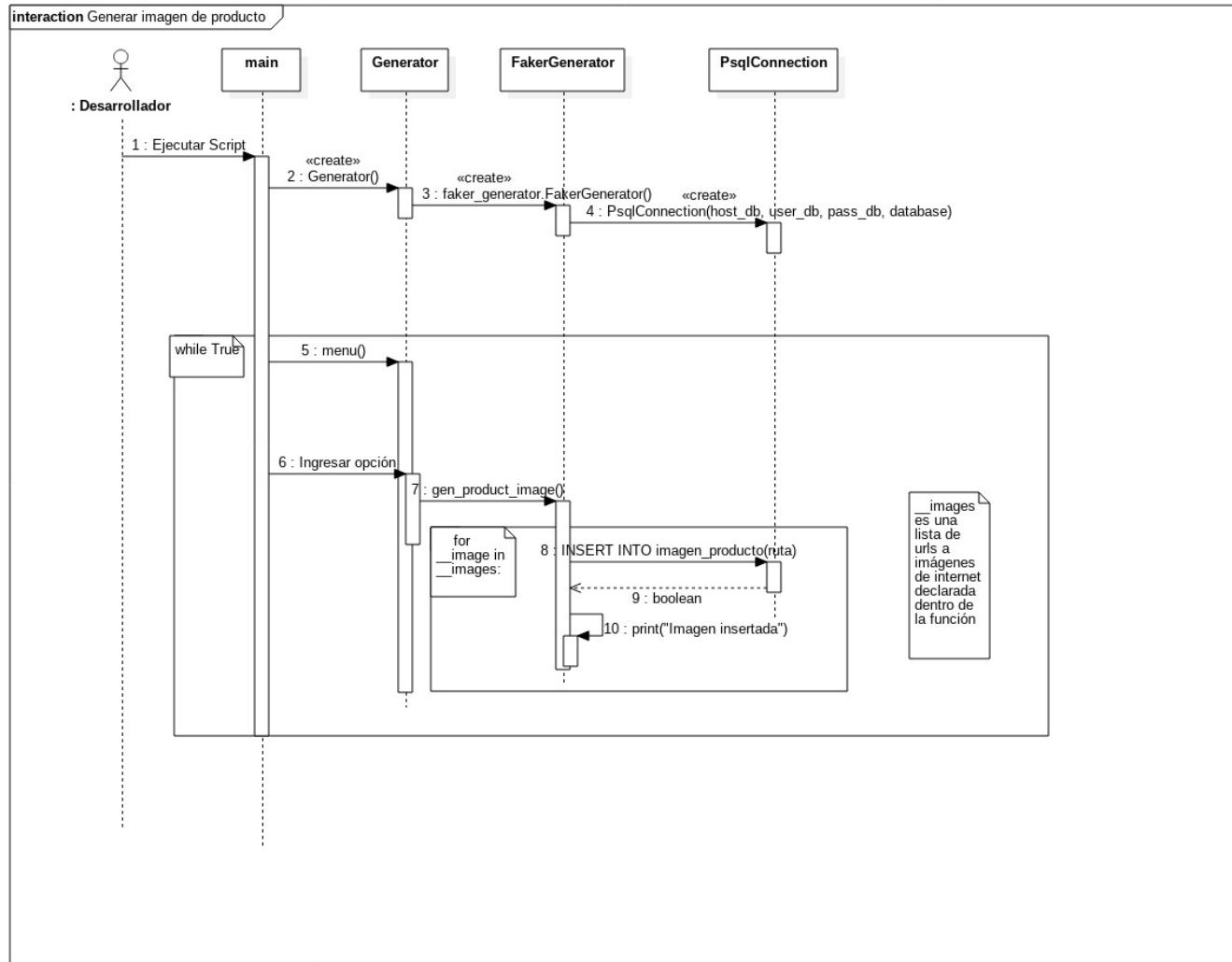


Figura 4.26: Diagrama de secuencia para generar imágenes pertenecientes a productos.

### Asignar imágenes a productos

Este método se encarga de asignar imágenes a productos de manera aleatoria. El diagrama de secuencia se puede apreciar en la figura 4.27. Cabe mencionar que este método satisface el requerimiento funcional RFGRA8.

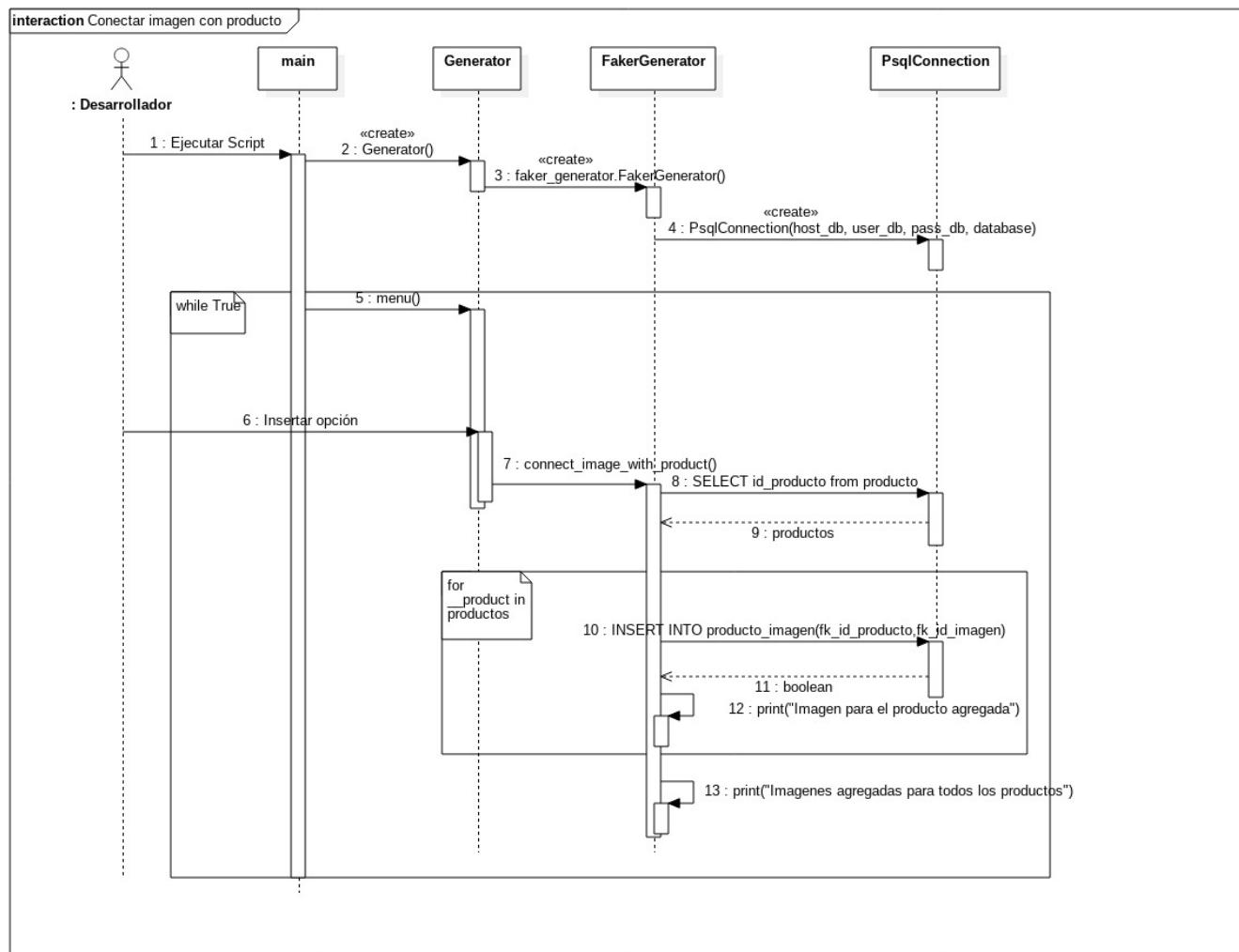


Figura 4.27: Diagrama de secuencia para asignar imágenes a productos.

### Generar empleados

Método creado con el objetivo de crear empleados con las personas ficticias que han sido registradas anteriormente por el método para generar personas; este método satisface el requerimiento funcional RFGRA9 y su respectivo diagrama de secuencia se encuentra en la figura 4.28.

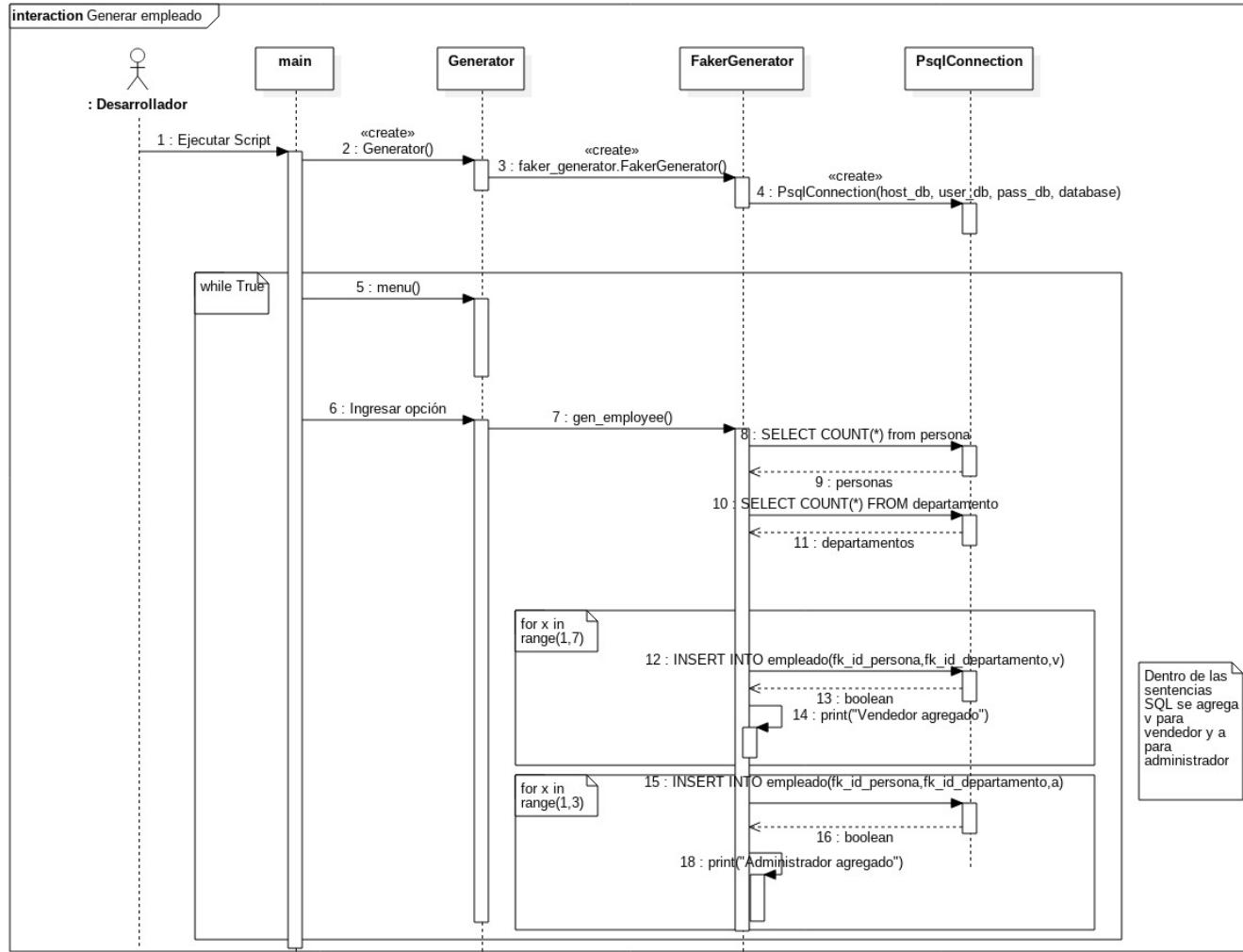


Figura 4.28: Diagrama de secuencia para generar empleados.

### Generar clientes

En el diagrama de secuencia que se muestra en la figura 4.29 muestra la interacción del método encargado de generar clientes ficticios que satisfacen el requerimiento funcional RFGRA10.

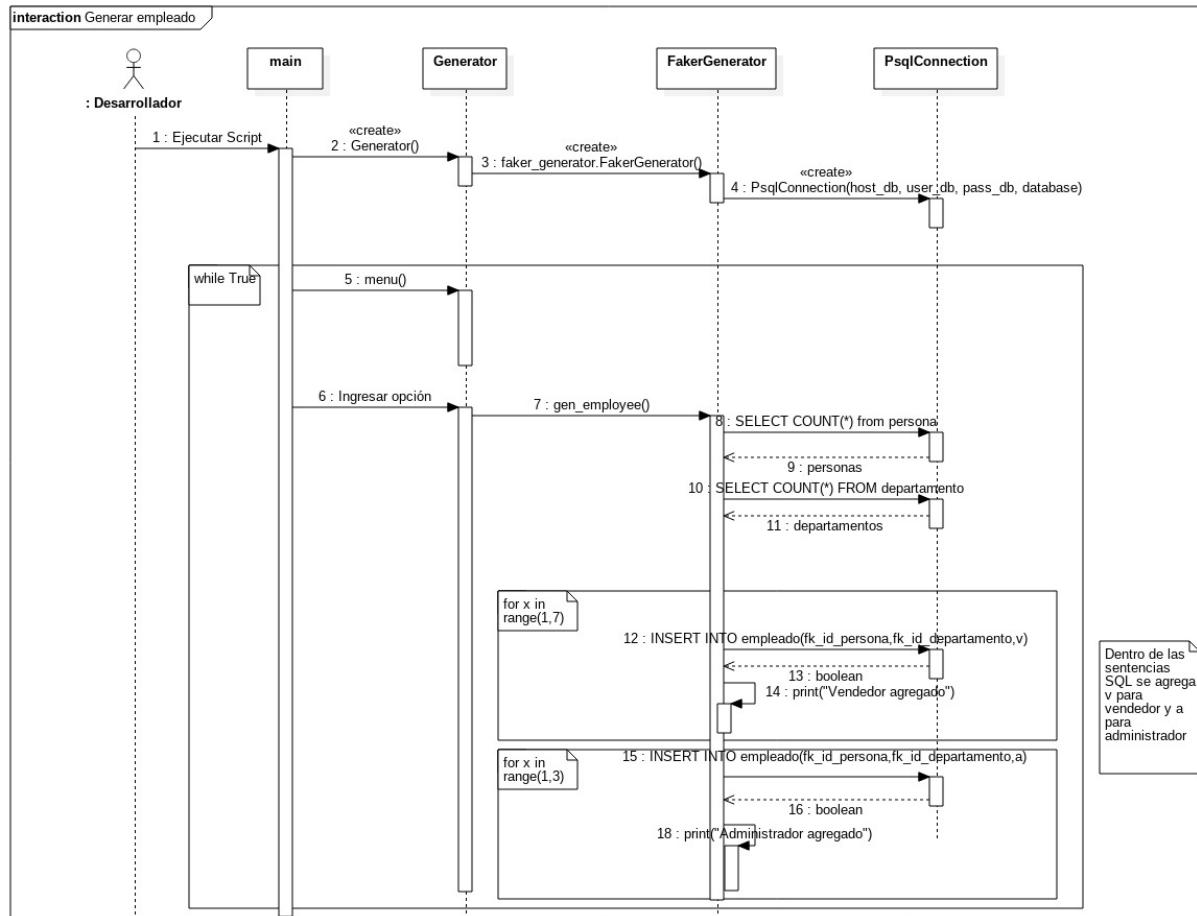


Figura 4.29: Diagrama de secuencia para generar clientes.

**Asignar productos favoritos a clientes** Este método se encarga de asignar productos favoritos a clientes de manera que crea un patrón dentro de los datos. El diagrama de secuencia se puede apreciar en la figura 4.30. Cabe mencionar que este método satisface el requerimiento funcional **RFGRA11**.

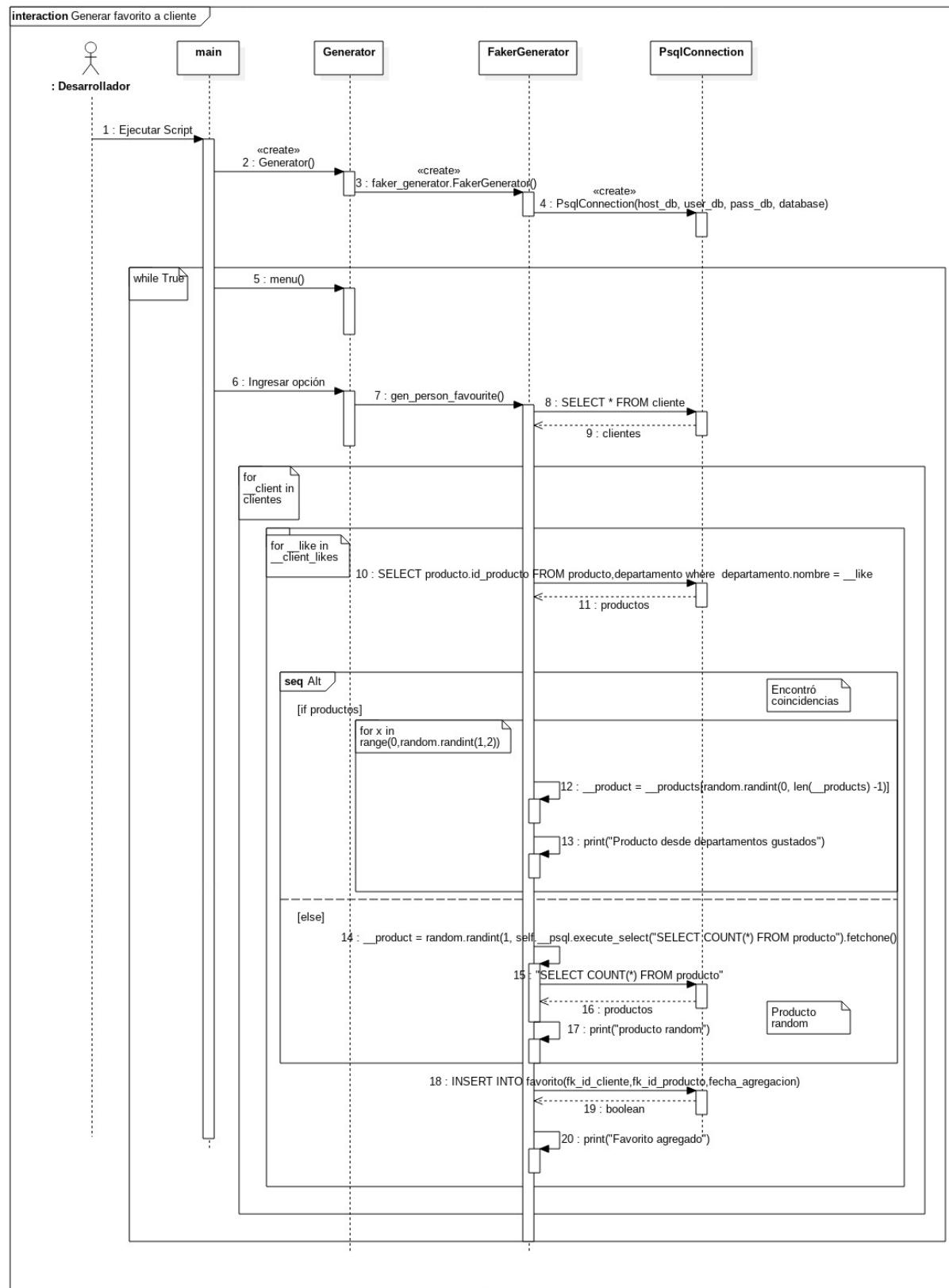


Figura 4.30: Diagrama de secuencia para asignar productos favoritos a cliente

La Figura 4.31 muestra el menú del Generador de Registros Artificiales.

```
~/Documents/ESCOM/TT/TTUltraMaximoTurbo(master*) » python generadorRegistros/generador.py
1.- Generador de personas
2.- Generador de direcciones
3.- Generador de tiendas
4.- Generador de departamentos
5.- Generador de productos
6.- Generador de images para productos
7.- Generador de conectar imagenes y productos
8.- Generador de empleados
9.- Generador de clientes
10.- Generador de favoritos a clientes
11.- Salir
Selecciona el numero de opción
```

Figura 4.33: UIRA: Menú

Figura 4.31: UIRA: Menú

## 4.8. Prototipo 2.1: Generador de Registros Artificiales (GRA)

En la sección actual se encuentran las modificaciones y/o agregaciones que se hicieron al Generador de Registros Artificiales.

### 4.8.1. Modificaciones realizadas

- Se agregó el caso de uso **CUGRA11: Agregar compras a clientes**.
- Se agregó el caso de uso **CUGRA12: Agregar logros y nivel a clientes**.

### 4.8.2. Análisis

#### Diagrama de casos de uso

En la figura 4.32 se muestra el diagrama de casos de uso del módulo Generador de Registros Artificiales.

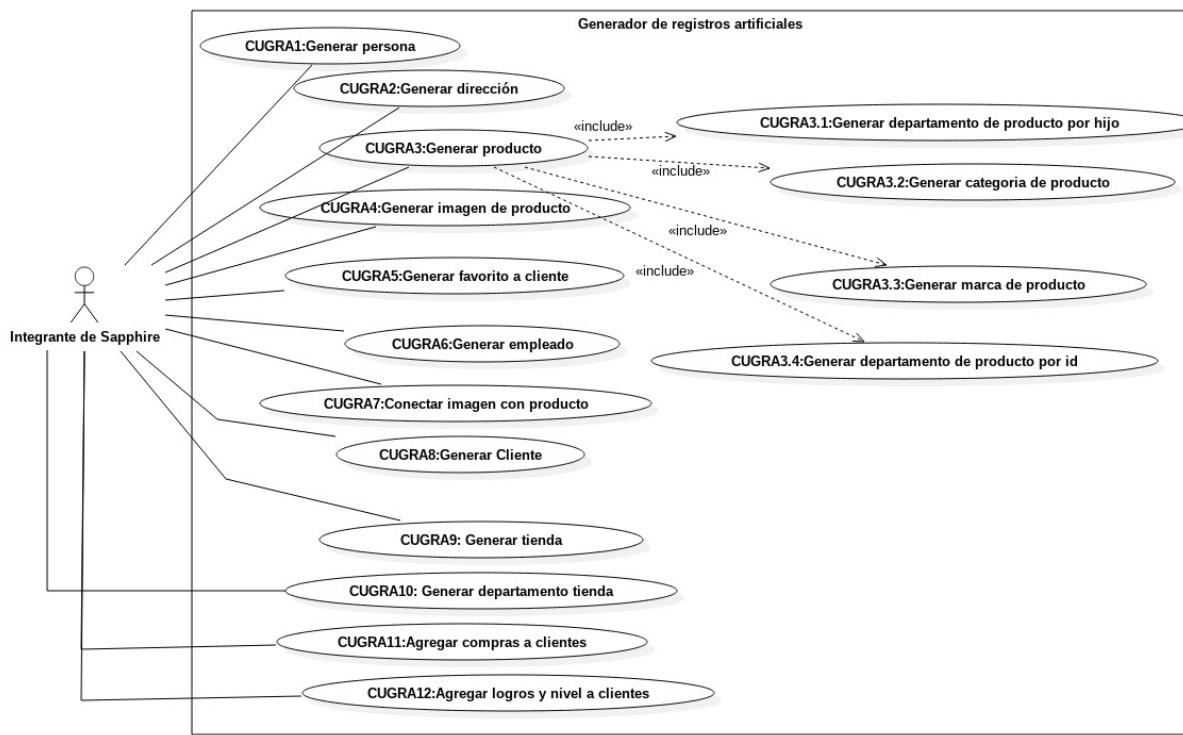


Figura 4.32: Diagrama de casos de uso módulo Generador de Registros Artificiales.

### Diagrama de clases

La descripción de los elementos en el diagrama de clases (figura 4.33) es la siguiente:

- Generator:** Clase encargada de la interacción con el usuario, muestra el menú y lee la opción ingresada por el usuario.
- FakerGenerator:** Clase encargada de tener la lógica para generar los registros artificiales.
- MysqlConnection:** Clase encargada de manejar la comunicación entre el módulo GRA y el gestor de base de datos MySQL.
- PsqlConnection:** Clase encargada de manejar la comunicación entre el módulo GRA y el gestor de base de datos PostgreSQL.

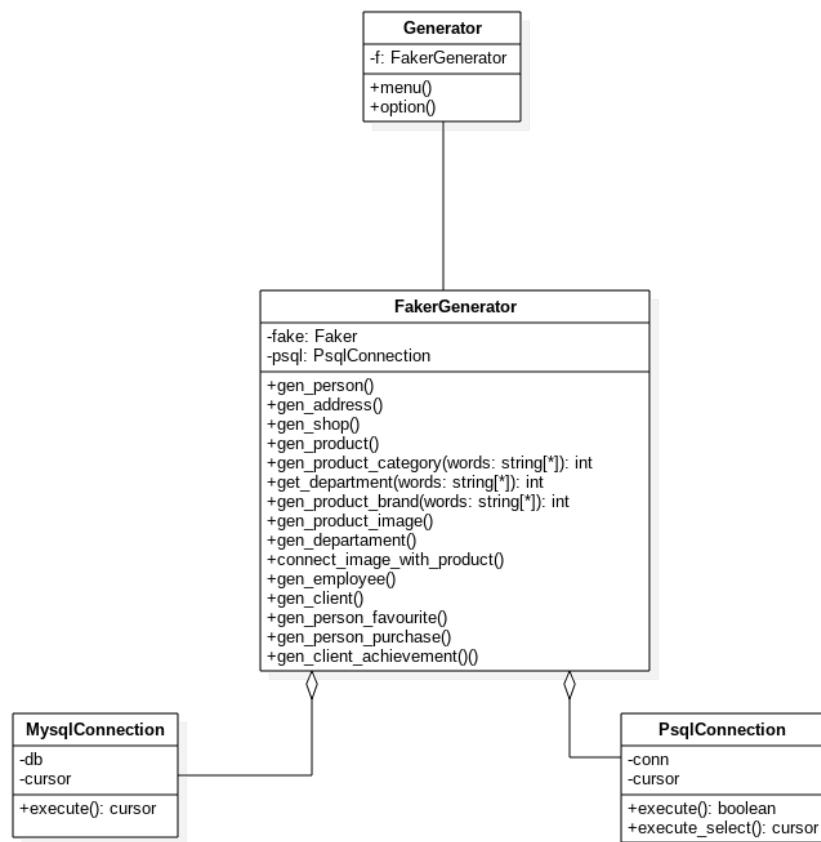


Figura 4.33: Diagrama de clases de módulo Generador de Registros Artificiales.

#### 4.8.3. Diseño

##### Diagramas de secuencia

Durante este apartado se realizará una pequeña explicación de cada método agregado en este prototipo dentro de la clase FakerGenerator, además, dentro del mismo se muestran los diagramas de secuencia de dichos métodos separados dentro de sus respectivos apartados.

##### Agregar compras a clientes

Método encargado de cubrir el requerimiento funcional **RFGRA12** que permite realizar el registro de compras ficticias en el repositorio de datos. En la figura 4.34 se muestra su diagrama de secuencia.

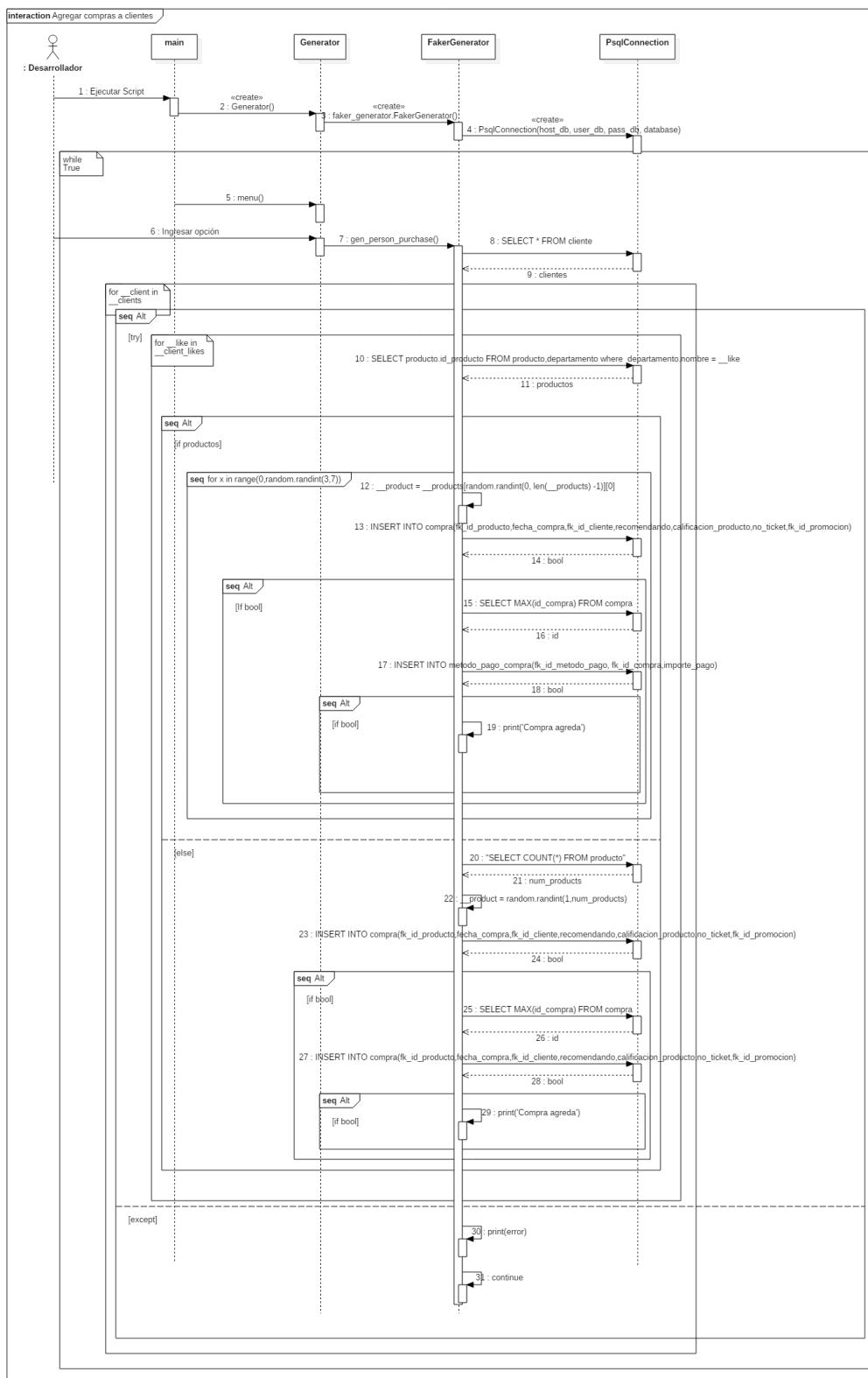


Figura 4.34: Diagrama de secuencia para generar compras a clientes.

### Asignar logros y nivel a clientes

Método encargado de cubrir el requerimiento funcional **RFGRA13** que actualiza los logros y el nivel de los clientes. En la figura 4.35 se muestra su diagrama de secuencia.

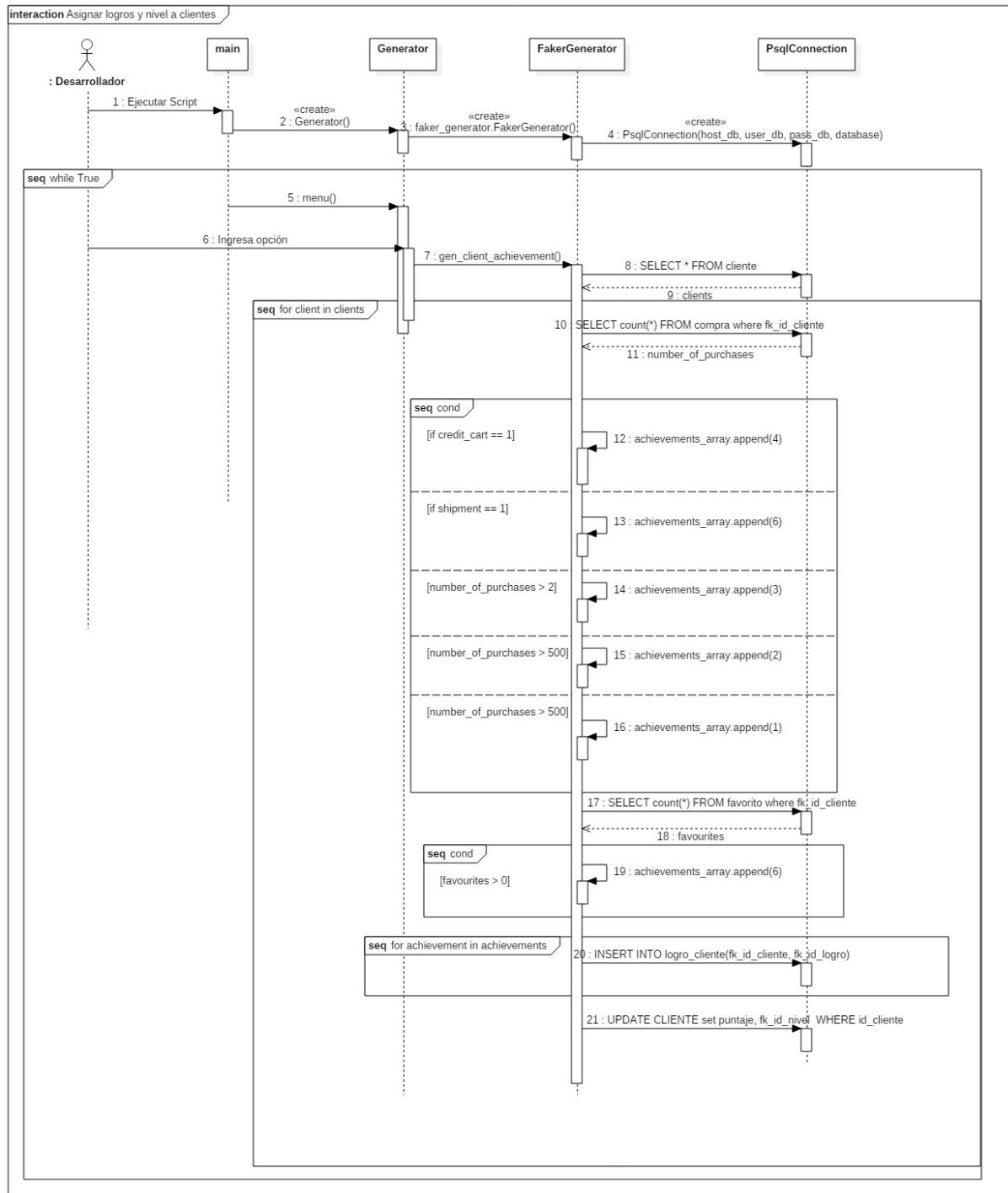


Figura 4.35: Diagrama de secuencia para asignar logros y nivel a clientes.

Finalmente la figura 4.36 muestra el menú del Generador de Registros Artificiales actualizado.

```
C:\Users\ruben\Documents\projects\tt\generadorRegistros (master -> origin)
(tt) λ python generador.py
1.- Generador de personas
2.- Generador de direcciones
3.- Generador de tiendas
4.- Generador de departamentos
5.- Generador de productos
6.- Generador de imágenes para productos
7.- Generador de conectar imágenes y productos
8.- Generador de empleados
9.- Generador de clientes
10.- Generador de favoritos a clientes
11.- Generador de compras a clientes
12.- Asignar logros y nivel a clientes
13.- Salir
Selecciona el numero de opción
```

Finalmente la figura 4.36 muestra el menú del Generador de Registros Artificiales actualizado.

```
python generadorRegistros
1.- Generador de personas
2.- Generador de direcciones
3.- Generador de tiendas
4.- Generador de departamentos
5.- Generador de productos
6.- Generador de imágenes para productos
7.- Generador de conectar imágenes y productos
8.- Generador de empleados
9.- Generador de clientes
10.- Generador de favoritos a clientes
11.- Generador de compras a clientes
12.- Salir
```

Figura 4.36: UIRA: Menú

# CAPÍTULO 5

---

## Aplicaciones móviles

---

Este capítulo abarca todo el desarrollo de los diferentes prototipos a realizar sobre las dos aplicaciones móviles con SO Android, contempladas en el sistema, siendo la primera, enfocada al uso de los clientes de las tiendas departamentales y la segunda dirigida a los vendedores dentro de dichos negocios.

### 5.1. Aplicación Interactiva Difusora de Productos (AIDP)

La Aplicación Interactiva Difusora de Productos, tiene como objetivo brindar un apoyo en la toma de decisiones de los clientes respecto a la compra de los diferentes tipos de productos que se puedan encontrar en un centro comercial y que resulten ser de su interés.

A continuación se presenta en la figura 5.1, la arquitectura de este módulo y posteriormente en la parte inferior, una pequeña explicación sobre los módulos que componen a dicha arquitectura.

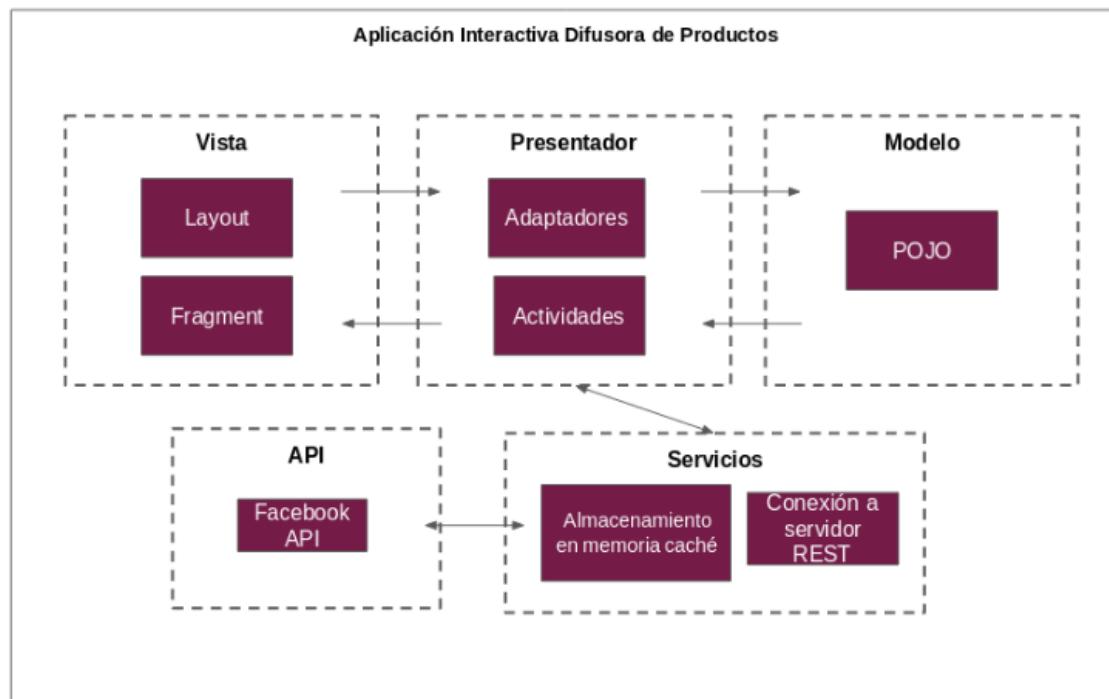


Figura 5.1: Arquitectura del módulo AIDP.

En la imagen mostrada anteriormente (Figura 5.1) se pueden observar las 3 áreas principales que componen a dicha arquitectura, mismas que corresponden a una arquitectura MVP.

- Vista: Corresponde a los archivos .xml los cuales corresponden a los layouts y fragmentos que desplegarán la información visual a los clientes y usuarios de la aplicación.
- Presentador: Correspondiente a las interfaces de los adaptadores y actividades, los cuales se encargan de interconectar las clases simples JAVA con los layouts y así mismo, del envío y recepción de peticiones al servidor REST y a la memoria caché del celular.
- Modelo: Corresponde a los POJO, mismos que se refieren a clases simples JAVA.
- Servicios: Son tanto la conexión al servidor REST como el almacenamiento en memoria caché del dispositivo móvil.
- API: Dentro de esta sección se encuentra la API de Facebook, utilizadas para obtener los datos e información del usuario.

### 5.1.1. Prototipo 1: Diseño inicial de la aplicación

#### 5.1.1.1. Análisis

Se ha definido previamente la problemática a resolver, identificando los componentes principales que deberán ser integrados al sistema, [95] para ello, se utilizan ciertas herramientas tales como la definición de requerimientos funcionales, que ha sido presentada previamente en el capítulo de “Bosquejo general de la aplicación” con el título de “**Requerimientos Funcionales de Aplicación Interactiva Difusora de Productos**”. En la sección de análisis se planifica la integración de dichos requerimientos con el fin de saber cuales son las funciones que se requieren para que la aplicación móvil funcione adecuadamente.

La figura 5.2 presenta los casos de uso del módulo de Aplicación Interactiva Difusora de Productos. En ella se observan 8 casos de uso de los cuales se extienden y se incluyen otros casos de uso a su vez. También es importante destacar que hay una dependencia para el caso de uso 6.7.1 “Añadir productos a favoritos”.

### Casos de uso de AIDP

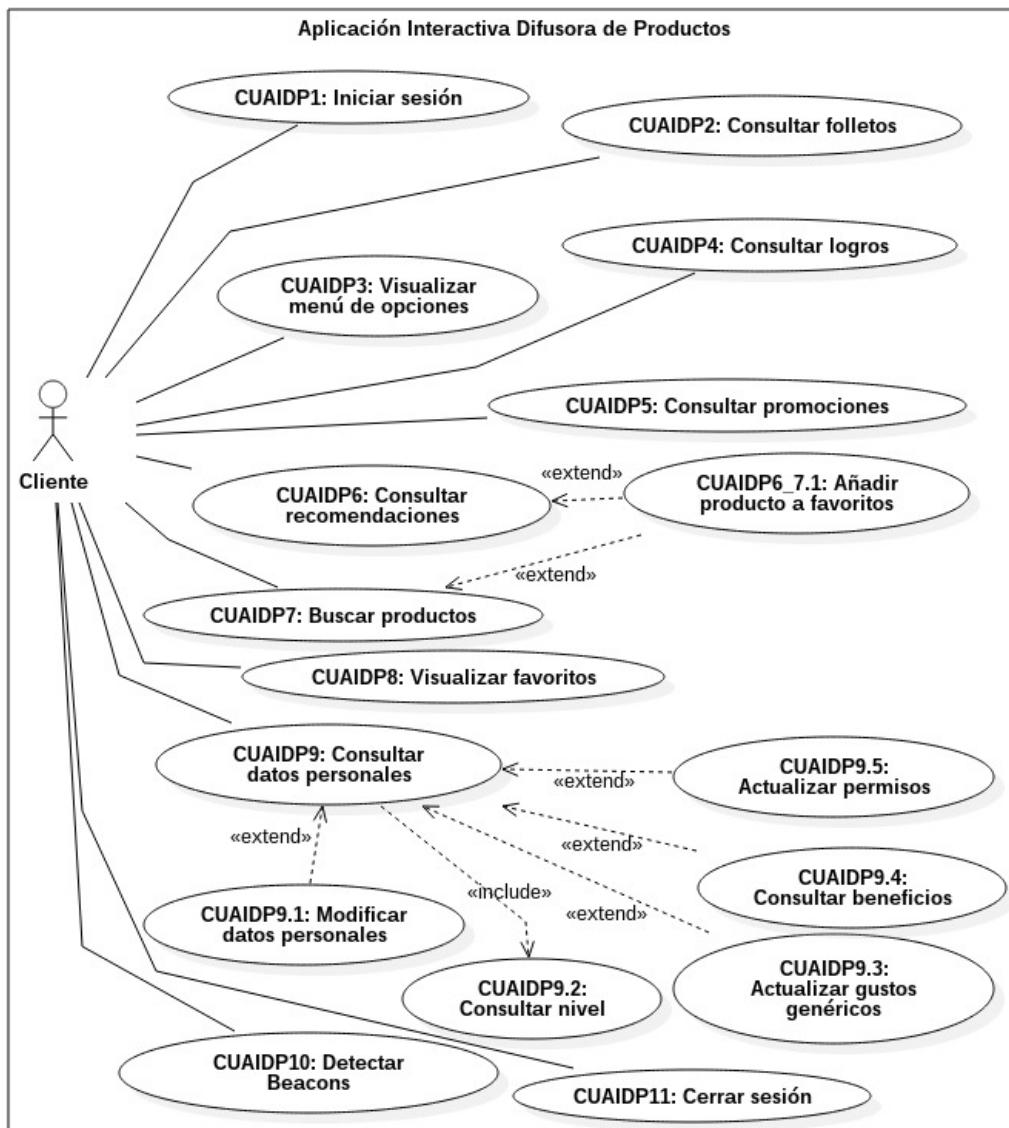


Figura 5.2: Casos de uso de la Aplicación Interactiva Difusora de Productos.

### 5.1.1.2. Diseño

Posteriormente en esta etapa y a partir de los requerimientos definidos anteriormente, se muestran tanto el diagrama de casos de uso de la aplicación, como los diagramas de secuencia que permiten una mejor comprensión del funcionamiento que cada clase tendrá en el módulo. Así mismo, se muestra el flujo de navegación de la aplicación.

## Diagrama de clases

El diagrama de la figura 5.3 se dividió en 3 secciones para obtener una mejor visualización, dichas secciones corresponden a la figura 5.4, 5.5 y 5.6.

*Nota: Las clases mostradas en este prototipo muestran únicamente los métodos y atributos básicos de cada una de ellas debido a que estas no contemplan correctamente el funcionamiento que proveerá cada una de esas clases. Por dicha razón, en los prototipos posteriores se describirán los elementos tanto de las clases que se han plasmado como de las que aún no han sido desarrolladas.*

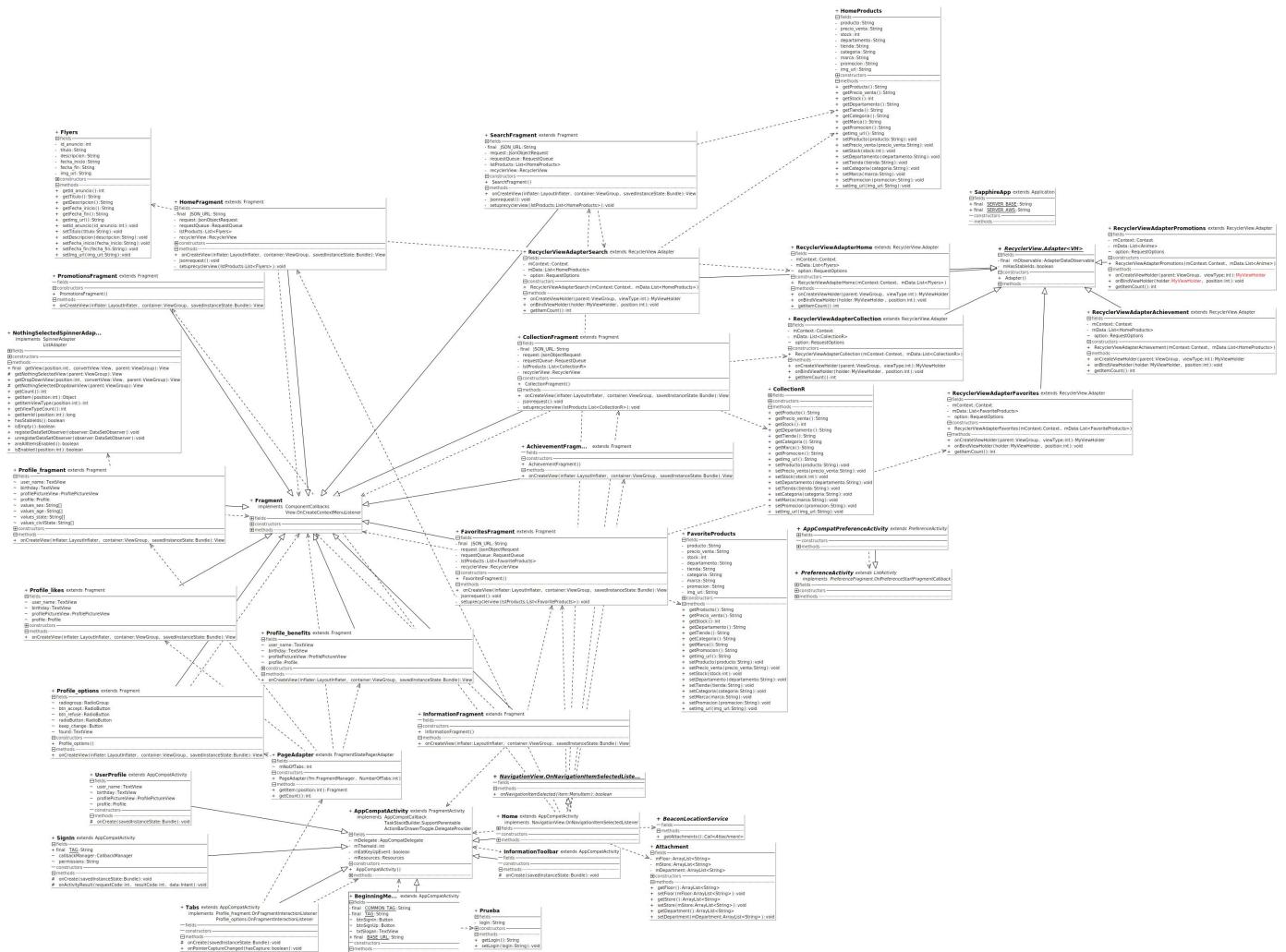


Figura 5.3: Diagrama de clases (Visualización completa).

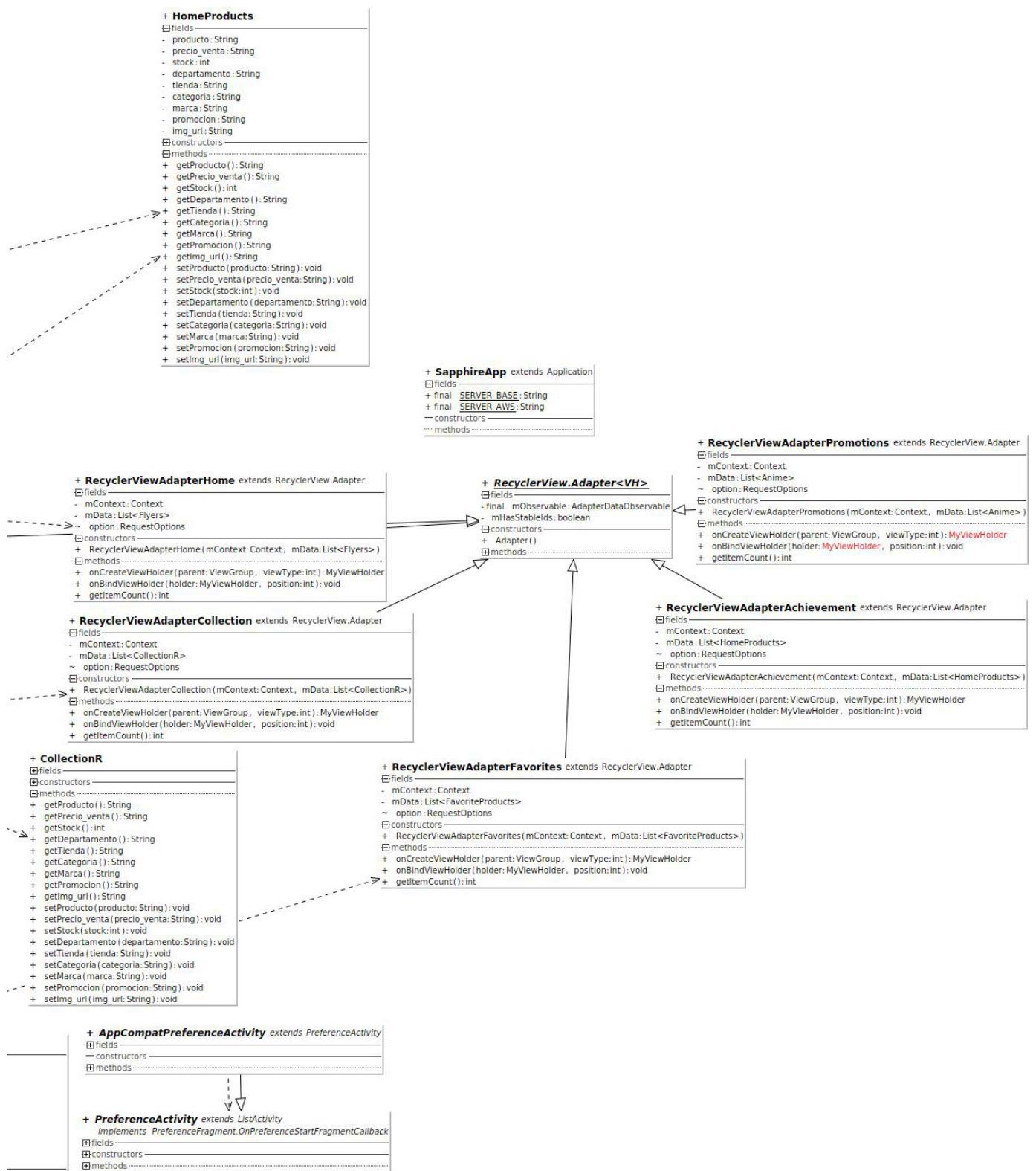


Figura 5.4: Diagrama de clases (Parte 1).

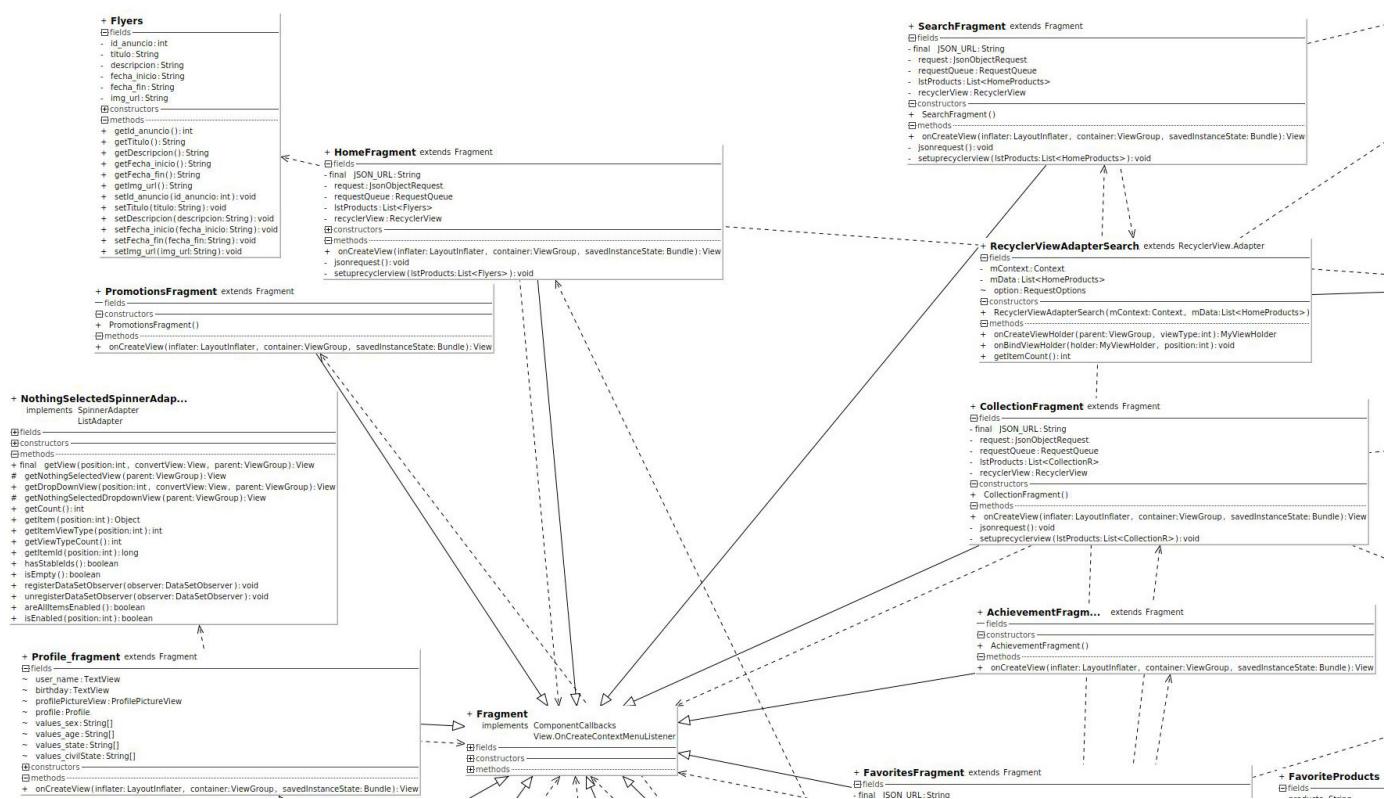


Figura 5.5: Diagrama de clases (Parte 2).

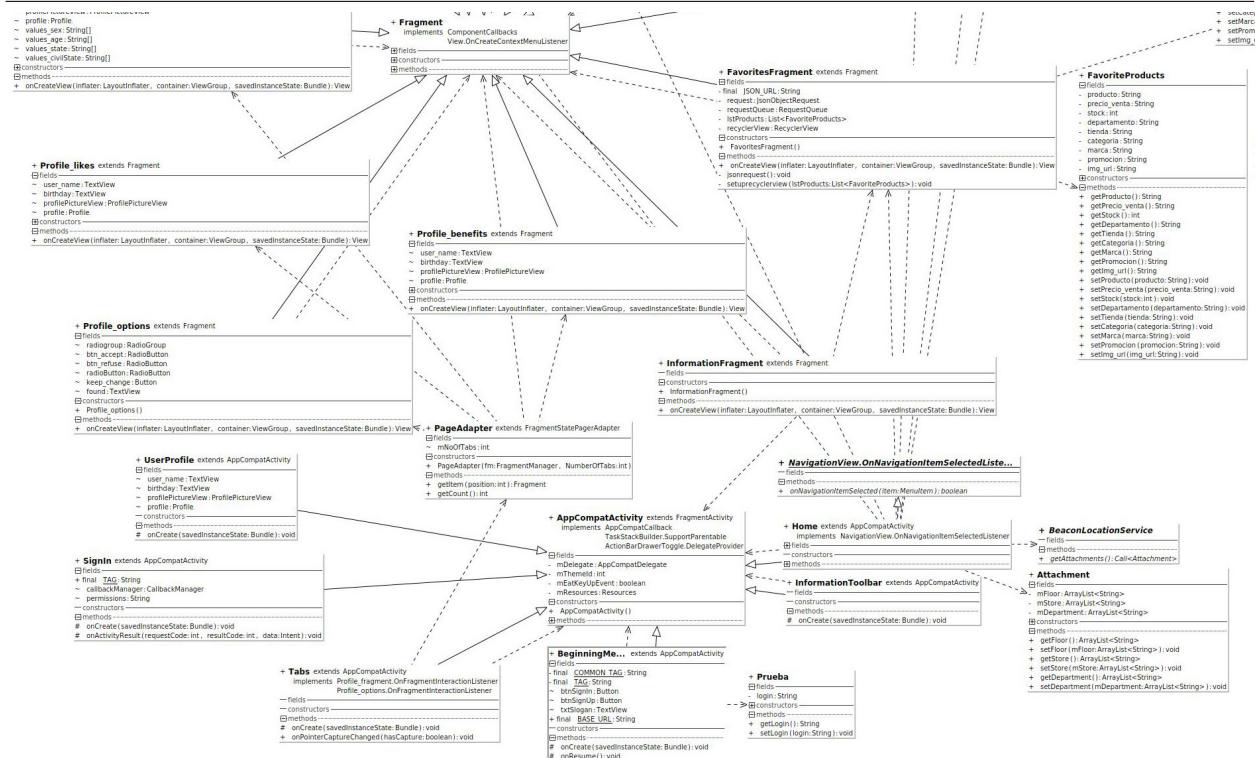


Figura 5.6: Diagrama de clases (Parte 3)

## Flujo de navegación de la Aplicación Interactiva Difusora de Productos

En la figura 5.7 se muestra el diagrama de como es el flujo de navegación de la aplicación del cliente, se ha dividido en dos partes debido al tamaño de este, la figura 5.8 y 5.9.

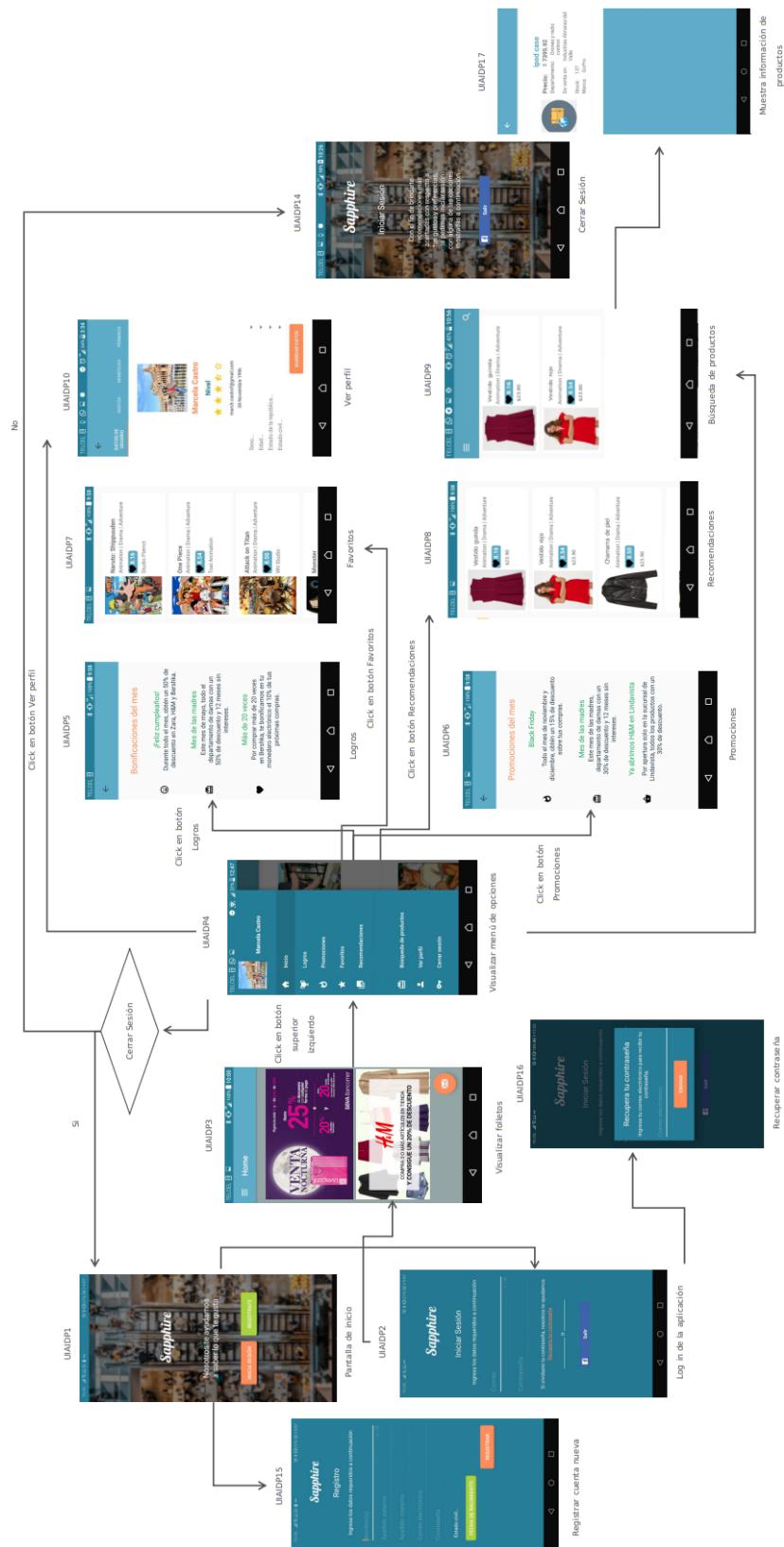


Figura 5.7: Flujo de navegación de la Aplicación Interactiva Difusora de Productos (Visualización completa).

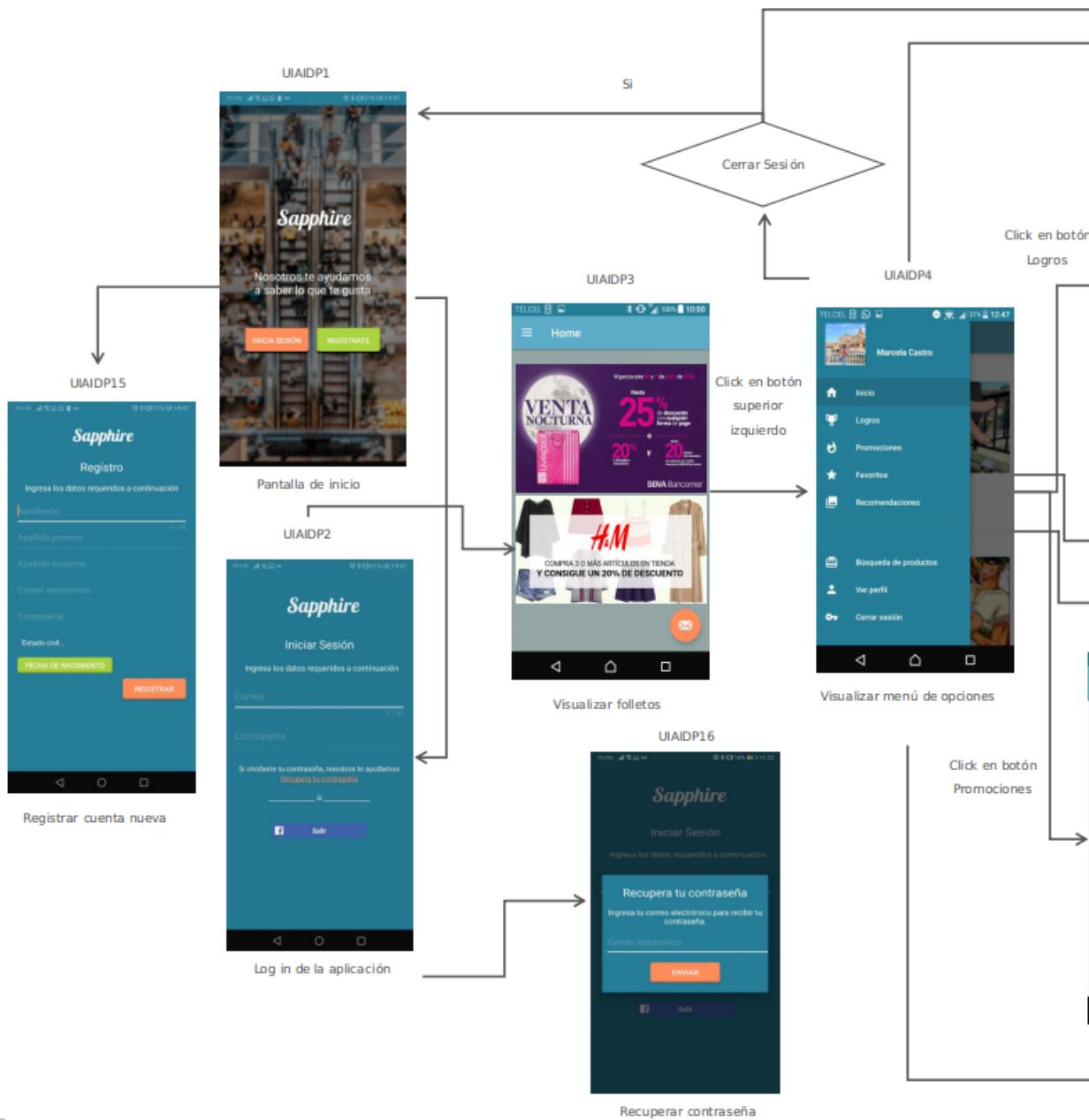


Figura 5.8: Flujo de navegación de la Aplicación Interactiva Difusora de Productos (Parte 1).

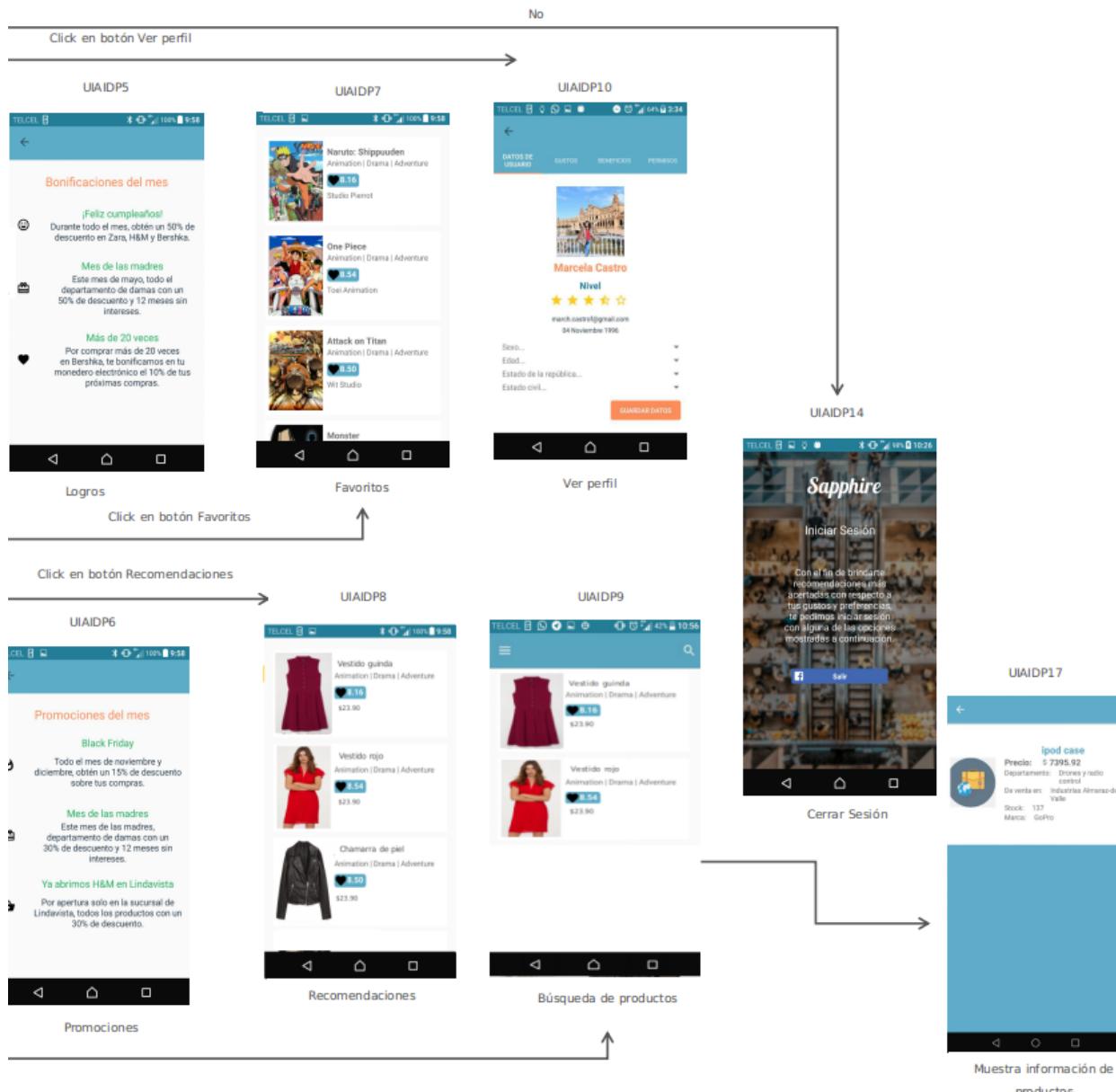


Figura 5.9: Flujo de navegación de la Aplicación Interactiva Difusora de Productos (Parte 2).

La figura 5.10 muestra 3 pantallas que son las vías alternativas para iniciar sesión por primera vez en la aplicación. El usuario puede elegir la opción de ingresar con su cuenta previa de Facebook, mediante el ingreso de otra cuenta diferente o por medio de la creación de una cuenta completamente nueva.

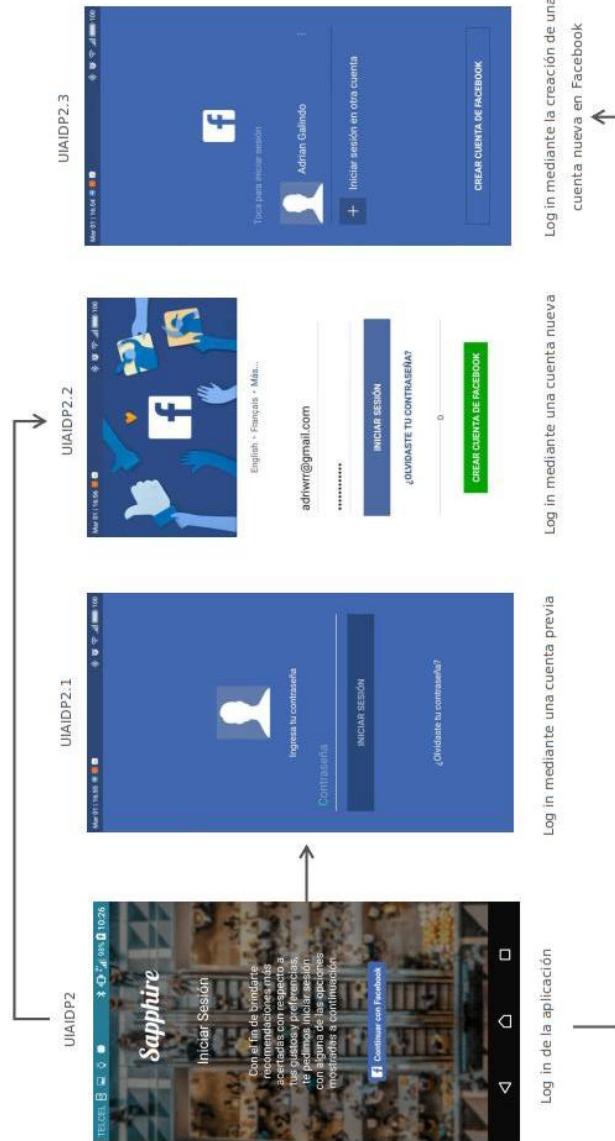


Figura 5.10: Flujo de navegación de los métodos para iniciar sesión (Visualización completa).

La figura 5.11 muestra 4 pantallas que se derivan de una de las opciones del menú lateral principal “Ver perfil” mostrado en la figura 5.7, esta figura de igual manera fue dividida en dos secciones (figura 5.12 y figura 5.13) para una mejor visualización de su contenido.

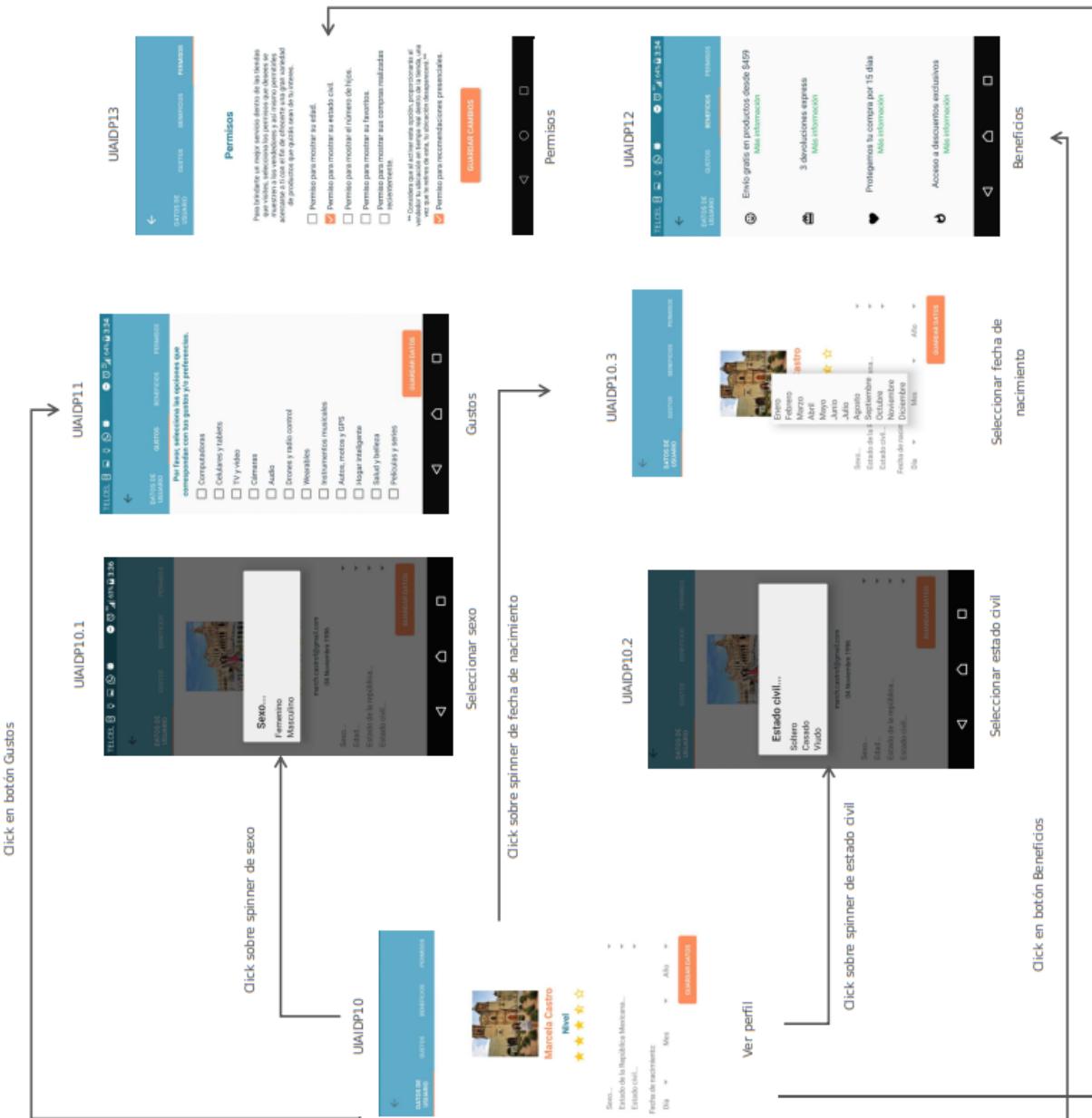


Figura 5.11: Flujo de navegación de la Aplicación Interactiva Difusora de Productos (Visualización completa).



Figura 5.12: Flujo de navegación de la Aplicación Interactiva Difusora de Productos (Parte 1).

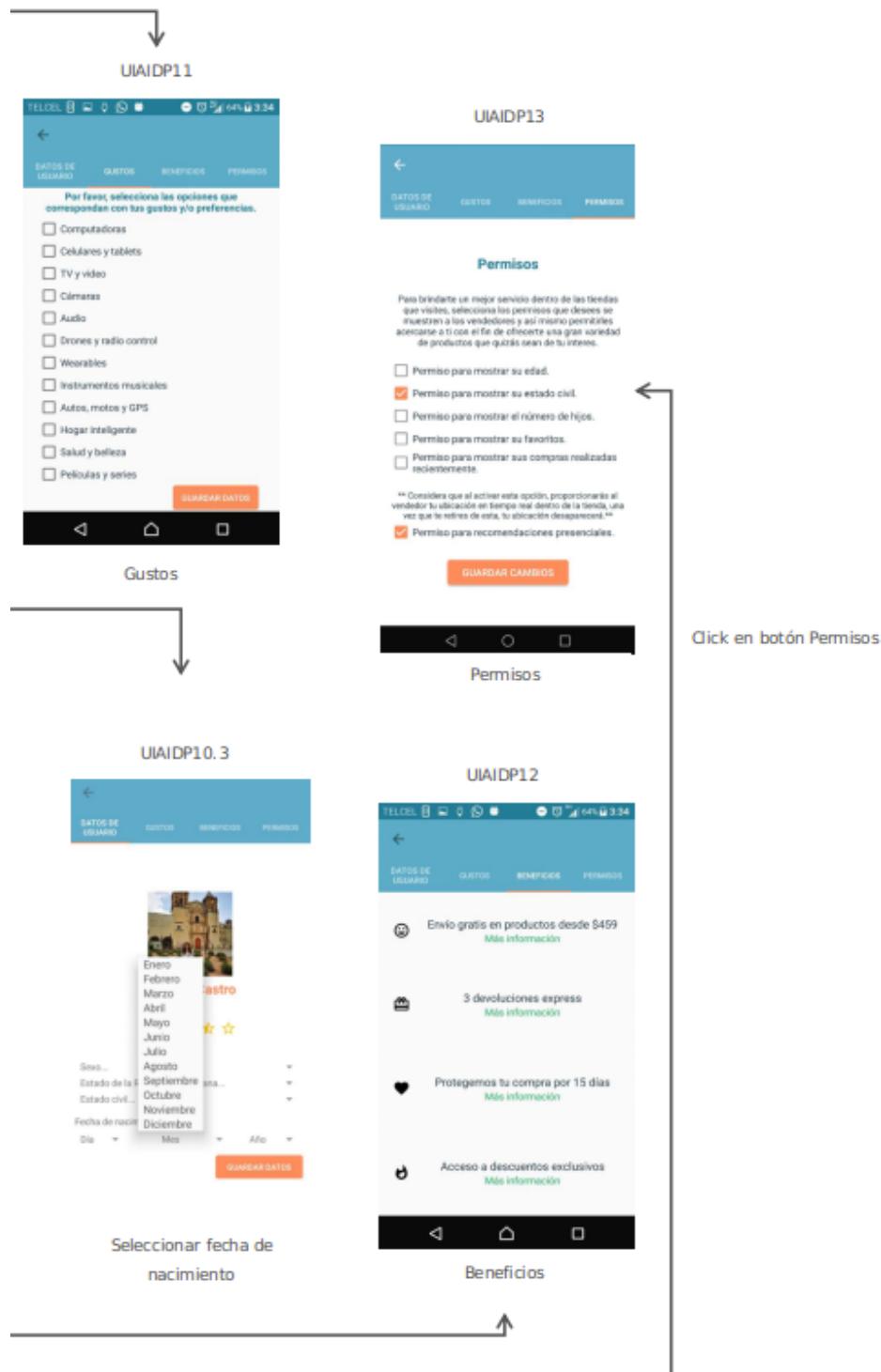


Figura 5.13: Flujo de navegación de la Aplicación Interactiva Difusora de Productos (Parte 2).

## UIAIDP1 - Pantalla de inicio

### Objetivo

Mostrar la pantalla inicial de la aplicación.

### Diseño

Esta pantalla (figura 5.14), aparece al iniciar la aplicación. Muestra el logo y slogan del proyecto y de igual manera proporciona la opción para iniciar sesión.



Figura 5.14: UIAIDP1 Pantalla de inicio.

### Salidas

Ninguna.

### Entradas

Ninguna.

## UIAIDP2 - Log in de la aplicación

### Objetivo

Mostrar las opciones de inicio de sesión.

### Diseño

La figura 5.15, es la pantalla que se muestra al usuario con el fin de informarles la opción que estos tienen para iniciar sesión e ingresar al sistema.



Figura 5.15: UIAIDP2 Log in de la aplicación.

### Salidas

Ninguna.

### Entradas

Correo electrónico y contraseña del usuario.

### UIAIDP2.1 - Log in mediante una cuenta previa

#### Objetivo

Mostrar una opción para el inicio de sesión.

#### Diseño

La figura 5.16, es la pantalla que se muestra al usuario para que este ingrese su contraseña y agilizar de esta forma el inicio de sesión a la aplicación.

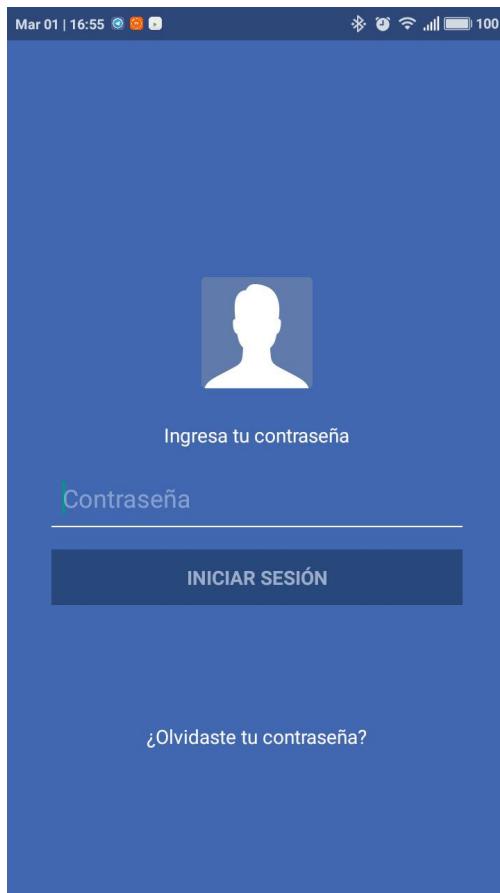


Figura 5.16: UIAIDP2.1 Log in mediante una cuenta previa.

#### Salidas

Ninguna.

#### Entradas

Contraseña del usuario.

## UIAIDP2.2 - Log in mediante una cuenta nueva

### Objetivo

Mostrar una opción para el inicio de sesión.

### Diseño

La figura 5.17, es la pantalla que se muestra al usuario para que este ingrese una cuenta diferente a la que ya tenga registrada en la aplicación móvil.



Figura 5.17: UIAIDP2.2 Log in mediante una cuenta nueva.

### Salidas

Ninguna.

### Entradas

Correo electrónico y contraseña del usuario.

### UIAIDP2.3 - Log in mediante la creación de una cuenta nueva en Facebook

#### Objetivo

Mostrar las opciones de inicio de sesión.

#### Diseño

La figura 5.18, es la pantalla que se muestra al usuario en la cual puede seleccionar la opción de creación de una cuenta completamente nueva.

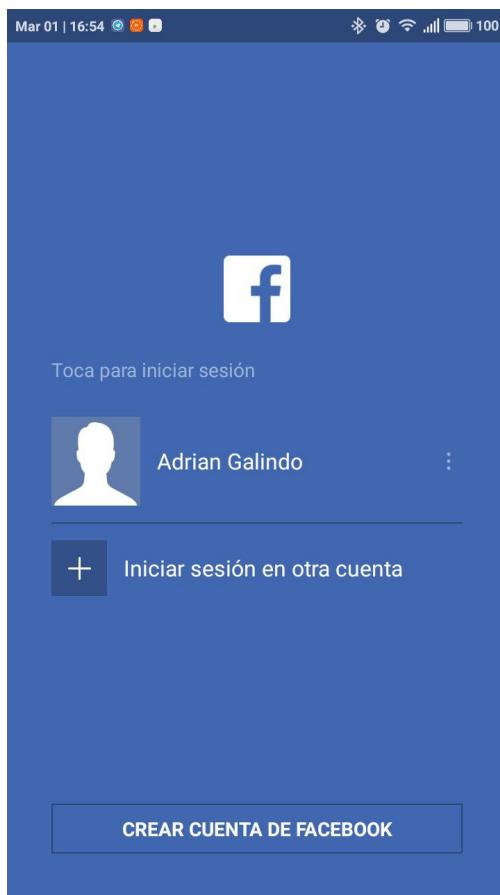


Figura 5.18: UIAIDP2.3 Log in mediante la creación de una cuenta nueva en Facebook.

#### Salidas

Ninguna.

#### Entradas

Correo electrónico, contraseña y nombre de perfil.

## UIAIDP2.4 - Log in solicitado por la API de Facebook

### Objetivo

Realizar el inicio de sesión en la aplicación Sapphire utilizando la API de Facebook.

### Diseño

La figura 5.19, muestra la pantalla en la cual el usuario deberá ingresar su correo electrónico y contraseña de Facebook con el fin de iniciar sesión en la aplicación de Sapphire por medio de su cuenta de Facebook.



Figura 5.19: UIAIDP2.4 Log in solicitado por la API de Facebook.

### Salidas

Ninguna.

### Entradas

Correo electrónico y contraseña.

## UIAIDP3 - Visualizar folletos

### Objetivo

Mostrar las promociones y productos con descuento.

### Diseño

La figura 5.20, muestra la pantalla de los diferentes productos que se encuentran en las diferentes tiendas y que cuentan con algún descuento o promoción en particular. Cumple la función de un folleto o propaganda proporcionado al ingresar a una tienda.



Figura 5.20: UIAIDP3 Visualizar folletos.

### Salidas

Ninguna.

### Entradas

Ninguna.

## UIAIDP3.1 - Detectar Beacons

### Objetivo

Mostrar alerta al usuario al detectar un Beacon.

### Diseño

La figura 5.21, muestra la pantalla anterior “UIAIDP3 - Visualizar folletos” en la cual se despliega una pequeña alerta en la parte inferior con la cuál se le notifica al cliente que ha ingresado a un nuevo piso o departamento.



Figura 5.21: UIAIDP3.1 Detectar Beacons.

### Salidas

Ninguna.

### Entradas

Ninguna.

## UIAIDP4 - Visualizar menú de opciones

### Objetivo

Mostrar las opciones de navegación con las que el usuario cuenta.

### Diseño

La pantalla inferior (figura 5.22), muestra un menú en la parte superior izquierda el cual despliega las diferentes opciones en las que el usuario puede encontrar ofertas o productos de su agrado.

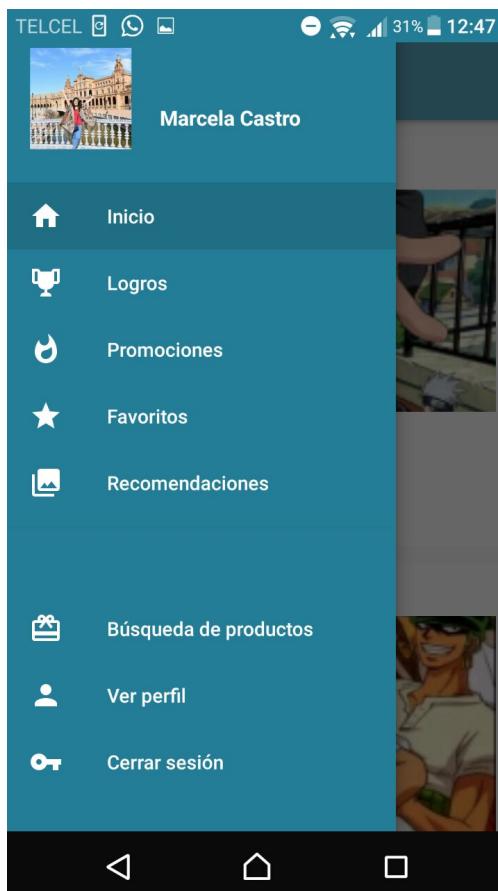


Figura 5.22: UIAIDP4 Visualizar menú de opciones.

### Salidas

Ninguna.

### Entradas

Ninguna.

## UIAIDP5 - Logros

### Objetivo

Mostrar las bonificaciones que las tiendas tienen con motivo de una fecha especial.

### Diseño

En la figura 5.23, el usuario visualiza las bonificaciones que se realizan al usuario con motivo de una fecha especial en el mes como su cumpleaños o navidad, por ejemplo.



Figura 5.23: UIAIDP5 Logros.

### Salidas

Ninguna.

### Entradas

Ninguna.

## UIAIDP6 - Promociones

### Objetivo

Mostrar las promociones que las tiendas ofrecen al cliente.

### Diseño

Las diferentes promociones que se realizan en un mes en específico por ejemplo en diciembre, el “Black Friday” o descuentos por el día de las madres, son mostradas en esta pantalla (figura 5.24).

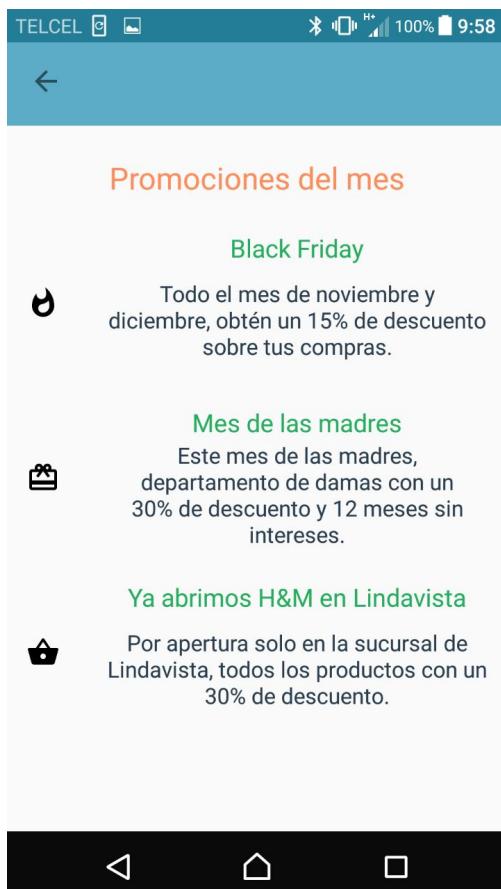


Figura 5.24: UIAIDP6 Promociones.

### Salidas

Ninguna.

### Entradas

Ninguna.

## UIAIDP7 - Favoritos

### Objetivo

Mostrar los productos que el cliente ha marcado como favoritos.

### Diseño

La figura 5.25 muestra la pantalla en la que el cliente puede visualizar los productos que previamente ha marcado como favoritos.

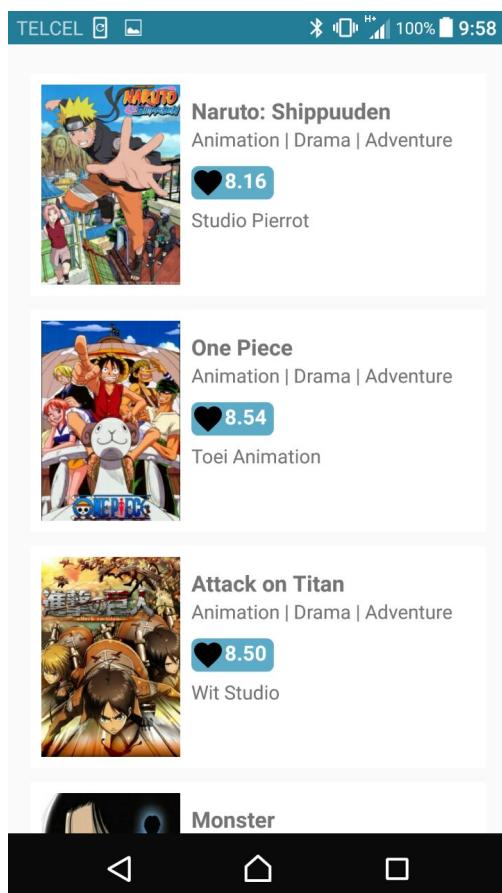


Figura 5.25: UIAIDP7 Favoritos.

### Salidas

Ninguna.

### Entradas

Ninguna.

## UIAIDP8 - Recomendaciones

### Objetivo

Mostrar las recomendaciones generadas por el módulo de recomendaciones FC.

### Diseño

La imagen de la figura 5.26 presenta la pantalla donde se muestran todos los productos que han sido recomendados por parte del módulo de recomendaciones, mismos que han sido seleccionados con base a los gustos del cliente.

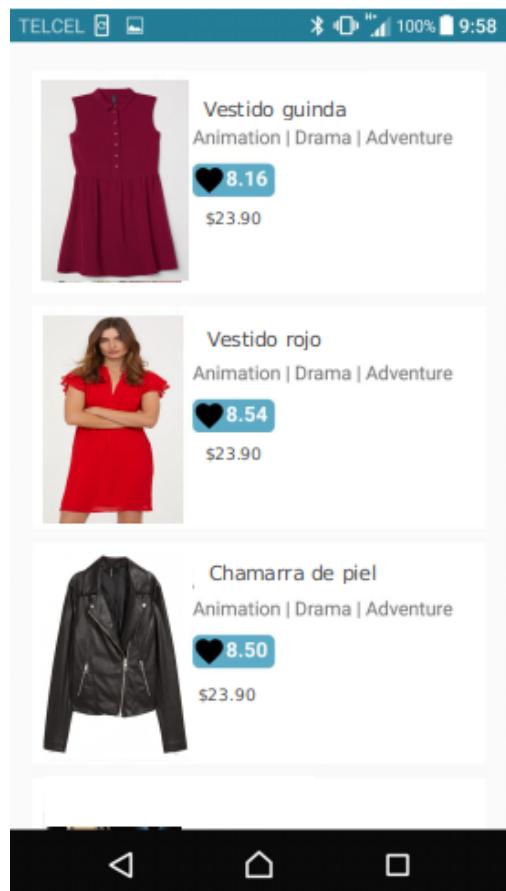


Figura 5.26: UIAIDP8 Recomendaciones.

### Salidas

Ninguna.

### Entradas

Ninguna.

## UIAIDP9 - Búsqueda de productos

### Objetivo

Desplegar los productos que busque el cliente.

### Diseño

En la figura 5.27 que muestra la pantalla de búsqueda de productos, el usuario tiene la opción de buscar los productos que sean de su interés por ejemplo un vestido y visualizarlos según sean localizados.

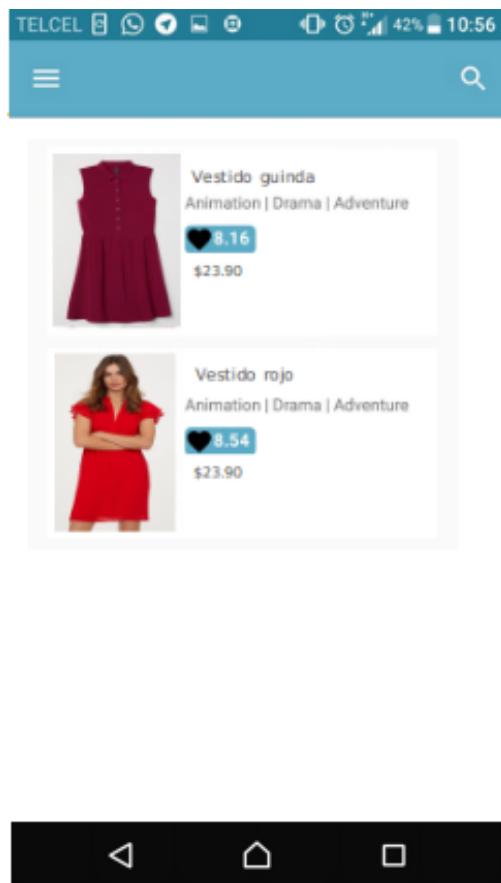


Figura 5.27: UIAIDP9 Búsqueda de productos.

### Salidas

Ninguna.

### Entradas

Ninguna.

## UIAIDP10 - Ver perfil

### Objetivo

Visualizar los datos del usuario.

