



Universidad del Azuay

Facultad de Ciencias de la Administración Escuela de Ingeniería de Sistemas y Telemática

“ASISTENTE DE COMPRAS”

**Trabajo de graduación previo a la obtención del título
de:**

“Ingeniero en Sistemas y Telemática”

Autores:

Javier Alejandro Polo Vásquez

Patricio Xavier López Yépez

Director:

Mgst. Catalina Veronica Astudillo Rodríguez

Cuenca, Ecuador

2016

DEDICATORIA

La elaboración del presente proyecto se la dedicamos a nuestros padres, hermanos, familiares y amigos ya que gracias a su constante apoyo hemos podido culminar esta meta de muchas más por cumplir.

AGRADECIMIENTOS

A la Mgst. Catalina Astudillo y Francisco Salgado Arteaga, PhD, por habernos guiado a la realización del presente trabajo y ayudarnos a conseguir esta meta.

A todos nuestros profesores de la Universidad del Azuay y de igual manera a su personal administrativo, por todo el conocimiento y experiencias compartidas durante los años de estudio.

A WebRatio SRL Ecuador, en especial al Ing. Matteo Silva, Gerente, y a todo el personal por el apoyo recibido durante la ejecución de este proyecto.

INDICE DE CONTENIDOS

DEDICATORIA	ii
AGRADECIMIENTOS	iii
INDICE DE CONTENIDOS	iv
INDICE DE GRÁFICOS.....	ix
INDICE DE TABLAS.....	xii
RESUMEN.....	xiii
ABSTRACT	xiv
INTRODUCCIÓN.....	1
CAPÍTULO 1: INDAGACIÓN EXPLORATORIA	2
Introducción	2
1.1 <i>Scrum</i>	2
1.1.1 La esencia del <i>Scrum</i>	3
1.1.2 Roles en <i>Scrum</i>	3
1.1.2.1 Rol de dueño del producto.....	3
1.1.2.2 Rol de equipo de desarrollo.....	4
1.1.2.3 Rol de facilitador	4
1.1.3 Artefactos del <i>Scrum</i>	5
1.1.3.1 Lista de tareas del producto.....	5
1.1.3.2 Lista de tareas de la iteración	7
1.1.3.3 Gráfico de trabajo pendiente de la iteración	7
1.1.3.4 Lista de impedimentos	7
1.1.4 Reuniones durante la iteración	7
1.1.4.1 Primera parte de la planificación de la iteración	8
1.1.4.2 Segunda parte de la planificación de la iteración	8
1.1.4.3 Reunión diaria.....	9
1.1.4.4 Revisión de la iteración	9
1.1.4.5 Retrospectiva de la iteración.....	10
1.1.4.6 Reunión de estimación	10

1.1.5	Gestión de las entregas del producto	11
1.2	Modelo Vista Controlador	11
1.2.1	Ventajas.....	12
1.2.2	Utilidad de MVC.....	12
1.2.3	Definición de las capas.....	12
1.2.3.1	Modelo	12
1.2.3.2	Vista	13
1.2.3.3	Controlador	13
1.3	IFML.....	14
1.4	WebRatio	20
1.4.1	Conceptos principales de Modelado con WebRatio	20
	Elementos.....	20
1.4.1.1	Aplicaciones web.....	21
1.4.1.1.5		31
1.4.1.2	Aplicaciones móviles	32
1.5	Servicios web REST	35
CAPÍTULO 2: APLICACIÓN WEB CON WEBRATIO.....		36
	Introducción	36
2.1	Análisis de requisitos del sistema.....	36
2.2	Historias de usuario	37
2.3	Desarrollo de las historias de usuario.....	37
2.3.1	Iteración 1.....	37
2.3.1.1	US001 – Modelo de dominio	37
2.3.1.2	US002 – Gestión de Roles de usuario	39
2.3.1.3	US003 – Registro de usuarios.....	41
2.3.2	Iteración 2.....	49
2.3.2.1	US014 – Interfaz del sistema.....	49
2.3.3	Iteración 3.....	52
2.3.3.1	US004 – Gestión de Tiendas.....	52

2.3.3.2 US005 – Gestión de Responsables de tiendas	56
2.3.3.3 US006 – Gestión de categorías de productos.....	57
2.3.3.4 US007 – Gestión de productos.....	60
2.3.4 Iteración 4.....	62
2.3.4.1 US008 – Búsqueda de productos.....	62
2.3.4.2 US0011 – Productos y tiendas favoritas.....	64
2.3.5 Iteración 5.....	65
2.3.5.1 US0013 – Reseñas.....	65
2.3.5.2 US009 – Carrito de compras	67
2.3.6 Iteración 6.....	68
2.3.6.1 US010 – Orden de compra.....	68
CAPÍTULO 3: APLICACIÓN MÓVIL CON WEBRATIO.....	70
Introducción	70
3.1 Análisis de requisitos del sistema.....	70
3.2 Historias de usuario	70
3.3 Desarrollo de las historias de usuario.....	71
3.3.1 Iteración 1	71
3.3.1.1 US001 – Modelo de dominio	71
3.3.1.2 US003 – Registro de usuarios.....	74
3.3.2 Iteración 2	76
3.3.2.1 US014 – Interfaz del sistema.....	76
3.3.3 Iteración 3	78
3.3.3.1 US008 – Búsqueda de productos.....	78
3.3.3.2 US011 – Productos y tiendas favoritas.....	81
3.3.3.3 US013 – Reseñas.....	83
3.3.4 Iteración 4	85
3.3.4.1 US012 – Notificaciones automáticas por ofertas en las tiendas.....	85
3.3.5 Iteración 5	87
3.3.5.1 US009 – Carrito de compras	87

3.3.6 Iteración 6	89
3.3.6.1 US010 – Orden de compra.....	89
3.3.7 Iteración 7	90
CONCLUSIONES	91
BIBLIOGRAFÍA	94
ANEXOS	96
Anexo 1	96
Terminología.....	96
Termino propio	96
Termino original.....	96
Anexo 2	100
Análisis de requisitos.....	100
2.1 Gestión de tiendas.....	100
2.2 Gestión de responsables de tiendas	100
2.3 Gestión de productos	101
2.4 Gestión de categorías de productos.....	101
2.5 Búsqueda de productos.....	102
2.6 Registro de usuarios.....	102
2.7 Roles de usuario.....	103
2.7.1 Administrador del sistema	103
2.7.2 Responsable de tienda	103
2.7.3 Despachador	104
2.7.4 Cliente.....	105
2.8 Órdenes de compra.....	105
2.9 Carrito de compras	106
2.10 Despachadores	106
2.11 Productos y tiendas favoritas	106
2.12 Facturación electrónica	107
2.13 Envío de notificaciones.....	107

2.14 Reseñas	107
Anexo 3	108
Definición de las historias de usuario	108
3.1 US000 – Configuración de ambiente de desarrollo	108
3.2 US001 – Modelo de dominio	108
3.3 US002 – Gestión de Roles de usuario	109
3.4 US003 – Registro de usuarios.....	110
3.5 US004 – Gestión de Tiendas	111
3.6 US005 – Gestión de responsables de tiendas	111
3.7 US006 – Gestión de categorías de productos.....	112
3.8 US007 – Gestión de productos	113
3.9 US008 – Búsqueda de productos.....	114
3.10 US009 – Carrito de compras	115
3.11 US010 – Orden de compra.....	115
3.12 US011 – Productos y tiendas favoritas	116
3.13 US012 – Notificaciones automáticas por ofertas en las tiendas.....	117
3.14 US013 – Reseñas	118
3.15 US014 – Interfaz del sistema	119
Anexo 4	120
4.1 Lista de tareas del Producto	120
4.2 Iteración 1	127
4.3 Iteración 2	130
4.4 Iteración 3	132
4.5 Iteración 4	135
4.6 Iteración 5	137
4.7 Iteración 6	139
4.8 Iteración 7	141

INDICE DE GRÁFICOS

Gráfico 1 – Visión general del <i>Scrum</i>	3
Gráfico 2 – Diagrama Modelo Vista Controlador (MVC)	11
Gráfico 3 – Ejemplo de entidad	21
Gráfico 4 – Ejemplo de entidad completa	23
Gráfico 5 – Ejemplo de generalización	24
Gráfico 6 – Ejemplo de relación 1 a muchos (1 a N)	25
Gráfico 7 – Ejemplo de Vista del sitio	25
Gráfico 8 – Modelo de dominio aplicación web	38
Gráfico 9 – Modelado de Gestión de roles de usuario	39
Gráfico 10 – Interfaz Roles de usuario	39
Gráfico 11 – Interfaz Detalle rol de usuario	40
Gráfico 12 – Interfaz Crear/Editar rol de usuario	41
Gráfico 13 – Modelado Registro de usuarios	42
Gráfico 14 – Interfaz registro de usuarios	42
Gráfico 15 – Formato de correo de confirmación de la cuenta	43
Gráfico 16 – Modelado Recuperar cuenta	43
Gráfico 17 – Interfaz Recuperar cuenta	43
Gráfico 18 – Modelado Gestión de usuarios	44
Gráfico 19 – Interfaz Usuarios	45
Gráfico 20 – Interfaz Detalle de usuario	46
Gráfico 21 – Interfaz Crear/Editar usuario	46
Gráfico 22 – Modelado Perfil de usuario	47
Gráfico 23 – Interfaz Perfil	48
Gráfico 24 – Interfaz Actualizar datos	48
Gráfico 25 – Interfaz Cambiar contraseña	49
Gráfico 26 – Logo	49
Gráfico 27 – Color principal aplicación web	50
Gráfico 28 – Color fuente principal	50
Gráfico 29 – Color principal de botones	51
Gráfico 30 – Interfaz Inicio de sesión	51
Gráfico 31 – Plantilla de página principal	52
Gráfico 32 – Interfaz pantalla responsiva	52
Gráfico 33 – Modelado Gestión de tiendas	52
Gráfico 34 – Interfaz tiendas	53
Gráfico 35 – Interfaz Detalles de la tienda	54
Gráfico 36 – Interfaz Crear/Editar usuario	55

Gráfico 37 – Interfaz Crear/Editar usuario	56
Gráfico 38 – Asignar responsable a tienda	57
Gráfico 39 – Modelado Gestión de categorías de productos	57
Gráfico 40 – Interfaz categoría de productos	58
Gráfico 41 – Interfaz Detalles de la categoría de productos	59
Gráfico 42 – Interfaz Crear/Editar categoría de productos.....	59
Gráfico 43 – Modelado Gestión de productos	60
Gráfico 44 – Interfaz productos	60
Gráfico 45 – Interfaz detalles del producto.....	61
Gráfico 46 – Interfaz Crear/Editar producto.....	62
Gráfico 47 – Modelado búsqueda de productos.....	63
Gráfico 48 – Interfaz búsqueda de productos	63
Gráfico 49 – Modelado productos y tiendas favoritas	64
Gráfico 50 – Ejemplo de producto marcado como favorito	65
Gráfico 51 – Ejemplo de producto sin marcar como favorito.....	65
Gráfico 52 – Modelado reseñas	66
Gráfico 53 – Ejemplo de crear reseña.....	66
Gráfico 54 – Modelado carrito de compras	67
Gráfico 55 – Interfaz carrito de compras	67
Gráfico 56 – Modelado mis órdenes de compra.....	68
Gráfico 57 – Interfaz mis órdenes de compra	69
Gráfico 58 – Modelo Entidad – Relación DataService	72
Gráfico 59 – Modelo Entidad – Relación Aplicación Móvil	73
Gráfico 60 – IFML Móvil Pantalla Iniciar sesión	74
Gráfico 61 – Interfaz inicio de sesión	75
Gráfico 62 – Interfaz nuevo cliente.....	75
Gráfico 63 – Interfaz recuperar contraseña.....	76
Gráfico 64 – Logo	76
Gráfico 65 – Color principal aplicación móvil.....	77
Gráfico 66 – Color fuente principal	77
Gráfico 67 – Color principal de botones	77
Gráfico 68 – Menú lateral oculto & Menú Lateral desplegado.....	78
Gráfico 69 – Modelo IFML Sección Productos.....	79
Gráfico 70 – Interfaz búsqueda de productos	80
Gráfico 71 – Interfaz detalle de producto	80
Gráfico 72 – Modelo IFML Productos favoritos	81
Gráfico 73 – Interfaz productos favoritos.....	81