### Diseño

Esta pantalla (figura 5.28), despliega los datos y la imagen del usuario y de igual manera, proporciona al usuario la posibilidad de modificar ciertos datos específicos tales como su edad y estado civil.

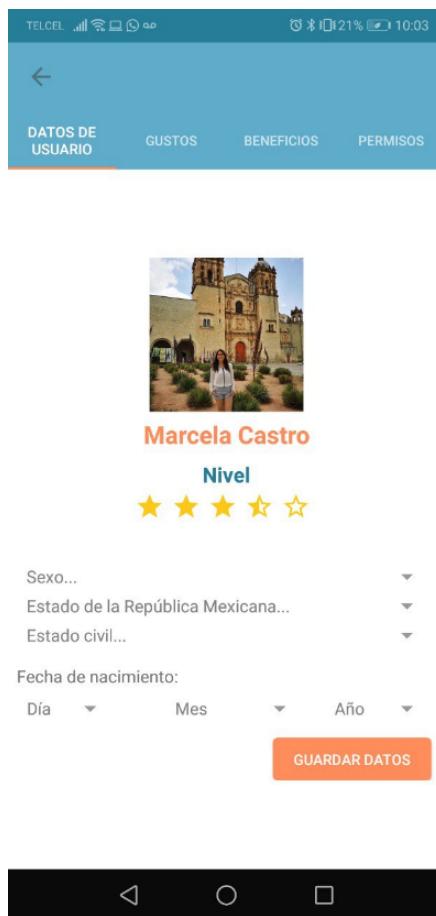


Figura 5.28: UIAIDP10 Ver perfil.

### Salidas

Ninguna.

### Entradas

Sexo, edad y estado civil.

## UIAIDP10.1 - Seleccionar sexo

### Objetivo

Seleccionar el sexo del usuario.

### Diseño

La figura 5.29 se muestra el despliegue de las dos diferentes opciones para que el usuario elija el sexo que corresponde.



Figura 5.29: UIAIDP10.1 Seleccionar sexo.

### Salidas

Ninguna.

### Entradas

Sexo.

## UIAIDP10.2 - Seleccionar estado civil

### Objetivo

Seleccionar su estado civil actual.

### Diseño

La figura 5.30 se muestra el despliegue de los estados civiles de los que el usuario seleccionará su opción.

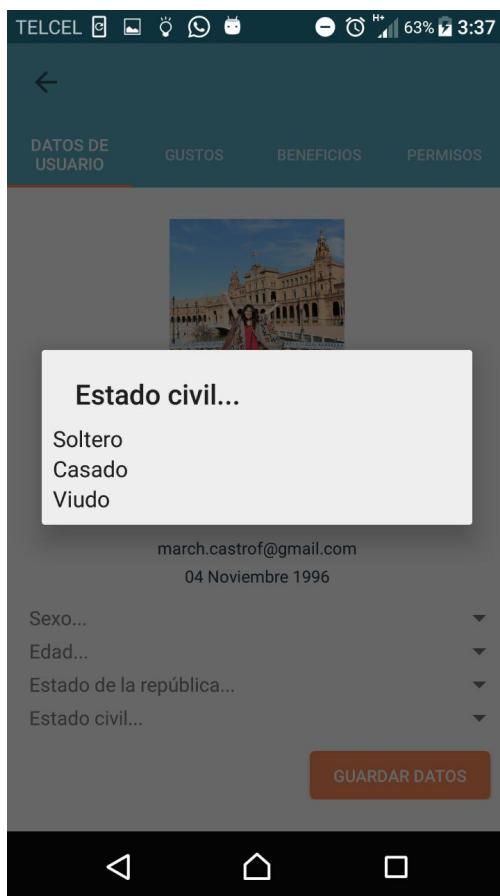


Figura 5.30: UIAIDP10.4 Seleccionar estado civil.

### Salidas

Ninguna.

### Entradas

Estado civil.

### UIAIDP10.3 - Seleccionar fecha de nacimiento

#### Objetivo

Seleccionar la fecha de nacimiento del usuario.

#### Diseño

La figura 5.31 se muestra el despliegue de los 3 datos que conforman la fecha de nacimiento.

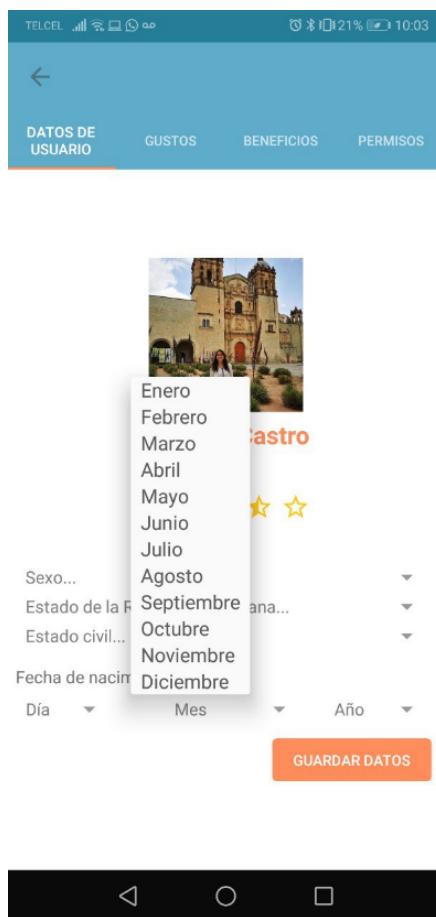


Figura 5.31: UIAIDP10.2 Seleccionar fecha de nacimiento.

#### Salidas

Ninguna.

#### Entradas

Día, mes y año de nacimiento.

## UIAIDP11 - Gustos

### Objetivo

Elegir los gustos que se asemejen más a los del usuario.

### Diseño

Esta pantalla (figura 5.32), muestra una serie de categorías diferentes que el usuario puede o no seleccionar según correspondan con sus gustos.

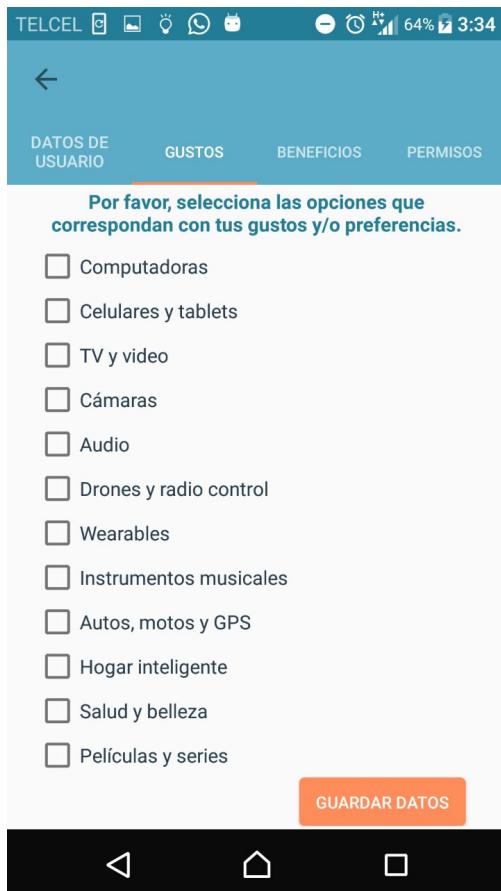


Figura 5.32: UIAIDP11 Gustos.

### Salidas

Ninguna.

### Entradas

Gustos.

## UIAIDP12 - Beneficios

### Objetivo

Consultar los beneficios que ofrece el nivel.

### Diseño

La figura 5.33 muestra la sección de Beneficios en la que dependiendo el nivel en el que se encuentre el cliente, este obtendrá diferentes beneficios que serán más atractivos para el usuario entre más alto sea su nivel.



Figura 5.33: UIAIDP12 Beneficios.

### Salidas

Ninguna.

### Entradas

Ninguna.

## UIAIDP13 - Permisos

### Objetivo

Solicitar al cliente permiso para mostrar en la aplicación de vendedor los diferentes permisos.

### Diseño

La figura 5.34 muestra los permisos de los cuales el cliente decidirá si va a permitir a los vendedores tanto aproximarse a él para ofrecerle diferentes productos que pueden resultar de su interés, como visualizar su información básica personal.

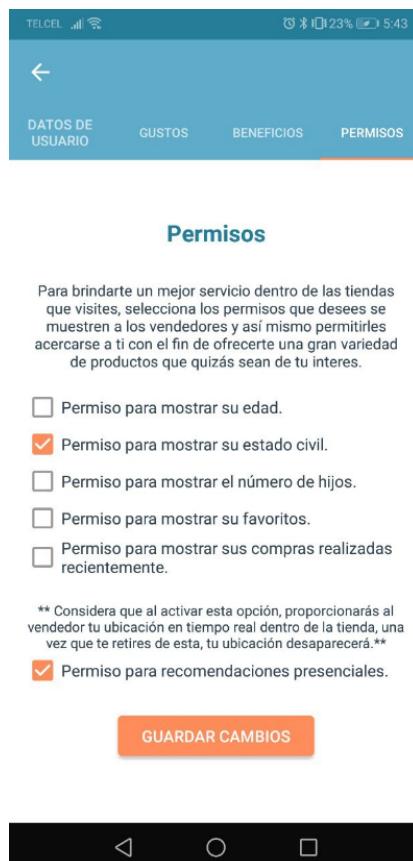


Figura 5.34: UIAIDP13 Proporcionar permisos.

### Salidas

Ninguna.

### Entradas

Permisos.

## UIAIDP14 - Cerrar sesión

### Objetivo

Cerrar la sesión activa.

### Diseño

Esta pantalla (figura 5.35), muestra una de las opciones que el usuario tiene para cerrar su sesión dentro de la aplicación, también puede hacer dicha acción desde el menú de opciones, presionando el último botón “Cerrar sesión”.



Figura 5.35: UIAIDP14 Cerrar sesión.

### Salidas

Ninguna.

### Entradas

Ninguna.

## 5.1.2. Prototipo 2: Conexión de la AIDP con Beacon

### 5.1.2.1. Análisis

Dentro de este prototipo se satisface el requerimiento funcional [Detectar Beacons](#), definido previamente en el capítulo del “Bosquejo general de la aplicación” con el título de “Requerimientos funcionales de la Aplicación Interactiva Difusora de Productos (RFAIDP)” y mismo que ya ha sido contemplado en los casos de uso mostrados en la figura 5.40 en el prototipo 1.

### 5.1.2.2. Diseño

En el prototipo siguiente, se añadió a la AIDP la conexión con los Beacon, se muestra a continuación el diagrama de secuencia que muestra el funcionamiento que realiza la aplicación al detectar dicho dispositivo cerca.

#### Diagramas de secuencia

##### Detectar Beacons

El diagrama de la figura 5.36 muestra la secuencia desarrollada con el fin de obtener las zonas de proximidad de los Beacons y así poder mostrar en el dispositivo del cliente una alerta cuando este ingrese a dicha zona.

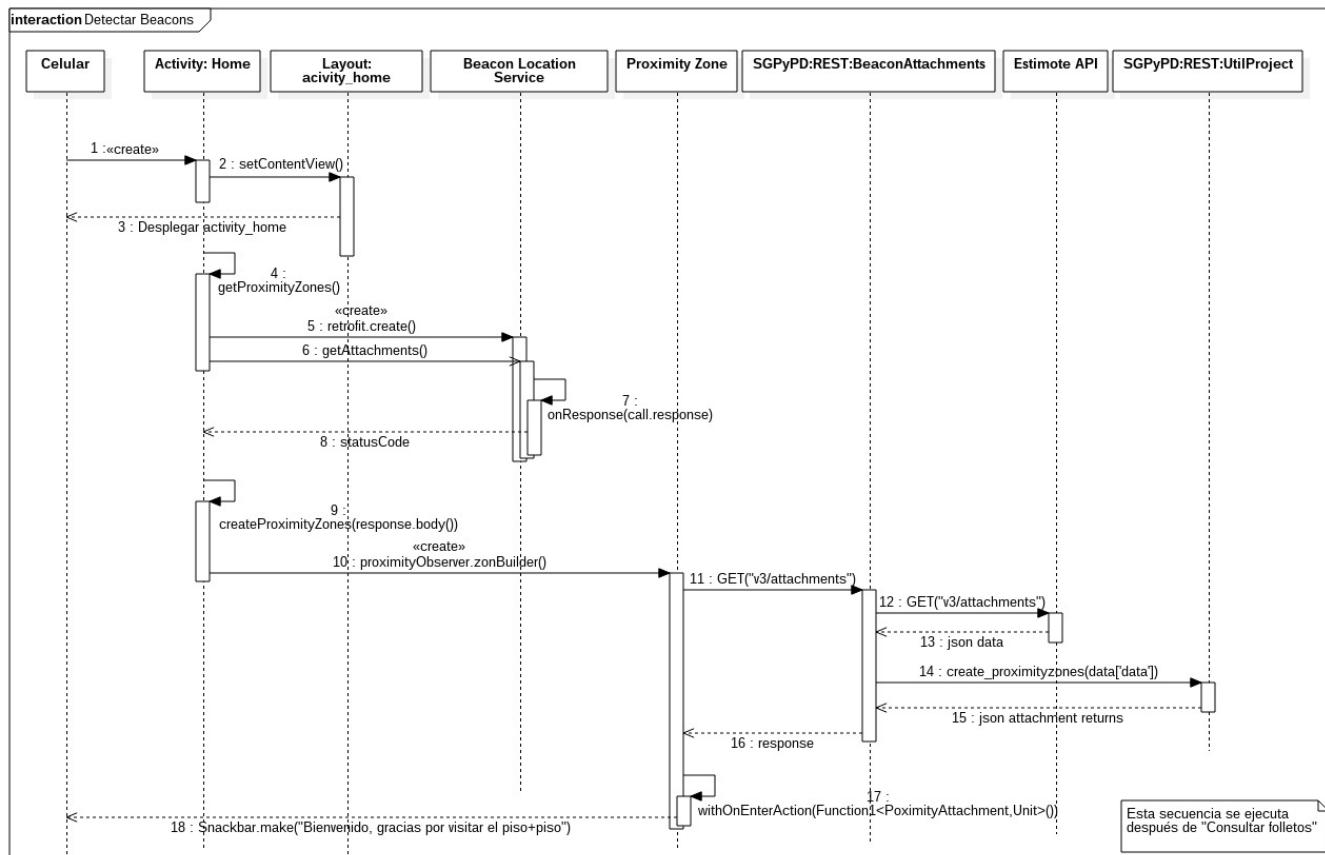


Figura 5.36: Diagrama de secuencia para detectar Beacons (Visualización completa).

El diagrama de la figura 5.37 se dividió en dos secciones con el fin de mostrar una mejor visualización. Dicho diagrama muestra la secuencia desarrollada con el fin de obtener las zonas de proximidad de los Beacons, sin embargo a este se le añade también la funcionalidad de publicar la ubicación del cliente en Kafka.

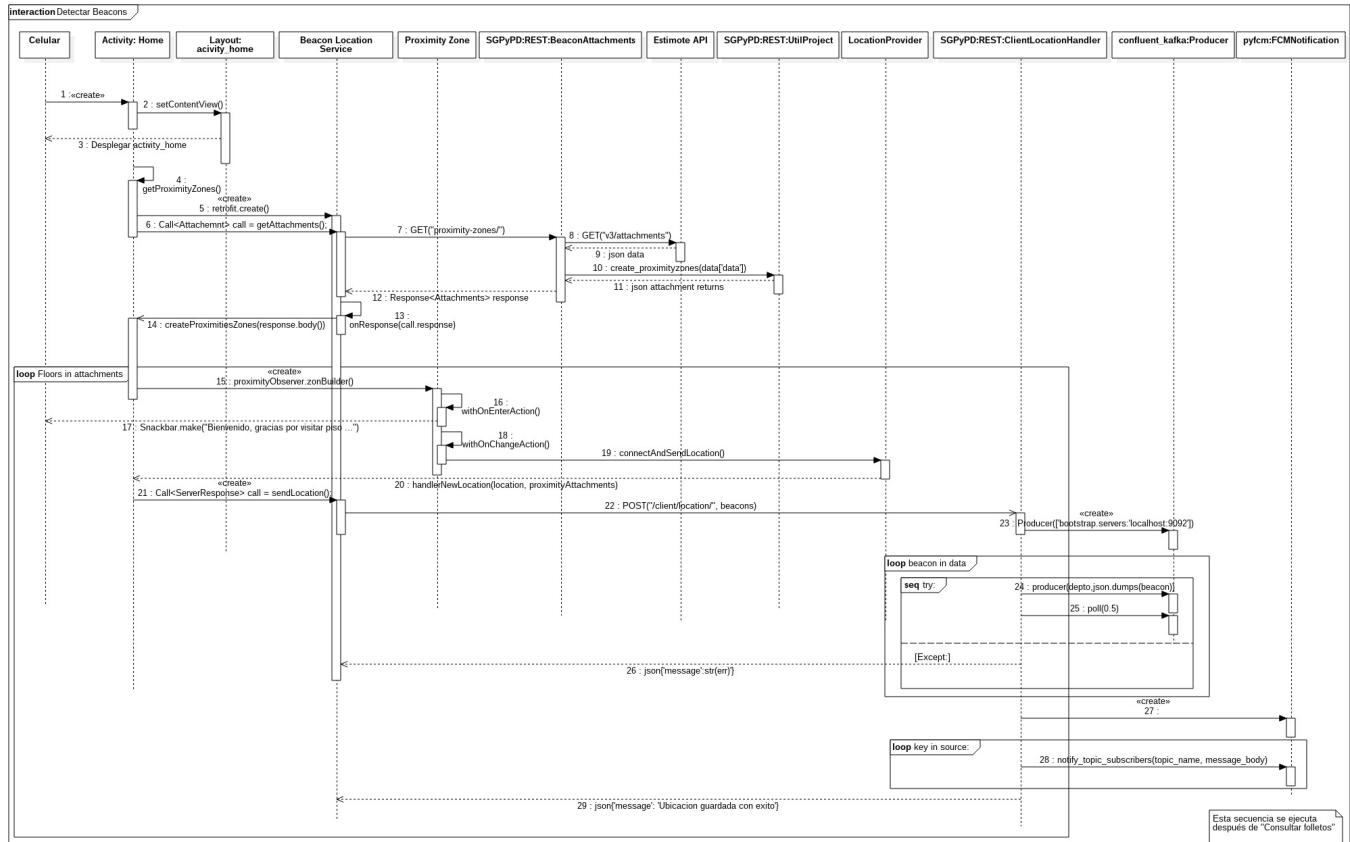


Figura 5.37: Diagrama de secuencia para detectar Beacons (Visualización completa).

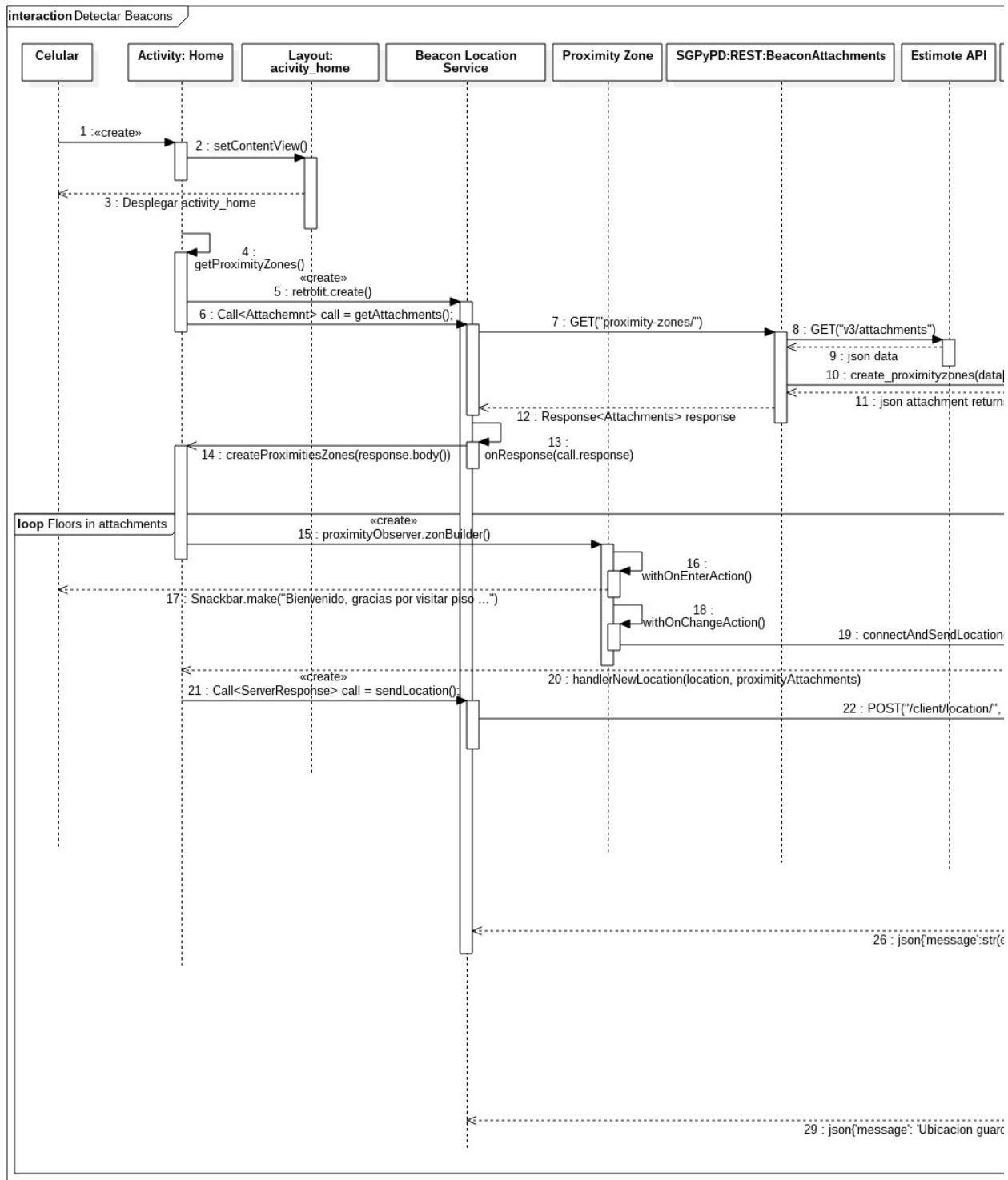


Figura 5.38: Diagrama de secuencia para detectar Beacons (Parte 1).

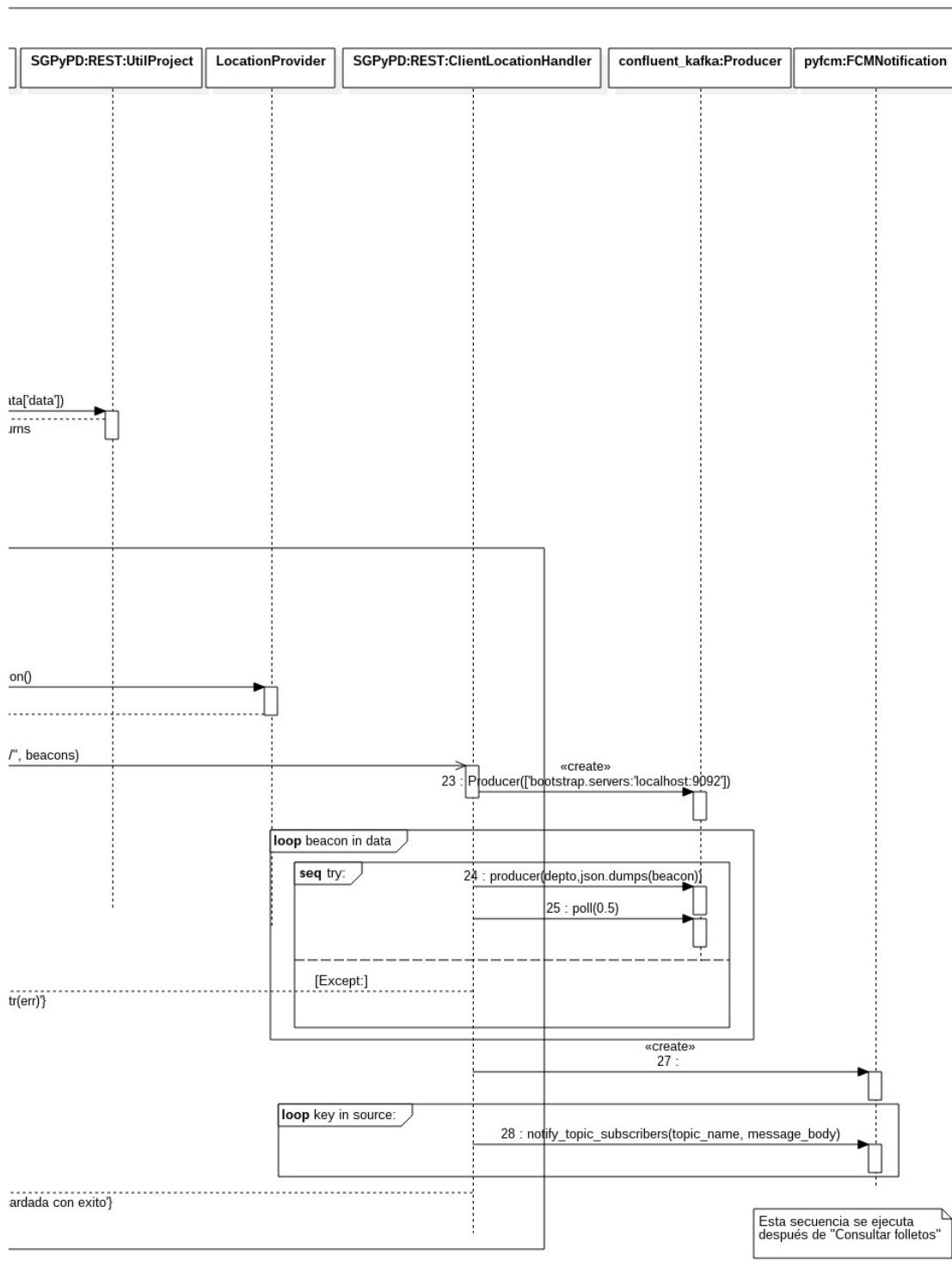


Figura 5.39: Diagrama de secuencia para detectar Beacons (Parte 2).

### 5.1.3. Prototipo 3: Integración de módulos del sistema

#### 5.1.3.1. Análisis

Dentro de este prototipo se realiza la inclusión de todos los servicios propuestos en los requerimientos funcionales definidos previamente en el capítulo del “Bosquejo general de la aplicación” con el título de “[Requerimientos Funcionales de Aplicación Interactiva Difusora de Productos](#)”, sin embargo, se añadieron los requerimientos siguientes:

- Registrar cuenta nueva.
- Recuperar contraseña.
- Eliminar producto de favoritos.
- Mostrar información de productos.

Id	Nombre	Descripción
RFAIDP16	Registrar cuenta nueva	<p>El cliente podrá crear una cuenta nueva a partir del ingreso de datos solicitados en el formulario como:</p> <ul style="list-style-type: none"> <li>• Nombre,</li> <li>• Apellido paterno,</li> <li>• Apellido materno,</li> <li>• Correo electrónico,</li> <li>• Contraseña,</li> <li>• Estado civil,</li> <li>• Fecha de nacimiento.</li> </ul>
RFAIDP17	Recuperar contraseña	En caso de olvidar su contraseña, el cliente podrá solicitar que le sea enviada al correo electrónico que introduzca.
RFAIDP18	Eliminar producto de favoritos	En caso de que el cliente haya añadido un producto a la sección de favoritos y no lo desee más, tendrá la opción de eliminar dicho producto de esta sección al presionar sobre algún producto en particular.
RFAIDP19	Mostrar información de productos	Si el cliente presiona sobre un artículo en particular se desplegará una nueva ventana con la información de dicho producto e imágenes de este.

Cuadro 5.1: Requerimientos añadidos a los Requerimientos Funcionales de la Aplicación Interactiva Difusora de Productos.

La figura [5.40](#) muestra todos los casos de uso de la Aplicación Interactiva Difusora de Productos incluyendo los mencionados anteriormente.

## Casos de uso de AIDP

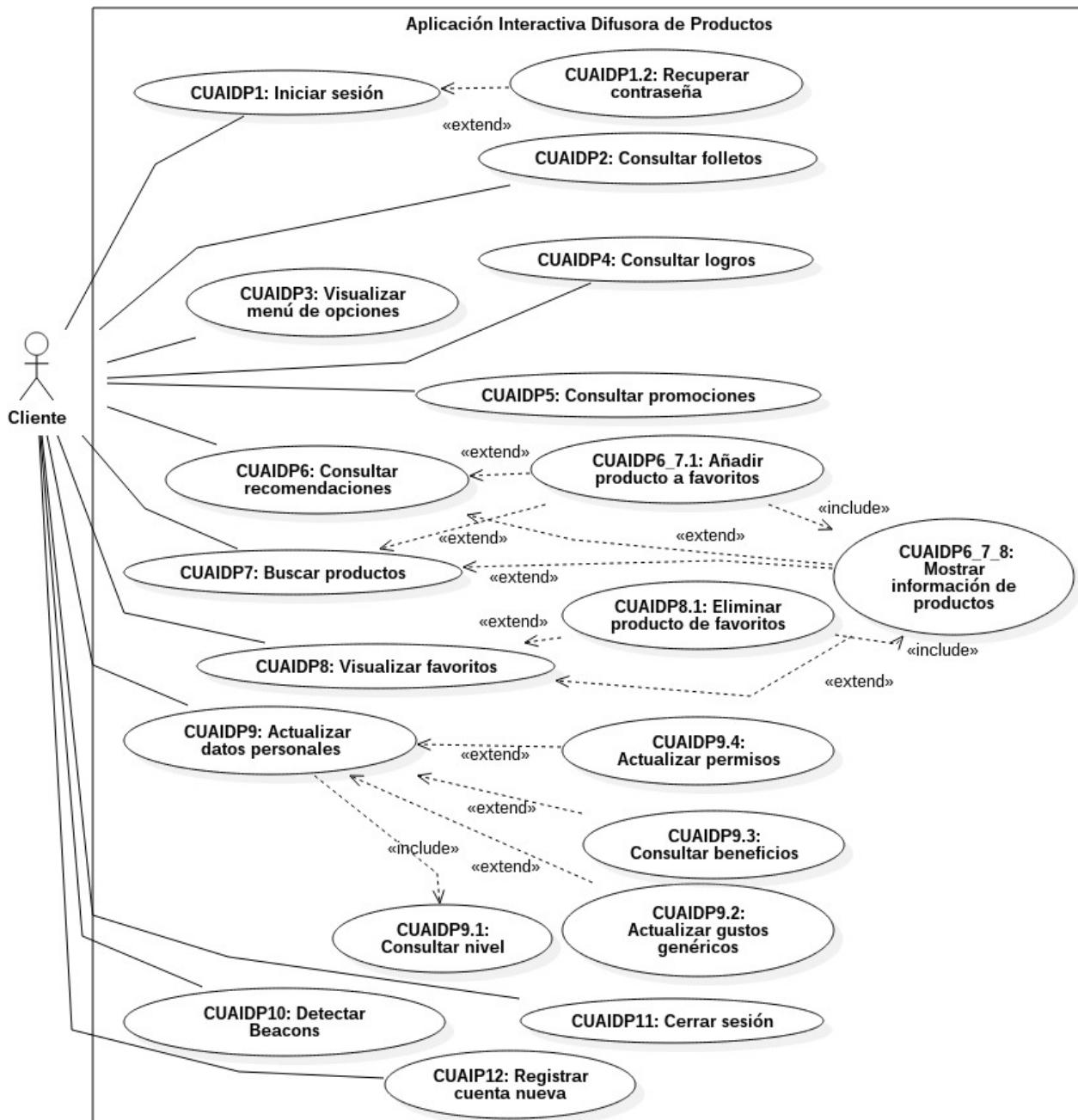


Figura 5.40: Casos de uso de la Aplicación Interactiva Difusora de Productos.

## Diagrama de clases

En esta sección se muestran las clases utilizadas para este prototipo y de las cuales se hacen uso para el la implementación de los diagramas de secuencia.

Las primeras figuras mostradas a continuación, son las clases POJO (por sus siglas en inglés Plain Old Java Object), las cuales en los diagramas posteriores, se podrá visualizar la dependencia que estas tienen con respecto a otras clases, sin embargo debido al tamaño de los diagramas, se optó por mostrar por separado los atributos y métodos de estas clases para una mejor visualización.

*Nota: Es importante mencionar que estas clases al ser POJO son únicamente utilizadas como medio para la obtención y transformación de los datos que son recibidos desde el sistema de gestión, procesamiento y proveedor de datos de Retail, a un formato nativo de Java.*



Figura 5.41: Clases POJO (Parte 1).

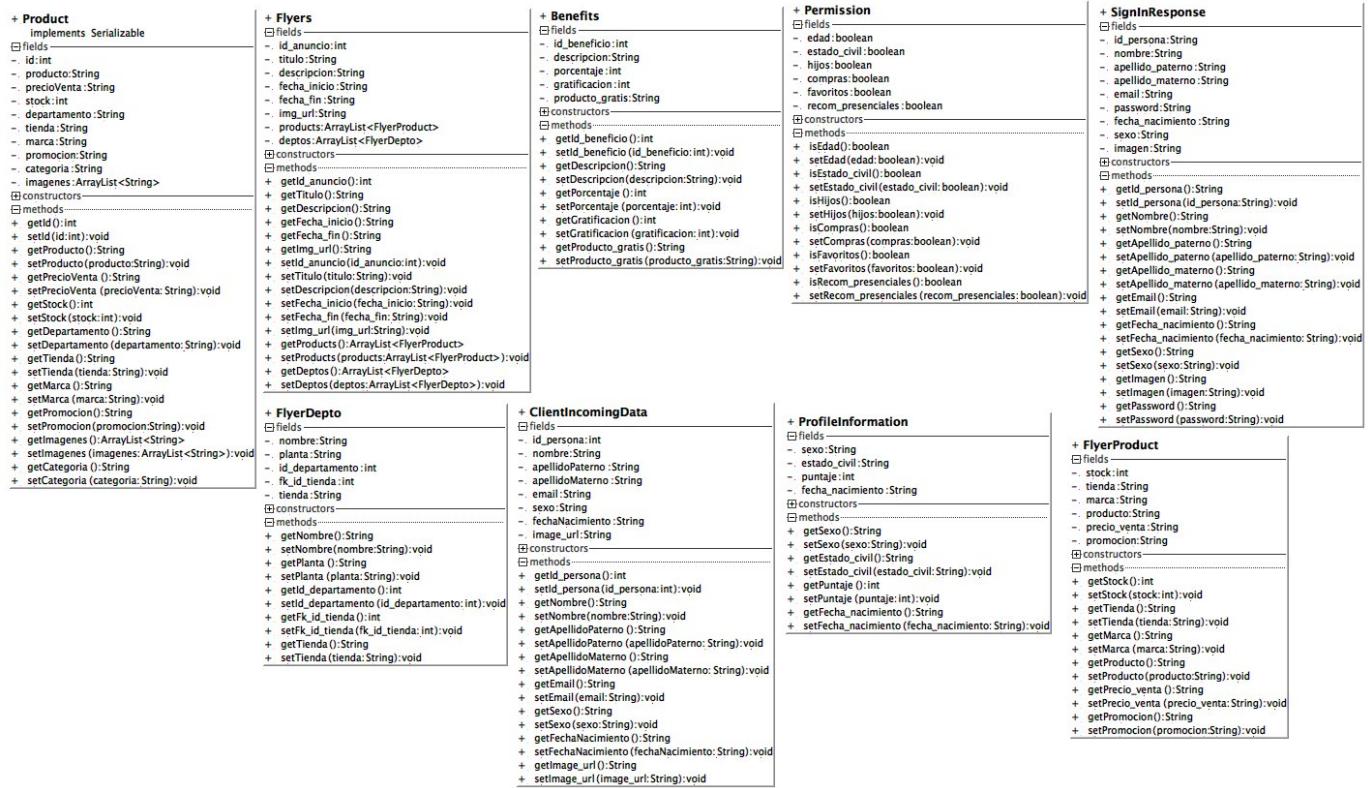


Figura 5.42: Clases POJO (Parte 2).

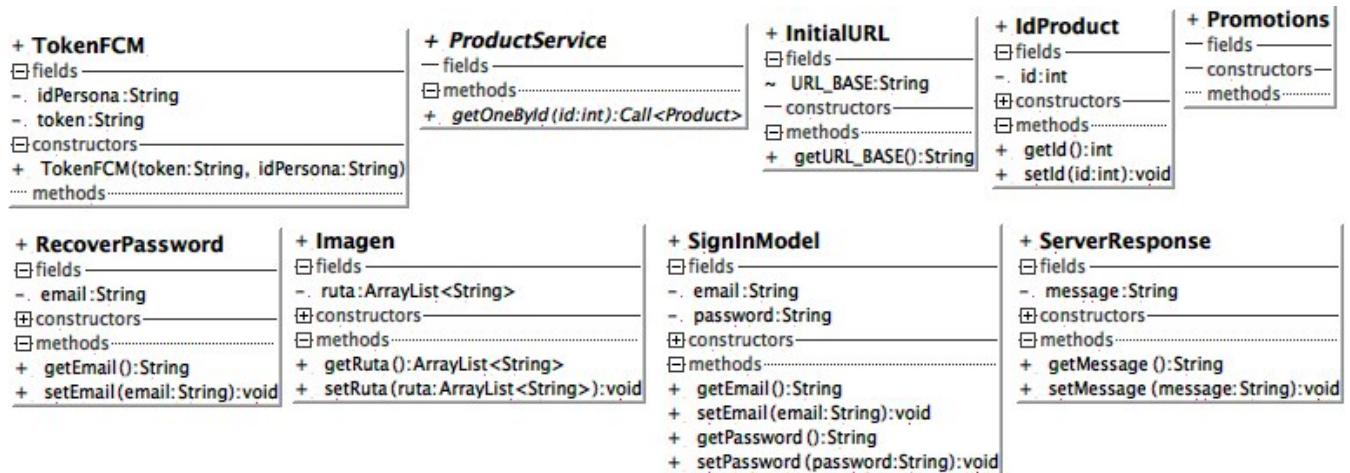


Figura 5.43: Clases POJO (Parte 3).

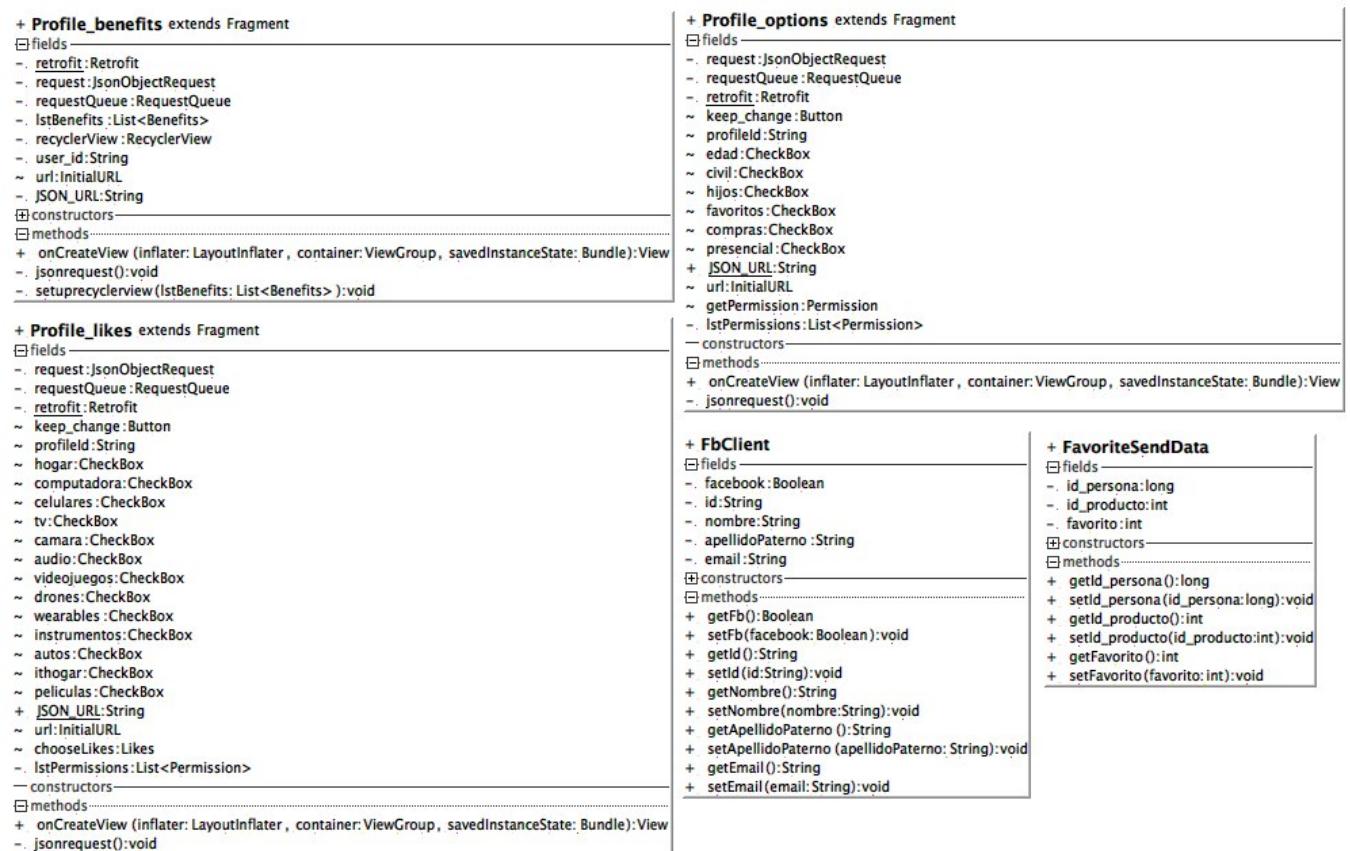


Figura 5.44: Clases POJO (Parte 4).

Los diagramas de la figura 5.46 y 5.47, muestran las clases atributos y métodos de las clases que ya se han plasmado en el diagrama de la figura 5.45, esto se realiza con el fin de una mejor visualización de las clases mencionadas.

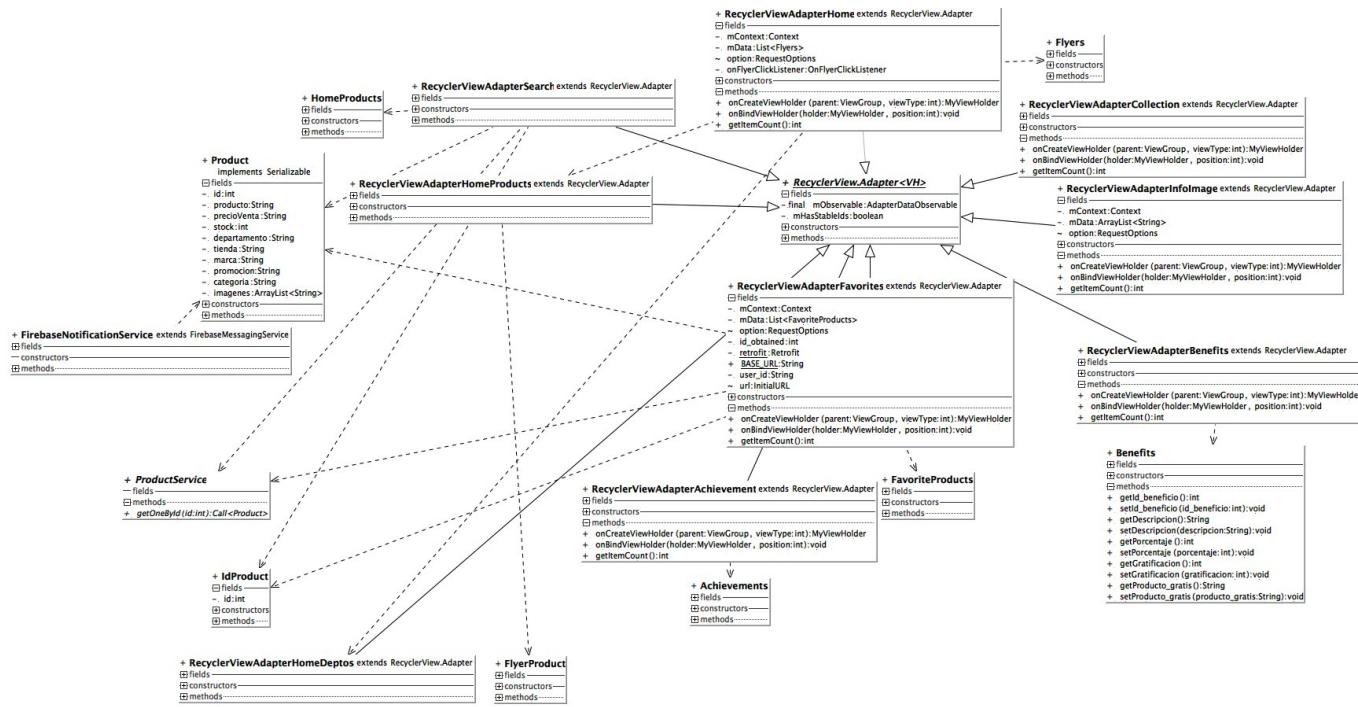


Figura 5.45: Diagrama de clases del prototipo 2 de la AIDP (Visualización completa).

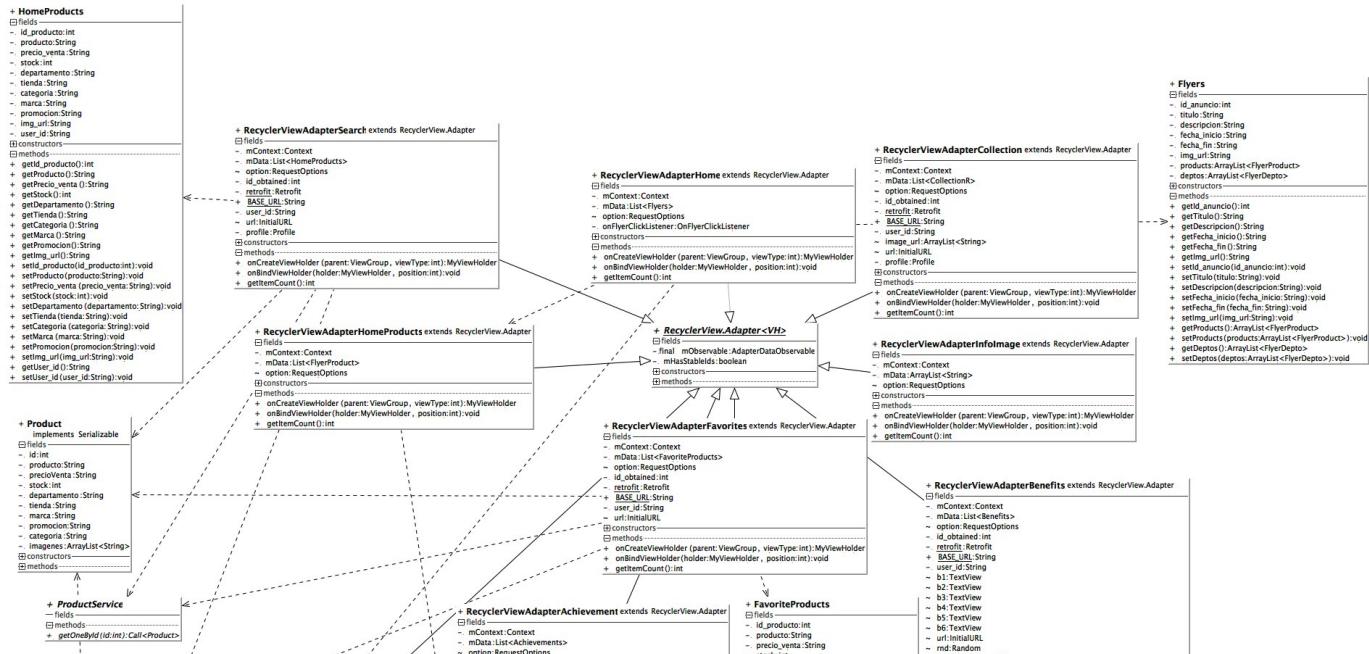


Figura 5.46: Diagrama de clases del prototipo 2 de la AIDP (Parte 1).

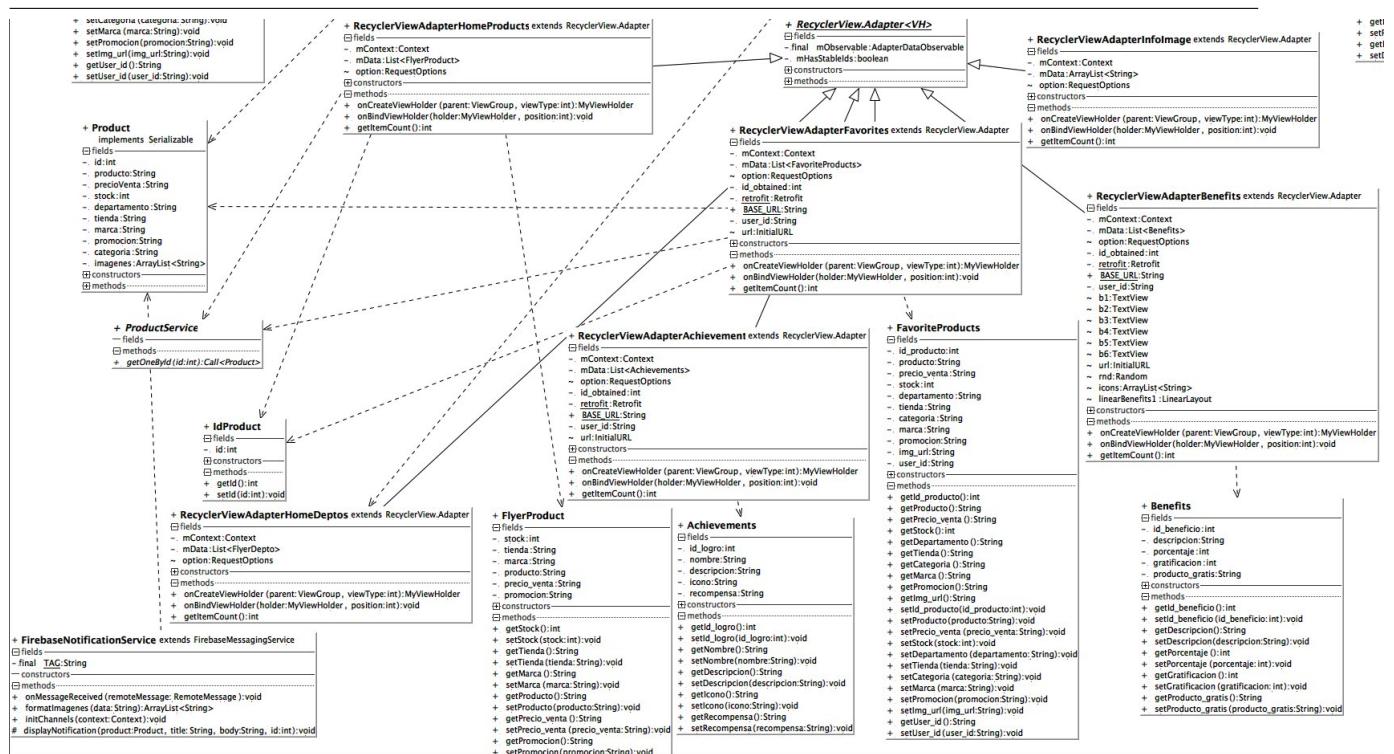


Figura 5.47: Diagrama de clases del prototipo 2 de la AIDP (Parte 2)

La descripción de los elementos de la parte 1 del diagrama de clases es la siguiente:

- **RecyclerView.Adapter**: Clase encargada de transformar los modelos de datos a objetos que entienda el componente visual.

Las clases enlistadas a continuación son como se muestra en el diagrama, una generalización de la clase RecyclerView.Adapter, por lo cual tienen comportamientos similares y son las encargadas de modelar los datos para su correcta visualización, dichos datos son tanto la información de un producto en particular, los productos con promociones, los departamentos con promociones o los logros y beneficios que un usuario puede tener.

- **RecyclerViewAdapterSearch**: Datos de productos
  - **RecyclerViewAdapterHomeProducts**: Datos de productos con promociones.
  - **RecyclerViewAdapterHomeDeptos**: Datos de departamentos con promociones.
  - **RecyclerViewAdapterAchievement**: Datos de logros.
  - **RecyclerViewAdapterAdapterHome**: Datos de promociones en general.
  - **RecyclerViewAdapterCollection**: Datos de productos recomendados.
  - **RecyclerViewAdapterInfoImage**: Imágenes de productos.
  - **RecyclerViewAdapterBenefits**: Datos de beneficios.
  - **RecyclerViewAdapterFavorites**: Datos de productos almacenados como favoritos.

El diagrama de la figura 5.48 muestra la parte 2 del diagrama de clases.

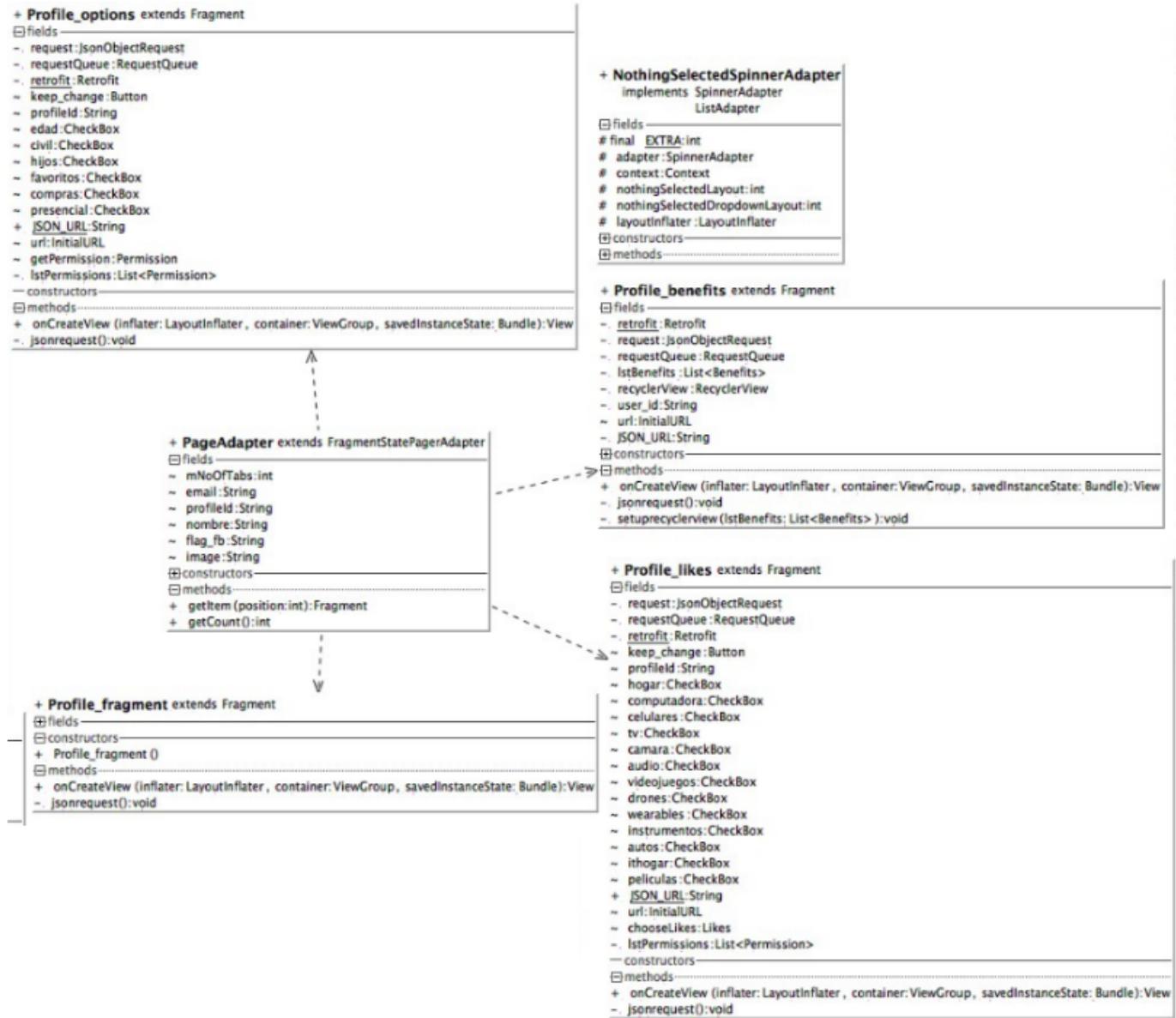


Figura 5.48: Diagrama de clases del prototipo 2 de la AIDP (Visualización completa).

La descripción de los elementos de la parte 2 del diagrama de clases es la siguiente:

- **PageAdapter:** Clase que nos permite mostrar distintas pantallas desplazando la página actual.
- **Profile\_options:** Clase encargada de recibir desde el sistema de gestión, procesamiento y proveedor de datos de Retail, los permisos que el usuario ha seleccionado y guardado para posteriormente desplegarlos.
- **Profile\_benefits:** Clase encargada de recibir los beneficios que el usuario obtiene con respecto al nivel en el que se encuentre desde el sistema de gestión, procesamiento y proveedor de datos de Retail para posteriormente desplegarlos.

- **Profile\_likes**: Clase encargada de recibir desde el sistema de gestión, procesamiento y proveedor de datos de Retail, los gustos que el usuario ha seleccionado y guardado para posteriormente desplegarlos.
  - **Profile\_fragment**: Clase encargada de recibir la información personal de cada usuario desde el sistema de gestión, procesamiento y proveedor de datos de Retail.

Los diagramas de las figuras 5.49 y 5.50 muestran la tercera y última parte del diagrama de clases.

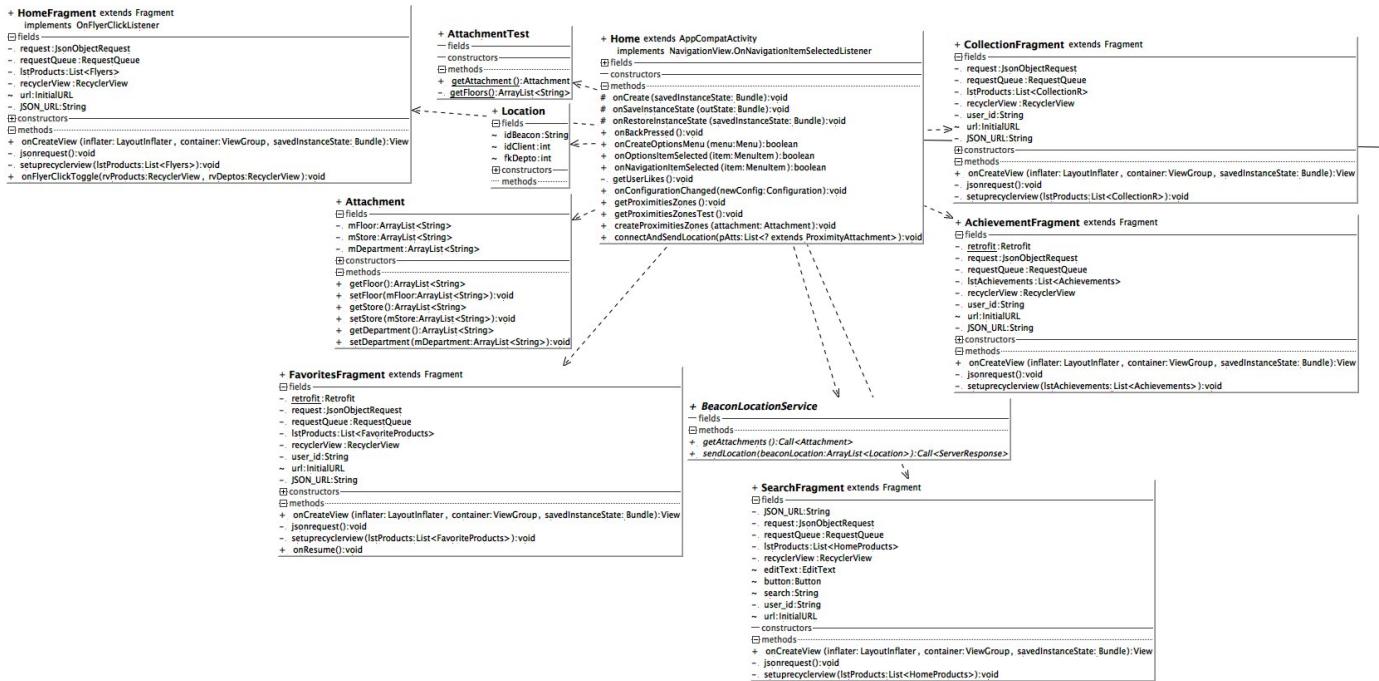


Figura 5.49: Diagrama de clases del prototipo 2 de la AIDP (Visualización completa).

La descripción de los elementos de la primera parte de la parte 3 del diagrama de clases es la siguiente:

- **Home:** Clase que nos permite mostrar distintas pantallas desplazando la página actual.

Las clases enlistadas a continuación muestran una dependencia de la clase Home, debido a que en Home se realiza el funcionamiento del menú y este a su vez, proporciona la funcionalidad para llamar a las distintas opciones de dicho menú. Cada una de esas opciones son:

- **HomeFragment:** Clase encargada de obtener desde el sistema de gestión, procesamiento y proveedor de datos de Retail, los folletos promocionales y productos o departamentos que igualmente cuentan con alguna promoción.
  - **CollectionFragment:** Clase encargada de obtener desde el sistema de gestión, procesamiento y proveedor de datos de Retail, los productos recomendados para cada uno de los usuarios.
  - **SearchFragment:** Clase encargada de obtener desde el sistema de gestión, procesamiento y proveedor de datos de Retail, los diferentes productos que pueden existir respecto a la búsqueda de un cliente.
  - **FavoritesFragment:** Clase encargada de obtener desde el sistema de gestión, procesamiento y proveedor de datos de Retail, los productos que el cliente ha almacenado en su sección de favoritos.
  - **AchievementFragment:** Clase encargada de obtener desde el sistema de gestión, procesamiento y proveedor de datos de Retail, los logros que un cliente ha obtenido debido a las diferentes compras

realizadas.

- **Attachment:** Clase que retrofit mapea, es la respuesta de una petición HTTP GET en formato JSON, contiene arreglos de String para generar zonas de proximidad por pisos, tiendas o departamentos.
- **Location:** Clase que nos permite mostrar distintas pantallas desplazando la página actual.
- **AttachmentTest:** Clase únicamente con fines de prueba para no obtener los attachments desde el servidor.
- **BeaconLocationService:** Interfaz que define el método saveLocation para guardar la ubicación actual de Beacons y el método getAttachments que obtiene las claves - valor necesarias para generar las zonas de proximidad, esta interfaz utiliza retrofit que nos permite consumir la API REST en la aplicación.

Los diagramas de las figuras 5.51 y 5.52 muestran las desplegadas que previamente ya se muestran en la figura 5.50, esto con el fin de una mejor visualización de los atributos y métodos de cada una de ellas.

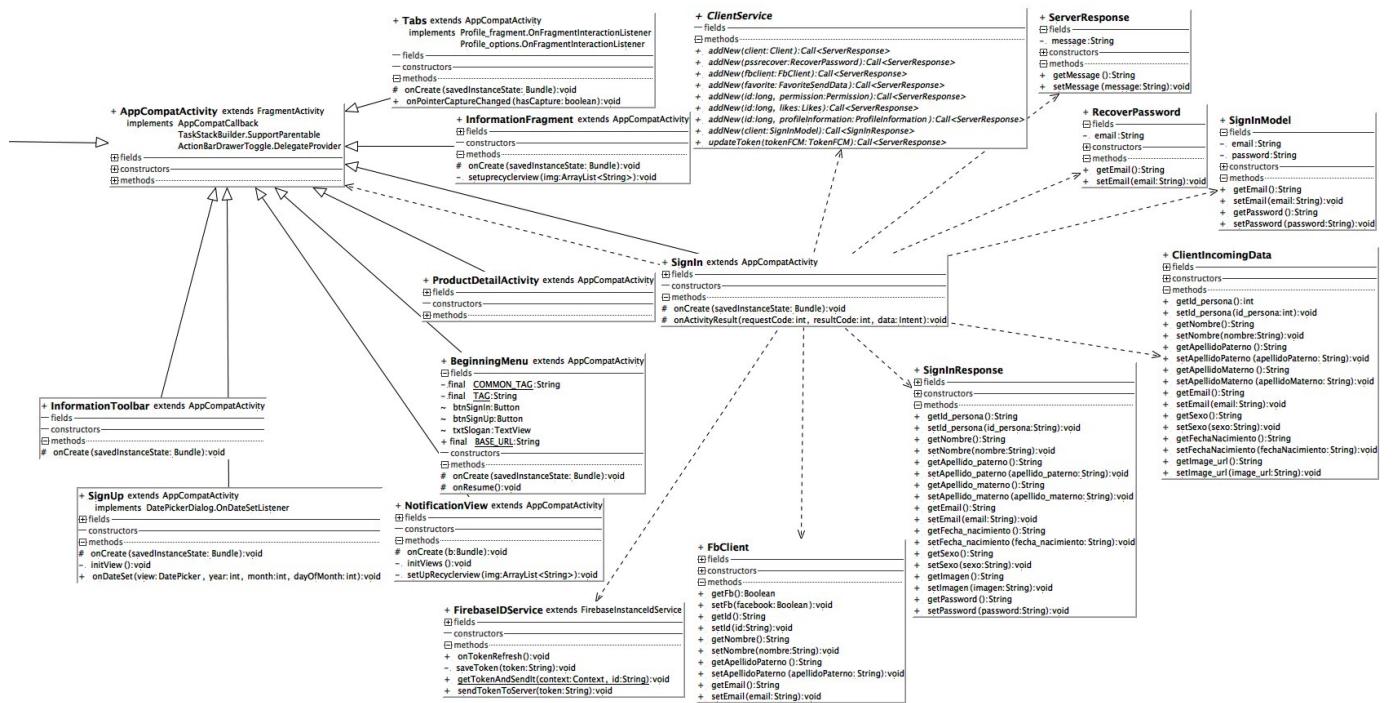


Figura 5.50: Diagrama de clases del prototipo 2 de la AIDP (Visualización completa).

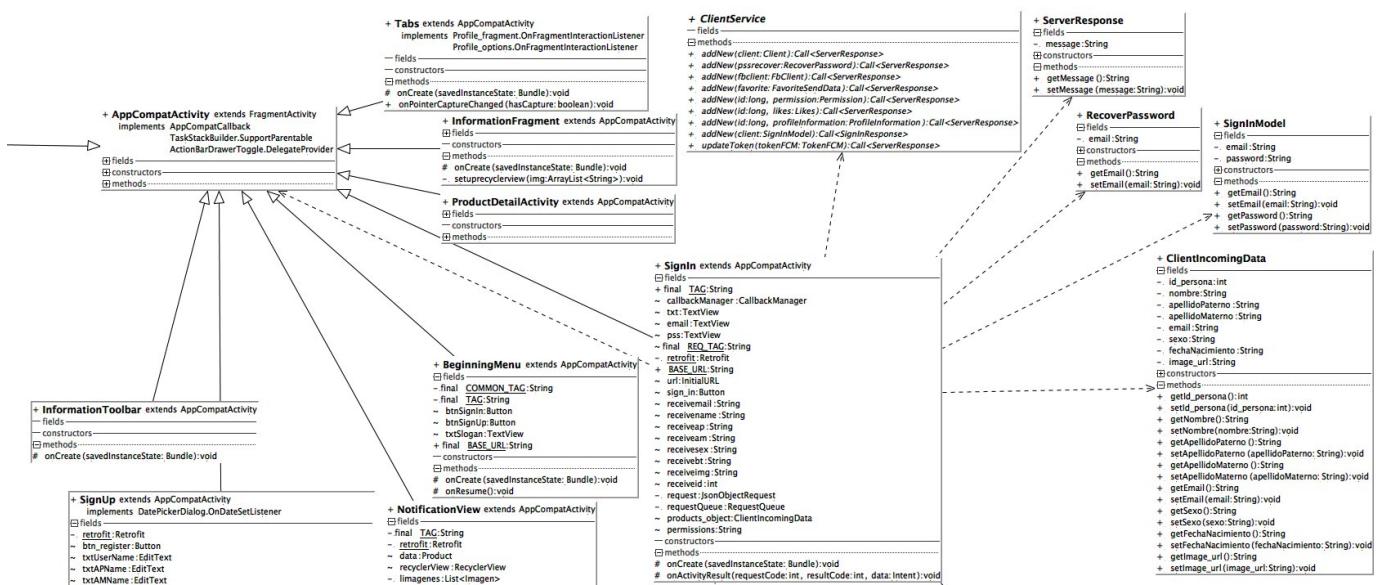


Figura 5.51: Diagrama de clases del prototipo 2 de la AIDP (Parte 1).

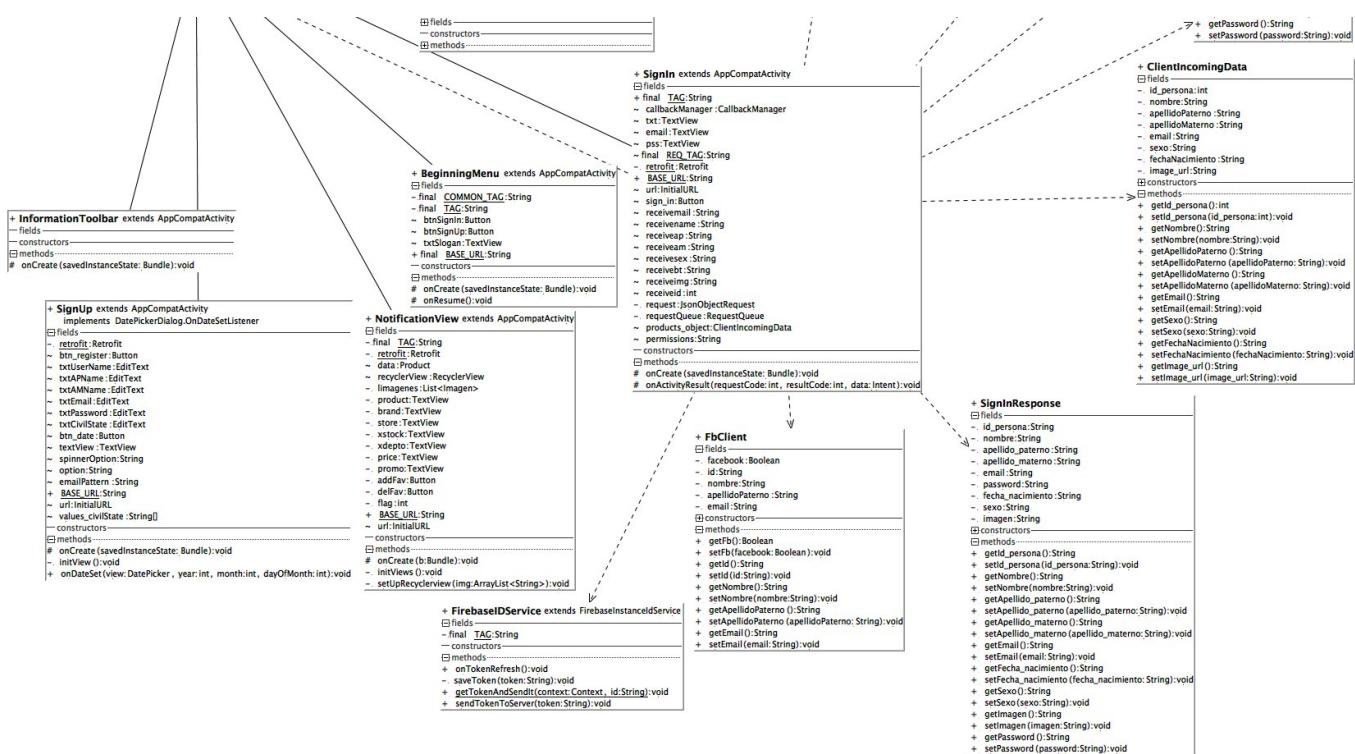


Figura 5.52: Diagrama de clases del prototipo 2 de la AIDP (Parte 2).

La descripción de los elementos de la segunda parte de la parte 3 del diagrama de clases es la siguiente:

- **AppCompatActivity**: Clase base para actividades que utilizan las características de la barra de acciones de la biblioteca de soporte de Android.
  - **SignUp**: Clase encargada de mostrar al usuario un pequeño formulario con el fin de que este registre

---

una cuenta nueva en la aplicación.

- **SignIn:** Clase encargada del inicio de sesión de un usuario, en dicha clase, tanto si el inicio de sesión es con una cuenta de la aplicación como si es con Facebook, se envían los datos al servidor mediante el uso de la interfaz ClientService.
- **BegginingMenu:** Es la clase que muestra la pantalla inicial de la aplicación que únicamente es informativa.
- **NotificationView:** Clase en donde se muestra la información de un producto que se ha recomendado y mismo que ha sido enviado al cliente mediante una notificación.
- **ProductDetailActivity:** Clase encargada de mostrar la información de un producto en particular.
- **Tabs:** Clase encargada de manejar las opciones que se muestran en un menú que hace uso de las clases explicadas en la parte 2 del diagrama de clases.
- **InformationFragment:** Clase encargada de recibir la información de un producto en particular incluyendo a su vez la obtención de un arreglo de imágenes respectivas a dicho producto.
- **FirebaseIDService:** Clase que extiende de FirebaseInstanceIdService y funciona para obtener el token actual de Firebase Cloud Messaging para el dispositivo.
- **ClientService:** Interfaz utilizada para hacer peticiones POST al servidor y comprobar que los datos sean correctos para un inicio de sesión exitoso o para obtener la información de gustos, permisos e información de perfil de un usuario, entre otros.

#### 5.1.3.2. Diseño

En esta sección se muestra el diseño de las pantallas de los requerimientos que fueron agregados para este prototipo mismos que se localizan en la parte superior en la sección de [Análisis](#).

## UIAIDP15 - Registrar cuenta nueva

### Objetivo

Registra una cuenta nueva para el caso en el que el usuario no desee ingresar con su cuenta de Facebook.

### Diseño

Esta pantalla (figura 5.53), nos muestra los datos requeridos para que el usuario pueda generar una cuenta nueva.



Figura 5.53: UIAIDP15 Registrar cuenta nueva.

### Salidas

Ninguna.

### Entradas

Nombre, apellido paterno, apellido materno, correo electrónico, contraseña, estado civil y fecha de nacimiento.

## UIAIDP16 - Recuperar contraseña

### Objetivo

En caso de que el usuario haya olvidado su contraseña, mediante un correo electrónico recupera esta.

### Diseño

Esta pantalla (figura 5.54), nos muestra una ventana emergente que se despliega con el fin de solicitar un correo electrónico al usuario y así enviarle esta.



Figura 5.54: UIAIDP16 Recuperar contraseña.

### Salidas

Ninguna.

### Entradas

Correo electrónico.

## UIAIDP17 - Muestra información de productos

### Objetivo

Proporciona la información del producto que el usuario ha seleccionado.

### Diseño

Esta pantalla (figura 5.55), despliega la información del producto que ha sido seleccionado por el usuario y en caso de que este contenga más imágenes aparte de la principal, las despliega. De igual manera muestra el botón para añadir un producto a la sección de Favoritos.



Figura 5.55: UIAIDP17 Muestra información de productos.

### Salidas

Alerta de notificación de producto añadido a la sección de Favoritos.

### Entradas

Ninguna.

## UIAIDP18 - Eliminar producto de favoritos

### Objetivo

Despliega un botón que permite la eliminación de un producto de la sección de Favoritos.

### Diseño

Esta pantalla (figura 5.56), despliega la información del producto y de igual manera muestra un botón el cual al presionarlo elimina el producto seleccionado de la sección de Favoritos.



Figura 5.56: UIAIDP18 Eliminar producto de favoritos.

### Salidas

Alerta de notificación de eliminación del producto de la sección de Favoritos.

### Entradas

Ninguna.

A continuación, se muestran los diagramas de secuencia que fueron elaborados a partir de cada uno de los módulos del sistema que fueron siendo añadidos a la AIDP.

### Diagramas de secuencia

#### Iniciar Sesión

El diagrama de la figura 5.57 se dividió en dos secciones para obtener una mejor visualización, dichas secciones corresponden a la figura 5.58 y 5.59. En la figura 5.57 se muestra el diagrama de secuencia para iniciar sesión en la aplicación móvil, en el cual se observan las 3 diferentes alternativas mediante con las que el cliente puede iniciar sesión y acceder a la aplicación.

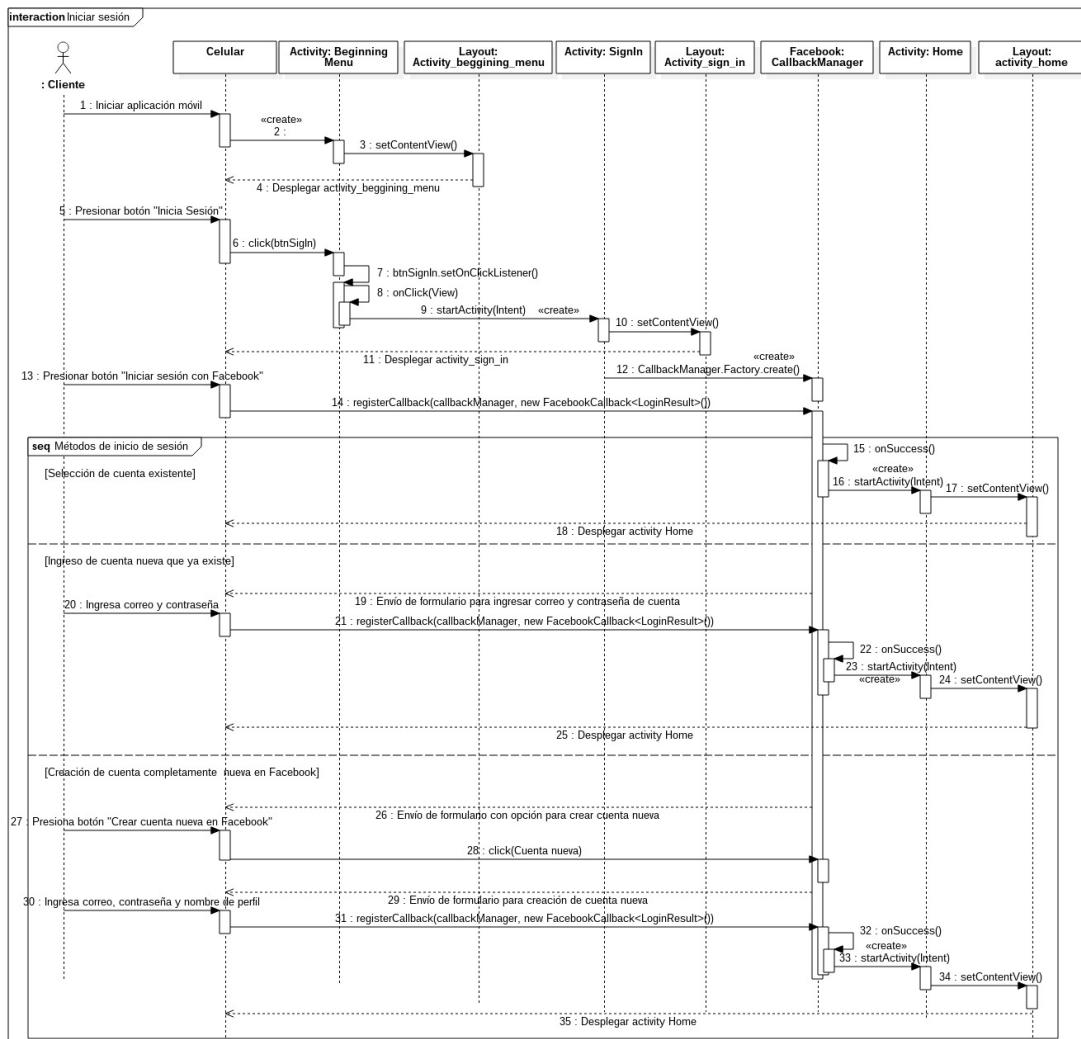


Figura 5.57: Diagrama de secuencia para iniciar sesión (Visualización completa).

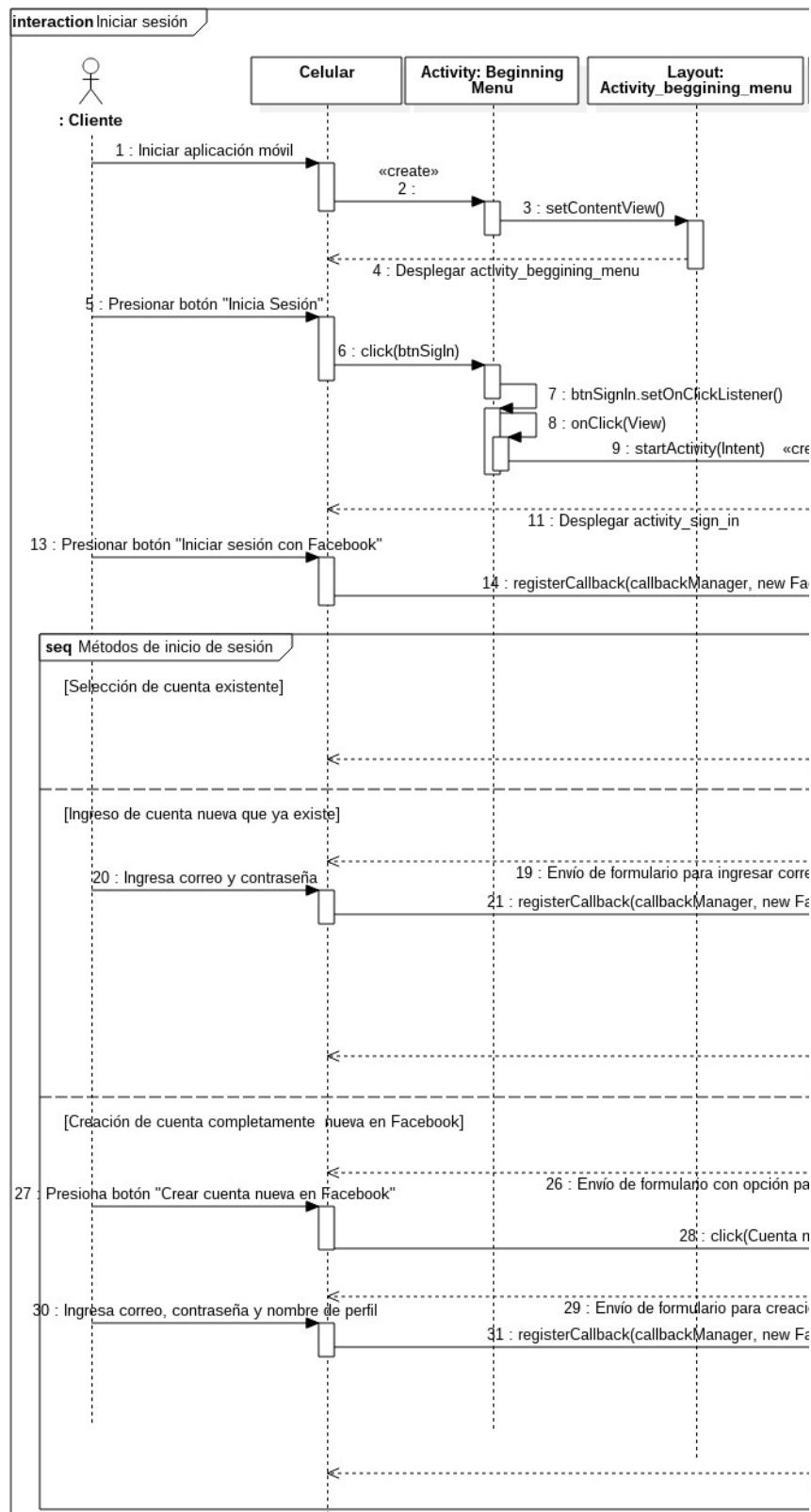


Figura 5.58: Diagrama de secuencia para iniciar sesión (Parte 1).

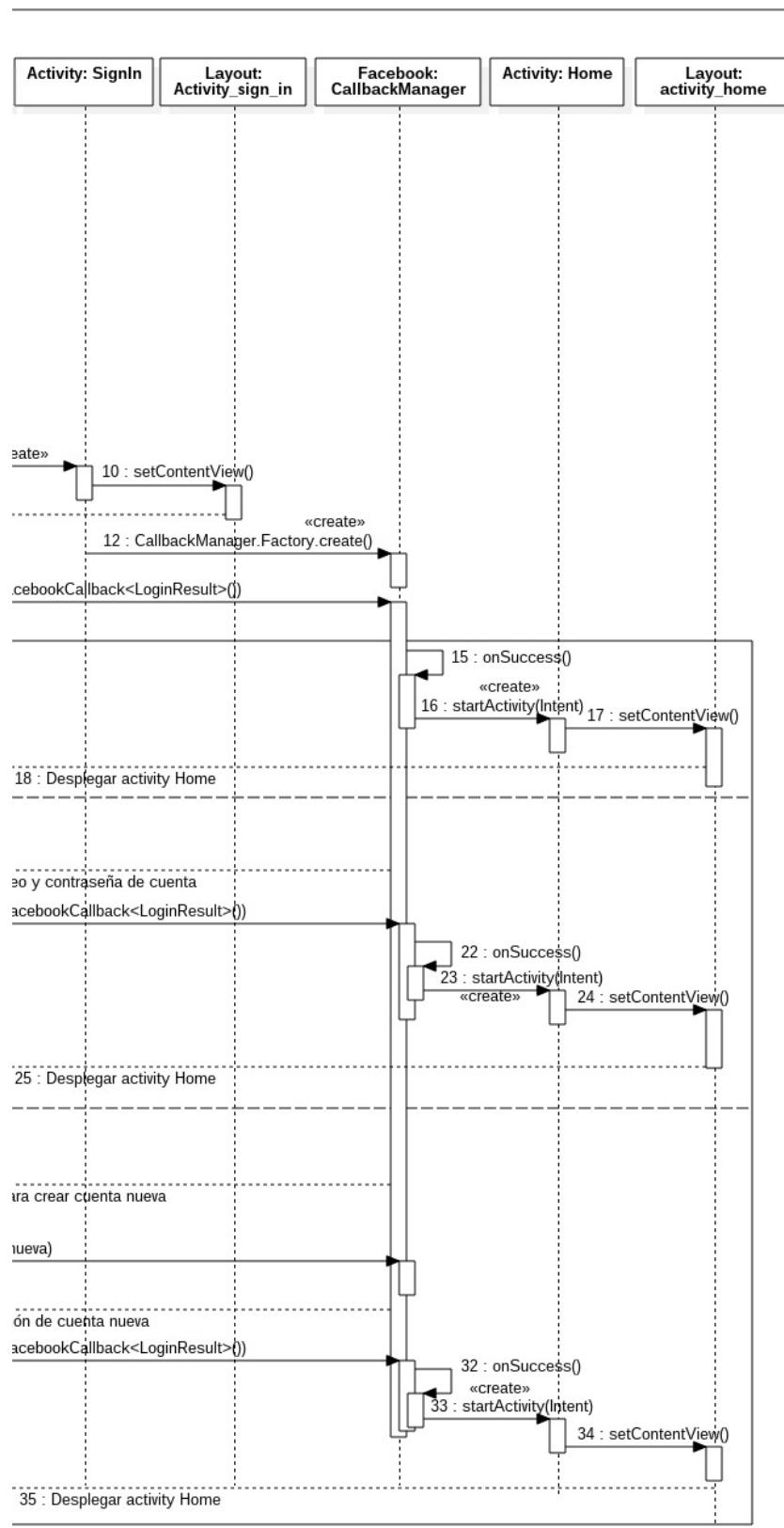


Figura 5.59: Diagrama de secuencia para iniciar sesión (Parte 2).

### Recuperar contraseña

La figura 5.60 se dividió en dos secciones para obtener una mejor visualización, dichas secciones corresponden a la figura 5.58 y 5.59. En la figura 5.60 se muestra el diagrama de secuencia para recuperar la contraseña de un usuario en caso de que este no se haya registrado con Facebook en la aplicación móvil.

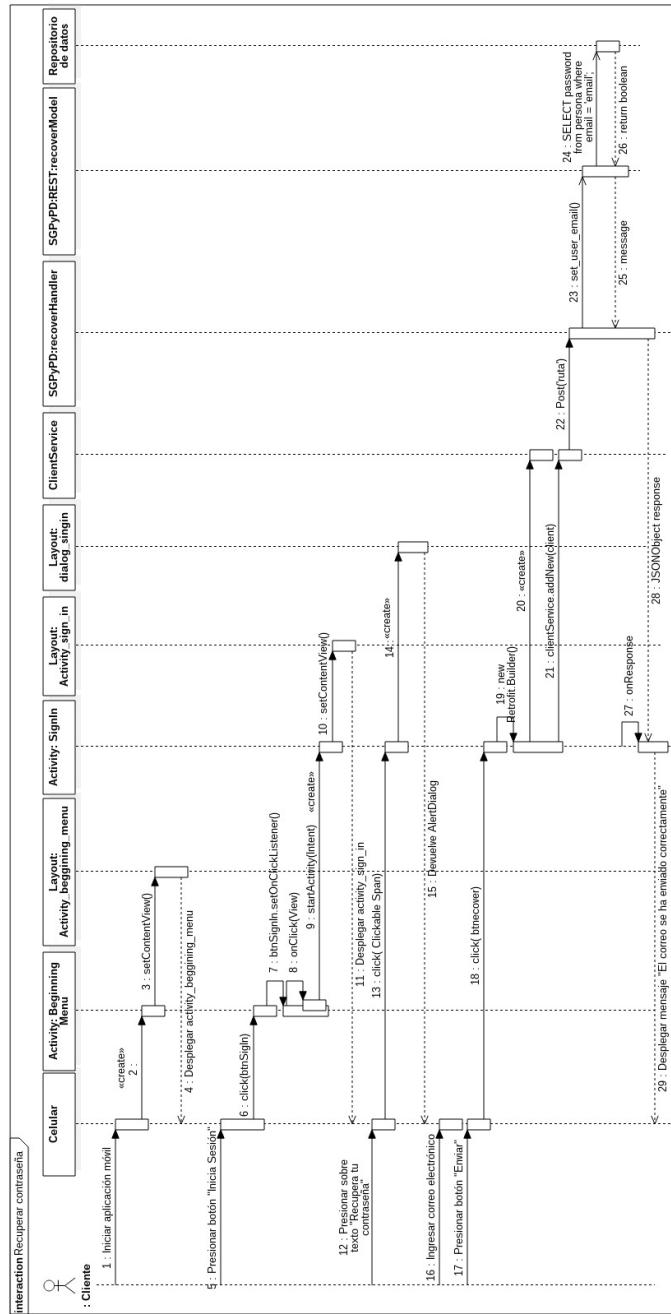


Figura 5.60: Diagrama de secuencia para recuperar contraseña (Visualización completa).

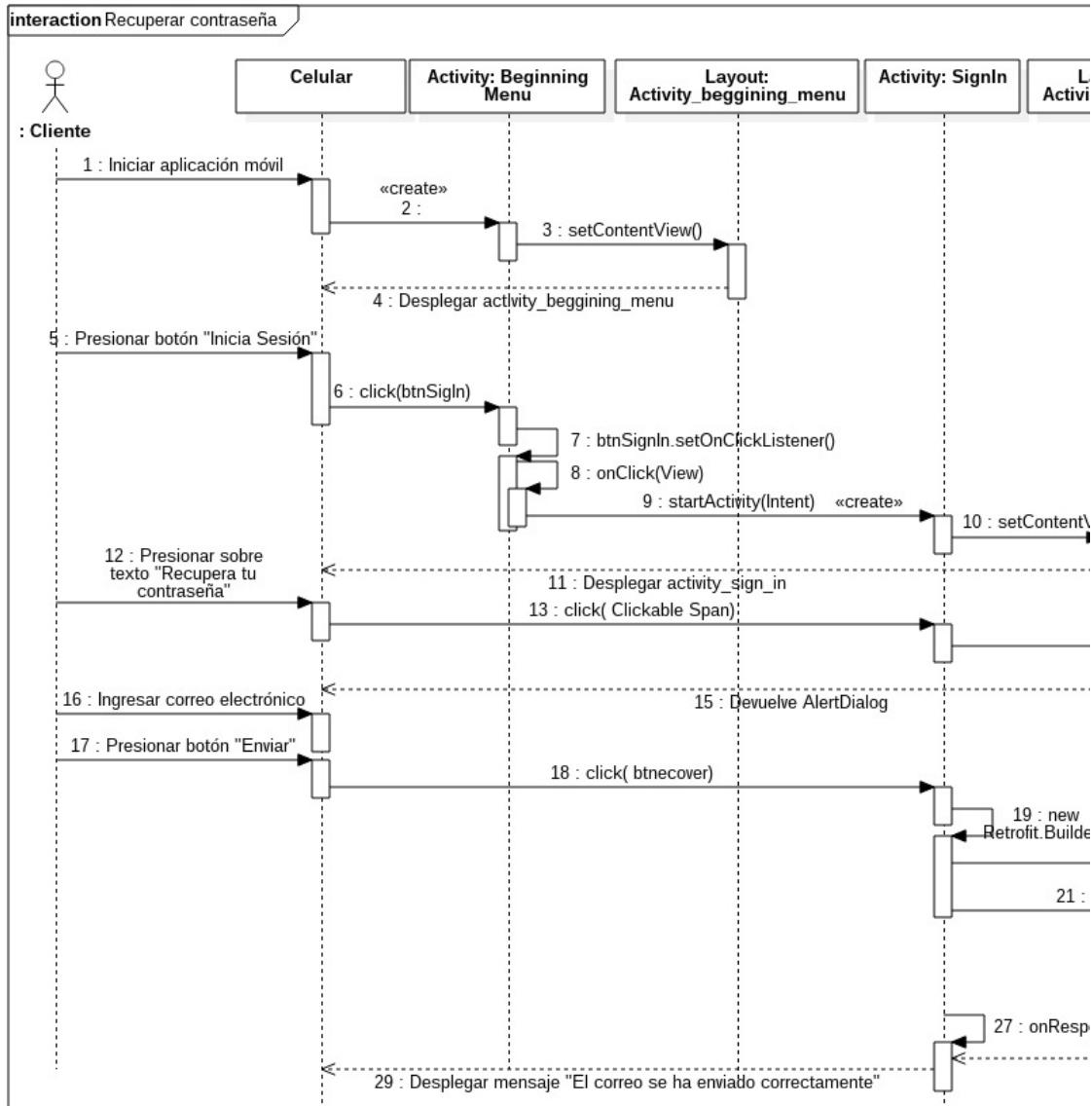


Figura 5.61: Diagrama de secuencia para recuperar contraseña (Parte 1).

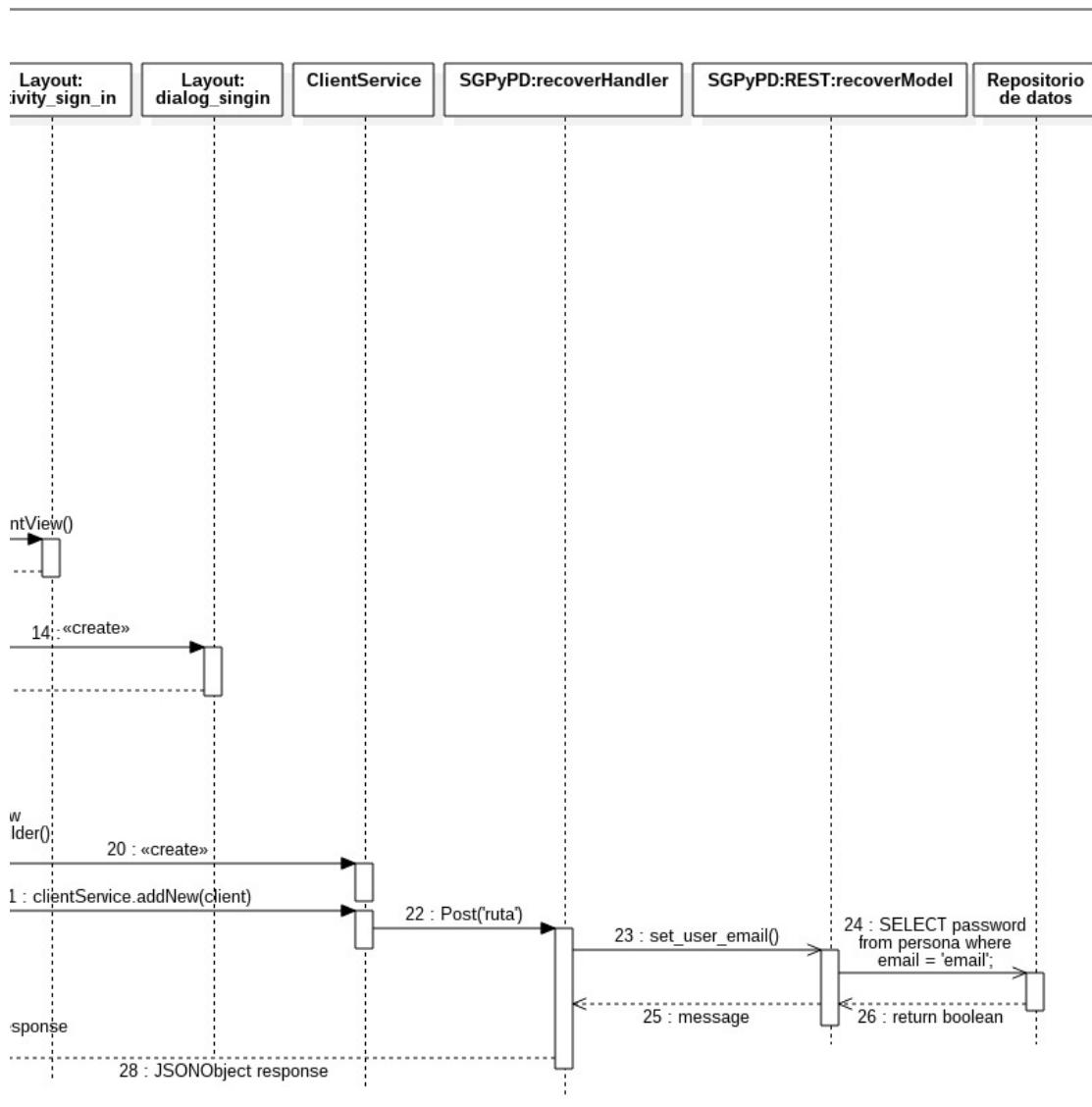


Figura 5.62: Diagrama de secuencia para recuperar contraseña (Parte 2).

### Registrar cuenta nueva

El diagrama de la figura 5.63 se dividió en dos secciones para obtener una mejor visualización, dichas secciones corresponden a la figura 5.64 y 5.65. Este diagrama corresponde a la secuencia “Registrar cuenta nueva” mediante la cual, el usuario puede crear una cuenta en caso de que no desee iniciar sesión con Facebook.