Gráfico 74 – Modelo IFML tiendas favoritas.....	82
Gráfico 75 – Interfaz tiendas favoritas.....	82
Gráfico 76 – Interfaz detalle de tienda	83
Gráfico 77 – Interfaz Localización de tienda	83
Gráfico 78 – Modelo IFML Productos – Reseñas.....	84
Gráfico 79 – Interfaz listado – ingreso de reseñas.....	85
Gráfico 80 – Modelo IFML Servicio Web Notificaciones	86
Gráfico 81 – Notificación Nuevas Ofertas	86
Gráfico 82 – Notificación órdenes de compra	87
Gráfico 83 – Modelo IFML Carrito de Compras.....	88
Gráfico 84 – Interfaz Carrito de Compras	89
Gráfico 85 – Modelo IFML Mis órdenes de compra	89
Gráfico 86 – Interfaz Mis órdenes de compra – Detalle de Orden de Compra	90
Gráfico 87 – Caso de uso usuario administrador	103
Gráfico 88 – Caso de uso usuario responsable de tienda.....	104
Gráfico 89 – Caso de uso usuario despachador	104
Gráfico 90 – Caso de uso cliente	105
Gráfico 91 - Gráfico de trabajo pendiente de la iteración 1	129
Gráfico 92 – Gráfico de trabajo pendiente de la iteración 2	131
Gráfico 93 – Gráfico de trabajo pendiente de la iteración 3	134
Gráfico 94 – Gráfico de trabajo pendiente de la iteración 4	136
Gráfico 95 – Gráfico de trabajo pendiente de la iteración 5	138
Gráfico 96 – Gráfico de trabajo pendiente de la iteración 6	140
Gráfico 97 – Gráfico de trabajo pendiente de la iteración 7	142

INDICE DE TABLAS

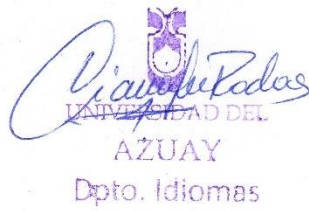
Tabla 1 – Conceptos esenciales de IFML	19
Tabla 2 - US000 – Configuración de ambiente de desarrollo.....	108
Tabla 3 - US001 – Modelo de dominio	109
Tabla 4 - US002 – Gestión de Roles de usuario	110
Tabla 5 - US003 – Registro de usuarios	111
Tabla 6 - US004 – Gestión de Tiendas	111
Tabla 7 - US005 – Gestión de responsables de tiendas	112
Tabla 8 - US006 – Gestión de categorías de productos	113
Tabla 9 - US007 – Gestión de productos	114
Tabla 10 - US008 – Búsqueda de productos	114
Tabla 11 - US009 – Carrito de compras.....	115
Tabla 12 - US010 – Orden de compra	116
Tabla 13 - US011 – Productos y tiendas favoritas	117
Tabla 14 - US012 – Notificaciones automáticas por ofertas en las tiendas	118
Tabla 15 - US013 – Reseñas	119
Tabla 16 - US014 – Interfaz del sistema	119
Tabla 17 - Lista de tareas del producto	126
Tabla 18 - Iteración 1	128
Tabla 19 - Iteración 2	130
Tabla 20 - Iteración 3	133
Tabla 21 - Iteración 4	135
Tabla 22 - Iteración 5	137
Tabla 23 - Iteración 6	139
Tabla 24 - Iteración 7	141

RESUMEN

La investigación y el sistema desarrollados buscan dar solución a un problema común: realizar compras en supermercados, tiendas o almacenes. Con este propósito se desarrolló un asistente de compras que permite emitir órdenes de compra en cualquier local comercial afiliado al sistema que gestiona nuestras aplicaciones. El asistente comprende desde la exploración de tiendas y productos hasta el despacho de las órdenes de compra. Las aplicaciones tanto móviles (Android – iOS) como web se desarrollaron utilizando la tecnología WebRatio y el marco de desarrollo ágil *Scrum* que nos permiten cumplir los proyectos definidos en tiempos oportunos.

ABSTRACT

The research and system have been developed to provide a solution to a common problem: conduct purchases in supermarkets, stores or grocery stores. With this goal, a shopping system was developed to facilitate purchase orders in any commercial outlet affiliated with the system that manages our applications. The assistant covers the exploration of stores and products to the purchase order dispatch. The mobile (Android – iOS) and web applications were developed using WebRatio technology and the *Scrum* agile software development framework which allows for the timely completion of defined projects.



Translated by:


Melita Vega

March 10, 2016

INTRODUCCIÓN

El presente documento presenta una solución a uno de los problemas del quehacer diario, realizar compras. Debido a esto planteamos crear una aplicación para dispositivos móviles y una aplicación web que permita visualizar un catálogo de productos ordenados por categorías y mantener un registro de los productos comprados en las diferentes tiendas o supermercados.

El proyecto se realiza en tres etapas definidas. La primera es una etapa de análisis que permite recopilar toda la información necesaria para determinar los requerimientos representados en casos de uso, con lo cual podremos desarrollar las aplicaciones propuestas, dotándolas de todas las características necesarias para que pueda ser implementada.

La segunda parte comprende el desarrollo de los aplicativos utilizando el marco de desarrollo SCRUM, que permite producir en tiempos cortos y en diferentes etapas acumulativas, en lapsos de tiempo de una semana, hasta finalizar el desarrollo de las aplicaciones.

La tercera y última parte comprende el desarrollo progresivo, conforme avance el desarrollo de las funcionalidades, esta se refiere a las pruebas del producto, para las cuales contaremos con diferentes dispositivos móviles (Android, iOS). Estas pruebas nos permitirán comprobar la correcta funcionalidad del producto.

CAPÍTULO 1: INDAGACIÓN EXPLORATORIA

Introducción

En este capítulo se presentan los conceptos fundamentales que se utilizarán en el desarrollo del proyecto. Los términos que aquí se describen, se expresan en lengua castellana, de acuerdo con la interpretación que los autores dimos a los conceptos en inglés que se encuentran divulgados sobre el tema. El anexo 1 muestra la relación entre los términos castellanos con los correspondientes del inglés.

1.1 *Scrum*

Según Rubin (2013) *Scrum* es un marco de trabajo que nos permite desarrollar productos y servicios innovadores. *Scrum* es una “estrategia de desarrollo incremental a través de iteraciones y revisiones periódicas” (Palacio, 2014).

Ya que el desarrollo de software implica acciones tales como aprender nuevas tecnologías e innovar en los procesos, *Scrum* nos permite aplicar buenas prácticas de desarrollo cuando de trabajo colaborativo se trata, enfatizando el desarrollo del producto en tiempos cortos.

Se toma especial atención tanto en el producto resultante como en la eficacia de las prácticas utilizadas para su desarrollo. Conforme se avanza en el desarrollo y se cumplen las metas del producto, las prácticas se van mejorando; al final se repite el proceso hasta terminar el producto. (Deemer, Benefield, Larman, & Vodde, 2012)

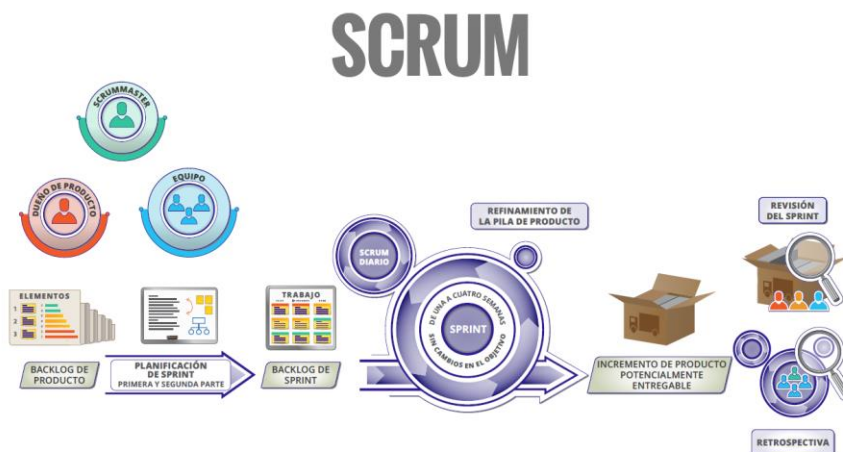


Gráfico 1 – Visión general del *Scrum*.

Fuente: (Deemer, Benefield, Larman, & Vodde, 2012)

1.1.1 La esencia del *Scrum*

De acuerdo con Hundermark (2009), para que *Scrum* funcione al desarrollar un proyecto, deben cumplirse los supuestos siguientes:

- El equipo tiene los objetivos claros
- Los equipos se organizan en función del trabajo a realizar
- Las entregas se programan con funciones definidas y significativas
- Las personas involucradas en el proyecto tienen retroalimentación
- El trabajo se revisa en equipo, con el compromiso de mejorar
- El trabajo avanza con un progreso real
- Existe comunicación entre todos los miembros del equipo

1.1.2 Roles en *Scrum*

De acuerdo a las afirmaciones de Deemer y colaboradores (2012), a diferencia de un equipo de desarrollo tradicional en una fábrica de software, en *Scrum*, no existe un individuo que tiene el rol de gerente de proyectos, las responsabilidades de este rol se encuentran divididas en tres roles que conforman a un equipo *Scrum*:

1.1.2.1 Rol de dueño del producto

Es quien gestiona el producto y el retorno de la inversión del mismo; además tiene la responsabilidad de:

- Recolectar requerimientos.

- Administrar y priorizar las tareas en la lista de tareas del producto.
- Aceptar las versiones de software al terminar cada iteración.
- Administrar las entregas.

1.1.2.2 Rol de equipo de desarrollo

Es el encargado de construir lo que el cliente indica; el equipo debe tener toda la experiencia y conocimientos necesarios para que, en el lapso de tiempo de una iteración, pueda desarrollar y entregar un producto, además (Hundermark, 2009) nos menciona las siguientes responsabilidades:

- Estimación de cada uno de los ítems de la lista de tareas del producto.
- Seguimiento de su propio progreso.
- Ser auto – organizados, pero siempre tomando en cuenta del compromiso de entrega al dueño del producto.

1.1.2.3 Rol de facilitador

Es quien ayuda a mantener el uso de *Scrum* a lo largo del proceso de desarrollo del producto. El facilitador no es precisamente el jefe de los miembros del equipo en un sentido tradicional, ni el jefe del proyecto, ni el líder, ni el representante del equipo. El énfasis de sus funciones y responsabilidades se concentran en las siguientes tareas:

- Asegurarse de que el entorno de trabajo sea el idóneo para el equipo.
- Eliminar todos los impedimentos que puedan presentarse en el desarrollo del producto
- Hacer todo lo que esté a su alcance para ayudar al equipo, al dueño del producto y a la organización a tener éxito
- Incentivar al uso de prácticas modernas de desarrollo.
- Proteger al equipo de interferencias externas.

“El facilitador es un *coach* y formador” (Deemer, Benefield, Larman, & Vodde, 2012), ya que se debe asegurar de que todos, incluido el cliente y el (los) gerente(s) de la organización comprendan correctamente los principios y buenas prácticas que se pueden adoptar con *Scrum*, esto con el fin de tener éxito con el desarrollo de producto.

Deemer et al. (2012) nos dicen, además, que el facilitador y el cliente no pueden ser la misma persona dentro del desarrollo de un proyecto. Los enfoques que deben tener los dos roles son totalmente diferentes, y combinar los dos roles dentro de un

mismo individuo puede llegar a ocasionar confusión y conflictos, trayendo consigo grandes problemas al desarrollo del producto.