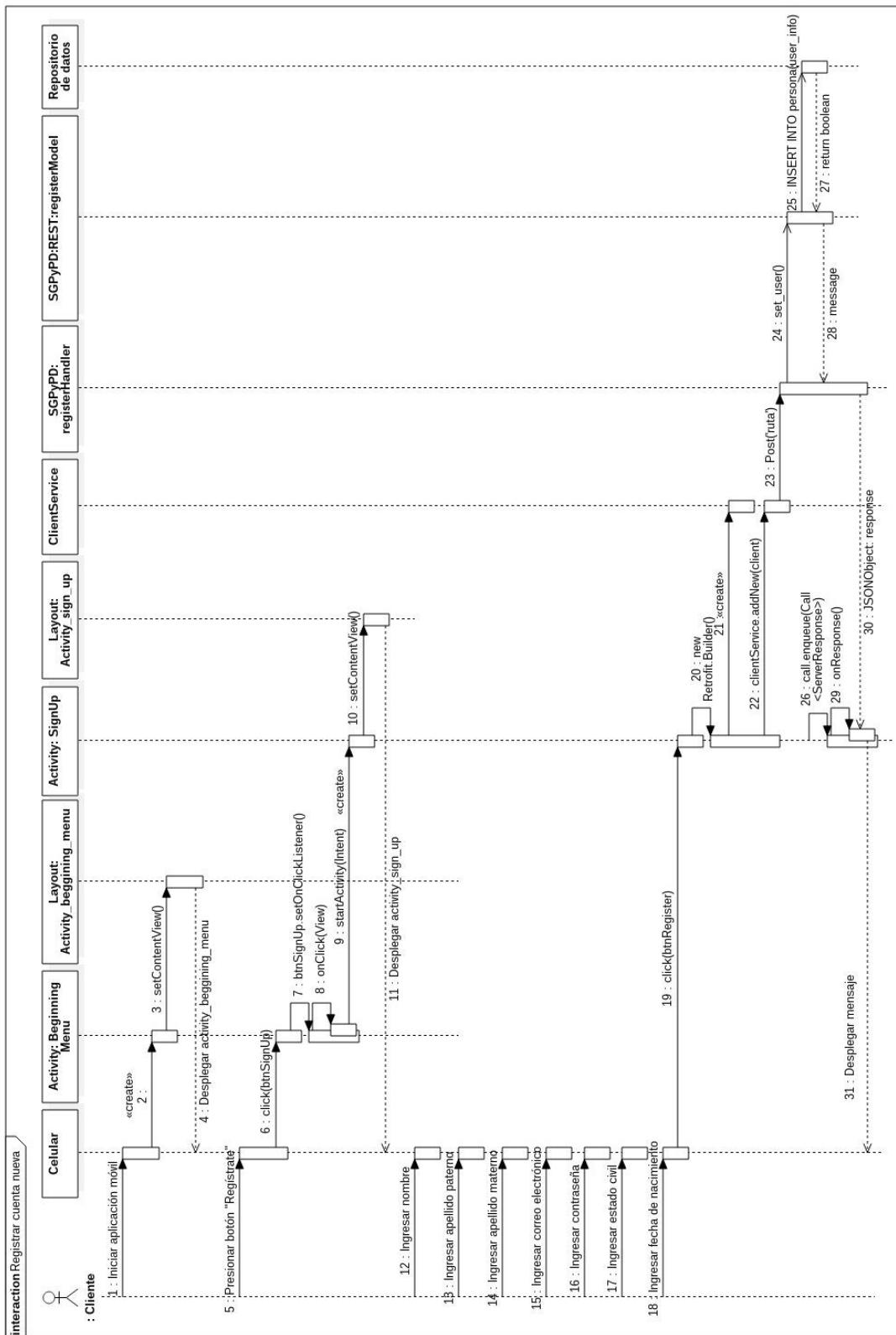


Figura 5.63: Diagrama de secuencia para registrar cuenta nueva (Visualización completa).

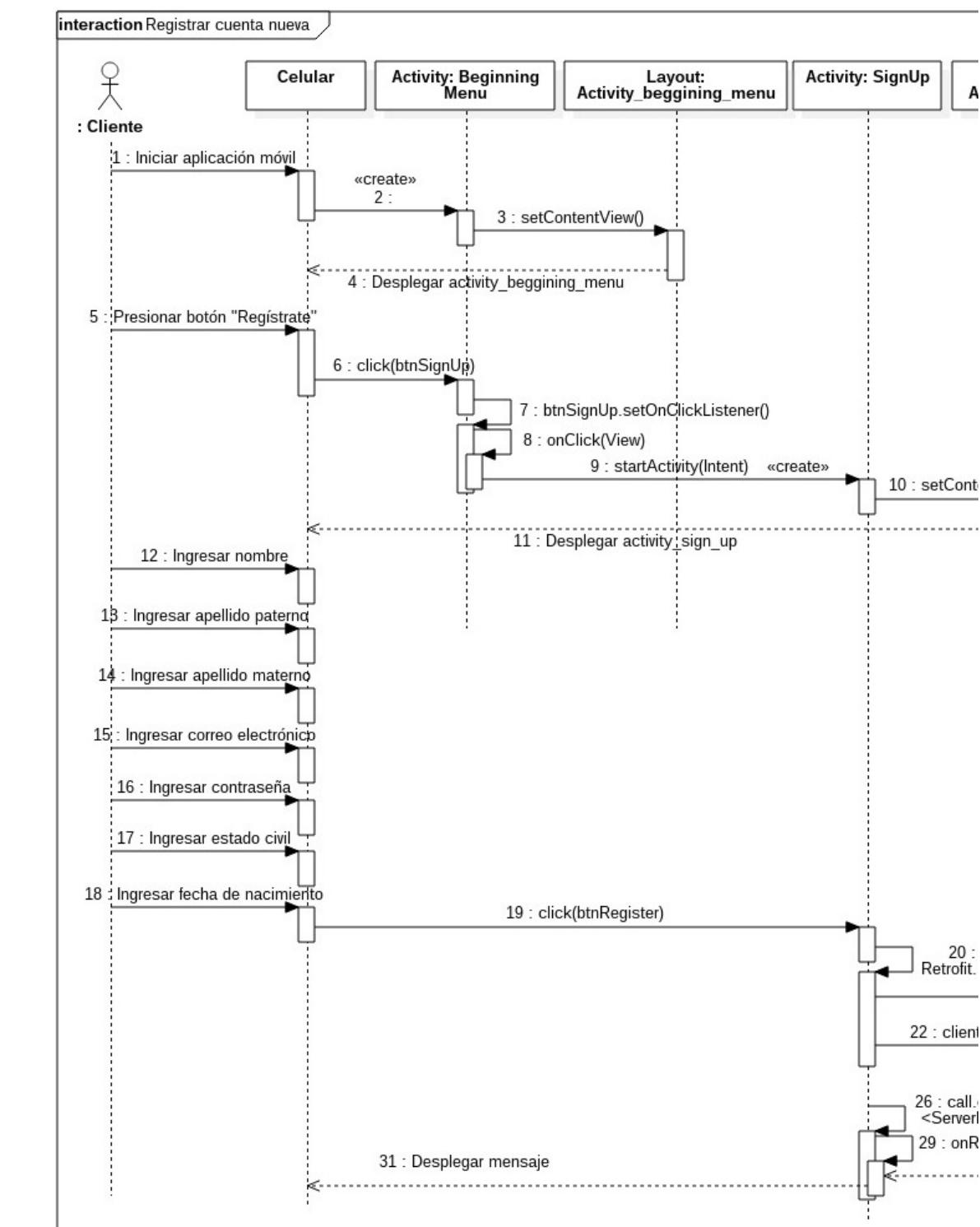


Figura 5.64: Diagrama de secuencia para registrar cuenta nueva (Parte 1).

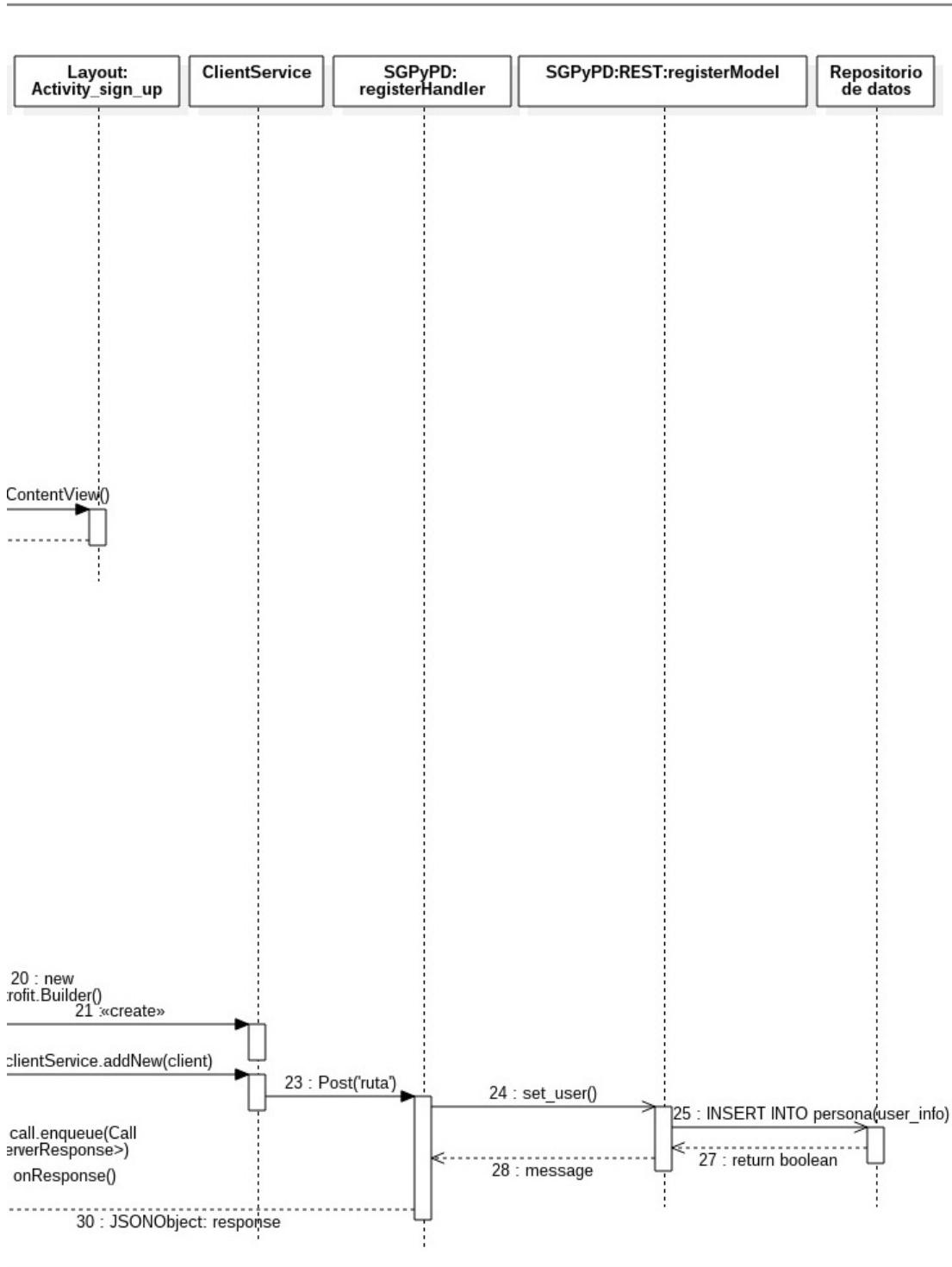


Figura 5.65: Diagrama de secuencia para registrar cuenta nueva (Parte 2).

## Consultar folletos

El diagrama de la figura 5.66 se dividió en dos secciones para obtener una mejor visualización, dichas secciones corresponden a la figura 5.67 y 5.68. Este diagrama corresponde a la secuencia “Consultar folletos” mismo que es desplegado al iniciar sesión en la aplicación y dentro del cual se muestran las promociones y descuentos de tiendas y productos.

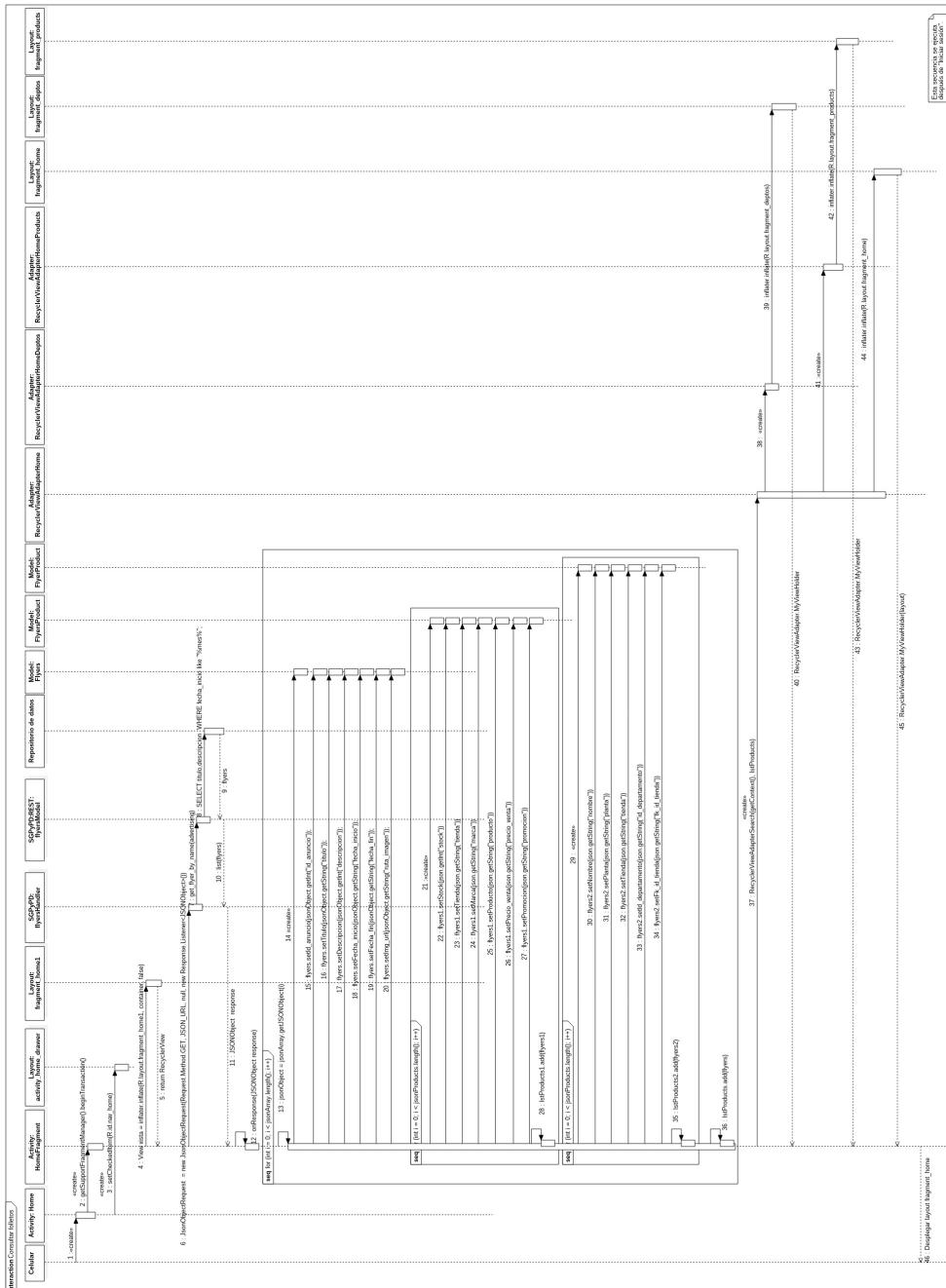


Figura 5.66: Diagrama de secuencia para consultar folletos (Visualización completa).

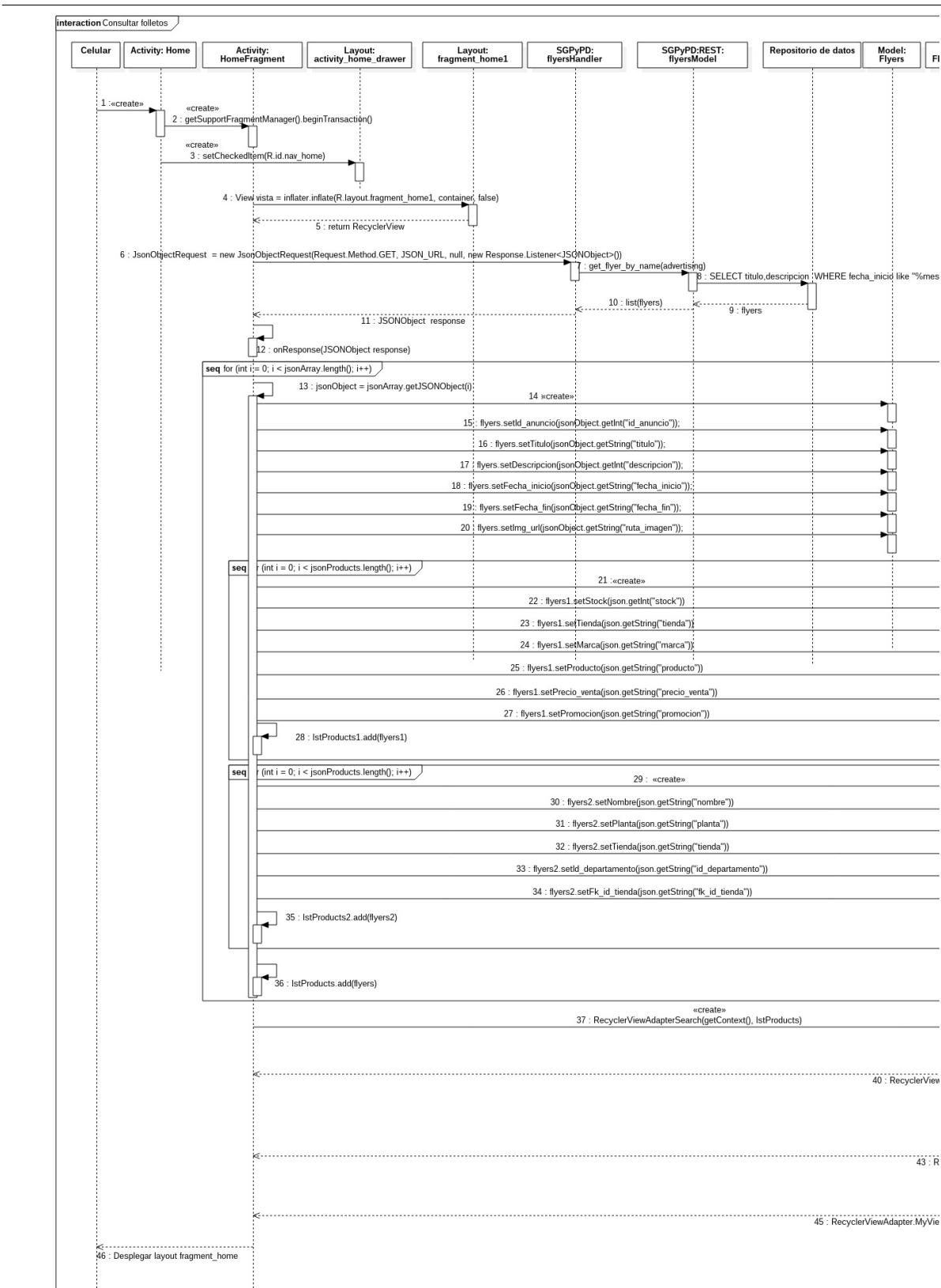


Figura 5.67: Diagrama de secuencia para consultar folletos (Parte 1).

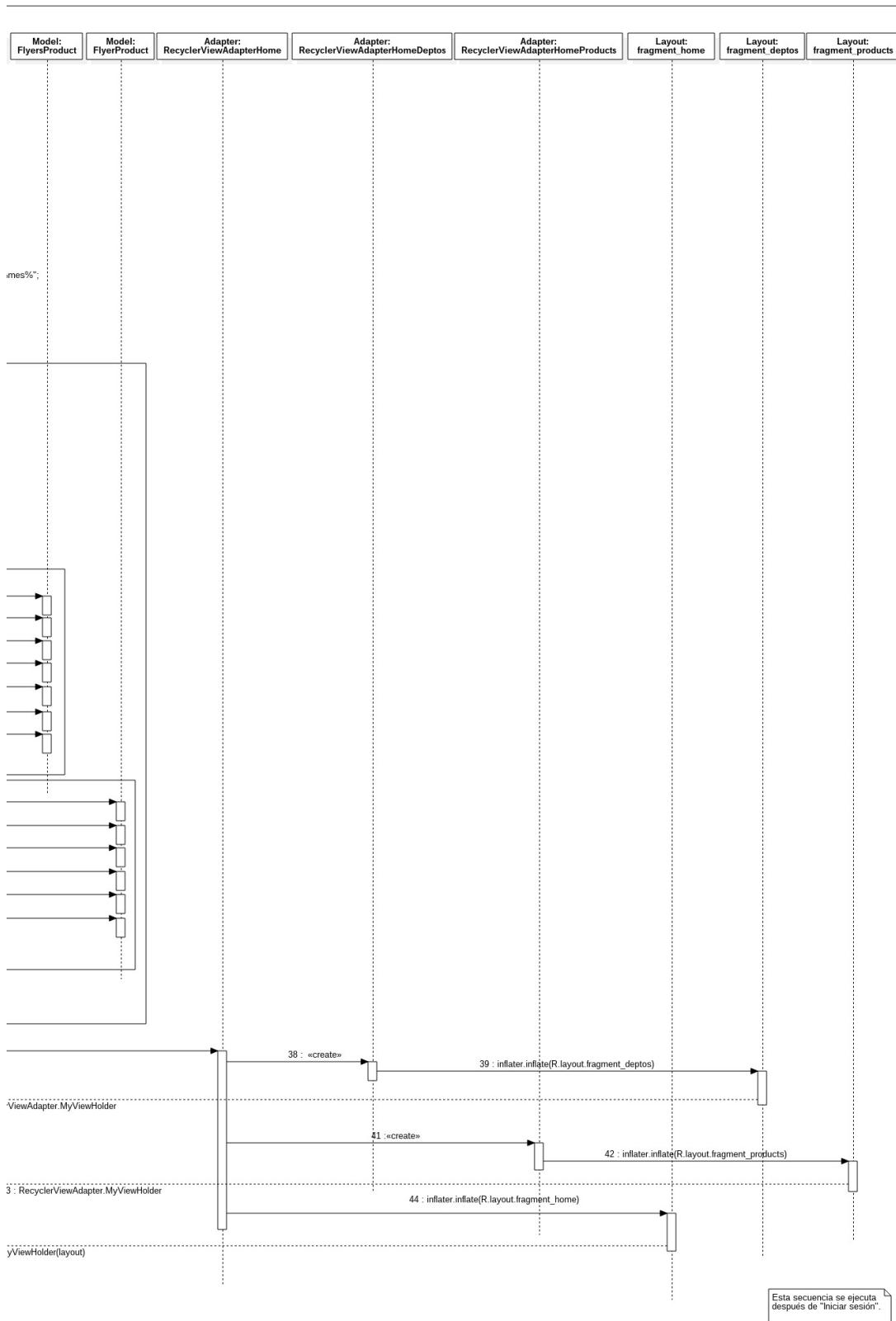


Figura 5.68: Diagrama de secuencia para consultar folletos (Parte 2).

### Visualizar menú de opciones

El diagrama de la figura 5.69, se dividió en dos secciones para una mejor visualización, estas corresponden a la figura 5.70 y 5.71. Este diagrama pertenece a la secuencia “Visualizar menú de opciones”, en el cual como su nombre lo dice, se ejecuta el proceso en el que el usuario puede observar las opciones que el menú le proporciona dentro del sistema.

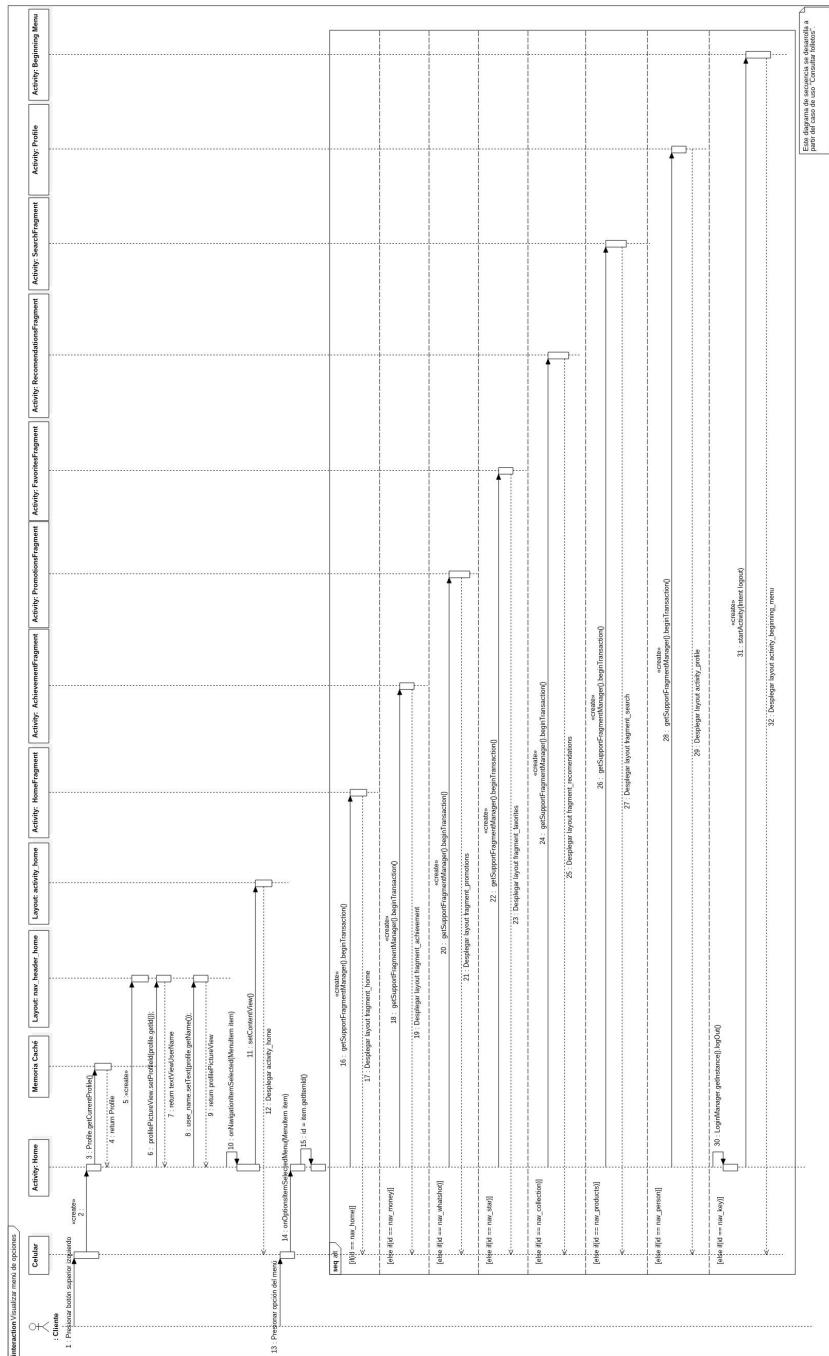


Figura 5.69: Diagrama de secuencia para visualizar menú de opciones (Visualización completa).

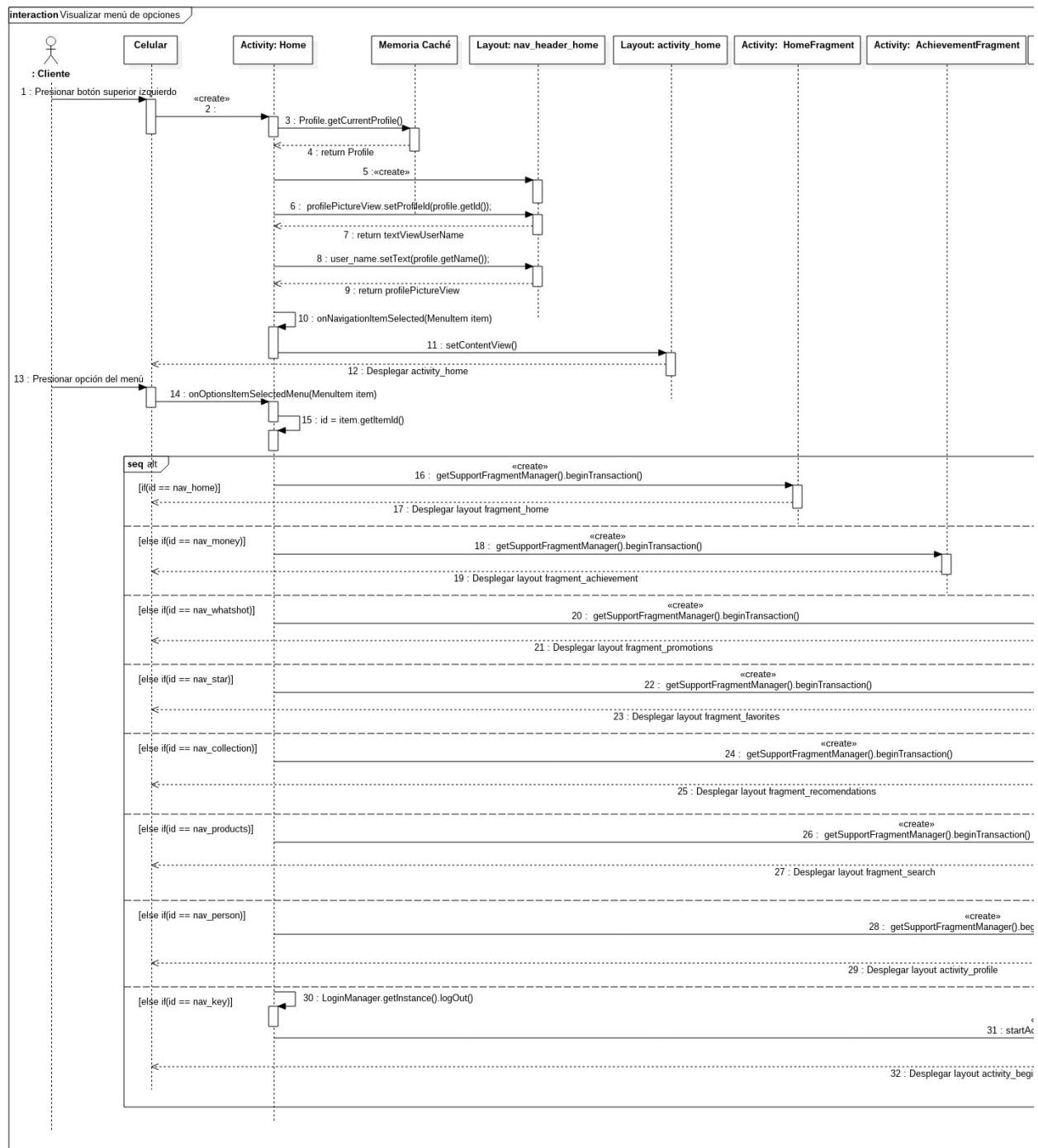


Figura 5.70: Diagrama de secuencia para visualizar menú de opciones (Parte 1).

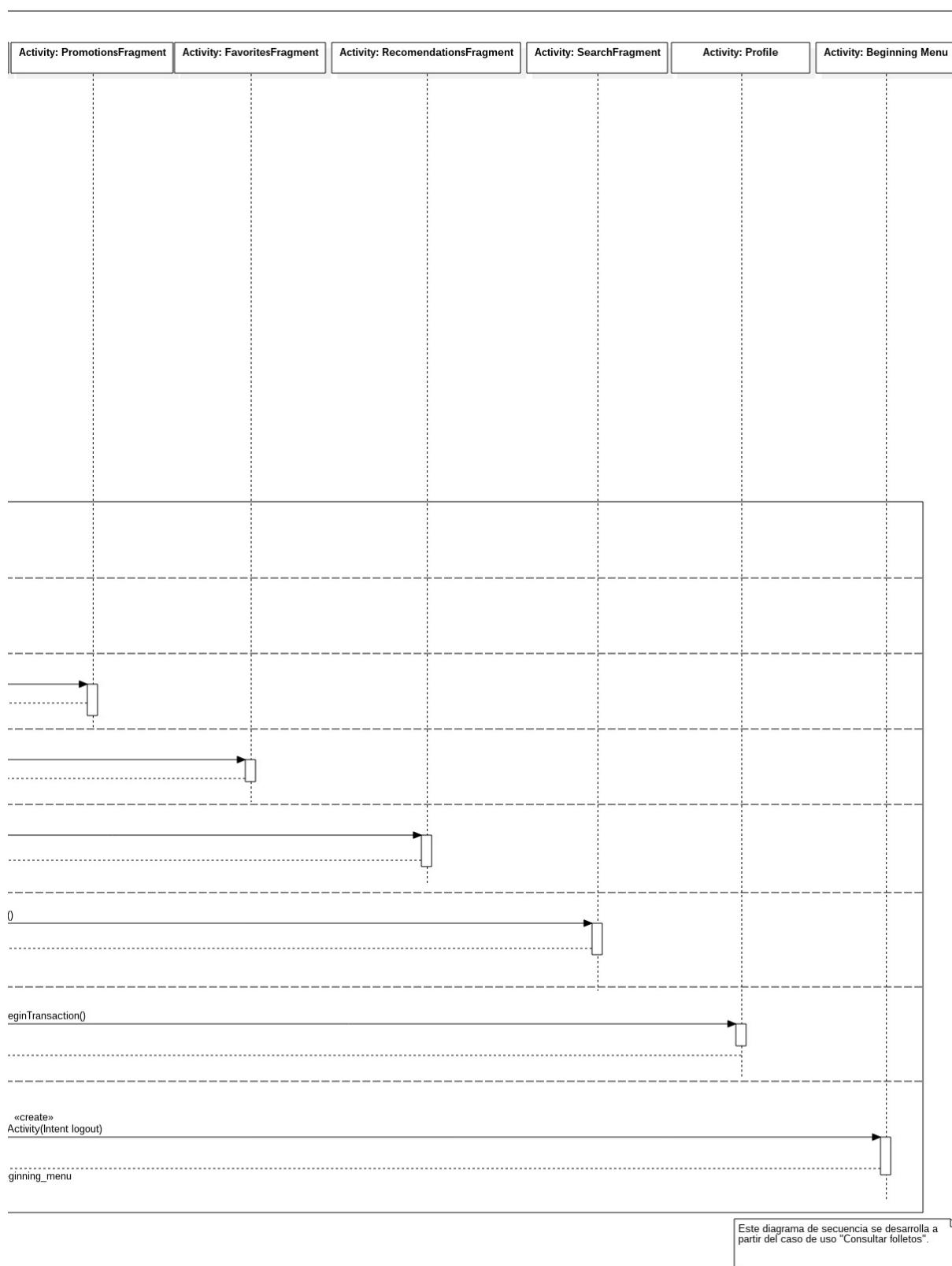


Figura 5.71: Diagrama de secuencia para visualizar menú de opciones (Parte 2).

## Consultar logros

El diagrama de la figura 5.72 corresponde a la secuencia “Consultar logros”, dicha opción se localiza dentro del menú de opciones y despliega la lista de beneficios que obtiene el usuario al obtener un logro en la aplicación.

Este se dividió en dos secciones para una mejor visualización, estas corresponden a la figura 5.73 y 5.74.

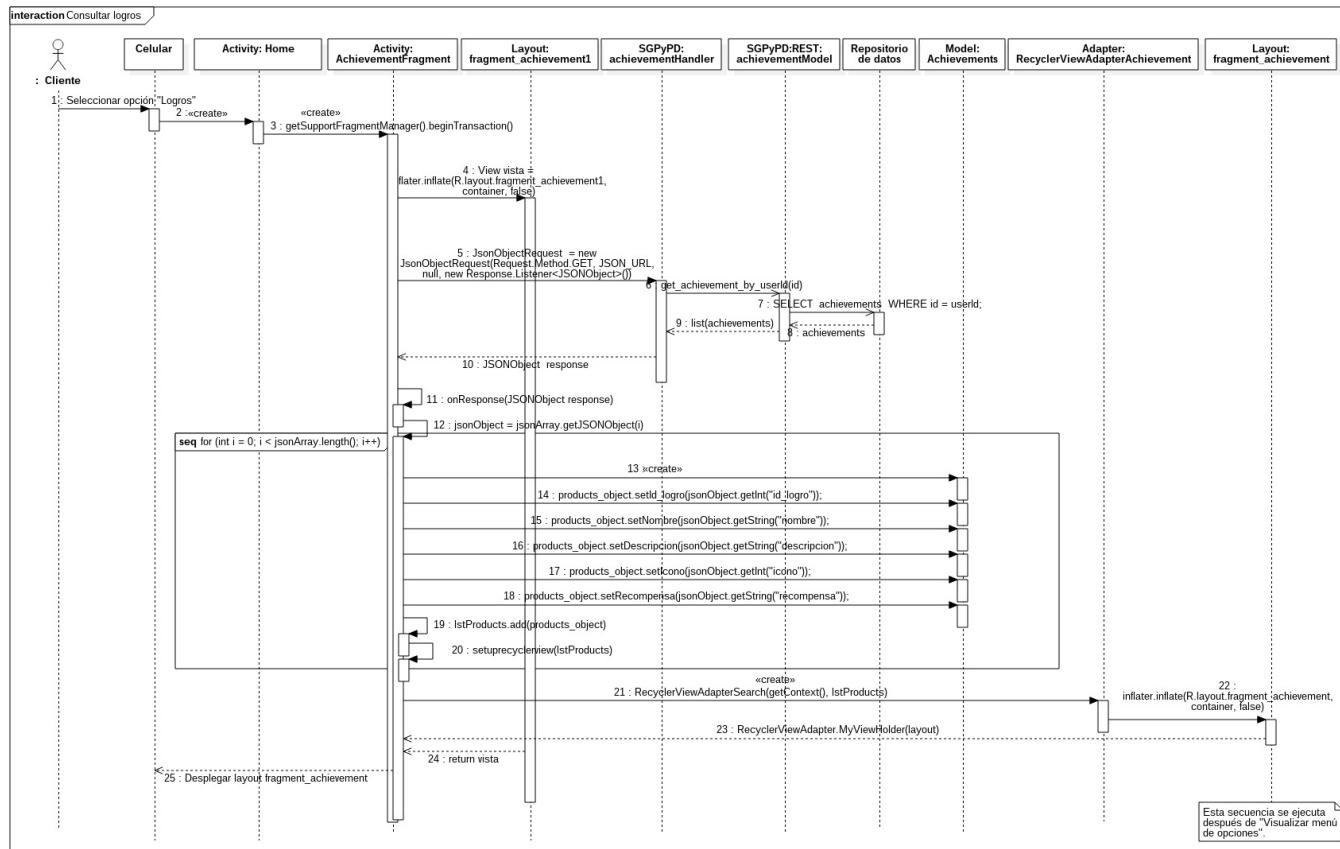


Figura 5.72: Diagrama de secuencia para consultar logros (Visualización completa).

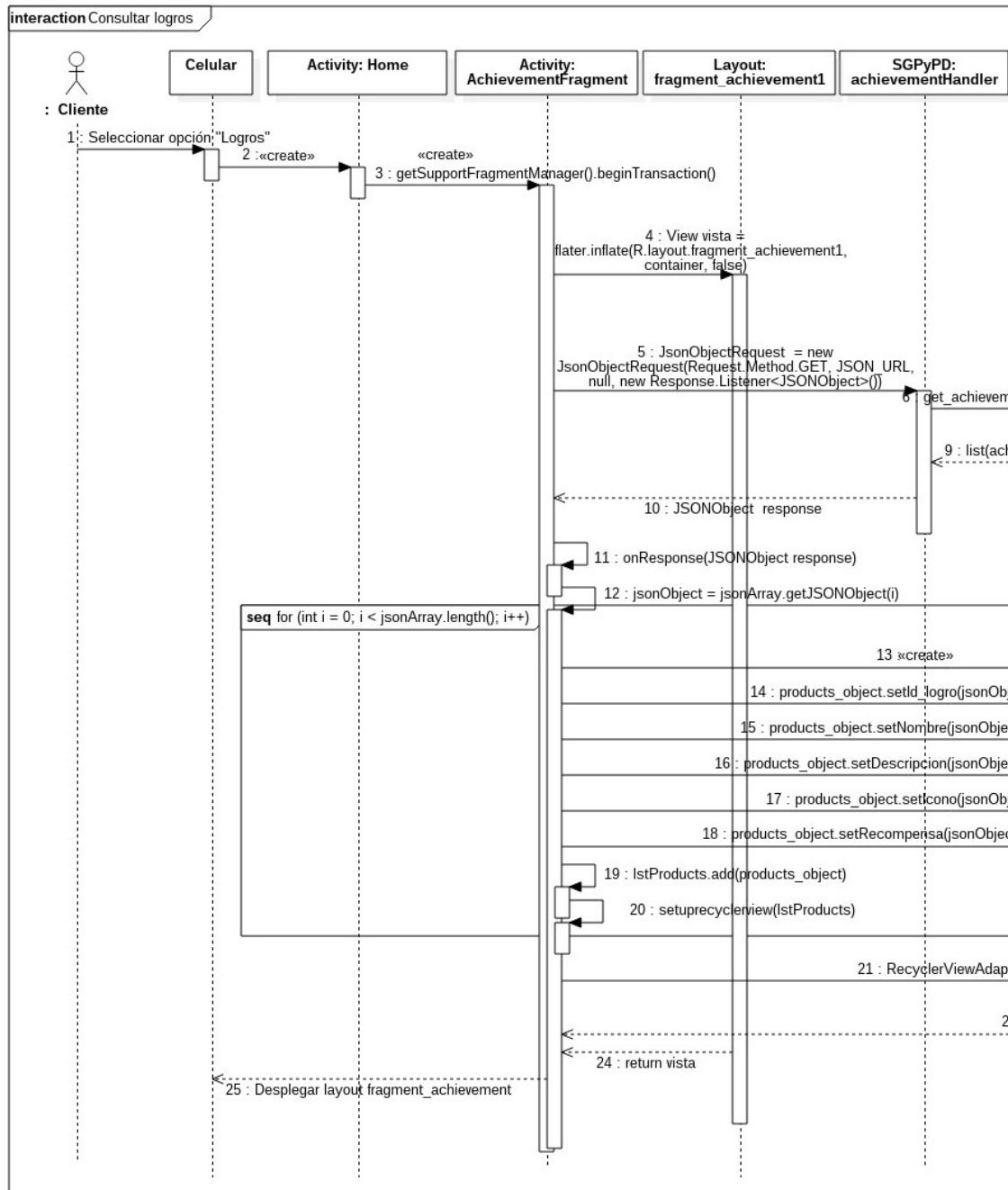


Figura 5.73: Diagrama de secuencia para consultar logros (Parte 1).

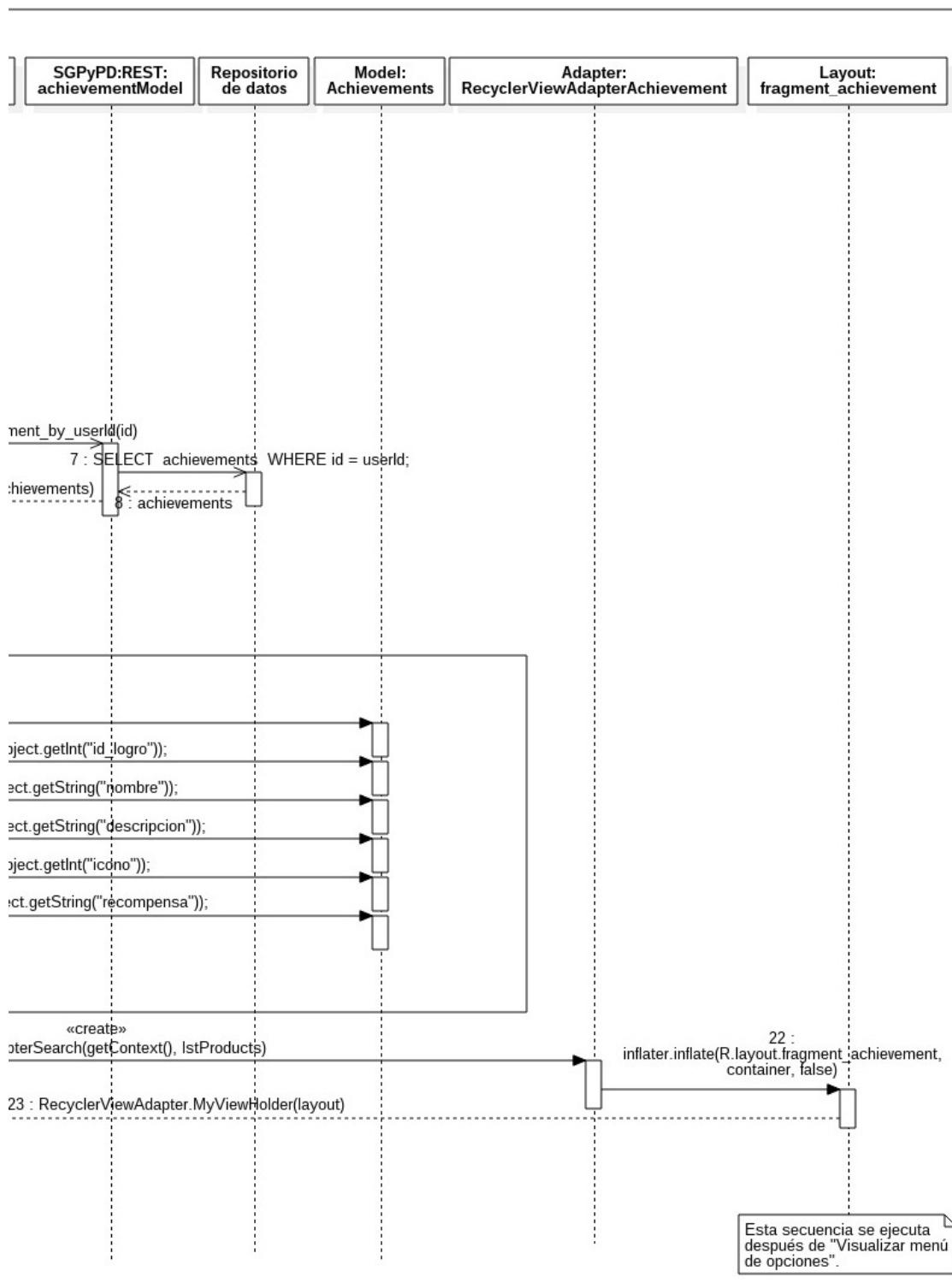


Figura 5.74: Diagrama de secuencia para consultar logros (Parte 2).

### Consultar promociones

El diagrama de la figura 5.75 corresponde a la secuencia “Consultar promociones”, dicha opción se localiza dentro del menú de opciones y despliega la lista de promociones que hay en las diferentes tiendas debido a una fecha en especial como Navidad o el día de las madres.

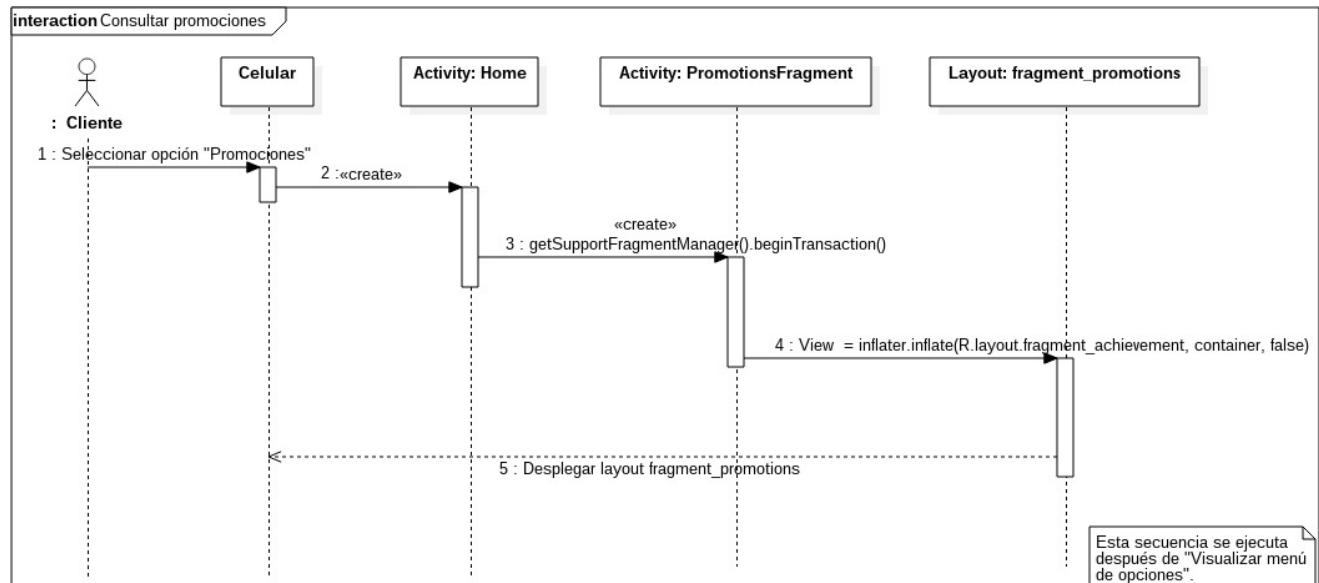


Figura 5.75: Diagrama de secuencia para consultar promociones (Visualización completa).

### Visualizar favoritos

El diagrama de la figura 5.76, se dividió en dos secciones para una mejor visualización, estas corresponden a la figura 5.77 y 5.78. Este diagrama pertenece a la secuencia “Visualizar favoritos”, en el cual como su nombre lo dice, permite al usuario observar los productos que ha seleccionado anteriormente como favorito.

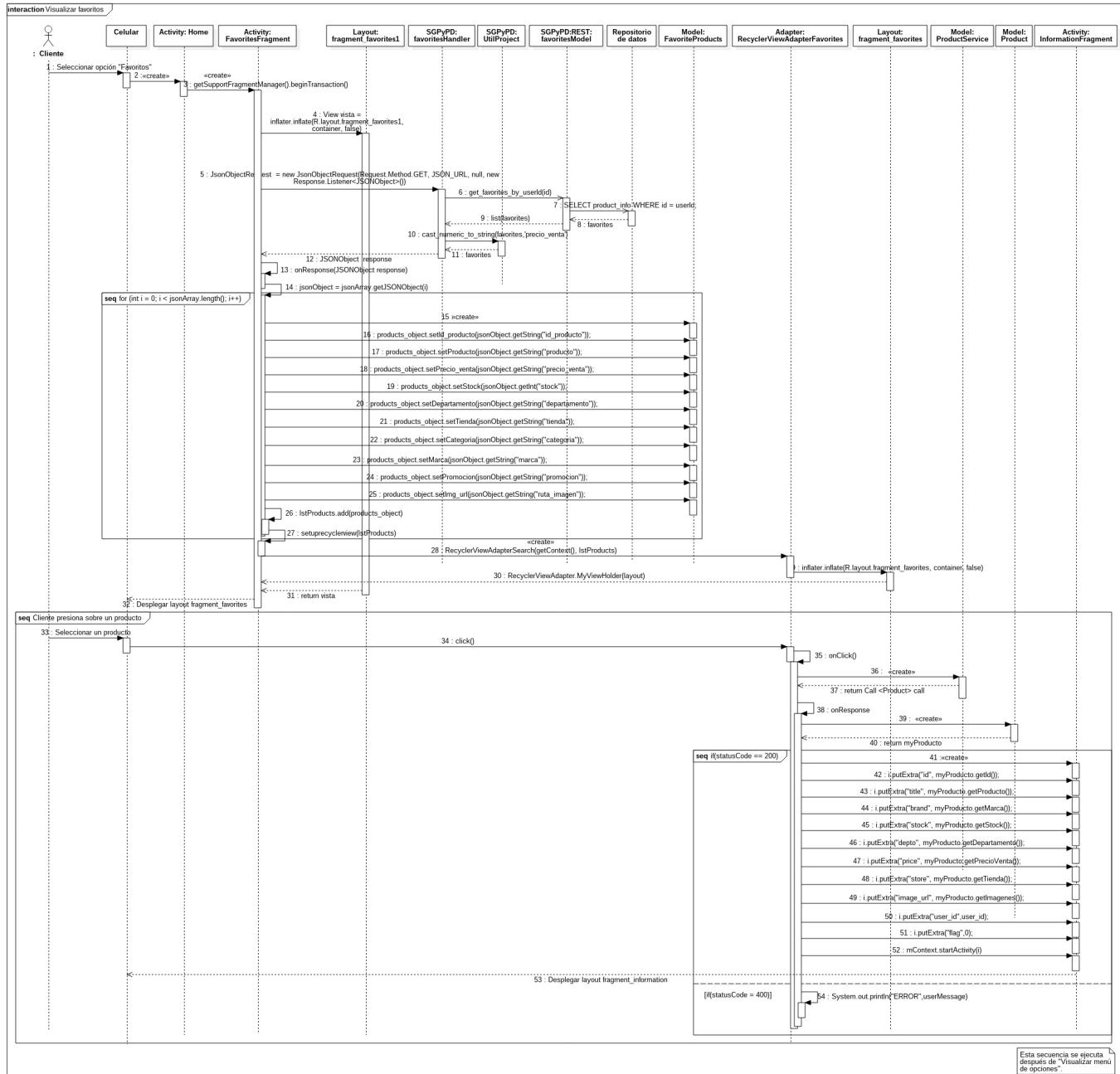


Figura 5.76: Diagrama de secuencia para visualizar favoritos (Visualización completa).

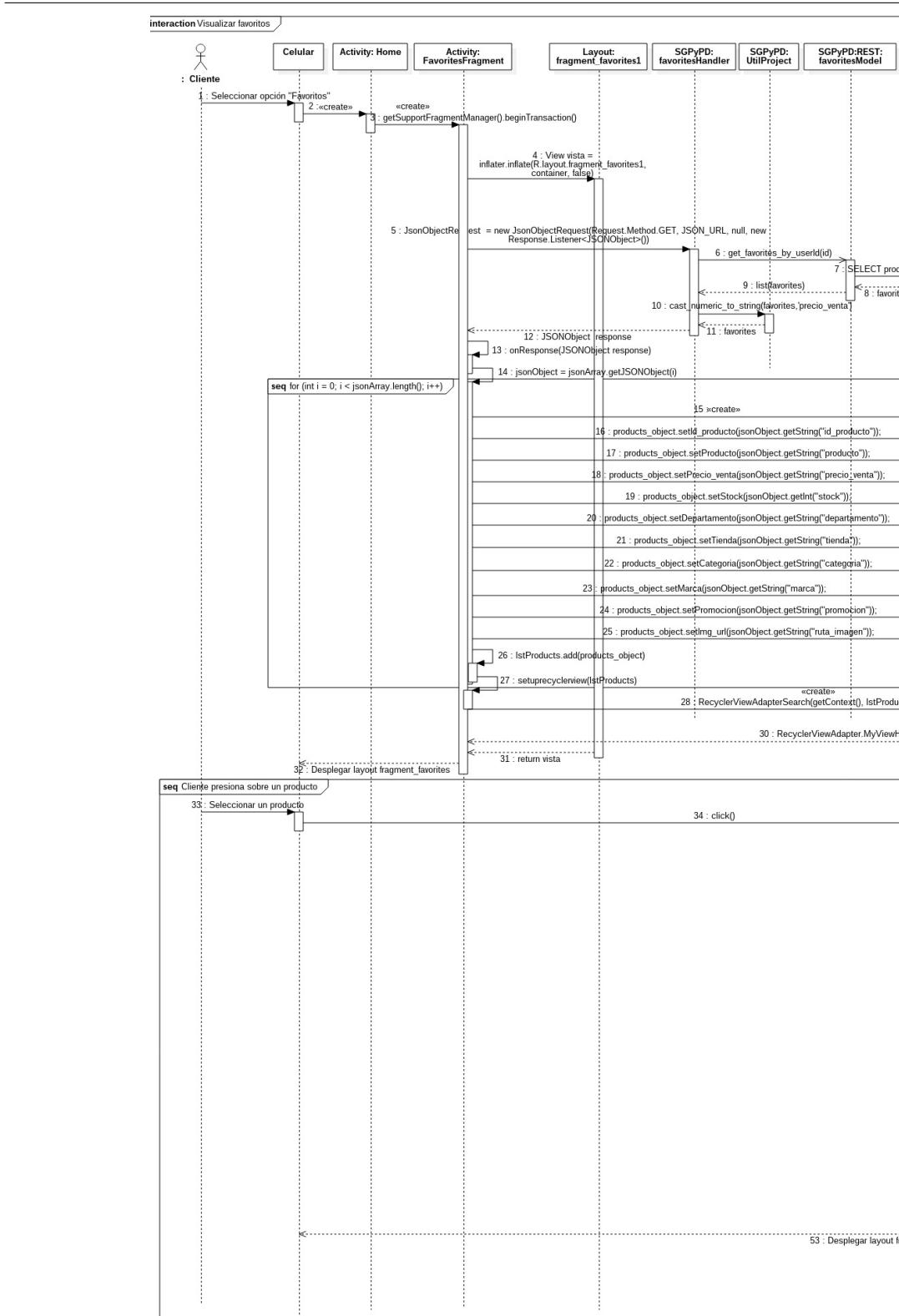


Figura 5.77: Diagrama de secuencia para visualizar favoritos (Parte 1).

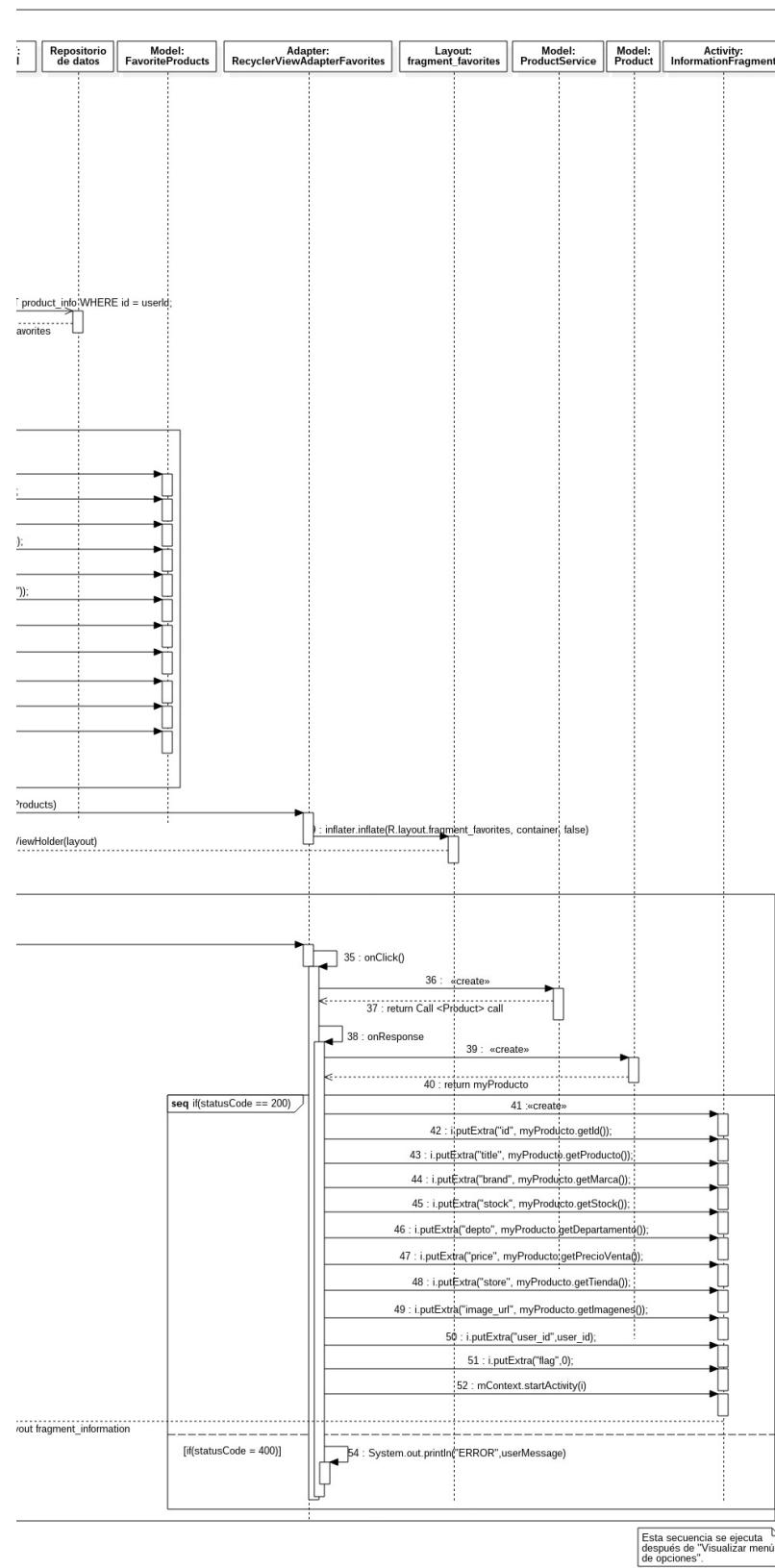


Figura 5.78: Diagrama de secuencia para visualizar favoritos (Parte 2).

### Añadir/Eliminar producto de favoritos

El diagrama de la figura 5.79 se muestran las secuencias “Añadir producto a favoritos” y “Eliminar producto de favoritos”. Es importante recalcar que se observan dichas secuencias juntas debido a que ambas hacen uso de una misma clase para realizar su funcionamiento, únicamente difiere en ellas la visibilidad de uno u otro botón, mismo que igual se plasma en dicho diagrama.

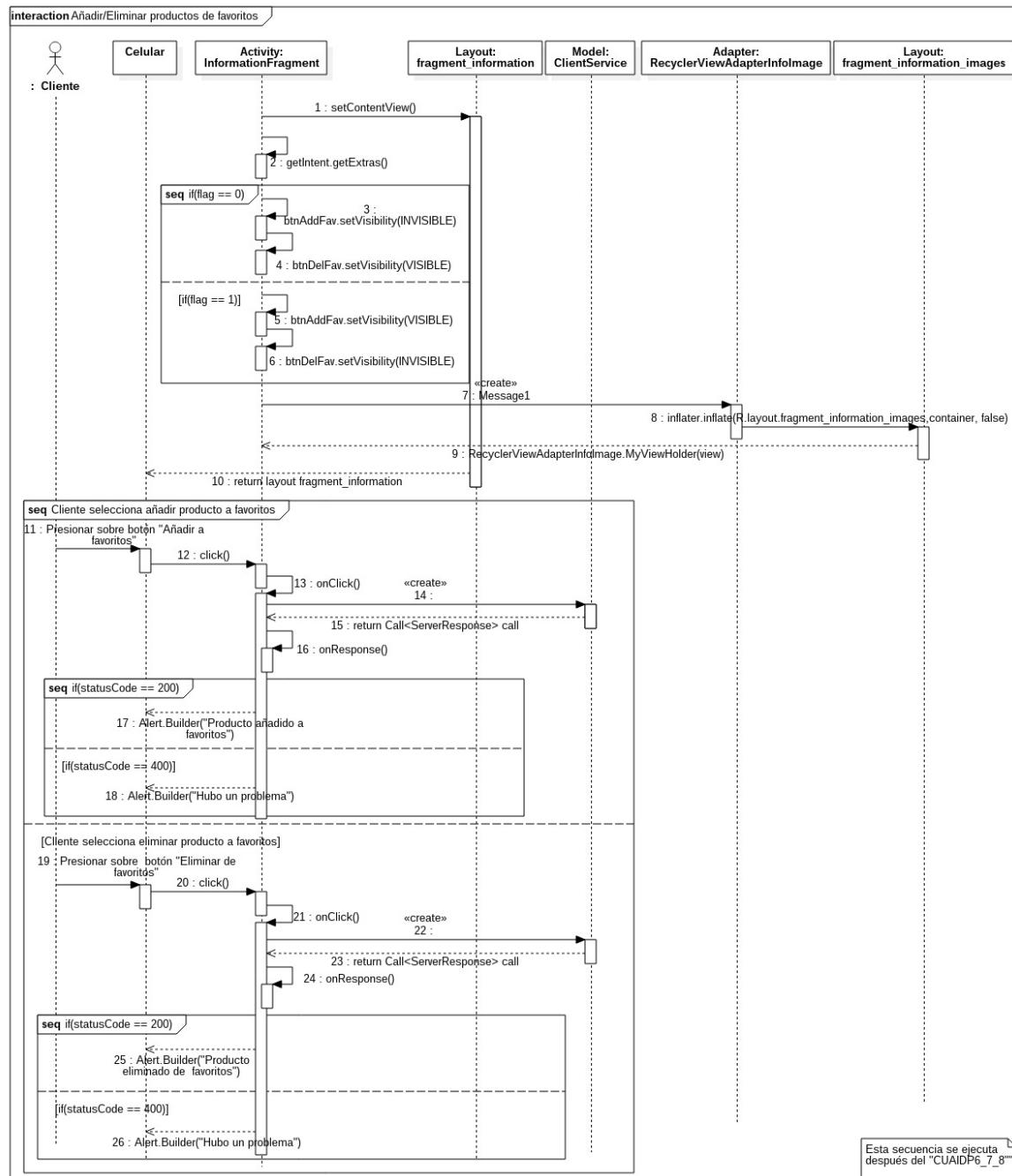


Figura 5.79: Diagrama de secuencia para añadir o eliminar productos a favoritos (Visualización completa).

## Consultar recomendaciones

El diagrama de la figura 5.80, se dividió en dos secciones para una mejor visualización, estas corresponden a la figura 5.81 y 5.82. Este diagrama pertenece a la secuencia “Consultar recomendaciones”. En esta sección aún no se ha implementado la funcionalidad de recibir los productos recomendados particularmente para un cliente, sin embargo se observa la secuencia del como se desplegarían los productos obtenidos.

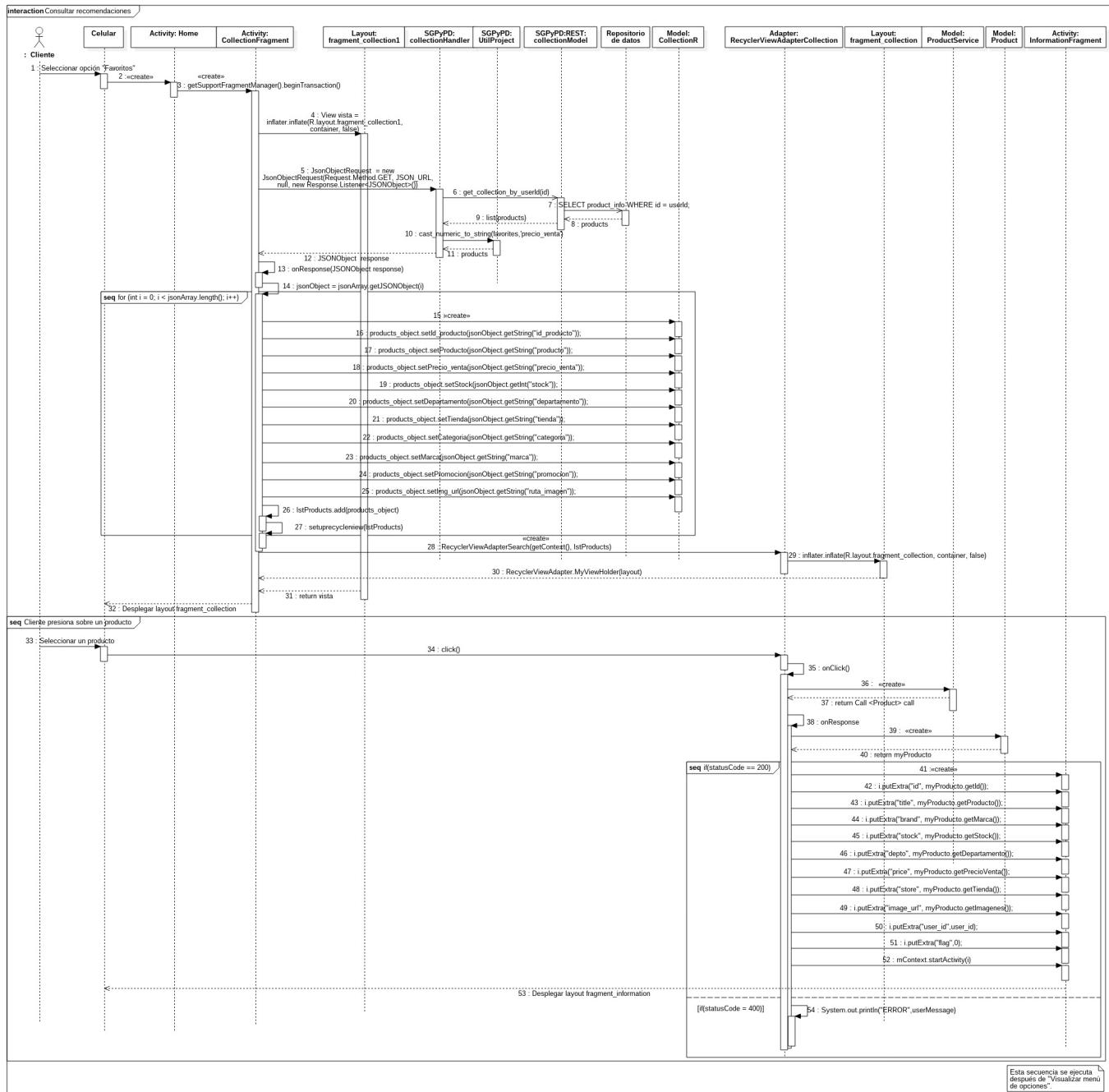


Figura 5.80: Diagrama de secuencia para consultar recomendaciones (Visualización completa).

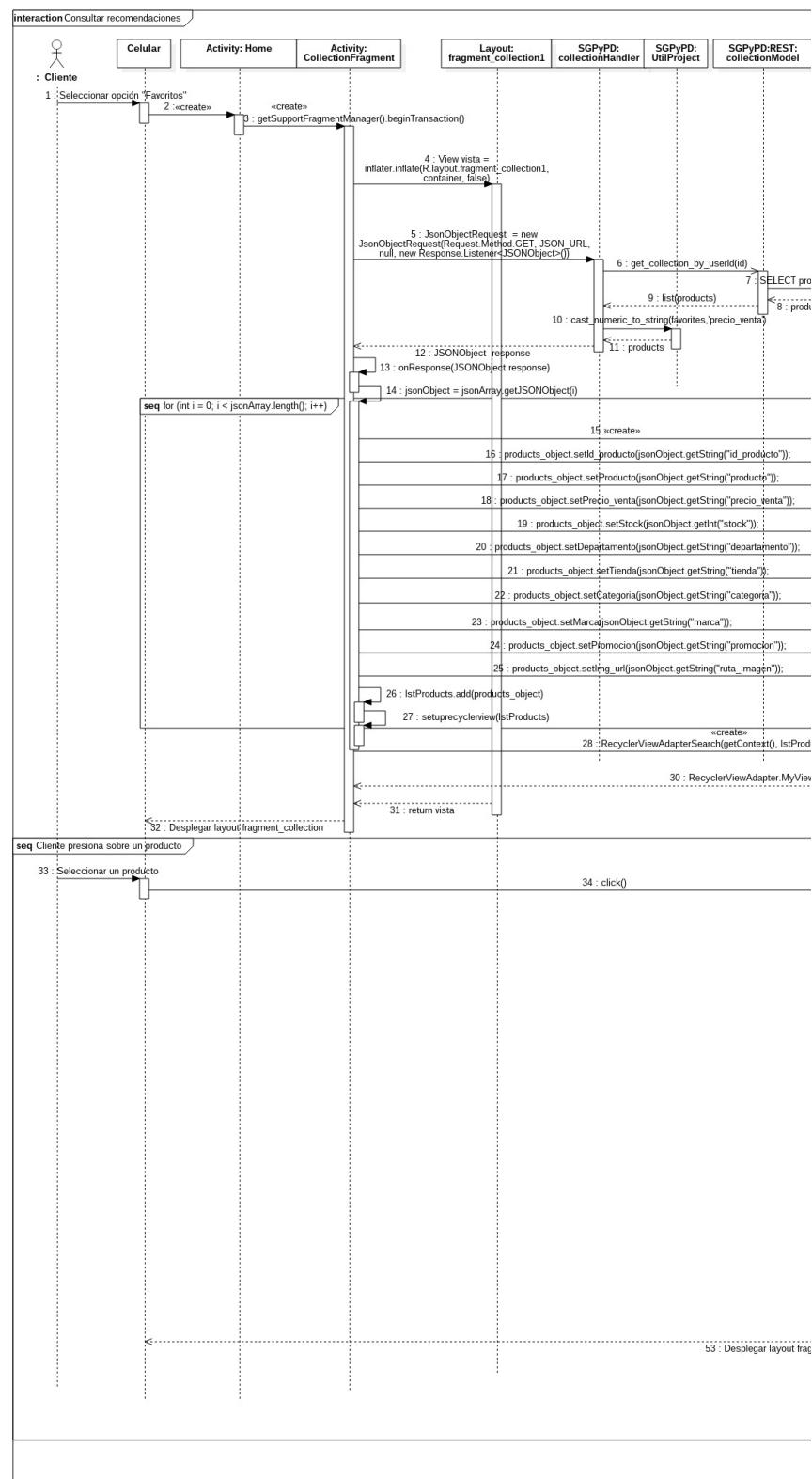


Figura 5.81: Diagrama de secuencia para consultar recomendaciones (Parte 1).

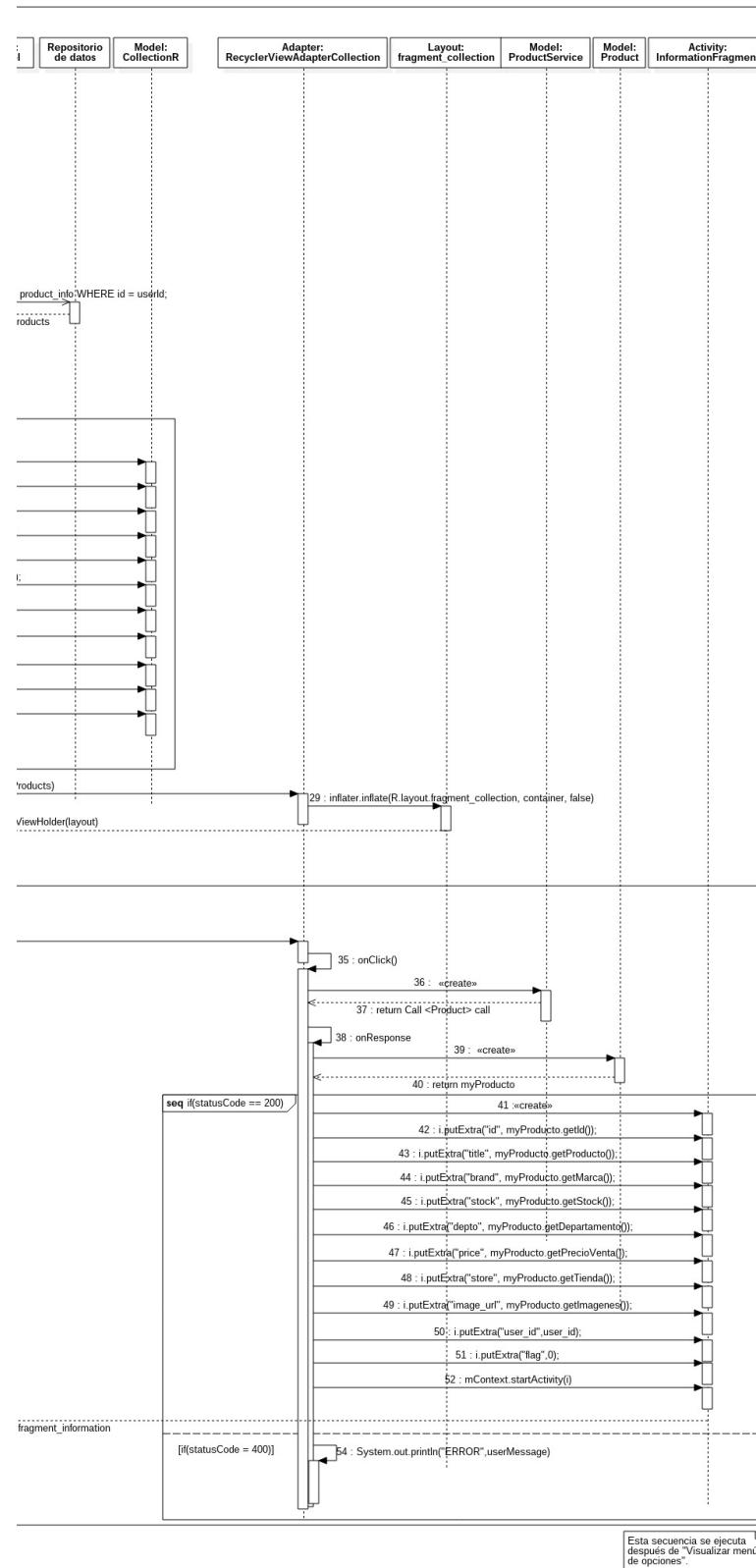


Figura 5.82: Diagrama de secuencia para consultar recomendaciones (Parte 2).

## Buscar productos

El diagrama de la figura 5.83, se dividió en dos secciones para una mejor visualización, estas corresponden a la figura 5.84 y 5.85. Este diagrama pertenece a la secuencia “Buscar productos”. Esta sección permite al usuario buscar por nombre productos que resulten ser de su interés.

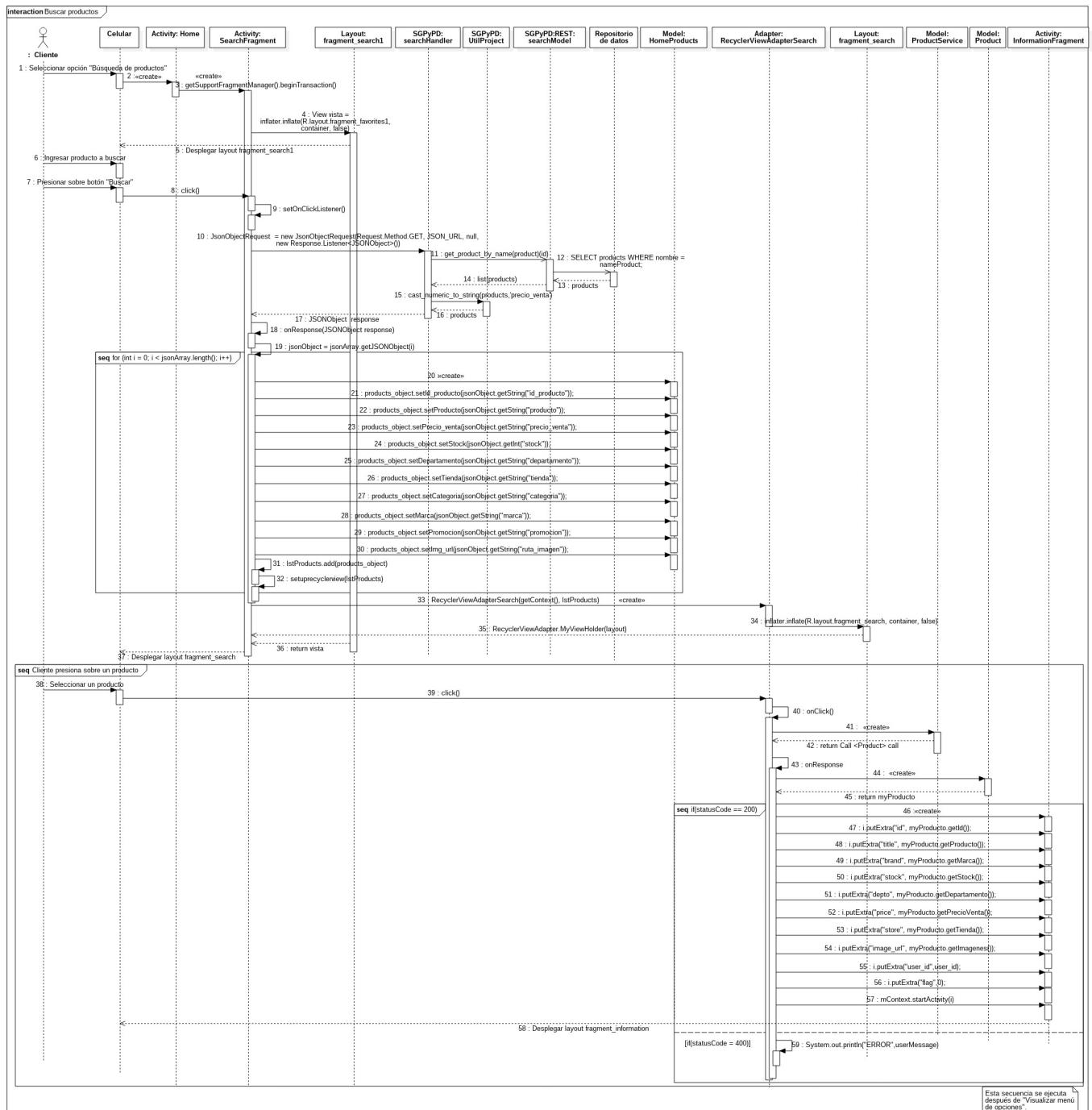


Figura 5.83: Diagrama de secuencia para buscar productos (Visualización completa).

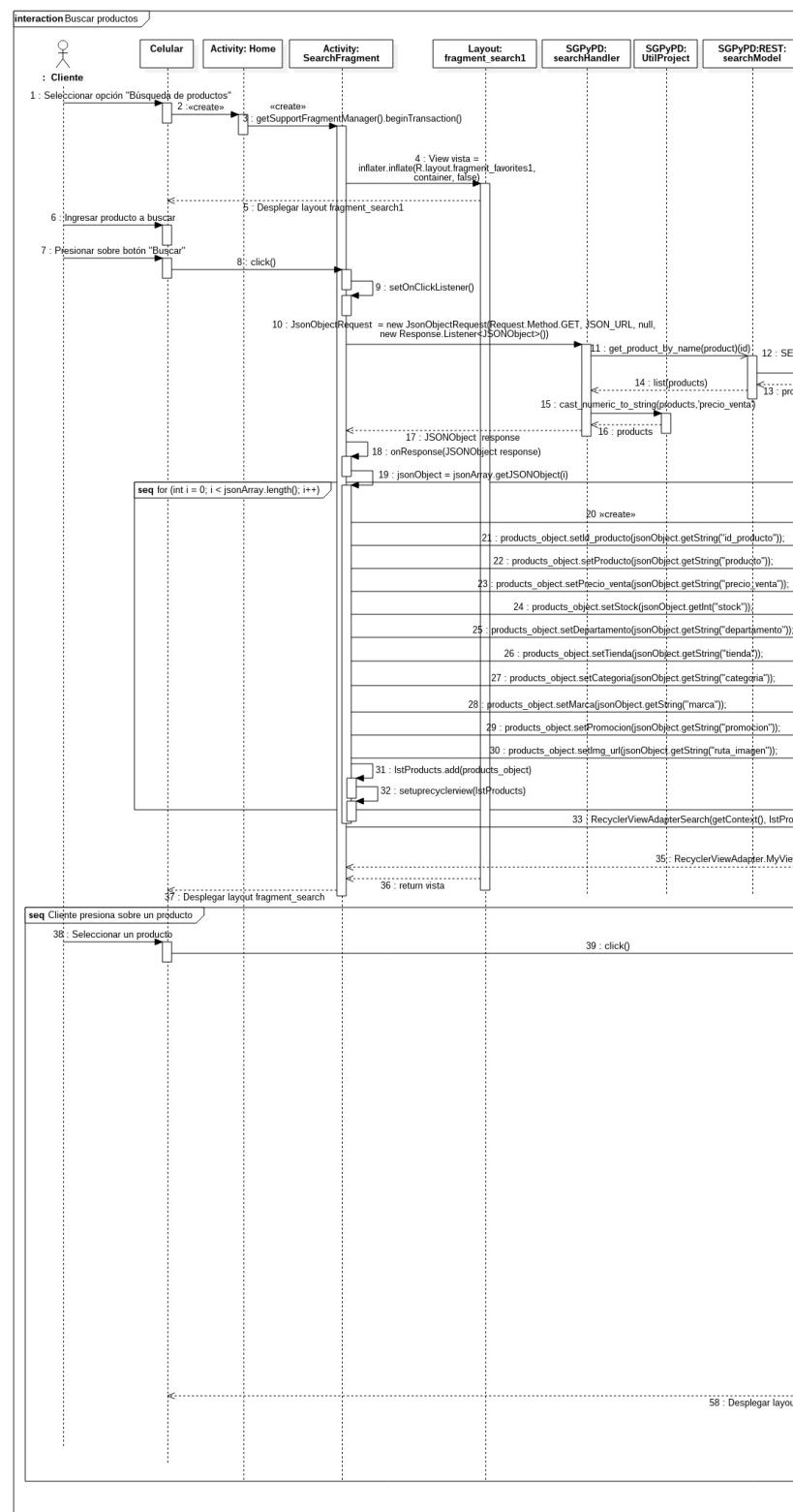


Figura 5.84: Diagrama de secuencia para buscar productos (Parte 1).

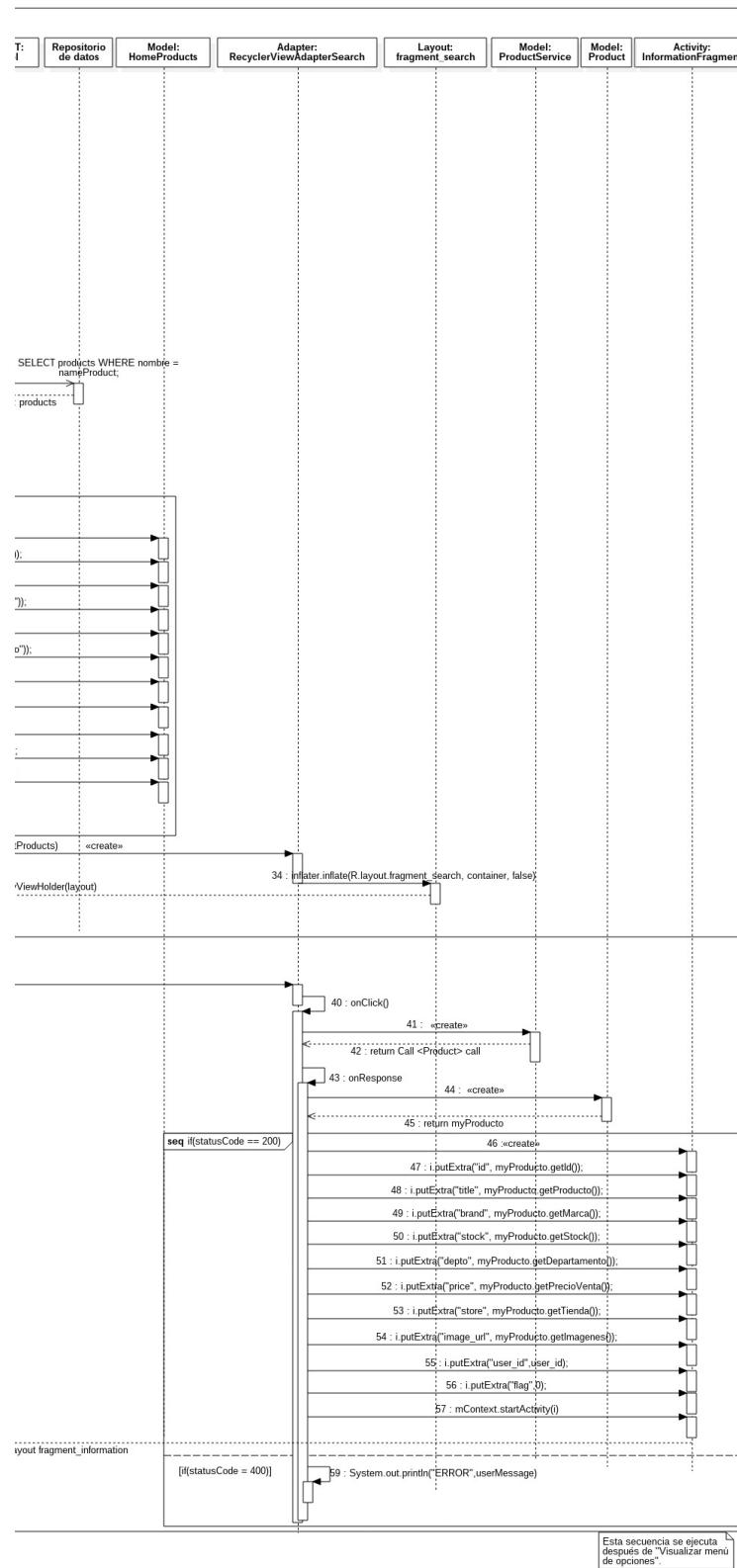


Figura 5.85: Diagrama de secuencia para buscar productos (Parte 2).

### Actualizar datos personales y consultar nivel

El diagrama de la figura 5.86, se dividió en dos secciones para una mejor visualización, estas corresponden a la figura 5.87 y 5.88. Este diagrama pertenece a la secuencia “Consultar datos personales”. Nota: *Este diagrama contiene la secuencia para el caso de uso “ CUAIDP9: Actualizar datos personales” y “ CUAIDP9.1: Consultar nivel” debido a que al solicitar los datos de un usuario al servidor, igualmente se recibe el nivel en el que el cliente se encuentra en ese momento. Sin embargo, es importante mencionar que al utilizar una clase en común con otros casos de uso, se ven en este plasmado solamente la funcionalidad básica del “ CUAIDP9.2: Actualizar gustos genéricos”, “ CUAIDP9.3: Consultar beneficios” y del “ CUAIDP9.4: Actualizar permisos”.*

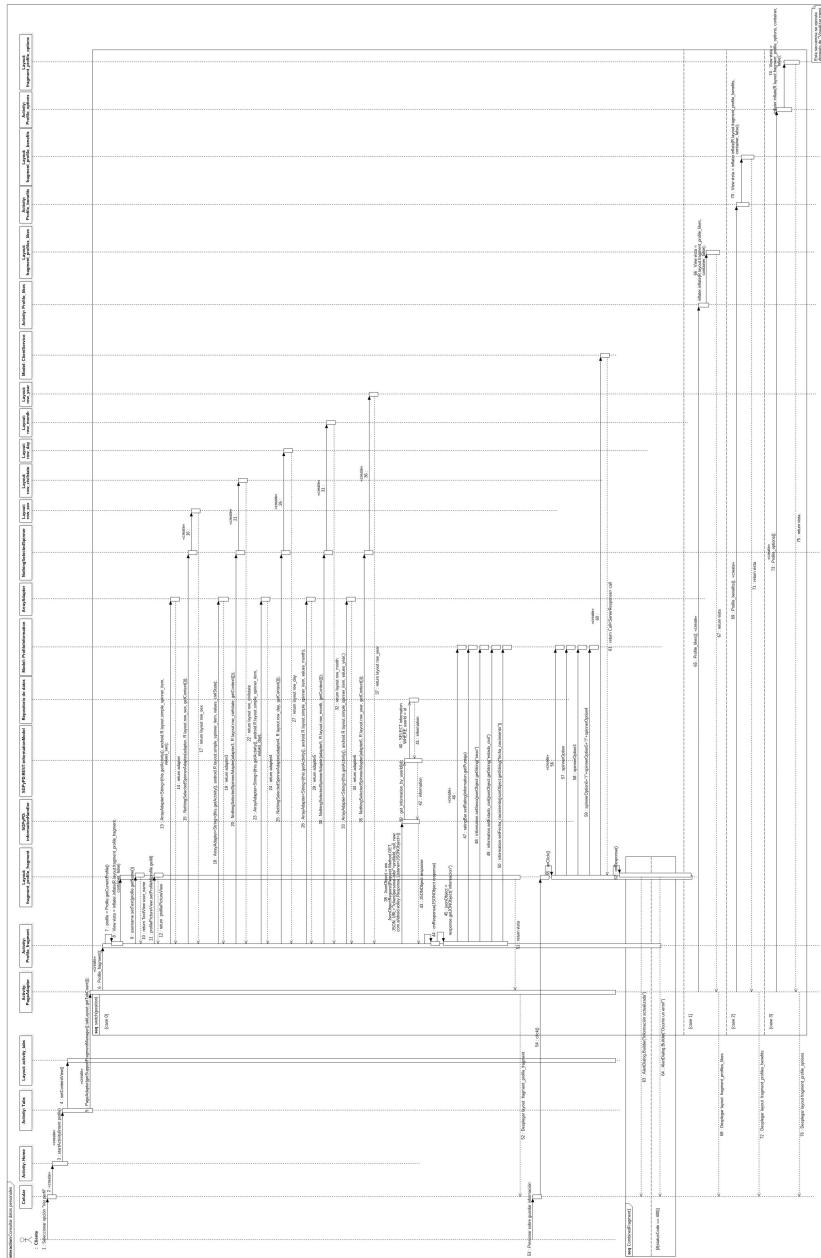


Figura 5.86: Diagrama de secuencia para consultar datos personales (Visualización completa).



Figura 5.87: Diagrama de secuencia para consultar datos personales (Parte 1).

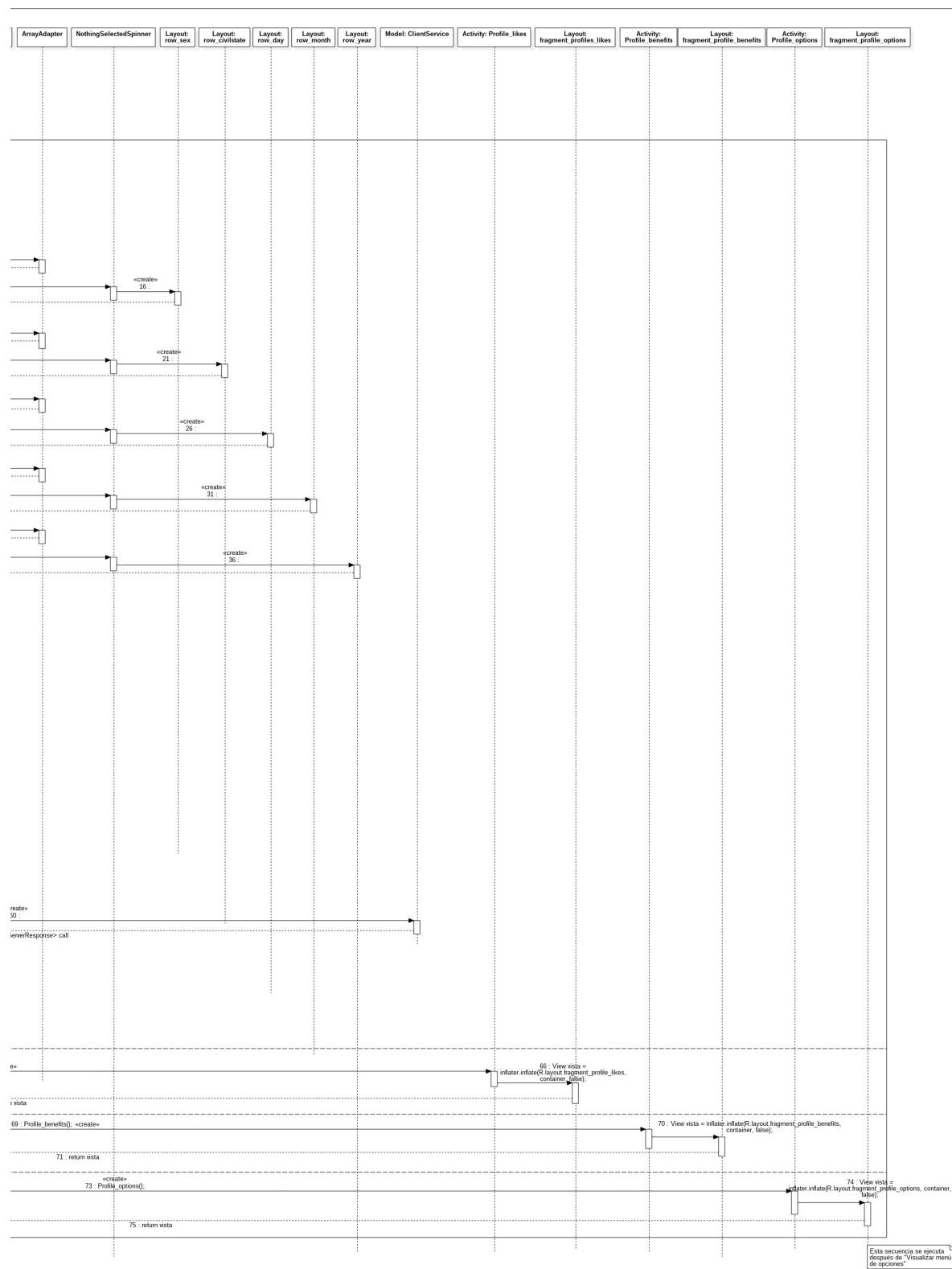


Figura 5.88: Diagrama de secuencia para consultar datos personales (Parte 2).

## Actualizar gustos genéricos

El diagrama de la figura 5.89, se dividió en dos secciones para una mejor visualización, estas corresponden a la figura 5.90 y 5.91. Este diagrama pertenece a la secuencia “Actualizar gustos genéricos”. Esta sección permite al usuario marcar las categorías que se asemejen más a sus gustos.

*Nota: Este diagrama contiene la secuencia para el caso de uso “ CUAIDP9.2: Actualizar gustos genéricos”, sin embargo, al utilizar una clase en común con otros casos de uso, se ven en este plasmado solamente la funcionalidad básica del “ CUAIDP9: Actualizar datos personales”, “ CUAIDP9.3: Consultar beneficios” y del “ CUAIDP9.4: Actualizar permisos”.*

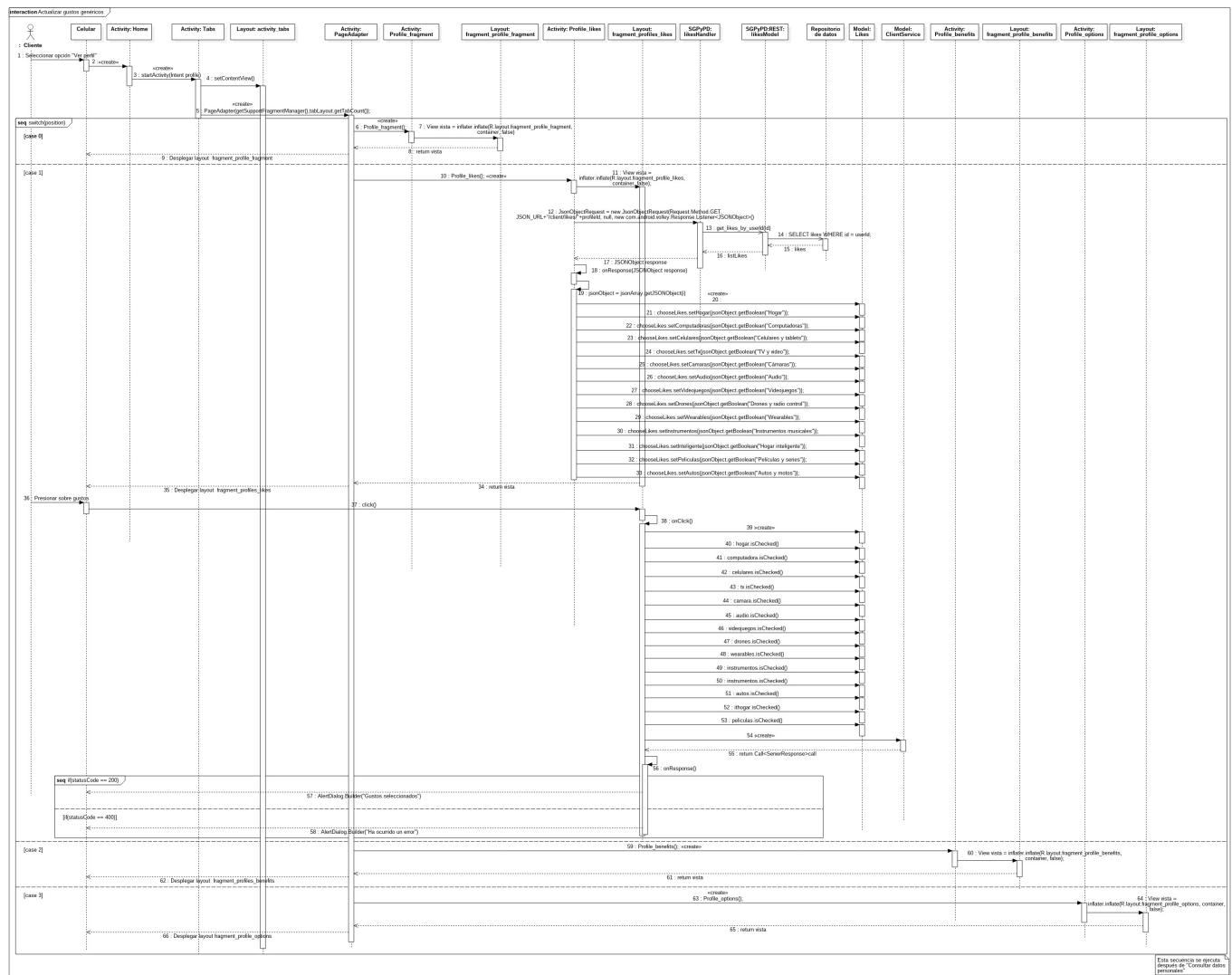


Figura 5.89: Diagrama de secuencia para actualizar gustos genéricos (Visualización completa).



Figura 5.90: Diagrama de secuencia para actualizar gustos genéricos (Parte 1).

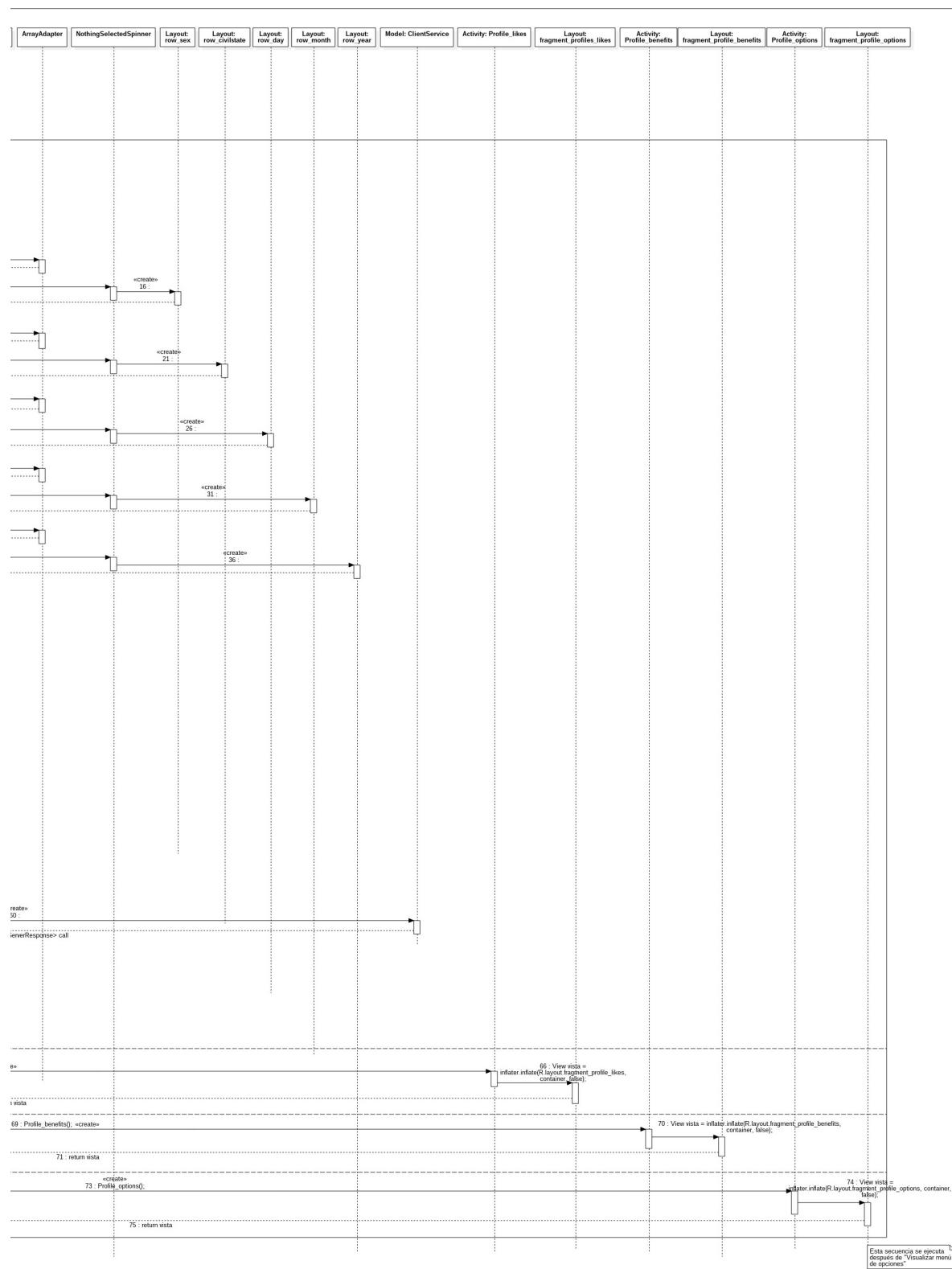


Figura 5.91: Diagrama de secuencia para actualizar gustos genéricos (Parte 2).

## Consultar beneficios

El diagrama de la figura 5.92, se dividió en dos secciones para una mejor visualización, estas corresponden a la figura 5.93 y 5.94. Este diagrama pertenece a la secuencia “Consultar beneficios”. Esta sección permite al usuario consultar los beneficios que ha adquirido con respecto al nivel en el que se encuentra actualmente. *Nota: Este diagrama contiene la secuencia para el caso de uso “ CUAIDP9.3: Consultar beneficios” , sin embargo, al utilizar una clase en común con otros casos de uso, se ven en este plasmado solamente la funcionalidad básica del “ CUAIDP9: Actualizar datos personales”, “ CUAIDP9.2: Actualizar gustos genéricos” y del “ CUAIDP9.4: Actualizar permisos”.*

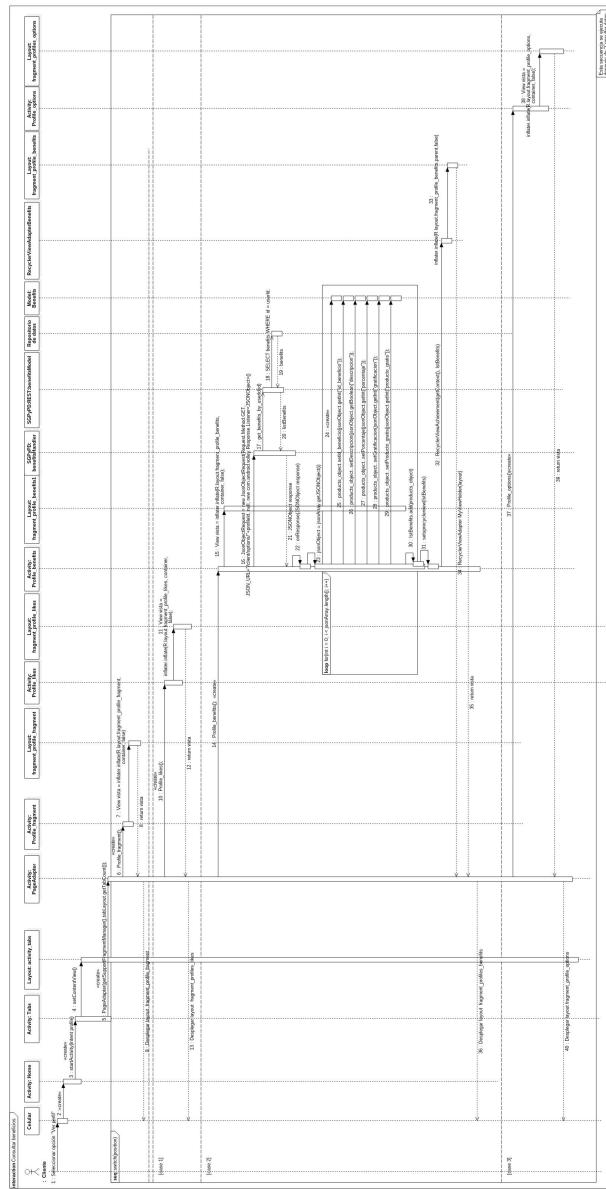


Figura 5.92: Diagrama de secuencia para consultar beneficios (Visualización completa).

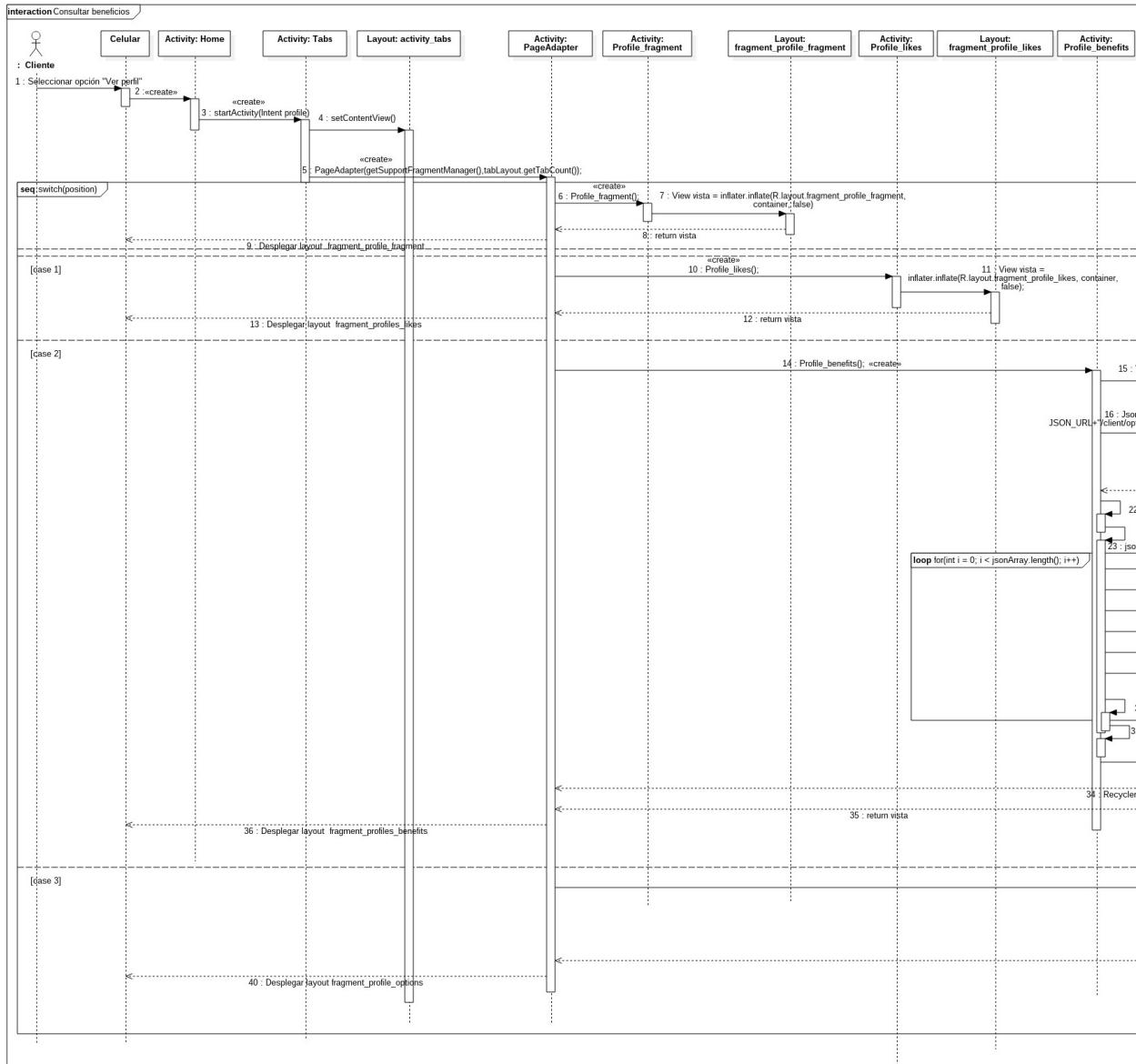


Figura 5.93: Diagrama de secuencia para consultar beneficios (Parte 1).

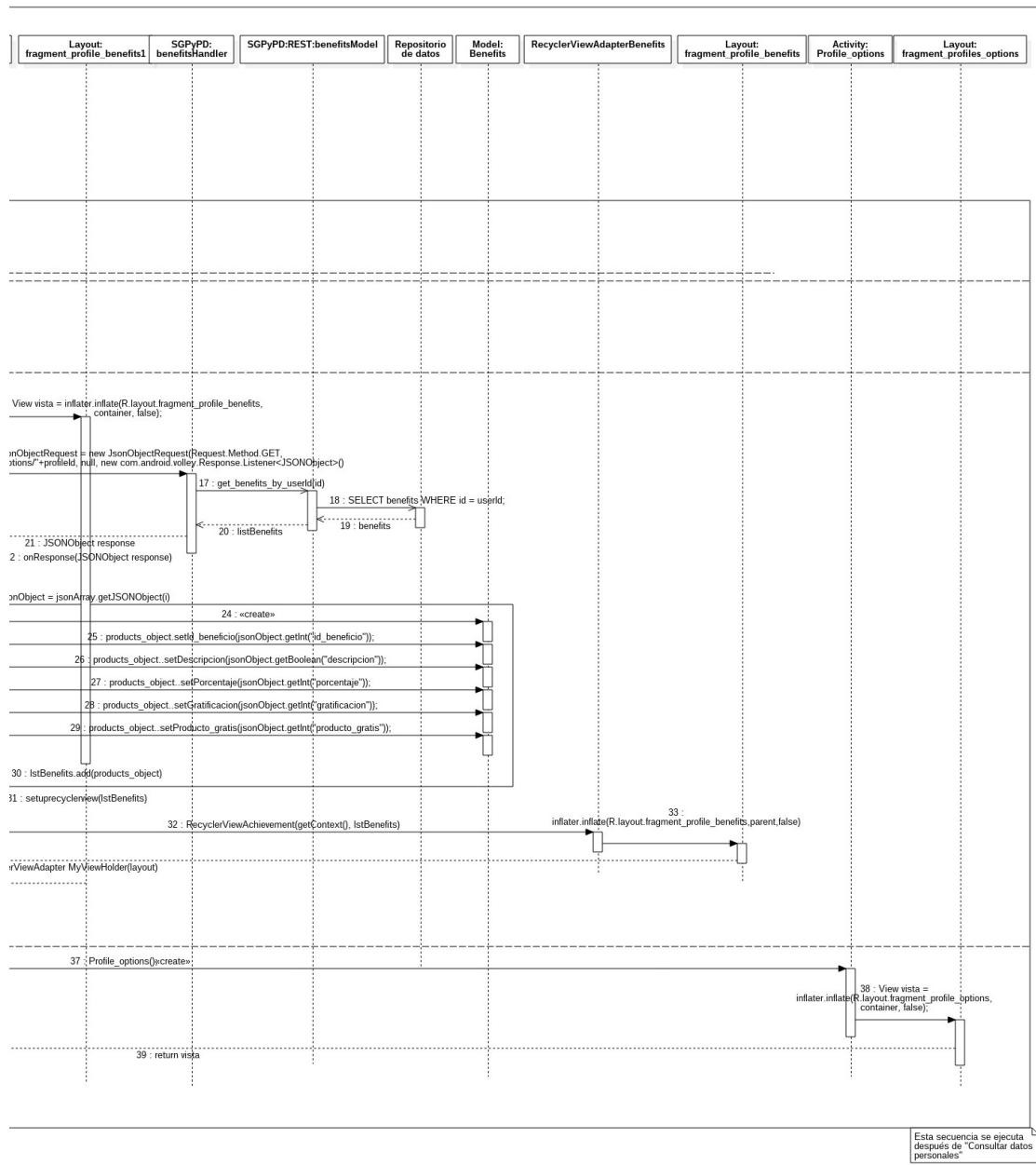


Figura 5.94: Diagrama de secuencia para consultar beneficios (Parte 2).

### Actualizar permisos

El diagrama de la figura 5.95, se dividió en dos secciones para una mejor visualización, estas corresponden a la figura 5.96 y 5.97. Este diagrama pertenece a la secuencia “Actualizar permisos”. Esta sección permite al usuario marcar los permisos a los que dará acceso al vendedor para poder ofrecer productos más acorde con sus preferencias.

*Nota: Este diagrama contiene la secuencia para el caso de uso “CUAIDP9.4: Actualizar permisos”, sin embargo, al utilizar una clase en común con otros casos de uso, se ven en este plasmado solamente la funcionalidad básica del “CUAIDP9: Actualizar datos personales”, “CUAIDP9.2: Actualizar gustos genéricos” y del “CUAIDP9.3: Consultar beneficios”.*

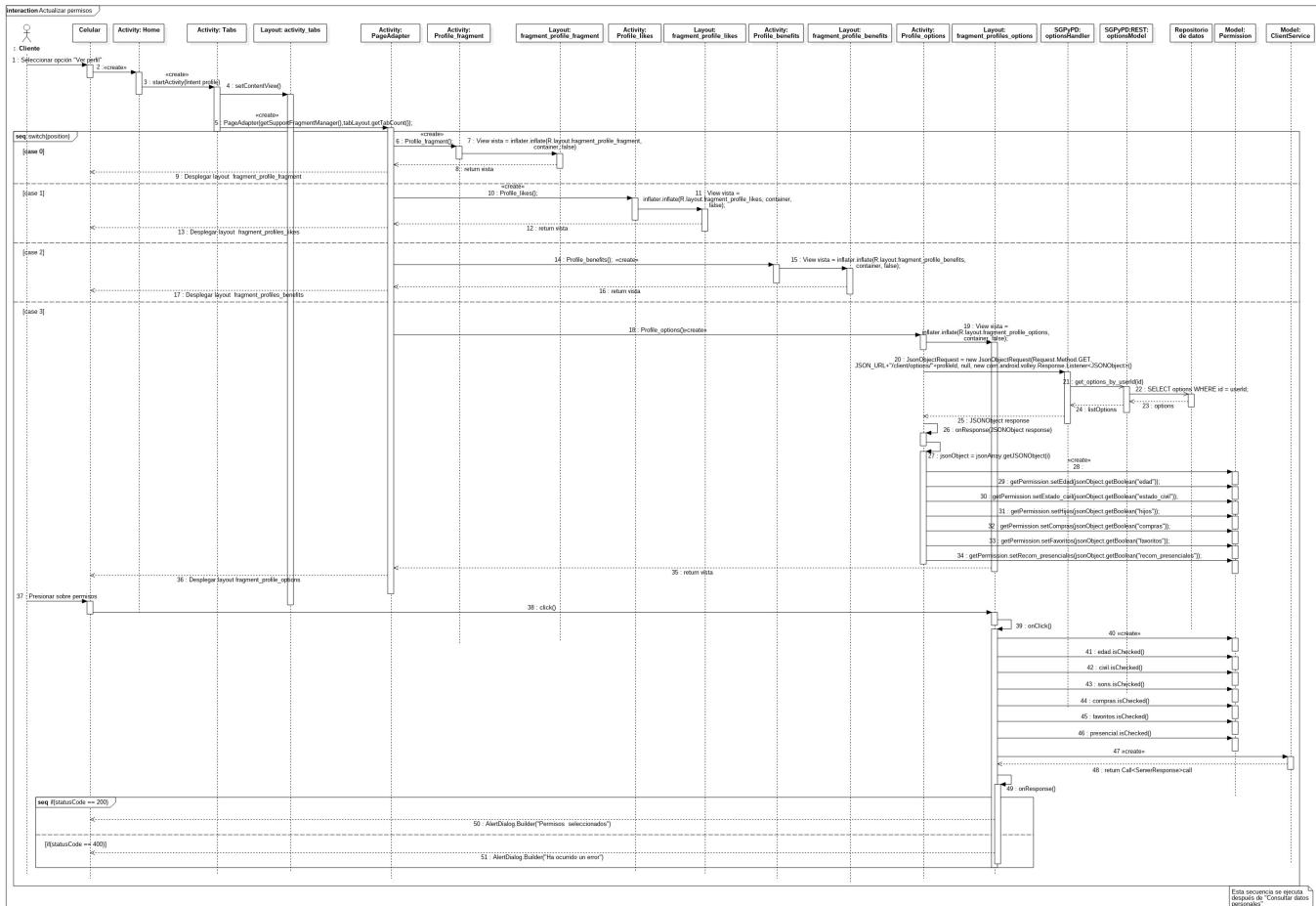


Figura 5.95: Diagrama de secuencia para actualizar permisos (Visualización completa).

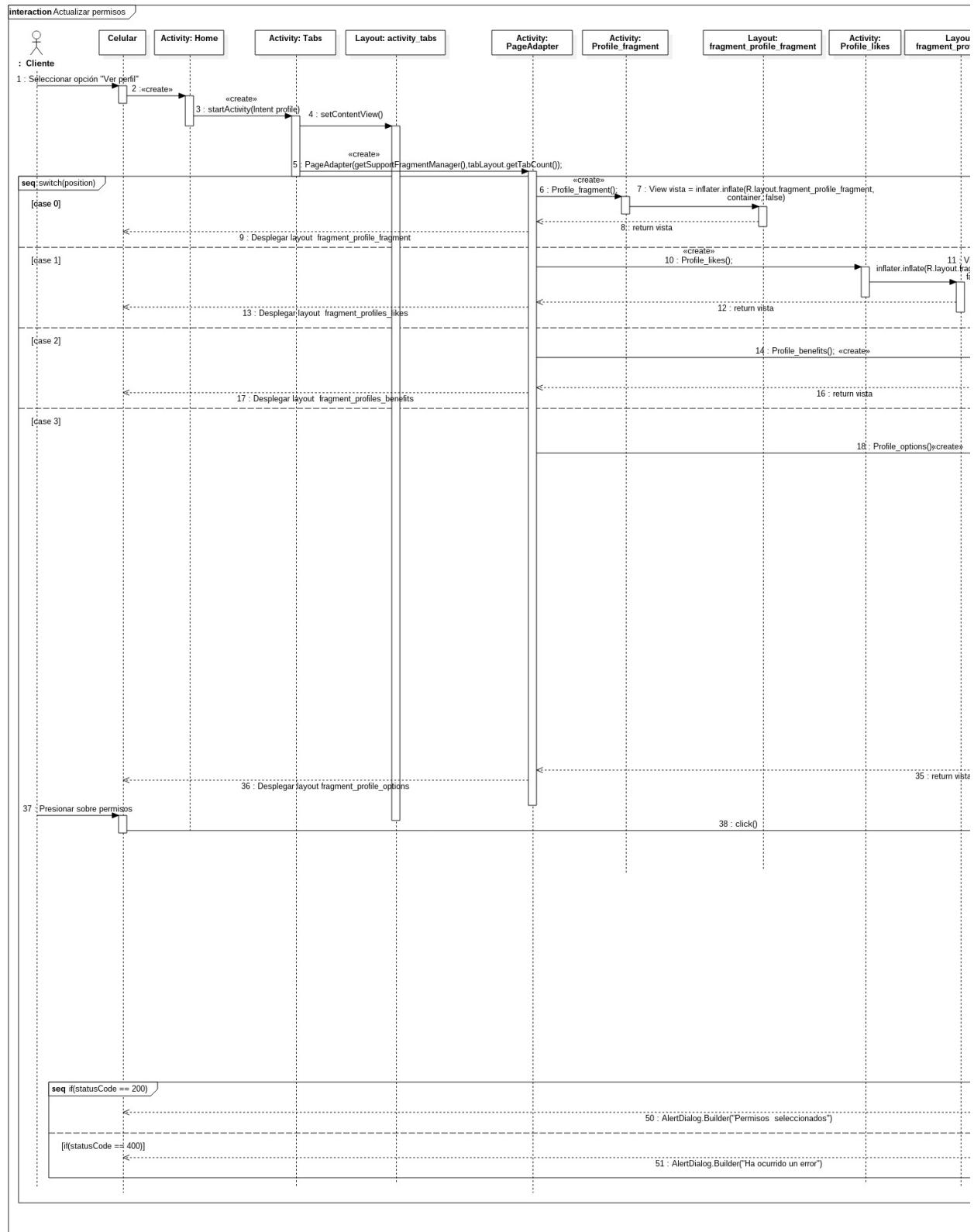


Figura 5.96: Diagrama de secuencia para actualizar permisos (Parte 1).

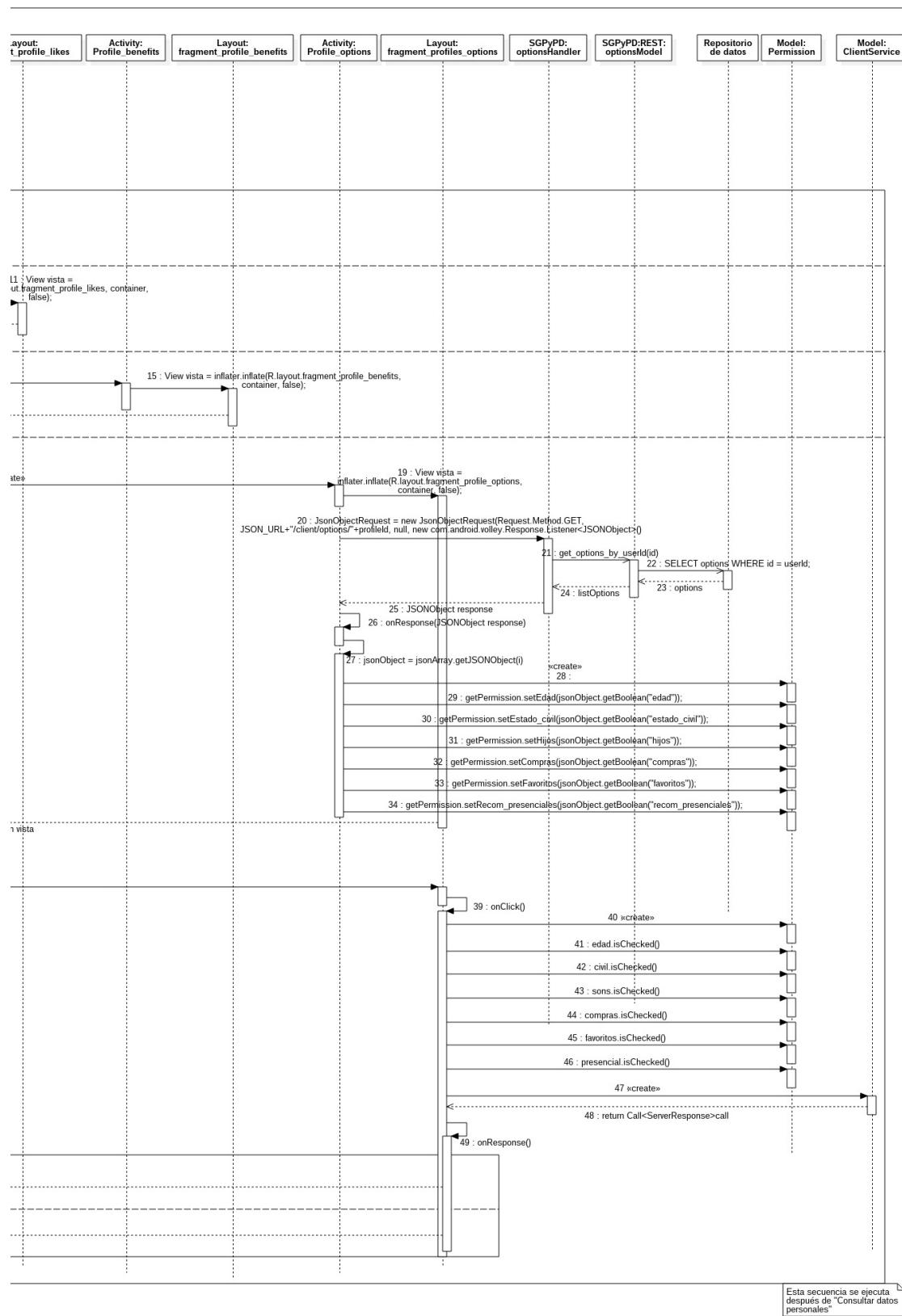


Figura 5.97: Diagrama de secuencia para actualizar permisos (Parte 2).

## Cerrar sesión

El diagrama de la figura 5.98, muestra la secuencia que se realiza para cerrar la sesión actual de Facebook.

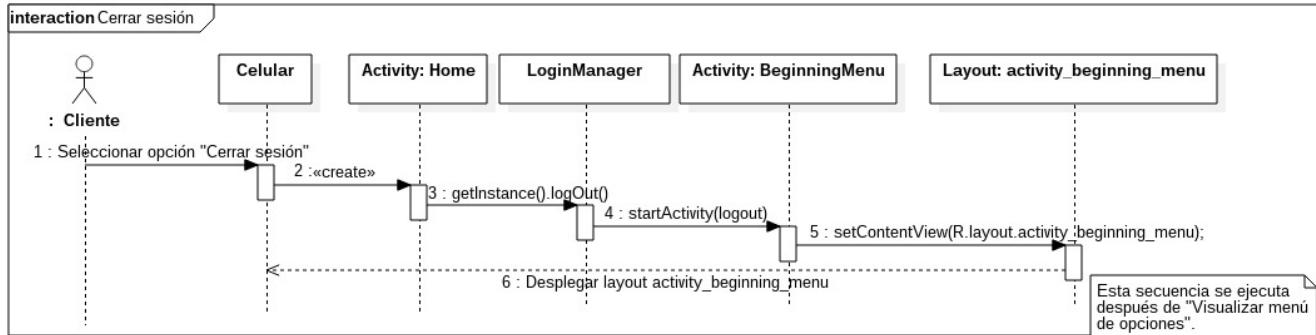


Figura 5.98: Diagrama de secuencia para cerrar sesión (Visualización completa).

### 5.1.4. Prototipo 4: Integración de módulos restantes del sistema

#### 5.1.4.1. Análisis

Dentro de este prototipo se muestra la inclusión de los servicios restantes que fueron propuestos en los requerimientos funcionales definidos previamente en el capítulo del “Bosquejo general de la aplicación” con el título de “[Requerimientos Funcionales de Aplicación Interactiva Difusora de Productos](#)”, sin embargo, es importante mencionar la inclusión del requerimiento **Solicitar apoyo a vendedor** y la eliminación el requerimiento “Consultar promociones” debido a que este era un requerimiento que ya estaba considerado como parte del requerimiento “Consultar folletos” pues dentro de dichos folletos, se muestran las promociones explicadas en el “[RFAIDP3](#)”.

Id	Nombre	Descripción
RFAIDP20	Solicitar apoyo a vendedor	Si el cliente presiona sobre la opción “Solicitar apoyo” localizada en el menú de opciones, un vendedor podrá acercarse a él para proveer la ayuda que él requiera.

Cuadro 5.2: Requerimiento añadido a los Requerimientos Funcionales de la Aplicación Interactiva Difusora de Productos.

## Casos de uso de AIDP

La figura 5.99, muestra la versión final de los casos de uso de nuestra AIDP.

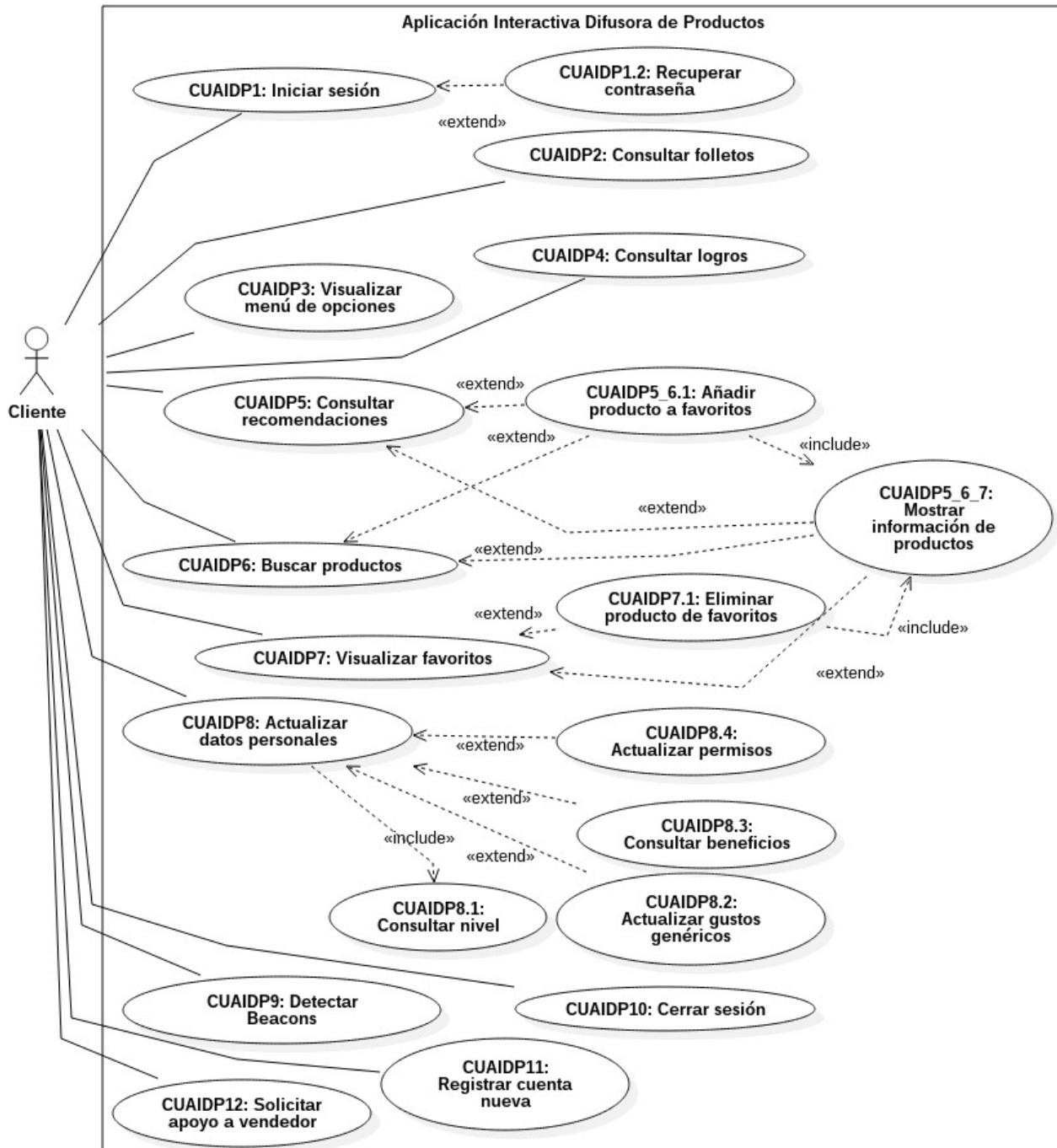


Figura 5.99: Casos de uso de la Aplicación Interactiva Difusora de Productos.

### 5.1.4.2. Diseño

#### Flujos de navegación de la AIDP

En esta sección se visualiza en las figuras 5.100 y 5.103 el mapa de navegación con la interfaz finalizada de la Aplicación Interactiva Difusora de Productos. Así mismo, también se observa la descripción de la pantalla del requerimiento agregado y descrito en la sección de [Análisis](#).

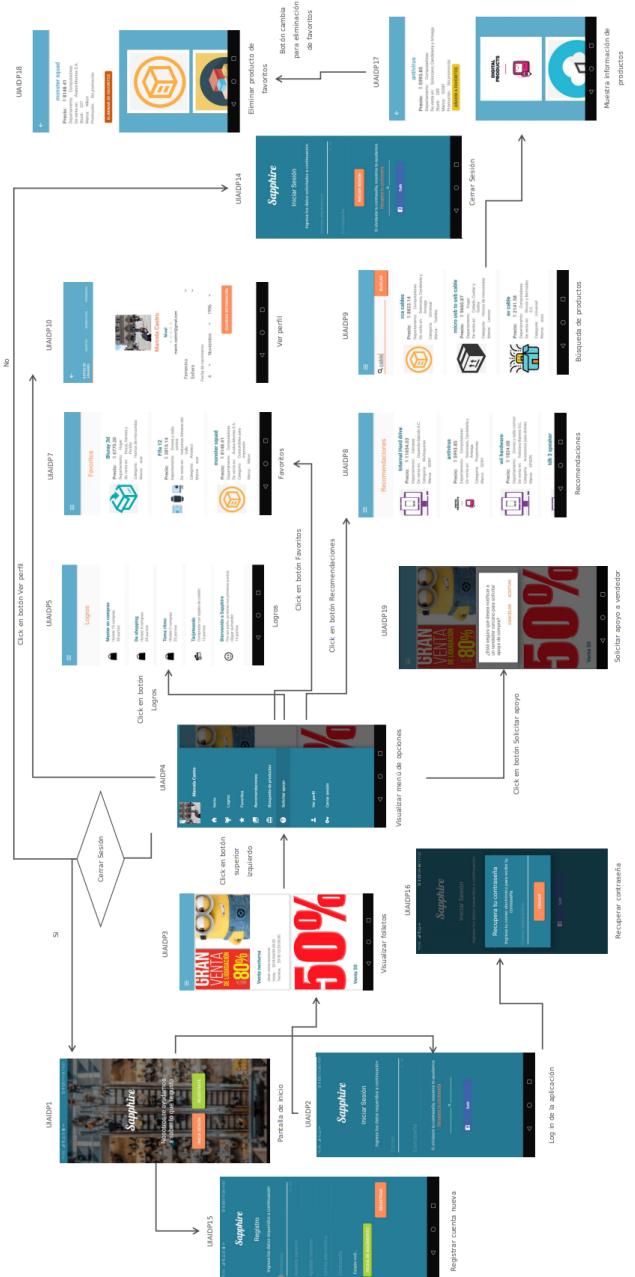


Figura 5.100: Flujo de navegación de la Aplicación Interactiva Difusora de Productos (Visualización completa).

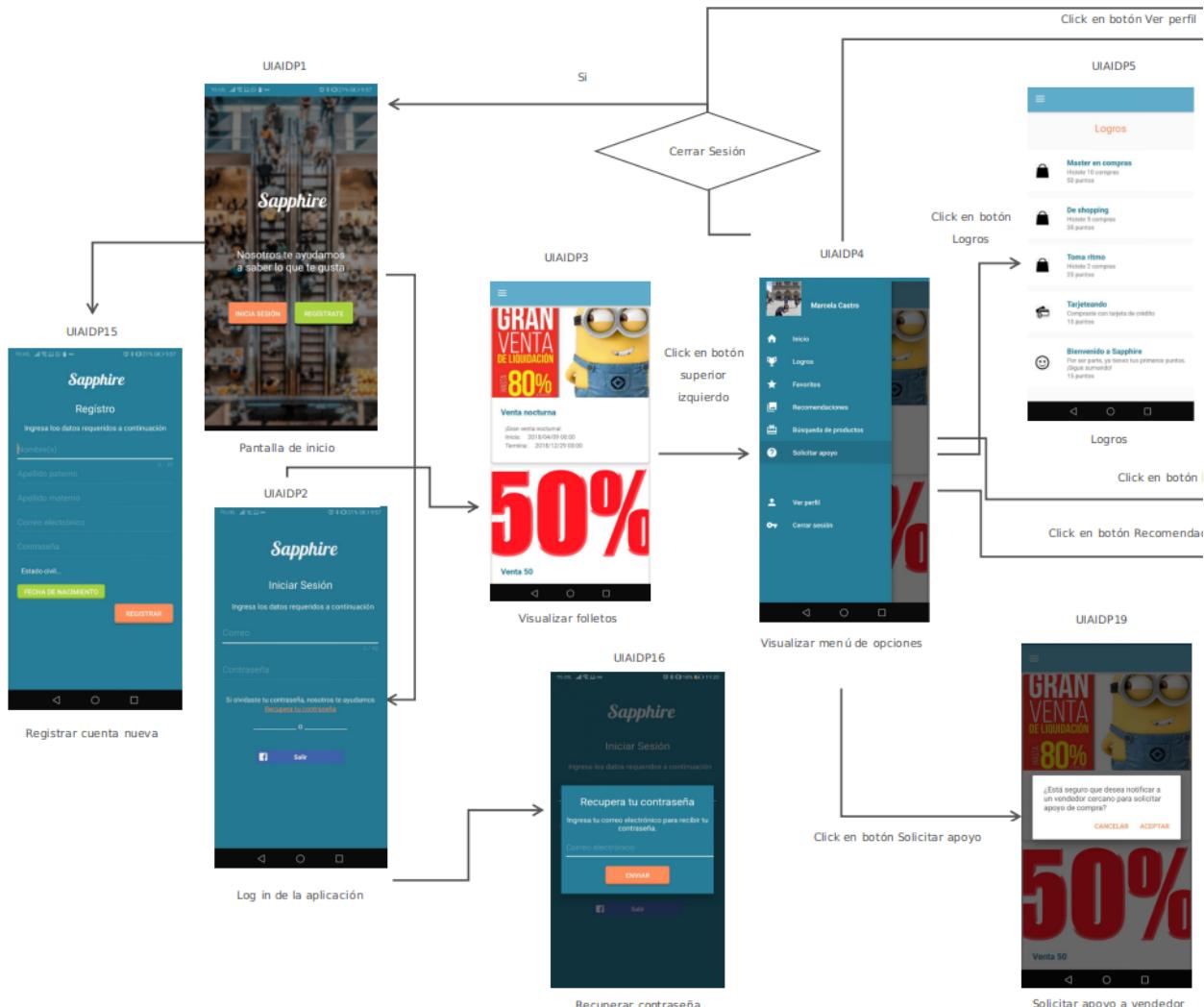


Figura 5.101: Flujo de navegación de la Aplicación Interactiva Difusora de Productos (Parte 1).



Figura 5.102: Flujo de navegación de la Aplicación Interactiva Difusora de Productos (Parte 2).

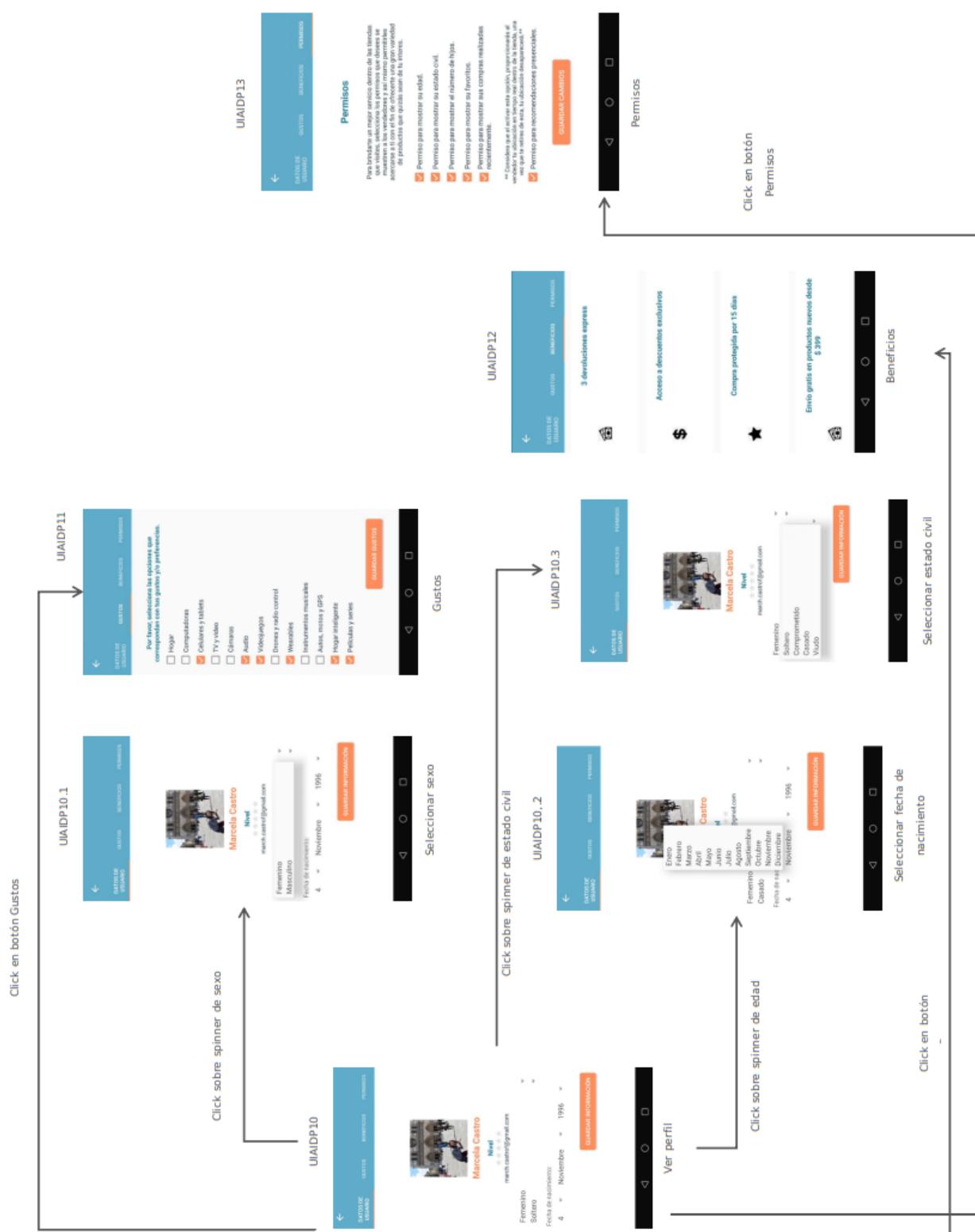


Figura 5.103: Flujo derivado de navegación de la Aplicación Interactiva Difusora de Productos (Visualización completa).



Figura 5.104: Flujo derivado de navegación de la Aplicación Interactiva Difusora de Productos (Parte 1).



Figura 5.105: Flujo derivado de navegación de la Aplicación Interactiva Difusora de Productos (Parte 2).

## UIAIDP19 - Solicitar apoyo a vendedor

### Objetivo

Proporciona la funcionalidad de notificar a un vendedor que es requerido por parte de un cliente.

### Diseño

Esta pantalla (figura 5.106), muestra el menú de opciones en el cual ha sido agregada una nueva funcionalidad en la cual el cliente al presionar sobre ella, puede solicitar al vendedor que se encuentre más cercano a él apoyo para la realización de sus compras.



Figura 5.106: UIAIDP19 Solicitar apoyo a vendedor.

### Salidas

Alerta de notificación para solicitar a un vendedor.

### Entradas

Ninguna.

*Nota: Es importante mencionar que en este prototipo únicamente se muestra en la figura 5.100 y 5.103, el mapa de navegación con las interfaces actualizadas, sin embargo, únicamente se realiza la descripción de la pantalla del requerimiento **Solicitar apoyo a vendedor**, ya que a pesar de que las nuevas interfaces mostradas en el mapa de navegación han cambiado, la funcionalidad de cada pantalla mostrada sigue manteniéndose con respecto a lo planificado en los re-*

querimientos. Cada pantalla se explica en el “[Prototipo 1: Diseño inicial de la aplicación](#)” en la descripción de pantallas.

Por otra parte, en el prototipo presente, se añaden los últimos servicios que corresponden al inicio de sesión con una cuenta registrada en Sapphire y a solicitar apoyo a un vendedor.

### Diagramas de secuencia

El diagrama mostrado en la figura 5.107, se observa el procedimiento con el cual el cliente en caso de requerir ayuda por parte del personal de la tienda, puede notificar al vendedor más cercano a él con el fin de que este acuda a ayudarlo.

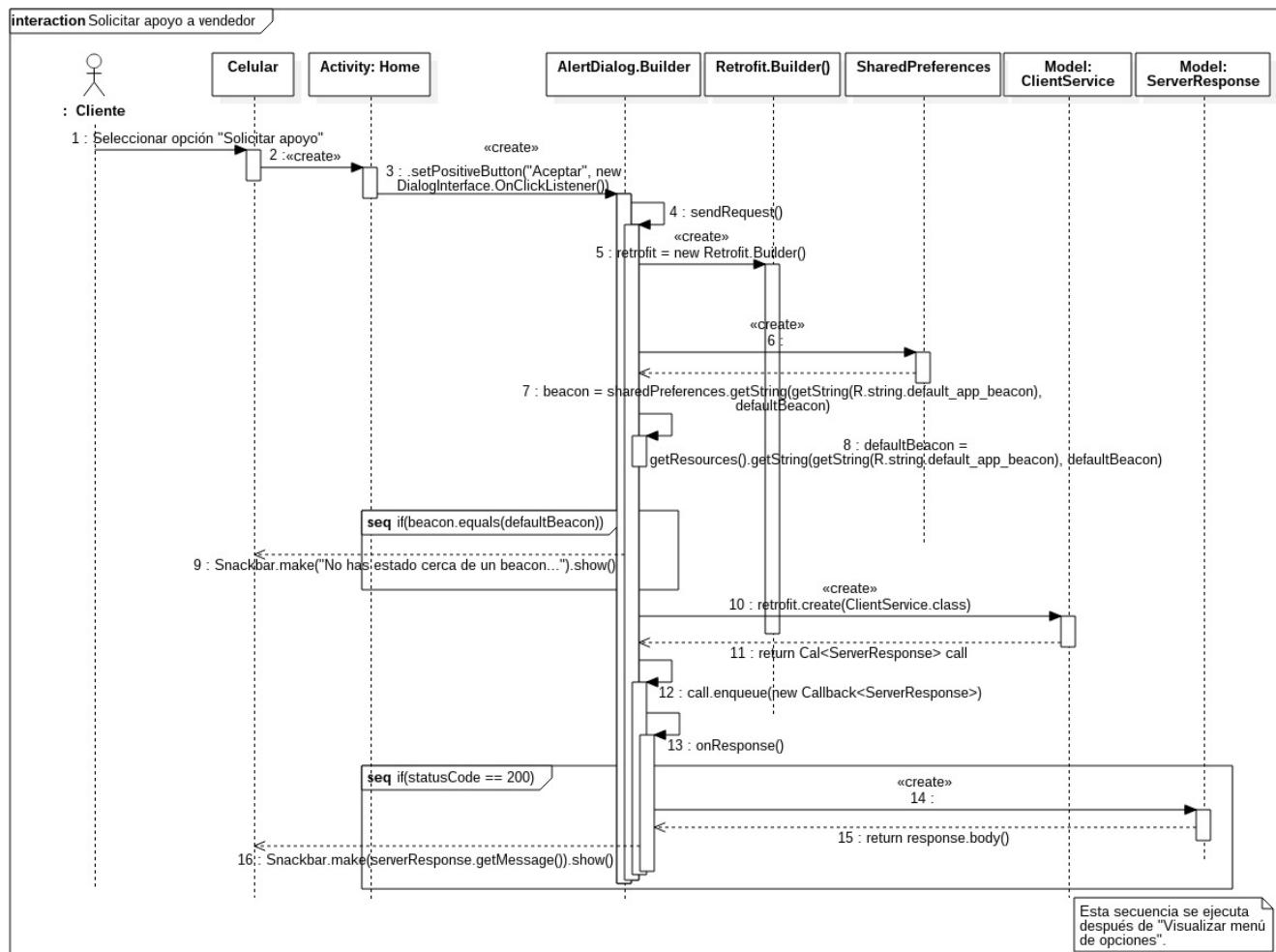


Figura 5.107: Diagrama de secuencia de solicitar apoyo a vendedor (Visualización completa).

En este diagrama (figura 5.108), se observa la obtención de datos desde el servidor con el fin de desplegar una foto de perfil al usuario y su nombre.

El diagrama de la figura 5.108, fue dividido en dos secciones visualizadas en las figuras 5.109 y 5.110 con el fin de obtener una mejor visualización.

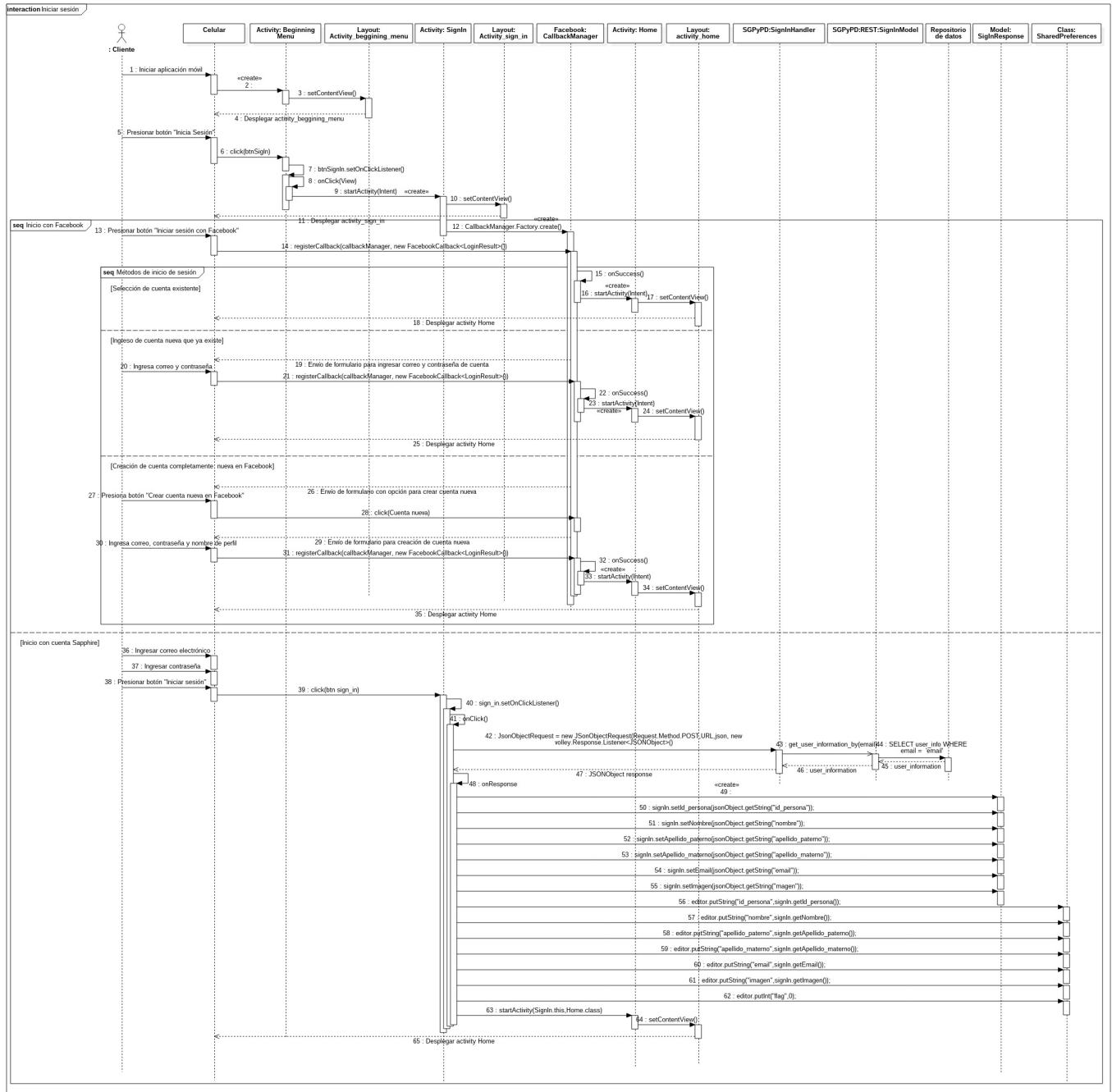


Figura 5.108: Diagrama de secuencia de inicio de sesión con cuenta de Sapphire (Visualización completa).

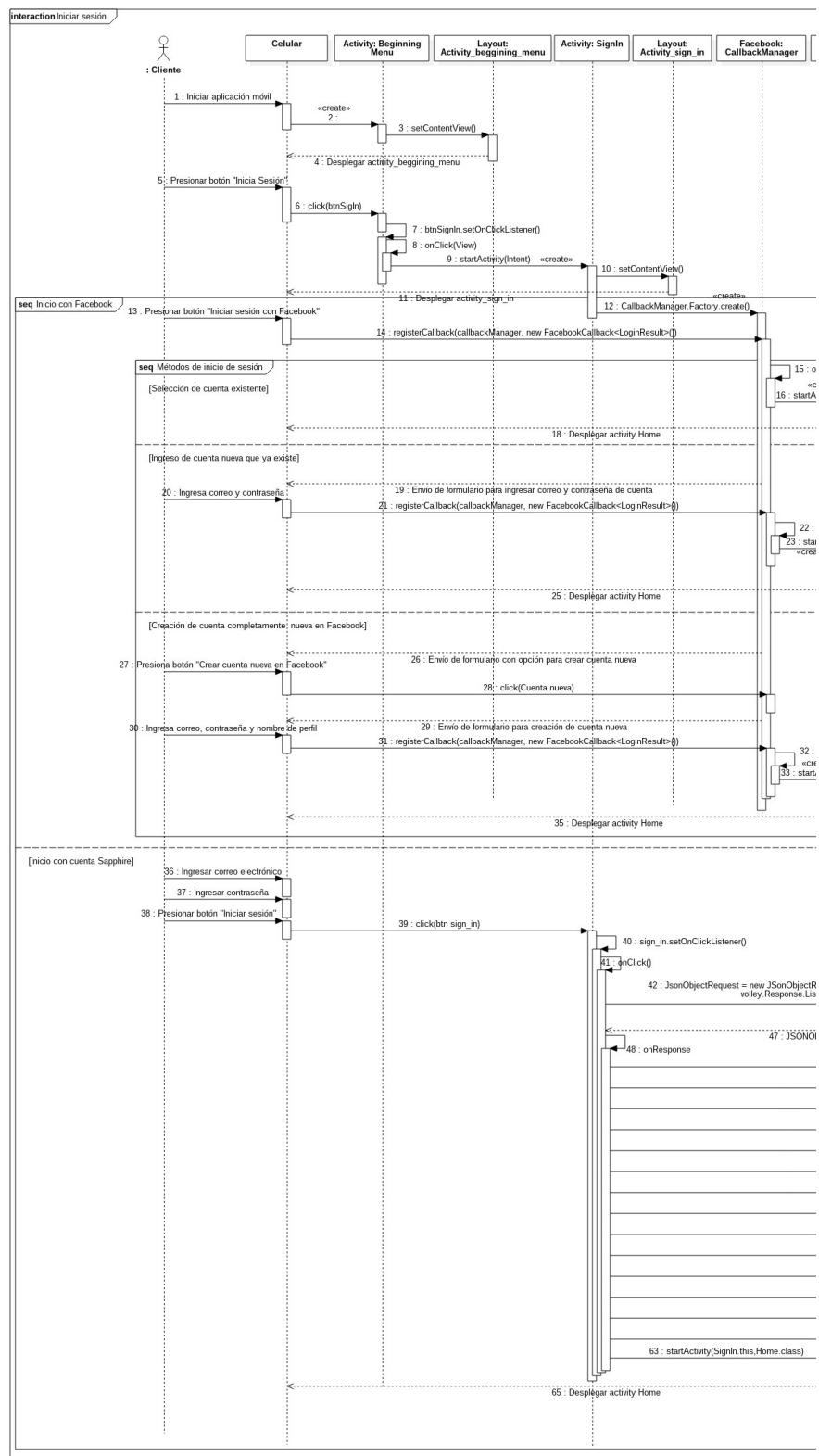


Figura 5.109: Diagrama de secuencia de inicio de sesión con cuenta de Sapphire (Parte 1).

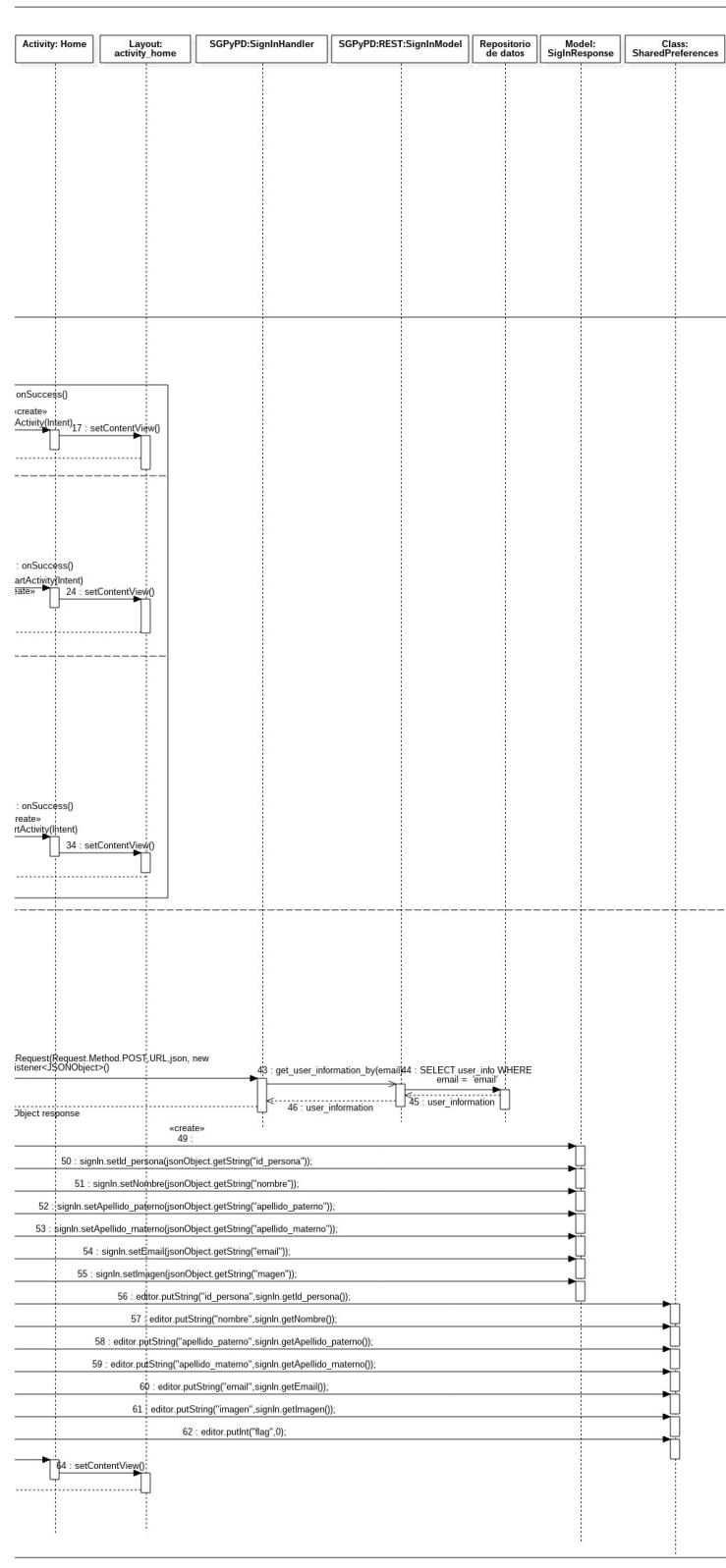


Figura 5.110: Diagrama de secuencia de inicio de sesión con cuenta de Sapphire (Parte 2).

## 5.2. Aplicación Interactiva para el Personal de Ventas (AIPV)

La Aplicación Interactiva para el Personal de Ventas tiene como objetivo ser un apoyo al vendedor para conocer más sobre las preferencias de un cliente, poder otorgarle una atención personalizada y/o asistirlo en la elección de la compra de un producto.

La figura 5.111 presenta la arquitectura de este módulo.

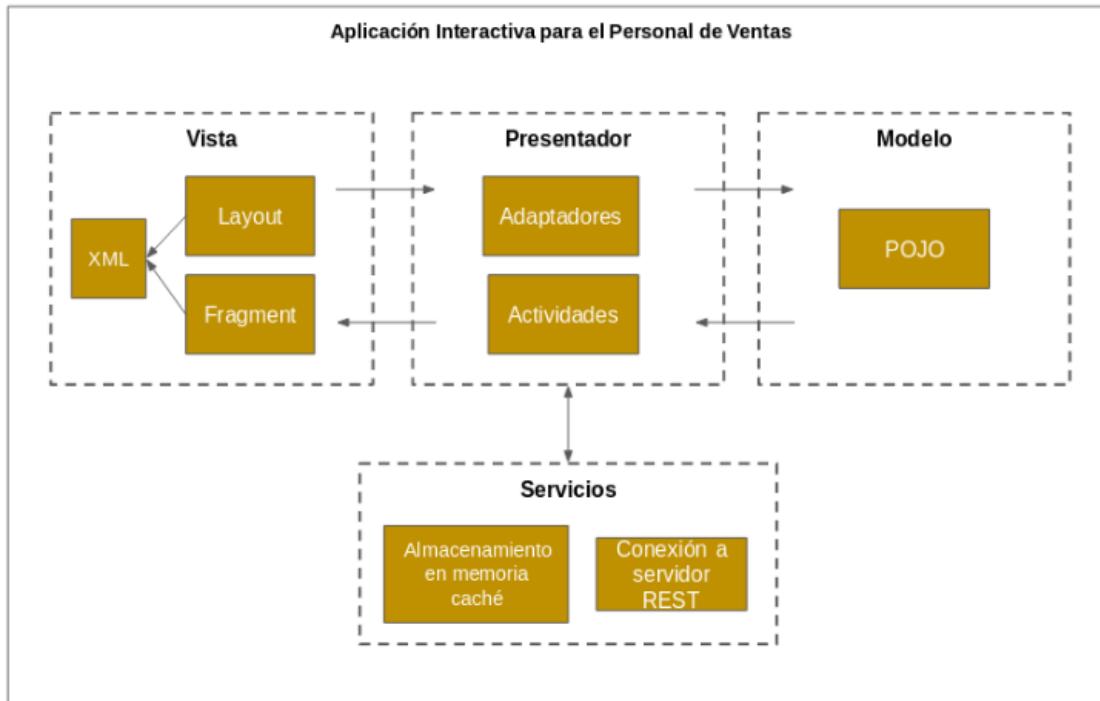


Figura 5.111: Arquitectura del módulo AIPV.

### 5.2.1. Prototipo 1: Diseño inicial de la aplicación y descubrimiento de Beacons

El prototipo 1 de la aplicación incluye el diseño inicial de la misma, así como el descubrimiento de los Beacons dentro del comercio, con el fin de mantener un registro de donde se encuentran. Para ello Estimote nos permite crear zonas de proximidad mediante claves - valor que se establecen en el Beacon, cuando el usuario vendedor entra en una de estas zonas de proximidad la aplicación envía la ubicación para registrar las coordenadas actuales del Beacon.

#### 5.2.1.1. Análisis

Dentro del análisis para el desarrollo de este prototipo se incorporan los requerimientos funcionales RFAPV1 Iniciar sesión y RFAPV2 Enviar de ubicación de Beacons, definidos previamente en el capítulo del “Bosquejo general de la aplicación” con el título de “Requerimientos funcionales de la Aplicación Interactiva para el Personal de Ventas”.

#### Casos de uso de la AIPV.

La figura 5.112 muestra los casos de uso del prototipo 1 de la AIPV.

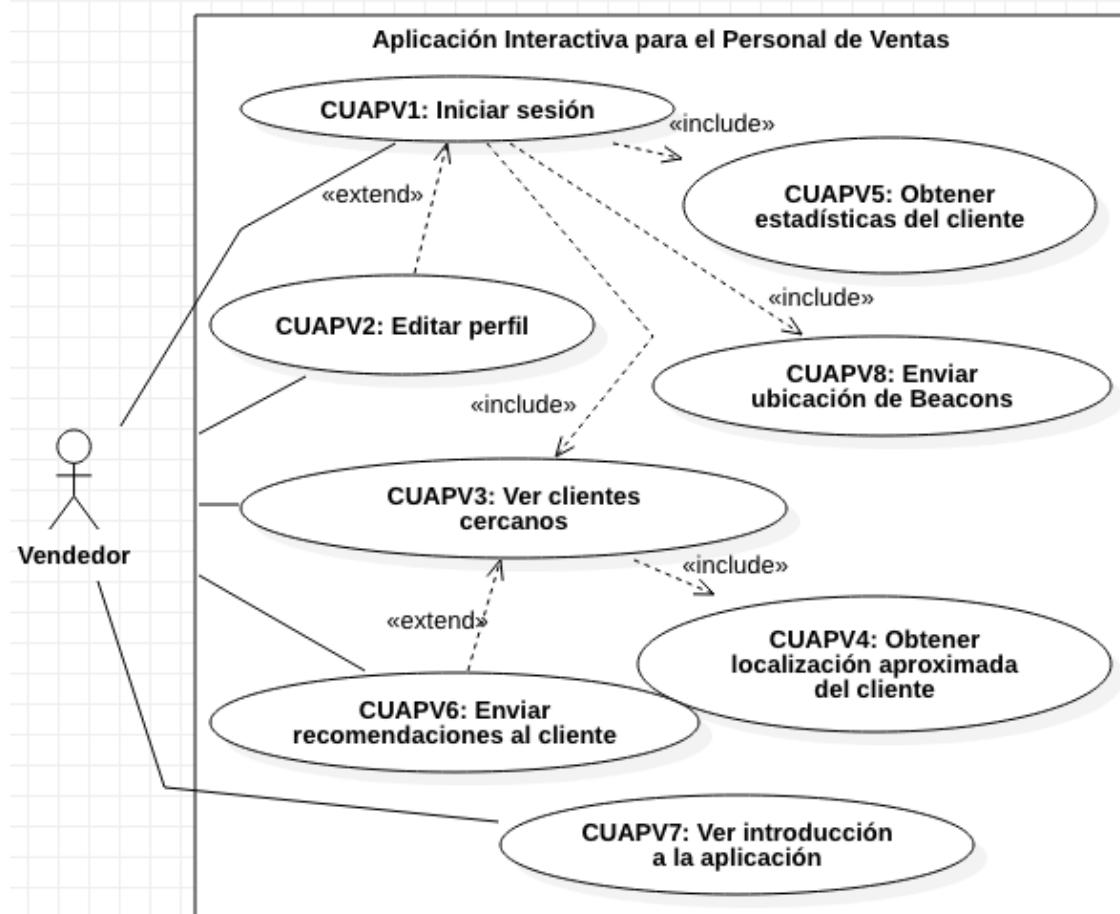


Figura 5.112: Casos de uso de la AIPV.

## Diagrama de clases.

La figura 5.113 presenta el diagrama de clases para el prototipo 1 de la AIPV. Para una mejor visualización, el diagrama se ha dividido en 3 partes.

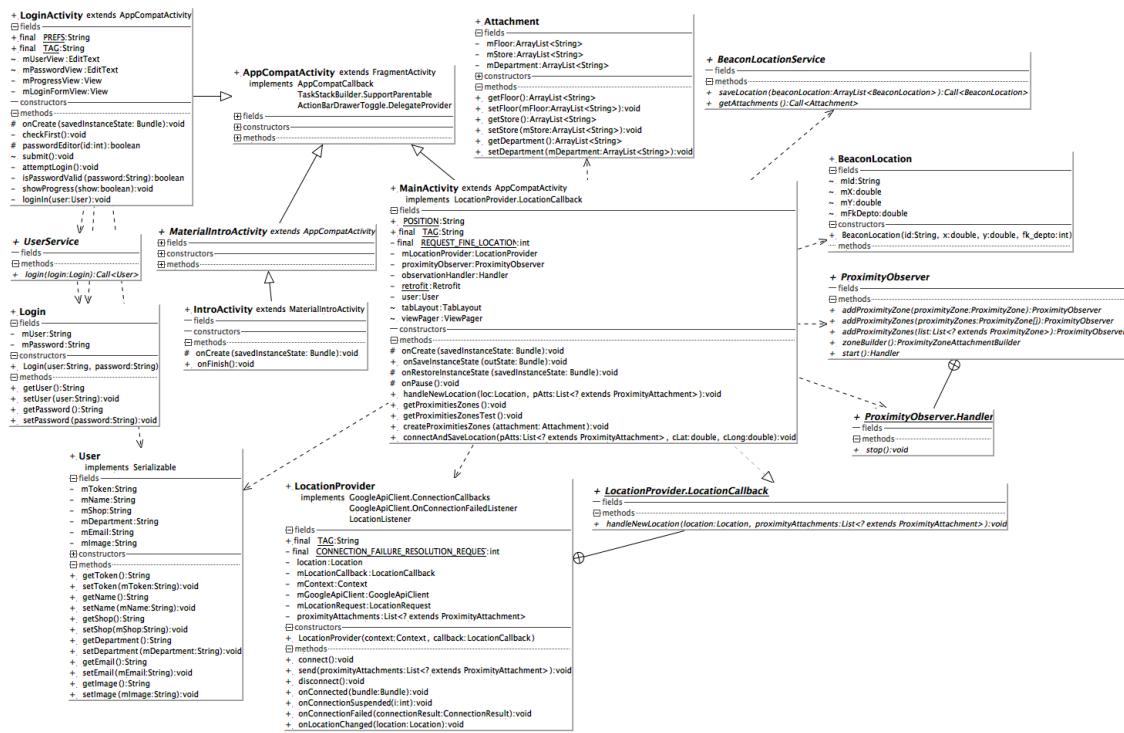


Figura 5.113: Diagrama de clases del prototipo 1 de la AIPV (Visualización completa).

La descripción de los elementos de la parte 1 del diagrama de clases es la siguiente:

- **AppCompatActivity**: Clase base para actividades que utilizan las características de la barra de acciones de la biblioteca de soporte de Android.
- **LoginActivity**: Clase encargada del inicio de sesión, solicita el nombre de usuario y contraseña al vendedor, utiliza la interfaz UserService para hacer una petición POST al servidor y comprobar que los datos sean correctos para un inicio de sesión exitoso.
- **UserService**: Interfaz que define el método login, esta interfaz utiliza retrofit que nos permite consumir la API REST en la aplicación.
- **Login**: Clase que retrofit convierte y coloca en el cuerpo de una petición HTTP POST, contiene el nombre de usuario y contraseña proporcionados por el vendedor.
- **IntroActivity**: Clase encargada de mostrar una pequeña introducción de lo que es posible realizar el vendedor desde la aplicación. Extiende de la clase MaterialIntroActivity.
- **MainActivity**: Clase principal de la aplicación, en ella se muestra la lista de usuarios cercanos y el fragmento para editar el perfil del vendedor. Implementa la interfaz LocationProvider.LocationCallback para obtener la localización de un Beacon al entrar a una zona de proximidad, además de usar la interfaz BeaconLocationService para hacer una petición GET y obtener las claves - valor necesarias para generar las zonas de proximidad, también se utiliza la misma interfaz para hacer una petición POST al servidor y guardar la ubicación.

- **Attachment:** Clase que retrofit mapea, es la respuesta de una petición HTTP GET en formato JSON, contiene arreglos de String para generar zonas de proximidad por pisos, tiendas o departamentos.

La figura 5.114 contiene las clases previamente descritas.

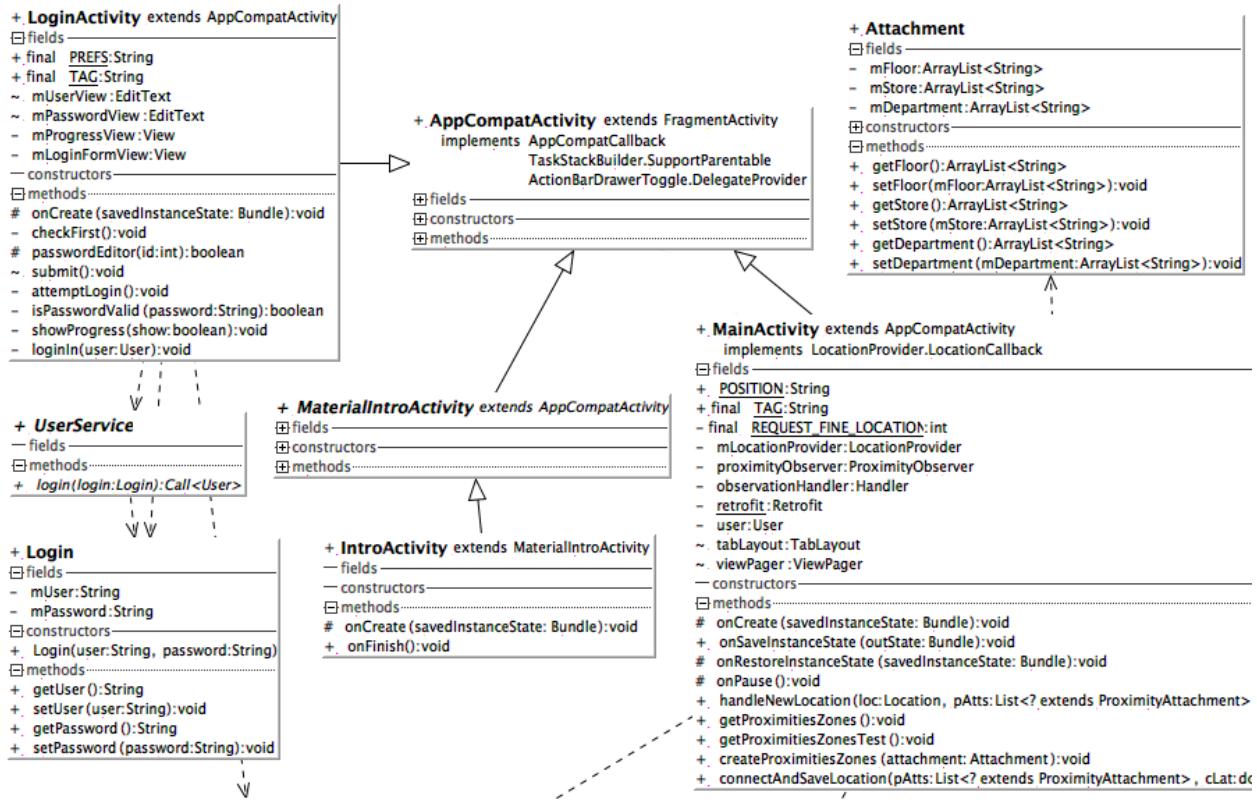


Figura 5.114: Diagrama de clases del prototipo 1 de la AIPV (Parte 1) .

La descripción de los elementos de la parte 2 del diagrama de clases es la siguiente:

- **BeaconLocationService:** Interfaz que define el método saveLocation para guardar la ubicación actual de Beacons y el método getAttachments que obtiene las claves - valor necesarias para generar las zonas de proximidad, esta interfaz utiliza retrofit que nos permite consumir la API REST en la aplicación.
- **BeaconLocation:** Clase en la que se guarda el id de un Beacon y su ubicación actual, se envía en un ArrayList el cual retrofit convierte y coloca en el cuerpo de una petición HTTP POST.
- **ProximityObserver:** Interfaz de Estimote que define los métodos para agregar zonas de proximidad en la aplicación y comenzar el descubrimiento de Beacons.
- **LocationProvider.LocationCallback:** Interfaz que define el método handleNewLocation que se encarga de vincular una lista de Beacons con una localización.

La figura 5.115 contiene las clases previamente descritas.

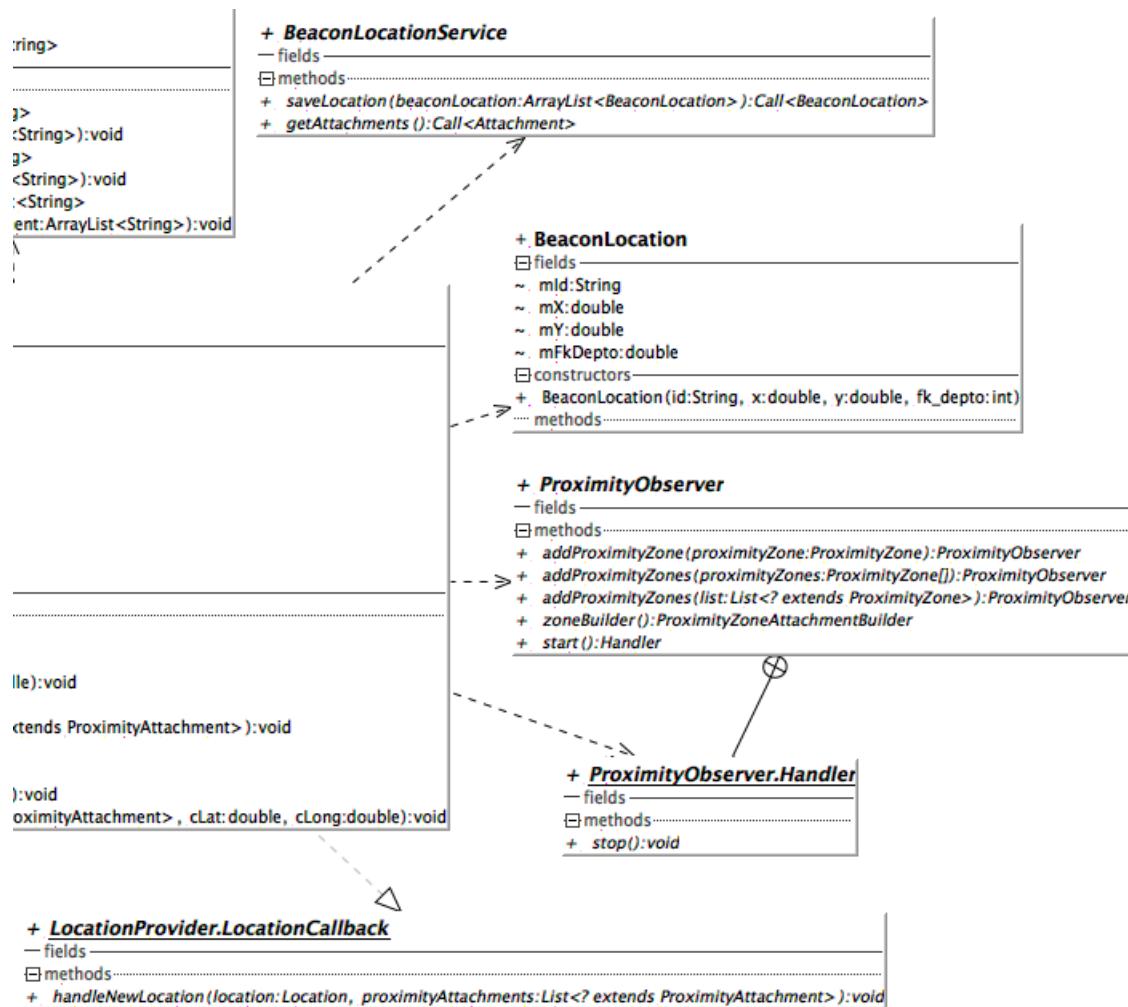


Figura 5.115: Diagrama de clases del prototipo 1 de la AIPV (Parte 2).

La descripción de los elementos de la parte 3 del diagrama de clases es la siguiente:

- **User:** Clase que retrofit mapea, es la respuesta de la petición HTTP POST para el inicio de sesión la cual está en formato JSON, contiene la información básica del usuario vendedor.
- **LocationProvider:** Clase que obtiene la geolocalización del dispositivo móvil mediante la clase LocationRequest de Google Play Services.

La figura 5.116 contiene las clases previamente descritas.

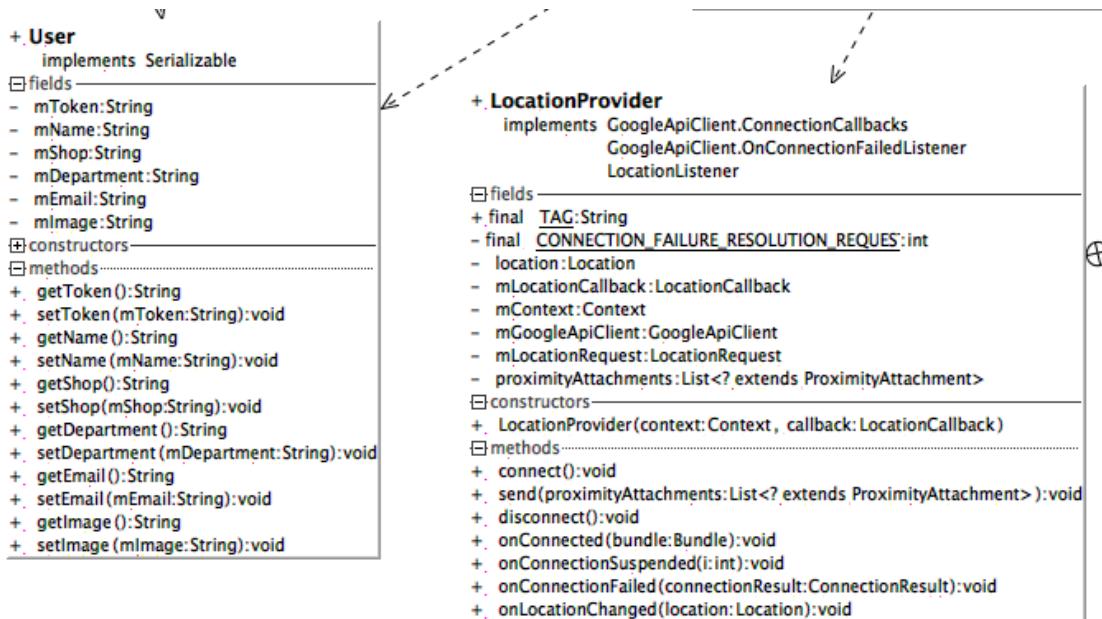


Figura 5.116: Diagrama de clases del prototipo 1 de la AIPV (Parte 3).

### 5.2.1.2. Diseño

A partir de los requerimientos definidos para este prototipo se muestran los casos de uso, diagramas de secuencia, el flujo de navegación y las interfaces de la aplicación.

#### Diagramas de secuencia

##### Iniciar sesión.

En la figura 5.117 se muestra el diagrama de secuencia para iniciar sesión en la AIPV, el cual describe los pasos que se llevan a cabo para que el usuario vendedor inicie sesión satisfactoriamente.

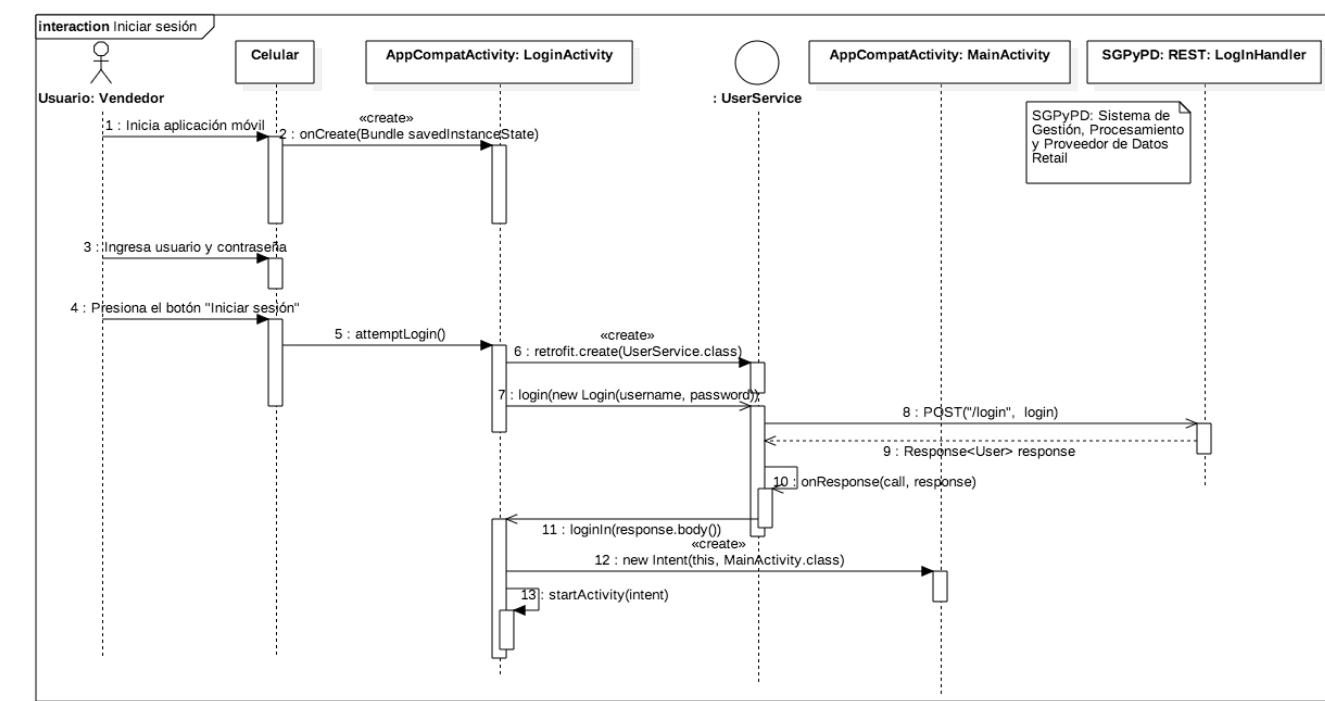


Figura 5.117: Diagrama de secuencia para el inicio de sesión de un vendedor.

Ver introducción a la aplicación.

En la figura 5.118 se muestra el diagrama de secuencia para ver una introducción a la AIPV, es decir, mostrarle al usuario vendedor lo que es posible realizar dentro de la aplicación.

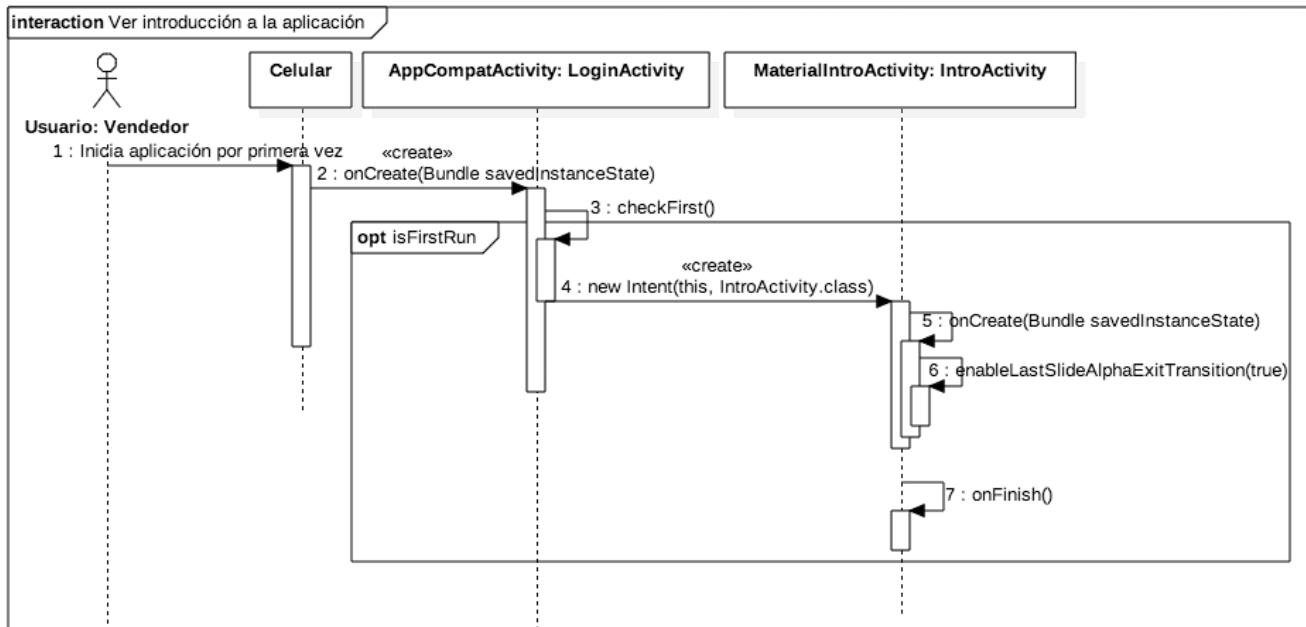


Figura 5.118: Diagrama de secuencia para ver una introducción a la aplicación.

### Enviar ubicación de Beacons.

En la figura 5.119 se muestra el diagrama de secuencia para enviar la ubicación de los Beacons desde la AIPV, el cual describe los pasos que se llevan a cabo para que la aplicación cree las zonas de proximidad y pueda detectar correctamente los Beacons, para una mejor visualización el diagrama se ha dividido en 2 partes las cuales se muestran en las figuras 5.120 y 5.121.

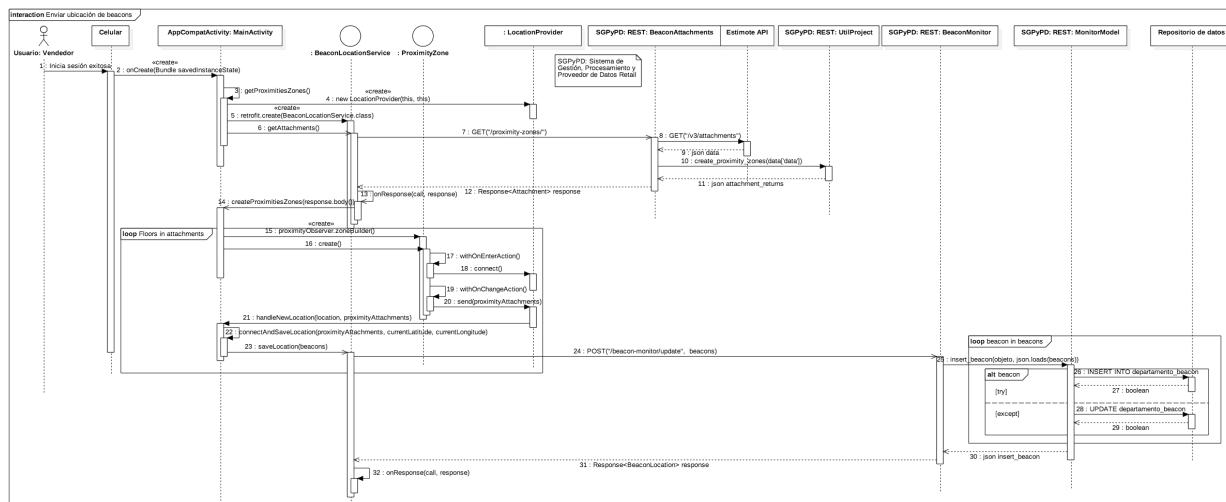


Figura 5.119: Diagrama de secuencia para enviar la ubicación de Beacons (Visualización completa).

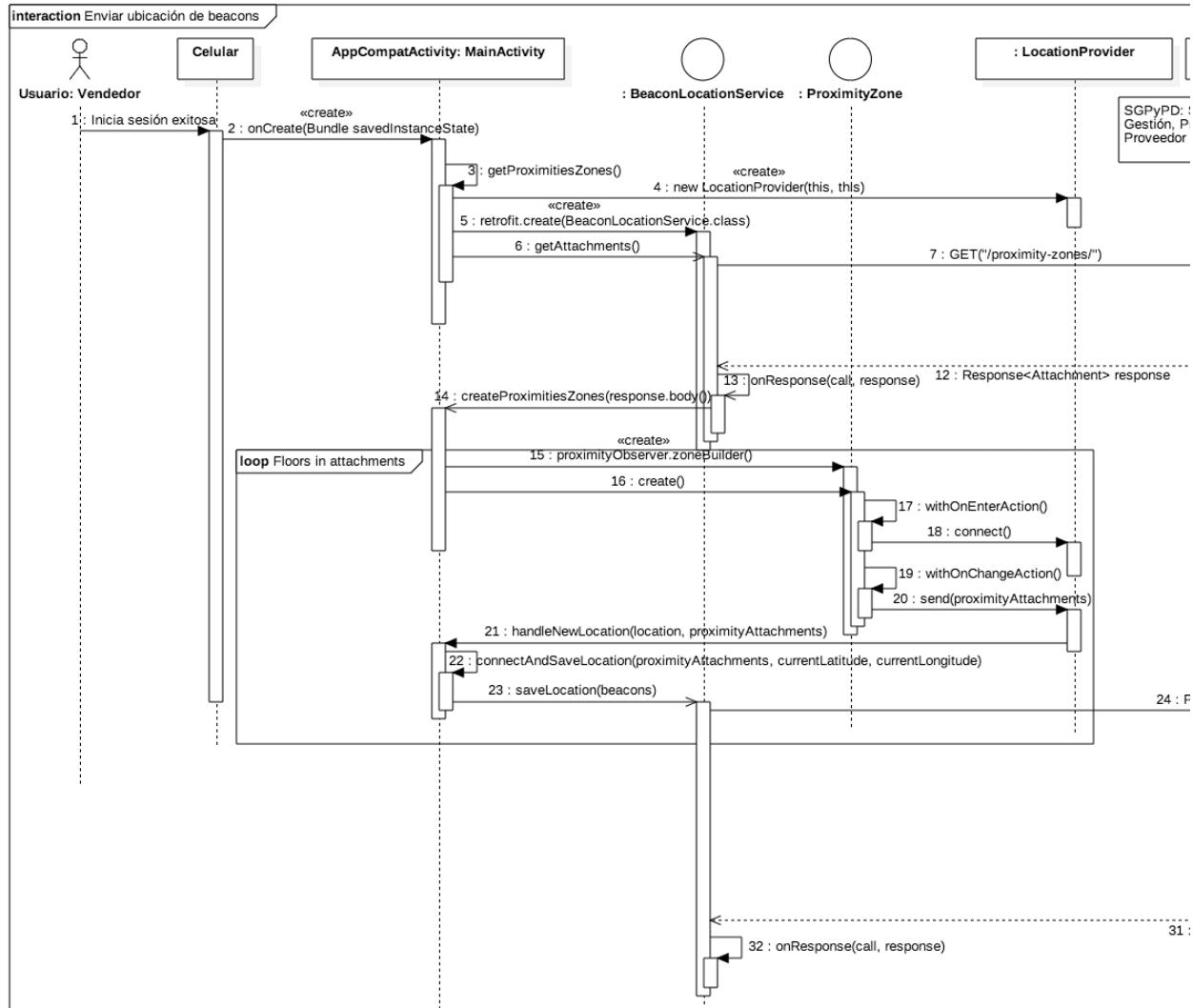


Figura 5.120: Diagrama de secuencia para enviar la ubicación de Beacons (Parte 1).