1.1.3 Artefactos del *Scrum*

De acuerdo con las afirmaciones de Hundermark (2009) cuando se realiza un proyecto con *Scrum*, este únicamente define cuatro artefactos sobre los que todo el equipo trabajará a lo largo del desarrollo del proyecto:

- Lista de tareas del producto
- Lista de tareas de la iteración
- Gráfico de trabajo pendiente
- Lista de impedimentos

Sin embargo, el hecho que no se mencionen más artefactos o documentos no quiere decir que cuando se realiza un proyecto con *Scrum*, no se necesite de documentación, la documentación es necesaria siempre, pero con *Scrum* lo que se busca es que únicamente la documentación valiosa sea utilizada en todo el proceso de desarrollo del producto, de esta forma se elimina el tiempo invertido en análisis y revisión de documentación innecesaria.

1.1.3.1 Lista de tareas del producto

Es una lista ordenada y priorizada de funcionalidades vistas desde la perspectiva del cliente. La lista de tareas del producto evoluciona constantemente durante toda la vida del producto; todas las funcionalidades que se encuentran dentro de la lista de tareas del producto deben ser realizadas en algún momento por el equipo de desarrollo, claro está con un orden de prioridad definido por el cliente.

Normalmente dentro del desarrollo de un producto con *Scrum*, a la lista de tareas del producto se lo irá actualizando constantemente, se agregarán ítems, a varios de los ítems existentes se los eliminará debido a que ya no son necesarios, a otros se los dividirá en ítems más pequeños. Tarde o temprano a todos los ítems restantes se los deberá estimar, con el fin de conocer el tiempo que costará desarrollarlos y de esta manera conocer también su ubicación dentro de la lista de tareas del producto.

La lista de tareas del producto no contiene historias de usuario, únicamente contiene elementos que pueden ser transformados ya sea en casos de uso o en historias de

usuario que son útiles para el equipo de desarrollo, pero el enfoque principal de la lista de tareas del producto es proporcionarle valor al cliente el producto.

Según Deemer y colaboradores (2012) una lista de tareas del producto se considera bueno cuando es profundo, además de cumplir con lo siguiente:

- a) Se ha detallado correctamente: los elementos que cuentan con mayor prioridad dentro de la lista de tareas del producto deben estar más detallados que el resto, debido a que se trabajarán y entregarán más pronto, conforme se avanza con el desarrollo del producto el nivel de detalle de los elementos en la lista de tareas del producto se irá ampliando
- b) Todos los elementos que conforman la siguiente versión del producto a entregarse, deben estar correctamente estimados: de ser necesario revisarse y re estimarse con cada iteración; las estimaciones se realizan de acuerdo al esfuerzo que deberá realizarse para cumplir con el desarrollo de cada elemento de la lista de tareas del producto, algunas veces se debe añadir un valor de riesgo en las diferentes tareas, ya sea por la complejidad de la tarea o porque esta depende de varias anteriores para poder ser realizada. Cuando se realiza la estimación, se la proporciona al cliente, de la misma forma el cliente con otros interesados en el proyecto puede proveer información acerca de los elementos a incluirse en la lista de tareas del producto, lo que puede inferir en la estimación.
- c) Debido a que la lista de tareas del producto está en constante refinamiento, para cada una de las Iteraciones, se pueden añadir, eliminar, modificar tanto la prioridad como el detalle de los elementos en la lista de tareas: Este tipo de cambios normalmente se dan cuando el cliente actualiza la lista de tareas de acuerdo a sus necesidades, nueva información provista, mejorar rendimiento, superar obstáculos técnicos, entre otros.
- d) Todos los elementos de la lista de tareas del producto están priorizados: de tal forma que los elementos que cuentan con una mayor prioridad deberán producir el máximo valor a cambio del menor costo.

1.1.3.2 Lista de tareas de la iteración

Según las afirmaciones de Hundermark (2009), el Lista de tareas de la iteración es una lista o tablero de tareas que el equipo de desarrollo se ha comprometido a realizar durante la iteración; la lista de tareas de la iteración ayuda al equipo y a los interesados en el proyecto a conocer el estado real de la planificación y avance actual de las tareas en la iteración.

1.1.3.3 Gráfico de trabajo pendiente de la iteración

Básicamente es una gráfica que ayuda al equipo de desarrollo a monitorizar el progreso de cada uno de los miembros, y que da una clara visión de las posibilidades de alcanzar los compromisos planteados al finalizar una determinada iteración. El gráfico de trabajo pendiente de la iteración se lo actualiza diariamente, de esta manera todos los miembros del equipo podrán visualizar cuantas horas restan para poder concluir con la iteración (Hundermark, 2009).

1.1.3.4 Lista de impedimentos

Comprende toda la lista de impedimentos encontrados por el equipo para el desarrollado de una o varias funcionalidades, que impiden que el desarrollo del producto avance de acuerdo a lo programado. El facilitador debe estar en la capacidad de eliminar o solucionar todos estos impedimentos para que el equipo trabaje conforme lo estimado.

1.1.4 Reuniones durante la iteración

Una iteración es la parte primordial en el ciclo de desarrollo de *Scrum*, dentro de la iteración se realizan reuniones que pueden clasificarse en:

- Reuniones para planificar, primera parte de la planificación de la iteración – segunda parte de la planificación de la iteración.
- Reuniones diarias
- Reuniones de revisión como revisión de la iteración, retrospectiva de la iteración.

Todos los tipos de reuniones antes nombrados, son reuniones que deben realizarse y marcan el inicio y el fin de una iteración, la longitud de las Iteraciones siempre es fija y nunca debe extenderse, generalmente los equipos de desarrollo *Scrum* eligen

Iteraciones con duraciones de dos, tres y máximo cuatro semanas según describe Hundermark (2009).

1.1.4.1 Primera parte de la planificación de la iteración

En esta reunión de planificación, el cliente presenta todas las funcionalidades que desea sean implementadas, el equipo analiza dichas funcionalidades y consulta las que no estén suficientemente claras, esto con la finalidad de poder comprometerse a entregarlas al final de la iteración. El equipo decide que funcionalidades puede entregar considerando el tiempo de duración de la iteración, el número de miembros del equipo, habilidades y destrezas de sus miembros, y los criterios de aceptación para cada una de las funcionalidades.

El cliente debe estar presente durante la realización de esta reunión, su función principal será guiar y responder todas las inquietudes del equipo, el facilitador debe asegurar que todos los interesados en el proyecto cuya presencia sea necesaria concurren, al menos telefónicamente.

Si se encuentran elementos a ser desarrollados que no fueron estimados, pero que deben desarrollarse dentro de la iteración, deben ser descritos inmediatamente e incluidos a la lista de tareas.

Como puntos clave de esta reunión el equipo de desarrollo se compromete ante el cliente a entregar las funcionalidades descritas en la lista de tareas del producto, que consideran realizables al finalizar la iteración.

El tiempo de duración recomendado para esta sesión es proporcional al tiempo en duración de la iteración, por ejemplo, si la iteración es de dos semanas, la reunión de la primera parte de la planificación de la iteración 1 deberá ser de dos horas.

1.1.4.2 Segunda parte de la planificación de la iteración

Durante esta sesión el equipo deberá estar en la capacidad de crear un diseño a alto nivel de todos los elementos que se ha comprometido a desarrollar en la iteración; los resultados de esta reunión de planificación será el documento de la lista de tareas de la iteración, es decir, la lista de tareas que el equipo deberá realizar en la iteración.

Durante esta reunión de planificación el equipo podrá tener preguntas adicionales de aclaración con respecto a los requerimientos de las tareas a realizar, el facilitador debe asegurarse que el cliente y los interesados en el proyecto se encuentren disponibles para responder cualquier inquietud relacionada con los requerimientos.

Debido a que esta reunión tiene un límite de tiempo pre–establecido (igual al tiempo tomado en la primera parte de la planificación de la iteración) no siempre se podrá concluir con un diseño completo de la Lista de tareas de la iteración, por ende, en el desarrollo de la iteración, se podrán ir agregando tareas a la Lista de tareas de la iteración conforme se las vaya descubriendo.

1.1.4.3 Reunión diaria

Es una reunión de actualización del progreso de las tareas realizadas por cada uno de los miembros del equipo, dado que el desarrollo del producto se lo hace de manera colaborativa, es esencial que todo el equipo comunique sus progresos, para que el resto del equipo tenga una idea clara del avance del trabajo de todo el equipo.

Una reunión diaria no debe durar más de 15 minutos y debe realizarse a la hora convenida por todos los miembros del equipo, debido a que es una reunión breve, de acuerdo con Deemer y colegas (2012) todo el equipo deberá estar preparado para informar al resto del equipo lo siguiente:

- ¿Qué se realizó desde la última reunión?
- ¿Qué se realizará hasta la próxima reunión?
- ¿Qué problemas o bloqueos se han encontrado?

Cuando se han respondido a estas preguntas por parte de cada uno de los miembros, el facilitador deberá ayudar a solucionar los bloqueos encontrados por los miembros del equipo en caso de necesitarse más aclaraciones o surgen dudas sobre el desarrollo del proyecto, se deberá tomar un tiempo más, posterior a la reunión diaria.

1.1.4.4 Revisión de la iteración

Posterior a la finalización de la iteración, se realiza la reunión de inspección de todo el proceso realizado por el equipo, en esta reunión entre los puntos principales a considerarse están la revisión del software realizado por el equipo y el obtener

retroalimentación por parte de los asistentes para poder adaptar las sugerencias a las siguientes iteraciones a realizarse.

El facilitador deberá invitar a todos los interesados del producto para que asistan a esta reunión, los mismos que con sus comentarios o sugerencias ayudarán al mejoramiento del producto en las iteraciones subsiguientes (Hundermark, 2009). Posterior a la realización de esta reunión, las consecuencias para el desarrollo del proyecto son varias, entre las cuales se debe contemplar la cancelación del proyecto y la aprobación para continuar trabajando en la siguiente iteración.

El tiempo a usarse para una reunión de revisión de la iteración es proporcional al número de semanas de la iteración, por ejemplo, si la iteración tiene una duración de dos semanas, dos horas serán suficientes para realizar esta reunión.

1.1.4.5 Retrospectiva de la iteración

Es la última y una de las reuniones más importantes dentro de una iteración y no debe omitirse. La retrospectiva está centrada en la revisión del proceso de desarrollo del producto y ayuda al equipo a comprender como ha sido su desempeño en general a lo largo de toda la iteración.

Como conclusiones de esta reunión se pueden obtener los puntos a mejorar sobre el proceso de desarrollo, lo que incluye habilidades, técnica, buenas prácticas y herramientas de desarrollo.

1.1.4.6 Reunión de estimación

Esta reunión no es muy común en *Scrum*, pero es utilizada cuando existe un número de entregas del producto constantes en las que se pueden incluir características y funcionalidades valiosas que cumplan el criterio de funcionalidad lista o terminada.

El cliente invita a todo el equipo *Scrum* a participar de esta reunión en la cual se va a estimar el costo que supondrá realizar todos los nuevos elementos incluidos en la lista de tareas o a recalcular el costo de los ya pre existentes, y de ser necesario dividirlos en ítems más pequeños para que puedan ser realizados en las siguientes iteraciones.

1.1.5 Gestión de las entregas del producto

Una entrega del producto se da cuando el cliente decide que el software desarrollado a lo largo de las iteraciones está listo para lanzarse. De acuerdo a la visión que tiene *Scrum*, el producto está generalmente listo al final de cada iteración, lo que supondría que cuando se planea una entrega no debe existir trabajo pendiente al final de la revisión de cada iteración.

En caso de que existan ítems por terminar o afinar para poder lanzar el producto, es necesario una iteración de lanzamiento, en la cual se realizará todo el trabajo pendiente hasta que el producto este totalmente terminado.