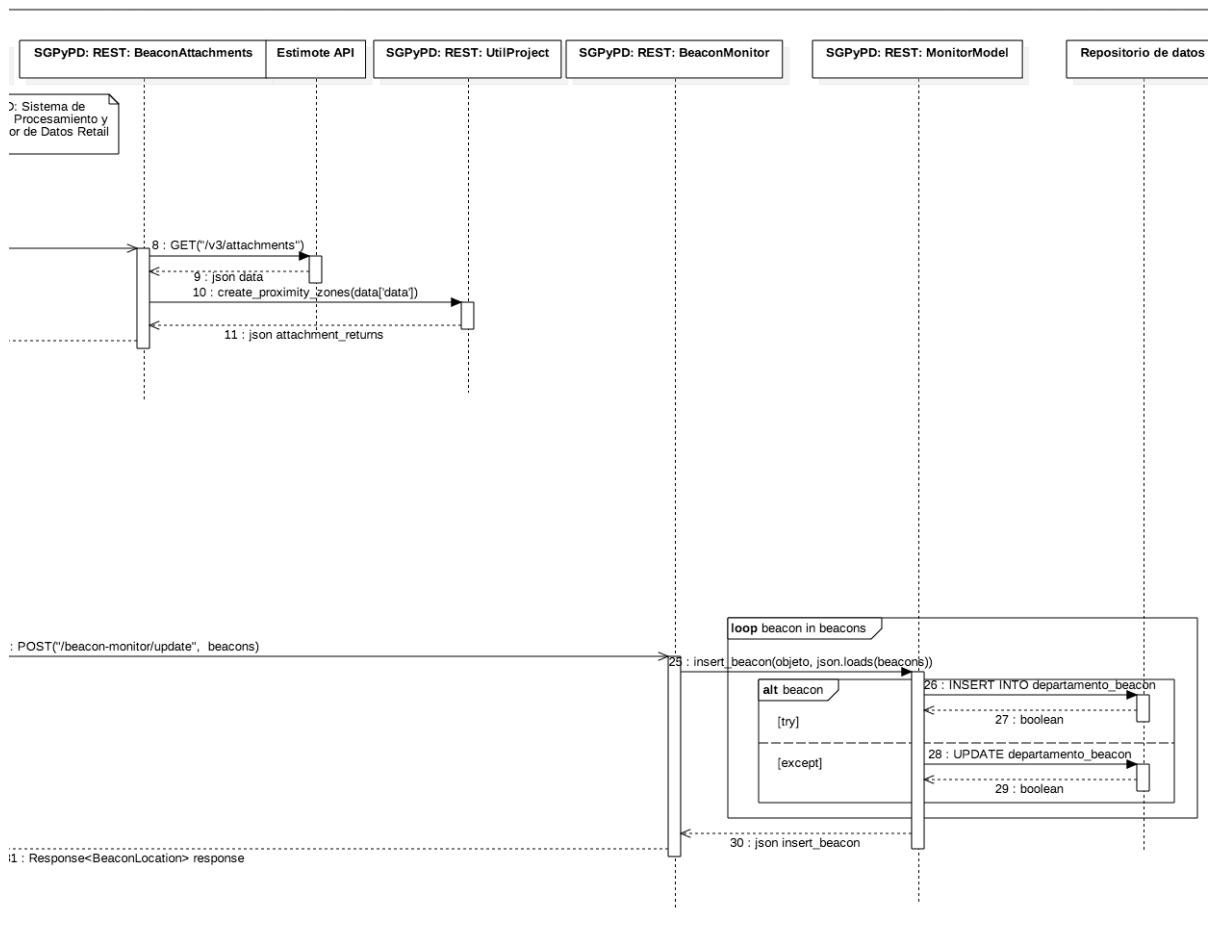


Figura 5.121: Diagrama de secuencia para enviar la ubicación de Beacons (Parte 2).

### Flujo de navegación de la AIPV.

La siguiente figura 5.122 muestra como es el flujo de navegación de la aplicación.

#### Interfaces de usuario

##### UIAPV1 - Login

##### Objetivo

Controlar el acceso al sistema mediante un usuario y contraseña a fin de que a los usuarios vendedores se les muestre un listado de compradores cercanos para ofrecerles una atención personalizada.

##### Diseño

La figura 5.123 presenta la pantalla que aparece al iniciar la aplicación. Para lograr un inicio de sesión exitoso se debe ingresar el nombre de usuario y contraseña correctas.

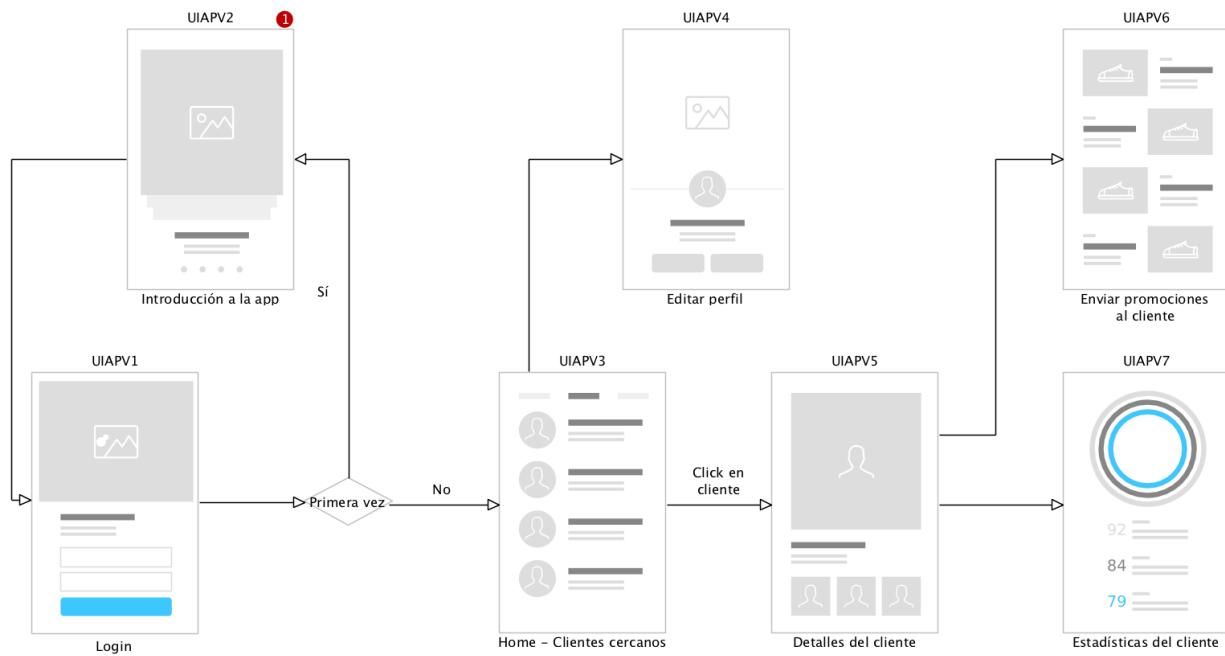


Figura 5.122: Flujo de navegación de la AIPV.



Figura 5.123: UIAPV1 Login.

### Salidas

Ninguna.

## Entradas

Nombre de usuario y contraseña.

## UIAPV2 - Introducción a la aplicación

### Objetivo

Informar al usuario sobre lo que puede hacer desde la aplicación.

### Diseño

La figura 5.124 presenta la pantalla que aparece únicamente la primera vez que se inicia la aplicación.



Figura 5.124: UIAPV2 Introducción a la aplicación.

## Salidas

Ninguna.

## Entradas

Ninguna.

## UIAPV3 - Home: Clientes cercanos

### Objetivo

Mostar al usuario vendedor una lista de los clientes potenciales cercanos para ofrecerles una atención personalizada.

### Diseño

La figura 5.125 presenta la pantalla principal que se muestra después de haber iniciado correctamente sesión, en la parte superior tiene dos opciones, comenzando de izquierda a derecha: Ver clientes y Editar perfil.



Figura 5.125: UIAPV3 Home: Clientes cercanos.

### Salidas

Ninguna.

### Entradas

Ninguna.

### UIAPV4 - Editar perfil

### Objetivo

Permitir al usuario vendedor editar sus datos básicos (nombre, apellidos, email y contraseña).

### Diseño

La figura 5.126 presenta la pantalla que se muestra al seleccionar la segunda opción del menú superior de la aplicación (de izquierda a derecha).



Figura 5.126: UIAPV4 Editar perfil.

### Salidas

Ninguna.

### Entradas

Nombre(s), apellidos, email y contraseña.

### UIAPV5 - Detalles del cliente

#### Objetivo

Mostrar información relevante del cliente que permita ofrecerle una atención personalizada, además de permitirle enviar recomendaciones del departamento en el que se encuentre.

#### Diseño

La figura 5.127 presenta la pantalla que aparece al seleccionar un cliente de la lista de clientes cercanos.

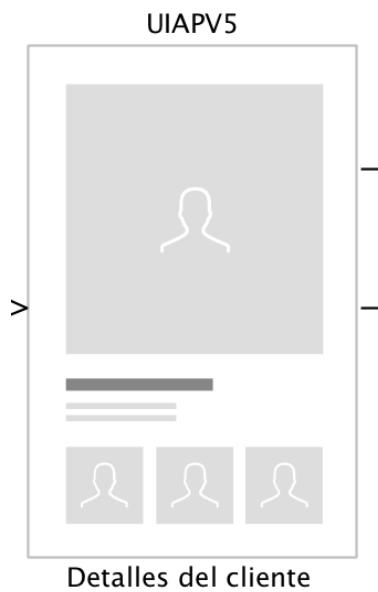


Figura 5.127: UIAPV5 Detalles del cliente.

**Salidas**

Ninguna.

**Entradas**

Ninguna.

**UIAPV6 - Enviar recomendaciones al cliente**

**Objetivo**

Permite al usuario vendedor enviar a un cliente una recomendación en particular del departamento en el que se encuentre.

**Diseño**

La figura 5.128 presenta la pantalla que se muestra al seleccionar la opción “Enviar promoción” desde el detalle de un cliente.



Figura 5.128: UIAPV6 Enviar recomendaciones al cliente.

**Salidas**

Ninguna.

**Entradas**

Ninguna.

**UIAPV7 - Estadísticas del cliente**

**Objetivo**

Mostrar al usuario vendedor las estadísticas de compra de un cliente.

**Diseño**

La figura 5.129 presenta la pantalla que se muestra al seleccionar la opción “Estadísticas del cliente” desde el detalle de un cliente.

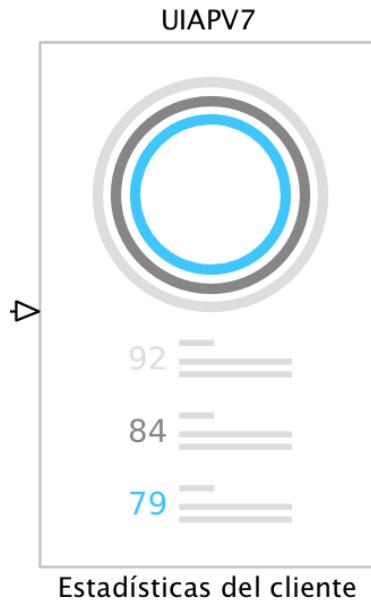


Figura 5.129: UIAPV7 Estadísticas del cliente.

#### **Salidas**

Ninguna.

#### **Entradas**

Ninguna.

### **5.2.2. Prototipo 2: Integración con Apache Kafka**

El prototipo 2 de la aplicación incluye las modificaciones en el diseño inicial de la aplicación para mostrar a los clientes que se hayan conectado a un Beacon que pertenece al departamento actual donde se encuentra el vendedor y la integración de la aplicación con los servicios REST para obtener la información de los mismos. De igual forma dentro de este prototipo se incluye el diseño para que el vendedor edite sus datos de perfil y su respectiva integración con el servidor para hacer la persistencia de los datos.

#### **5.2.2.1. Análisis**

Dentro del análisis para el desarrollo de este prototipo se incorporan los requerimientos funcionales [RFAPV3 Editar perfil del vendedor](#) y [RFAPV2 Visualizar clientes cercanos](#), definidos previamente en el capítulo del “Bosquejo general de la aplicación” con el título de “Requerimientos funcionales de la Aplicación Interactiva para el Personal de Ventas”.

#### **Casos de uso de la AIPV.**

La figura [5.112](#) definida previamente en el prototipo 1, muestra los casos de uso de la AIPV.

## Diagrama de clases.

La figura 5.130 presenta el diagrama de clases para el prototipo 2 de la AIPV. Para una mejor visualización, el diagrama se ha dividido en 3 partes.

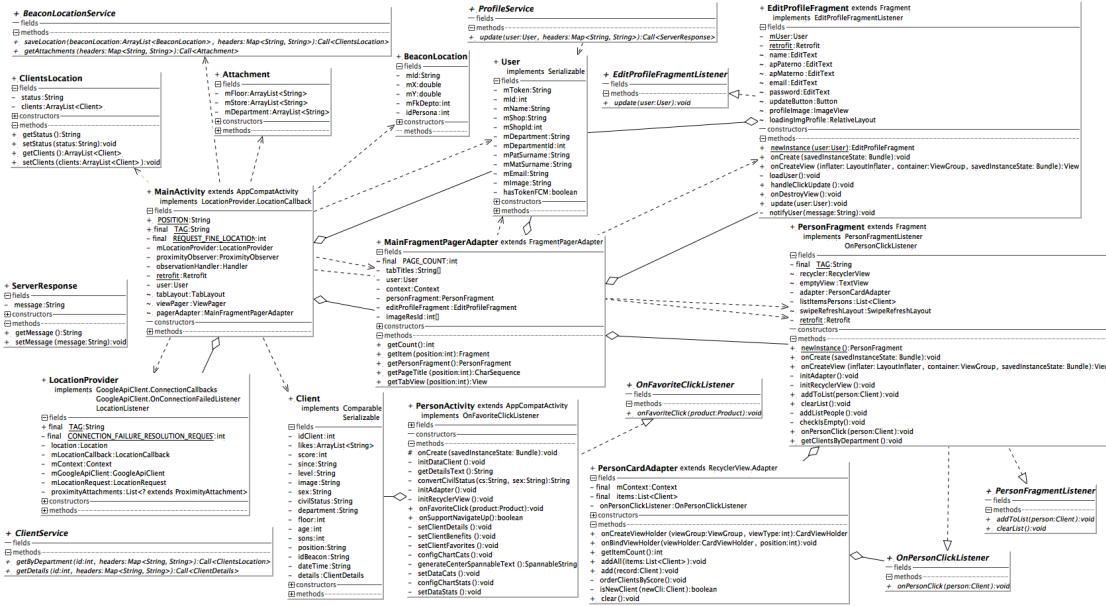


Figura 5.130: Diagrama de clases del prototipo 2 de la AIPV (Visualización completa).

La descripción de los nuevos elementos de la figura 5.131 es la siguiente:

- **ClientsLocation:** Clase que retrofit mapea para la respuesta de la API REST al enviar la ubicación de un Beacon la cual contiene un estatus y un arreglo de Clientes que son los cercanos al Beacon.
- **ServerResponse:** Clase que retrofit mapea para la respuesta de algunos endpoints de la API REST, por ejemplo al actualizar la información de perfil de un vendedor. La clase contiene un mensaje que el servidor envía para notificar los cambios realizados.
- **Client:** Clase con la información necesaria de un cliente cercano.
- **ClientService:** Interfaz que define dos métodos GET , el primero getByDepartment, para obtener a los clientes más recientes cercanos a un departamento por su id; getDetails para obtener más detalles sobre un cliente en específico mediante su id de persona.

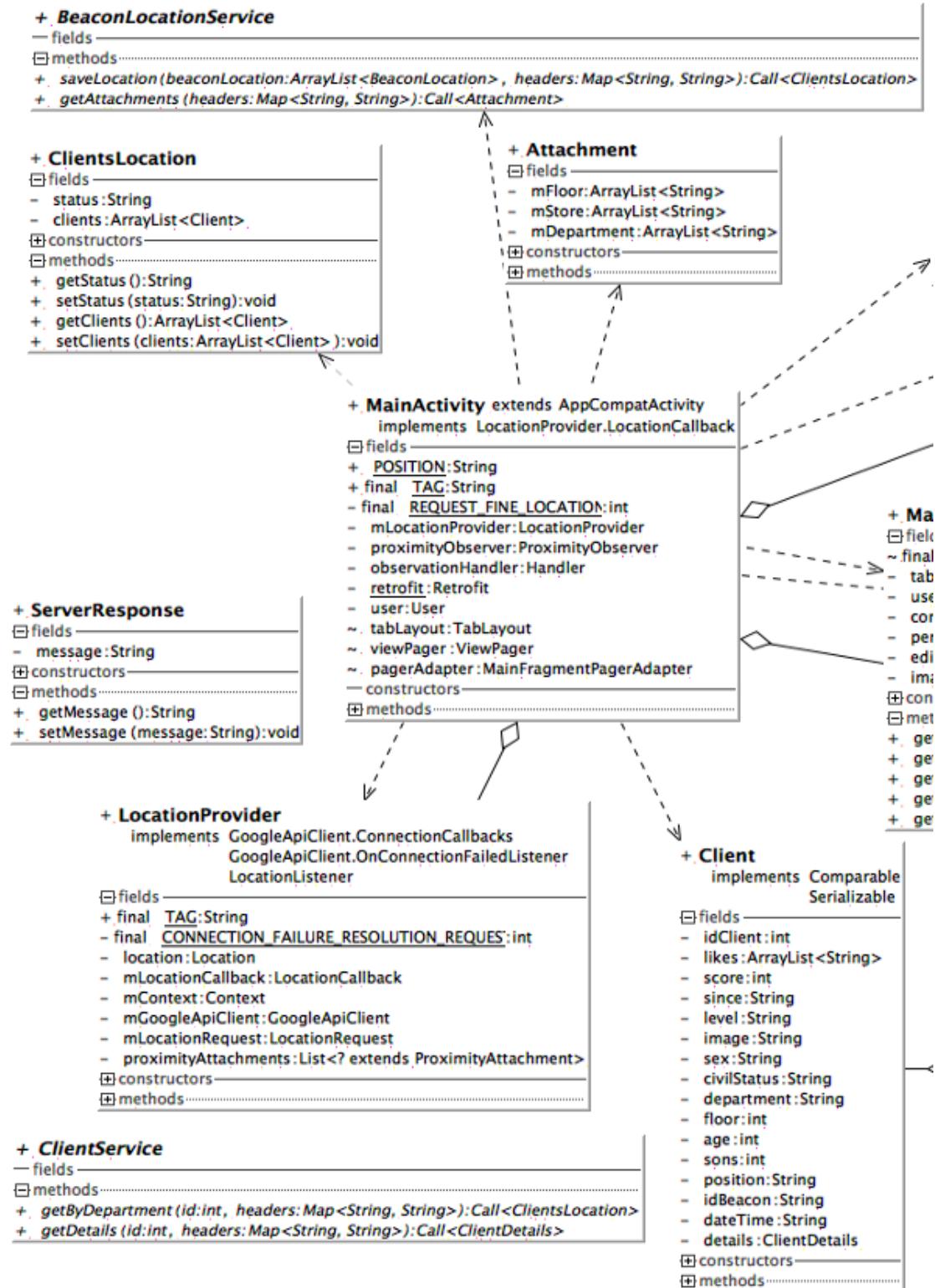


Figura 5.131: Diagrama de clases del prototipo 2 de la AIPV (Parte 1).

---

La descripción de los nuevos elementos de la figura 5.132 es la siguiente:

- **ProfileService**: Interfaz que define el método update de tipo PUT, que permite enviar la información del perfil del vendedor para ser actualizada, recibe como respuesta la clase ServerResponse.
- **EditProfileFragmentListener**: Interfaz que define el método update, la interfaz se implementa en la clase EditProfileFragment y sirve para enviar la información del perfil del vendedor para ser actualizada.
- **MainFragmentPagerAdapter**: Clase que extiende de FragmentPagerAdapter y nos permite crear un paginador de fragmentos los cuales incluye a PersonFragment y EditProfileFragment, en esta clase se crean nuevas instancias de las mismas para organizarlas en forma de pestañas.
- **PersonActivity**: Clase que extiende de AppCompatActivity y en la que se muestra exclusivamente los datos de un cliente.
- **PersonCardAdapter**: Clase que extiende de RecyclerView.Adapter y funciona como el adaptador del arreglo de clientes que se muestran en un CardView, esto nos permite ocupar una sola vista para todos los clientes y ordenarlos horizontalmente en forma de lista.

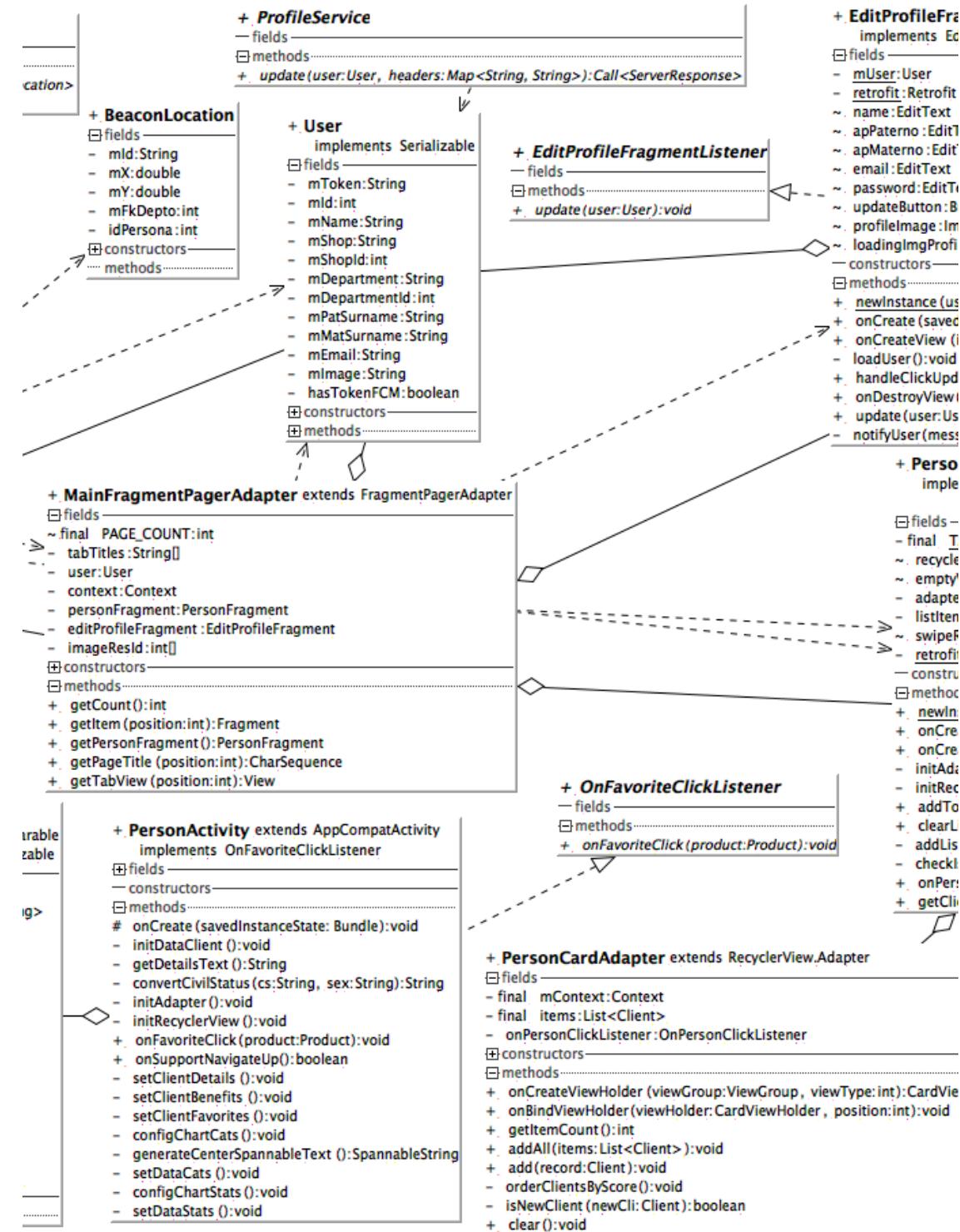


Figura 5.132: Diagrama de clases del prototipo 2 de la AIPV (Parte 2) .

---

La descripción de los nuevos elementos de la figura 5.133 es la siguiente:

- **PersonFragmentListener:** Interfaz que define los métodos addToList y clearList, la interfaz se implementa en la clase PersonFragment, los métodos sirven para agregar y eliminar datos de la lista de los clientes que se muestran en el fragmento.
- **OnPersonClickListener:** Interfaz que define el método onPersonClick, la interfaz se implementa en la clase PersonFragment y sirve para iniciar la actividad PersonActivity desde el fragmento PersonFragment.
- **EditProfileFragment:** Clase que extiende de Fragment e implementa la clase EditProfileFragmentListener, es el fragmento en el que se muestra el formulario para actualizar los datos del perfil del cliente.
- **PersonFragment:** Clase que extiende de Fragment e implementa las clases PersonFragmentListener y OnPersonClickListener, es el fragmento en el que se muestra la lista de los clientes cercanos.

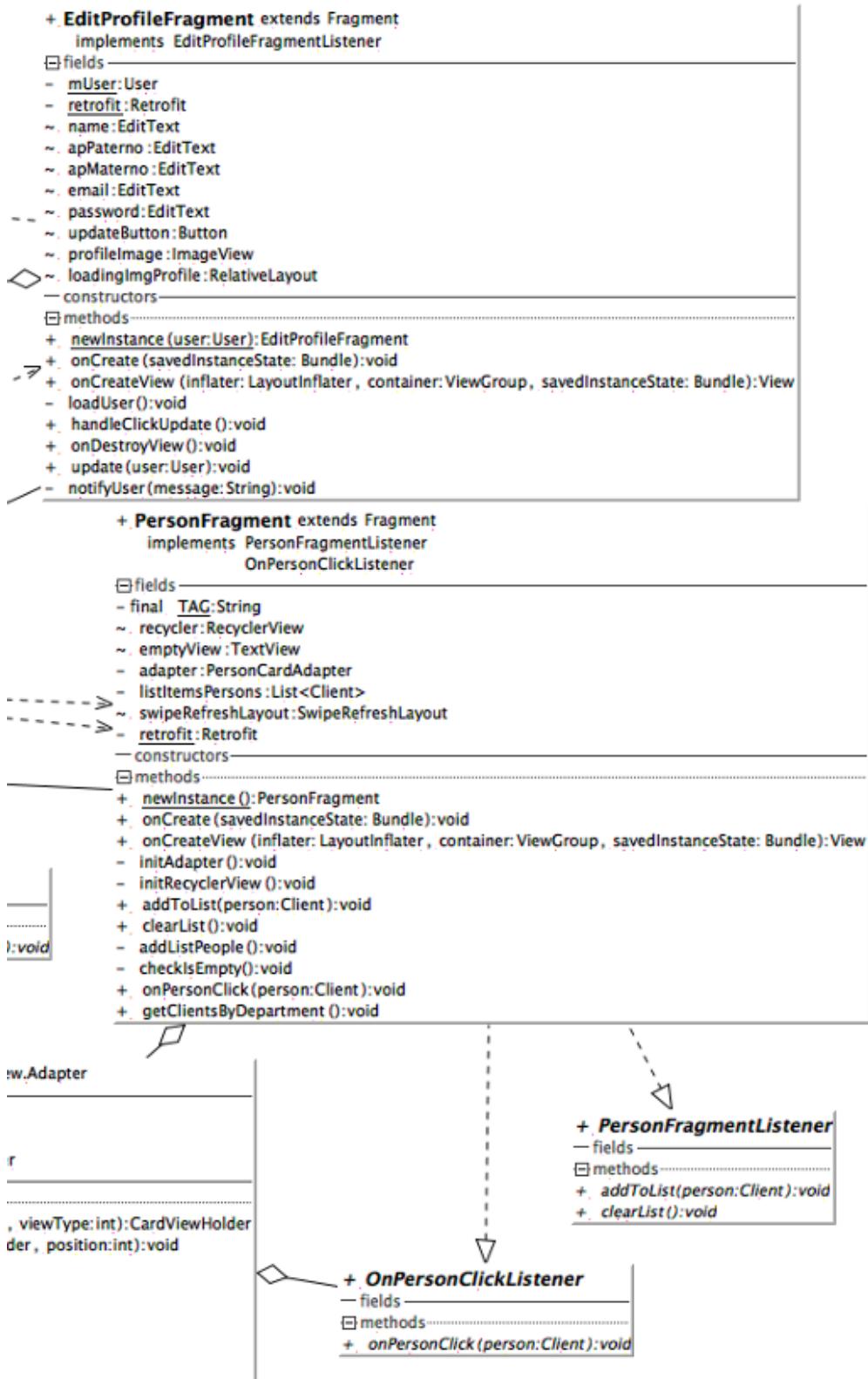


Figura 5.133: Diagrama de clases del prototipo 2 de la AIPV (Parte 3) .

### 5.2.2.2. Diseño

A partir de los requerimientos definidos para este prototipo se muestran los casos de uso y diagramas de secuencia.

#### Diagramas de secuencia.

##### Obtener estadísticas del cliente.

En la figura 5.134 se muestra el diagrama de secuencia para editar perfil en la AIPV, el cual describe los pasos que se llevan a cabo para que el usuario vendedor edite sus datos de perfil correctamente. Para una mejor visualización el diagrama se ha dividido en 2 partes las cuales se muestran en las figuras 5.135 y 5.136

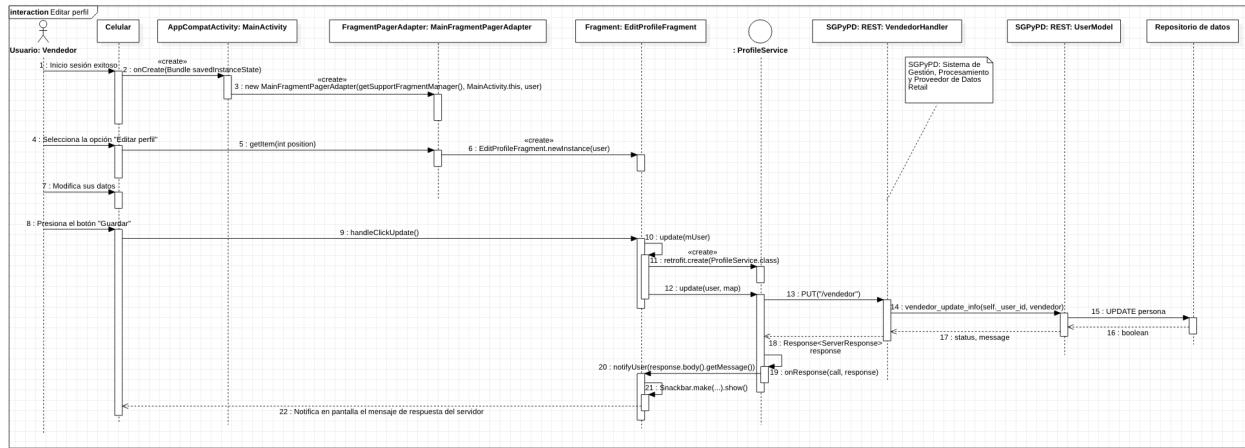


Figura 5.134: Diagrama de secuencia para editar perfil de un vendedor.

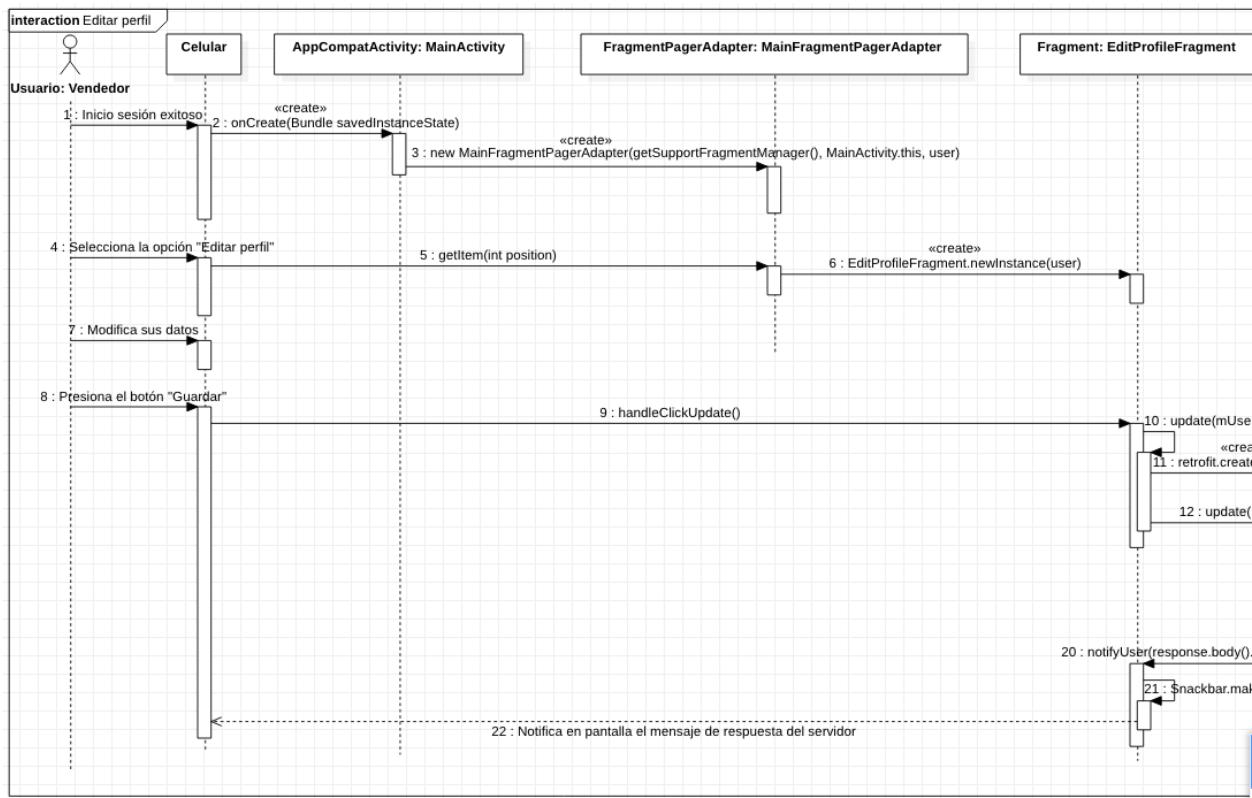


Figura 5.135: Diagrama de secuencia para editar perfil de un vendedor (Parte 1).

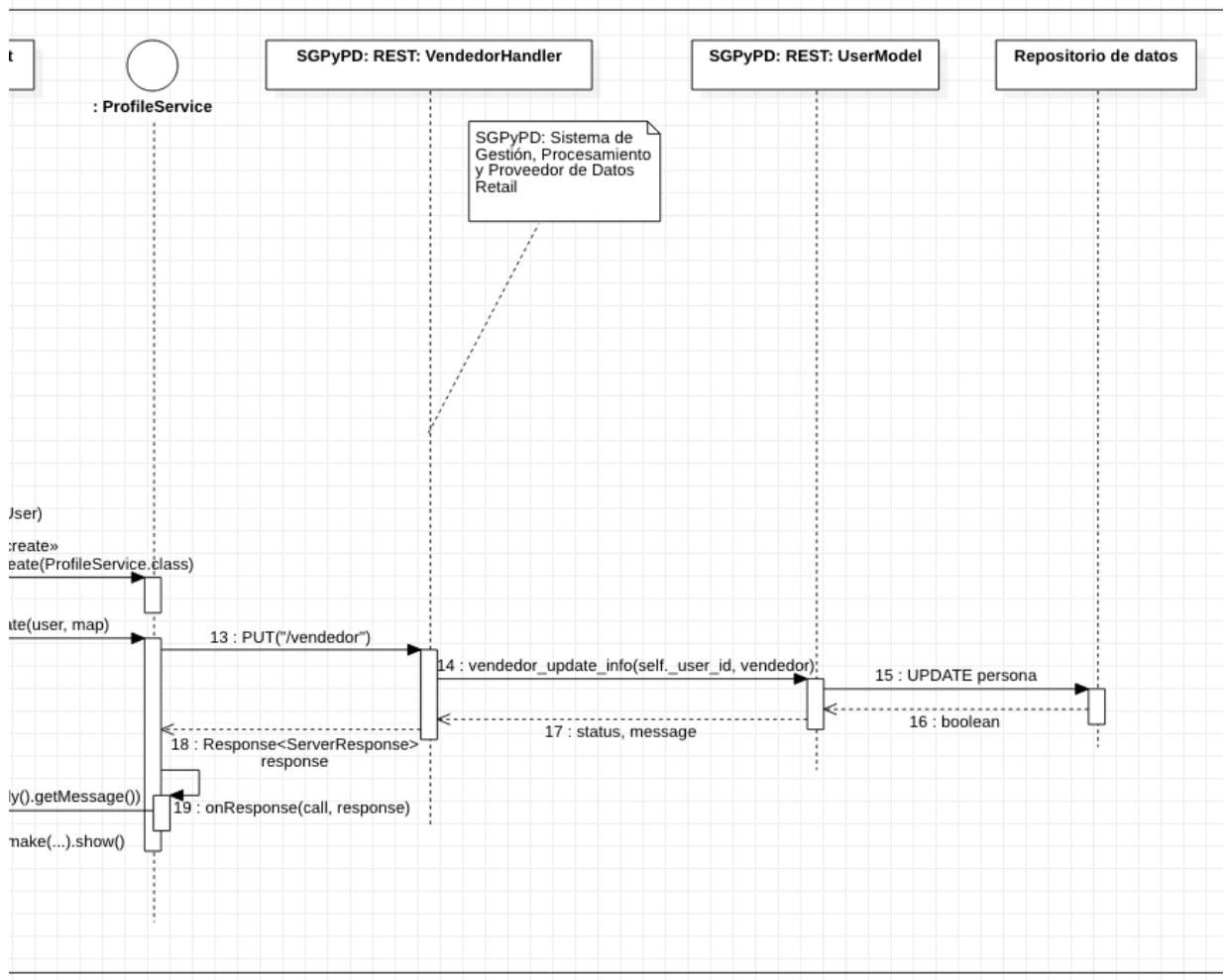


Figura 5.136: Diagrama de secuencia para editar perfil de un vendedor (Parte 2).

### Ver clientes cercanos.

En la figura 5.137 se muestra el diagrama de secuencia para ver clientes cercanos en la AIPV, es decir, mostrarle al usuario vendedor los usuarios que se han conectado previamente a un Beacon en el departamento que se encuentre. Para una mejor visualización el diagrama se ha dividido en 2 partes las cuales se muestran en las figuras 5.138 y 5.139 .

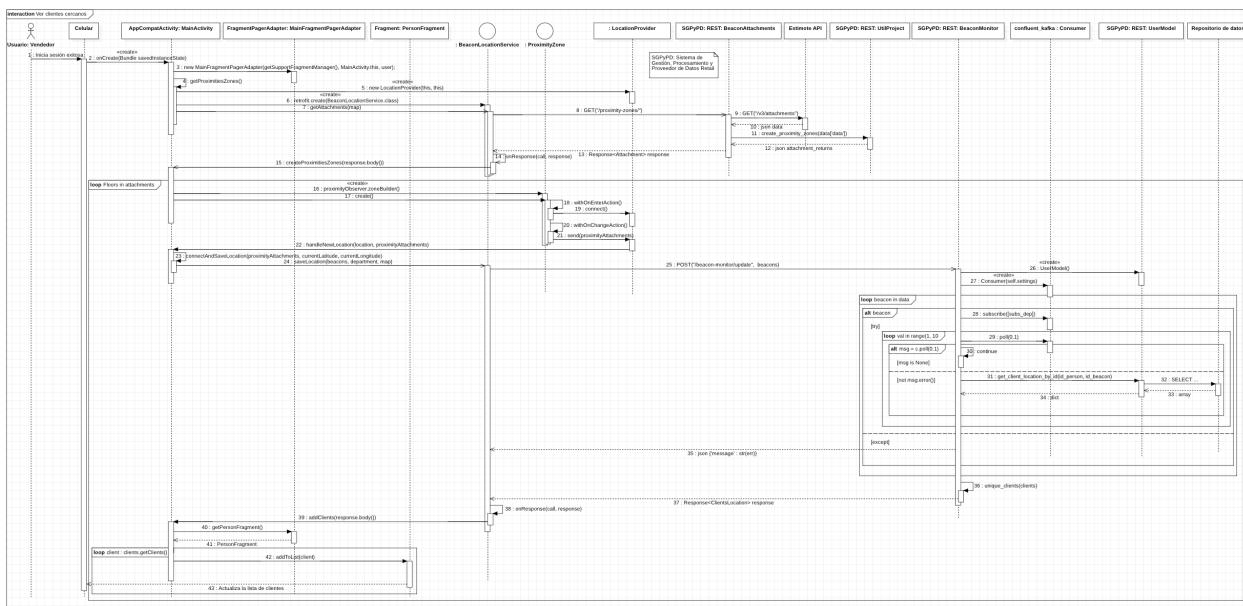


Figura 5.137: Diagrama de secuencia para ver a los clientes cercanos en la aplicación.

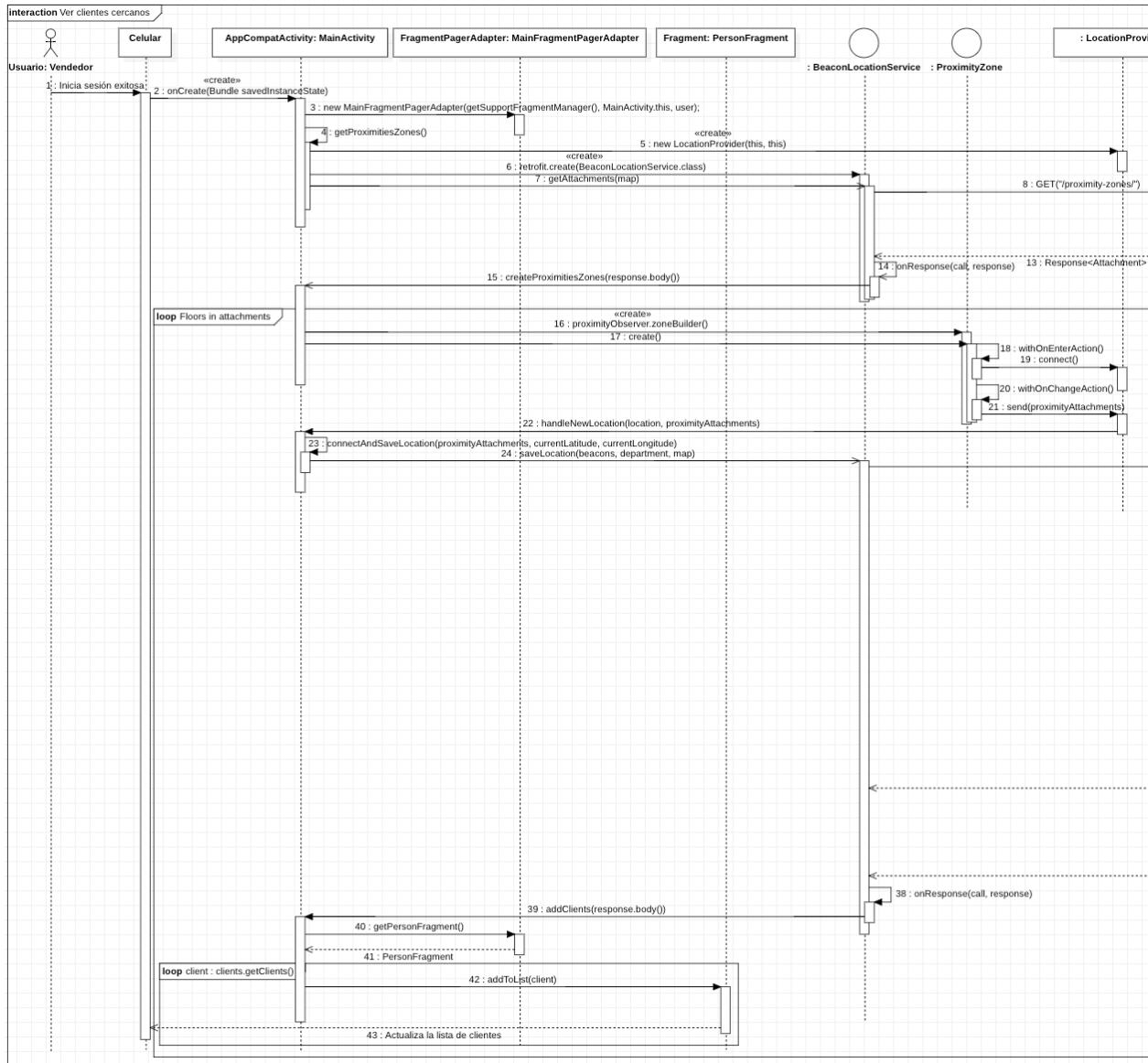


Figura 5.138: Diagrama de secuencia para ver a los clientes cercanos en la aplicación (Parte 1).

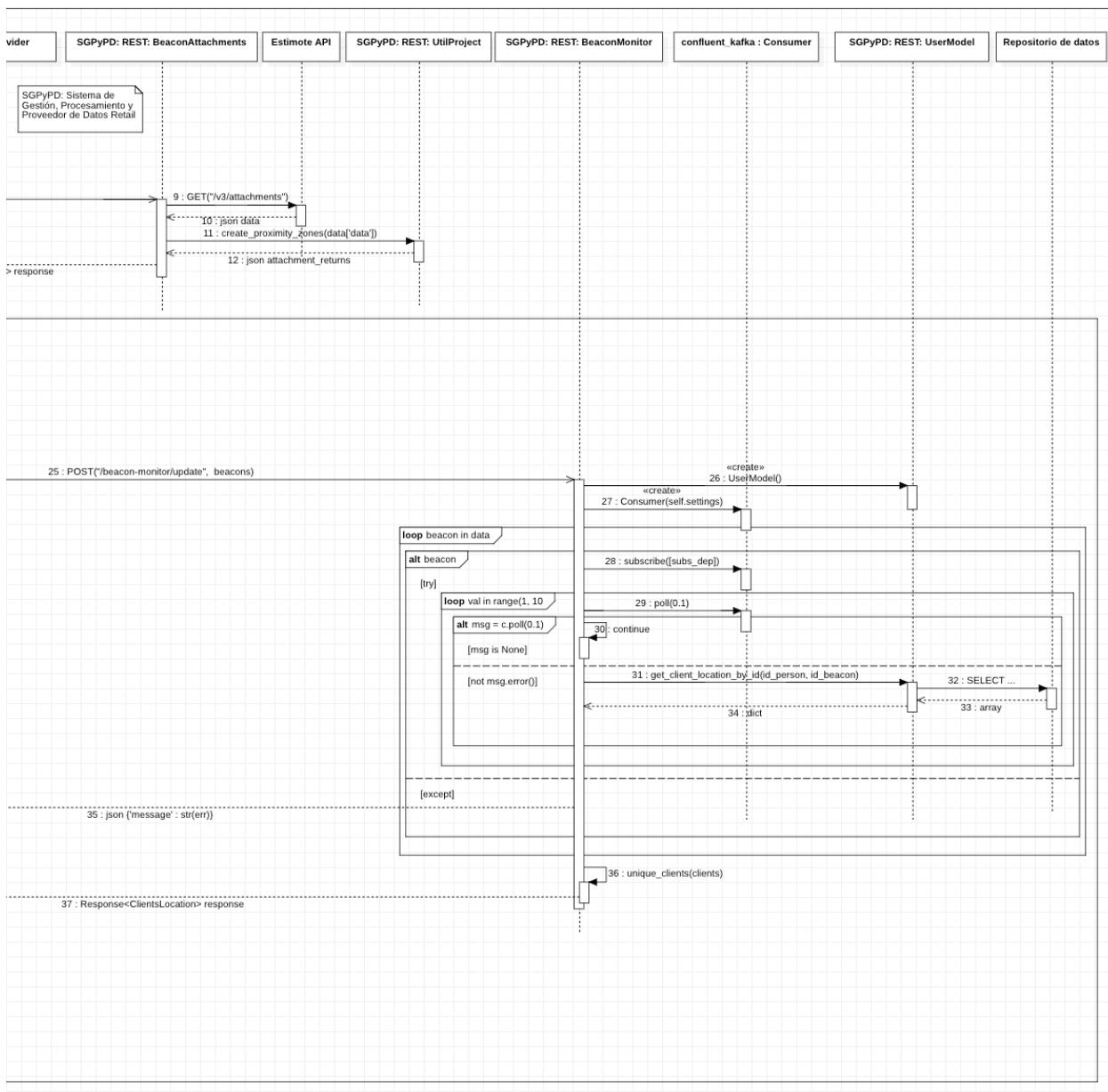


Figura 5.139: Diagrama de secuencia para ver a los clientes cercanos en la aplicación (Parte 2).

### 5.2.3. Prototipo 3: Integración con servidor (Detalles y estadísticas de clientes y envío de recomendaciones)

El prototipo 3 de la aplicación incluye las modificaciones en el diseño inicial de la aplicación para mostrar los detalles y estadísticas de un cliente cercano al vendedor y la integración de la aplicación con los servicios REST para obtener dichos datos. De igual forma dentro de este prototipo se incluye el diseño para que el vendedor envié recomendaciones de productos del departamento actual a un cliente y su respectiva integración con el servidor para el envío de las mismas en forma de notificación por medio de Firebase Cloud Messaging (FCM).

### 5.2.3.1. Análisis

Dentro del análisis para el desarrollo de este prototipo se incorporan los requerimientos funcionales **RFAPV5 Obtener detalles y estadísticas de un cliente** y **RFAPV6 Enviar recomendaciones a clientes**, definidos previamente en el capítulo del “Bosquejo general de la aplicación” con el título de “Requerimientos funcionales de la Aplicación Interactiva para el Personal de Ventas”.

### Casos de uso de la AIPV.

La figura 5.112 definida previamente en el prototipo 1, muestra los casos de uso de la AIPV.

### Diagrama de clases.

La figura 5.140 presenta el diagrama de clases para el prototipo 3 de la AIPV. Para una mejor visualización, el diagrama se ha dividido en 3 partes.

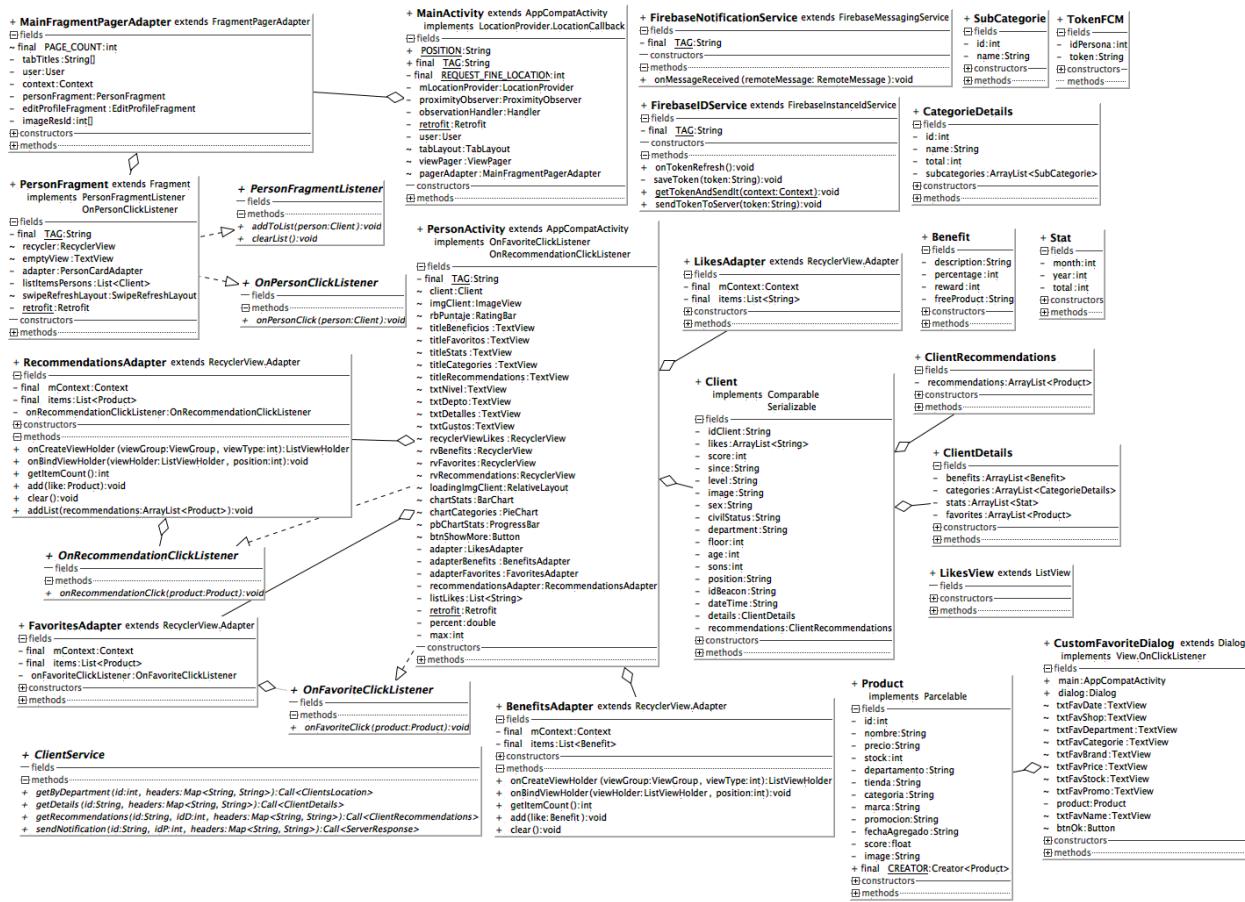


Figura 5.140: Diagrama de clases del prototipo 3 de la AIPV (Visualización completa).

La descripción de los nuevos elementos de la figura 5.141 es la siguiente:

- **PersonActivity**: Versión actualizada de la clase que extiende de **AppCompatActivity** y en la que se muestran los detalles de un cliente.

- **RecommendationsAdapter:** Clase que extiende de RecyclerView.Adapter y funciona como el adaptador del arreglo de recomendaciones de productos del departamento actual que le podrían interesar al cliente del cual se están mostrando sus detalles.
- **OnRecommendationClickListener:** Interfaz que define el método onRecommendationClick el cual recibe un producto seleccionado por el vendedor para ser enviado como recomendación al cliente.

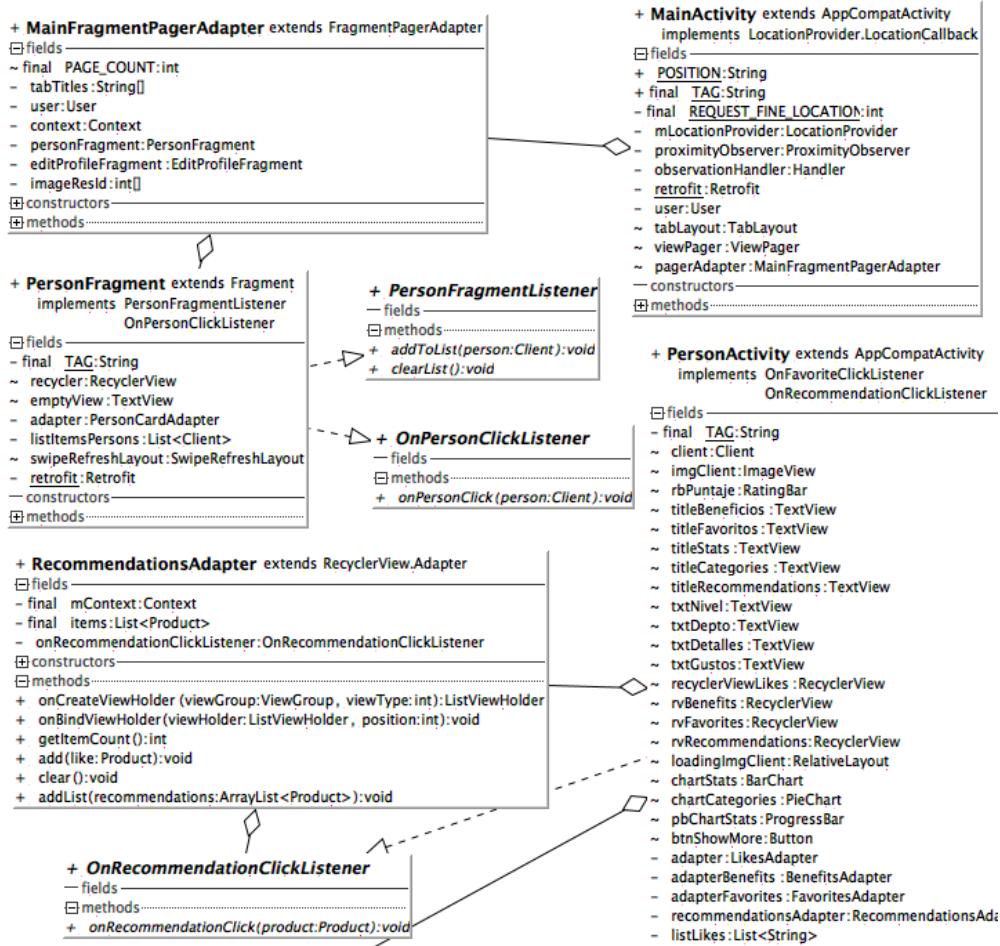


Figura 5.141: Diagrama de clases del prototipo 3 de la AIPV (Parte 1) .

La descripción de los nuevos elementos de la figura 5.142 es la siguiente:

- **FirebaseNotificationService:** Clase que extiende de FirebaseMessagingService y que nos permite recibir cualquier notificación en la aplicación.
- **FirebaseIDService:** Clase que extiende de FirebaseInstanceIdService y funciona para obtener el token actual de Firebase Cloud Messaging para el dispositivo.
- **TokenFCM:** Clase POJO que se utiliza para relacionar al vendedor con la sesión actual y el token de Firebase Cloud Messaging que se genera en la aplicación lo cual se envía en formato JSON al servidor para actualizar dicho token.
- **ClientDetails:** Clase POJO que contiene todos los detalles de un cliente: beneficios, categorías más compradas, estadísticas y productos favoritos.

- **CategorieDetails:** Clase POJO que se recibe como JSON al obtener las estadísticas del cliente, en este caso sobre las categorías que más compra.
- **SubCategorie:** Clase POJO que representa una subcategoría, estas subcategorías se encuentran en forma de arreglo dentro de cada una de las categorías que más compra un cliente.
- **Benefit:** Clase POJO que contiene los datos necesarios de un beneficio de acuerdo al nivel del cliente, se encuentra en forma de arreglo dentro de los detalles del cliente.
- **Stat:** Clase POJO, especifica el total de compras que realizó el cliente en un año y mes específicos, se encuentra en forma de arreglo dentro de los detalles del cliente.
- **ClientRecommendations:** Clase POJO que se recibe como JSON después de obtener las estadísticas del cliente, contiene un arreglo de productos del departamento que podrían gustarle al cliente.
- **LikesAdapter:** Clase que extiende de RecyclerView.Adapter y funciona como el adaptador del arreglo de los gustos de un cliente que se muestran en forma de lista horizontal.
- **LikesView:** Clase que extiende de ListView y nos permite configurar el tamaño de la vista que se ocupa para desplegar los gustos del cliente en forma de lista.

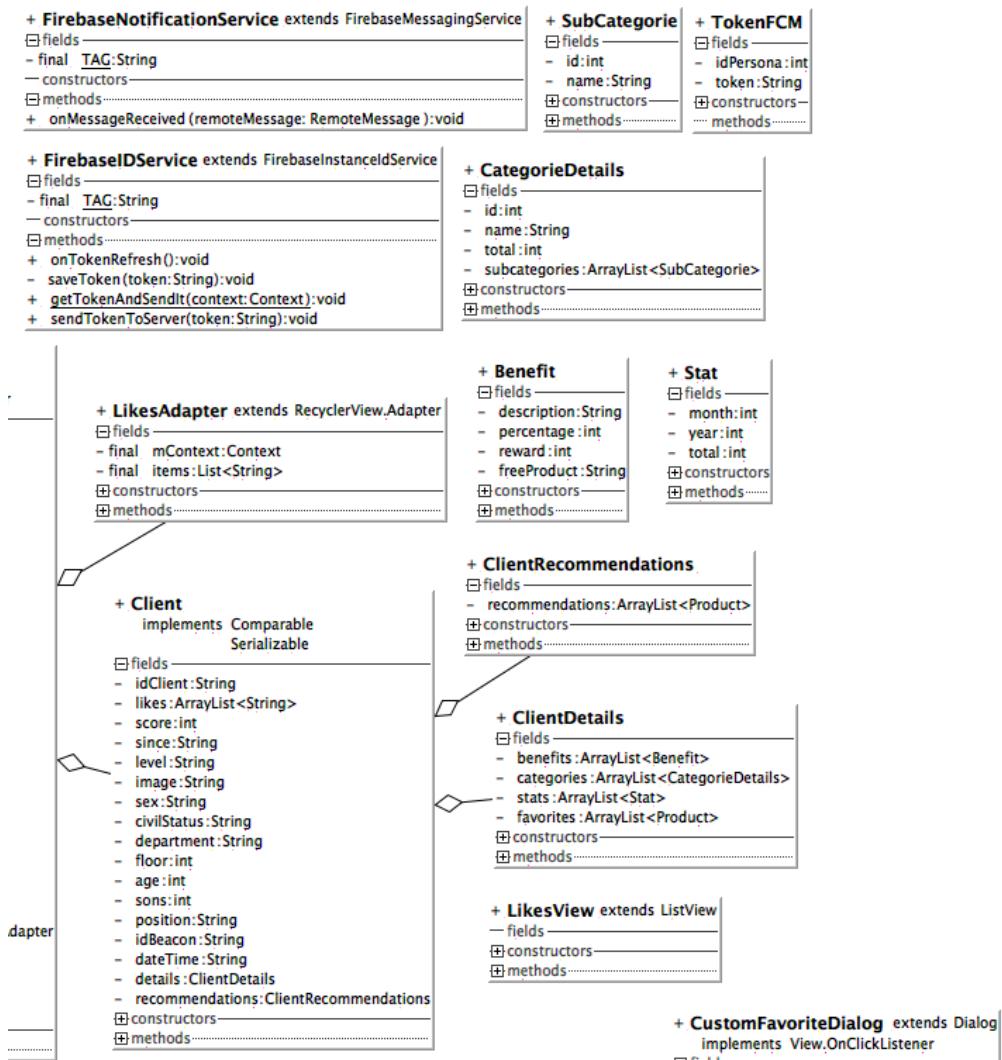


Figura 5.142: Diagrama de clases del prototipo 3 de la AIPV (Parte 2) .

La descripción de los nuevos elementos de la figura 5.143 es la siguiente:

- **FavoritesAdapter:** Clase que extiende de RecyclerView.Adapter y funciona como el adaptador del arreglo de productos favoritos de un cliente que se muestran en forma de lista horizontal.
- **OnFavoriteClickListener:** Interfaz que define el método onFavoriteClick, la interfaz se implementa en la clase PersonActivity y sirve para mostrar el dialogo CustomFavoriteDialog en el cual se muestra la descripción del producto.
- **BenefitsAdapter:** Clase que extiende de RecyclerView.Adapter y funciona como el adaptador del arreglo de beneficios de un cliente que se muestran en forma de lista horizontal.
- **Product:** La clase POJO actualizada que incorpora ahora los nuevos atributos score y image, los cuales se muestran en los productos que se podrían recomendar al cliente.
- **CustomFavoriteDialog:** Clase que extiende de Dialog y la cual se utiliza para mostrar los detalles de un producto que el cliente tiene en favoritos.

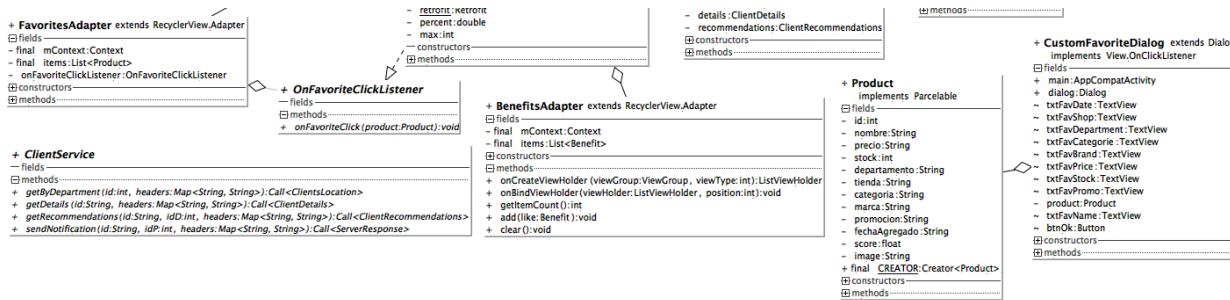


Figura 5.143: Diagrama de clases del prototipo 3 de la AIPV (Parte 3) .

### 5.2.3.2. Diseño

A partir de los requerimientos definidos para este prototipo se muestran los diagramas de secuencia.

#### Diagramas de secuencia.

##### Obtener estadísticas del cliente.

En la figura 5.144 se muestra el diagrama de secuencia para obtener las estadísticas de un cliente, el cual describe los pasos que se llevan a cabo para que el usuario vendedor pueda ver información adicional del cliente como beneficios por su nivel, 5 de sus productos favoritos, el total de compras realizadas en los últimos 6 meses y las categorías de productos más compradas. Estos datos se muestran únicamente si el cliente otorga los permisos para cada uno de ellos. Para una mejor visualización el diagrama se ha dividido en 2 partes las cuales se muestran en las figuras 5.145 y 5.146 .

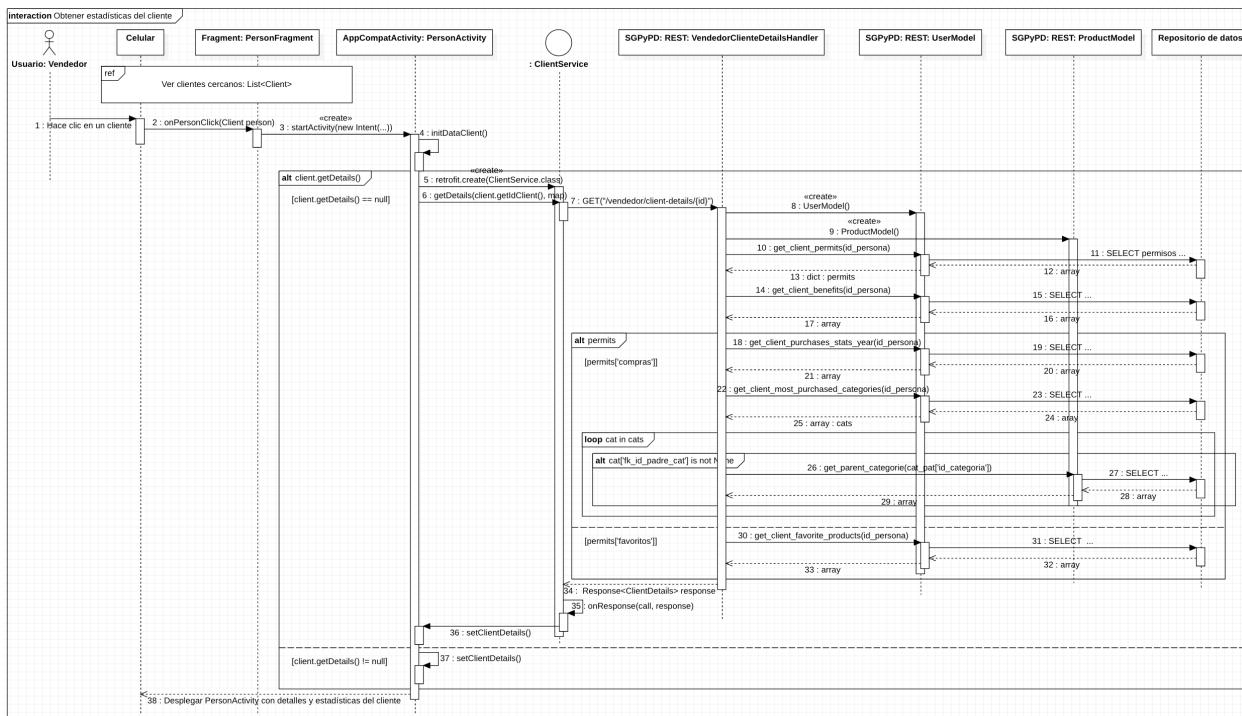


Figura 5.144: Diagrama de secuencia para obtener estadísticas de un cliente.

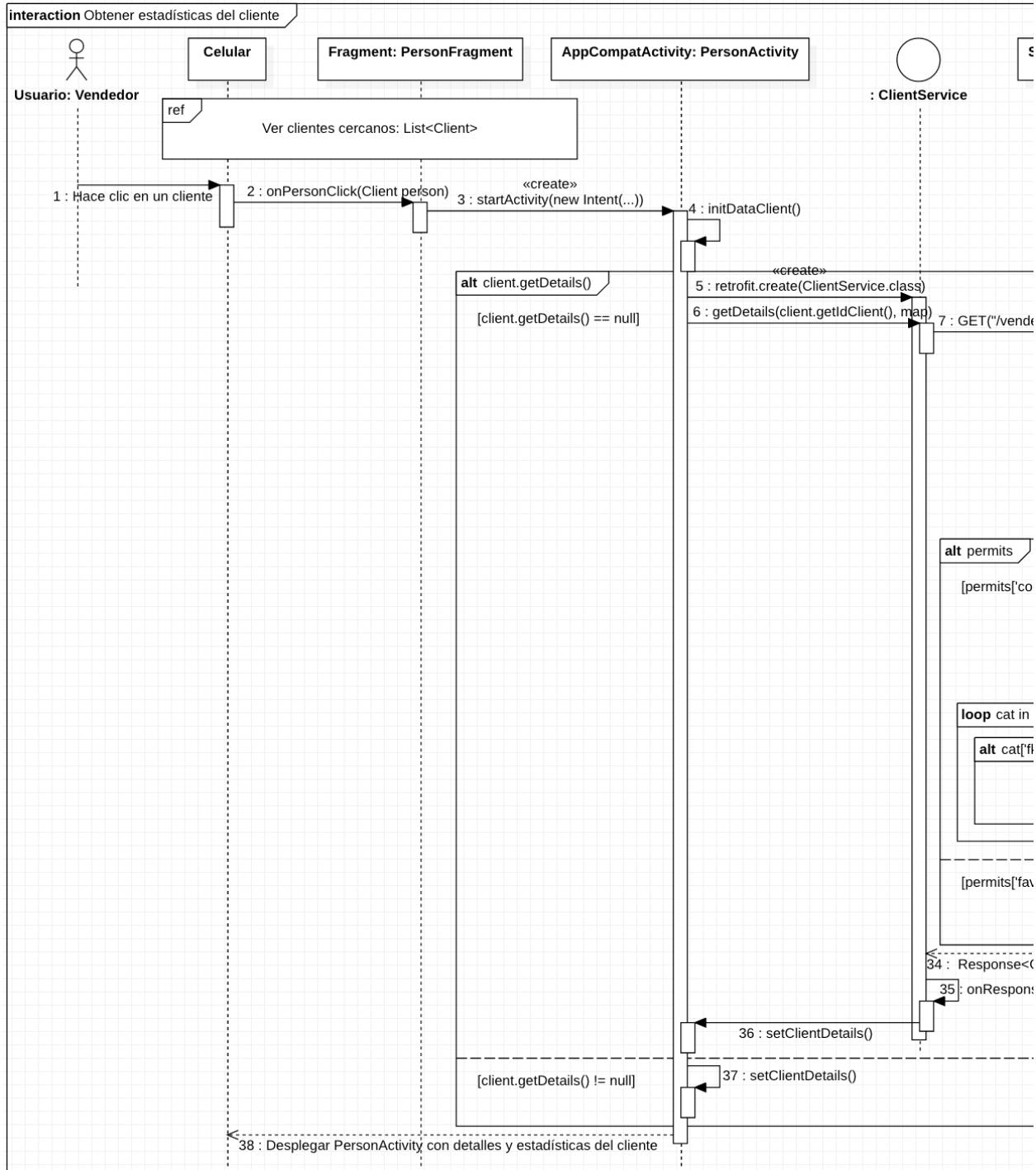


Figura 5.145: Diagrama de secuencia para obtener estadísticas de un cliente (Parte 1).

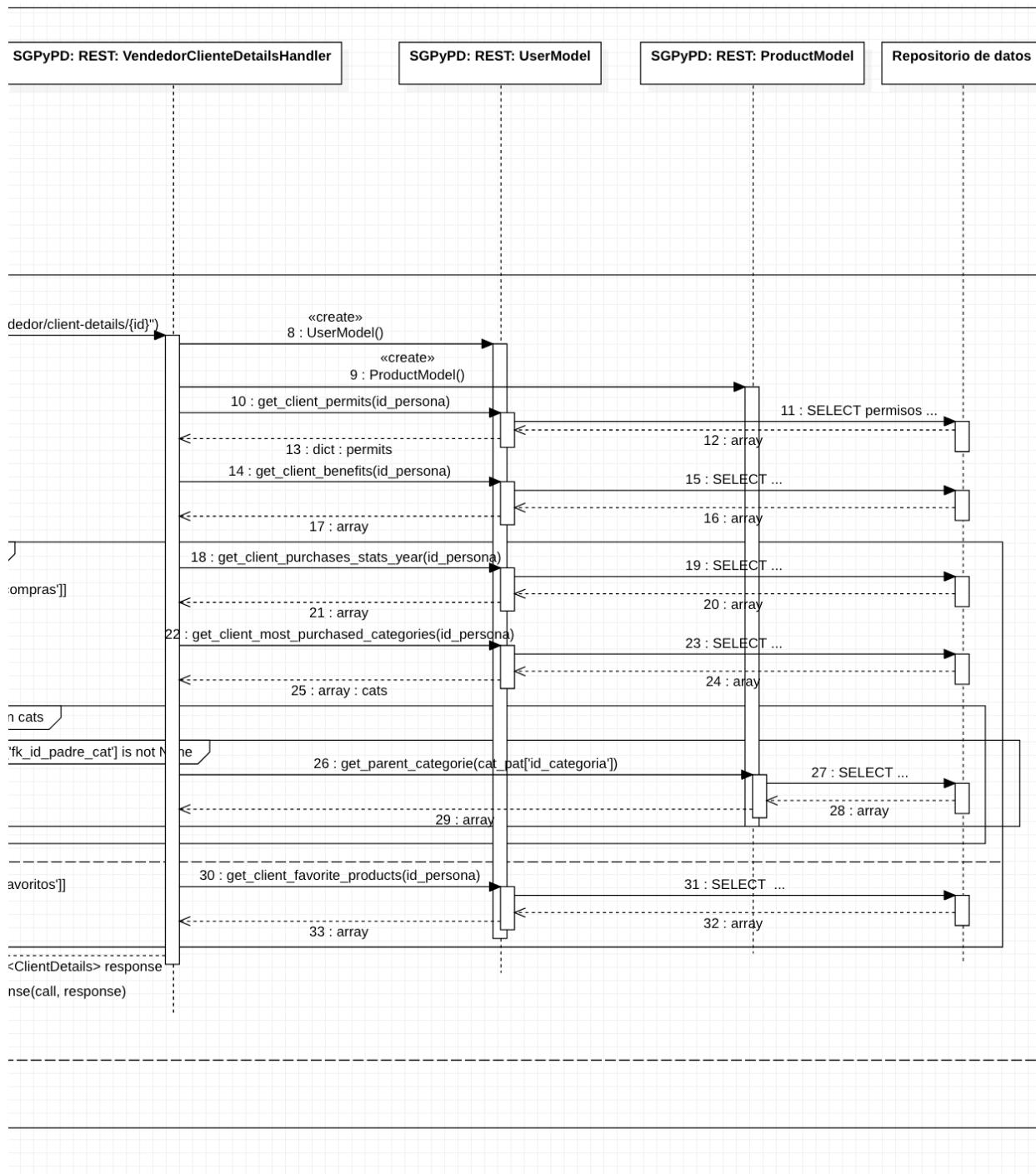


Figura 5.146: Diagrama de secuencia para obtener estadísticas de un cliente (Parte 2).

### Obtener recomendaciones para cliente.

En la figura 5.147 se muestra el diagrama de secuencia para obtener recomendaciones de productos para un cliente exclusivamente para el departamento en el que se encuentra, esto ocurre acto seguido de que se obtienen las estadísticas del cliente del servidor por lo que se coloca al inicio como referencia el diagrama

de secuencia “Obtener estadísticas del cliente”. Para una mejor visualización el diagrama se ha dividido en 2 partes las cuales se muestran en las figuras 5.148 y 5.149 .

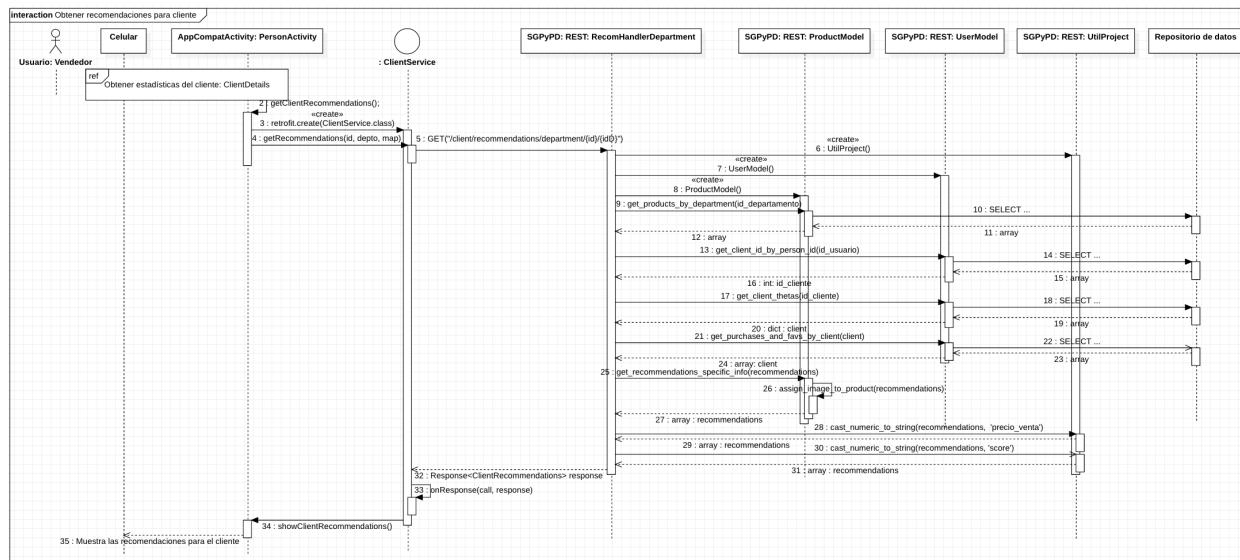


Figura 5.147: Diagrama de secuencia para obtener recomendaciones para un cliente.

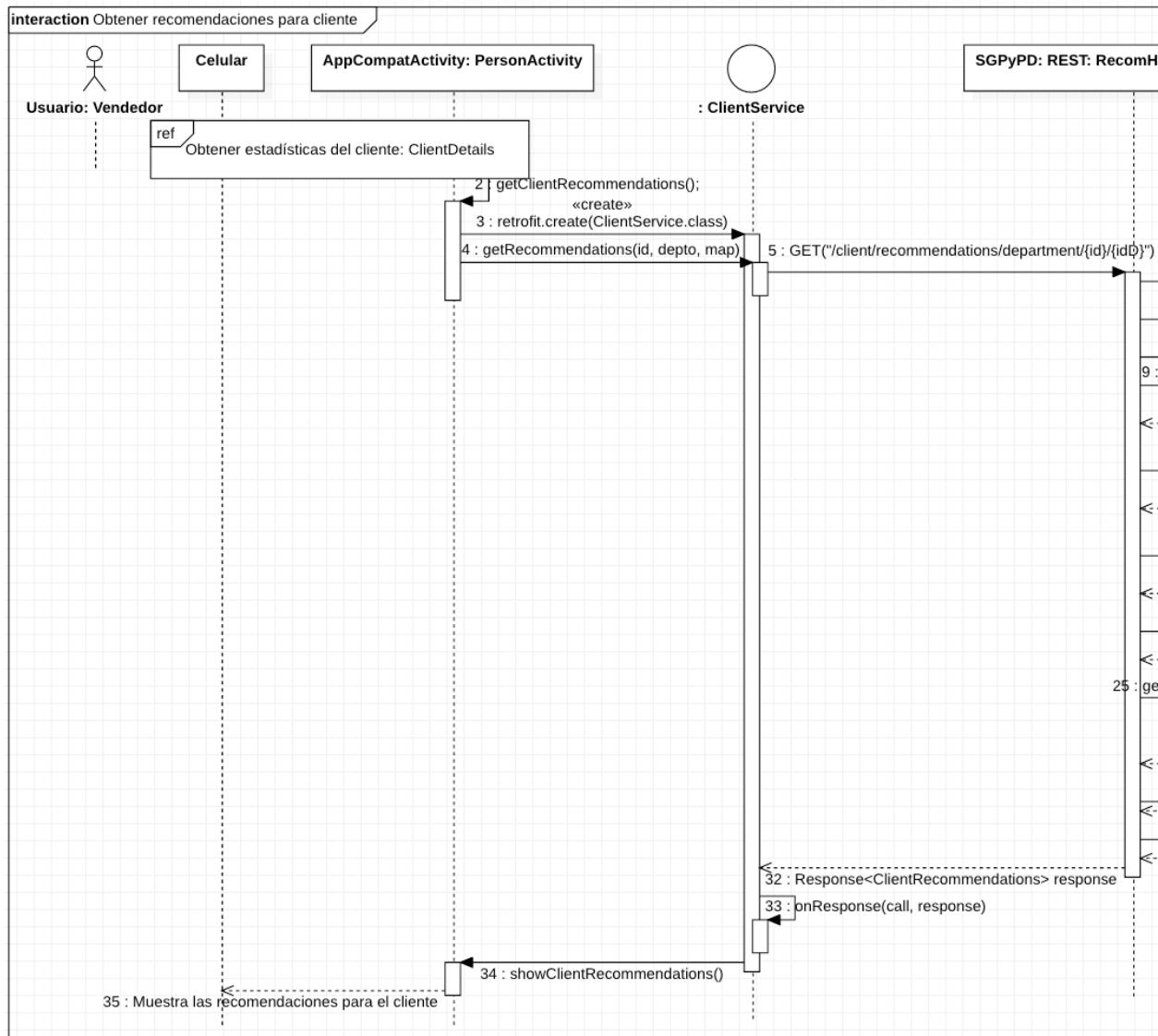


Figura 5.148: Diagrama de secuencia para obtener recomendaciones para un cliente (Parte 1).

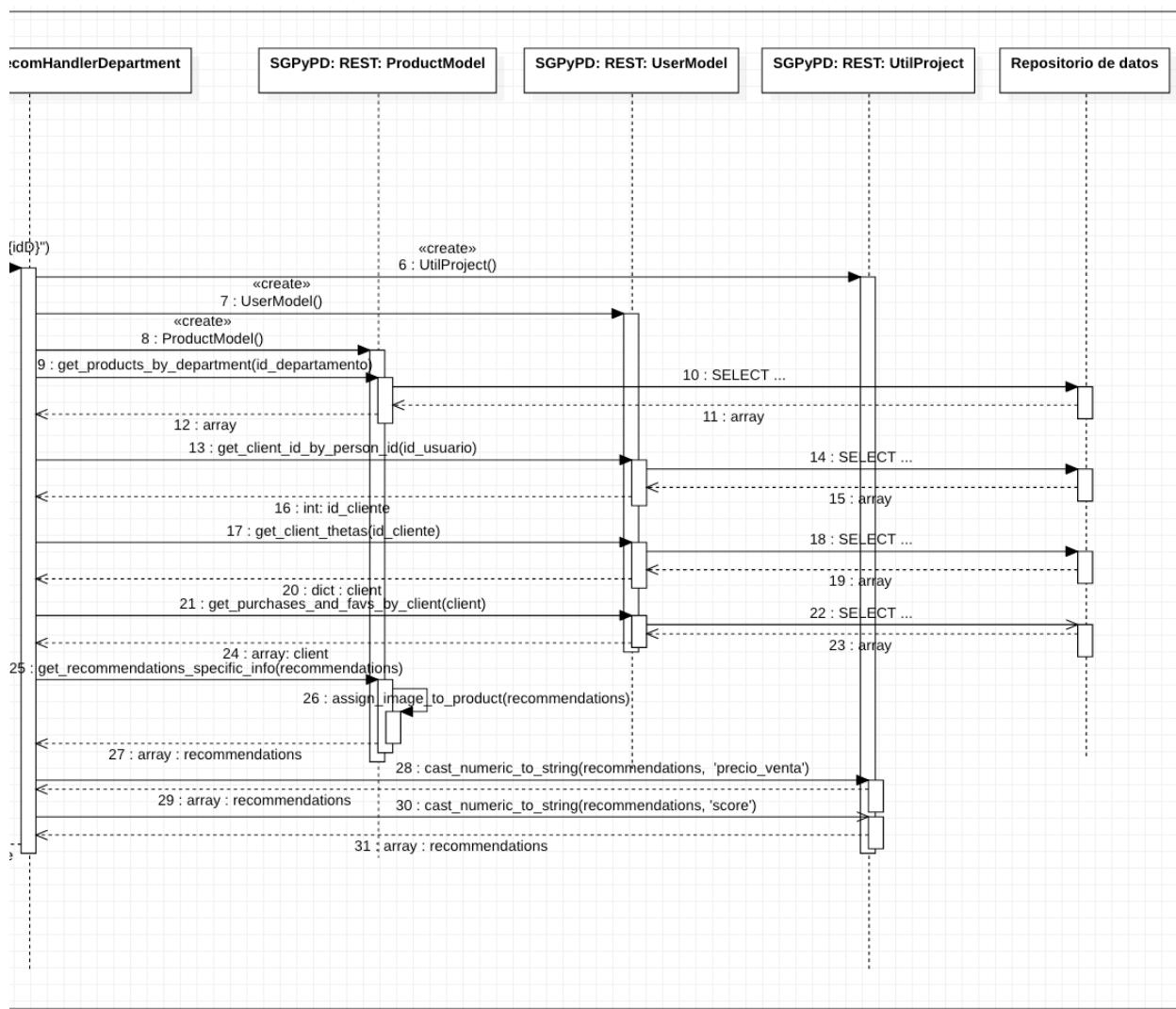


Figura 5.149: Diagrama de secuencia para obtener recomendaciones para un cliente (Parte 2).

#### Enviar recomendaciones al cliente.

En la figura 5.150 se muestra el diagrama de secuencia para enviar recomendaciones a un cliente, el cual describe los pasos que se llevan a cabo para que el usuario vendedor pueda enviar una recomendación de un producto del departamento actual al cliente mediante una notificación de Firebase Cloud Messaging (FCM), es importante mencionar que esto ocurre después de que se obtienen las recomendaciones para el cliente por lo que se coloca al inicio como referencia el diagrama de secuencia “Obtener recomendaciones para cliente”. Para una mejor visualización el diagrama se ha dividido en 2 partes las cuales se muestran en las figuras 5.151 y 5.152 .

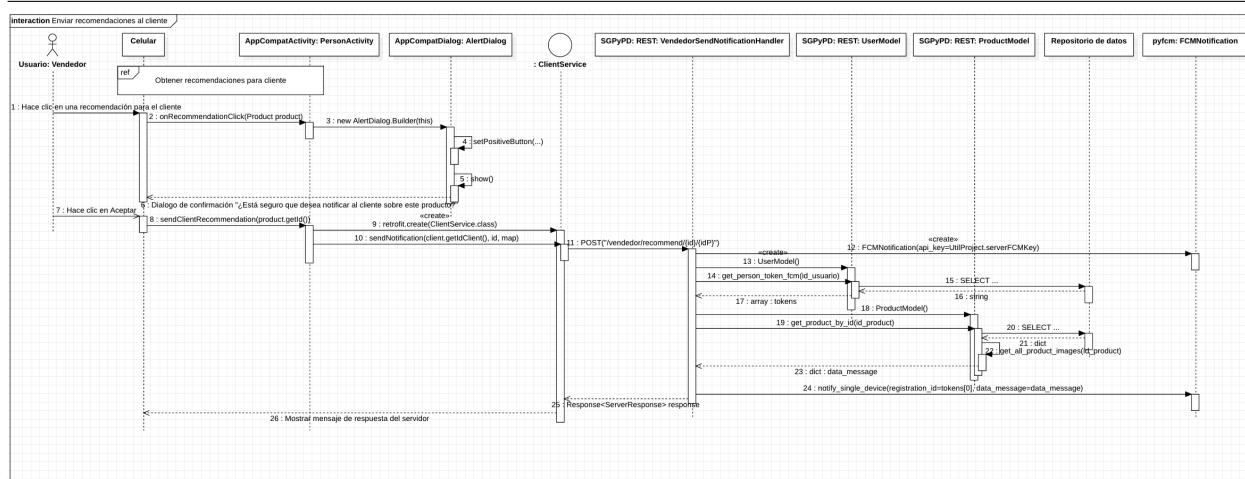


Figura 5.150: Diagrama de secuencia para enviar recomendaciones a un cliente.

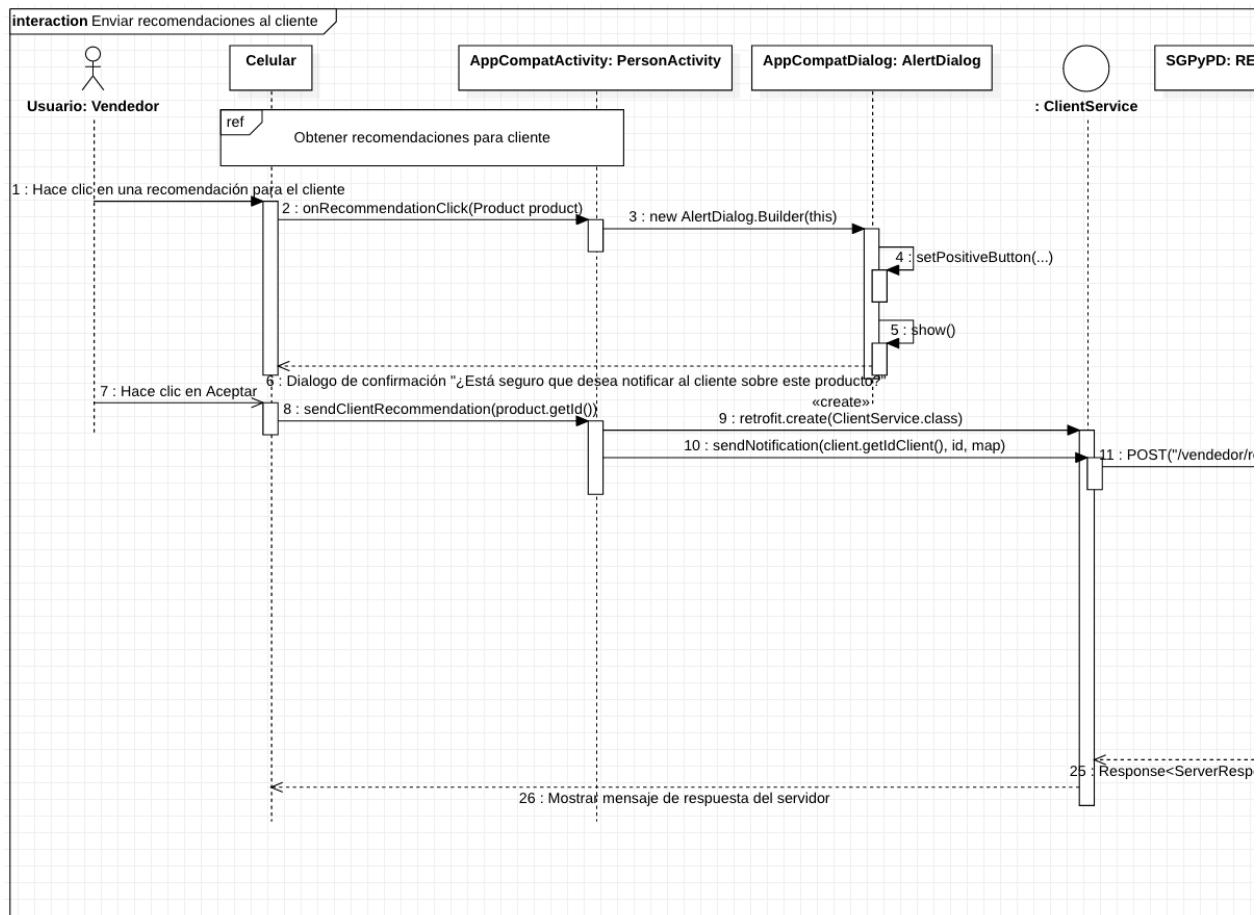


Figura 5.151: Diagrama de secuencia para enviar recomendaciones a un cliente (Parte 1).

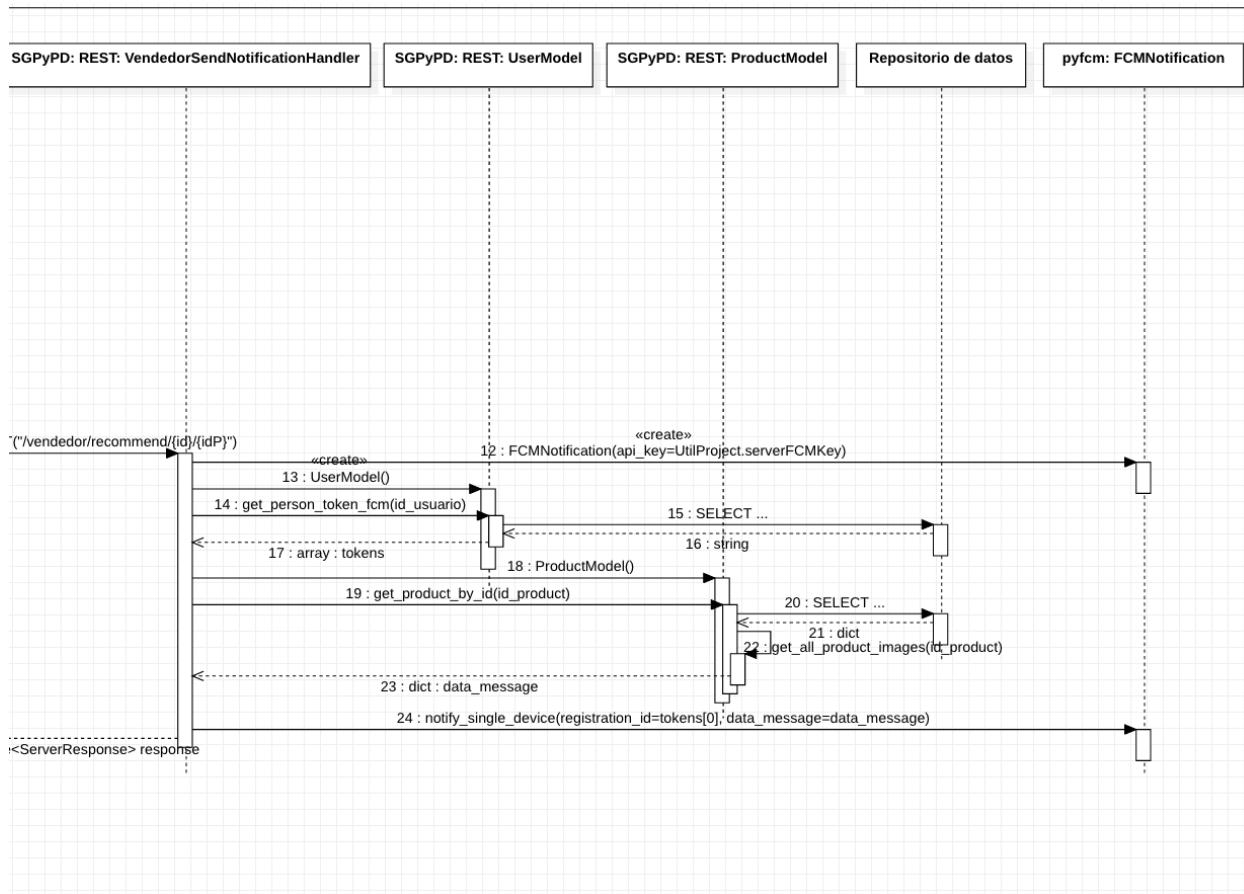


Figura 5.152: Diagrama de secuencia para enviar recomendaciones a un cliente (Parte 2).

## 5.2.4. Prototipo 4: Prototipo final

El prototipo final de la aplicación incluye la actualización de las vistas finales de la aplicación, así como las últimas modificaciones que se realizaron para la integración de la aplicación con Firebase Cloud Messaging para el envío y recepción de notificaciones.

### 5.2.4.1. Análisis

Dentro del análisis de este prototipo se realiza la inclusión del siguiente requerimiento funcional a los ya definidos previamente en el capítulo del “Bosquejo general de la aplicación” con el título de “[Requerimientos Funcionales de la Aplicación Interactiva para el Personal de Ventas](#)”:

- Actualizar token de Firebase Cloud Messaging (FCM)

Id	Nombre	Descripción
RFAIPV7	Actualizar token de Firebase Cloud Messaging (FCM)	La aplicación permitirá actualizar el token de FCM para suscribirse a las notificaciones de cada uno de los departamentos a los que el vendedor se dirija, es decir, cada que el vendedor cambie de departamento de igual forma lo hará el token.

Cuadro 5.3: Requerimiento añadido a los Requerimientos Funcionales de la Aplicación Interactiva para el Personal de Ventas.

### Casos de uso de la AIPV.

La figura 5.153 muestra las modificaciones finales de los casos de uso de la AIPV, en la siguiente lista se detalla cuales fueron.

- CUAPV4 Obtener localización aproximada del cliente fue modificado y ahora es CUAPV4 Obtener detalles y estadísticas del cliente, entre los detalles del cliente se indica en que departamento y a que hora se encontraba en el cliente.
- CUAPV5 Obtener estadísticas del cliente fue eliminado ya que dichas estadísticas se obtienen en conjunto con los detalles y en su lugar se encuentra ahora el CUAPV5 Actualizar token de Firebase Cloud Messaging (FCM) para controlar la suscripción de notificaciones del vendedor dependiendo del departamento en el que se encuentre.