1.2 Modelo Vista Controlador

Según describen Qureshi y Sabir (2013) “MVC o Modelo–Vista–Controlador es un estilo de arquitectura de software que divide una aplicación o una de sus partes en tres capas definidas”:

- El Modelo, que contiene la lógica de negocio, es decir todos los procedimientos que interactúan con los datos.
- La Vista, comprende una representación del modelo de una forma adecuada para el cliente, generalmente mediante la interfaz de usuario

El Controlador, actúa entre el Modelo y la Vista, es decir gestiona el flujo de información entre ellos.

MVC fue originalmente desarrollado para separar las entradas, del procesamiento y de las salidas (Qureshi & Sabir, 2013), las mismas que serían las acciones que realiza cada una de las capas antes nombradas.

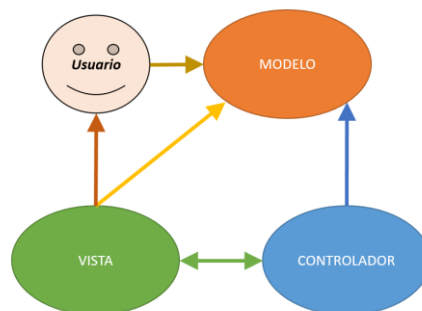


Gráfico 2 – Diagrama Modelo Vista Controlador (MVC)

Fuente: (López Yépez & Polo Vásconez, 2016)

1.2.1 Ventajas

- Separación entre los componentes de un programa, es decir, podemos implementarlos por separado
- Interfaz de Programación de Aplicaciones API claramente definida, se puede reemplazar el Modelo, Vista o Controlador sin dificultad.

La conexión entre el Modelo y la Vista, se realiza en tiempo de ejecución y no en tiempo de compilación.

Al incorporar la arquitectura MVC al diseño de un producto software, todos los componentes a desarrollarse se pueden construir por separado y luego unirlos en tiempo de ejecución. Cuando se realicen las diferentes pruebas de integración entre las partes desarrolladas, en caso de que alguno de los elementos implementados funcione incorrectamente, simplemente será corregido y actualizado sin que el resto se vea afectado.

1.2.2 Utilidad de MVC

Al patrón MVC generalmente se lo encuentra y se lo utiliza cuando se requiere desarrollar aplicaciones con interfaces complejas (Fernández Romero & Díaz González, 2012). Este tipo de patrón, se usa frecuentemente cuando se requiere que los datos estén desacoplados de la vista o de la interfaz, cuando se logra esto, se puede reutilizar o reusar las partes que conforman el sistema.

Cualquier modificación que se realiza sobre la vista no tiene mayor impacto en la programación o en los datos.

El patrón MVC es muy utilizado en el desarrollo de páginas web, cuando la capa de la vista es una página HTML que visualiza información al usuario dinámicamente, el modelo consta de una base de datos y el controlador es toda la lógica existente en la programación de la página.

1.2.3 Definición de las capas

Según (Fernández Romero & Díaz González, 2012), las tres capas del Modelo Vista Controlador (MVC) son: Según Fernández y colaboradores (2012)(Fernández Romero & Díaz González, 2012), las tres capas del Modelo Vista Controlador (MVC) son:

1.2.3.1 Modelo

Es la capa del modelo que representa los datos del programa, es decir, maneja todo lo referente a los datos que se van a utilizar en el programa y se encarga de sus

transformaciones, como está claro, la capa Modelo, no tiene conocimiento de la Vista, ni del Controlador, tampoco existen referencias hacia las capas antes mencionadas, el propio sistema es el encargado de interactuar y mantener activos los enlaces entre el Modelo y las Vistas que este maneja.

El rol que tiene la capa Modelo, es contener los datos del programa y la funcionalidad asociada a estos datos (Qureshi & Sabir, 2013).

Fernández y colaboradores (2012)(Fernández Romero & Díaz González, 2012), manifiesta que la capa Modelo cuenta con varias responsabilidades, entre las que se encuentran:

- Acceder a la capa de almacenamiento de datos, esto debido a que lo ideal en MVC es que el almacenamiento de los datos e información sean independientes de las capas de MVC, especialmente de la capa Modelo.
- En esta capa, se definen las reglas de negocio, es decir todas las funcionalidades del sistema relacionadas con los datos.
- Notifica a la vista los cambios que sufren los datos externamente, por ejemplo, cuando son actualizados por otro software.

1.2.3.2 Vista

Es la capa encargada de presentar visualmente los datos de la capa Modelo, según lo explicado por Fernández et al. (2012), la Vista, “genera una representación visual y muestra los datos al usuario, interactúa preferentemente con el controlador, pero es posible que trate directamente con el Modelo a través de una referencia al propio Modelo.”

La vista tiene como objetivos:

- Recibir todos los datos procesados por la capa Controlador o Modelo y presentarlos al usuario.
- Mantener un registro de su controlador asociado.

1.2.3.3 Controlador

Es la capa del modelo que le da significado a las órdenes enviadas por el usuario, es decir, actúa sobre los datos de la capa Modelo, e interactúa entre las capas Vista y Modelo, en caso de existir cambios en las capas vista y modelo, esta capa entra en

funcionamiento. Esta capa interactúa con la capa Modelo, con una referencia directa al propio Modelo (Fernández Romero & Díaz González, 2012).

El controlador además es el encargado de:

- Recibir los eventos entrada del sistema.
- Contiene las reglas de gestión de los eventos, estas reglas generalmente definen las acciones a tomarse de acuerdo a peticiones a la capa modelo o a las diferentes vistas asociadas.

1.3 IFML

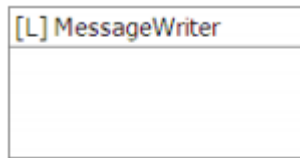
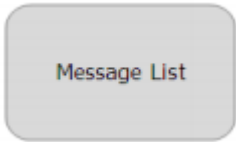
IFML (*Interaction Flow Modeling Language*) es un lenguaje de modelo de desarrollo de aplicaciones que “ha sido diseñado para expresar el contenido, la interacción con el usuario y el comportamiento de la Interfaz” (Garcia, 2012) a través de modelos de flujo de interacción que describen las principales dimensiones de una aplicación, su distribución de control, los datos y la lógica de negocio en los diferentes niveles de la arquitectura.

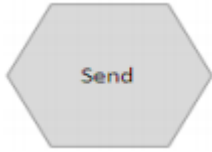
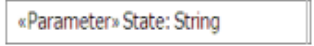
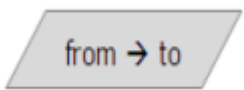
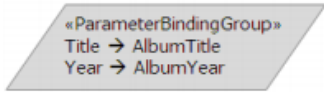
Además, es necesario conocer que IFML se enfoca en la capa “Vista” del modelo (MVC) y como este interactúa con las capas de modelo y controlador, siendo adoptado como estándar en herramienta de desarrollo WebRatio (Rossi, 2013). La definición de IFML para cada una de las capas del modelo MVC que expresa la Object Management Group (2015) se presentan a continuación:

- Vista: IFML se compone de la capa Vista y la descripción de los elementos que se exponen al usuario para la interacción.
- Controlador: IFML en la capa del controlador permite al diseñador de la aplicación, especificar los efectos de las interacciones del usuario y eventos del sistema en la aplicación mediante la definición de los eventos relevantes que el controlador debe cuidar y manejar.
- Modelo: IFML en la capa de modelo permite manejar la especificación de las referencias a los objetos de datos que incorporan el estado de la aplicación y aparecen en la interfaz gráfica de usuario (GUI), así como también, de la referencia a las acciones que se desencadena por la interacción del usuario.

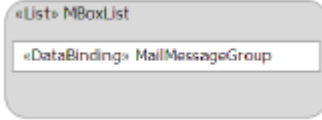
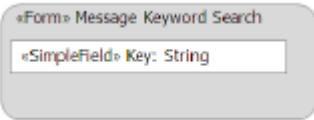
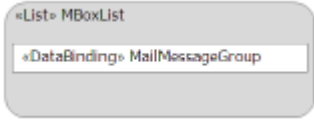
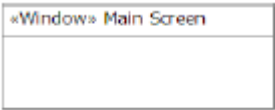
IFML soporta plataformas gráficas independientes de las necesidades de interfaz que requiera un usuario, como son: computadores, teléfonos móviles, PDAs, etc. (Object

Managment Group, 2015). IFML representa sus conceptos esenciales en las siguientes tablas:

Concepto	Significado	Notación IFML	Ejemplo a nivel de implementación
Contenedor de vista	Un elemento de la interfaz, que comprende a su vez elementos que muestran contenido de apoyo a la interacción con otros contenedores de vistas.		Página web / Panel de una página web
Contenedor de vista XOR	Un contenedor de vista comprende uno o varios contenedores de vista hijos que se muestran alternativamente.		Paneles con pestañas en Java / marcos en HTML
Contenedor de vista destacado	Un contenedor de vista que es accesible desde cualquier otro elemento de la interfaz de usuario, sin tener explícitos Flujos de Interacción entrantes.		Un vínculo de cierre de sesión en los sitios HTML que es visible en todas las páginas.
Contenedor de vista por defecto	Un contenedor de vista que se presentará de forma predeterminada para el usuario, cuando se accede al contenedor que lo encierra.		Una página de bienvenida
Componente de vista	Un elemento de la interfaz que muestra contenido o acepta entradas.		Una lista en HTML. Una galería de imágenes en JavaScript. Un formulario de entrada.
Evento	Una ocurrencia que afecta el estado de la aplicación.		Presionar el botón guardar de un formulario

Concepto	Significado	Notación IFML	Ejemplo a nivel de implementación
Acción	Una parte de la lógica de negocio lanzada por un evento; puede ser del lado del servidor (por defecto) o del lado del cliente, que se denota como [cliente].		Una actualización a la base de datos. El envío de un correo electrónico. La corrección ortográfica de un texto.
Flujo de navegación	Actualización de los elementos de la interfaz a la vista o desencadenantes de una acción provocada por la ocurrencia de un evento. Los datos pueden estar asociados con el flujo a través de enlaces de parámetros.		Envío y recepción de parámetros en la solicitud HTTP
Flujo de datos	Pasar datos entre Componentes de vista o "Acciones" como consecuencia de una previa interacción del usuario		Pasar los datos de un formulario a un módulo de carga de datos
Parámetro	Un valor tecleado y nombrado	Opcionalmente se muestra. Si es necesario pueden designarse como sigue: 	Cadena de parámetros de consulta HTTP, enviar parámetros HTTP, variables JavaScript y parámetros de función.
Enlace de parámetro	Especificación de que un parámetro de entrada de una fuente, cuando se asocia con un parámetro de salida hacia un destino.		Paso de un parámetro entre dos componentes.
Grupo de enlaces de parámetro	Conjunto de enlaces de parámetro asociados a un flujo de interacción (traerlo de		Paso de varios parámetros entre un módulo de cálculo de impuestos y un

Concepto	Significado	Notación IFML	Ejemplo a nivel de implementación
	navegación o por flujo de datos).		módulo de guardar el formulario
Expresión de activación	Expresión booleana asociada con un elemento de vista, partes de componentes de vista, o eventos. Si la expresión es verdadera el elemento está habilitado.		Un enlace se activa si la cantidad de productos disponibles es superior a 10
Expresión de un flujo de interacción	Determina cuál de los Flujos de interacción seguirá, como consecuencia de la ocurrencia de un evento.		Activado por un evento después de seleccionar un determinado valor en un campo de selección.
Módulo	Parte de la interfaz de usuario y sus acciones correspondientes, que pueden ser reutilizados para mejorar la mantenibilidad de los modelos IFML.		Modulo para la interfaz que permita la edición del perfil de usuario
Puerto de entrada	Un punto de interacción entre un módulo y su entorno, que recoge los flujos de interacción y parámetros que llegan al módulo.		Parámetros necesarios para crear un usuario del sistema
Puerto de salida	Un punto de interacción entre el módulo y su entorno, que recoge los flujos de interacción y parámetros que van desde el módulo.		Mensaje de respuesta luego de la ejecución exitosa de un módulo
Parte de un componente de vista	Una parte de un componente de vista que no puede vivir por su cuenta. Puede desencadenar	 Ejemplos:	Campos en un formulario

Concepto	Significado	Notación IFML	Ejemplo a nivel de implementación
	eventos y cuenta con flujos de interacción salientes y entrantes. Una parte de un componente de vista puede contener otras partes de un componente de vista.	 <pre> «DataBinding» MailMessage «ConditionalExpression» MailMessage in MailMessageGroup2MailMessage(MBox) «SimpleField» to: String </pre>	
Evento seleccionar	Evento que denota la selección de un solo elemento de la interfaz de usuario.		Selección de una fila en una tabla
Evento enviar	Evento que activa un paso de parámetros entre los elementos de flujo de interacción.		Un botón de envío del formulario en HTML.
Lista	Componente de vista usado para mostrar una lista de instancias de enlaces de datos.	 <pre> «List» MBoxList «DataBinding» MailMessageGroup </pre>	Tabla con filas de elementos de la misma clase.
Formulario	Componente de vista usado para mostrar un formulario que se compone de campos.	 <pre> «Form» Message Keyword Search «SimpleField» Key: String </pre>	Formulario HTML
Detalles	Componente de vista usado para mostrar los detalles de una instancia de enlace de datos específico.	 <pre> «List» MBoxList «DataBinding» MailMessageGroup </pre>	Detalles de un perfil de usuario
Ventana	Un contenedor de vista representado como una ventana.	 <pre> «Window» Main Screen </pre>	Una página HTML o una ventana de escritorio

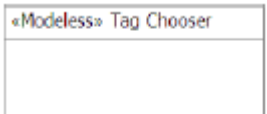
Concepto	Significado	Notación IFML	Ejemplo a nivel de implementación
Ventana de bloqueo	Un contenedor de vista representado en una nueva ventana que, cuando aparece, bloquea la interacción en todos los demás contenedores, previamente activos		Una ventana emergente modal en HTML
Ventana superpuesta	Un contenedor de vista representado en una nueva ventana, que cuando aparece, se superpone sobre el resto de contenedores, previamente activos, que se mantienen activos.		Una ventana emergente en HTML

Tabla 1 – Conceptos esenciales de IFML

Fuente: (Object Management Group, 2015)

Según Brambilla y Butti (2014). IFML se diseñó para expresar el contenido, la interacción con el usuario y el comportamiento del *Interfaz*, sin embargo, no se enfoca en el modelado de contenidos como el diseño y la apariencia de la Interfaz, por lo que IFML se diseñó pensando en aplicaciones de negocio en donde el uso intensivo de datos es de principal interés para quienes administran proyectos de desarrollo informático que forman parte de las siguientes categorías:

- Aplicaciones móviles
- Aplicaciones cliente–servidor
- Aplicaciones de escritorio
- Aplicaciones web tradicionales basadas en HTML
- Aplicaciones multicanales y contextualizadas
- (Aplicaciones de Internet enriquecidas (RIAs) soportadas por el estándar HTML 5.

- Interfaces de usuario-máquina (HMI) integradas para el control de las aplicaciones.

1.4 WebRatio

WebRatio es un entorno de desarrollo de aplicaciones web y móviles que permite la interpretación de la lógica del negocio gracias a su integración con el estándar de modelado IFML (Brambilla & Butti, 2014).

“WebRatio genera aplicaciones Web y móviles rápidas, seguras, escalables y robustas a partir de modelos IFML y BPMN” (Brambilla & Butti, 2014). Las aplicaciones que WebRatio genera, son fácilmente adaptables a los estándares HTML5, CSS3 y Java sin que afecten mayormente en cambios de códigos y tecnologías. (Brambilla & Butti, 2014)

Características principales de WebRatio según Garcia (2012):

- WebRatio dispone de las funcionalidades para diseñar modelos de datos navegables utilizando IFML.
- WebRatio permite utilizar cualquier gestor de base de datos.
- WebRatio genera un único fichero XML con todos los modelos definidos para un proyecto.
- WebRatio crea automáticamente las tablas, atributos y relaciones creadas en el Modelo de dominio para la base de datos.
- Finalmente, genera código Java automáticamente para una aplicación Web a partir del modelo en IFML.

1.4.1 Conceptos principales de Modelado con WebRatio

WebRatio cuenta con muchos conceptos importantes y básicos que permiten entender el desarrollo de aplicaciones usando el estándar de modelado IFML, los conceptos necesarios que se aplicarán en el desarrollo del presente trabajo de tesis se describen a continuación:

Elementos

Para el modelado de una aplicación debemos usar los siguientes elementos:

1.4.1.1 Aplicaciones web

Una aplicación web desarrollada en WebRatio tiene varias características o conceptos que conforman su estructura y permiten definirla usando el estándar de modelado IFML, estas características son:

1.4.1.1.1 Proyecto WebRatio

Un proyecto WebRatio mantiene la información del modelo de la aplicación Web. El proyecto incluye la información sobre el modelo de dominio y sobre los elementos del modelo Web. Esta información se actualiza constantemente a medida que el usuario cambia el modelo en el área de trabajo. (WebRatio, Webratio Enterprise Platform, 2014)

1.4.1.1.2 Modelo de dominio

El Modelo de dominio representa gráficamente el modelo de la base de datos, cuáles son sus entidades y la relación entre ellas. El modelo de dominio está formado por uno o más conjuntos de los siguientes conceptos:

1.4.1.1.2.1 Entidad

Entidad es el concepto central del modelo Entidad–Relación. Una entidad representa una descripción de las características comunes de un conjunto de objetos del mundo real. Como ejemplos de entidades se puede mencionar: persona, automóviles, artistas y sus álbumes. Como todos los conceptos del modelo Entidad–Relación, las entidades se especifican utilizando una notación gráfica. Dichas entidades se denotan por medio de rectángulos, con el nombre de la entidad en la parte superior. (WebRatio, Webratio Enterprise Platform, 2014)

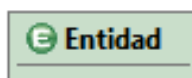


Gráfico 3 – Ejemplo de entidad

Fuente: (López Yépez & Polo Vásquez, 2016)

1.4.1.1.2.2 Atributo

Un atributo representa las propiedades de los objetos del mundo real que son relevantes para los fines de una aplicación. Como ejemplos de atributos tenemos el nombre, la dirección y la fotografía de una persona. (WebRatio, Webratio Enterprise Platform, 2014)

Los atributos se asocian con el concepto de Entidad, con el significado que todas las instancias de la entidad se caracterizan por el mismo conjunto de atributos. En otras palabras, la entidad es un descriptor de las propiedades comunes de un conjunto de objetos, y tales propiedades se expresan como atributos. (WebRatio, Webratio Enterprise Platform, 2014)

Es admisible que una instancia de entidad pueda tener un valor nulo para uno o más atributos, (WebRatio, Webratio Enterprise Platform, 2014) estos atributos pueden subdividirse en las siguientes categorías:

1.4.1.1.2.2.1 Atributo clave

Todas las instancias de una entidad deben ser distinguibles, por medio de una identificación inequívoca. Para expresar la identidad única de instancias de entidad, uno o más atributos pueden definirse como la clave primaria de la entidad. Los Atributos como clave primaria deben satisfacer algunas restricciones, no requeridas para atributos regulares:

- su valor debe ser definido (es decir, no NULL) para cada instancia de la entidad
- su valor debe ser único, lo que significa que no deben existir dos instancias de la entidad con el mismo valor de los atributos clave.

Una buena práctica para definir la clave principal de las entidades, es utilizar un atributo de propósito especial llamado identificador de objeto (abreviado como OID), cuyo único objetivo es asignar un identificador distinto para cada instancia de una entidad. (WebRatio, Webratio Enterprise Platform, 2014)

1.4.1.1.2.2.2 Atributos derivados

Hay cuatro tipos de atributos derivados:

- **Atributo Constante:** el valor es una constante.
- **Atributo Importado:** el valor ha sido importado desde otro atributo de una entidad accesible desde la entidad actual por medio de un camino de una relación con uno como máximo de la cardinalidad.
- **Atributo Calculado:** el valor se calcula aplicando operadores aritméticos de otros atributos o se obtiene aplicando una función agregada al atributo.
- **Atributo Derivado Manual:** permite escribir manualmente la consulta de derivación.

1.4.1.1.2.2.3 Tipos de atributos

Los atributos pueden tener tipos, lo que significa que ellos asumen los valores de los dominios bien definidos, como por ejemplo el conjunto de números enteros o números de punto flotante. El modelo entidad–relación no prescribe ningún conjunto específico de tipos de atributos, pero es una buena práctica expresar los tipos de atributos en el modelo de dominio. (WebRatio, Webratio Enterprise Platform, 2014)

Los atributos de una entidad pueden estar asociados con los tipos de datos habituales, con el apoyo de la mayoría de los lenguajes de programación y productos de base de datos. Estos tipos de datos pueden incluir BLOB, boolean, decimal, float, integer, password, string, text, time, timestamp y URL. (WebRatio, Webratio Enterprise Platform, 2014)

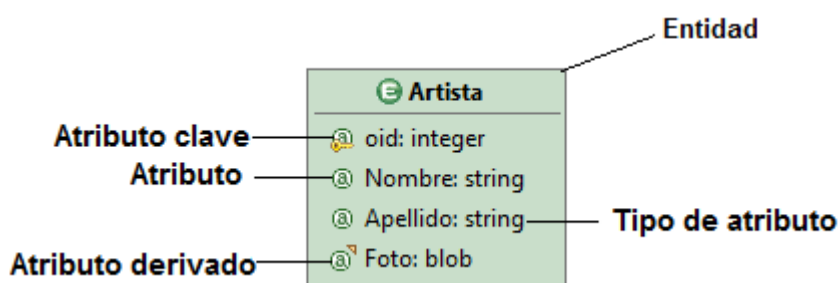


Gráfico 4 – Ejemplo de entidad completa

Fuente: (López Yépez & Polo Vásconez, 2016)

1.4.1.1.2.3 Generalización ↗

El modelo entidad–relación permite al diseñador organizar entidades en una jerarquía, donde comparten algunas características comunes. La jerarquía de generalización básica (también llamada jerarquía Es–Uno o Is–A) tiene una súper–entidad y uno o más sub–entidades. (WebRatio, Webratio Enterprise Platform, 2014)

Una jerarquía de generalización no se limita a dos niveles, pero una sub–entidad puede a su vez ser especializada en una o más sub–entidades, dando una arbitraria jerarquía de profundidad. (WebRatio, Webratio Enterprise Platform, 2014)

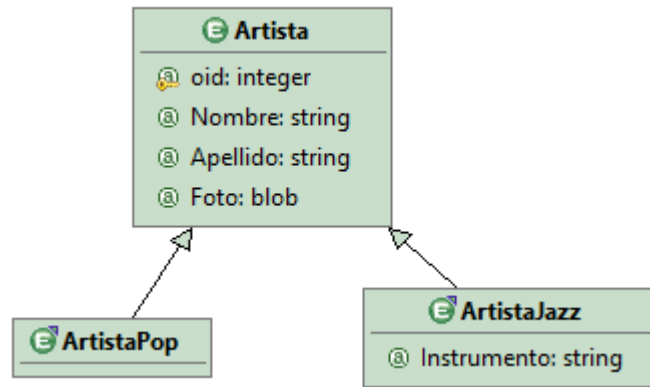


Gráfico 5 – Ejemplo de generalización

Fuente: (López Yépez & Polo Vásconez, 2016)

1.4.1.1.2.4 Relación

Una Relación representa conexiones semánticas entre entidades, como la asociación entre un artista y su álbum, o entre un artista y sus críticas. El significado de la asociación se describe por el nombre de la relación, que se establece por el diseñador. Por ejemplo, la relación entre el artista y los álbumes que él o ella ha publicado podría ser nombrado "publicación". La forma más simple de relación es la relación binaria, que conecta dos entidades. Las relaciones entre más de dos entidades, denominadas relaciones N-aria, no se les permite por IFML y se pueden expresar de manera equivalente por medio de múltiples relaciones binarias. (WebRatio, Webratio Enterprise Platform, 2014)