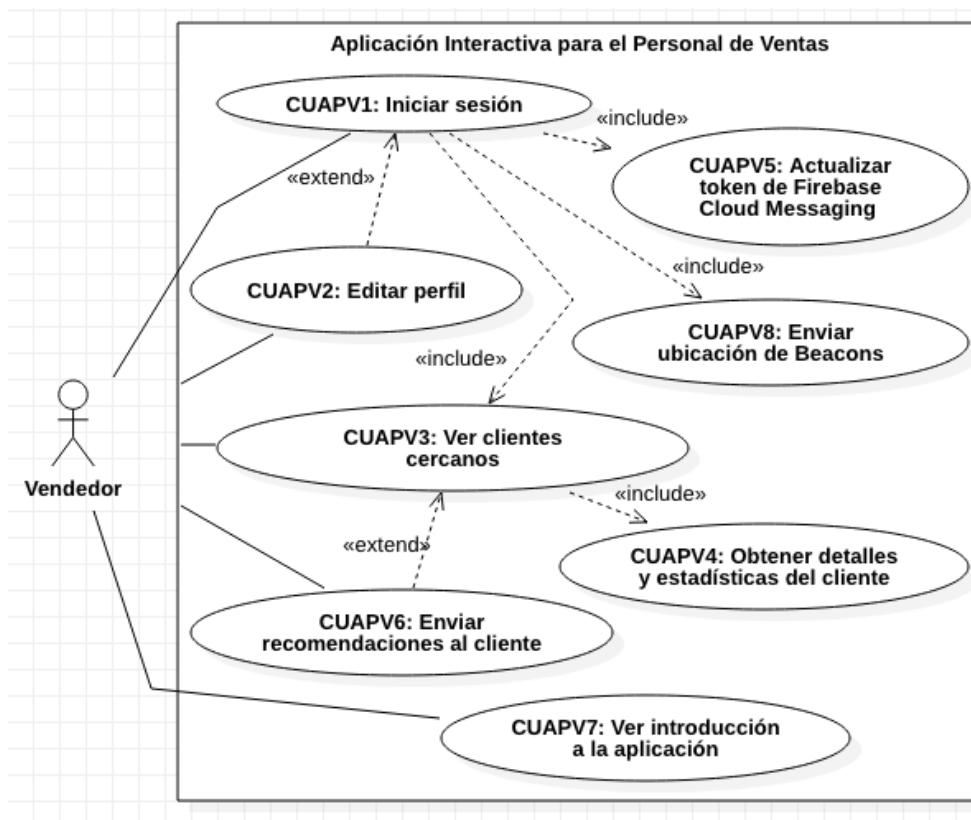


Figura 5.153: Casos de uso de la AIPV.

### 5.2.4.2. Diseño

A continuación se muestra el diagrama de secuencia para actualizar el token de FCM, el flujo de navegación y el diseño final de las interfaces de la aplicación.

#### Diagramas de secuencia.

##### Actualizar el token de FCM.

En la figura 5.154 se muestra el diagrama de secuencia para actualizar el token de FCM, el cual describe los pasos que se llevan a cabo cada que el vendedor se encuentra dentro de la zona de proximidad de un Beacon de diferente departamento al anterior, esto para mantenerlo suscrito a las notificaciones del departamento actual. Para una mejor visualización el diagrama se ha dividido en 3 partes las cuales se muestran en las figuras 5.155, 5.156 y 5.157 .

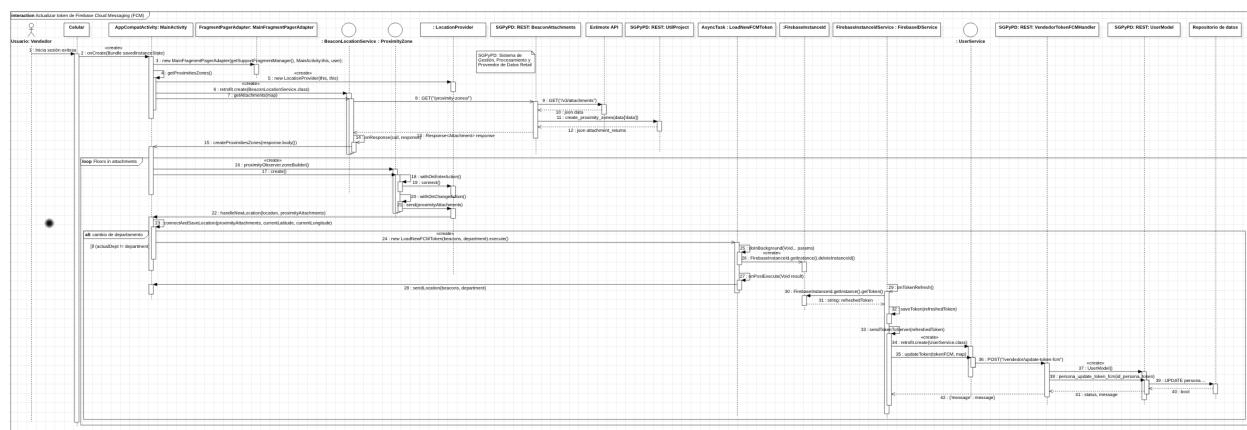


Figura 5.154: Diagrama de secuencia para actualizar el token de FCM.

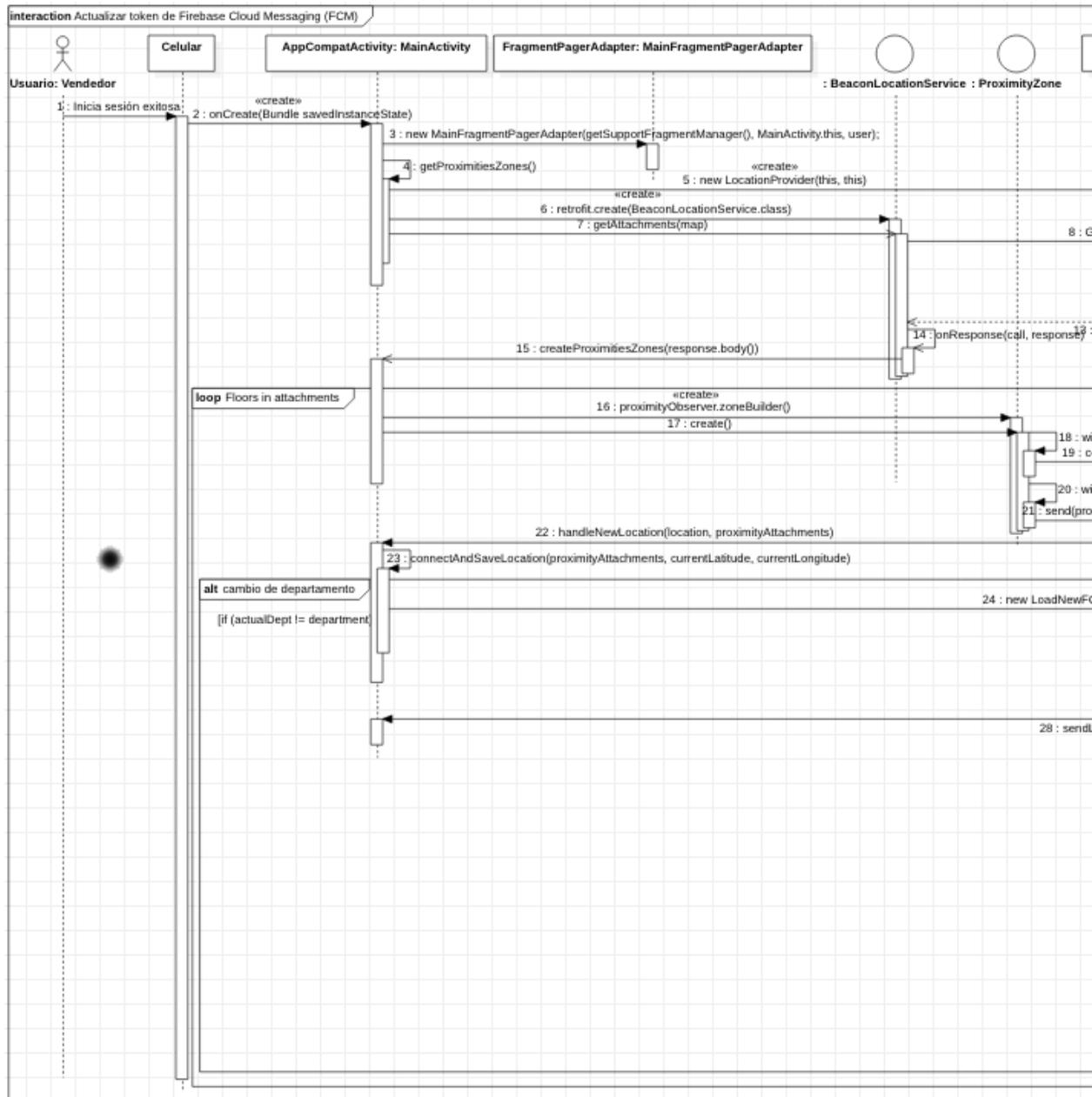


Figura 5.155: Diagrama de secuencia para actualizar el token de FCM (Parte 1).

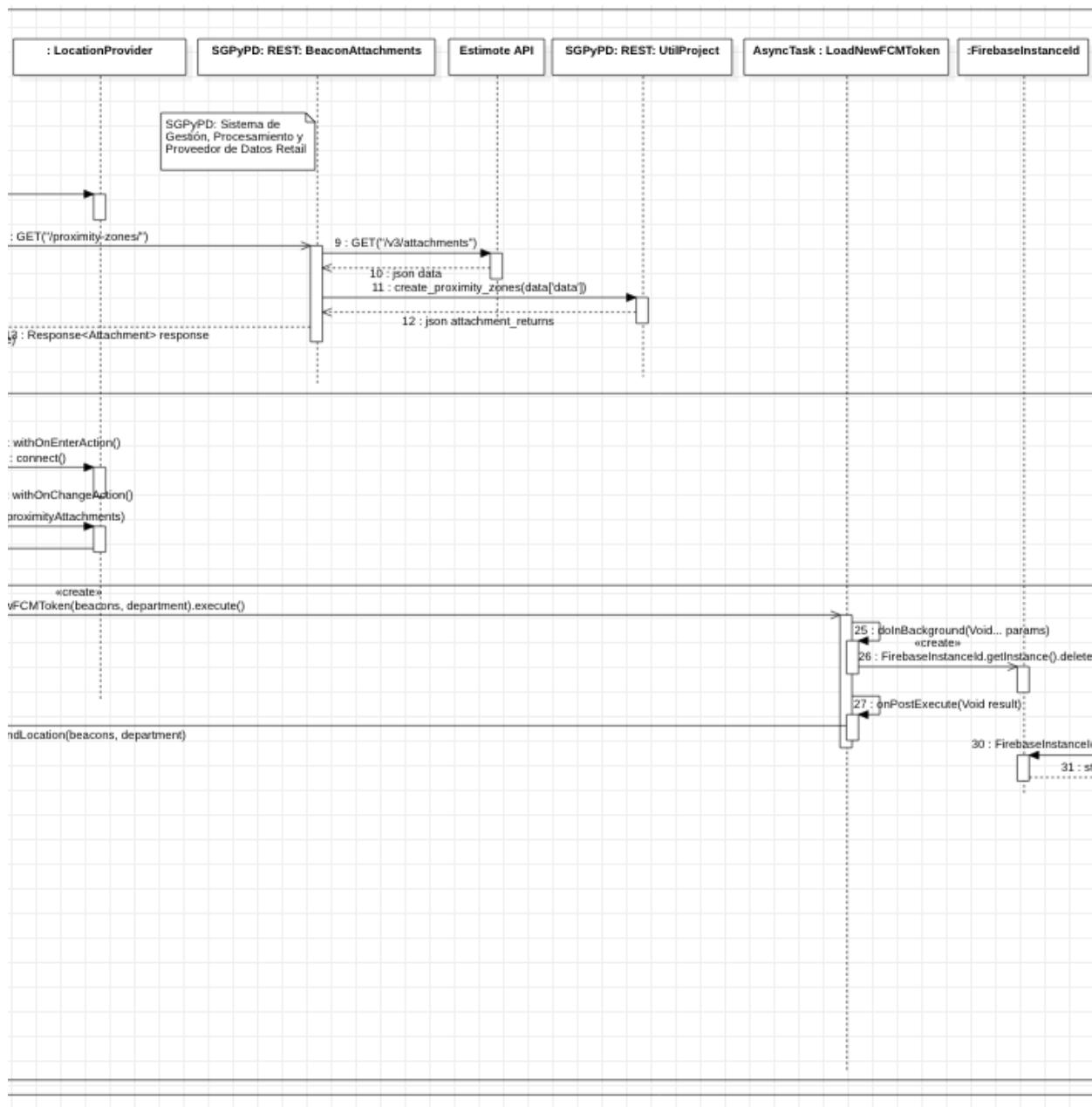


Figura 5.156: Diagrama de secuencia para actualizar el token de FCM (Parte 2).

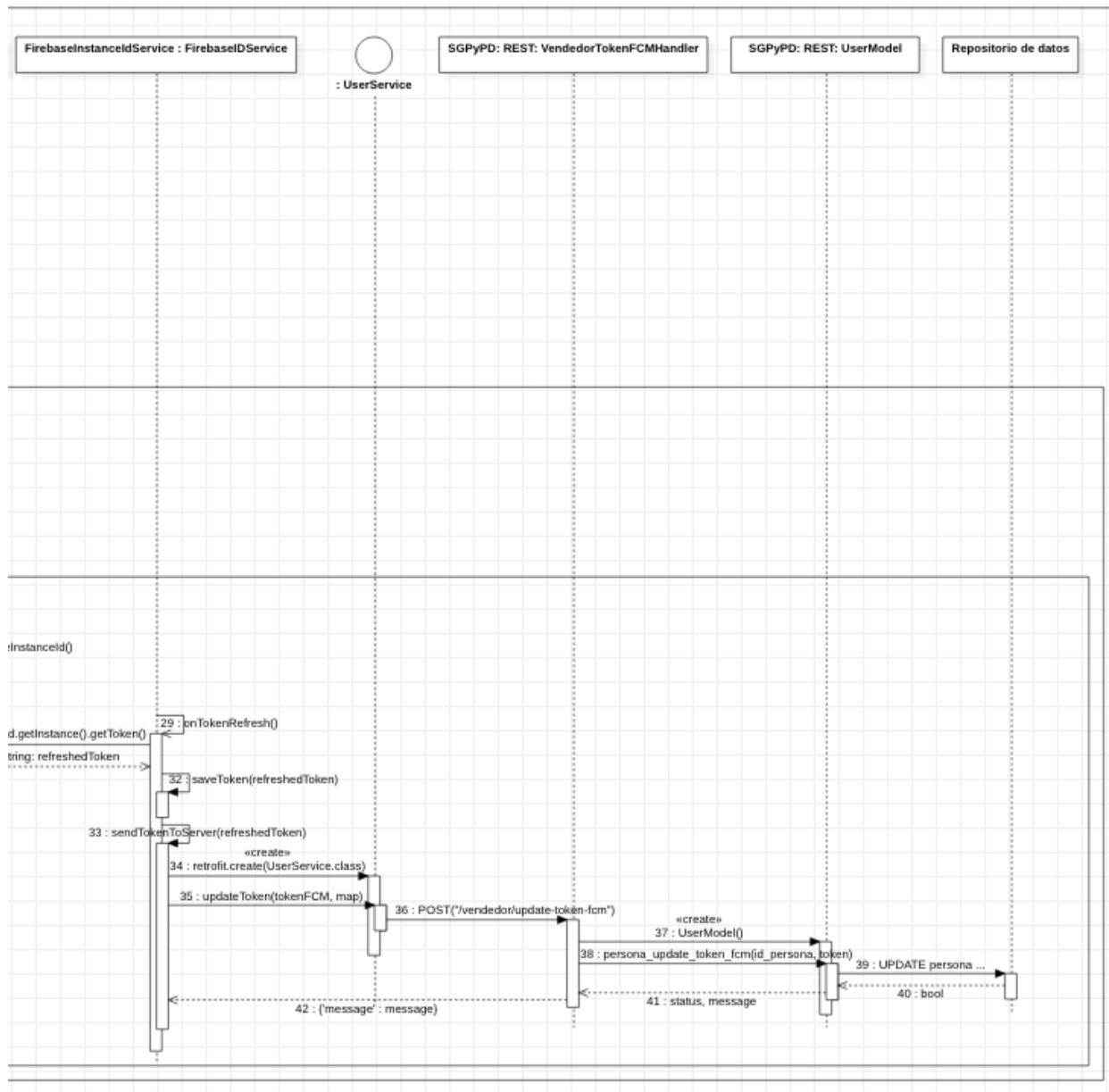


Figura 5.157: Diagrama de secuencia para actualizar el token de FCM (Parte 3).

## Flujo de navegación final de la AIPV.

La siguiente figura 5.158 muestra como es el flujo final de navegación de la aplicación.

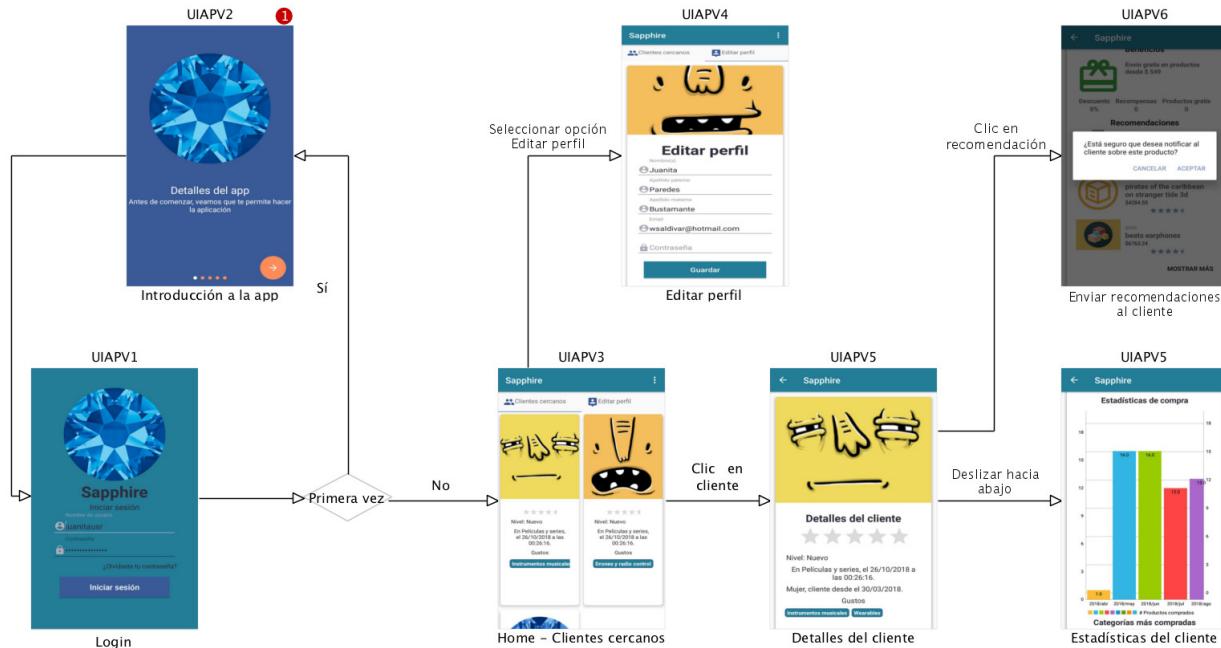


Figura 5.158: Flujo final de navegación de la AIPV.

## Interfaces de usuario

### UIAPV1 - Login

#### Objetivo

Controlar el acceso a las funcionalidades de la aplicación mediante el inicio de sesión por medio de las credenciales: nombre de usuario y contraseña.

#### Diseño

La figura 5.159 presenta la pantalla del prototipo final que aparece al iniciar la aplicación. Para lograr un inicio de sesión exitoso se debe ingresar el nombre de usuario y contraseña correctas.



Figura 5.159: UIAPVF1 Login.

### Salidas

Ninguna.

### Entradas

Nombre de usuario y contraseña.

### UIAPV2 - Introducción a la aplicación

#### Objetivo

Indicar al usuario las funcionalidades que tiene la aplicación y darle a conocer los límites de lo que puede hacer con ella.

#### Diseño

La figura 5.160 presenta la pantalla que aparece únicamente la primera vez que se inicia la aplicación, contiene 5 diapositivas en total, las siguientes se muestran en las figuras 5.161 a 5.164 .



Figura 5.160: UIAPVF2 Introducción a la aplicación.

### Salidas

Ninguna.

### Entradas

Ninguna.



Figura 5.161: Introducción a la aplicación - Ver clientes cercanos



Figura 5.162: Introducción a la aplicación - Perfil de compra de los clientes



Figura 5.163: Introducción a la aplicación - Más sobre el cliente



Figura 5.164: Introducción a la aplicación - Editar perfil

#### UIAPV3 - Home: Clientes cercanos

##### Objetivo

Mostrar al usuario vendedor una lista de los clientes cercanos para ofrecerles recomendaciones de productos y/o una atención personalizada.

### Diseño

La figura 5.165 presenta la pantalla principal que se muestra después de haber iniciado correctamente sesión, en la parte superior tiene dos opciones, de izquierda a derecha: Ver clientes y Editar perfil. Esta vista se muestra cuando no hay clientes cercanos, para actualizar la vista el usuario vendedor debe acercarse a un Beacon, lo cual hará que se obtengan los más recientes y se mostrará la notificación 5.167.



No hay clientes cercanos,  
☺ intente acercandose a un ⚡ Beacon para sincronizar.

Figura 5.165: UIAPVF3 Home: Clientes cercanos (Vacía).

### Salidas

Ninguna.

### Entradas

Ninguna.

### UIAPV3-1 - Home: Clientes cercanos (con datos)

### Objetivo

Mostrar al usuario vendedor una lista de los clientes cercanos para ofrecerles recomendaciones de productos y/o una atención personalizada.

### Diseño

La figura 5.166 presenta la pantalla principal 5.165 con una lista de clientes cercanos, en la cual cada uno contiene una imagen, su puntaje en forma de estrellas del 1 al 5, su nivel, en que departamento y a que hora se encontraba cerca, así como una lista de sus gustos.

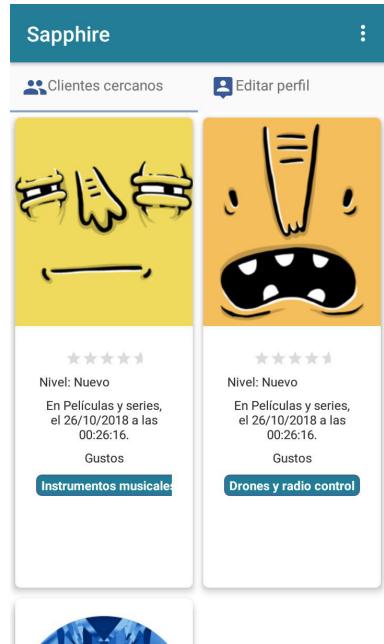


Figura 5.166: UIAPVF3-1 Home: Clientes cercanos .

### Salidas

Ninguna.

### Entradas

Ninguna.

### UIAPV3 - Notificaciones

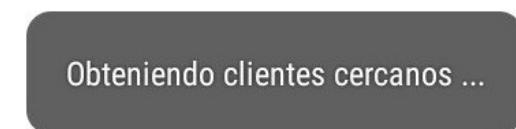


Figura 5.167: NAPV1 Notificación clientes cercanos.

### UIAPV4 - Editar perfil

### Objetivo

Permitir al usuario vendedor editar sus datos básicos (nombre(s), apellido paterno y materno, email y contraseña).

### Diseño

La figura 5.168 presenta la pantalla que se muestra al seleccionar la segunda opción del menú superior de la aplicación (de izquierda a derecha) “Editar perfil”. Nota: La contraseña se puede dejar en blanco para no modificarla. Al editar los datos y dar clic en el botón “Guardar” se muestra la notificación 5.169



Figura 5.168: UIAPVF4 Editar perfil.

### Salidas

Ninguna.

### Entradas

Nombre(s), apellido paterno y materno, email y contraseña.

### UIAPV4 - Notificaciones

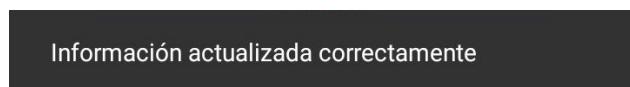


Figura 5.169: NAPV2 Notificación confirmación de actualización de datos.

### UIAPV5 - Detalles del cliente

### Objetivo

Mostrar información relevante del cliente que permita ofrecerle una atención personalizada, además de permitirle enviar recomendaciones del departamento en el que se encuentre.

### Diseño

La figura 5.170 presenta la pantalla que aparece al dar clic en un cliente de la lista de clientes cercanos, se muestran los datos de acuerdo a los permisos que el cliente haya otorgado desde la AIDP. Al iniciar esta pantalla se muestra la notificación 5.173 seguida de la notificación 5.174.



Figura 5.170: UIAPVF5 Detalles del cliente.

### Salidas

Ninguna.

### Entradas

Ninguna.

### UIAPV5 - Detalles del cliente (Beneficios y recomendaciones)

### Objetivo

Mostrar los beneficios del cliente en una lista horizontal de acuerdo a su nivel, además de productos del departamento actual que el vendedor podría recomendarle.

### Diseño

La figura 5.171 presenta la pantalla donde se muestran los beneficios en una lista horizontal y las recomendaciones disponibles para el cliente.



Figura 5.171: UIAPVF5-2 Detalles del cliente: Beneficios y recomendaciones.

### Salidas

Ninguna.

### Entradas

Ninguna.

### UIAPV5 - Detalles del cliente (Productos favoritos)

#### Objetivo

Mostrar los productos favoritos del cliente para que el vendedor tenga en cuenta que productos recomendarle y/o ofrecerle en el caso que esto fuera requerido.

#### Diseño

La figura 5.171 presenta la pantalla donde se muestran los productos favoritos en una lista horizontal.

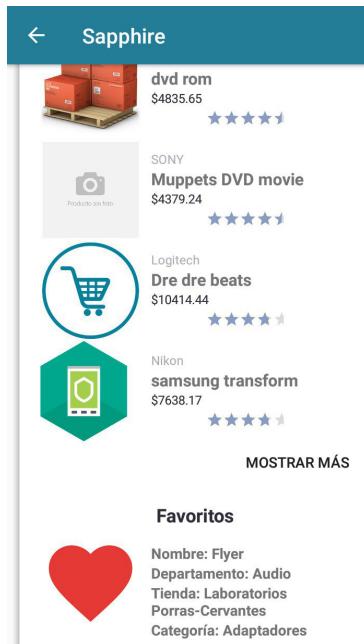


Figura 5.172: UIAPVF5-3 Detalles del cliente: Productos favoritos.

### Salidas

Ninguna.

### Entradas

Ninguna.

### UIAPV5 - Notificaciones

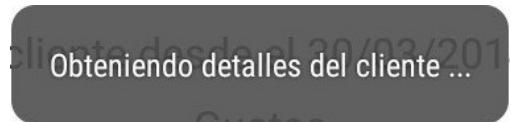


Figura 5.173: NAPV3 Notificación detalles del cliente.



Figura 5.174: NAPV4 Notificación recomendaciones para el cliente.

## UIAPV6 - Confirmación de envío de recomendación al cliente

### Objetivo

Permite al usuario vendedor enviar a un cliente una recomendación de un producto en particular del departamento en el que se encuentre.

### Diseño

La figura 5.128 se presenta el dialogo de confirmación que se muestra al dar clic sobre uno de los productos de la lista de recomendaciones para el cliente. Al aceptar el envío de la recomendación se muestra en la parte inferior de la pantalla la notificación 5.176 y al finalizar de enviarse se muestra la notificación 5.177.

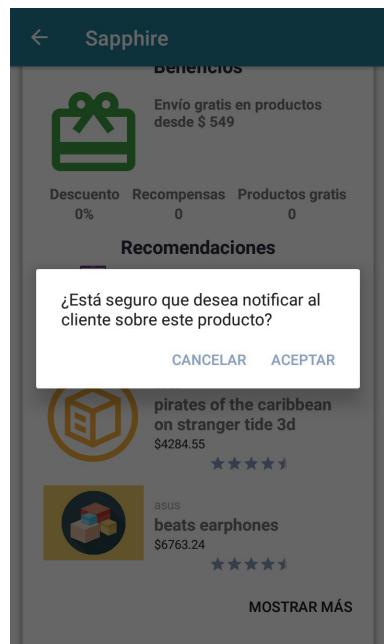


Figura 5.175: UIAPVF6 Enviar recomendaciones al cliente.

### Salidas

Ninguna.

### Entradas

Ninguna.

#### UIAPV6 - Notificaciones

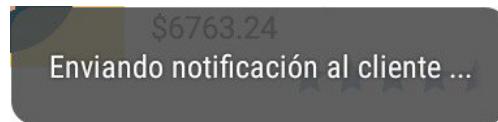


Figura 5.176: NAPV5 Notificación recomendación en proceso de envío.

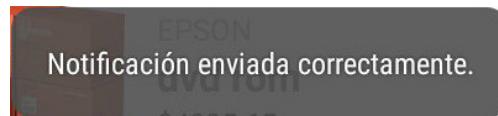


Figura 5.177: NAPV6 Notificación recomendación enviada.

#### UIAPV7 - Estadísticas del cliente

##### Objetivo

Mostrar al usuario vendedor las estadísticas de compra de un cliente de los últimos 6 meses.

##### Diseño

La figura 5.178 presenta la gráfica que se muestra al deslizar la pantalla 5.170 hacia abajo.

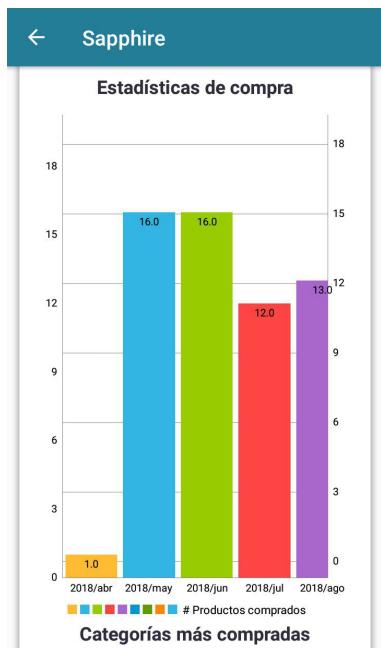


Figura 5.178: UIAPVF7 Estadísticas del cliente.

### Salidas

Ninguna.

### Entradas

Ninguna.

### UIAPV7 - Estadísticas del cliente (Categorías más compradas)

#### Objetivo

Mostrar al usuario vendedor las categorías que el cliente más ha comprado en los últimos 6 meses.

#### Diseño

La figura 5.179 presenta la gráfica que se muestra al deslizar la pantalla 5.170 hacia abajo.



Figura 5.179: UIAPVF8 Categorías más compradas.

### Salidas

Ninguna.

### Entradas

Ninguna.

# CAPÍTULO 6

---

## Servidores

---

Dentro de este capítulo se muestran las aplicaciones que se ejecutan dentro de servidores Web, se seleccionaron dos servidores, Tornado y Node.js. Un pequeño resumen de cada uno podrá ser encontrado en sus respectivas secciones.

### 6.1. Node.js con Express

Node.js® es una construcción en tiempo de ejecución JavaScript creado en el motor de JavaScript V8 de Chrome. Node.js utiliza un modelo de E/S no bloquante y controlado por eventos que lo hace liviano y eficiente. El ecosistema de paquetes de Node.js, npm, es el mayor ecosistema de bibliotecas de código abierto del mundo [105]. El logotipo de Node.js se encuentra en la figura 6.1

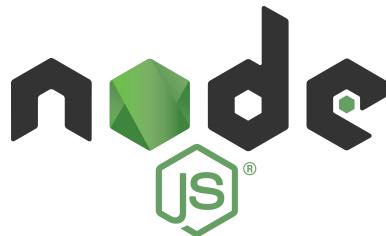


Figura 6.1: Logo de Node.js. [105].

Express.js, o simplemente Express, es un framework Web para Node.js, lanzado como software libre y de código abierto bajo la Licencia MIT. Está diseñado para crear aplicaciones Web y API. Express ha sido llamado como framework base estándar para Node.js [106]. El respectivo logotipo del framework se puede visualizar en la figura 6.2



Figura 6.2: Logo de Express. [106].

### 6.1.1. Panel de Administración (PA)

A continuación en la figura 6.3 se presenta la arquitectura del prototipo Panel de Administración.

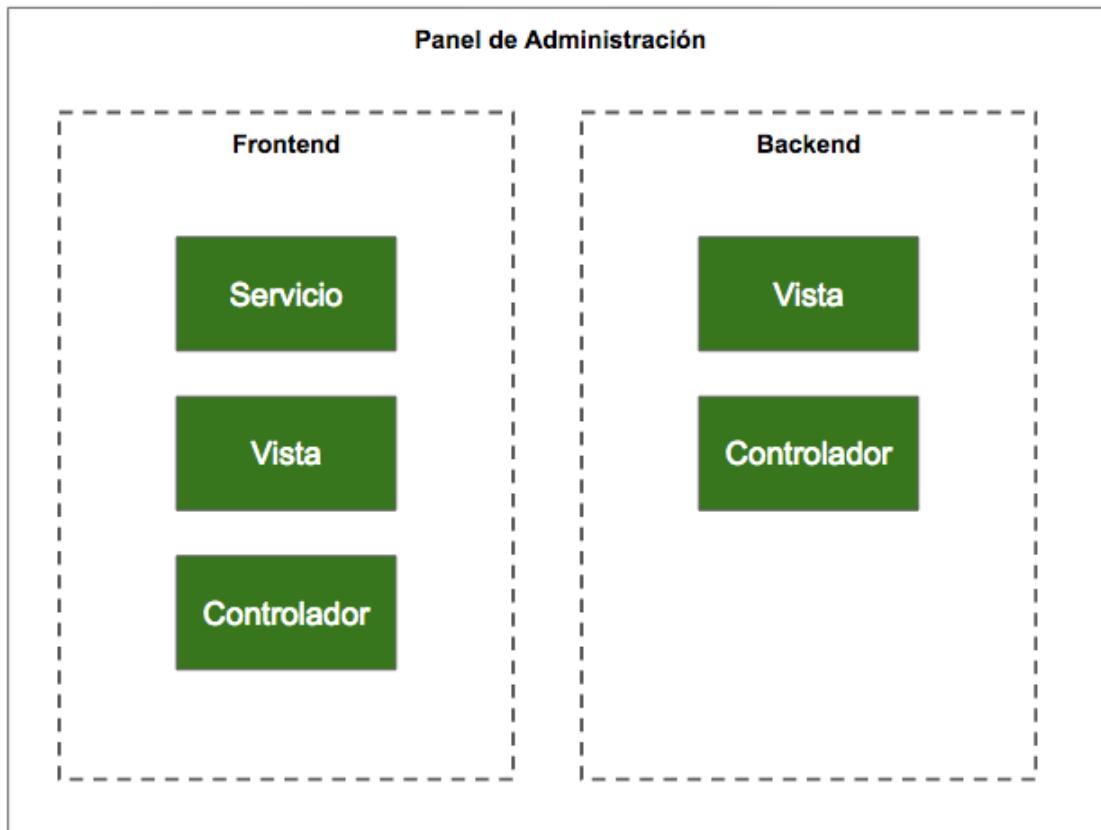


Figura 6.3: Arquitectura del módulo PA.

La descripción de cada elemento de la arquitectura es la siguiente:

- Frontend: Se enfoca en el usuario, en todo con lo que podemos interactuar y lo que vemos mientras navegamos.
  - Servicio: Se encarga de hacer peticiones HTTP a un servicio externo (API REST, microservicio, Web service)
  - Vista: Se refiere a la pantalla que le es mostrada al usuario final.
  - Controlador: Se encarga de toda la lógica de negocio de la vista.
- Backend: Está enfocado en hacer que todo lo que está detrás de un sitio web funcione correctamente.
  - Vista: Se refiere a la pantalla que le es mostrada al usuario final.
  - Controlador: Controlador: Se encarga de toda la lógica de negocio de la vista.

### 6.1.2. Prototipo 1: Diseño inicial del Panel de Administración

#### 6.1.2.1. Análisis

#### 6.1.2.2. Diseño

### Diseño de interfaz de usuario

La Figura 6.4 muestra el diseño de la interfaz de usuario que contendrá el Panel de Administración.

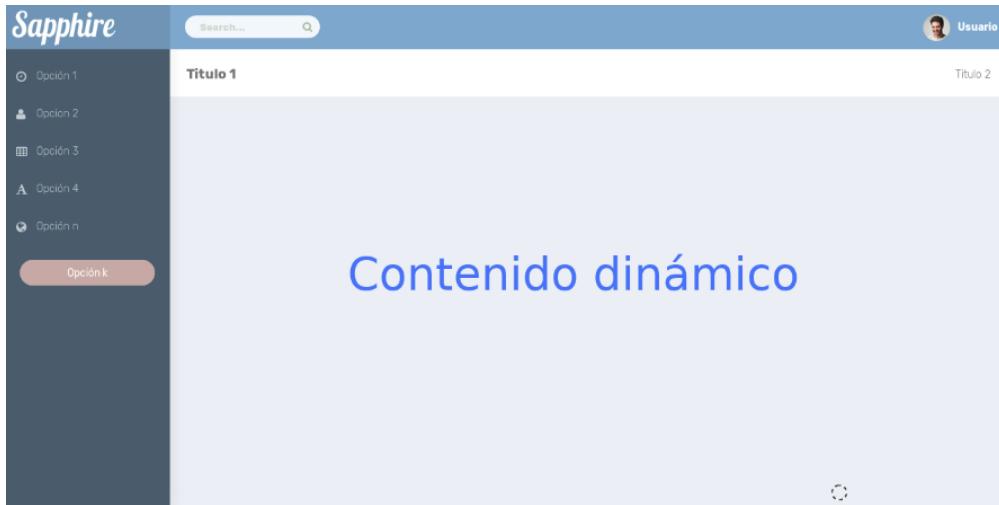


Figura 6.4: Diseño general del PA.

La Figura 6.5 muestra como se ve la interfaz de usuario del Panel de Administración desde una computadora.

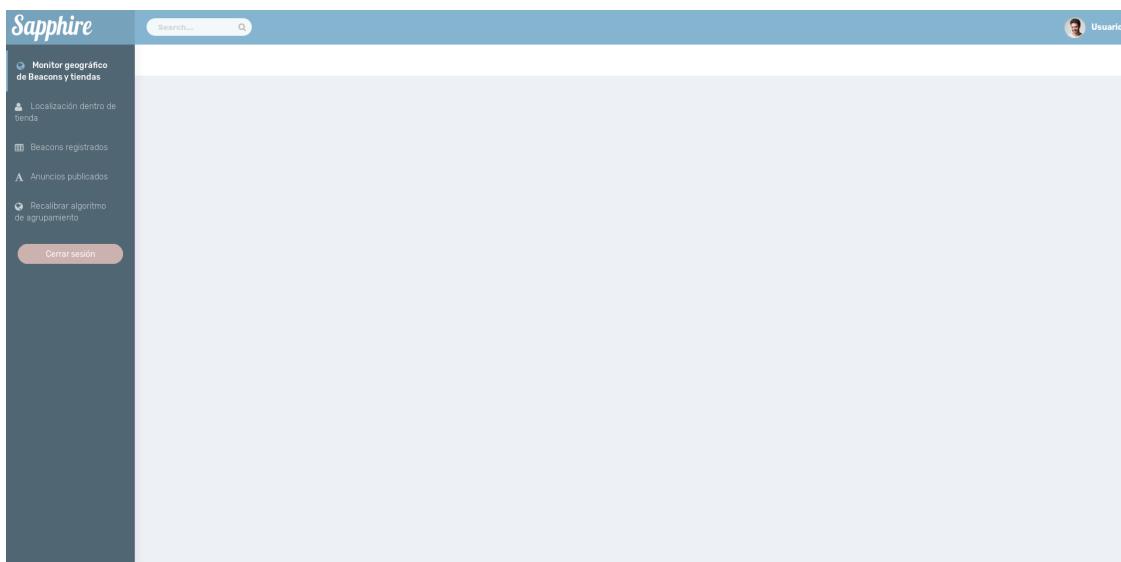


Figura 6.5: UIPanel1: Diseño computadora.

La Figura 6.6 muestra como se ve la interfaz de usuario del Panel de Administración desde una tableta.

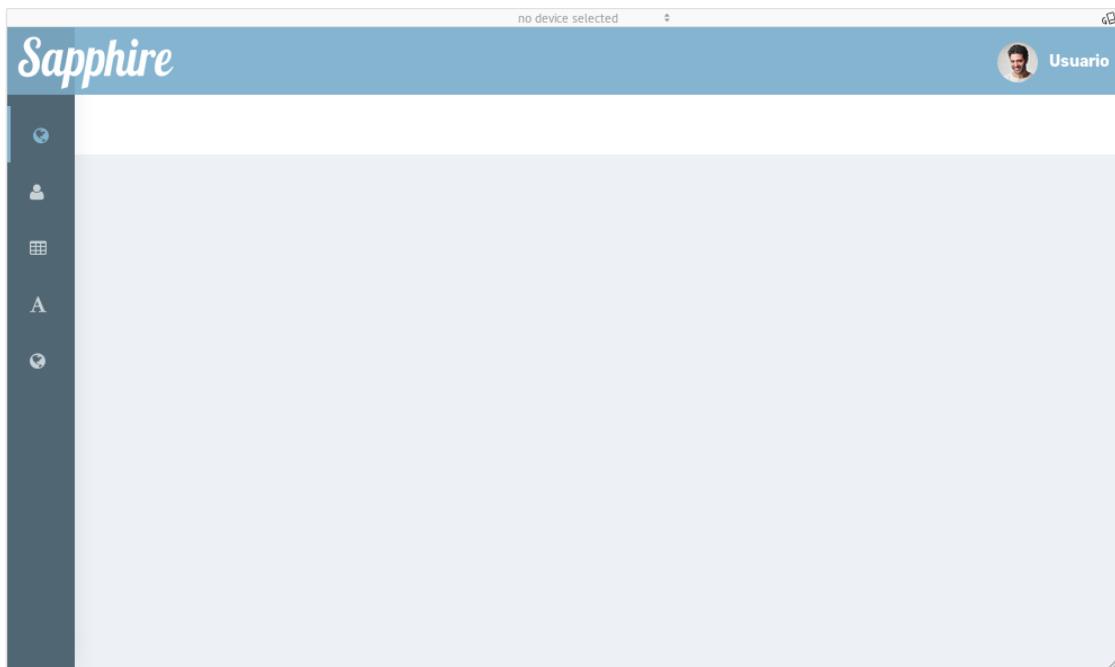


Figura 6.6: UIPanel2: Diseño tableta.

### 6.1.3. Prototipo 2: Monitor geográfico de beacons

#### 6.1.3.1. Análisis

Dentro de este prototipo se satisface el requerimiento funcional [Monitorear Beacons y tiendas](#), definido previamente en el capítulo del “Bosquejo general de la aplicación” con el título de “Requerimientos funcionales del Panel de Administración”.

#### Diagrama de casos de uso

La figura 6.36 muestra todos los casos de uso del Panel de Administración que serán realizados a lo largo de los diferentes prototipos pertenecientes al Panel de Administración.

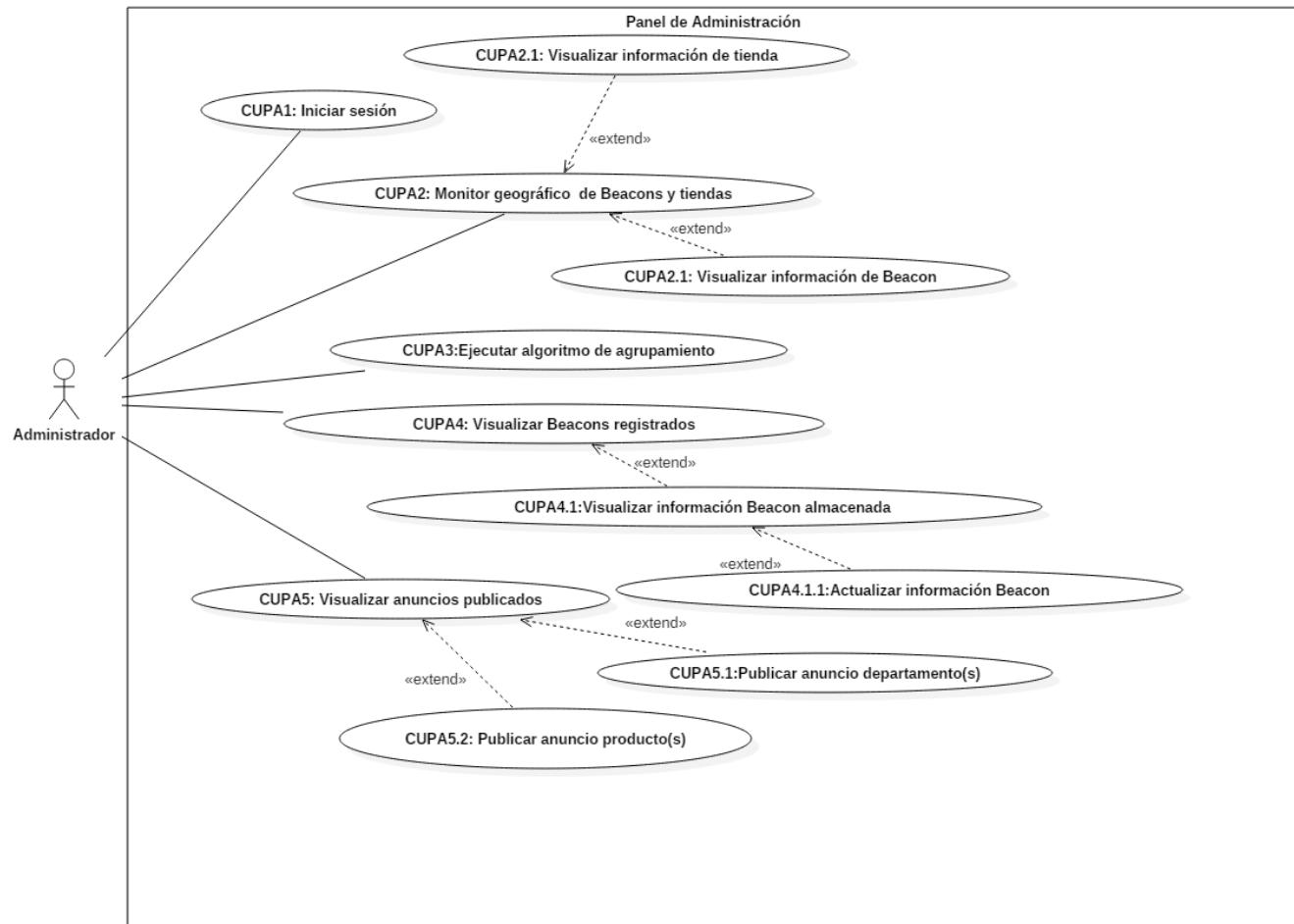


Figura 6.7: Casos de uso PA.

### 6.1.3.2. Diseño

#### Diagramas de secuencia

El diagrama de secuencia que satisface el requerimiento dentro de este prototipo se puede ver en la figura 6.8. Para una mejor visualización la figura se repartió en dos secciones que se encuentran en 6.9 y 6.10.

#### Monitor Geográfico de Beacons y tiendas

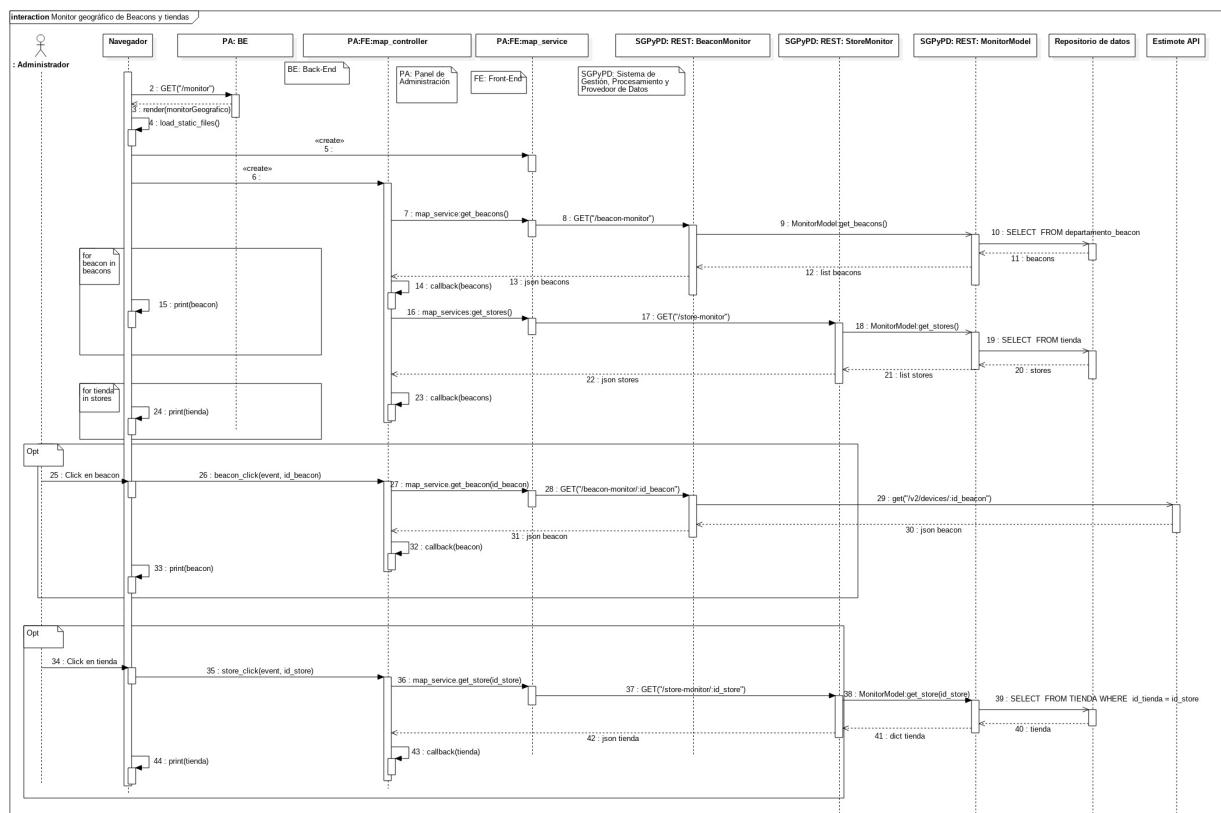


Figura 6.8: Diagrama de secuencia del monitor geográfico (Visualización completa).

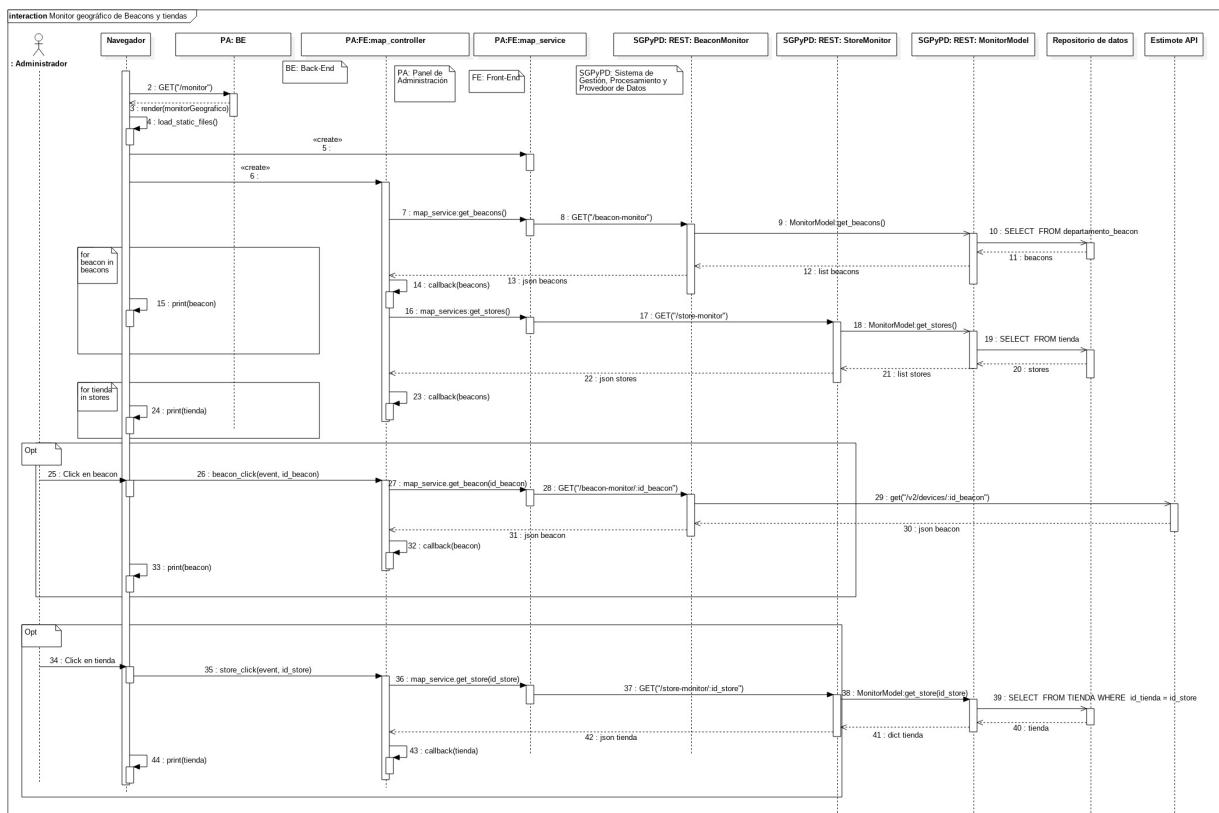


Figura 6.9: Diagrama de secuencia del monitor geográfico (Parte 1).

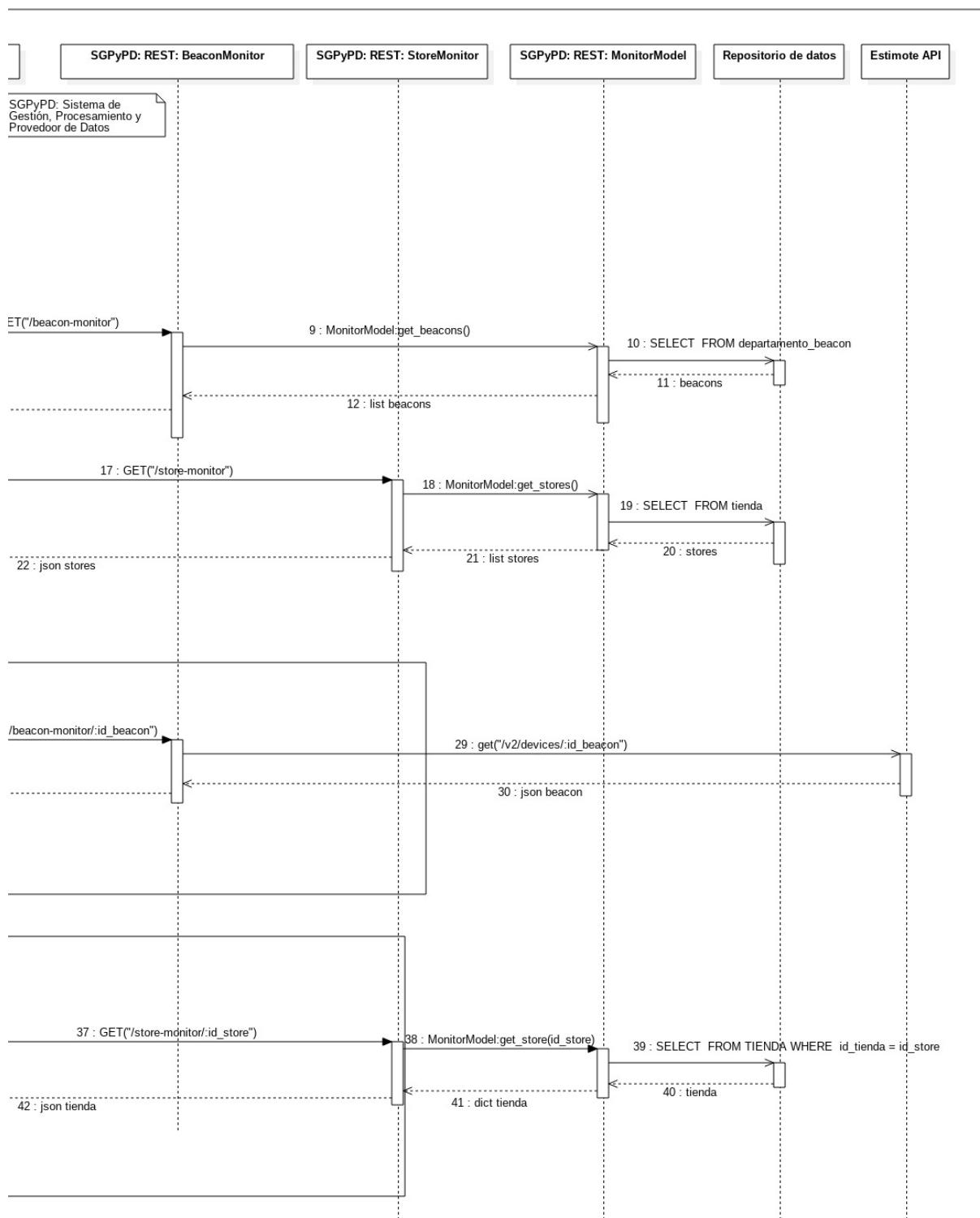


Figura 6.10: Diagrama de secuencia del monitor geográfico (Parte 2).

### Diseño de interfaz de usuario

En la imagen 6.11 se muestra el resultado de este prototipo.

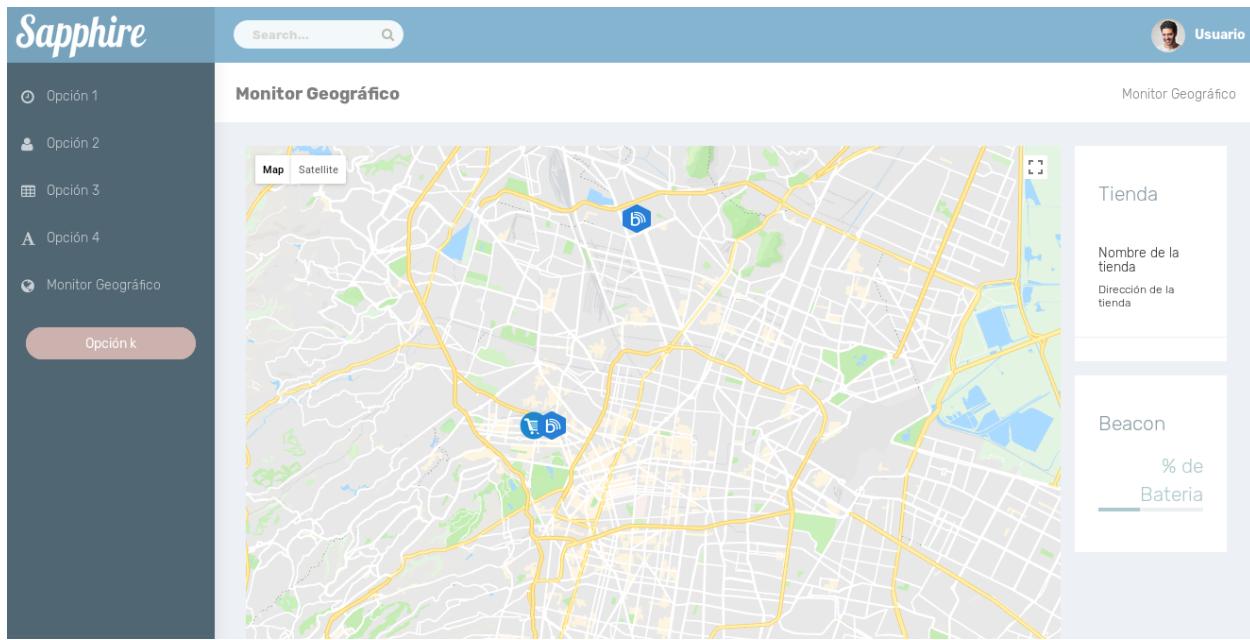


Figura 6.11: Diseño monitor geográfico.

La Figura 6.12 muestra como se ve la interfaz de usuario del Panel de Administración para el monitor de Beacons y tiendas desde una computadora.

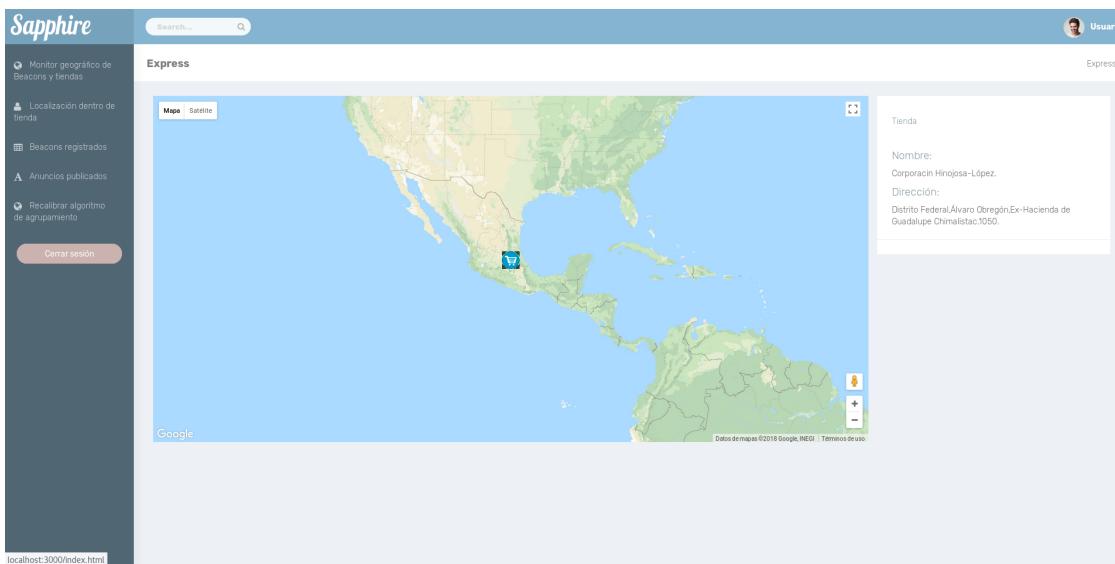


Figura 6.12: UIPanel3: Monitor Beacons y tiendas computadora.

La Figura 6.13 muestra como se ve la interfaz de usuario del Panel de Administración para el monitor de Beacons y tiendas desde una tableta.

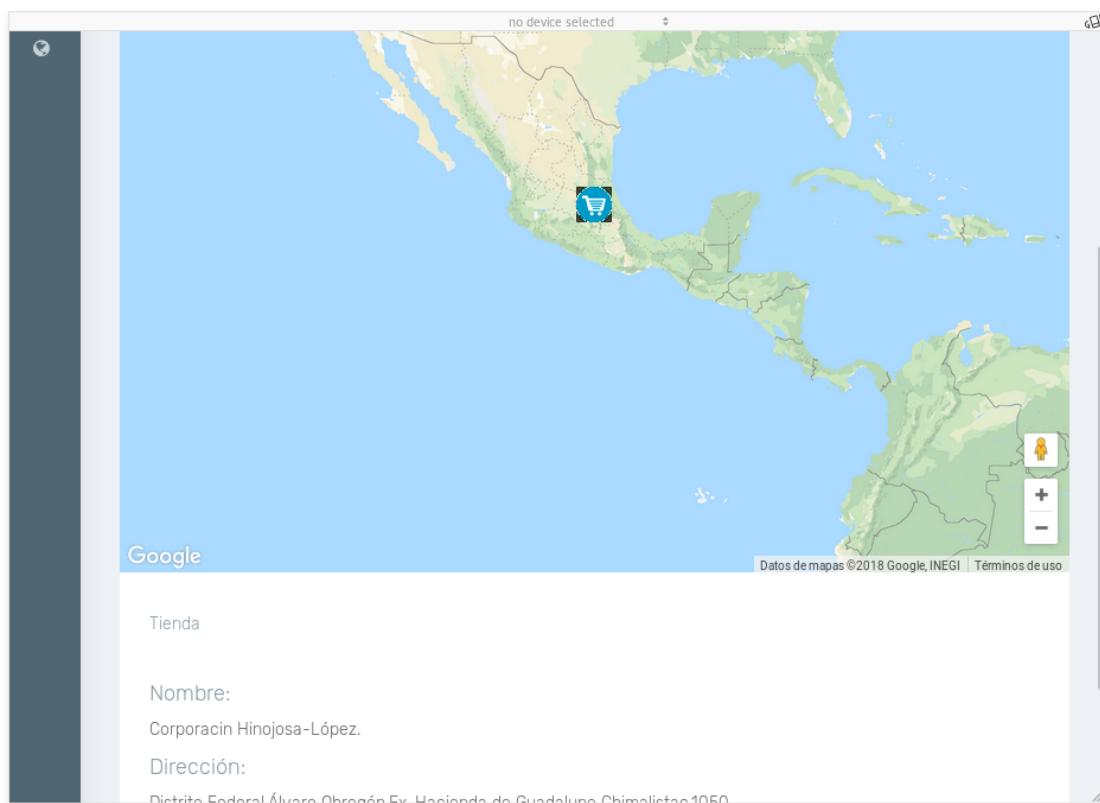


Figura 6.13: UIPanel4: Monitor Beacons y tiendas tableta.

La figura 6.42 muestra como es el flujo de navegación general del Panel de Administración.

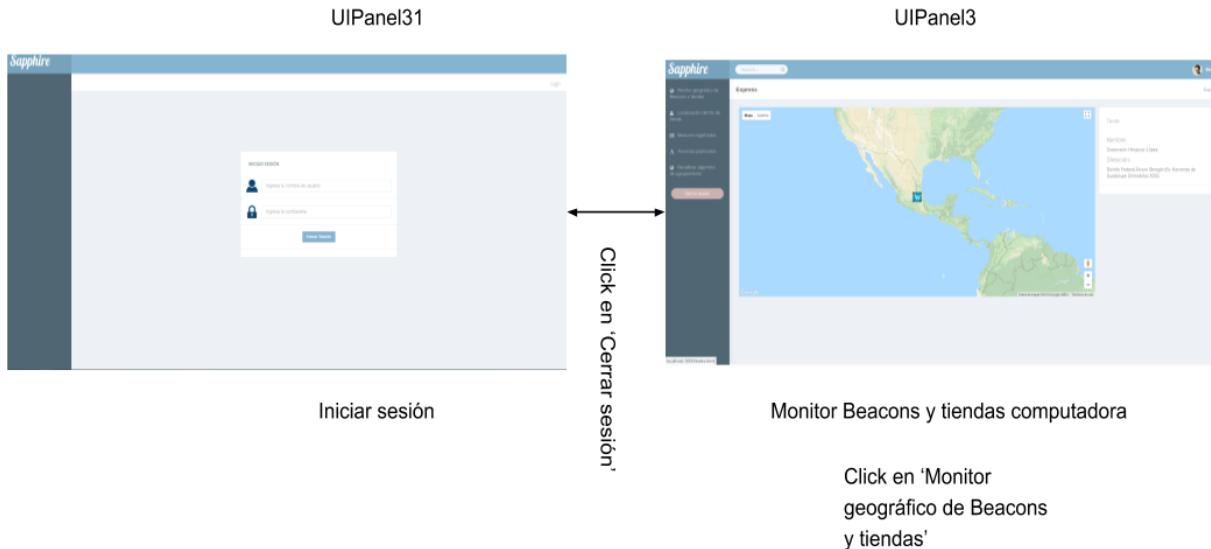


Figura 6.14: Flujo de navegación general Panel de Administración.

#### 6.1.4. Prototipo 3: Integración con servicios REST a panel de administración

##### 6.1.4.1. Análisis

Dentro de este prototipo se satisface los requerimientos funcionales [Iniciar sesión](#), [Visualizar información de anuncios publicados](#) y [Visualizar información de Beacons](#) definidos previamente en el capítulo del “Bosquejo general de la aplicación” con el título de “Requerimientos funcionales del Panel de Administración”.

##### Diagrama de casos de uso

En la figura 6.15 se puede visualizar el diagrama de casos de uso actualizado.

Modificaciones realizadas:

- Se agregó el caso de uso **Cerrar sesión** al diagrama.
- Se agregó el caso de uso **CUPA5.1:Publicar anuncio**
- Se cambió el caso de uso **CUPA5.1** del prototipo 2 a **CUPA5.1.2:Añadir departamento(s)**
- Se cambió el caso de uso **CUPA5.2** del prototipo 2 a **CUPA5.1.1:Añadir producto(s)**

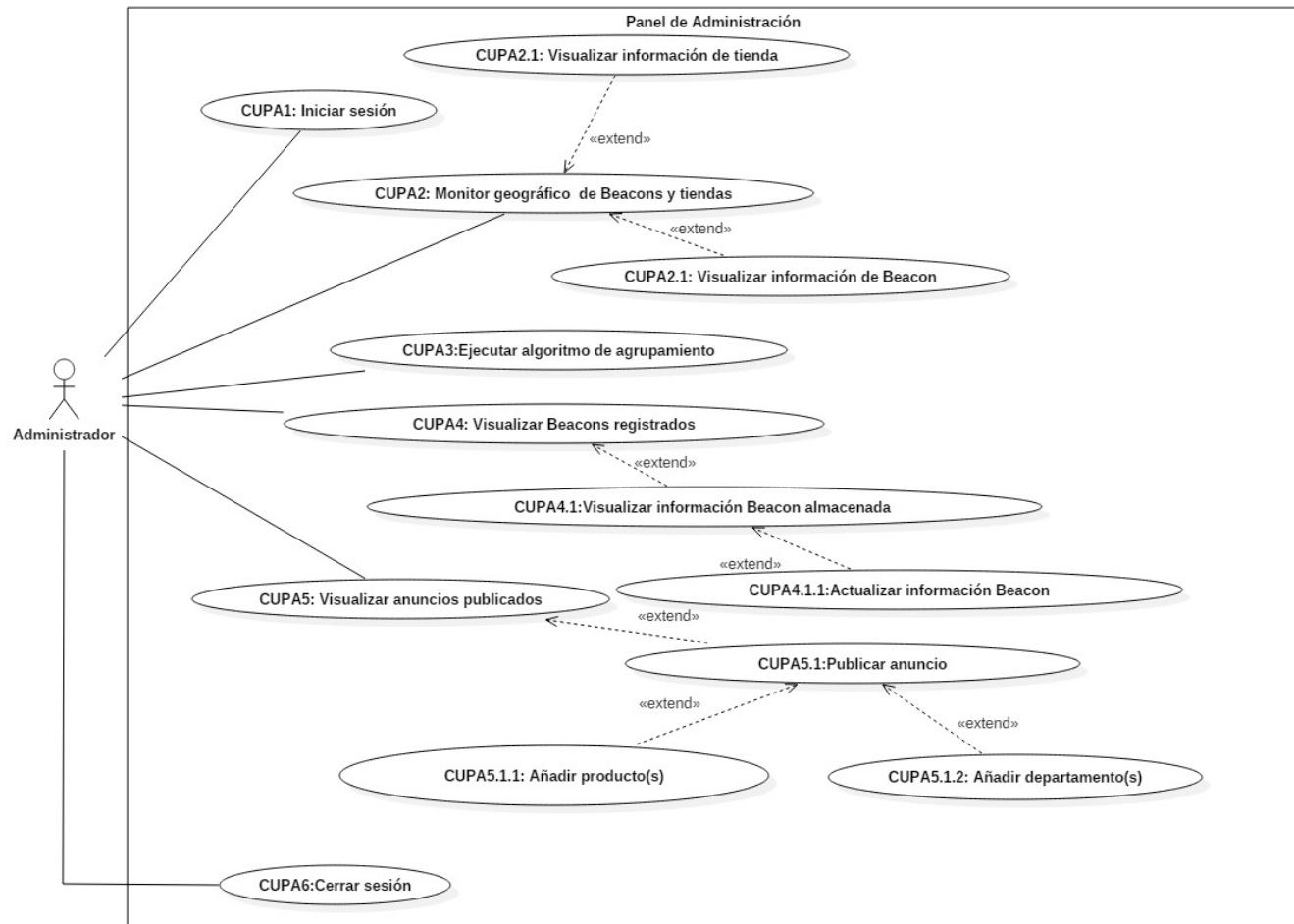


Figura 6.15: Diagrama de casos de uso.

#### 6.1.4.2. Diseño

##### Diagramas de secuencia

La figura 6.16 muestra el diagrama que satisface el caso de uso **Iniciar sesión** que se muestra en el diagrama de casos de uso del primer prototipo.

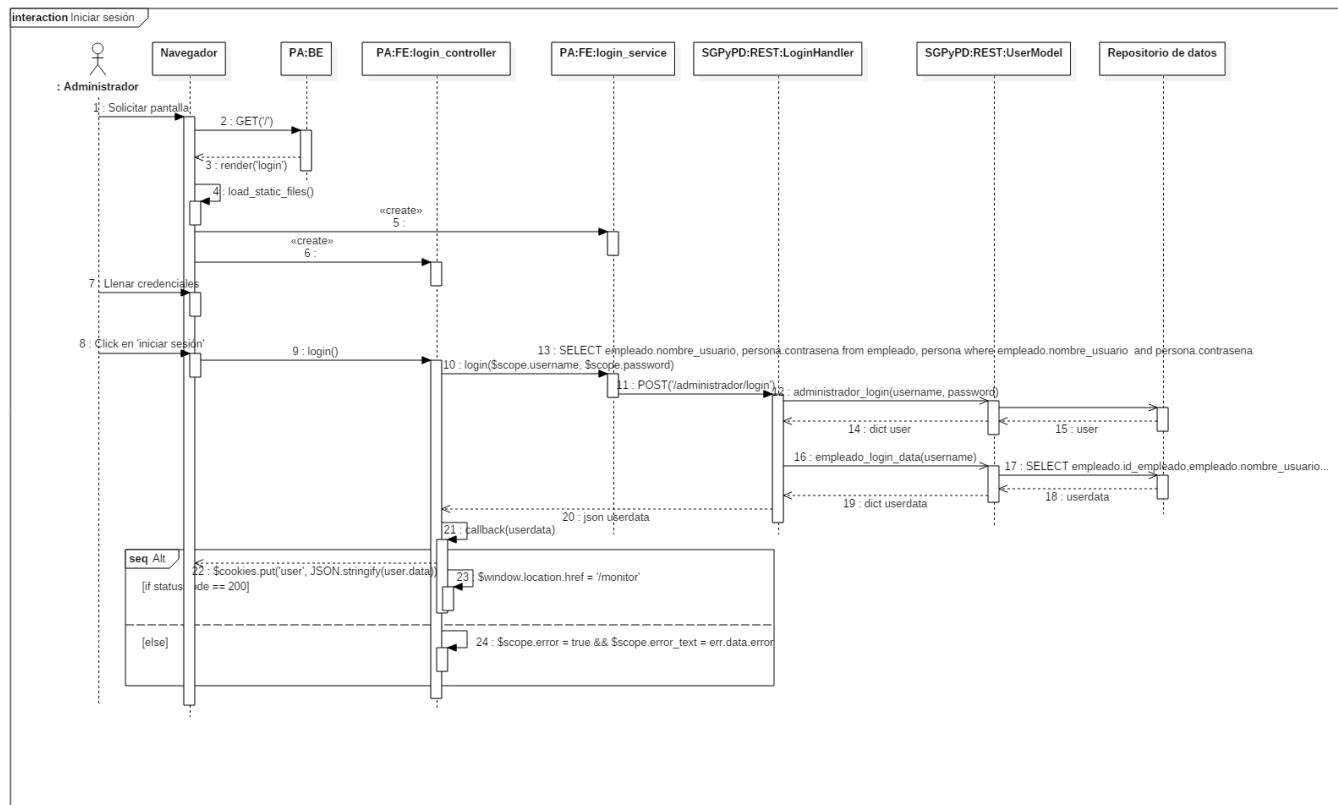


Figura 6.16: Diagrama de secuencia Iniciar sesión.

La figura 6.17 muestra el diagrama que satisface el caso de uso **Cerrar sesión** que se muestra en el diagrama de casos de uso del prototipo en cuestión.

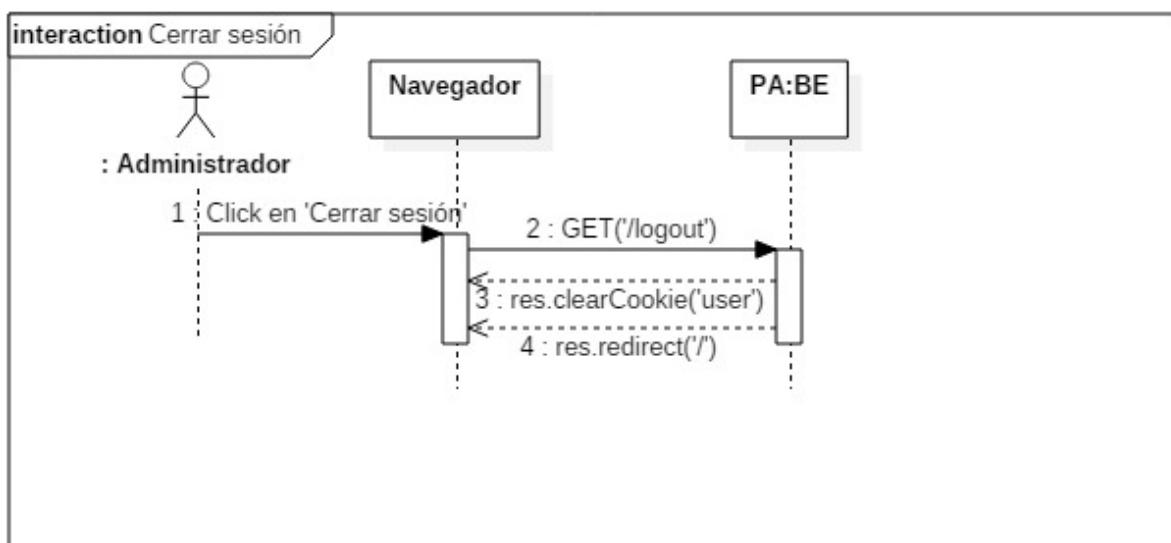


Figura 6.17: Diagrama de secuencia Cerrar sesión.

La figura 6.24 muestra el diagrama que satisface el caso de uso [Visualizar anuncios publicados](#) que se muestra en el diagrama de casos de uso.

Notas:

1. El diagrama fue separado en varias figuras para su visualización ([Figura 6.23](#), [Figura 6.19](#), [Figura 6.20](#), [Figura 6.21](#), [Figura 6.22](#)).
2. Dentro del diagrama se encuentran también los casos de uso [Publicar anuncio](#), [Añadir departamento\(s\)](#), [Añadir producto\(s\)](#)

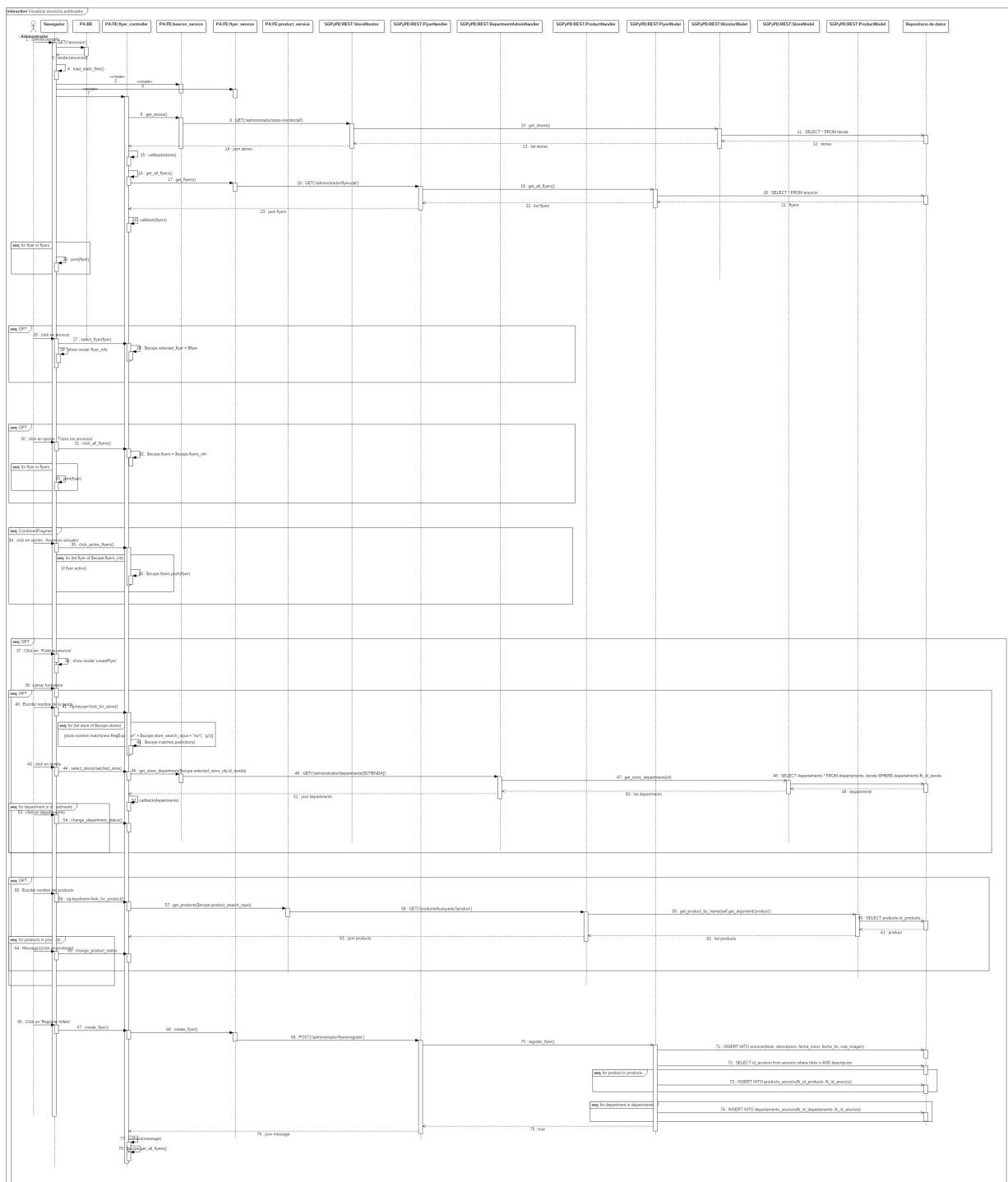


Figura 6.18: Diagrama de secuencia Visualizar anuncios publicados (Visualización completa).

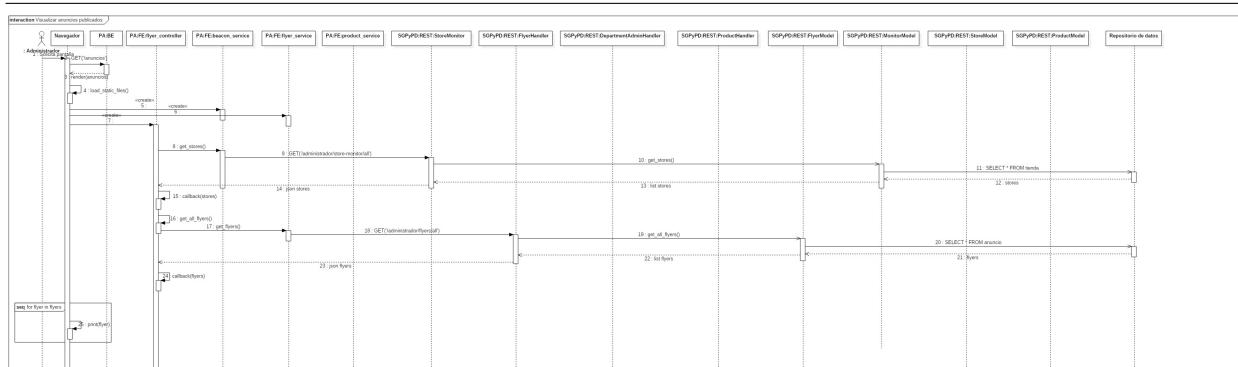


Figura 6.19: Diagrama de secuencia Visualizar anuncios publicados (Parte 1).

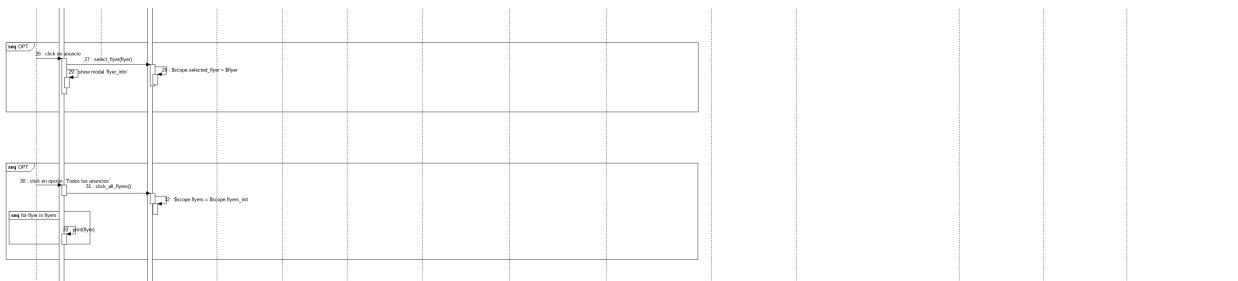


Figura 6.20: Diagrama de secuencia Visualizar anuncios publicados (Parte 2).

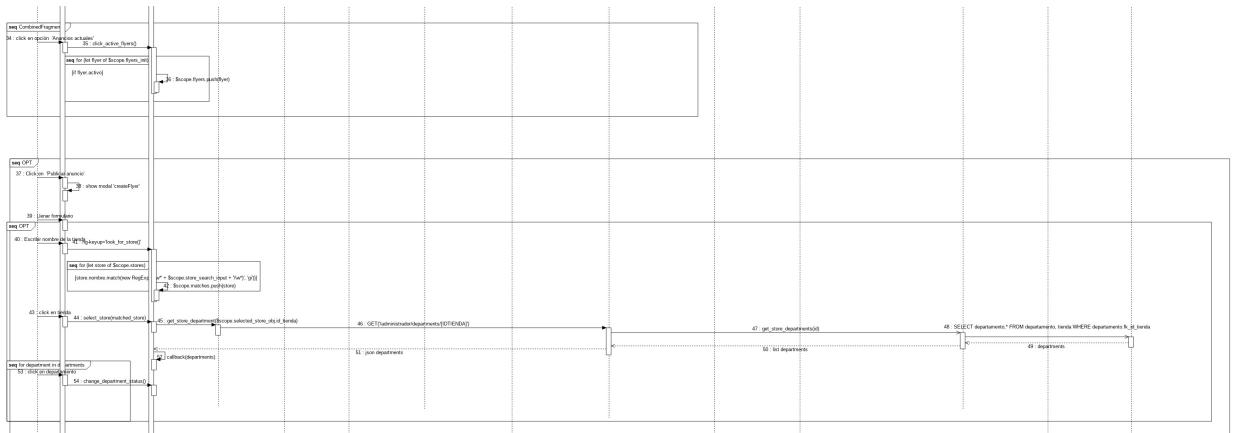


Figura 6.21: Diagrama de secuencia Visualizar anuncios publicados (Parte 3).

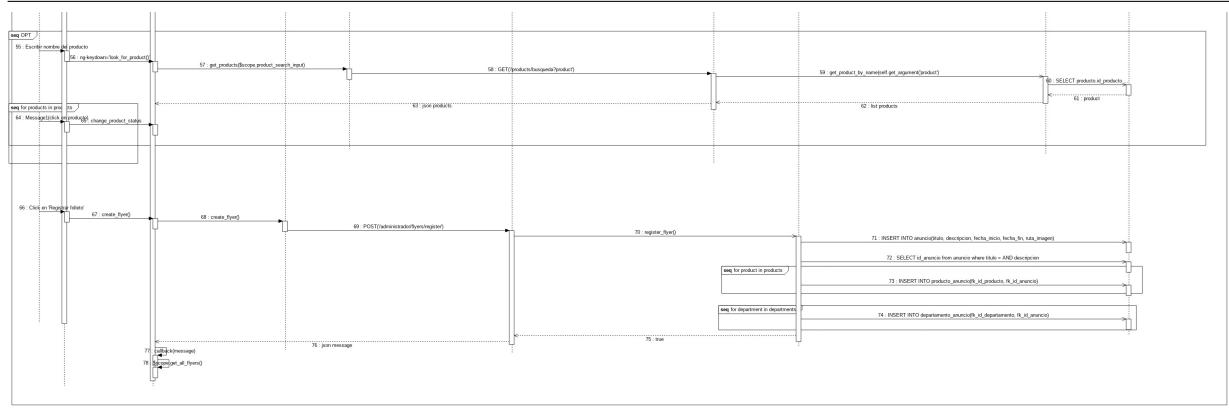


Figura 6.22: Diagrama de secuencia Visualizar anuncios publicados (Parte 4).

La figura 6.24 muestra el diagrama satisfactorio para el caso de uso **Visualizar Beacons registrados** que se muestra en el diagrama de casos de uso.

Notas:

1. El diagrama fue separado en varias figuras para su visualización (Figura 6.24, Figura 6.26, Figura 6.28).
2. Dentro del diagrama se encuentran también los casos de uso **Visualizar información de Beacon almacenada**, **Actualizar información de Beacon**.

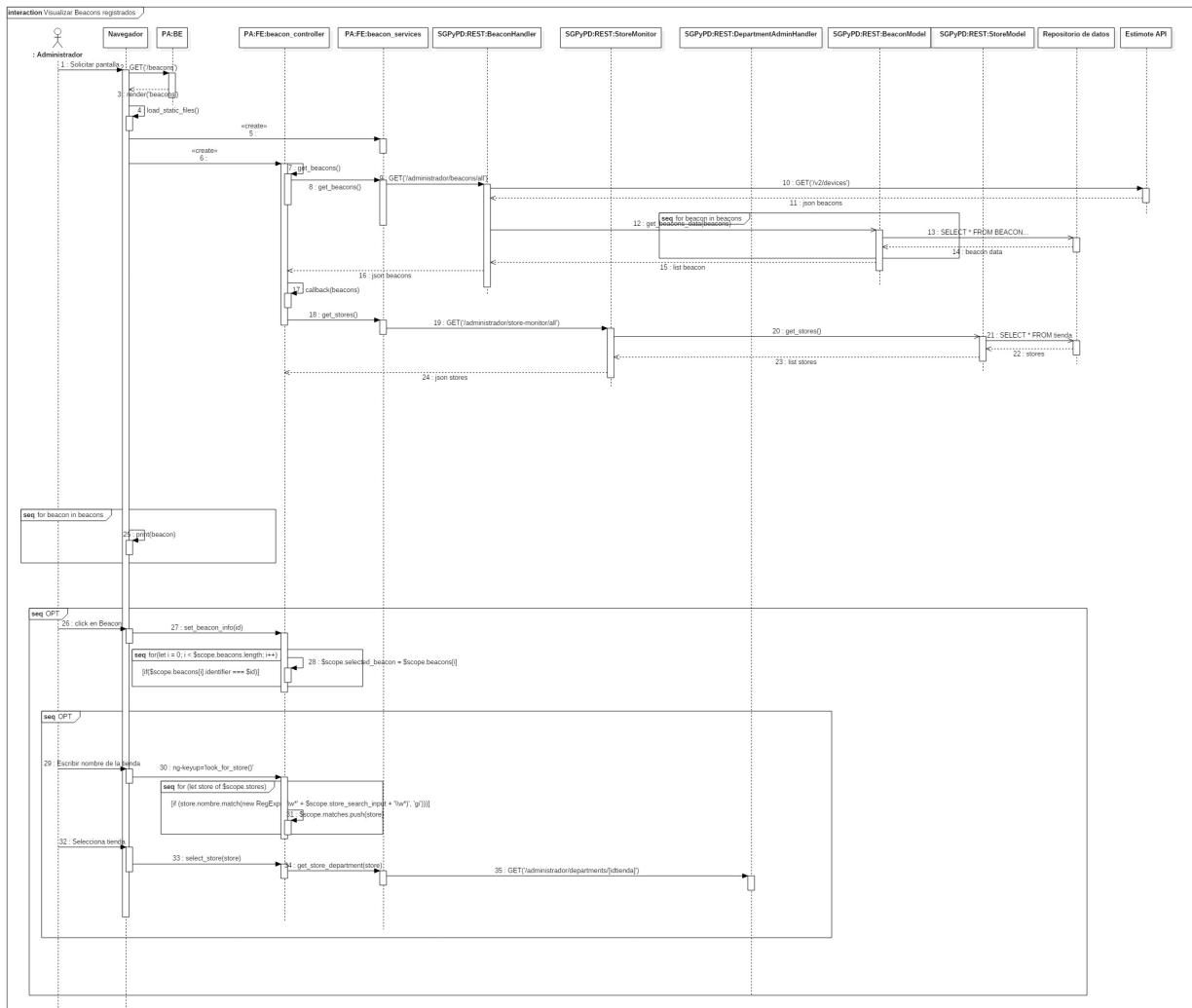


Figura 6.23: Diagrama de secuencia Visualizar Beacons registrados (Visualización completa).

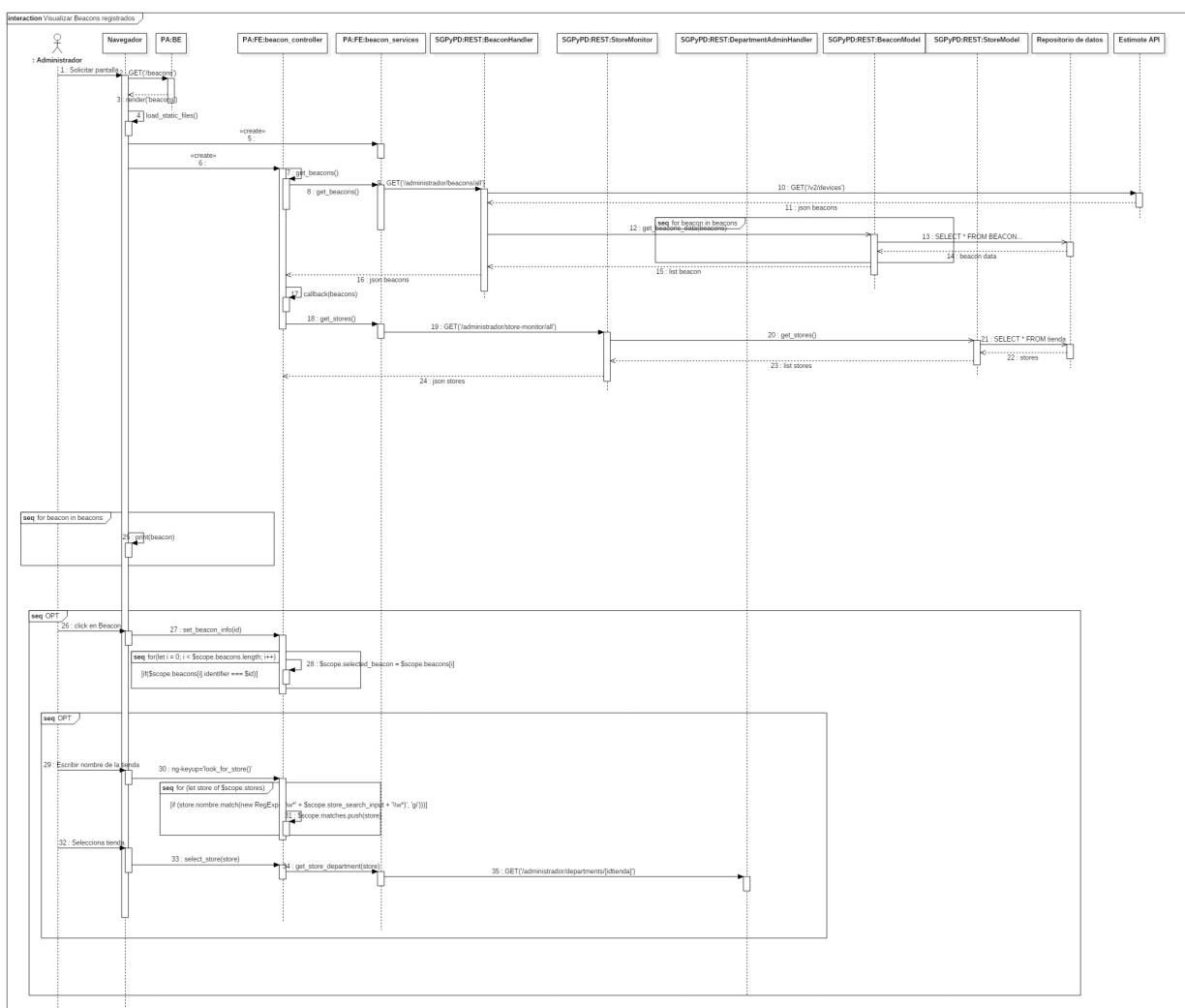


Figura 6.24: Diagrama de secuencia Visualizar Beacons registrados (Visualización completa).

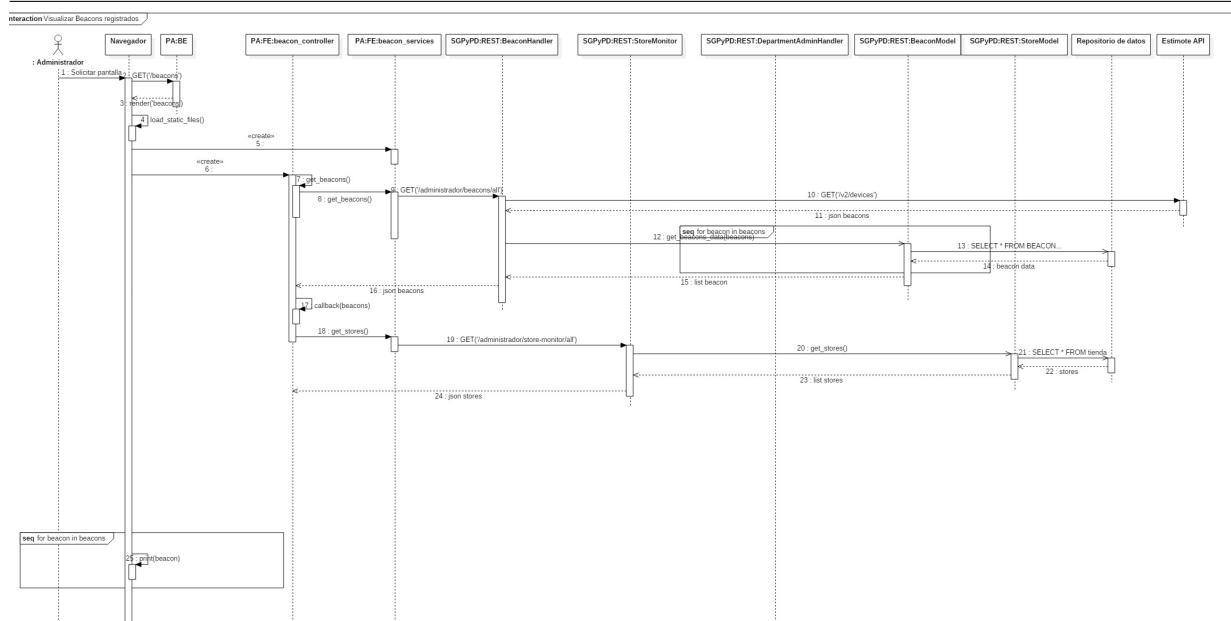


Figura 6.25: Diagrama de secuencia Visualizar Beacons registrados (Parte 1).