Cada relación binaria se caracteriza por dos roles de relación, cada una expresando la función que una de las entidades participantes juega en la relación. (WebRatio, Webratio Enterprise Platform, 2014)

Por ejemplo, la relación de publicación entre un artista y su álbum se puede descomponer en dos roles de relación, uno de artista al álbum, nombrado Publica, y uno de los álbumes del artista, nombrado Publicado Por. Por lo tanto, un rol de la relación puede ser considerado como la conexión de una entidad de origen con una entidad de destino. (WebRatio, Webratio Enterprise Platform, 2014)

Entonces los roles de una relación pueden ser anotados con restricciones de cardinalidad máximos, que denota el número máximo de objetos de la entidad de destino a la que cualquier objeto de la entidad de origen puede estar relacionado. Los valores posibles son uno y muchos. Sobre la base de sus limitaciones máximas

de cardinalidad, las relaciones se llaman "uno a uno o 1 a 1", "uno a muchos o 1 a N", y "muchos a muchos o N a N". (WebRatio, Webratio Enterprise Platform, 2014)

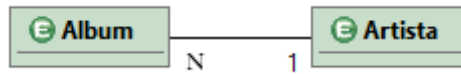


Gráfico 6 – Ejemplo de relación 1 a muchos (1 a N)

Fuente: (López Yépez & Polo Vásconez, 2016)

1.4.1.1.3 Modelo Web

El Modelo Web permite organizar jerárquicamente todos los elementos que forman parte de una aplicación, estos elementos los definimos a continuación:

1.4.1.1.3.1 Vista del sitio

- La vista del sitio está empaquetada en una aplicación que se entrega a los usuarios, esta encierra páginas, áreas, etc. Es decir, los elementos que se pueden definir en el Modelo Web. (WebRatio, Webratio Enterprise Platform, 2014)

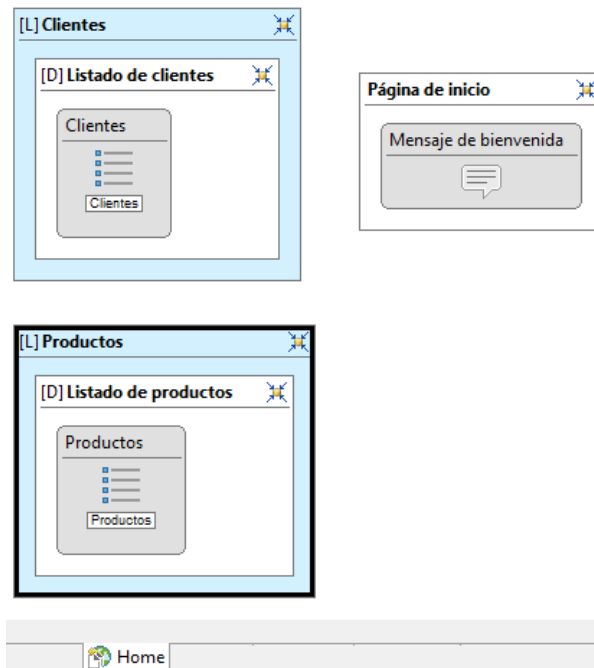


Gráfico 7 – Ejemplo de Vista del sitio

Fuente: (López Yépez & Polo Vásconez, 2016)

1.4.1.1.3.2 Vista del servicio

La Vista del servicio es un conjunto de puertos que exponen la funcionalidad de un servicio web a través de operaciones. (WebRatio, Webratio Enterprise Platform, 2014)

1.4.1.1.3.3 Definiciones de módulo

Definiciones de módulo, es una colección de módulos que están formados por contenido reutilizable, operaciones unitarias y grupos de operaciones. (WebRatio, Webratio Mobile Platform, 2015)

1.4.1.1.3.4 Área

Un Área es un contenedor de páginas o, de forma recursiva, otras sub-áreas, que pueden ser utilizados para dar una organización jerárquica a una Vista del sitio. (WebRatio, Webratio Mobile Platform, 2015)

1.4.1.1.3.5 Página

Una página es un contenedor de componentes, otras sub-páginas y páginas alternativas. (WebRatio, Webratio Mobile Platform, 2015)

Las principales propiedades de una página son:

- **Página de inicio:** es la página en la dirección por defecto del sitio o se presenta cuando el usuario inicia una sesión en la aplicación. La página de inicio debe ser única en la vista del sitio. En la pantalla, una pequeña "casa" en el icono de la página indica esta propiedad habilitada en la página. (WebRatio, Webratio Enterprise Platform, 2014)
- **Página por defecto:** se presenta de forma predeterminada cuando se accede a su área circundante. La página por defecto dentro de un área debe ser única. En la pantalla, una pequeña "D" dentro del icono de la página indica la propiedad predeterminada de una página. (WebRatio, Webratio Enterprise Platform, 2014)
- **Punto prominente:** es la propiedad de una página que indica que es accesible desde todas las otras páginas o áreas. En la pantalla, una pequeña "L" en el icono de la página que indica la propiedad activa. (WebRatio, Webratio Enterprise Platform, 2014)

1.4.1.1.3.6 Módulo

Un módulo es una referencia a un fragmento reutilizable de un modelo de flujo de interacción. La navegación entra desde un puerto de entrada y la salida desde un puerto de salida. (WebRatio, Webratio Mobile Platform, 2015)

1.4.1.1.3.7 Puerto

Un puerto es un conjunto de operaciones de servicios Web. (WebRatio, Webratio Enterprise Platform, 2014)

1.4.1.1.3.8 Selector

Un Selector es un conjunto de criterios de selección que recupera las instancias de la entidad del componente. (WebRatio, Webratio Enterprise Platform, 2014)

Componentes de vista definidos sobre entidades pueden requerir un selector para determinar las instancias de la entidad que se utilizará para construir su contenido. El selector contiene una o más condiciones a un componente o un campo precargado. Tales condiciones se utilizan para hacer la consulta sobre la fuente de datos, que en realidad determina los casos en que se basa el contenido de los componentes. (WebRatio, Webratio Enterprise Platform, 2014) Hay tres tipos de condiciones:

- Condición de atributos
- Condición de clave
- Condición de relación

1.4.1.1.3.8.1 Condición de atributos

Una condición de atributo selecciona sólo las instancias de la entidad para la que el valor de algún atributo satisface un predicado. Por ejemplo, una condición de atributo para una Lista Simple puede ser: "incluir en un índice de combinaciones sólo las combinaciones con precio menor de \$300" (WebRatio, Webratio Enterprise Platform, 2014)

1.4.1.1.3.8.2 Condición de clave

Una condición de clave selecciona sólo las instancias de la entidad para la cual el valor de algún atributo clave satisface un predicado. (WebRatio, Webratio Enterprise Platform, 2014)

1.4.1.1.3.8.3 Condición de relación

Una condición de relación selecciona sólo las instancias de la entidad que están relacionadas con alguna otra instancia de la entidad por medio de una cadena específica de relaciones. Por ejemplo, una condición basada en la relación de una lista simple puede ser: "incluir en esta lista sólo combinaciones de productos que pertenecen a una categoría específica". (WebRatio, Webratio Enterprise Platform, 2014)

1.4.1.1.3.9 Componente de vista

Un Componente de vista es el elemento atómico para especificar el contenido de una página Web. (WebRatio, Webratio Mobile Platform, 2015)

1.4.1.1.3.10 Parámetro de contexto

Un parámetro de contexto es una variable de nivel de aplicación que se utiliza para almacenar valores durante la sesión de un usuario o para toda la duración de la aplicación. (WebRatio, Webratio Mobile Platform, 2015)

1.4.1.1.3.11 Tarea

Una tarea es una secuencia de operaciones y grupos de operaciones para ser programada en un momento o intervalo específico. (WebRatio, Webratio Enterprise Platform, 2014)

1.4.1.1.3.12 Flujos

No existen páginas ni componentes en aislamiento, porque los hipertextos del mundo real están hechos de páginas conectadas, las cuales contienen varias piezas interrelacionadas de contenido y los comandos permiten al usuario interactuar con la aplicación (WebRatio, Webratio Enterprise Platform, 2014). Para expresar estas características, las páginas y los componentes pueden ser vinculados, para especificar los caminos permitidos de navegación entre las páginas, las selecciones que se ofrecen al usuario, y el efecto de la interacción del usuario en el contenido de los componentes que se muestran en la página. El usuario tiene que considerar tres tipos de flujos:

- Flujo, es una conexión orientada entre dos componentes o páginas.
- Flujo correcto, es una conexión orientada entre dos componentes o entre una operación y una página que continúa si una operación tiene éxito.

- Flujo incorrecto, es una conexión orientada entre dos componentes o entre una operación y una página que se continúa si una operación falla.

1.4.1.1.3.12.1 Flujo →

Un Flujo está representado por el concepto de ancla. Un ancla es un dispositivo activo, por lo que el usuario puede interactuar con el hipertexto (hipertexto maneja la vinculación de fragmentos textuales o gráficos a otros fragmentos).

1.4.1.1.3.12.2 Flujo correcto →

Las operaciones pueden tener múltiples flujos salientes. Típicamente, tienen un flujo de salida, seguido después de la ejecución exitosa: el Flujo correcto.

1.4.1.1.3.12.3 Flujo incorrecto →

Las operaciones pueden tener múltiples flujos salientes. Por lo general, tienen un flujo de salida para cuando la operación falla: el Flujo incorrecto.

1.4.1.1.3.13 Enlace de parámetros ➡

Un enlace de parámetros es un valor asociado con un Flujo entre componentes, los cuales transportan, como un efecto de la navegación del Flujo, parámetros desde el componente de origen al componente de destino. (WebRatio, Webratio Mobile Platform, 2015)

1.4.1.1.3.14 Página principal 📄

La Página principal es una página especial que contiene componentes distintos, sub-páginas y páginas alternativas que durante el proceso de generación se insertan en las páginas que se refieren a la Página principal. (WebRatio, Webratio Mobile Platform, 2015)

1.4.1.1.3.15 Acción 📁

Una Acción es un conjunto de operaciones y acciones reutilizables dentro del modelado de una aplicación. (WebRatio, Webratio Mobile Platform, 2015)

1.4.1.1.3.16 Grupo de operaciones 🌻

Grupo de operaciones, es un conjunto de operaciones. Si la propiedad de transacción de un Grupo de operaciones está activada, el mencionado grupo se convierte en una secuencia de operaciones atómicamente ejecutadas, es decir, o bien todas las

operaciones se ejecutan con éxito, o toda la secuencia es retrotraída (WebRatio, Webratio Enterprise Platform, 2014). Las transacciones son un concepto fundamental de los sistemas de bases de datos, que otorgan la sincronización correcta de múltiples usuarios concurrentes que operan en el mismo contenido; que están soportados de forma nativa en la mayoría de productos de gestión de base de datos, y también ofrecen sistemas middleware de última generación. En particular, una transacción garantiza las propiedades "acid" (la llamada mediante la adopción de las iniciales de cada propiedad como un acrónimo):

- Atomicidad: todas las operaciones involucradas se han completado con éxito, la producción de un nuevo estado de base de datos, o el estado de la base de datos inicial se deja sin cambios.
- Consistencia: la realización de una transacción no debe violar la integridad de los datos.
- Aislamiento: cada ejecución de transacción es independiente de la ejecución simultánea de otras transacciones.
- Durabilidad: efectos de las transacciones que se completan con éxito se registran persistentemente.

1.4.1.1.3.17 Operación

Una Operación es un nuevo tipo de componentes, que se pueden colocar fuera de páginas y vincularlos a otras operaciones, o para ver los componentes definidos dentro de las páginas. (WebRatio, Webratio Mobile Platform, 2015)

1.4.1.1.3.18 Disparador

Un Disparador puede ser añadido a tareas, estas tareas se ejecutarán de acuerdo a su configuración en el modelo para un horario pre programado. (WebRatio, Webratio Mobile Platform, 2015)

1.4.1.1.3.19 Regla de validación

Una regla de validación es una condición lógica que se evalúa en la entrada de contenido por parte del usuario en un campo o componente. (WebRatio, Webratio Enterprise Platform, 2014)

1.4.1.1.3.20 Variable

El punto de partida para la especificación de una expresión de activación es la especificación de una o más variables (WebRatio, Webratio Enterprise Platform,

2014). Las variables pueden ser definidas en las páginas de nivel superior y en Páginas principales.

Hay dos tipos de variables:

- Variable basada en componentes: una variable asociada a un componente. Para la variable se define un par <componente, el parámetro>. No es posible proporcionar este tipo de variables como parámetro objetivo vinculante porque no hay una regla determinista y aceptable para evaluar la variable.
- Variable libre: una variable no asociada a un componente. Se proporciona como destino de los parámetros. El proceso de la asignación de valor a una variable libre es:
 - Comprueba el valor traído por un flujo
 - Si no hay un valor, el valor predeterminado se asigna a la variable.

1.4.1.1.3.21 Expresión de activación

Una Expresión de activación es una expresión Groovy que devuelve un valor booleano. (WebRatio, Webratio Enterprise Platform, 2014)

1.4.1.1.4 Servicio/Proveedores de datos

Los Servicio/Proveedores de datos proporcionan datos, como las bases de datos, o la prestación de servicios, al igual que el servidor SMTP y proveedor de servicios de Internet, que son necesarios para que la aplicación pueda trabajar en su totalidad (WebRatio, Webratio Enterprise Platform, 2014)

1.4.1.1.4.1 Fuente de datos

Un proyecto WebRatio puede incluir una o más fuentes de datos, que son servicios externos con el que una aplicación WebRatio puede interoperar. (WebRatio, Webratio Enterprise Platform, 2014)

1.4.1.1.4.2 Servidor SMTP

Un servidor SMTP es la configuración del servidor que permite la gestión de los correos electrónicos. El servidor se encargará de la entrega de mensajes en nombre del usuario. (WebRatio, Webratio Enterprise Platform, 2014)

1.4.1.1.5 Estilo

Un *estilo* es un conjunto de plantillas de diseño que permiten crear el marcado de aplicaciones web en un determinado lenguaje (por ejemplo, HTML o WML). El contenido de las páginas generadas no se limita al margen de beneficio, pero puede incluir cualquier tipo de componentes del lado del cliente, al igual que las películas de Flash, applets Java o controles ActiveX, y cualquier tipo de instrucciones de script del lado del cliente, como Java Script o rutinas Visual Basic Script. El concepto fundamental de Estilo es el de la presentación. (WebRatio, Webratio Enterprise Platform, 2014)

En esencia, una plantilla de diseño es una mezcla de diferentes lenguajes: lenguajes de marcas, componentes del lado del cliente, expresiones Groovy y scriptlets. Los scriptlets y expresiones Groovy pertenecen a algunos elementos de la especificación del Modelo Web (por ejemplo, la aparición de una lista simple) y produce la salida basándose en los elementos coincidentes (por ejemplo, el código HTML para visualizar la lista simple).

Los Estilos se organizan de forma modular, de modo que los diferentes aspectos de la presentación de la página se pueden especificar y cambiar de forma independiente. Crear una hoja de estilo implica abordar ocho temas independientes de presentación (Layout): Página, Grilla, Celda, Marco, Componente, Atributo, Flujo y Campo.

1.4.1.2 Aplicaciones móviles

Una aplicación móvil desarrollada en (WebRatio, Webratio Mobile Platform, 2015) tiene varias características o conceptos que conforman su estructura y permiten definirla usando el estándar de modelado IFML, muchas de estas características son las mismas descritas anteriormente para las aplicaciones web, agregando las siguientes:

1.4.1.2.1 Modelo de dominio

1.4.1.2.1.1 Clase

Una Clase es el concepto central del modelo UML. Una clase es una descripción de las características comunes de un conjunto de objetos del mundo real. Una clase es el equivalente a una entidad en el modelado IFML de una aplicación móvil.

1.4.1.2.1.2 Asociación

Las Asociaciones son relaciones semánticas binarias entre Clases, como en el modelo de diseño de bases de datos populares. Las Asociaciones tienen un nombre y se compone de dos funciones:

- Rol de asociación1 : representa el Rol directo entre dos Clases
- Rol de asociación2 : representa el Rol inverso entre dos Clases

Los Roles representan la misma asociación semántica "as seen (como se ve)" de cada una de las dos *Clases* participantes. Los Roles tienen un nombre y tienen restricciones de cardinalidad, que expresan el número de instancias de la Clase que deben o pueden tomar parte de la Asociación.

1.4.1.2.1.3 Atributos condición de restricción

Una condición basada en un valor de atributo. La propiedad Valor es obligatoria si el predicado elegido requiere de un valor de entrada para calcular el resultado booleano.

1.4.1.2.1.4 Servicios de datos

El servicio de datos contiene todas las clases expuestas a través de los servicios de datos REST.

1.4.1.2.1.5 Clase de servicio

Una Clase de servicio es una referencia a una clase en el Modelo de dominio por el cual un servicio puede estar expuesto.

1.4.1.2.1.6 Método de servicio

Un Método de servicio representa un método único de servicio de datos REST expuesto por la Clase de servicio.

1.4.1.2.1.7 Operación de servicio de usuario

Uno de los métodos disponibles en la clase que representa al usuario (por ejemplo, registro, login, actualización).

1.4.1.2.1.8 Servicios de usuario

Los servicios de usuario contienen todos los métodos REST utilizados para gestionar las clases elegidas para representar al usuario.

1.4.1.2.2 Elementos de un proyecto

1.4.1.2.2.1 Proyecto móvil

Un Proyecto móvil mantiene la información del modelo de la aplicación Móvil. El proyecto incluye la información sobre el Modelo de dominio y elementos Modelo Web

1.4.1.2.2.2 Proyecto de servicio de datos

Un Proyecto de servicio de datos mantiene la información del modelo de la aplicación Web que se conecta con una aplicación móvil y una base de datos externa.

1.4.1.2.2.3 Rol

Un Rol define los permisos de acceso de un usuario a los servicios expuestos en el Proyecto.

1.4.1.2.2.4 Vista de aplicación

Una Vista de aplicación es un grupo de Pantallas o Pantallas dedicadas al cumplimiento de los requisitos de uno o más grupos de usuarios.

1.4.1.2.2.5 Evento regresar

Un Evento regresar es el destino para un Evento de salto. Todos los valores recibidos a lo largo del salto se pasan por flujos salientes.

1.4.1.2.2.6 Pantalla

Una Pantalla es un contenedor de componentes y de barras de herramientas.

1.4.1.2.2.7 Evento selección

El Evento selección representa la interacción del usuario al seleccionar elementos de un Componente de vista. Este se puede colocar en un Componente de vista y permite la ejecución de un Flujo de navegación.

1.4.1.2.2.8 Evento enviar

El Evento enviar representa la interacción del usuario al enviar datos desde un formulario. Este se puede colocar en un Componente de vista y permite la ejecución de un Flujo de navegación.

1.4.1.2.2.9 Pantalla de ajuste

Una Pantalla de ajuste es un contenedor de pantalla o, de forma recursiva, otro conjunto de sub-pantallas, que se puede utilizar para dar una organización jerárquica a la vista de una aplicación.

1.4.1.2.2.10 Barra de herramientas

Una Barra de herramientas es una página especial que contiene componentes distintos, otras sub-páginas y páginas alternativas que durante el proceso de generación se insertan en las páginas que se refieren a las páginas principales.

1.5 Servicios web REST

Los servicios web de tipo REST se usa para describir interfaces entre sistemas que utilicen el protocolo HTTP para obtener datos o para ejecutar operaciones sobre los datos. Las operaciones son descritas en el estándar WADL, estas operaciones son elementos centrales representados como recursos, estos recursos son entidades abstractas identificadas por URIs que pueden ser identificadas por entidades uniformes. (Alarcon, Wilde, & Bellido, Hypermedia-Driven RESTful Service, 2011)

El esquema URI, a través de HTTP puede proveer diferentes métodos estándar de comunicación Obtener, Poner, Enviar, Eliminar y Opciones. (Alarcon & Wilde, Linking Data from RESTful Services, 2010). WebRatio publica automáticamente servicios web basados en REST a través de un Proyecto de servicio de datos, el cual administra y gestiona los servicios necesarios para consular o actualizar una base de datos relacional.

Estos servicios o aplicaciones motor serán consumidos desde la aplicación móvil para alimentar su Modelo de dominio con los datos de una base de datos como MySQL, Oracle, DB2, etc. Replicando la estructura de las tablas y las relaciones entre ellas.

La sincronización de los datos es gestionada completamente por esta aplicación motor, la cual puede ser configurada a gusto del usuario, en donde se puede elegir que datos sincronizar y cada cuanto tiempo. Esta sincronización no se ve afectada por la conexión a internet, debido a que cuando el usuario se encuentre sin conexión los datos son almacenados en la base del teléfono móvil y sincronizados en cuanto posea una conexión, a esto se le conoce como modalidad fuera de línea.

CAPÍTULO 2: APLICACIÓN WEB CON WEBRATIO

Introducción

En este capítulo se definen los requisitos y los casos de uso que se involucran con el desarrollo de la aplicación web. Además, se explica el resultado de cada una de las historias de usuario desarrolladas, el modelado del caso de uso y su resultado en la interfaz web.

2.1 Análisis de requisitos del sistema

El “Asistente de compras” es una aplicación web que permitirá el control de solicitudes de compra para productos en tiendas que se encuentren registradas en el sistema.

Luego de visitar tiendas en línea como (Amazon, 2015), (OLX, 2015) y (Mercado Libre, 2015) se ha determinado que es necesario contar con los siguientes requisitos:

- 2.1 Gestión de tiendas
- 2.2 Gestión de responsables de tiendas
- 2.3 Gestión de productos
- 2.4 Gestión de categorías de productos
- 2.5 Búsqueda de productos
- 2.6 Registro de usuarios
- 2.7 Roles de usuario
 - 2.7.1 Administrador del sistema
 - 2.7.2 Responsable de tienda
 - 2.7.3 Despachador
 - 2.7.4 Cliente
- 2.8 Órdenes de compra
- 2.9 Carrito de compras
- 2.10 Despachadores
- 2.11 Productos y tiendas Favoritas
- 2.12 Facturación electrónica
- 2.13 Envío de notificaciones
- 2.14 Reseñas

Para el detalle de los requisitos ver el [anexo 2](#)

2.2 Historias de usuario

A partir de los requisitos, las historias de usuario necesarias para el desarrollo de la aplicación web nombrada “Asistente de compras” se definen a continuación:

- 3.1 US000 – Configuración de ambiente de desarrollo
- 3.2 US001 – Modelo de dominio
- 3.3 US002 – Gestión de Roles de usuario
- 3.4 US003 – Registro de usuarios
- 3.5 US004 – Gestión de Tiendas
- 3.6 US005 – Gestión de responsables de tiendas
- 3.7 US006 – Gestión de categorías de productos
- 3.8 US007 – Gestión de productos
- 3.9 US008 – Búsqueda de productos
- 3.10 US009 – Carrito de compras
- 3.11 US010 – Orden de compra
- 3.12 US011 – Productos y tiendas favoritas
- 3.13 US012 – Notificaciones automáticas por ofertas en las tiendas
- 3.14 US013 – Reseñas
- 3.15 US014 – Interfaz del sistema

Para el detalle de las historias de usuario ver el [anexo 3](#)

2.3 Desarrollo de las historias de usuario

Posterior a la definición de las historias de usuario se inició con el desarrollo de las mismas, para las cuales se realizó la planificación de los Iteraciones necesarios para finalizar la aplicación web, dichos Iteraciones se encuentra detallados en la lista de tareas del producto en el [anexo 4](#).