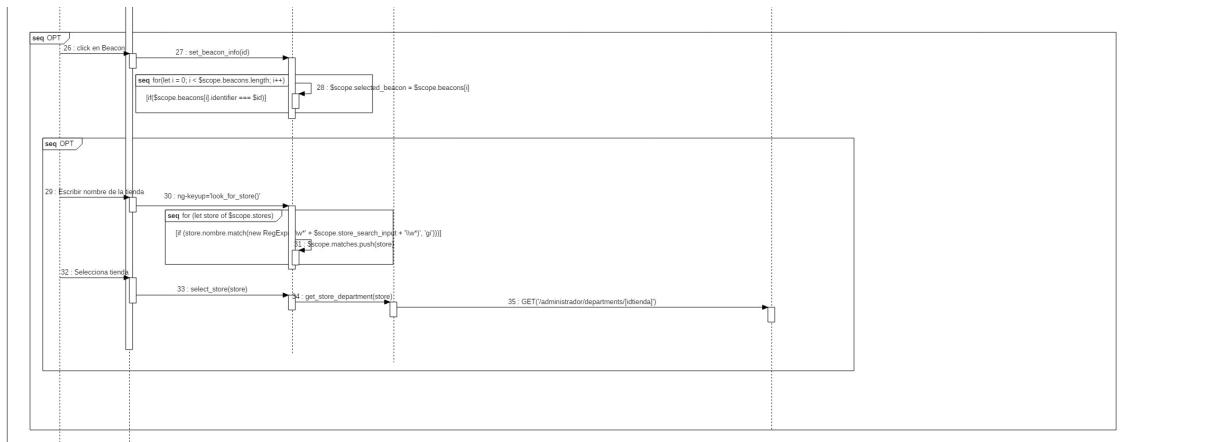


Figura 6.26: Diagrama de secuencia Visualizar Beacons registrados (Parte 2).

## Diseño de interfaz de usuario

En la Figura 6.27 se muestra la interfaz de usuario que satisface el requerimiento Iniciar sesión.

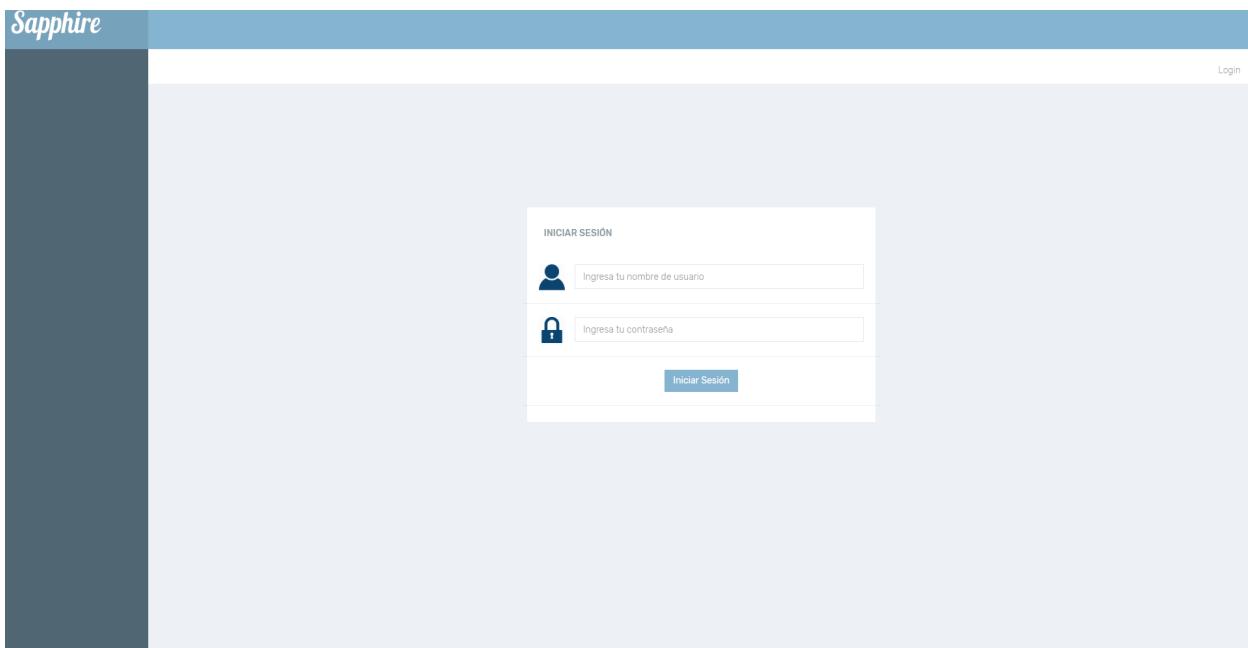


Figura 6.27: UIPanel31: Iniciar sesión.

En la Figura 6.28 se muestra la interfaz de usuario que satisface el requerimiento [Visualizar información de anuncios publicados](#).

Título	Descripción	Fecha de inicio	Fecha de fin
Venta nocturna	¡Gran venta nocturna!	2018/04/09 00:00	2018/04/09 00:00
Venta 50	¡Todos los productos al 50%!	2018/04/09 00:00	2018/04/09 00:00
Productos con 20% de descuento	¡Productos seleccionados con 20%!	2018/04/09 00:00	2018/04/09 00:00
Departamentos al 30%	¡Departamentos participantes con 30%!	2018/04/09 00:00	2018/04/09 00:00

Figura 6.28: UIPanel32: Anuncios publicados.

En la Figura 6.29 se muestra la pantalla que muestra la información detallada de un anuncio que satisface el requerimiento [Visualizar información de anuncios publicados](#).



Figura 6.29: UIPanel33: Información específica de un anuncio.

En la Figura 6.30 se muestra el formulario para registrar un nuevo anuncio que satisface el requerimiento Visualizar información de anuncios publicados.

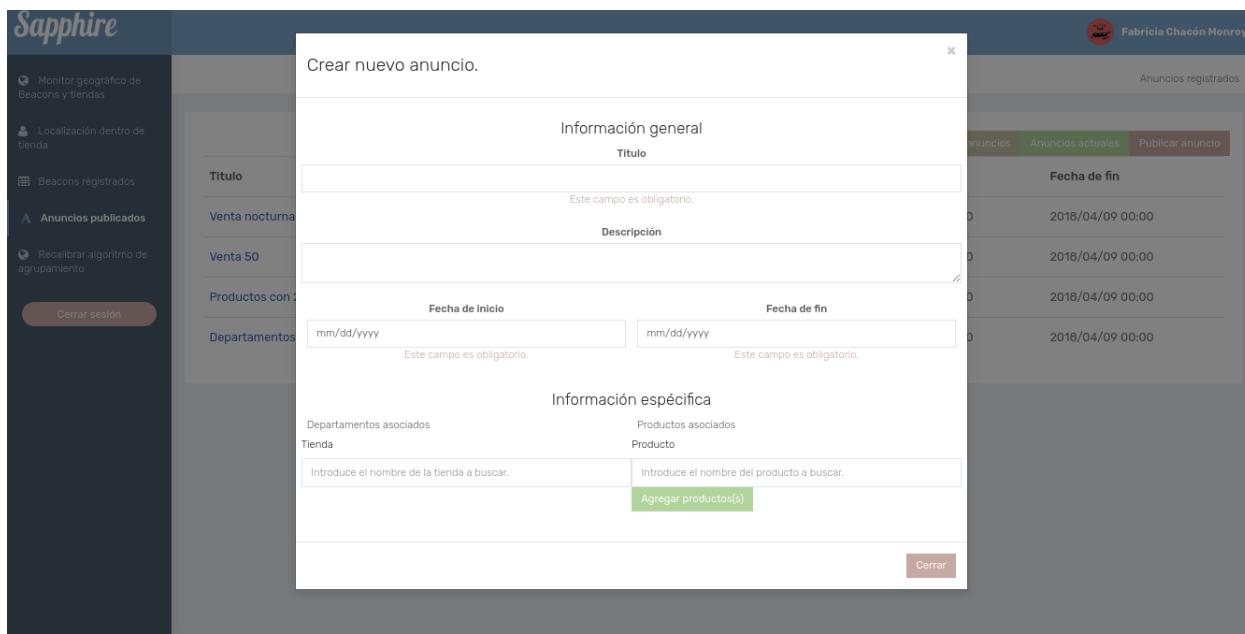


Figura 6.30: UIPanel34: Crear nuevo anuncio.

En la Figura 6.31 se muestra la interfaz de usuario que satisface el requerimiento Visualizar información de Beacons.

The screenshot shows a sidebar with navigation options: Monitor geográfico de Beacons y tiendas, Localización dentro de tienda, Beacons registrados (selected), Anuncios publicados, Recalibrar algoritmo de agrupamiento, and Cerrar sesión. The main area displays a table of registered Beacons:

ID	Nombre	Porcentaje de batería	Tienda	Departamento	Planta	Dirección	Última sincronización
668afdd788d1a7d5ad6362b203fbaf16	blueberry	88% de Batería	Rincón y Bermúdez S.C.	Películas y series	2	Real de Catorce, Catorce, San Luis Potosí, 78550	2018/06/30 02:14
1ee3c45ce62b556c89009dd0ef32841c	beetroot	94% de Batería	Rincón y Bermúdez S.C.	Películas y series	2	Real de Catorce, Catorce, San Luis Potosí, 78550	2018/06/30 02:14
6e4daeeba11bc7121196842625955b17	candy	89% de Batería	Sin tienda asignada.	Sin departamento asignado.	Sin planta asignada.	Sin dirección asignada.	Aún no se sincroniza este Beacon.
fc01e9234e736863d70ca39ac635621b	ice	91% de Batería	Sin tienda	Sin departamento	Sin planta	Sin dirección asignada.	Aún no se sincroniza este Beacon.

Figura 6.31: UIPanel35: Visualizar Beacons.

En la Figura 6.32 se muestra la pantalla que permite visualizar la información detallada de un Beacon y la posibilidad de modificar su información que satisface el requerimiento **Visualizar información de Beacons**.

The screenshot shows the same sidebar as Figure 6.31. A modal dialog box titled "Editar Información del Beacon." is open, showing the current information for the beacon with ID 668afdd788d1a7d5ad6362b203fbaf16:

Información actual	
Tienda Actual	Rincón y Bermúdez S.C.
Departamento Actual	Películas y series
Planta actual	2

The dialog also contains a "Tienda" input field with placeholder text "Selecciona una tienda para posteriormente asignarle un departamento." and a "Cerrar" button.

Figura 6.32: UIPanel36: Información de Beacon específica.

### Flujo de navegación del Panel de Administración.

La figura 6.42 muestra como es el flujo de navegación general del Panel de Admnistración.

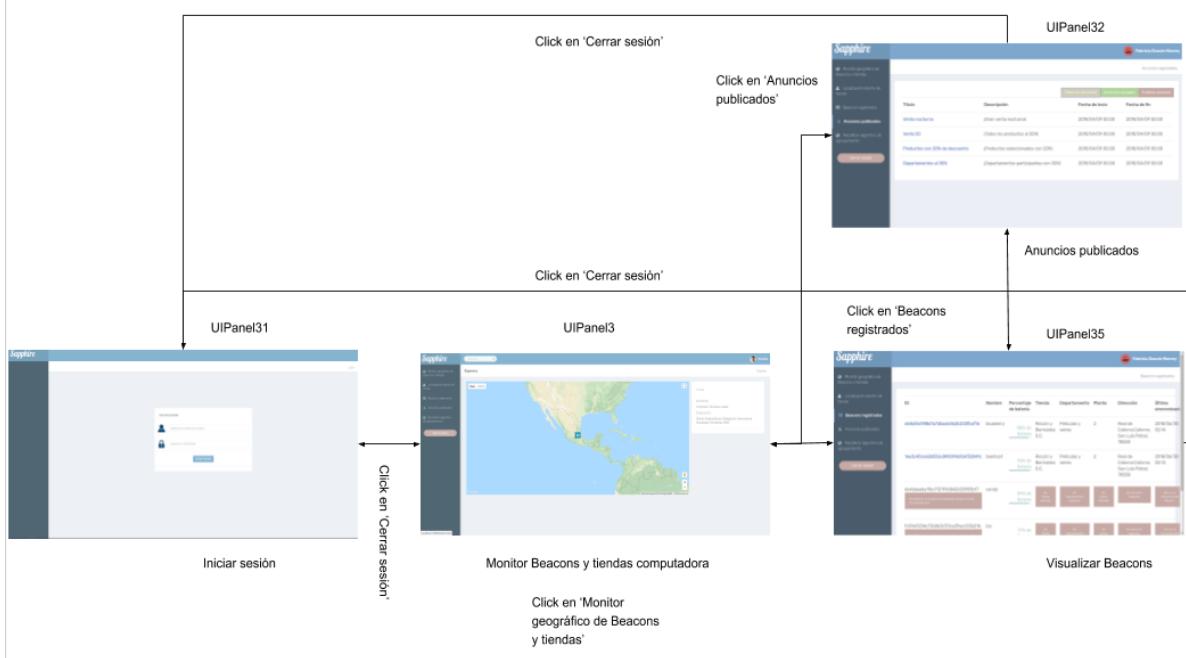


Figura 6.33: Flujo de navegación general Panel de Administración.

La figura 6.43 muestra como es el flujo de navegación de la pantalla Beacons registrados.

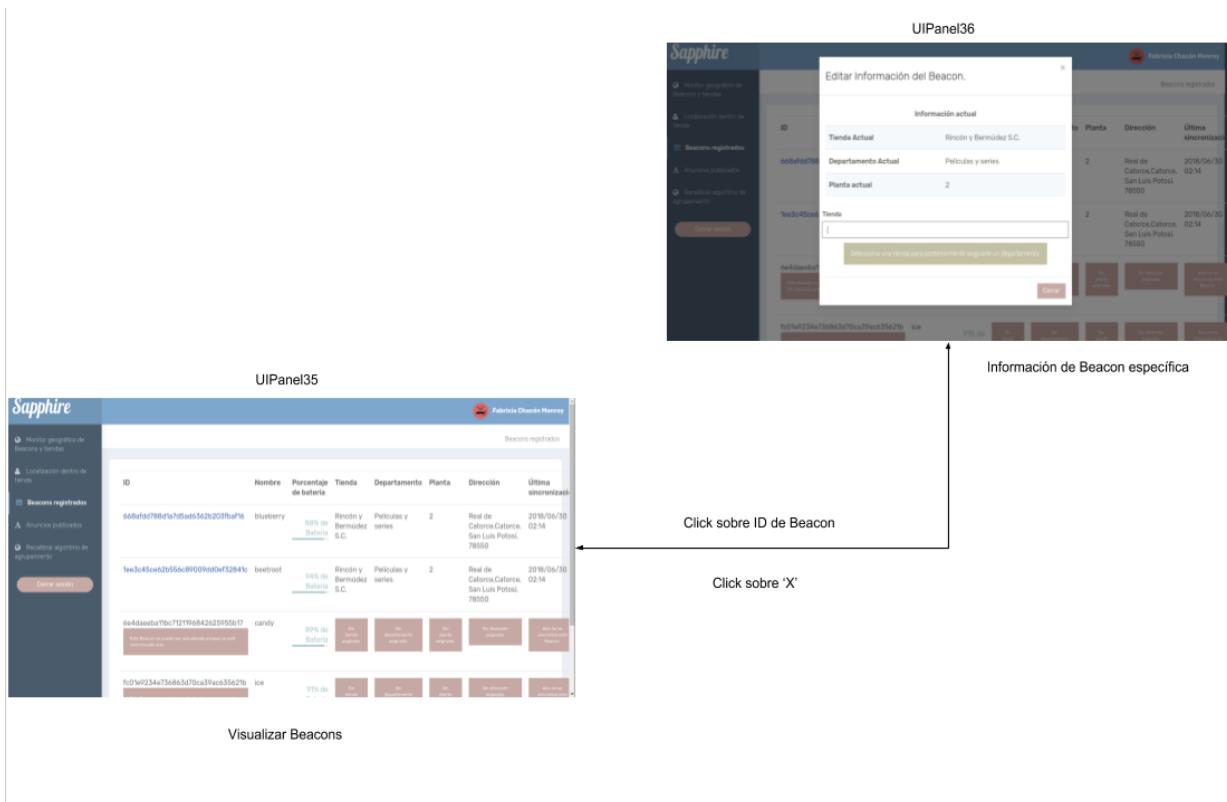


Figura 6.34: Flujo de navegación de pantalla Beacons registrados.

La figura 6.43 muestra como es el flujo de navegación de la pantalla Anuncios publicados.

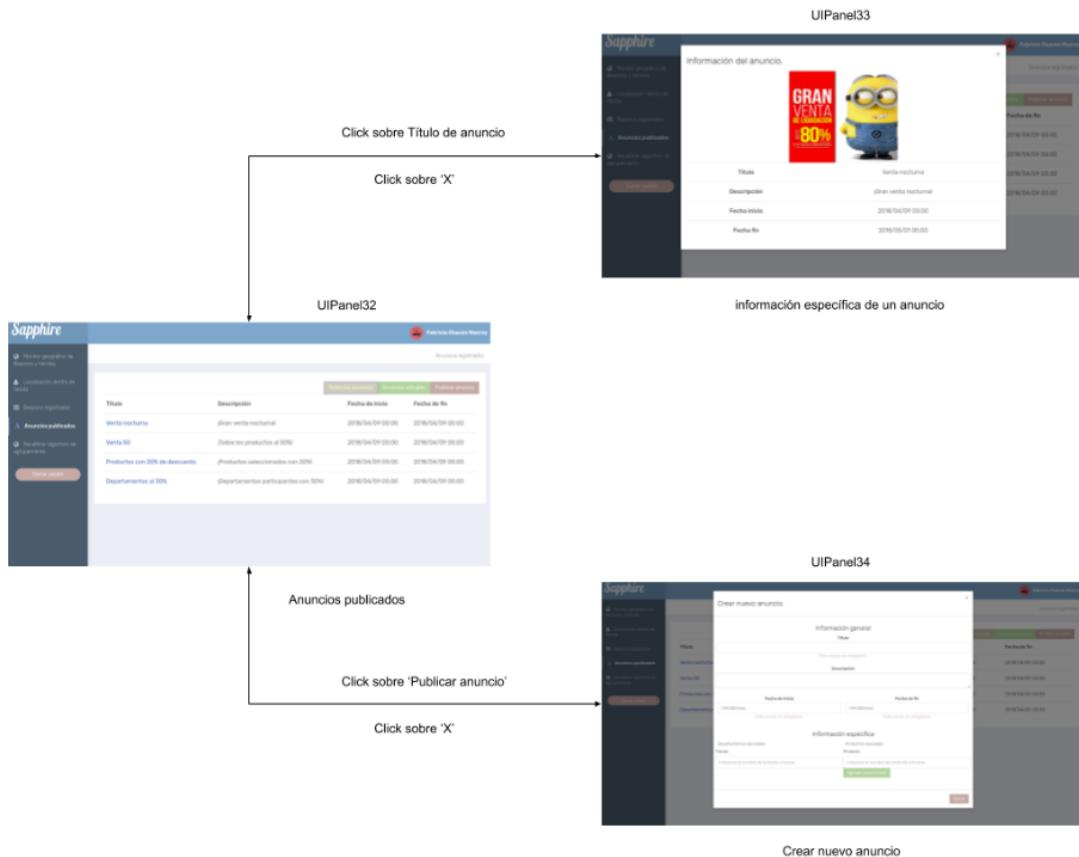


Figura 6.35: Flujo de navegación de pantalla Anuncios publicados.

### 6.1.5. Prototipo 4: Integración de pantalla para ejecutar algoritmo k-means

Dentro de este prototipo se satisface el requerimiento funcional [Ejecutar algoritmo de agrupamiento](#), definido previamente en el capítulo del “Bosquejo general de la aplicación” con el título de “Requerimientos funcionales del Panel de Administración”.

#### 6.1.5.1. Análisis

##### Diagrama de casos de uso

En la figura 6.15 se puede visualizar el diagrama de casos de uso actualizado.  
 Modificaciones realizadas:

- Ninguna.

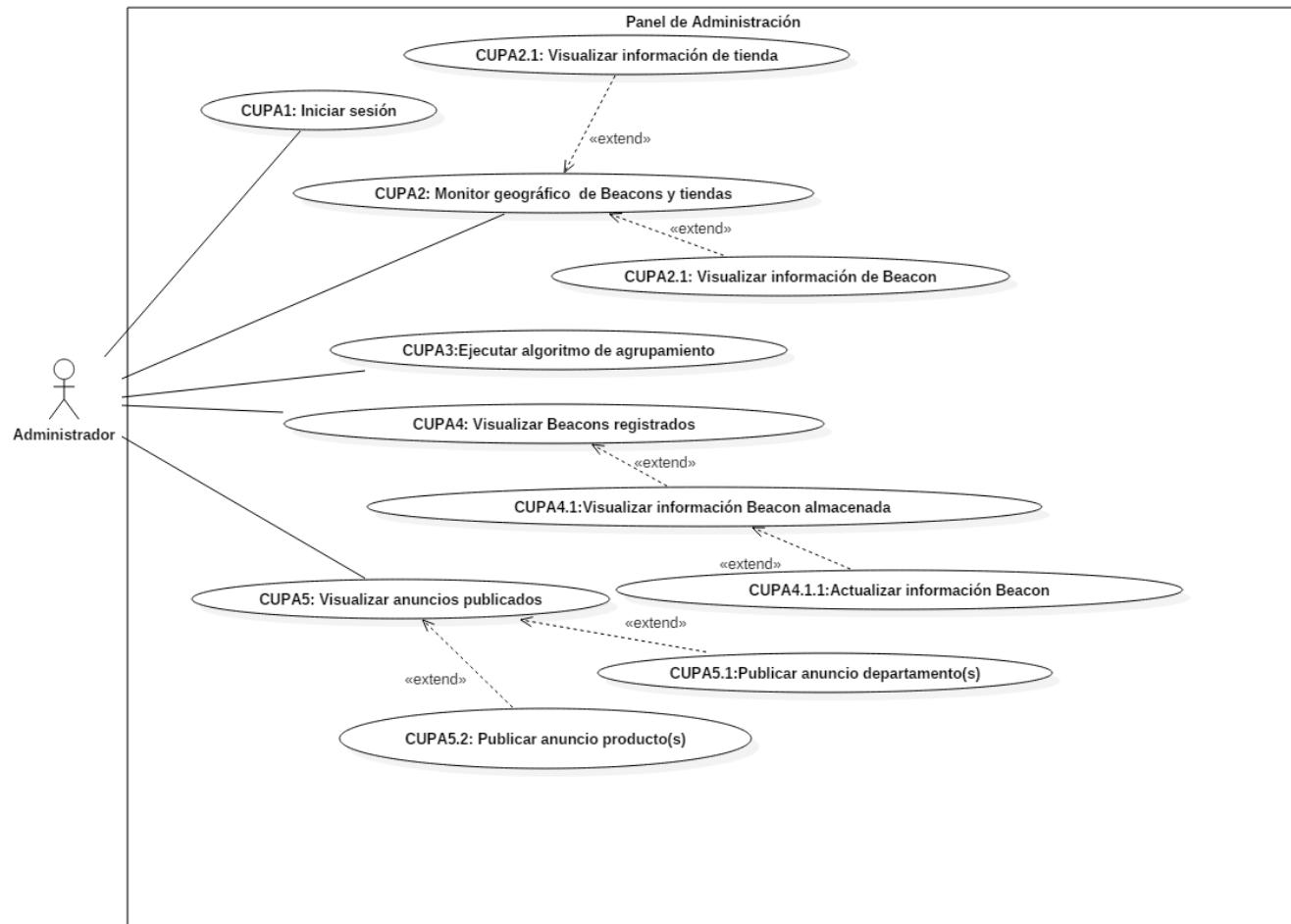


Figura 6.36: Casos de uso PA.

---

#### 6.1.5.2. Diseño

##### Diagramas de secuencia

Las figuras [6.37](#) muestra el diagrama que satisface el caso de uso **Ejecutar algoritmo de agrupamiento** que se muestra en el diagrama de casos de uso.

*El diagrama fue separado en tres figuras para mejor visualización ([6.37](#), [6.38](#), [6.39](#))*

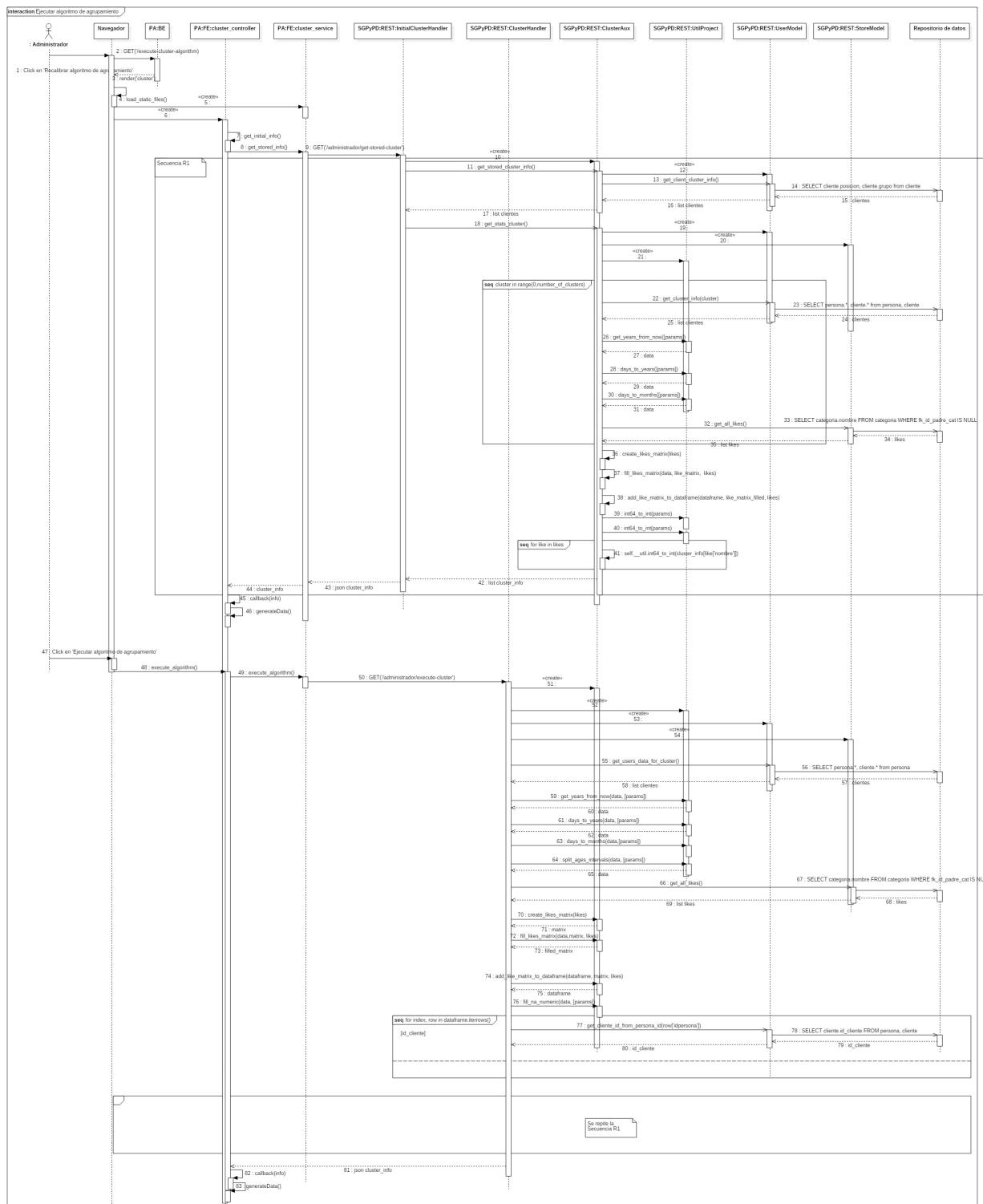
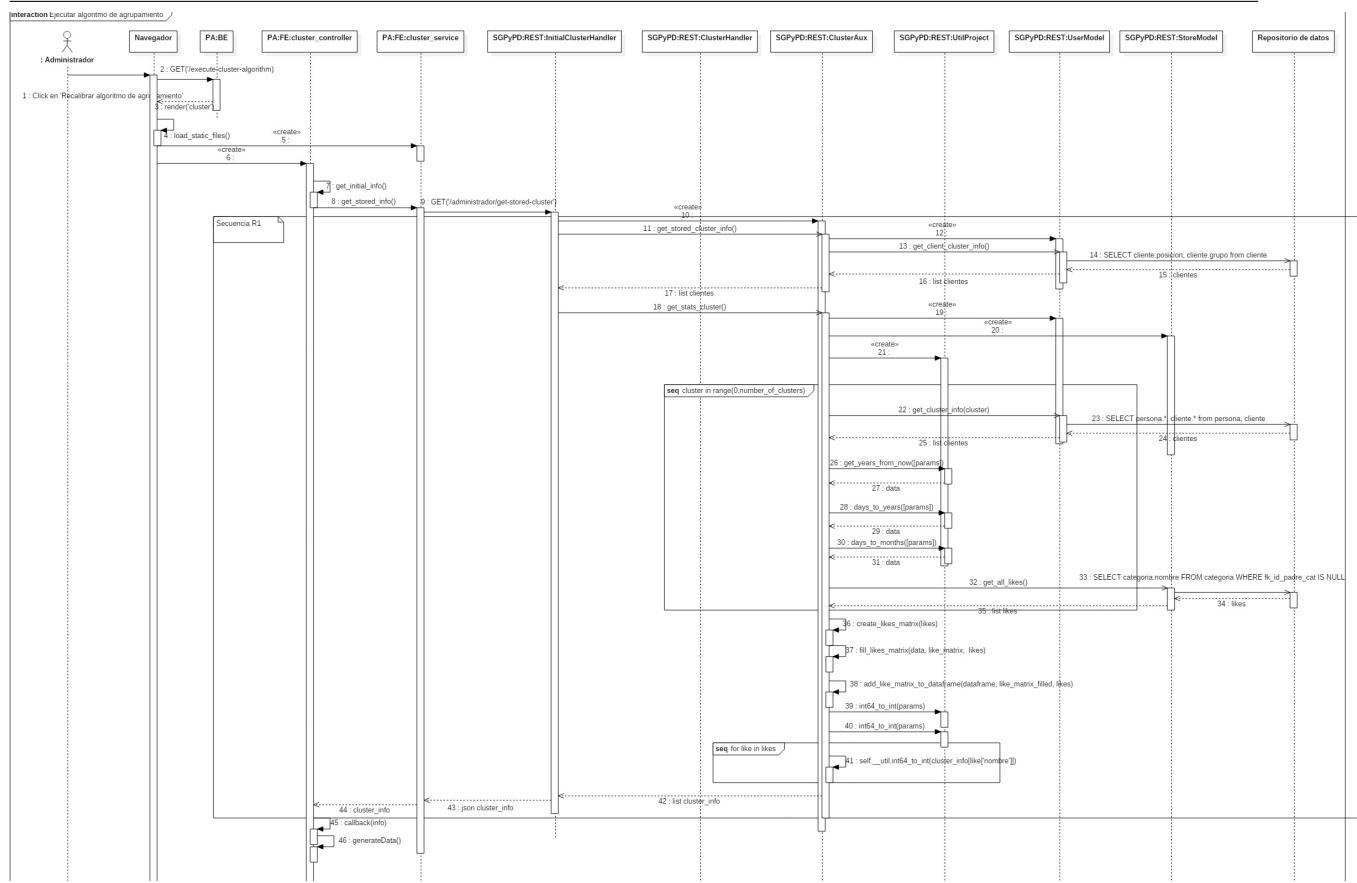


Figura 6.37: Diagrama de secuencia Ejecutar algoritmo de agrupamiento (Visualización completa).



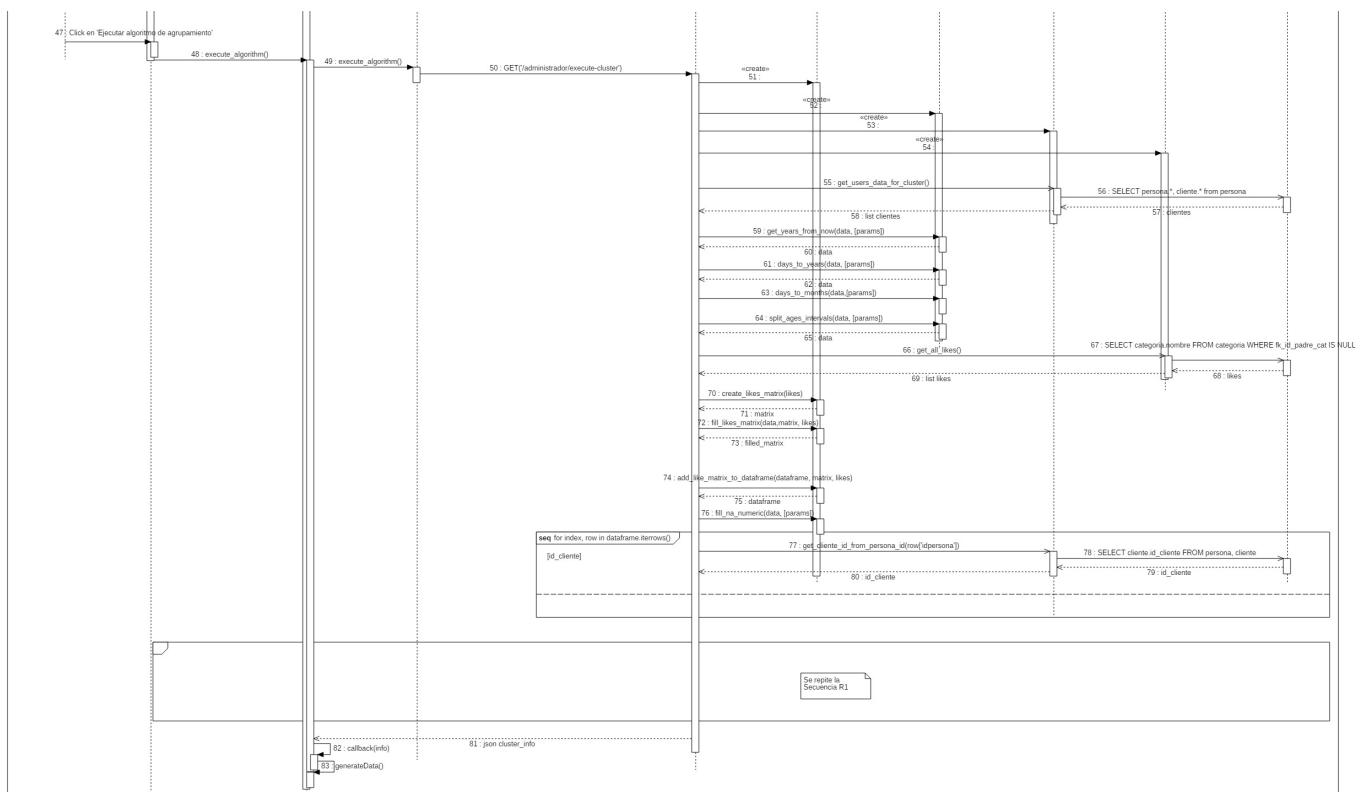


Figura 6.39: Diagrama de secuencia Ejecutar algoritmo de agrupamiento (Parte 2).

## Diseño de interfaz de usuario

En las Figuras 6.40 y 6.41 se muestra la interfaz de usuario que satisface el requerimiento Ejecutar algoritmo de agrupamiento.



Figura 6.40: UIPanel37: Ejecutar algoritmo de agrupamiento (Parte 1).

Cluster 0	Cluster 1	Cluster 2			
Edad promedio	36	Edad promedio	20	Edad promedio	9
Meses promedio en el sistema	5.0	Meses promedio en el sistema	5.0	Meses promedio en el sistema	1.0
Solteros	71	Solteros	26	Solteros	3
Casados	50	Casados	31	Casados	1
Hombres	56	Hombres	20	Hombres	
Mujeres	65	Mujeres	37	Mujeres	
Top gustos	Top gustos	Top gustos			
Drones y radio control:27	Instrumentos musicales:13	Videojuegos:3			
TV y video:22	Películas y series:12	Películas y series:2			
Instrumentos musicales:21	Drones y radio control:11	Películas y series:2			

Figura 6.41: UIPanel38: Ejecutar algoritmo de agrupamiento (Parte 2).

### Flujo de navegación del Panel de Administración.

La figura 6.42 muestra como es el flujo de navegación general del Panel de Admnistración.

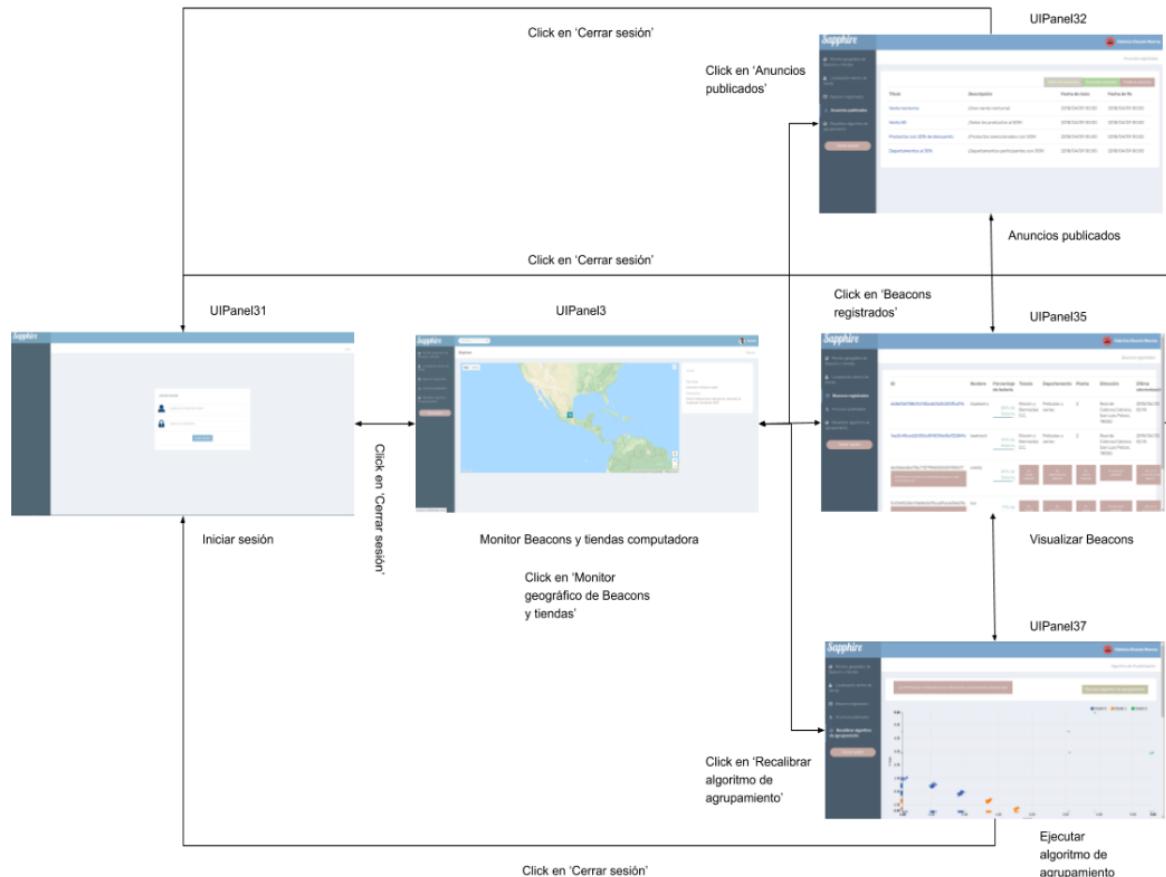


Figura 6.42: Flujo de navegación general Panel de Administración.

La figura 6.43 muestra como es el flujo de navegación de la pantalla Beacons registrados.

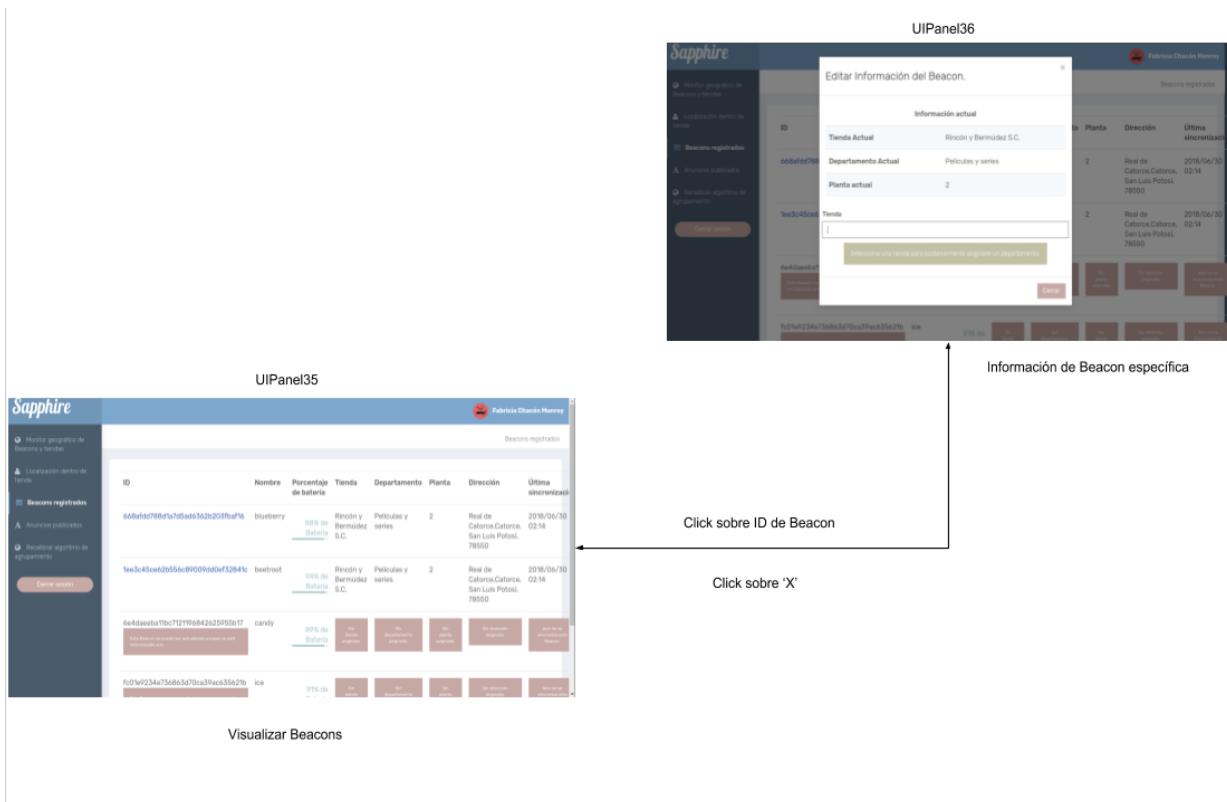


Figura 6.43: Flujo de navegación de pantalla Beacons registrados.

La figura 6.43 muestra como es el flujo de navegación de la pantalla Anuncios publicados.

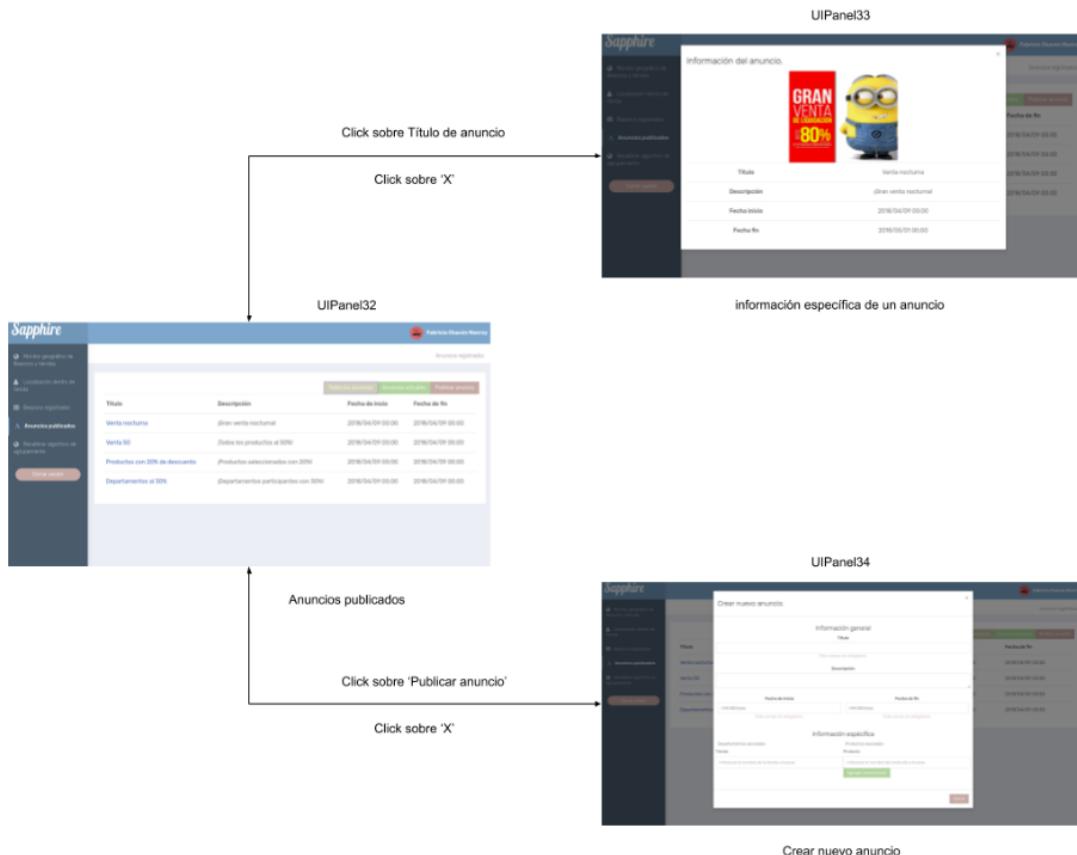


Figura 6.44: Flujo de navegación de pantalla Anuncios publicados.

### 6.1.5.3. Pruebas

Las pruebas del sistema fueron hechas sobre una laptop Lenovo Thinkpad T460 con las siguientes características:

- Procesador Intel® Core™ i5-5300U (3M Cache, hasta 2,90 GHz).
- 8 GB RAM.
- 240 GB Solid State Drive ROM.
- Sistema Operativo Fedora 27.

#### Prueba 1.

- Número de clientes: 184
- Número de clusters: 2

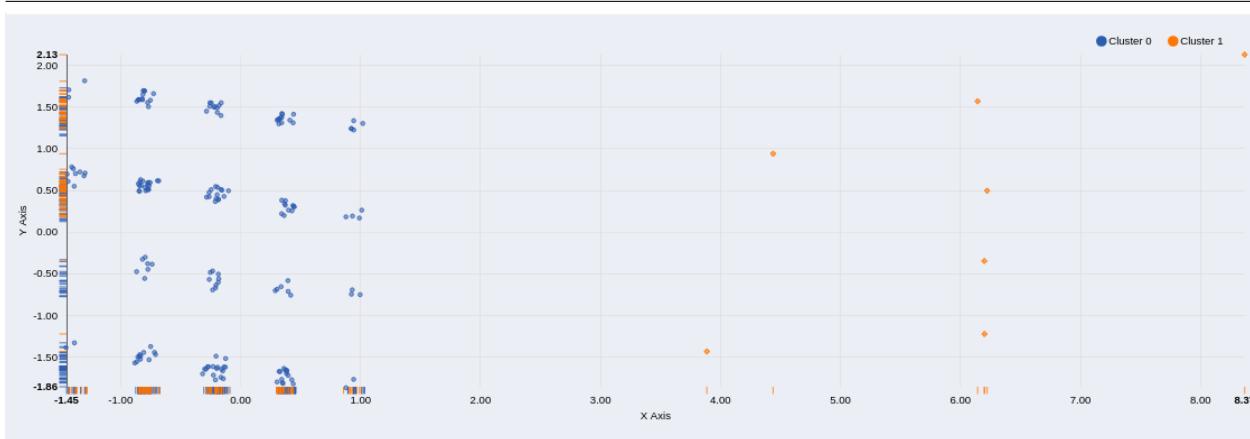


Figura 6.45: Prueba 1.

Cluster 0	Cluster 1
Edad promedio	31
Meses promedio en el sistema	7.0
Solteros	96
Casados	82
Hombres	75
Mujeres	102
<b>Top gustos</b>	<b>Top gustos</b>
Drones y radio control:38	Videojuegos:3
Instrumentos musicales:34	Películas y series:2
TV y video:32	Películas y series:2

Figura 6.46: Prueba 1 características de cada cluster.

### Prueba 2.

- Número de clientes:184
- Número de clusters: 3

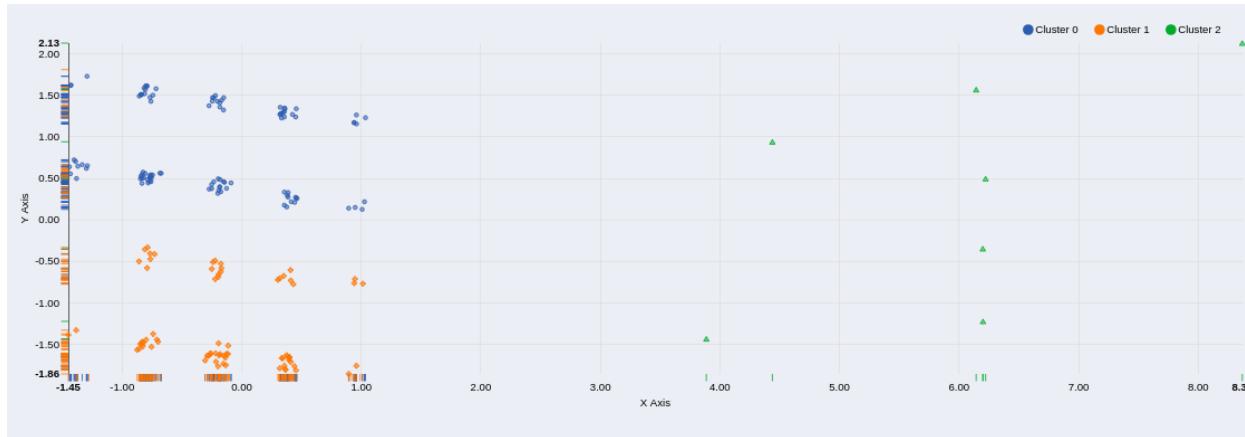


Figura 6.47: Prueba 2.

Cluster 0	Cluster 1	Cluster 2			
Edad promedio	32	Edad promedio	30	Edad promedio	13
Meses promedio en el sistema	7.0	Meses promedio en el sistema	7.0	Meses promedio en el sistema	2.5714285714285716
Solteros	55	Solteros	41	Solteros	3
Casados	50	Casados	32	Casados	2
Hombres	43	Hombres	32	Hombres	
Mujeres	61	Mujeres	41	Mujeres	
<b>Top gustos</b>	<b>Top gustos</b>	<b>Top gustos</b>			
TV y video:20	Drones y radio control:27	Videojuegos:3			
Celulares y tablets:19	Instrumentos musicales:19	Películas y series:2			
Instrumentos musicales:15	Películas y series:15	Películas y series:2			

Figura 6.48: Prueba 2 características de cada cluster.

### Prueba 3.

- Número de clientes:184
- Número de clusters: 4

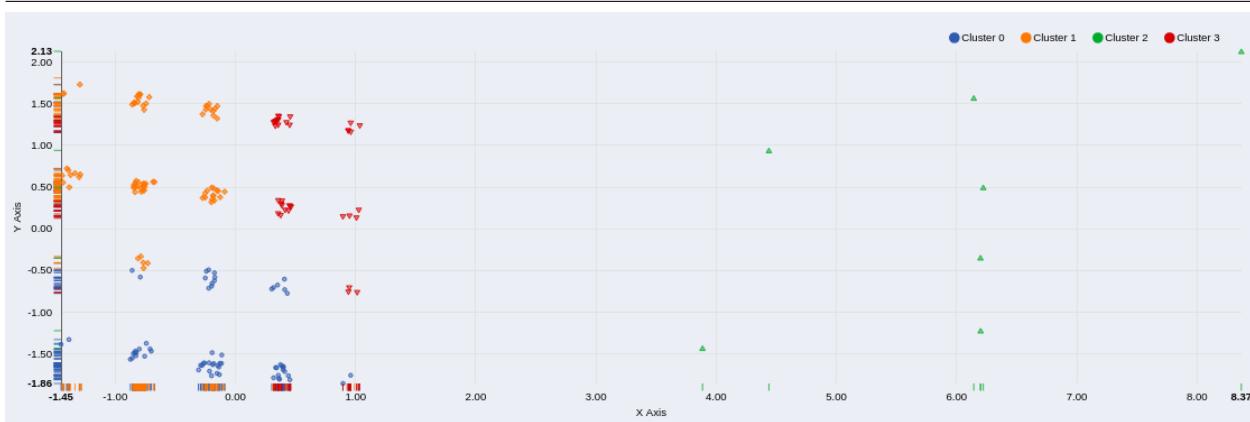


Figura 6.49: Prueba 3.

Cluster 0		Cluster 1		Cluster 2	
Edad promedio	29	Edad promedio	38	Edad promedio	13
Meses promedio en el sistema	7.0	Meses promedio en el sistema	7.0	Meses promedio en el sistema	2.5714285714285716
Solteros	39	Solteros	42	Solteros	3
Casados	26	Casados	35	Casados	2
Hombres	30	Hombres	33	Hombres	
Mujeres	35	Mujeres	44	Mujeres	
<b>Top gustos</b>		<b>Top gustos</b>		<b>Top gustos</b>	
Drones y radio control:27		Celulares y tablets:18		Videojuegos:3	
Instrumentos musicales:18		TV y video:15		Películas y series:2	
Cámaras:15		Instrumentos musicales:12		Películas y series:2	

Figura 6.50: Prueba 3 características de cada cluster (Parte 1).

Cluster 3	
Edad promedio	20
Meses promedio en el sistema	7.0
Solteros	15
Casados	21
Hombres	12
Mujeres	23
Top gustos	
TV y video:	7
Autos y motos:	6
Autos y motos:	6

Figura 6.51: Prueba 3 características de cada cluster (Parte 2).

#### 6.1.5.4. Conclusiones

De acuerdo con las pruebas realizadas se decidió que el número de clusters óptimo es tres, ya que, separa los datos de una manera muy natural (como se muestra en la gráfica 6.47) y a pesar de que existe una muy poca diferencia en la edad promedio entre el cluster 0 y el cluster 1 los gustos de cada cluster son diferentes y eso es un factor importante a tomar en cuenta.

## 6.2. Tornado

Tornado es un framework Web de Python y una biblioteca de red asíncrona, desarrollada originalmente en FriendFeed. Mediante el uso de E / S de red sin bloqueo, Tornado puede escalar a decenas de miles de conexiones abiertas, lo que lo hace ideal para largos sondeos, WebSockets y otras aplicaciones que requieren una conexión de larga duración para cada usuario. En la figura 6.52 se puede apreciar el logo del framework [104].



Figura 6.52: Logo de Tornado. [104]

### 6.2.1. Sistema de gestión, procesamiento y proveedor de datos de Retail

#### 6.2.2. Prototipo 1: Desarrollo de servicios REST

##### 6.2.2.1. Análisis

Dentro del análisis para el desarrollo de este prototipo se incorporan los siguientes requerimientos funcionales:

- RFGPPR6 Registrar folletos.
- RFGPPR7 Obtener folletos: Administrador.
- RFGPPR8 Proveer Beacons registrados.
- RFGPPR9 Proveer establecimientos registrados.
- RFGPPR10 Proveer departamentos dentro de tienda.
- RFGPPR11 Insertar/Actualizar Beacons del sistema.
- RFGPPR12 Obtener atributos clave-valor de Beacons.
- RFGPPR13 Proveer token de autenticación: Administrador.
- RFGPPR14 Proveer token de autenticación: Vendedor.
- RFGPPR15 Actualizar token de Firebase Cloud Messaging (FCM) para vendedor.
- RFGPPR16 Actualizar datos básicos del perfil del vendedor.
- RFGPPR17 Obtener clientes en departamento.
- RFGPPR18 Obtener detalles y estadísticas de cliente.
- RFGPPR19 Obtener recomendaciones por departamento.
- RFGPPR20 Enviar recomendaciones a cliente.
- RFGPPR21 Insertar/Actualizar Beacons de desde la AIPV.
- RFGPPR22 Registrar clientes.
- RFGPPR23 Obtener información del cliente.
- RFGPPR24 Proveer de productos.
- RFGPPR25 Proveer de folletos.
- RFGPPR26 Agregar producto a favoritos.
- RFGPPR27 Actualizar token de Firebase Cloud Messaging para cliente.
- RFGPPR28 Publicar ubicación de cliente en Kafka.
- RFGPPR29 Obtener recomendaciones globales.

De igual manera, se incorporan los requerimientos funcionales de la sección RFAIDP Requerimientos Funcionales de Aplicación Interactiva Difusora de Productos.

- RFAIDP1 Iniciar sesión.
- RFAIDP2 Consultar folletos.
- RFAIDP4 Consultar logros.
- RFAIDP5 Añadir producto a favoritos.
- RFAIDP6 Visualizar favoritos.

- RFAIDP7 Consultar recomendaciones.
- RFAIDP8 Buscar productos.
- RFAIDP9 Actualizar datos personales.
- RFAIDP11 Actualizar gustos genéricos.
- RFAIDP12 Consultar beneficios.
- RFAIDP13 Actualizar permisos.
- RFAIDP16 Registrar cuenta nueva.
- RFAIDP17 Recuperar contraseña.
- RFAIDP18 Eliminar producto de favoritos.
- RFAIDP19 Mostrar información de productos.

Definidos previamente en el capítulo del “Bosquejo general de la aplicación” con el título de “Requerimientos funcionales del Sistema de Gestión, Procesamiento y Proveedor de Datos Retail”.

#### 6.2.2.2. Diseño

Dentro de esta subsección se encuentra la documentación de los servicios REST separados por requerimientos.

*Nota: En algunos servicios se hace uso de una notación específica ([texto, tipo de dato]), esta notación significa que el contenido del lado izquierdo de la coma es dinámico, por otro lado, el contenido derecho es el tipo de dato que debe ser el contenido dinámico.*

##### RFGPPR6 Registrar folletos

- Ruta: /administrador/flyers/[filter]

- Método: POST
- Tipo de parámetro: ruta
- Opciones: 'register'
- Cuerpo de la petición:

```

1 {
2     "data": {
3         "name": "string",
4         "description": "string",
5         "init_date": "string",
6         "end_date": "string",
7         "products": [integer,...],
8         "departments": [integer,...]
9     }
10 }
```

- Respuesta:

```

1  {
2      "message": "string"
3 }
```

### RFGPPR7 Obtener folletos: Administrador

- Ruta: /administrador/flyers/[filter]

- Método: GET
- Tipo de parámetro: ruta
- Opciones: 'all'
- Respuesta:

```

1  {
2      "flyers": [
3          {
4              "departments": [
5                  {
6                      "planta": "string",
7                      "nombre": "string",
8                      "tienda": "string",
9                      "fk_id_tienda": integer,
10                     "id_departamento": integer
11                 }, ...
12             ],
13             "fecha_inicio": "string",
14             "fecha_fin": "string",
15             "descripcion": "string",
16             "products": [
17                 {
18                     "precio_venta": "string",
19                     "producto": "string",
20                     "tienda": "string",
21                     "promocion": "string",
22                     "marca": "string",
23                     "stock": integer}, ...],
24                     "titulo": "string",
25                     "id_anuncio": integer,
26                     "ruta_imagen": "string"
27                 },
28             ...
29 }
```

### RFGPPR8 Proveer Beacons registrados

- Ruta: /beacon-monitor/[beacon]

- Método: GET
- Tipo de parámetro: ruta
- Opciones: 'all', [id de Beacon, string]
- Respuesta para opción 'all':

```

1 {"beacons": [
2     "id_beacon" : integer,
3     "fk_id_departamento": integer,
4     "posicion_geografica" : [float, float],
5 },
6 { ... },
7 { ... },
8 ... ]

```

- Respuesta para opción [id de Beacon, string]:

```

1 {
2     "identifier": "string",
3     "shadow": {
4         "id": integer,
5         "name": "string",
6         "tags": [],
7         "organization_id": integer,
8         "location": {
9             "timezone": "America/Los_Angeles"
10        },
11        "public": boolean,
12        "development_mode": boolean
13    },
14    "alerts_configuration": {
15        "beacon_stolen_notifications": "string",
16        "beacon_fallen_notifications": "string",
17        "beacon_battery_notifications": "string",
18        "updated_at": string
19    },
20    "alerts": [],
21    "hardware_type": "string",
22    "hardware_revision": "string",
23    "hardware_footprint": "string",
24    "color": "string",
25    "form_factor": "string",
26    "status_report": {
27        "battery_percentage": integer,
28        "battery_voltage": integer,
29        "estimated_battery_lifetime": integer,
30        "clock_offset": integer,
31        "firmware_version": "string",
32        "last_synced": string,
33        "public": boolean,
34        "broken_motion": boolean,
35        "broken_temperature": boolean,
36        "missing_telemetry_data": boolean
37    },
38    "settings": {
39        "general": {
40            "motion_detection_enabled": boolean,
41            "motion_only_enabled": boolean,
42            "dark_to_sleep_enabled": boolean,
43        }
44    }
45 }

```

```

42     "dark_to_sleep_threshold": integer,
43     "conditional_broadcasting": "string",
44     "flip_to_sleep_enabled": boolean,
45     "smart_power_mode_enabled": boolean,
46     "temperature_offset": integer,
47     "eddystone_configuration_service_enabled": boolean,
48     "nfc_records": [
49         {
50             "data": "string",
51             "type": "string"
52         }
53     ],
54     "automatic_firmware_update_enabled": boolean,
55     "motion_only_delay": integer,
56     "display_access_control": boolean,
57     "wifi": {
58         "enabled": boolean,
59         "ssid": "string",
60         "password": "string"
61     },
62     "templates": {
63         "default_template": "string",
64         "lock_template": boolean
65     },
66     "schedule": {
67         "enabled": boolean,
68         "start_time": integer,
69         "stop_time": integer
70     },
71     "whitelisted_domains": [integer, integer, ...],
72     "magnetometer_settings": [integer, integer, ...],
73     "estimote_secure_monitoring": {
74         "enabled": boolean,
75         "krps": integer,
76         "prps": integer,
77         "seed": "integer"
78     }
79 },
80     "advertisers": {
81         "connectivity": [
82             {
83                 "index": integer,
84                 "name": "string",
85                 "power": integer,
86                 "interval": integer,
87                 "shake_to_connect_enabled": boolean,
88                 "near_to_connect_enabled": boolean
89             }
90         ],
91         "eddystone_telemetry": [
92             {
93                 "index": integer,

```

```

94                     "name": "string",
95                     "enabled": boolean,
96                     "interval": integer,
97                     "power": integer
98                 }
99             ],
100            "eddystone_uid": [
101                {
102                    "index": integer,
103                    "name": "string",
104                    "enabled": boolean,
105                    "interval": integer,
106                    "power": integer,
107                    "instance_id": "string",
108                    "namespace_id": "string"
109                }
110            ],
111            "eddystone_url": [
112                {
113                    "index": integer,
114                    "name": "string",
115                    "enabled": boolean,
116                    "interval": integer,
117                    "power": integer,
118                    "url": "string"
119                }
120            ],
121            "estimote_location": [
122                {
123                    "index": int,
124                    "name": "string",
125                    "enabled": boolean,
126                    "interval": integer,
127                    "power": integer
128                }
129            ],
130            "estimote_telemetry": [
131                {
132                    "index": integer,
133                    "name": "string",
134                    "enabled": boolean,
135                    "interval": integer,
136                    "power": integer
137                }
138            ],
139            "generic": [
140                {
141                    "index": integer,
142                    "name": "string",
143                    "enabled": boolean,
144                    "power": integer,
145                    "interval": integer,

```

```

146             "payload": "string"
147         },
148     {
149         "index": integer,
150         "name": "string",
151         "enabled": boolean,
152         "power": integer,
153         "interval": integer,
154         "payload": "string"
155     },
156     {
157         "index": integer,
158         "name": "string",
159         "enabled": boolean,
160         "power": integer,
161         "interval": integer,
162         "payload": "string"
163     }
164 ],
165 "ibeacon": [
166     {
167         "index": integer,
168         "name": "string",
169         "enabled": boolean,
170         "uuid": "string",
171         "major": integer,
172         "minor": integer,
173         "power": integer,
174         "interval": integer,
175         "security": {
176             "enabled": boolean,
177             "real_id": integer,
178             "interval_scaler": integer
179         },
180         "non_strict_mode_enabled": boolean
181     }
182 ],
183 "link_network": [
184     {
185         "index": integer,
186         "name": "string",
187         "enabled": boolean,
188         "interval": integer,
189         "power": integer,
190         "link_id": integer,
191         "domain": "string",
192         "measurement_period": integer
193     }
194 ],
195 },
196 },
197 "pending_settings": {

```

```

198     "advertisers": {
199         "generic": [
200             {
201                 "index": integer,
202                 "name": "string",
203                 "enabled": boolean,
204                 "power": integer,
205                 "interval": integer,
206                 "payload": "string"
207             }
208         ]
209     },
210     "mesh": {
211         "name": "string",
212         "type": "string"
213     }
214     "store": "string",
215     "department" : "string",
216 }
217 }
```

### RFGPPR9 Proveer establecimientos registrados

- Ruta: /administrador/store-monitor/store
  - Método: GET
  - Tipo de parámetro: ruta
  - Opciones: 'all', [id de tienda, int]
  - Respuesta para opción 'all':

```

1 {"stores": [
2     "id_tienda" : integer,
3     "nombre" : "string",
4     "posicion_geografica" : [float, float],
5 ],
6 { ... },
7 { ... },
8 ... ]}
```

- Respuesta para opción [id de tienda, int]

```

1 {
2     "id_tienda" : integer,
3     "nombre" : "string",
4     "direccion" : "string",
5     "beacon_registrados" : integer
6     "posicion_geografica" : [float, float],
7 }
```

### RFGPPR10 Proveer departamentos dentro de tienda.

- Ruta: /administrador/departments/[id\_tienda]

- Método: GET
- Tipo de parámetro: ruta
- Opciones: [id\_tienda, int]
- Respuesta exitosa:

```

1  {
2      "departments": [
3          [
4              {
5                  "id_departamento": integer,
6                  "fk_id_tienda": integer,
7                  "nombre": string
8                  "planta": string
9              }
10             ...
11         ]
12     }
13 }
```

- Respuesta erronea:

```

1  {
2      "message": string
3 }
```

#### RFGPPR11 Insertar/Actualizar Beacons del sistema.

- Ruta: /administrador/departments/[option]
  - Método: POST
  - Tipo de parámetro: ruta
  - Opciones: [option, string]
  - Respuesta exitosa:

```

1  {
2      "status": string
3 }
```

- Respuesta erronea:

```

1  {
2      "message": string
3 }
```

#### RFGPPR12 Obtener atributos clave-valor de Beacons.

- Ruta: /administrador/departments/[option]
  - Método: POST
  - Tipo de parámetro: ruta
  - Opciones: [option, string]
  - Respuesta exitosa:

```

1  {
2      "floor": [
3          string, ...
4      ],
5      "fk_id_store": [
6          string, ...
7      ],
8      "fk_id_department": [
9          string, ...
10     ]
11 }

```

#### RFGPPR13 Proveer token de autenticación: Administrador

- Ruta: /administrador/login

- Método: POST
- Cuerpo de la petición

```

1  {
2      "user": "string",
3      "password": "string"
4 }

```

- Respuesta exitosa:

```

1  {
2      "email": "string",
3      "token": "string",
4      "shop_name": "string",
5      "image": "string",
6      "id_administrador": integer,
7      "department_name": "integer",
8      "departament_id": integer,
9      "mat_surname": "string",
10     "pat_surname": "string",
11     "name": "string",
12     "shop_id": integer
13 }

```

- Respuesta erronea:

```

1  {
2      "message": "string",
3 }

```

#### RFGPPR14 Proveer token de autenticación: Vendedor

- Ruta: /vendedor/login
- Método: POST
- Cuerpo de la petición

```

1  {
2      "user": "string",
3      "password": "string"
4 }
```

- Respuesta exitosa:

```

1  {
2      "email": "string",
3      "token": "string",
4      "shop_name": "string",
5      "image": "string",
6      "id_administrador": integer,
7      "department_name": "integer",
8      "departament_id": integer,
9      "mat_surname": "string",
10     "pat_surname": "string",
11     "name": "string",
12     "shop_id": integer
13 }
```

- Respuesta erronea:

```

1  {
2     "message": "string",
3 }
```

#### RFGPPR15 Actualizar token de Firebase Cloud Messaging (FCM) para vendedor

- Ruta: /vendedor/update-token-fcm

- Método: POST
- Cabeceras de la petición:

```

1  {
2     "Content-Type": "application/json",
3     "authorization" : string
4 }
```

- Cuerpo de la petición:

```

1  {
2     "id_persona": integer,
3     "token" : string
4 }
```

- Respuesta exitosa:

```

1  {
2     "message": string,
3 }
```

#### RFGPPR16 Actualizar datos básicos del perfil del vendedor.

- Ruta: /vendedor

- Método: PUT
- Cabeceras de la petición:

```

1  {
2      "Content-Type": "application/json",
3      "authorization" : string
4 }
```

- Cuerpo de la petición:

```

1  {
2      "id_persona": integer,
3      "name" : string,
4      "pat_surname" : string,
5      "mat_surname" : string,
6      "email" : string,
7      "password" : string
8 }
```

- Respuesta exitosa:

```

1  {
2      "message": string,
3 }
```

### RFGPPR17 Obtener clientes cercanos en departamento.

- Ruta: /vendedor/departamento/[department]
- Tipo de parámetro: ruta
- Opciones: [department, int]

- Método: GET
- Cabeceras de la petición:

```

1  {
2      "Content-Type": "application/json",
3      "authorization" : string
4 }
```

- Respuesta exitosa:

```

1  {
2      "status": string,
3      "clients": [
4          {
5              "id_person": integer,
6              "likes": [string,...],
7              "score": integer,
8              "since": string,
9              "level": integer,
10             "image": string,
11             "sex": char,
```

```

11         "civil_status": char,
12         "department": integer,
13         "floor": string,
14         "sons": integer,
15         "age": integer,
16         "position": [integer, integer],
17         "id_beacon": integer,
18         "time": string
19     },
20     ...
21 ]
22 }
```

### RFGPPR18 Obtener detalles y estadísticas de cliente.

- Ruta: /vendedor/client-details/[id\_persona]

- Método: GET
- Tipo de parámetro: ruta
- Opciones: [id\_persona, int]
- Respuesta exitosa:

```

1 {
2     "benefits": [
3         "description": string,
4         "percentage": integer,
5         "reward": integer,
6         "free_product": char,
7     },
8     ...
9 ],
10    "categories": [
11        "id_categorie": integer,
12        "name": string,
13        "total": integer,
14        "subcategories": [
15            "id_categorie": integer,
16            "name": string
17        },
18        ...
19    ],
20    ...
21 ],
22    "purchases_stats": [
23        "month": integer,
24        "year": integer,
25        "total": integer
26    },
27    ...
28 ],
29    "favorites": [
30        ...
31 ]
```

```

31         "id_producto": integer,
32         "producto": string,
33         "precio_venta": double,
34         "stock": integer,
35         "departamento": string,
36         "tienda": string,
37         "categoria": string,
38         "marca": string,
39         "promocion": string,
40         "fecha_agregacion":string ,
41     },
42     ...
43 ]
44 }
```

### RFGPPR19 Obtener recomendaciones por departamento.

- Ruta: /client/recommendations/department/[id\_usuario]/[id\_departamento]
- Tipo de parámetro: ruta
- Opciones: [id\_usuario,id\_departamento, int]
  - Método: GET
  - Cabeceras de la petición:

```

1 {
2     "Content-Type": "application/json",
3     "authorization" : string
4 }
```

- Respuesta exitosa:

```

1 {
2     "recommendations": [
3         {
4             "id_producto" : integer,
5             "producto": "string",
6             "precio_venta" : "string",
7             "stock" : integer,
8             "departamento" : "string",
9             "tienda" : "string",
10            "categoria" : "string",
11            "marca" : "string",
12            "promocion" : "string",
13            "score": float,
14            "image": "string"
15        },
16        ...
17    ]
18 }
```

### RFGPPR20 Enviar recomendaciones a cliente.

- Ruta: /vendedor/recommend/[id\_usuario]/[id\_product]

- Tipo de parámetro: ruta
- Opciones: [id\_usuario,id\_product, int]
  - Método: POST
  - Cabeceras de la petición:

```

1 {
2     "Content-Type": "application/json",
3     "authorization" : string
4 }
```

- Respuesta exitosa:

```

1 {
2     "message": "string"
3 }
```

#### RFGPPR21 Insertar/Actualizar Beacons de desde la AIPV.

- Ruta: /beacon-monitor/update]
  - Método: POST
  - Cabeceras de la petición:

```

1 {
2     "Content-Type": "application/json",
3     "authorization" : string
4 }
```

- Tipo de parámetro: ruta
- Opciones: [option, string]
- Cuerpo de la petición:

```

1 {
2     "id_beacon": string,
3     "x": double,
4     "y": double,
5     "fk_id_departamento": int,
6     "id_persona": int
7 }
```

- Respuesta exitosa:

```

1 {
2     "status": string,
3     "clients": [
4         {
5             "id_person": integer,
6             "likes": list,
7             "score" : integer,
8             "since" : string,
9             "level" : string,
10            "image" : string,
```

```

11         "sex" : string,
12         "civil_status" : string,
13         "department" : string,
14         "floor" : integer,
15         "sons" : integer,
16         "age" : integer,
17         "position" : string,
18         "id_beacon" : integer,
19         "time" : string
20     }
21     ...
22 ]
23 }
```

- Respuesta erronea:

```

1 {
2     "message": string
3 }
```

### RFGPPR22 Registrar clientes: Facebook

- Ruta: /client
  - Método: POST
  - Cuerpo de la petición:

```

1 {
2     "nombre" : "string",
3     "apellido_paterno" : "string",
4     "apellido_materno" : "string",
5     "email" : "string",
6     "password" : "string",
7     "estado_civil" : "string",
8     "fecha_nacimiento" : "string",
9     "no_hijos" : "string",
10    "facebook" : boolean
11 }
```

- Respuesta:

```

1 {
2     "message": "string"
3 }
```

### RFGPPR24 Proveer de productos

- Ruta: /products/[producto]
  - Método: GET
  - Tipo de parámetro: ruta
  - Opciones: 'all', [nombre del producto, string], [id del producto, int]
  - Respuesta para todas las opciones 'all' y [nombre del producto, string]:

```

1 {"products": [
2     "id_producto" : integer,
3     "producto": "string",
4     "precio_venta" : "string",
5     "stock" : integer,
6     "departamento" : "string",
7     "tienda" : "string",
8     "categoria" : "string",
9     "marca" : "string",
10    "promocion" : "string",
11    "imagen": "string"
12 },
13 {
14     ...
15 }
15

```

- Respuesta para opción [id del producto, int]:

```

1 {
2     "id_producto" : integer,
3     "producto": "string",
4     "precio_venta" : "string",
5     "stock" : integer,
6     "departamento" : "string",
7     "tienda" : "string",
8     "categoria" : "string",
9     "marca" : "string",
10    "promocion" : "string",
11    "imagen": "string"
12 }
12

```

## RFGPPR25 Proveer folletos

- Ruta: /dashboard-client/
  - Método: GET
  - Tipo de parámetro: ruta
  - Respuesta:

```

1 {
2     "flyers": [
3         {
4             "departments": [
5                 {
6                     "planta": "string",
7                     "nombre": "string",
8                     "tienda": "string",
9                     "fk_id_tienda": integer,
10                    "id_departamento": integer
11                }, ...
12         ],
13         "fecha_inicio": "string",
13

```

```

14     "fecha_fin": "string",
15     "descripcion": "string",
16     "products": [
17       {
18         "precio_venta": "string",
19         "producto": "string",
20         "tienda": "string",
21         "promocion": "string",
22         "marca": "string",
23         "stock": integer},
24         ...],
25         "titulo": "string",
26         "id_anuncio": integer,
27         "ruta_imagen": "string"
28     },
29   ...
}

```

#### RFGPPR27 Actualizar token de Firebase Cloud Messaging para cliente.

- Ruta: /client/update-token-fcm
  - Método: POST
  - Cuerpo de la petición:

```

1 {
2   "id_persona" : string,
3   "token": string
4 }

```

- Respuesta exitosa:

```

1 {
2   "message" : string
3 }

```

#### RFGPPR28 Publicar ubicación de cliente en Kafka.

- Ruta: /client/location
  - Método: POST
  - Cuerpo de la petición:

```

1 {
2   "id_person" : integer,
3   "fk_id_departament": integer,
4   "id_beacon": integer
5 }

```

- Respuesta exitosa:

```

1 {
2   "message" : string
3 }

```

---

A continuación se muestran los servicios REST para los **RFAIDP Requerimientos Funcionales de Aplicación Interactiva Difusora de Productos**.

### RFAIDP1 Iniciar sesión

- Ruta: /client/login/
  - Método: POST
  - Tipo de parámetro: ruta
  - Cuerpo de la petición:

```

1 {
2     "email": string,
3     "password": string,
4 }
```

- Respuesta:

```

1 {
2     "nombre" : "string",
3     "apellido_paterno" : "string",
4     "apellido_materno" : "string",
5     "email": "string",
6     "password" : "string",
7     "estado_civil" : "string",
8     "fecha_nacimiento" : "string",
9     "no_hijos" : "string",
10    "imagen" : "string"
11 }
```

### RFAIDP2 Consultar folletos

- Ruta: /dashboard-client/
  - Método: GET
  - Tipo de parámetro: ruta
  - Respuesta:

```

1 {
2     "flyers": [
3         {
4             "departments": [
5                 {
6                     "planta": "string",
7                     "nombre": "string",
8                     "tienda": "string",
9                     "fk_id_tienda": integer,
10                    "id_departamento": integer
11                }, ...
12            ],
13            "fecha_inicio": "string",
14            "fecha_fin": "string",
15            "descripcion": "string",
16        }
17    ]
18 }
```

```

16     "products": [
17         {
18             "precio_venta": "string",
19             "producto": "string",
20             "tienda": "string",
21             "promocion": "string",
22             "marca": "string",
23             "stock": integer},
24             ...],
25             "titulo": "string",
26             "id_anuncio": integer,
27             "ruta_imagen": "string"
28         },
29     ...

```

#### RFAIDP4 Consultar logros

- Ruta: /client/get-achievements/[idcliente]
  - Tipo de parámetro: ruta
  - Opciones: [id\_cliente, int]
  - Método: GET
  - Respuesta:

```

1 {
2     "id_logro": integer,
3     "nombre": "string",
4     "descripcion": "string",
5     "icono": "string",
6     "recompensa": "string"
7 }

```

#### RFAIDP5 Añadir producto a favoritos

- Ruta: /client/favourites
  - Método: POST
  - Cuerpo de la petición:

```

1 {
2     "id_persona": integer,
3     "id_producto": integer,
4     "favorito": boolean (true)
5 }

```

- Respuesta:

```

1 {
2     "message": "string"
3 }

```

#### RFAIDP6 Visualizar favoritos

- Ruta: /client/favourites?id\_persona=[idcliente]

- Tipo de parámetro: ruta
- Opciones: [id\_cliente, int]
- Método: GET
- Respuesta:

```

1  {
2      "id_producto": integer ,
3      "producto": "string",
4      "precio_venta": "string",
5      "stock": integer,
6      "departamento": "string",
7      "tienda": "string",
8      "marca": "string",
9      "promocion": "string",
10     "categoria": "string",
11     "image": "string"
12 }
```

#### RFAIDP7 Consultar recomendaciones

- Ruta: /client/get-global-recom/[idcliente]
- Tipo de parámetro: ruta
- Opciones: [id\_cliente, int]
- Método: GET
- Respuesta:

```

1  {
2      "id_producto": integer ,
3      "producto": "string",
4      "precio_venta": "string",
5      "stock": integer,
6      "departamento": "string",
7      "tienda": "string",
8      "marca": "string",
9      "promocion": "string",
10     "categoria": "string",
11     "image": "string"
12 }
```

#### RFAIDP8 Buscar productos

- Ruta: /products/busqueda?product=[idproducto]
- Método: GET
- Tipo de parámetro: ruta
- Opciones: [id\_producto, int]
- Respuesta:

```

1  {
2      "id_producto": integer ,
3      "producto": "string",
4      "precio_venta": "string",
5      "stock": integer,
6      "departamento": "string",
7      "tienda": "string",
8      "marca": "string",
9      "promocion": "string",
10     "categoria": "string",
11     "imagenes": "string"
12 }
```

## RFAIDP9 Actualizar datos personales

Para este caso de uso se presentan tanto la sección para mostrar los datos que ya fueron actualizados, como para modificar los datos de cada usuario.

- Ruta: /client/personal-info/[idcliente]
  - Método: GET
  - Tipo de parámetro: ruta
  - Opciones: [id\_cliente, int]
  - Respuesta:

```

1  {
2      "sexo": "string",
3      "estado_civil": "string",
4      "puntaje": int,
5      "fecha_nacimiento": "string"
6 }
```

- Ruta: /client/personal-info/[idcliente]
  - Método: POST
  - Tipo de parámetro: ruta
  - Opciones: [id\_cliente, int]
  - Cuerpo de la petición:

```

1  {
2      "sexo": "string",
3      "estado_civil": "string",
4      "puntaje": int,
5      "fecha_nacimiento": "string"
6 }
```

- Respuesta:

```

1  {
2      "message": "string"
3 }
```

## RFAIDP11 Actualizar gustos genéricos

Para este caso de uso se presentan tanto la sección para mostrar los datos que ya fueron actualizados, como para modificar dichas opciones.

- Ruta: /client/likes/[idcliente]

- Método: GET
- Tipo de parámetro: ruta
- Opciones: [id\_cliente, int]
- Respuesta:

```

1 {
2     "Hogar": boolean ,
3     "Computadoras": boolean,
4     "Celulares y tablets": boolean,
5     "TV y video": boolean,
6     "Camaras": boolean,
7     "Audio": boolean,
8     "Videojuegos": boolean,
9     "Drones y radio control": boolean,
10    "Wearables": boolean,
11    "Instrumentos musicales": boolean,
12    "Hogar inteligente": boolean,
13    "Peliculas y series": boolean,
14    "Autos y motos": boolean
15 }
```

- Ruta: /client/likes/[idcliente]

- Método: POST
- Tipo de parámetro: ruta
- Opciones: [id\_cliente, int]
- Cuerpo de la petición:

```

1 {
2     "Hogar": boolean ,
3     "Computadoras": boolean,
4     "Celulares y tablets": boolean,
5     "TV y video": boolean,
6     "Camaras": boolean,
7     "Audio": boolean,
8     "Videojuegos": boolean,
9     "Drones y radio control": boolean,
10    "Wearables": boolean,
11    "Instrumentos musicales": boolean,
12    "Hogar inteligente": boolean,
13    "Peliculas y series": boolean,
14    "Autos y motos": boolean
15 }
```

- Respuesta:

```

1 {
2     "message": "string"
3 }
```

### RFAIDP12 Consultar beneficios

- Ruta: /client/get-benefits/[idcliente]

- Tipo de parámetro: ruta
- Opciones: [id\_cliente, int]
- Método: GET
- Respuesta:

```

1 {
2     "id_beneficio": integer,
3     "descripcion": "string",
4     "porcentaje": integer,
5     "gratificacion": integer,
6     "producto_gratis": "string"
7 }
```

### RFAIDP13 Actualizar permisos

Para este caso de uso se presentan tanto la sección para mostrar los datos que ya fueron actualizados, como para modificar dichos permisos.

- Ruta: /client/permissions/[idcliente]

- Método: POST
- Tipo de parámetro: ruta
- Opciones: [id\_cliente, int]
- Cuerpo de la petición:

```

1 {
2     "edad": boolean ,
3     "estado_civil": boolean,
4     "hijos": boolean,
5     "compras": boolean,
6     "favoritos": boolean,
7     "recom_presenciales": boolean
8 }
```

- Respuesta:

```

1 {
2     "message": "string"
3 }
```

- Ruta: /client/permissions/[idcliente]
- Tipo de parámetro: ruta

- Opciones: [id\_cliente, int]

- Método: GET
- Respuesta:

```

1  {
2      "edad": boolean ,
3      "estado_civil": boolean,
4      "hijos": boolean,
5      "compras": boolean,
6      "favoritos": boolean,
7      "recom_presenciales": boolean
8 }
```

### RFAIDP16 Registrar cuenta nueva

- Ruta: /client
- Método: POST
- Cuerpo de la petición:

```

1  {
2      "nombre" : "string",
3      "apellido_paterno" : "string",
4      "apellido_materno" : "string",
5      "email": "string",
6      "password" : "string",
7      "estado_civil" : "string",
8      "fecha_nacimiento" : "string",
9      "no_hijos" : "string",
```

- Respuesta:

```

1  {
2      "message": "string"
3 }
```

### RFAIDP17 Recuperar contraseña

- Ruta: /client/retrieve-password
- Método: POST
- Cuerpo de la petición:

```

1  {
2      "email": "string"
3 }
```

- Respuesta:

```

1  {
2      "message": "string"
3 }
```

## RFAIDP18 Eliminar producto de favoritos

- Ruta: /client/favourites
- Método: POST
- Cuerpo de la petición:

```

1 {
2     "id_persona": integer,
3     "id_producto": integer,
4     "favorito": boolean (false)
5 }
```

- Respuesta:

```

1 {
2     "message": "string"
3 }
```

## RFAIDP19 Mostrar información de productos

- Ruta: /products/[idproducto]
- Método: GET
- Tipo de parámetro: ruta
- Opciones: [id\_producto, int]
- Respuesta:

```

1 {
2     "id_producto": integer ,
3     "producto": "string",
4     "precio_venta": "string",
5     "stock": integer,
6     "departamento": "string",
7     "tienda": "string",
8     "marca": "string",
9     "promocion": "string",
10    "categoria": "string",
11    "imagenes": ArrayList<String> "string"
12 }
```

### 6.2.3. Prototipo 2: Integración de algoritmo de agrupamiento K-medias

#### 6.2.3.1. Análisis

Dentro del análisis para el desarrollo de este prototipo se incorporan los siguientes requerimientos funcionales:

- RFGPPR15 Ejecutar algoritmo de agrupamiento.
- RFGPPR16 Obtener clusters y características.

definidos previamente en el capítulo del “Bosquejo general de la aplicación” con el título de “Requerimientos funcionales del Sistema de Gestión, Procesamiento y Proveedor de Datos Retail”.

### 6.2.3.2. Diseño

Dentro de esta subsección se encuentra la documentación de los servicios REST separados por requerimientos.

*Nota: En algunos servicios se hace uso de una notación específica ([texto, tipo de dato]), esta notación significa que el contenido del lado izquierdo de la coma es dinámico, por otro lado, el contenido derecho es el tipo de dato que debe ser el contenido dinámico.*

#### RFGPPR15 Ejecutar algoritmo de agrupamiento.

- Ruta: /administrador/execute-cluster

- Método: GET
- Respuesta exitosa:

```

1  {
2      "users": [
3          {
4              "posicion" : string,
5              "grupo" : integer,
6              "posicion_geografica" : [float, float]
7          }, ...
8      ],
9      "number_of_clusters" : integer
10     "info_clusters": [
11         {
12             "label" : string
13             "edad_promedio" : float,
14             "meses_registro": string
15             "estado_civil" : {}
16             "sexo" : {}
17             "Gusto1": {"0" : integer, "1" : integer}
18             .
19             .
20             .
21             "GustoN" : {"0" : integer, "1" : integer}
22         }, ...
23     ]
24 }
25
26 }
```

- Respuesta errónea:

```

1  {
2      "message": "string",
3 }
```

#### RFGPPR16 Obtener clusters y características.

- Ruta: /administrador/get-stored-cluster

- Método: GET
- Respuesta exitosa:

```

1  {
2      "users": [
3          {
4              "posicion" : string,
5              "grupo" : integer,
6              "posicion_geografica" : [float, float]
7          }, ...
8      ],
9      "number_of_clusters" : integer
10     "info_clusters": [
11         {
12             "label" : string
13             "edad_promedio" : float,
14             "meses_registro": string
15             "estado_civil" : {}
16             "sexo" : {}
17             "Gusto1": {"0" : integer, "1" : integer}
18             .
19             .
20             .
21             "GustoN" : {"0" : integer, "1" : integer}
22         }, ...
23     ]
24 }
25
26 }
```

- Respuesta erronea:

```

1  {
2     "message": "string",
3 }
```

#### 6.2.4. Prototipo 3: Integración de sistema de recomendación basado en filtrado colaborativo primera versión

Dentro del análisis para el desarrollo de este prototipo se incorporan los siguientes requerimientos funcionales:

- RFGPPR3 Generar módulo de recomendaciones basado el filtrado colaborativo.
- RFGPPR22 Obtener recomendaciones globales
- RFGPPR23 Obtener recomendaciones por departamento.

definidos previamente en el capítulo del “Bosquejo general de la aplicación” con el título de “Requerimientos funcionales del Sistema de Gestión, Procesamiento y Proveedor de Datos Retail”.

##### Simbología necesaria:

W := Matriz de pesos de los usuarios.

X := Matriz de pesos de productos.

p := valor de la predicción.

$W^{(usuario)}$  := Vector de pesos de un usuario.

$X^{(producto)}$  := Vector de pesos de un producto.

$y^{(i,j)}$  := Calificación para el producto i hecha por el usuario j.

$r(i, j) = 1$  := Producto i calificado por usuario j.

#### 6.2.4.1. Análisis

Durante esta sección se explicarán algunos de los elementos del algoritmo Filtrado Colaborativo comenzando con el diagrama de flujo de dicho algoritmo que se muestra en la figura 6.53

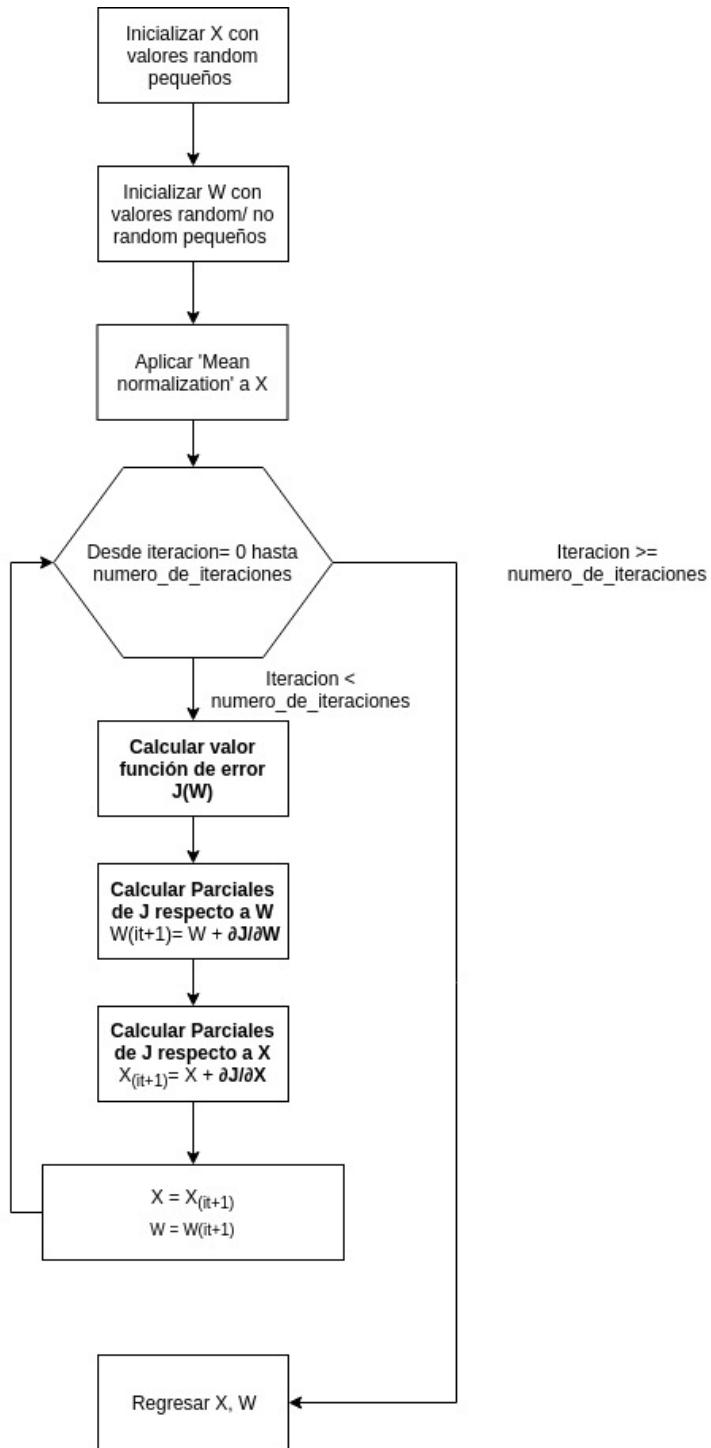


Figura 6.53: Diagrama de flujo de algoritmo Filtrado Colaborativo.

### Predicción de calificación por usuario

La calificación predicha de cada producto para cierto usuario se define de la siguiente manera:

$$p = W^{usuario^T} X^{producto} \quad (6.1)$$

6.1: Definición matemática de la predicción de un producto.

Esta ecuación significa lo siguiente: producto punto del vector de pesos de un usuario por el vector de pesos de cada producto.

### Función de error

La función de error seleccionada para este algoritmo se llama **Error cuadrático medio**. Esta medida se seleccionó porque por cada error cometido por el modelo éste se penaliza de una manera cuadrática además de tener una representación gráfica muy simple y lógica.

$$J = \frac{1}{num.productos} \sum_i^{num.productos} \sum_j^{num.usuarios} (W^{(j)T} X^{(i)} - y^{(i,j)})^2 \quad (6.2)$$

6.2: Definición matemática de la función de error.

### Gradiente descendiente

Esta técnica se utiliza para encontrar el mínimo de una función approximando éste de una manera numérica, como lo dice su nombre se hace encontrando la taza de cambio de  $J$  respecto a una variable (en el caso de Filtrado Colaborativo respecto a  $W$  y  $X$ )  $\frac{\partial(J)}{\partial(W)}$  y  $\frac{\partial(J)}{\partial(X)}$ , ya que, hay que minimizar el error de  $J$  respecto a dichas variables.

Finalmente, en la figura 6.54 se puede visualizar una representación gráfica en dos dimensiones de lo que hace el algoritmo.

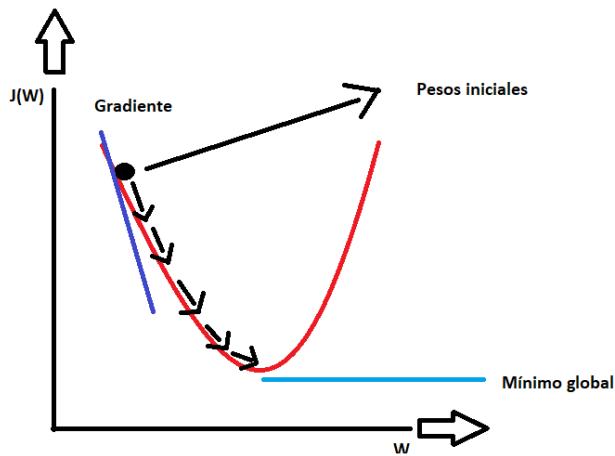


Figura 6.54: Gradiente descendiente.

Gradiente de  $J$  respecto a  $W$  ( $\frac{\partial(J)}{\partial(W)}$ )

Como se mencionó en la sección de Gradiente descendiente una de las variables respecto a las cuales se tiene que minimizar  $J$  es  $W$ , esto significa ajustar los Pesos de los usuarios de tal manera que  $J$  sea cada vez menor y por lo tanto lograr una generalización consistente con los datos. Esto se expresa de la siguiente manera:

$$W^{(j)} = W^{(j)} - \alpha \sum_{i:r(i,j)=1} ((W^j)^T X^{(i)} - y^{(i,j)}) X^{(i)} \quad (6.3)$$

### 6.3: Definición matemática del gradiente respecto a $W$ .

Como se puede observar en la ecuación 6.4 se hace la actualización de  $W^{(i)}$  restando un término, este término es el gradiente multiplicado por un coeficiente de aprendizaje ( $\alpha$ ) que establece el tamaño del movimiento hacia el mínimo.

#### Gradiente de $J$ respecto a $X$ ( $\frac{\partial(J)}{\partial(X)}$ )

Como se mencionó en la sección de Gradiente descendiente la otra variable a minimizar  $J$  es  $X$ , esto significa ajustar los Pesos de los productos de tal manera que  $J$  sea cada vez menor y por lo tanto lograr una generalización consistente con los datos. Esto se expresa de la siguiente manera:

$$X^{(i)} = X^{(i)} - \alpha \sum_{i:r(i,j)=1} ((W^j)^T X^{(i)} - y^{(i,j)}) W^{(j)} \quad (6.4)$$

### 6.4: Definición matemática del gradiente respecto a $X$ .

Como se puede observar en la ecuación 6.4 se hace la actualización de  $W^{(i)}$  restando un término, este término es el gradiente multiplicado por un coeficiente de aprendizaje ( $\alpha$ ) que establece el tamaño del movimiento hacia el mínimo.

#### Aplicar Mean normalization a $X$

*Este algoritmo es explicado y aplicado en el siguiente prototipo.*

#### 6.2.4.2. Diseño

##### RFGPPR22 Obtener recomendaciones globales

- Ruta: /client/get-global-recom/id\_usuario

- Método: GET
- Tipo de parámetro: ruta
- Opciones [id\_usuario, int]
- Respuesta exitosa:

```

1 { "products": [
2     "id_producto" : integer,
3     "producto": "string",
4     "precio_venta" : "string",
5     "stock" : integer,
6     "departamento" : "string",
7     "tienda" : "string",
8     "categoria" : "string",
9     "marca" : "string",
10    "promocion" : "string",
11    "imagen": "string"

```

```

12         "score" : float
13     } ,
14     { ... } ,
15     { ... } ,
16     ... ] }
```

- Respuesta erronea:

```

1 {
2     "message": "string",
3 }
```

### RFGPPR23 Obtener recomendaciones por departamento.

- Ruta: /client/recommendations/department/(id\_usuario)/id\_departamento
  - Método: GET
  - Tipo de parámetro: ruta
  - Opciones [id\_usuario, int], [id\_departamento, int]
  - Respuesta exitosa:

```

1     "products": [
2         {
3             "id_producto" : integer,
4             "producto": "string",
5             "precio_venta" : "string",
6             "stock" : integer,
7             "departamento" : "string",
8             "tienda" : "string",
9             "categoria" : "string",
10            "marca" : "string",
11            "promocion" : "string",
12            "imagen": "string"
13            "score" : float
14        } ,
15        { ... } ,
16        { ... } ,
17        ... ] }
```

- Respuesta erronea:

```

1 {
2     "message": "string",
3 }
```

#### 6.2.4.3. Pruebas

*Las pruebas del sistema fueron hechas sobre una laptop Lenovo Thinkpad T460 con las siguientes características:*

- Procesador Intel® Core™ i5-5300U (3M Cache, hasta 2,90 GHz).
- 8 GB RAM.
- 240 GB Solid State Drive ROM.

- Sistema Operativo Fedora 27.

**Prueba 1.**

- Número de clientes: 184
- Número de productos: 10001
- Número de compras: 144192
- Número de iteraciones: 30
- Número de parámetros: 23
- Tasa de aprendizaje: 0.01

**Prueba con validación: 70 % entrenamiento 30 % validación**

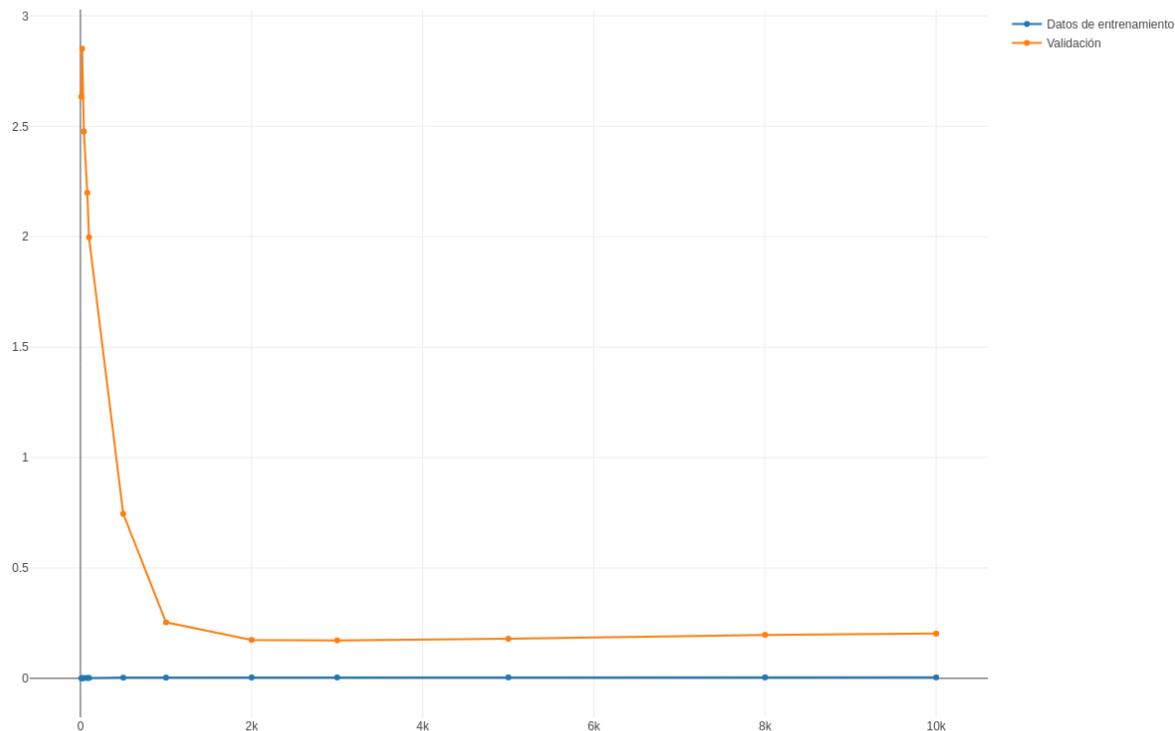


Figura 6.55: Prueba 1 con validación.

**Prueba de J respecto al número de iteraciones**

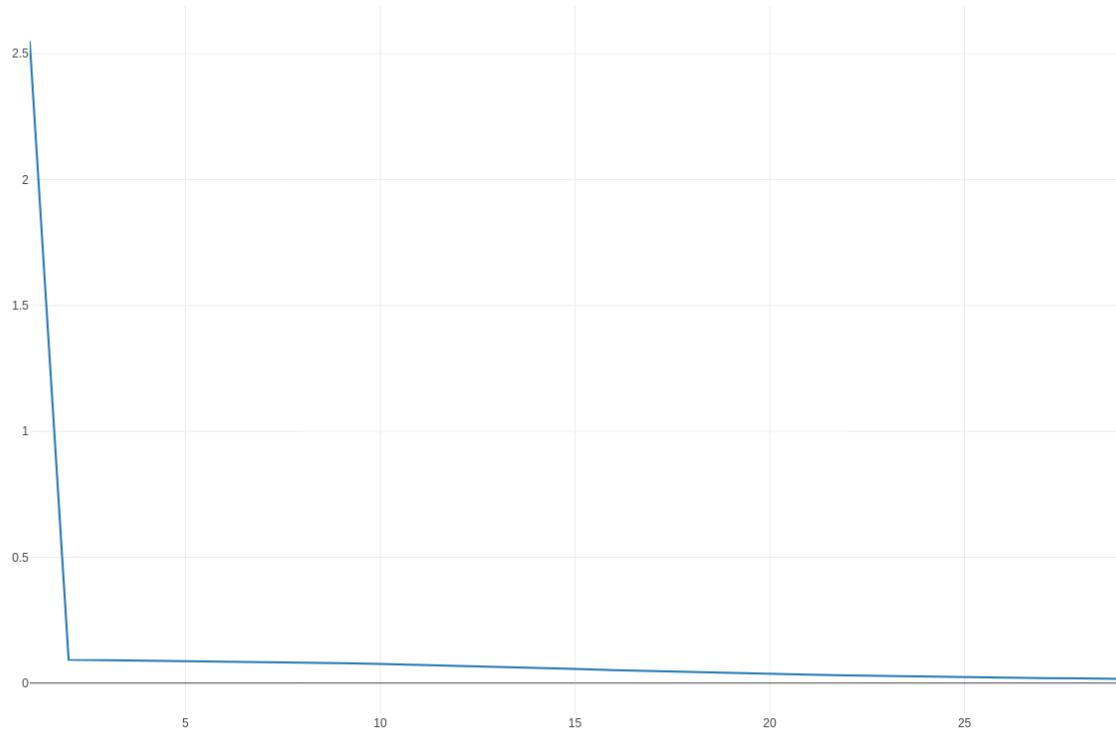


Figura 6.56: Prueba 1: J respecto al número de iteraciones.

**Prueba 2.**

- Número de clientes: 184
- Número de productos: 10001
- Número de compras: 144192
- Número de iteraciones: 15
- Número de parámetros: 13
- Tasa de aprendizaje: 0.01

**Prueba con validación: 70 % entrenamiento 30 % validación**

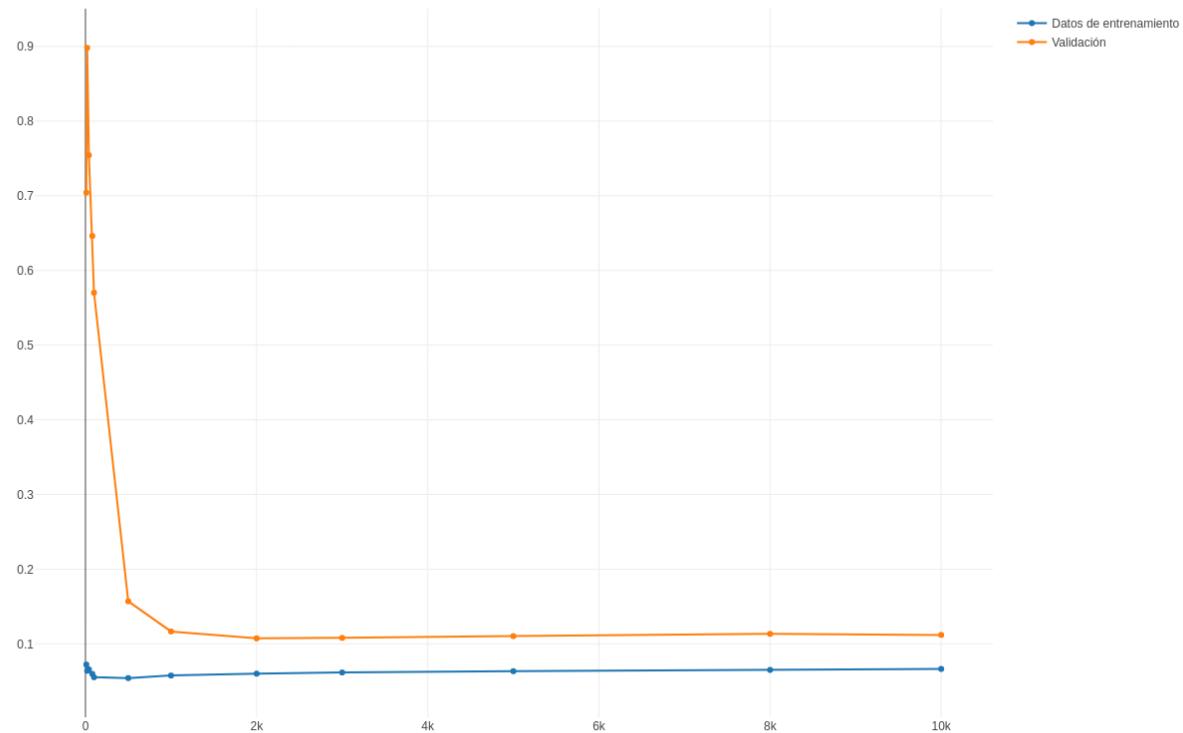


Figura 6.57: Prueba 2 con validación.

**Prueba de J respecto al número de iteraciones**

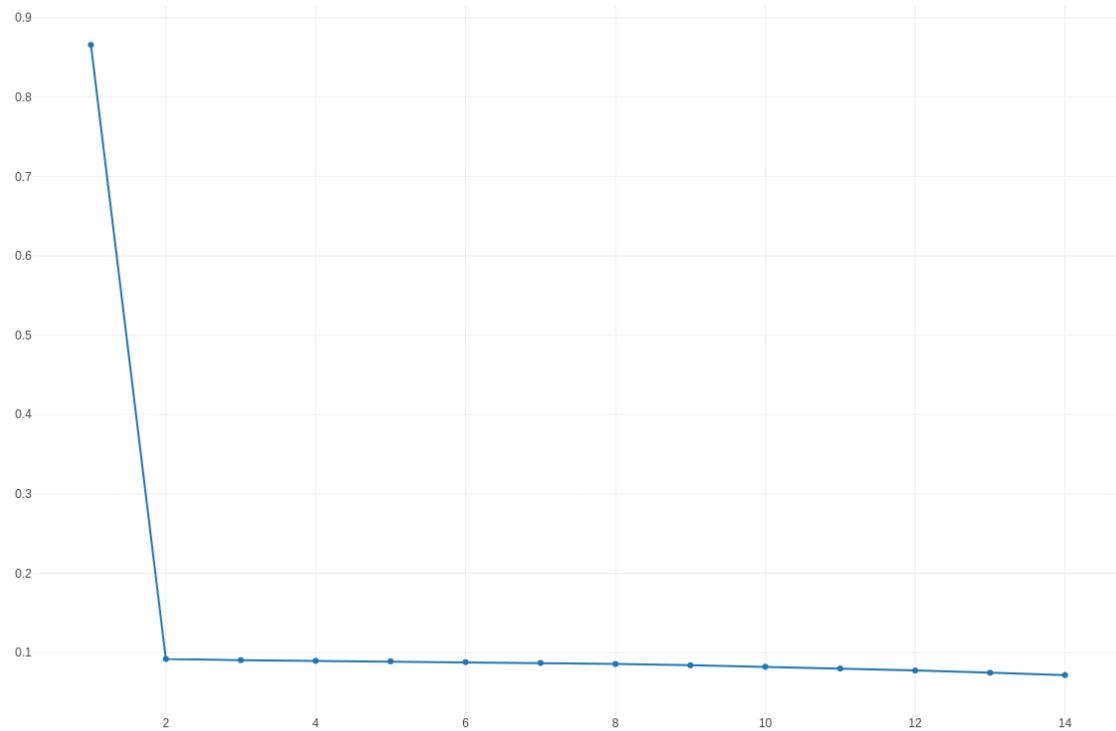


Figura 6.58: Prueba 2: J respecto al número de iteraciones.

**Prueba 3.**

- Número de clientes: 184
- Número de productos: 10001
- Número de compras: 144192
- Número de iteraciones: 5
- Número de parámetros: 13
- Tasa de aprendizaje: 0.01

**Prueba con validación: 70 % entrenamiento 30 % validación**

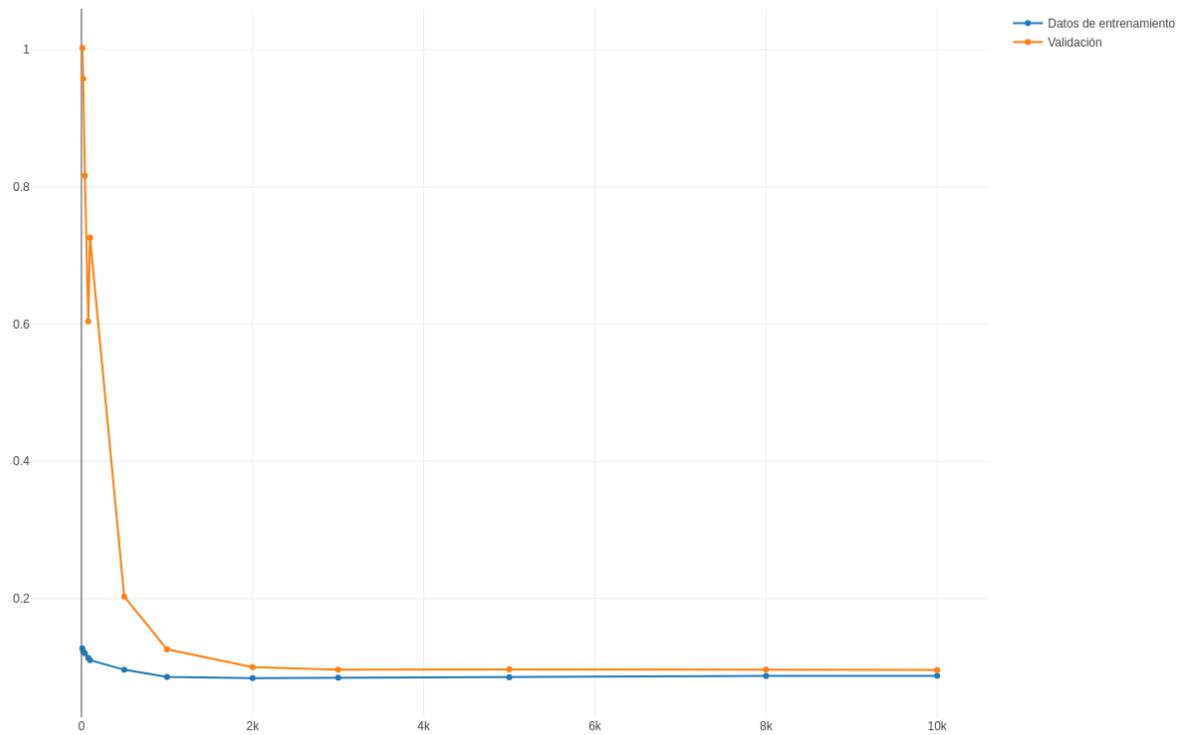


Figura 6.59: Prueba 3 con validación.

### Prueba de J respecto al número de iteraciones

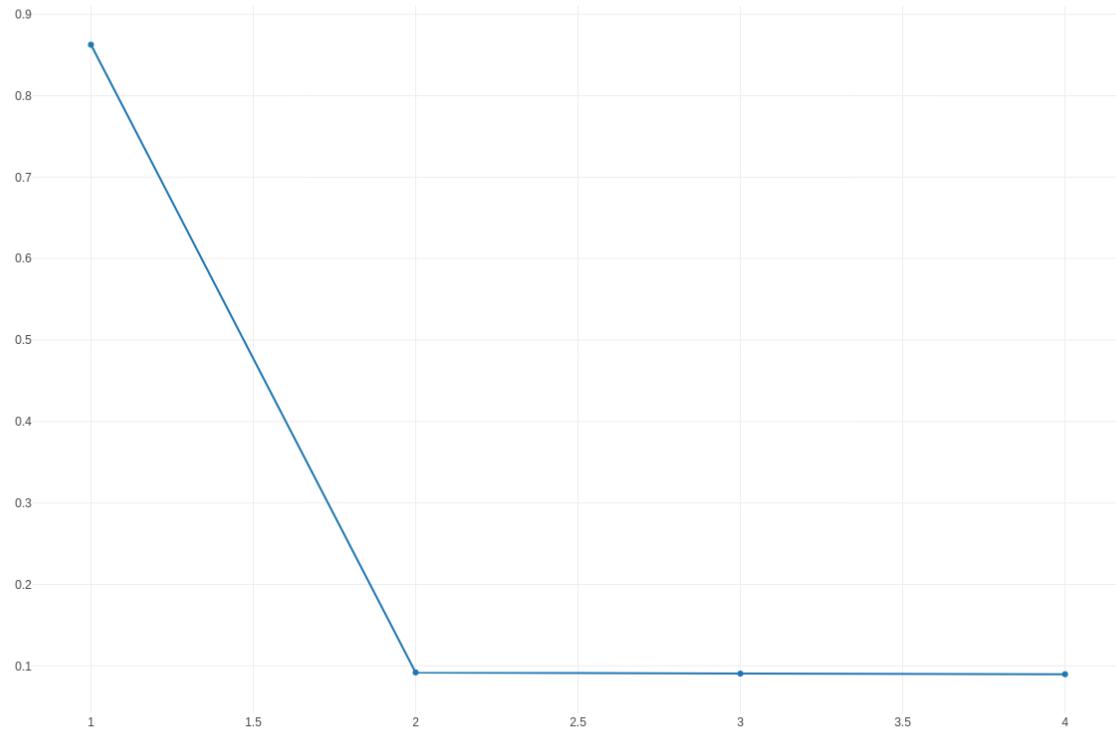


Figura 6.60: Prueba 3: J respecto al número de iteraciones.

#### 6.2.5. Prototipo 4: Integración de sistema de recomendación basado en filtrado colaborativo segunda versión

Dentro del análisis para el desarrollo de este prototipo se incorporan los siguientes requerimientos funcionales:

- RFGPPR3 Generar módulo de recomendaciones basado el filtrado colaborativo.
- RFGPPR22 Obtener recomendaciones globales
- RFGPPR23 Obtener recomendaciones por departamento.

definidos previamente en el capítulo del “Bosquejo general de la aplicación” con el título de “Requerimientos funcionales del Sistema de Gestión, Procesamiento y Proveedor de Datos Retail”.

#### Simbología necesaria:

$W$  := Matriz de pesos de los usuarios.

$X$  := Matriz de pesos de productos.

$p$  := valor de la predicción.

$W^{(usuario)}$  := Vector de pesos de un usuario.

$X^{(producto)}$  := Vector de pesos de un producto.

$y^{(i,j)}$  := Calificación para el producto i hecha por el usuario j.

$r(i, j) = 1$  := Producto i calificado por usuario j.

### 6.2.5.1. Análisis

Durante esta sección se explicarán algunas mejoras aplicadas al algoritmo de Filtrado Colaborativo.

#### Función de error

La función de error sigue siendo el **Error cuadrático medio** pero como se puede observar en la ecuación 6.2 existen dos sumatorias que corresponden a una técnica llamada Regularización, esta técnica nos permite poner una cota superior para los valores de los vectores X y W con el objetivo de evitar el famoso sobreentrenamiento.

$$J = \frac{1}{\text{num.productos}} \sum_i^{\text{num.productos}} \sum_j^{\text{num.usuarios}} (W^{(j)} X^{(i)} - y^{(i,j)})^2 + \frac{\lambda}{\text{num.productos}} \sum_{k=1}^{\text{num.productos}} X^{(k)}^2 + \frac{\lambda}{\text{num.usuarios}} \sum_{k=1}^{\text{num.usuarios}} W^{(k)}^2 \quad (6.5)$$

6.5: Definición matemática de la función de error.

#### Mean Normalization

##### Introducción al problema

	Artículo 1	Artículo 2	...	Artículo j	Artículo n
Usuario 1	cal_1_1			cal_1_j	cal_1_n
Usuario 2		cal_2_2			cal_2_n
.	.				
Usuario i	cal_i_1			cal_i_j	
Usuario m	?	?	?	?	?

Figura 6.61: Matriz Artículo-Usuario con usuario nuevo.

En la figura ?? se puede observar una Matriz Artículo-Usuario con un usuario nuevo sin compras ni calificaciones. Dentro de esta matriz se genera un problema: éste surge al realizar las predicciones, ya que, el vector de pesos (W) contiene únicamente ceros y al realizar la predicción de algún producto esta es cero para cualquier X.

#### Solución

Mean normalization es una técnica que consiste en obtener la media de las calificaciones dada a algún productos por los usuarios que han adquirido dicho producto. En términos matemáticos significa lo siguiente:

$$\frac{\sum_{i:r(i, producto)=1}^{num_{usuarios}} y^{producto,i}}{\sum_{i:r(i, producto)=1}^{num_{usuarios}} 1} = \mu_{producto} \quad (6.6)$$

### 6.6: Definición matemática de Mean normalization.

Una vez obtenida la calificación media de cada producto se tiene que sustraer dicha media de la calificación que le había asignado el usuario.

Finalmente, tomando en cuenta la calificación media del producto las predicciones cambian a la siguiente manera:

$$p = W^T X^{producto} + \mu_{producto} \quad (6.7)$$

### 6.7: Definición matemática de predicción con Mean normalization.

#### Mejoras en tiempos de respuesta

En el prototipo 3 llamado **Integración de sistema de recomendación basado en filtrado colaborativo primera versión** se ejecutaba el algoritmo cada vez que algún usuario cliente o usuario vendedor realizaban una petición, el tiempo de respuesta aproximado para recomendaciones globales era de 30 minutos y el tiempo aproximado para recomendaciones por departamento era de 5 segundos.

#### ¿Cómo se mejoró el tiempo de respuesta?

Se escribió un script en python que debe ser ejecutado cada cierto para tener las recomendaciones globales **precargadas** en archivos json con que tienen como nombre el identificador único de cada cliente. De igual manera, cada vez que se ejecuta dicho script en el repositorio de datos son almacenados los valores optimizados de los vectores X y W, de esta manera cada vez que se hace una petición de recomendaciones por departamentos simplemente se calculan las predicciones con la ecuación ??.

#### ¿Qué pasa cuando un cliente nuevo entra al sistema y no tiene aún su archivo de recomendaciones globales precargado?

Al no tener un archivo con sus recomendaciones precargadas significa que el script de python que lo genera aún no ha sido ejecutado y por ende su vector de W está completamente en ceros. La solución al problema es simplemente aplicar la ecuación ?? por cuestiones de tiempos de respuesta, se decidió simplemente hacer una consulta al repositorio de datos buscando los cien productos con mayor calificación media (esto nos proporciona el mismo resultado que la ecuación ??); de esta manera a un usuario nuevo se le hacen recomendaciones de los productos que más le han gustado a todos los clientes.

#### 6.2.5.2. Diseño

##### RFGPPR22 Obtener recomendaciones globales

- Ruta: /client/get-global-recom/id\_usuario

- Método: GET
- Tipo de parámetro: ruta
- Opciones [id\_usuario, int]
- Respuesta exitosa:

```

1 { "products": [
2   "id_producto" : integer,
3   "producto": "string",

```

```

4         "precio_venta" : "string",
5         "stock" : integer,
6         "departamento" : "string",
7         "tienda" : "string",
8         "categoria" : "string",
9         "marca" : "string",
10        "promocion" : "string",
11        "imagen": "string"
12        "score" : float
13      },
14      { ... },
15      { ... },
16    ... ]

```

- Respuesta erronea:

```

1 {
2   "message": "string",
3 }

```

### RFGPPR23 Obtener recomendaciones por departamento.

- Ruta: /client/recommendations/department/(id\_usuario)/id\_departamento

- Método: GET
- Tipo de parámetro: ruta
- Opciones [id\_usuario, int], [id\_departamento, int]
- Respuesta exitosa:

```

1   "products": [
2     {
3       "id_producto" : integer,
4       "producto": "string",
5       "precio_venta" : "string",
6       "stock" : integer,
7       "departamento" : "string",
8       "tienda" : "string",
9       "categoria" : "string",
10      "marca" : "string",
11      "promocion" : "string",
12      "imagen": "string"
13      "score" : float
14    },
15    { ... },
16    { ... },
17  ... ]

```

- Respuesta erronea:

```

1 {
2   "message": "string",
3 }

```

## 6.2.6. Pruebas

*Las pruebas del sistema fueron hechas sobre una laptop Lenovo Thinkpad T460 con las siguientes características:*

- Procesador Intel® Core™ i5-5300U (3M Cache, hasta 2,90 GHz).
- 8 GB RAM.
- 240 GB Solid State Drive ROM.
- Sistema Operativo Fedora 27.

*Nota: Durante estas pruebas se agregó la regularización*

### Prueba 1.

- Número de clientes: 184
- Número de productos: 10001
- Número de compras: 144192
- Número de iteraciones: 30
- Número de parámetros: 13
- Tasa de aprendizaje: 0.01
- Regularización: 0.01

**Prueba con validación: 70 % entrenamiento 30 % validación**

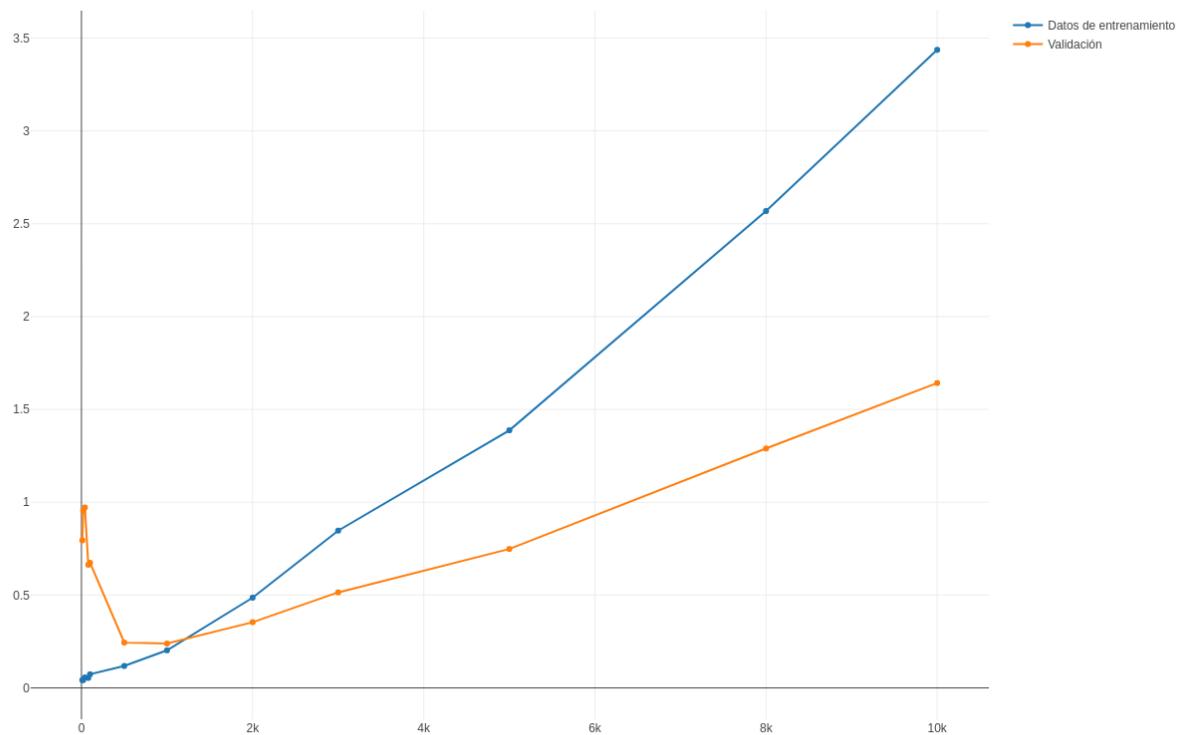


Figura 6.62: Prueba 1 con validación.

---

**Prueba de J respecto al número de iteraciones**

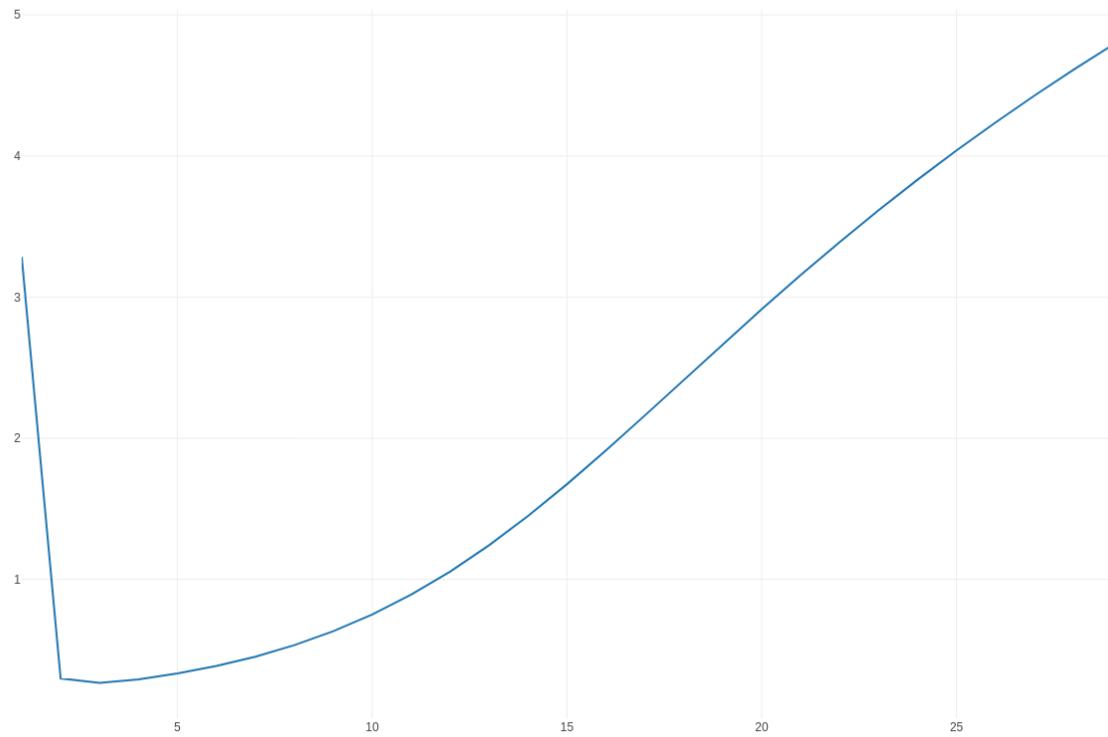


Figura 6.63: Prueba 1 con regularización: J respecto al número de iteraciones.

**Prueba 2.**

- Número de clientes: 184
- Número de productos: 10001
- Número de compras: 144192
- Número de iteraciones: 100
- Número de parámetros: 13
- Tasa de aprendizaje: 0.01
- Regularización: 0.000001

**Prueba con validación: 70 % entrenamiento 30 % validación**

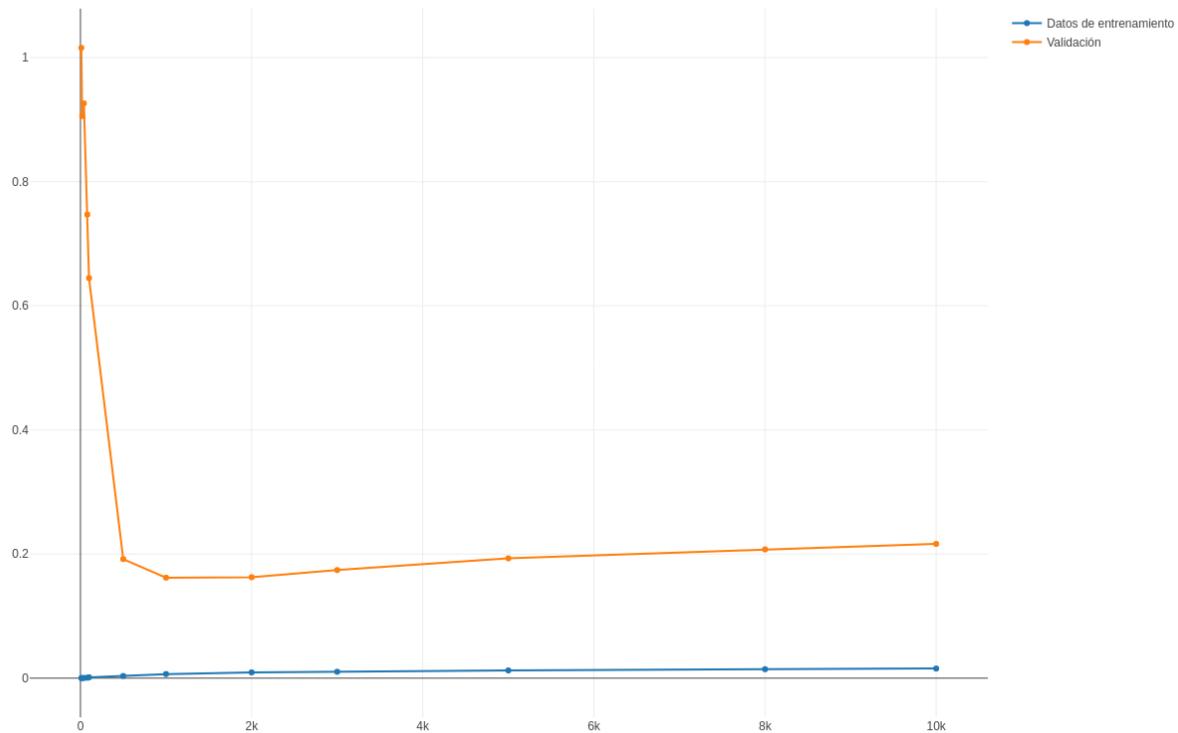


Figura 6.64: Prueba 2 con validación.

**Prueba de J respecto al número de iteraciones**

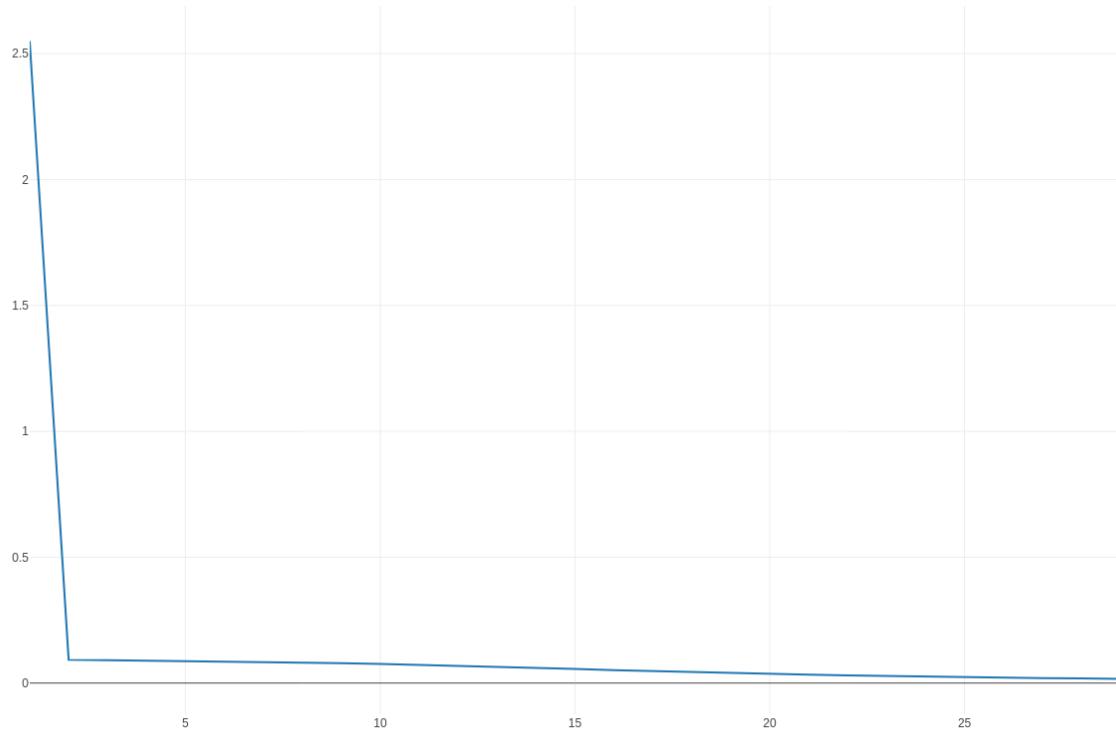


Figura 6.65: Prueba 2 con regularización: J respecto al número de iteraciones.

**Prueba 3.**

- Número de clientes: 184
- Número de productos: 10001
- Número de compras: 144192
- Número de iteraciones: 100
- Número de parámetros: 15
- Tasa de aprendizaje: 0.01
- Regularización: 0.000001

**Prueba con validación: 70 % entrenamiento 30 % validación**

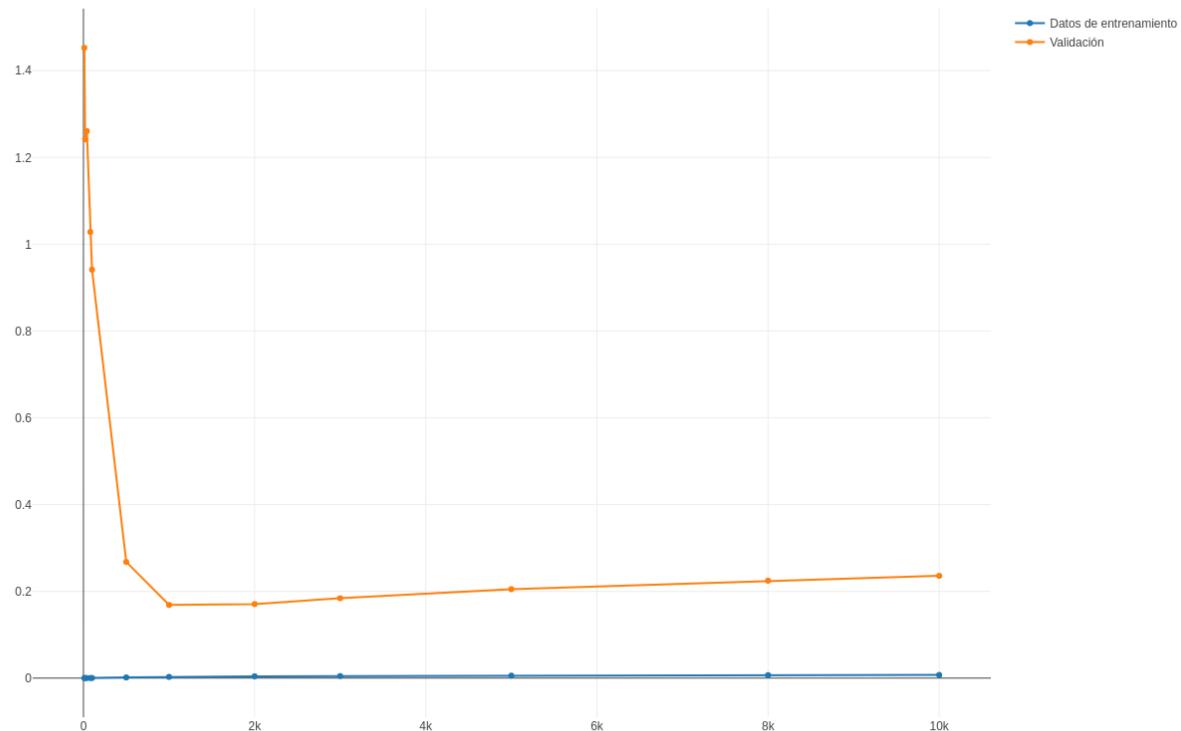


Figura 6.66: Prueba 3 con validación.

**Prueba de J respecto al número de iteraciones**

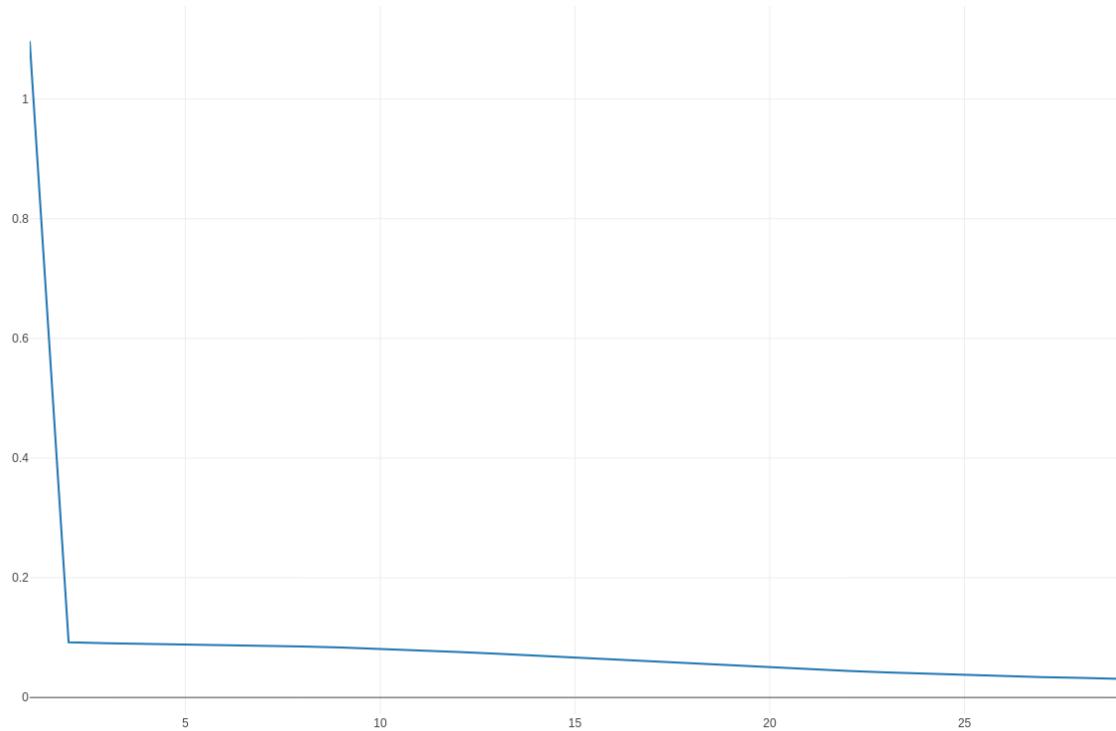


Figura 6.67: Prueba 3 con regularización: J respecto al número de iteraciones.

**Prueba 4.**

- Número de clientes: 184
- Número de productos: 10001
- Número de compras: 144192
- Número de iteraciones: 5
- Número de parámetros: 13
- Tasa de aprendizaje: 0.01
- Regularización: 0.000000005

**Prueba con validación: 70 % entrenamiento 30 % validación**

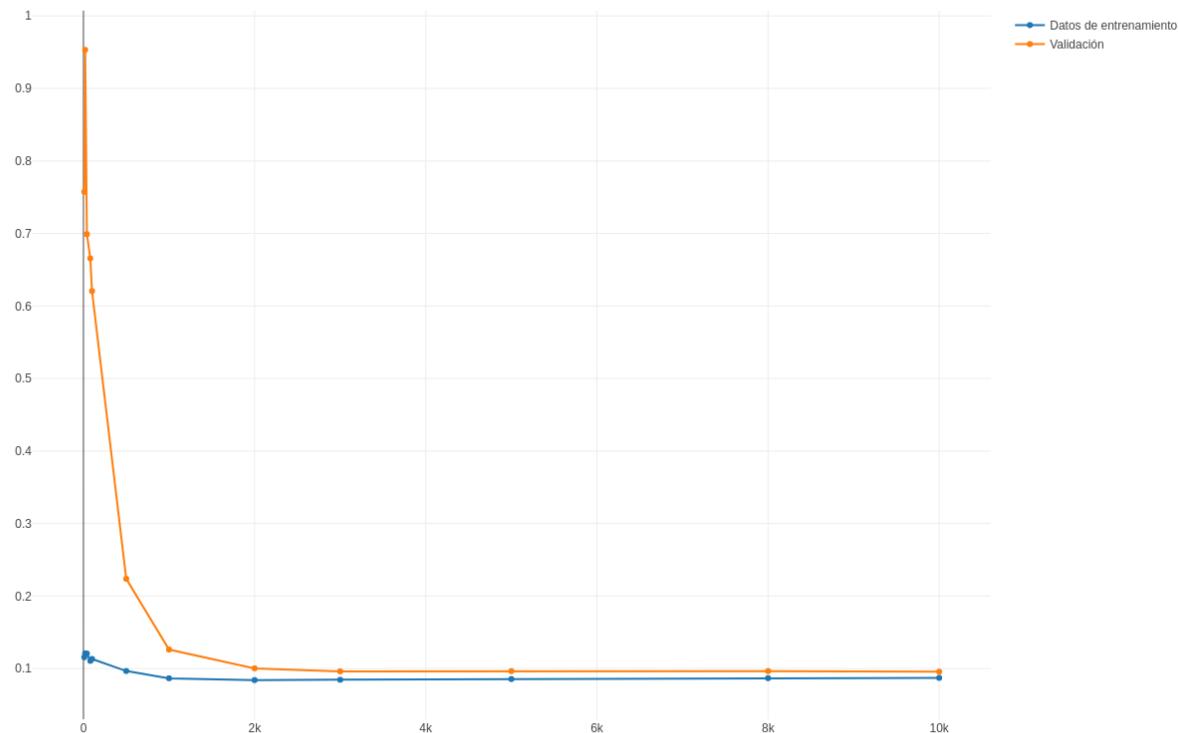


Figura 6.68: Prueba 3 con validación.

**Prueba de J respecto al número de iteraciones**

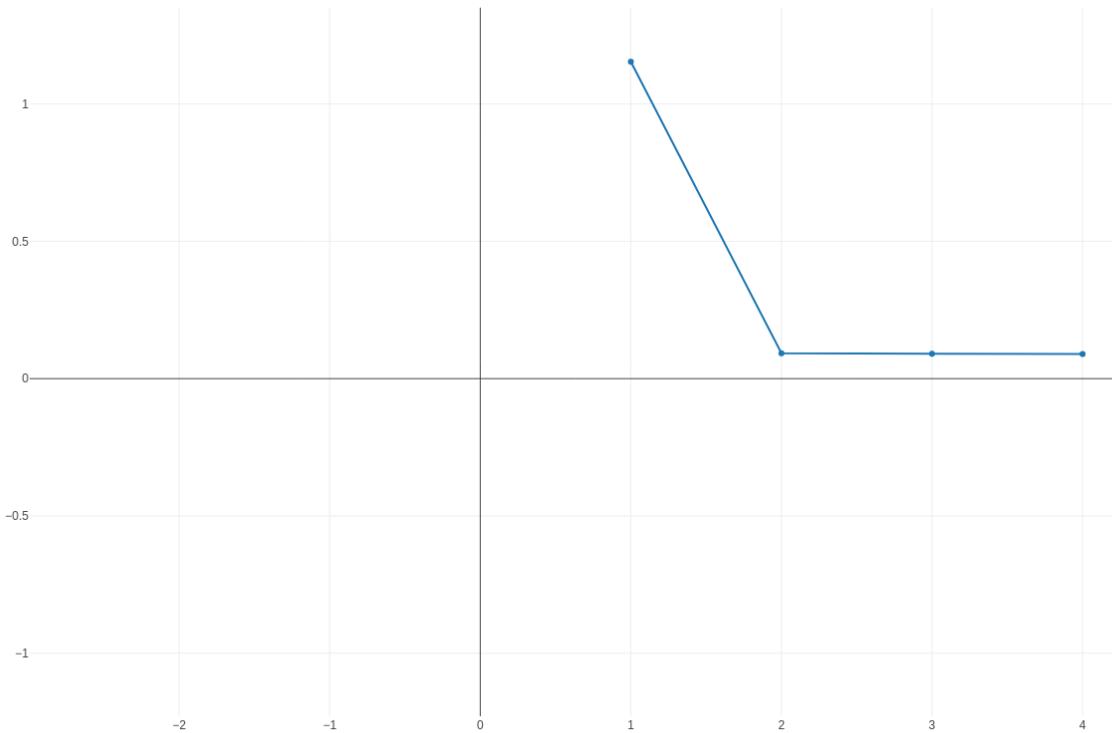


Figura 6.69: Prueba 3 con regularización: J respecto al número de iteraciones.

**Prueba 5.**

- Número de clientes: 184
- Número de productos: 10001
- Número de compras: 144192
- Número de iteraciones: 10
- Número de parámetros: 13
- Tasa de aprendizaje: 0.01
- Regularización: 0.000000005

**Prueba con validación: 70 % entrenamiento 30 % validación**

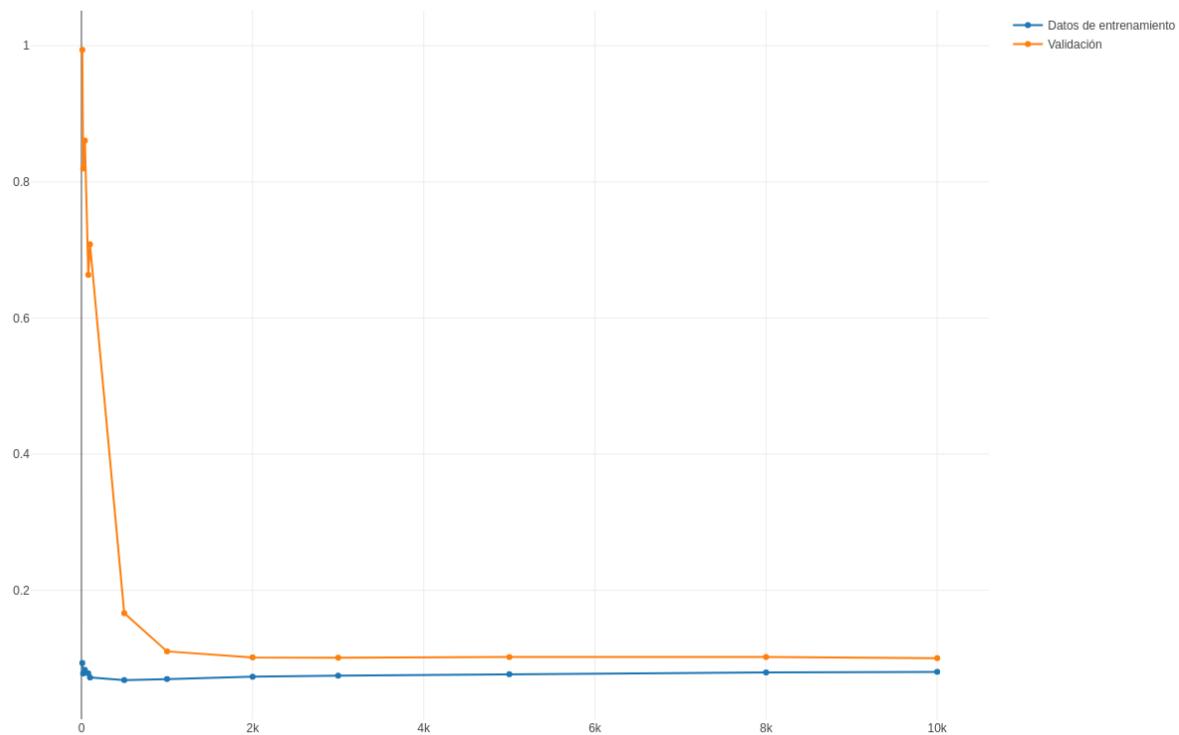


Figura 6.70: Prueba 3 con validación.

### Prueba de J respecto al número de iteraciones

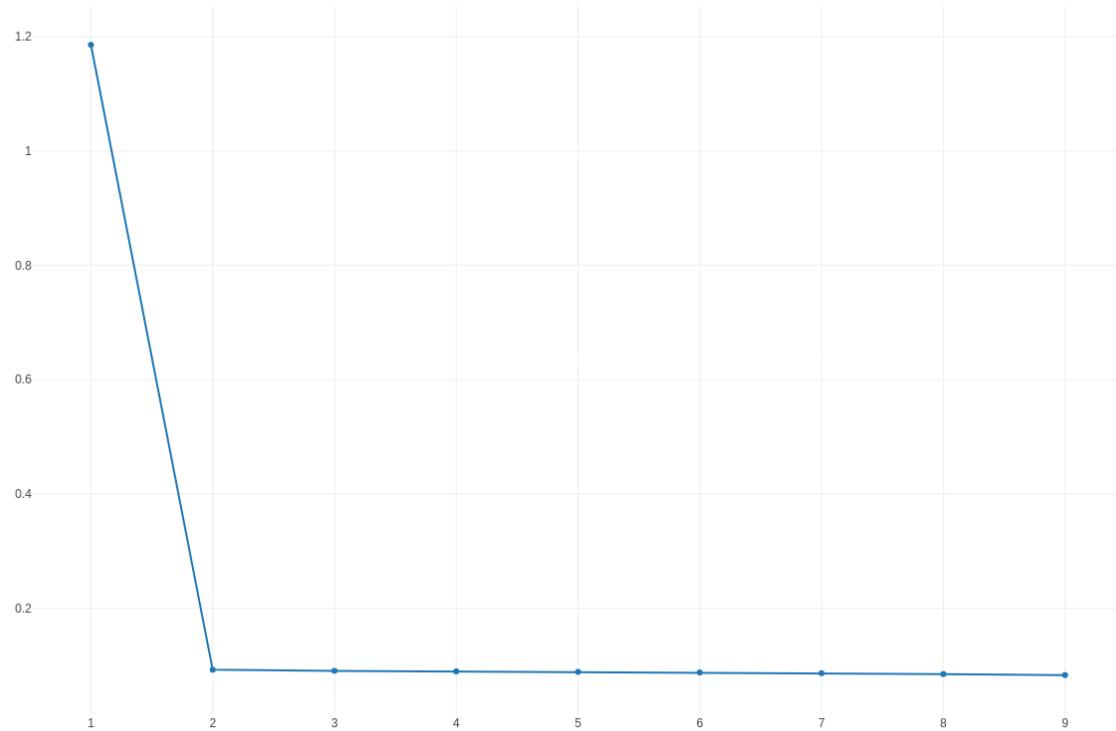


Figura 6.71: Prueba 3 con regularización: J respecto al número de iteraciones.

#### 6.2.6.1. Conclusiones

De acuerdo con las pruebas realizadas se decidió que la prueba número tres con regularización es la óptima, ya que, como se muestra en la gráfica 6.59 los datos de entrenamiento y validación tienen una convergencia muy cercana aproximadamente en 0.1. Por otro lado, dicha gráfica nos corrobora que no existe el sobreentrenamiento en nuestro modelo.

# CAPÍTULO 7

## Conclusiones

El presente proyecto contaba inicialmente con seis objetivos específicos derivados del objetivo general. A partir de los módulos y prototipos planteados inicialmente, se obtienen los siguientes resultados:

- Se desarrolló el prototipo de una aplicación móvil dirigida a los clientes de tiendas departamentales misma que obtiene satisfactoriamente promociones, recomendaciones personalizadas, anuncios y notificaciones sobre productos de interés para el usuario, provenientes del prototipo de la aplicación móvil para vendedores.
- El módulo encargado de la generación de registros artificiales, funcionó correctamente ya que proporcionó datos ficticios para simular un entorno real.
- El prototipo de la aplicación móvil dirigida al uso de los vendedores igualmente cumple las características planteadas inicialmente pues ofrece la funcionalidad de recomendar productos a los diferentes clientes y recibe notificaciones cuando un cliente requiere asistencia en sus compras.
- Se implementó una plataforma en línea con el fin de que los administradores tengan la posibilidad de gestionar anuncios, mismas que visualiza el cliente en su aplicación móvil. Así mismo, tiene la opción de observar gráficamente el mercado que se tiene actualmente junto con sus características más importantes.
- Los prototipos de ambas aplicaciones móviles, implementan la SDK de Estimote mediante la cual se obtiene la ubicación de los usuarios y de esta forma, el vendedor es capaz tanto de acudir con los clientes en caso de que requieran de ayuda como de realizar el envío de recomendaciones de productos.
- Se puso en funcionamiento un servidor REST encargado de realizar operaciones de lectura y escritura sobre el repositorio de datos y capaz de proveer recomendaciones a las aplicaciones móviles mencionadas anteriormente.

Con respecto al objetivo general en el Sistema de gestión, procesamiento y proveedor de datos de Retail se visualiza el diseño, implementación y resultados del sistema de recomendaciones basado en aprendizaje máquina, mismo que fungió como el núcleo de los módulos descritos anteriormente.

A partir de lo mencionado previamente, se puede corroborar que el proyecto propuesto proporcionó satisfactoriamente una solución a la problemática planteada al inicio del proyecto.

# CAPÍTULO 8

---

## Trabajo a futuro

---

### 8.1. Aplicación Interactiva Difusora de Productos

- Actualización de las dependencias de la SDK de Estimote a una nueva versión estable.
- Implementación de una base de datos local en el dispositivo móvil con el fin de almacenar los datos de ciertas promociones y productos y estos a su vez, puedan ser visualizados por el cliente sin necesidad de una conexión a Internet.

### 8.2. Aplicación Interactiva para el Personal de Ventas

- Actualización de las dependencias de la SDK de Estimote a una nueva versión estable.
- Visualización de un mapa en el cual el vendedor visualice la posición exacta en la cual se encuentra el cliente que solicita apoyo.
- Evitar el envío múltiple de notificaciones de recomendaciones de un mismo producto para los clientes.

### 8.3. Panel de Administración

- Funcionalidad para realizar el cambio de los attachments de cada Beacon registrado sin la necesidad de ingresar al Cloud de Estimote.
- Registro de anuncios para usuarios pertenecientes a un grupo específico.

### 8.4. Sistema de Recomendaciones

- Envío de notificaciones al cliente cada que el sistema encuentre una nueva recomendación vía correo electrónico.
- Agregar opción para cambiar el paradigma de aprendizaje a aprendizaje en línea con gradiente descendiente estocástica.

# CAPÍTULO 9

## Glosario de términos

### 9.0.1. Beacons

Los Beacons son pequeños dispositivos de posicionamiento que emiten señales Bluetooth 4.0 o BLE, mediante los cuales se puede realizar el envío de cualquier información cifrada a dispositivos que dispongan de este tipo de conectividad que se encuentren en un radio de acción de hasta 200 metros. Estos dispositivos funcionan de forma autónoma sin necesidad de carga y pueden ser colocados en diferentes tipos de superficies dentro de establecimientos con el fin de enviar y recibir información sobre el mismo [4].

En la figura 8.1 podemos observar los 3 elementos que constituyen internamente al Beacon, estos componentes son: la carcasa que cubre al emisor Bluetooth, dicho emisor y una pequeña pila que tendrá una duración de entre 1 y 5 años.



Figura 9.1: Arquitectura interna del Beacon [99].

#### 9.0.1.1. Firmware

El firmware en los Beacons es específico según el proveedor de estos y este permite controlar diferentes características tales como:

- Potencia de transmisión ó Tx Power, que se refiere a la potencia fija con la que los Beacons transmiten una señal, misma que disminuye a medida que se incrementa la distancia [100].

- Intervalo de emisión, que define la frecuencia con la que emite una señal el Beacon. Este intervalo dependerá del tiempo de emisión ya que en cuanto más bajo sea este intervalo, más energía será consumida por el dispositivo [100].

#### 9.0.1.2. Ventajas

- El tamaño de los Beacons es cómoda para las empresas, ya que va desde los 6 mm. hasta los 27 mm [101].
- No requieren ser cargados debido a que trabajan con Bluetooth Low Energy [101].
- Para el usuario la tecnología no resulta ser intrusiva puesto que antes de recibir notificaciones, debe instalar la app de la compañía que ofrece el servicio [101].
- La tecnología Bluetooth no depende de ninguna red de datos, por lo cual no consume la tarifa de datos del usuario [101].
- Permite conocer la geolocalización indoor del usuario [101].

Se muestra a continuación la tabla comparativa ?? entre los distintos Beacons existentes en el mercado, en la cual se puede visualizar la que el equipo ha seleccionado para este trabajo (Estimote) y las características que tanto la elegida como las demás proveen.

Beacons / Características	Estimote	BlueSense	Gelo	Gimbal	Kontakt	Sonic Notify
Reconocimiento	Han distribuido más de 10,000 kits. Próximos a convertirse en líderes manufac-tureros. Cuentan con su propia SDK	Son la empresa emergente con más reconocimiento en Reino Unido.	Funcionan en todo tipo de clima.	Desarrollados por Qual-comm, la única empresa grande de hardware que ha decidido manu-facturar Beacons.	Los favoritos en Beekn, blog dedicado especialmente a Beacons.	Los favoritos en Beekn, blog dedicado especialmente a Beacons.
Desventajas	Su SDK se encuentra en constantes actualizaciones.	Aún están trabajando en su SDK.	Sólo tienen soporte para iBeacon.	Se requiere pagar una fianza de utilización.	La fuerza de la señal depende de la orientación del Beacon.	Únicamente son vendidos a empresas de marketing.
Alcance máximo	70 metros	100 metros	S/I	50 metros	70 metros	S/I
Duración de batería	2 años	2 años	2 años	2-3 meses	4 años	S/I
Precio por paquetes	\$1,133.39 por 3 Beacons.	\$1,133.39 por 3 Beacons.	\$1,325.57 por 2 Beacons.	\$77.91 por 1 Beacon.	\$1,153.6 por 3 Beacons.	\$1,153.6 por 3 Beacons.

### 9.0.2. Bluetooth Low Energy (BLE)

El BLE, es una nueva tecnología digital de radio (inalámbrica) interoperable para pequeños dispositivos desarrollada por Bluetooth. Esta tecnología y el Bluetooth 4.0 son soportados por diferentes plataformas como:

- iOS5+ / iOS7
- Android 4.3+
- Apple OS X 10.6+
- Windows 8
- GNU/Linux 4.93+ [102]

BLE hace uso del Generic Access Profile(GAP) y del Generic Attribute Profile(GATT).

El GAP es responsable de definir la topología general de la red del BLE. Esta capa realiza el manejo de los modos de acceso y procedimientos del dispositivo tales como el descubrimiento del dispositivo (permitir que sea público o no) y determina como dos dispositivos pueden interactuar entre ellos.

GAP define diferentes roles para los dispositivos mismos que son categorizados en periféricos y centrales. Los periféricos son dispositivos pequeños, de baja potencia y de bajos recursos, que pueden conectarse a dispositivos centrales mucho más potentes. Un ejemplo de periférico es el Beacon; mientras que un dispositivo central corresponde normalmente a un teléfono móvil o una tableta que tienen una capacidad de proceso mucho mayor.

Por otra parte el GATT describe detalladamente el proceso de transferencia de los datos una vez que los dispositivos tienen una conexión establecida. Este se enfoca especialmente en como los datos son formados, empaquetados y enviados según las reglas descritas [103].

# CAPÍTULO 10

---

## Acrónimos

---

- GPS - Global Positioning System
- BLE - Bluetooth Low Energy
- IPN - Instituto Politécnico Nacional
- ESCOM - Escuela Superior de Cómputo
- TT - Trabajos Terminales
- API - Application Programming Interface
- REST - Representational State Transfer
- INEGI - Instituto Nacional de Estadística y Geografía
- SDK - Software Development Kit
- AI - Artificial Intelligence
- EBL - Explanation-Based Learning HTT
- DL - Deep Learning
- IBM - International Business Machines
- DNN - Deep Neuronal Network
- ANN - Artificial Neuronal Networks
- CBF - Colaborative Based Focus
- CF - Colaborative Filtering
- SVD - Singular Value Descomposition
- M2M - Machine to Machine
- M2P - Machine To Person

- P2P - Person To Person
- IoT - Internet of Things
- IoE - Internet of Everything
- RFID - Radio Frequency Identification
- IT - Information Technology
- PDA - Personal Digital Assistant
- CEO - Chief Executive Officer
- PC - Personal Computer
- UMPC - Ultra Mobile PC
- HTC - High Tech Computer Corporation
- HP - Hewlett-Packard
- TFT - Thin Film Transistor
- OS - Operative System
- CMOS - Complimentary Metal Oxide Semiconductor
- DLNA - Digital Living Network Alliance
- WP - Windows Phone
- VPN - Virtual Private Network
- GSM - Global System for Mobile communications
- HMI - High-Definition Multimedia Interface
- GNSS - Global Navigation Satellite System
- USB - Universal Serial Bus
- UWB - Ultra Wide Band
- UUID - Universally Unique Identifier
- URL - Uniform Ressource Locator
- TLM - Telemetry
- NFC - Near Field Communication
- Wi-Fi - Wireless Fidelity
- WiMAX - Worldwide Interoperability for Microwave Access
- QR - Quick Response barcode
- UI - User Interface
- CPU - Central Processing Unit
- JSON - JavaScript Object Notation

- HTTP - HyperText Transfer Protocol
- SQL - Structured Query Language
- AMD - Advanced Micro Devices
- RAM - Random Access Memory
- AMD - Advanced Micro Devices
- GTX - Game Tested eXtreme
- DDR5X - Double Data Rate 5 X
- HD - High Definition
- AngularJS - AngularJavaScript
- LISR - Ley de Impuesto Sobre la Renta
- TB - Tera Bytes
- GB - Giga Bytes
- MB - Mega Bytes
- GHz - Giga Hertz
- SACMEX - Sistema de Aguas de la Ciudad de México
- CFE - Comisión Federal de Electricidad
- TELMEX - Teléfonos Mexicanos
- kW/h - kiloWatt por hora
- PK - Primary Key
- FK - Foreign Key
- AIPV Aplicación Interactiva para el Personal de Ventas
- RFGRA: Requerimiento Funcional Generador Registros Artificiales
- RFAIDP - Requerimiento Funcional Aplicación Interactiva Difusora de Productos
- RFAPV - Requerimiento Funcional Aplicación Personal de Ventas
- RFPA - Requerimiento Funcional Panel de Administración
- RFGPPR - Requerimiento Funcional Gestor, Procesamiento y Proveedor de datos Retail
- POJO - Plain Old Java Object
- MVP - Modelo Vista Presentador
- PA - Panel de Administración

---

## Referencias y bibliografías

---

- [1] MARKETING-XXI.COM (2018), *Otros sistemas de venta — marketing-xxi.com.* [online] marketing-xxi.com. Disponible en: <http://www.marketing-xxi.com/otros-sistemas-de-venta-67.htm>
- [2] NEXT U.(2018)., *7 Beneficios de las ventas online vs. venta tradicional.* [online] nextu.com. Disponible en: <https://www.nextu.com/blog/7-beneficios-de-venta-online-vs-venta-tradicional/>
- [3] NICOLÁS, M. (2017), *Marketing de proximidad: la tecnología Beacon.* [online] oleoshop.com. Disponible en: <https://www.oleoshop.com/blog/tecnologia-beacon>
- [4] ZERINTIA.COM. (2018). , *¿Qué son los Beacons? iBeacons.* [online] zerintia.com. Disponible en: <http://www.zerintia.com/Beacons.html>
- [5] THE VERGE. (2018)., *Macy's begins iBeacon shopping test, will send alerts to your iPhone when you enter stores.* [online] theverge.com. Disponible en: <https://www.theverge.com/2013/11/21/5129336/macys-apple-ibeacon-support-herald-union-square-stores-shopkick>
- [6] RETAILDIVE.COM. (2018). , *How Macy's maximized in-store traffic with beacons during Black Friday — Retail Dive.* [online] retaildive.com. Disponible en: <https://www.retaildive.com/ex/mobilecommercedaily/how-macys-maximized-in-store-traffic-through-beacons-during-black-friday>
- [7] BERETAIL.ES. (2018). , *[“Macy's: Beacons y Gaming para enriquecer la experiencia en el punto de venta — Be-Retail”, Beretail.es, 2018.* [online] beretail.es. Disponible en: <http://beretail.es/blog/2016/03/31/macys-gaming-retail-punto-de-venta-marketing-proximidad/>
- [8] NFC WORLD. (2018)., *Carrefour installs Bluetooth beacons in 28 hypermarkets across Romania NFC World.* [online] nfcworld.com. Disponible en: <https://www.nfcworld.com/2015/07/23/336765/carrefour-installs-bluetooth-beacons-in-28-hypermarkets-across-romania/>
- [9] ONYX BEACON. (2018)., *World premiere: Carrefour Hypermarkets have installed large networks of iBeacons to guide and inform in-store customers.* [online] onyxbeacon.com. Disponible en: <https://www.onyxbeacon.com/world-premiere-carrefour-hypermarkets-have-installed-large-networks-of-ibeacons-to-guide-and-inform-in-store-customers/>
- [10] MCFARLAND, M. AND MCFARLAND, M. (2018)., *American Eagle Outfitters lures customers into fitting rooms with help of beacons.* [online] Washington Post. Disponible en: [https://www.washingtonpost.com/news/innovations/wp/2014/10/15/american-eagle-outfitters-lures-customers-into-fitting-rooms-with-help-of-beacons/?utm\\_term=.49f6db989f88](https://www.washingtonpost.com/news/innovations/wp/2014/10/15/american-eagle-outfitters-lures-customers-into-fitting-rooms-with-help-of-beacons/?utm_term=.49f6db989f88)

- 
- [11] FORBES.COM. (2018)., *Forbes Welcome.* [online] [forbes.com](http://forbes.com). Disponible en: <https://www.forbes.com/sites/jasonbloomberg/2016/03/22/mobile-first-drives-digital-strategy-at-american-eagle-outfitters/#a5d3c74e7d1d>
- [12] MAKEITAPP.COM. (2018)., *App y iBeacon para Marketing de proximidad - Makeitapp.* [online] [makeitapp.com](http://makeitapp.com). Disponible en: <https://www.makeitapp.com/es/ibeacon>
- [13] LÓPEZ G. B., BAUTISTA J.H Y CASTRO E. J. (2015)., *Trabajo Terminal "API para el desarrollo de sistemas de recomendación"* [online]. ESCOM Disponible en: <http://tesis.ipn.mx/bitstream/handle/123456789/21266/Api%20para%20el%20desarrollo%20de%20sistemas%20de%20recomendacion.pdf>
- [14] , *Trabajo Terminal "Sistema generador de recomendaciones para una tienda en-línea de videojuegos".* ESCOM (2010). Disponible en
- [15] GOOGLE DEVELOPERS. (2017), *Eddystone format — Beacons.* [online] [developers.google.com](http://developers.google.com). Disponible en: <https://developers.google.com/beacons/eddystone>
- [16] ESTADÍSTICAS A PROPÓSITO DEL DÍA MUNDIAL DE INTERNET (17 DE MAYO). (2017), 1st ed. [PDF] *Aguascalientes: Instituto Nacional de Estadística y Geografía*, pp.1-8. Disponible en: [http://www.inegi.org.mx/saladeprensa/aproposito/2017/internet2017\\_Nal.pdf](http://www.inegi.org.mx/saladeprensa/aproposito/2017/internet2017_Nal.pdf)
- [17] 7 COMUNICADO DE PRENSA NÚM. 296/17 (2017), 1st ed. [PDF] *Aguascalientes: Instituto Nacional de Estadística y Geografía*, pp.1-8. Disponible en: [http://www.inegi.org.mx/saladeprensa/boletines/2017/emec/emec2017\\_07.pdf](http://www.inegi.org.mx/saladeprensa/boletines/2017/emec/emec2017_07.pdf)
- [18] TECNOLÓGICO DE MONTERREY. (2017), *Ahorro: usa los recursos eficientemente.* [online] [cem.itesm.mx](http://www.cem.itesm.mx/ola/ahorro/index.html). Disponible en: <http://www.cem.itesm.mx/ola/ahorro/index.html>
- [19] NG, A. (2017), *Machine Learning.* [online] [coursera.com](https://www.coursera.org/learn/machine-learning/). Disponible en: <https://www.coursera.org/learn/machine-learning/>
- [20] EREMENKO, K. (2017), *Deep Learning A-Z™: Hands-On Artificial Neural Networks.* [online] [udemy.com](https://www.udemy.com/deeplearning/learn/v4/overview). Disponible en: <https://www.udemy.com/deeplearning/learn/v4/overview>
- [21] MITCHELL, T. (1997), *Machine Learning.* 1st ed. McGraw-Hill Science/Engineering/Math.
- [22] ZACHMAN, JOHN A.(1999) , *El modelado de las empresas: La arquitectura de Zachman.* Zachman Institute for Framework Advancement. Estados Unidos.
- [23] Y. ABU-MOSTAFA, M. MAGDON-ISMAIL AND H. LIN, *Learning from data a short course*, 1st ed. 2012, pp. 3-4.
- [24] *A Tutorial on Clustering Algorithms* [online] Disponible en: [https://home.deib.polimi.it/matteucc/Clustering/tutorial\\_html/kmeans.html](https://home.deib.polimi.it/matteucc/Clustering/tutorial_html/kmeans.html)
- [25] TOWERS, S. (2018). *K-means clustering — Polymatheia.* [online] Disponible en: <http://sherrytowers.com/2013/10/24/k-means-clustering/>
- [26] BROWNLEE, J. (2018) *Supervised and Unsupervised Machine Learning Algorithms - Machine Learning Mastery.* Machine Learning Mastery. Consultado el 13 Marzo 2018, Disponible en <https://machinelearningmastery.com/supervised-and-unsupervised-machine-learning-algorithms/>
- [27] V. GONZALEZ, *A Brief History of Machine Learning — Synergic Partners,* Synergicpartners.com, 2018. [Online] Disponible en: <http://www.synergicpartners.com/en/espanol-una-breve-historia-del-machine-learning/>.
- [28] *Tema 5 Aprendizaje automático*, UNIVERSIDAD DE ALMERIA, ESPAÑA. José del Sagrado, 2017, p 4.
- [29] *A BASIC INTRODUCTION To NEURAL NETWORKS*, Pages.cs.wisc.edu, 2018. [online] Disponible en: <http://pages.cs.wisc.edu/~bolo/shipyard/neural/local.html>

- 
- [30] UNDERSTANDING ACTIVATION FUNCTIONS IN NEURAL NETWORKS, *Medium*, 2018. [online]. Disponible en: <https://medium.com/the-theory-of-everything/understanding-activation-functions-in-neural-networks-9491262884e0>
- [31] F. ISINKAYE, Y. FOLAJIMI AND B. OJOKOH, *Recommendation systems: Principles, methods and evaluation"*, 2018. [online]. Disponible en: <https://www.sciencedirect.com/science/article/pii/S1110866515000341>
- [32] ROUSE, M. (2017)., *Internet de todo (IoE)*. [online] TechTarget SearchDataCenter en Español. Disponible en: <http://searchdatacenter.techtarget.com/es/definicion/Internet-de-todo-IoE> [Consultado el 20 Mar. 2018].
- [33] BANAFA, A. (2016)., *El Internet de Todo*. [online] OpenMind. Disponible en: <https://www.bbvaopenmind.com/el-internet-de-todo/> [Consultado el 20 Mar. 2018].
- [34] RODRÍGUEZ, F. (s. f.). , INFOTEC - Sistemas embebidos Smart: La era del Internet de las Cosas. [online] Infotec.mx. Disponible en: [https://www.infotec.mx/es\\_mx/infotec/sistemas\\_embebidos\\_smart\\_la-era\\_del\\_internet\\_de\\_las\\_cosas](https://www.infotec.mx/es_mx/infotec/sistemas_embebidos_smart_la-era_del_internet_de_las_cosas) [Consultado el 31 Mar. 2018].
- [35] Es.WIKIPEDIA.ORG. (s. f.). , *Computación ubicua*. [online] Disponible en: [https://es.wikipedia.org/wiki/Computaci%C3%B3n\\_ubicua](https://es.wikipedia.org/wiki/Computaci%C3%B3n_ubicua) [Consultado el 31 Mar. 2018].
- [36] SG.COM.MX. (2012). , *C%C3%B3mputo UbiCuo — SG Buzz*. [online] Disponible en: <https://sg.com.mx/content/view/1048> [Consultado el 31 Mar. 2018].
- [37] BANAFA, A. (2017)., *8 tendencias de Internet de las cosas para 2018*. [online] OpenMind. Disponible en: <https://www.bbvaopenmind.com/8-tendencias-de-internet-de-las-cosas-para-2018/> [Consultado el 20 Mar. 2018].
- [38] BANAFA, A. (2017)., *7 tendencias del Internet de las Cosas en 2017*. [online] Fumec.org. Disponible en: [http://fumec.org/v6/index.php?option=com\\_content&view=article&id=751:tendencias-iot-2017&catid=98&Itemid=442&lang=en](http://fumec.org/v6/index.php?option=com_content&view=article&id=751:tendencias-iot-2017&catid=98&Itemid=442&lang=en) [Consultado el 20 Mar. 2018].
- [39] ESPAÑA, I. (2017)., *Ventajas y desventajas del Internet de Las Cosas IoT*. [online] IoT España. Disponible en: <http://www.iot-espana.com/tecnologia/ventajas-y-desventajas-del-internet-de-las-cosas-iot/> [Consultado el 20 Mar. 2018].
- [40] D'MONTE, L. (2014)., *Internet of Everything gains momentum*. [online] <http://www.livemint.com/>. Disponible en: <http://www.livemint.com/Specials/34DC3bDLSCIItBaTfRvMBQO/Internet-of-Everything-gains-momentum.html> [Consultado el 20 Mar. 2018].
- [41] NAVARRO, T. (2016)., *Posicionamiento en interiores (indoor positioning)*. [online] Informática++. Disponible en: <http://informatica.blogs.uoc.edu/2016/04/21/posicionamiento-en-interiores-indoor-positioning/> [Consultado el 20 Mar. 2018].
- [42] CURRAN, K., FUREY E, LUNNEY T., SANTOS J., WOODS D. AND CAUGHEY A. (2011)., *An Evaluation of Indoor Location Determination Technologies*. *Journal of Location Based Services* Vol. 5, No. 2, pp: 61-78, June 2011, ISSN 1748-9725
- [43] ITBUSINESS-SOLUTIONS.COM.MX. (2017)., *¿Qué son los sistemas de posicionamiento en interiores?*. [online] Disponible en: <http://www.itbusiness-solutions.com.mx/que-son-los-sistemas-de-posicionamiento-en-interiores> [Consultado el 20 Mar. 2018].
- [44] MONTIEL PÉREZ, J., HERNÁNDEZ RUBIO, E. AND LÓPEZ BONILLA, J. (2012)., *Computación móvil*. [online] scielo.conicyt.cl. [online] Disponible en: [https://scielo.conicyt.cl/scielo.php?script=sci\\_arttext&pid=S0718-33052012000300001](https://scielo.conicyt.cl/scielo.php?script=sci_arttext&pid=S0718-33052012000300001) [Consultado el 31 Mar. 2018].
- [45] LOPEZ JIMENEZ, X. (2016)., *C%C3%B3mputo m%C3%B3vil*. [online] Disponible en: <https://prezi.com/rdlxf2-1z2kw/computo-movil/> [Consultado el 31 Mar. 2018].

- 
- [46] NOSRATI, M., KARIMI, R., HASANVAND H. A. (2012) , *Mobile Computing: Principles, Devices and Operating Systems.* (2012). 2nd ed. [ebook] Dadaab, Kenia: WAP Journal., pp.399 - 408. Disponible en: <https://pdfs.semanticscholar.org/4f57/0f358c1068a67021f0e93486fbf816c89530.pdf> [Consultado el 15 Abr. 2018].
- [47] WIKIMEDIA FOUNDATION, (s. f.) , *Personal digital assistant.* [online] Disponible en: [https://en.wikipedia.org/wiki/Personal\\_digital\\_assistant](https://en.wikipedia.org/wiki/Personal_digital_assistant). [Consultado el 15 Abr. 2018].
- [48] McCRAKEN, H. (2012)., *Newton, Reconsidered, Time Magazine, TIME.com.* [online] Disponible en: <http://techland.time.com/2012/06/01/newton-reconsidered/> [Consultado el 15 Abr. 2018].
- [49] MOSCARITOLO, A. (2012)., *Survey: 31 Percent of U.S. Internet Users Own Tablets.* [online] PCMag. Disponible en: <https://www.pcmag.com/article2/0,2817,2405972,00.asp> [Consultado el 15 Abr. 2018].
- [50] WIKIMEDIA FOUNDATION, (s. f.), *PC Ultra Móvil.* [online] Disponible en: [https://es.wikipedia.org/wiki/PC\\_Ultra\\_Móvil](https://es.wikipedia.org/wiki/PC_Ultra_Móvil) [Consultado el 15 Abr. 2018].
- [51] GESTIÓN, R. (2016)., *Teléfonos inteligentes: Esta es la historia y la evolución de los smartphones.* [online] Gestión. Disponible en: <https://gestion.pe/tendencias/telefonos-inteligentes-historia-evolucion-smartphones-147278?foto=1> [Consultado el 31 Mar. 2018].
- [52] WIKIMEDIA FOUNDATION, (s. f.), *Sistema operativo móvil.* [online] Disponible en: [https://es.wikipedia.org/wiki/Sistema\\_operativo\\_móvil](https://es.wikipedia.org/wiki/Sistema_operativo_móvil) [Consultado el 15 Abr. 2018].
- [53] WIKIMEDIA FOUNDATION, (s. f.), *Computadora corporal.* [online] Disponible en: [https://es.wikipedia.org/wiki/Computadora\\_corporal](https://es.wikipedia.org/wiki/Computadora_corporal) [Consultado el 15 Abr. 2018].
- [54] WIKIMEDIA FOUNDATION, (s. f.), *Windows Phone.* [online] Disponible en: <https://es.wikipedia.org/wiki/Windows.Phone> [Consultado el 15 Abr. 2018].
- [55] WIKIMEDIA FOUNDATION, (s. f.), *Windows 10 Mobile.* [online] Disponible en: [https://es.wikipedia.org/wiki/Windows\\_10\\_Mobile](https://es.wikipedia.org/wiki/Windows_10_Mobile) [Consultado el 15 Abr. 2018].
- [56] CULTURACIÓN, (s. f.), *Blackberry OS; sistema operativo móvil de RIM - Culturación.* [online] Disponible en: <http://culturacion.com/blackberry-sistema-operativo-movil-de-rim/> [Consultado el 15 Abr. 2018].
- [57] IDC: THE PREMIER GLOBAL MARKET INTELLIGENCE COMPANY. (2017)., *IDC: Smartphone OS Market Share.* [online] Disponible en: <https://www.idc.com/promo/smartphone-market-share/os> [Consultado el 15 Abr. 2018].
- [58] MEJÍA LLANO, J. (2017)., *Guía de marketing de proximidad: qué es, importancia, estrategias y claves de éxito + Infografía — Blog Marketing Digital, Social Media y Transformación Digital* [online] Disponible en: <http://www.juancmejia.com/marketing-digital/guia-de-marketing-de-proximidad-que-es-importancia-estrategias-y-claves-de-exito-infografia/> [Consultado el 18 Abr. 2018].
- [59] ALTIRIA. (s. f.), *Marketing NFC – Altiria.* [online] Disponible en: <https://www.altiria.com/marketing-nfc/> [Consultado el 20 Abr. 2018].
- [60] CHAI, J. (2014)., *¿Va revolucionar (por fin) el NFC el marketing de proximidad? - The Strategic Lab.* [online] The Strategic Lab. Disponible en: <http://thestrategiclab.com/va-revolucionar-por-fin-el-nfc-el-marketing-de-proximidad/> [Consultado el 20 Abr. 2018].
- [61] MARTÍNEZ, D. (2016). , *PostGreSQL vs. MySQL - GeekWare.* [online] GeekWare. Disponible en: <https://danielpecos.com/documents/postgresql-vs-mysql/#AEN115> [Consultado el 26 Abr. 2018].
- [62] LIBERTY CENTER ONE. (s.f.), *PostGRESQL vs MySQL vs SQL Server vs Oracle: Which DBMS Is The Best Choice For You? — Liberty Center One.* Disponible en: <https://www.libertycenterone.com/blog/postgresql-vs-mysql-vs-sql-server-vs-oracle/> [Consultado el 26 Abr. 2018].

- 
- [63] THE APP SOLUTIONS. (s.f.)., *Android vs iOS Development - What and Why is Better?*. [online] Disponible en: <https://theappsolutions.com/blog/development/ios-vs-android/> [Consultado el 26 Abr. 2018].
- [64] PHONE HOUSE (2017)., *iOS vs. Android: ventajas y diferencias de cada sistema operativo — Blog Oficial Phone House.* [online] Disponible en: <http://blog.phonehouse.es/2017/08/24/ios-vs-android-ventajas-y-diferencias-de-cada-sistema-operativo/> [Consultado el 26 Abr. 2018].
- [65] WIKIMEDIA FOUNDATION (s.f.), *IOS*. [online] Disponible en: <https://es.wikipedia.org/wiki/IOS> [Consultado el 26 Abr. 2018].
- [66] WIKIMEDIA FOUNDATION (s.f.), *Android*. [online] Disponible en: <https://es.wikipedia.org/wiki/Android> [Consultado el 26 Abr. 2018].
- [67] THE APPMAKER - CENTRO DE AYUDA. (2017)., *Cuanto tarda el proceso de aprobación y publicación de mi app?*. [online] Disponible en: <https://theappmaker8.zendesk.com/hc/es/articles/218462937-Cuanto-tarda-el-proceso-de-aprobación-y-publicación-de-mi-app-> [Consultado el 26 Abr. 2018].
- [68] HORNOSTAIEV, M. (2017)., *Java, Python, and PHP: Which is Better for Server Backends?*. [online] Erminesoft. Disponible en: <https://erminesoft.com/java-python-and-php-which-is-better-for-server-backends/> [Consultado el 26 Abr. 2018].
- [69] MAHAPATRA, S. (2016), *In a corporate company, should I choose a project based on C# and ASP.NET or Python using Django Framework?*. [online] Quora Disponible en: <https://www.quora.com/In-a-corporate-company-should-I-choose-a-project-based-on-C-and-ASP-NET-or-Python-using-Django-Framework> [Consultado el 26 Abr. 2018].
- [70] ZIEGLER, J. (2014), *Choosing the Right Programming Language for your Startup*. [online] AWS Amazon. Disponible en: <https://aws.amazon.com/es/blogs/startups/choosing-the-right-programming-language-for-your-startup/> [Consultado el 26 Abr. 2018].
- [71] WODEHOUSE, C. (s.f.), *Inside The Java Platform*. [online] UpWork. Disponible en: <https://www.upwork.com/hiring/development/the-java-platform/> [Consultado el 26 Abr. 2018].
- [72] ESTIMOTE.COM. (2018)., *Estimote*. [online] Disponible en: <https://estimote.com/products/>
- [73] BLOG.BEACONSTAC.COM. (2018)., *iBeacon vs Eddystone: Which one works better for your Pilot Project?*. [online] Disponible en: <https://blog.beaconstac.com/2016/01/ibeacon-vs-eddystone/>
- [74] NOTIBEACON. (2018)., *¿Cuál es la diferencia entre iBeacon y Eddystone Beacons?*. [online] Disponible en: <https://blog.notibeacon.com/2016/03/01/cual-es-la-diferencia-entre-ibeacon-y-eddystone-beacons/>
- [75] SUPPORT CENTER. (2018)., *iBeacon Parameters: UUID, Major and Minor*. [online] Disponible en: <https://support.kontakt.io/hc/en-gb/articles/201620741-iBeacon-Parameters-UUID-Major-and-Minor>
- [76] AVVEL INTERNATIONAL. (2018)., *Eddystone UID*. [online] Disponible en: <https://www.avvel.co.uk/edystone-uid/>
- [77] LANE, B. AND LANE, B. (2018)., *How to use Eddystone-URL and what is meant by “no need for an app”*. [online]. PassKit Blog. Disponible en: <https://blog.passkit.com/how-to-use-eddystone-url-and-what-is-meant-by-no-need-for-an-app/>
- [78] LANE, B. AND LANE, B. (2018)., *How to use Eddystone-URL and what is meant by “no need for an app”*. [online]. PassKit Blog. Disponible en: <https://blog.passkit.com/how-to-use-eddystone-url-and-what-is-meant-by-no-need-for-an-app/>
- [79] INTRO TO EDDYSTONE (2018)., *Developer.estimote.com.* (2018). [online]. Disponible en: <https://developer.estimote.com/edystone/>
- [80] ESTIMOTE. (2018)., *Developer.estimote.com.* (2018). [online] Disponible en: <https://developer.estimote.com/>

- 
- [81] THE BALANCE. (2018)., *How Do You Define Marketing in Business?*. [online] Disponible en: <https://www.thebalance.com/marketing-in-business-2948349>
- [82] MARTECH. (2018)., *Proximity Marketing: What is it? What types?*. [online] Disponible en: <https://martech.zone/proximity-marketing/>
- [83] JUANCMEJIA.COM. (2018)., *Guía de marketing de proximidad: qué es, importancia, estrategias y claves de éxito + Infografía — Blog Marketing Digital, Social Media y Transformación Digital — Juan Carlos Mejía Llano.* [online] Disponible en: [http://www.juancmejia.com/marketing-digital/guia-de-marketing-de-proximidad-que-es-importancia-estrategias-y-claves-de-exito-infografia/1\\_Que\\_es\\_el\\_marketing\\_de\\_proximidad](http://www.juancmejia.com/marketing-digital/guia-de-marketing-de-proximidad-que-es-importancia-estrategias-y-claves-de-exito-infografia/1_Que_es_el_marketing_de_proximidad)
- [84] ELFINANCIERO.COM.MX. (2018)., *En México el 85 % de los clientes móviles usan smartphones.* [online] Disponible en: <http://www.elfinanciero.com.mx/empresas/85-de-los-clientes-moviles-en-mexico-usan-smartphones.html>
- [85] RETAIL-INTELLIGENCE.FR. (2018)., *Les avantages du Marketing de proximité — Retail Intelligence.* [online] Disponible en: <http://www.retail-intelligence.fr/2012/02/les-avantages-du-marketing-de-proximite/>
- [86] DOCUMENTATION.MICROFOCUS.COM,2018, *Understanding Synchronous and Asynchronous Communication.* [online] Disponible en: <http://documentation.microfocus.com/help/index.jsp?topic=%2Fcom.microfocus.silkperformer.doc%2FGUID-6CC17B5B-71B7-4703-B9E6-C81835A5335A.html>
- [87] GESTIOPOLIS.COM, E.(2018)., *, ¿Qué es el estudio de factibilidad en un proyecto?*. [online] GestioPolis - Conocimiento en Negocios. Disponible en: <https://www.gestiopolis.com/que-es-el-estudio-de-factibilidad-en-un-proyecto/>
- [88] DX (2018)., *Factibilidad Técnica, Operativa y Económica.* [online] Es.slideshare.net. Disponible en: [https://es.slideshare.net/helodtk1/factibilidad-tecnica-operativa-y-economica-20908957?next\\_slideshow=1](https://es.slideshare.net/helodtk1/factibilidad-tecnica-operativa-y-economica-20908957?next_slideshow=1)
- [89] CONTADORMX.COM. (2018)., *Depreciación Fiscal de Activos Fijos – Porcentajes y Calculadora en Excel – ContadorMx.* [online] Disponible en: <https://contadormx.com/2015/07/29/depreciacion-fiscal-de-activos-fijos-porcentajes-y-calculadora-en-excel/>
- [90] SACMEX.CDMX.GOB.MX. (2018)., *Sacmex.cdmx.gob.mx.* (2018).[online] Disponible en: <http://www.sacmex.cdmx.gob.mx/storage/app/media/tarifas/tarifas172.pdf>
- [91] CONTADORMX.COM. (2018)., *Depreciación Fiscal de Activos Fijos – Porcentajes y Calculadora en Excel – ContadorMx.* [online] Disponible en: <https://contadormx.com/2015/07/29/depreciacion-fiscal-de-activos-fijos-porcentajes-y-calculadora-en-excel/>
- [92] TELMEX. (2018)., *TELMAX.* [online] Disponible en: <http://downloads.telmex.com/pdf/libro-tarifas-telmex.pdf>
- [93] NEUVOO.COM.MX. (2018)., *Salario Analista Sistemas en México.* [online]. Disponible en: <https://neuvoo.com.mx/salario/salario-Analista-Sistemas>
- [94] FERRUFINO, T. V. (2018)., *Análisis y Gestión del Riesgo.* [online] Tamaraferufino.blogspot.mx. Disponible en: <http://tamaraferufino.blogspot.mx/2010/10/analisis-y-gestion-del-riesgo.html>
- [95] ES.SLIDEShare.NET. (2018)., *Análisis de sistemas fases del diseño de sistemas.* [online] Disponible en: <https://es.slideshare.net/profmyriamsanuy/anlisis-de-sistemas-fases-del-diseo-de-sistemas>
- [96] PROMOCION DESCUENTOS (2018)., *Liverpool en linea: Venta Nocturna 10 y 11 de Junio 2016.* [online] Disponible en: <https://promociondescuentos.com/2016/06/liverpool-venta-nocturna-10-y-11-de-junio-2016/>

- 
- [97] AHORRADORAS.COM.(2018)., *Descuento del 20 por ciento en H& M.* [online] Disponible en: <https://www.ahorradoras.com/2016/04/descuento-del-20-en-hm/>
- [98] HM.COM. (2018)., *H & M offers fashion and quality at the best price.* [online] Disponible en: <http://www.hm.com/mx/>
- [99] MEDIUM. (2018)., *Playing with Estimote Beacon – Zoon – Medium.* [online] medium.com. Disponible en: <https://medium.com/@zoonref/playing-with-estimote-beacon-f62f7625f243>
- [100] VERDUGO, D. (2018)., *Detectando beacons en Android, protocolos, usos: Eddystone.* [online] Solidgear.. Disponible en: <https://solidgeargroup.com/beacons-en-android?lang=es>
- [101] USING BEACONS. (2018)., *Ventajas y desventajas de la Tecnología Beacon.* [online] Disponible en: <http://www.usingbeacons.com/ventajas-y-desventajas-de-la-tecnologia-beacon/>
- [102] MACIAS, J. (2018)., *Bluetooth BLE: el conocido desconocido.* [online] Solidgear. Disponible en: <https://solidgeargroup.com/bluetooth-ble-el-conocido-desconocido?lang=es>
- [103] PUNCHTHROUGH.COM. (2018)., *Punch Through.* [online] Disponible en: <https://punchthrough.com/bean/docs/guides/everything-else/how-gap-and-gatt-work/>
- [104] TORNADO WEB SERVER — TORNADO 5.0.2 DOCUMENTATION. (2018), [online] Disponible en: <http://www.tornadoweb.org/en/stable/>.
- [105] FOUNDATION, N. (2018). NODE.JS. NODE.JS., [online] Disponible en: <https://nodejs.org/en/>.
- [106] EXPRESS.JS. (2018). EN.WIKIPEDIA.ORG., [online] Disponible en: <https://en.wikipedia.org/wiki/Express.js>.

# CAPÍTULO 11

---

## Anexos

---

### Encuestas realizadas

¿Sabes que es un beacon?

65 respuestas

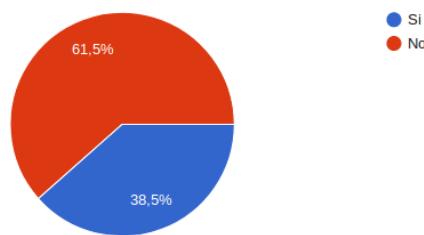


Figura 11.1: Gráfica 1.

¿Conoces los beneficios que los beacons proveen?

65 respuestas

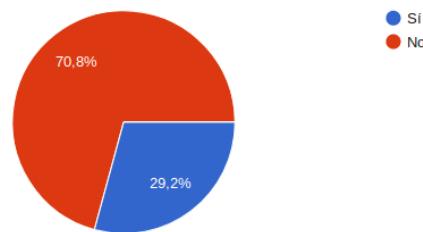


Figura 11.2: Gráfica 2.

¿Compartirías tu historial de compras de una tienda departamental con el objetivo de hacerte recomendaciones sobre otros artículos?

65 respuestas

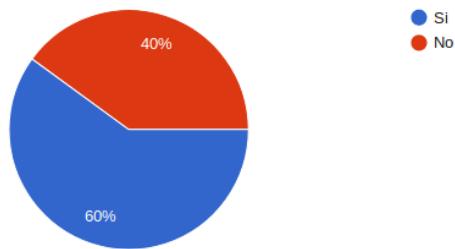


Figura 11.3: Gráfica 3.

¿Compartirías tu información pública de Facebook (páginas a las que has dado like, edad, sexo, fecha de cumpleaños, estado civil), con el fin de realizarte mejores recomendaciones?

65 respuestas

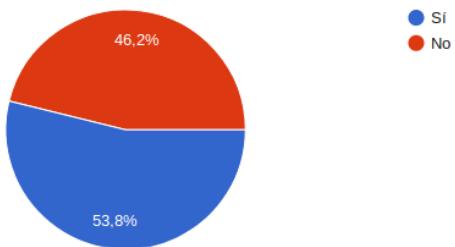


Figura 11.4: Gráfica 4.

¿Permitirías dar a conocer tu ubicación dentro de un establecimiento a los vendedores de dicho lugar para ayudarte en tu proceso de compra o recomendarte algún producto?

65 respuestas

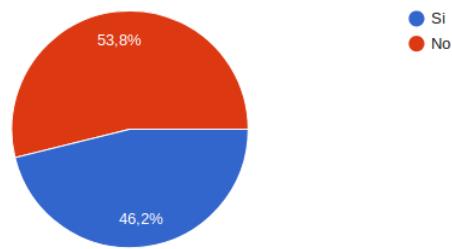


Figura 11.5: Gráfica 5.