El resultado de cada una de las historias de usuario es el siguiente:

2.3.1 Iteración 1

2.3.1.1 US001 – Modelo de dominio

El modelo de dominio maneja la estructura de los datos que la aplicación web necesita, a través del modelo Entidad – Relación

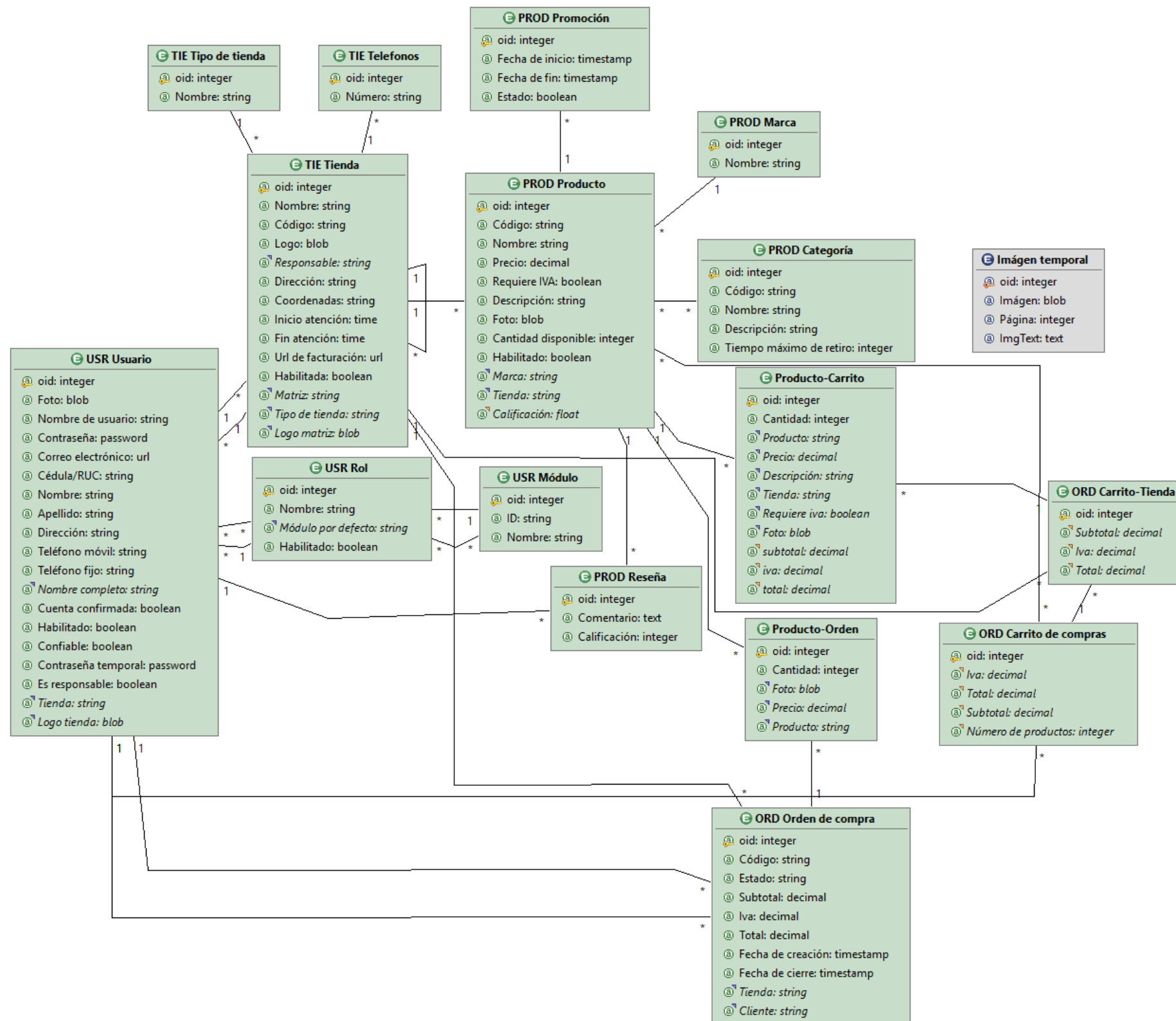


Gráfico 8 – Modelo de dominio aplicación web

Fuente: (López Yépez & Polo Vásconez, 2016)

2.3.1.2 US002 – Gestión de Roles de usuario

La gestión de roles de usuario controla los diferentes permisos de acceso a los módulos (páginas, enlaces, funcionalidades) que conforman la aplicación web, para esto se definió la estructura de modelado presentada en la siguiente imagen:

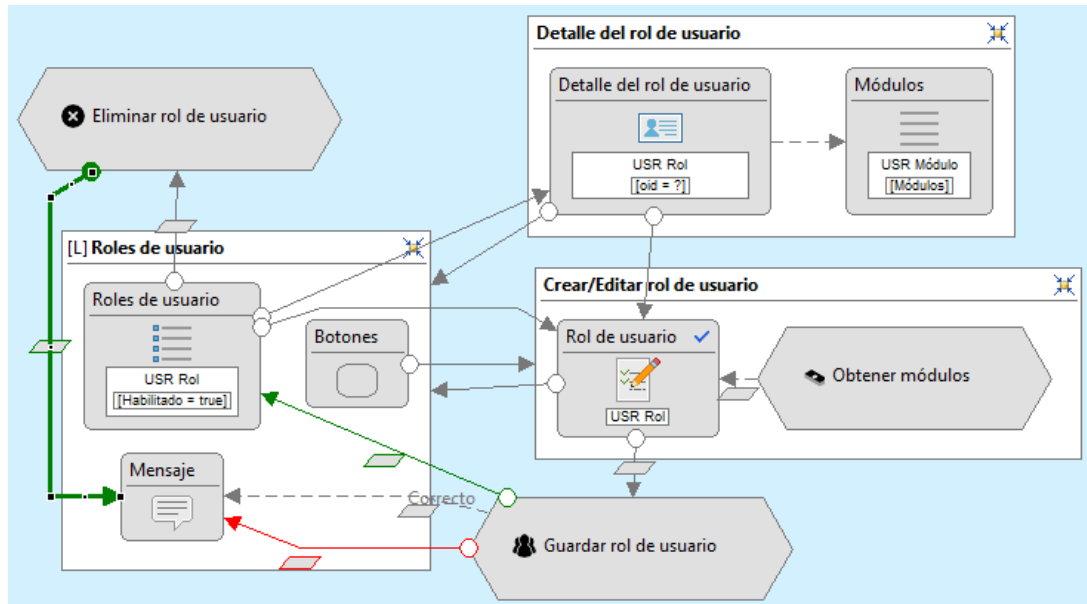


Gráfico 9 – Modelado de Gestión de roles de usuario

Fuente: (López Yépez & Polo Vásconez, 2016)

Esta estructura es representada gráficamente en cada una de sus interfaces web.

2.3.1.2.1 Roles de usuario

La interfaz Roles de usuario contiene el listado de roles de usuario, y permite acceder a la creación, modificación, eliminación y detalle de los roles de usuario que requiere la aplicación web.



Gráfico 10 – Interfaz Roles de usuario

Fuente: (López Yépez & Polo Vásconez, 2016)

Rastro web: permite al usuario conocer su ubicación dentro de la aplicación web.

Nuevo rol de usuario: acceso directo al formulario de “Crear/Editar rol de usuario”

Número de registros por página: permite al usuario modificar el número de registros que se muestran en pantalla (por defecto 10).

Buscador de roles de usuario: campo que permite al usuario buscar roles de usuario por su nombre o por el módulo por defecto.

Detalles: acceso directo a la página “Detalle del rol de usuario”.

Editar: acceso directo al formulario de “Crear/Editar rol de usuario”.

Eliminar: botón que permite la eliminación lógica de un rol de usuario.

Paginación: la paginación permite al usuario navegar por las diferentes páginas creadas para dividir la cantidad de registros que se muestran en pantalla, cuando estos superan el límite de registros por página (por defecto 10).

2.3.1.2.2 Detalle del rol de usuario

La interfaz de detalle del rol de usuario, permite al usuario visualizar los datos de configuración de cada uno de los roles de usuario existentes en el sistema.



Gráfico 11 – Interfaz Detalle rol de usuario

Fuente: (López Yépez & Polo Vásconez, 2016)

Rastro web: permite al usuario conocer su ubicación dentro de la aplicación web.

Editar: acceso directo al formulario de “Crear/Editar rol de usuario”.

Regresar: acceso directo al Listado de roles de usuario.

Datos del rol de usuario: permite al usuario visualizar los datos de configuración de un rol de usuario.

Sección de módulos: visualiza los módulos a los que puede acceder un usuario de acuerdo al rol que le ha sido asignado.

2.3.1.2.3 Crear/Editar rol de usuario

La interfaz Crear/Editar rol de usuario permite configurar los datos de un nuevo rol de usuario o uno previamente ingresado en el sistema.

The screenshot shows a web application interface for creating or editing a user role. The top navigation bar includes 'Asistente de compras', 'Perfil', 'Tiendas', 'Productos', 'Configuración del sistema', and 'Órdenes de compra'. The current page is 'Configuración del sistema / Crear/Editar rol de usuario'. Annotations with red arrows point to specific elements: 'Rastro web' points to the breadcrumb; 'Guardar rol de usuario' points to the save button; 'Regresar' points to the back button; 'Datos del rol de usuario' points to the 'Nombre' and 'Módulo por defecto' fields; and 'Sección de módulos' points to the 'Módulos' list. The 'Nombre' field contains 'Cliente' and the 'Módulo por defecto' field contains 'Inicio'. The 'Módulos' list includes various system functions, with 'Inicio', 'Mi carrito', and 'Mis órdenes de compra' checked.

Gráfico 12 – Interfaz Crear/Editar rol de usuario

Fuente: (López Yépez & Polo Vásconez, 2016)

Rastro web: permite al usuario conocer su ubicación dentro de la aplicación web.

Guardar: permite al usuario almacenar los datos del formulario “Crear/Editar rol de usuario”.

Regresar: acceso directo a la Listado de roles de usuario.

Datos del rol de usuario: configura los datos de un rol de usuario.

Sección de módulos: configura los módulos a los que puede acceder un.

2.3.1.3 US003 – Registro de usuarios

2.3.1.3.1 Registro de usuarios

El registro de usuarios permite a nuevos usuarios acceder al sistema y usar todas las funcionalidades definidas para el rol cliente. Para su gestión, se definió la estructura de modelado presentada en la siguiente imagen:

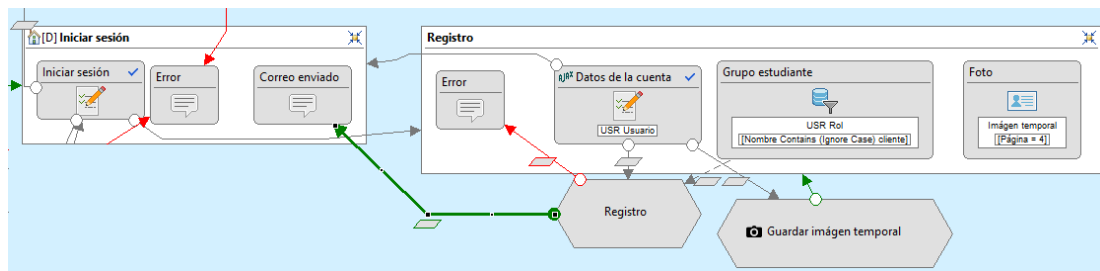


Gráfico 13 – Modelado Registro de usuarios

Fuente: (López Yépez & Polo Vásconez, 2016)

Esta estructura es representada gráficamente en la siguiente interfaz web:

Gráfico 14 – Interfaz registro de usuarios

Fuente: (López Yépez & Polo Vásconez, 2016)

Rastro web: permite al usuario conocer su ubicación dentro de la aplicación web.

Guardar: permite al usuario almacenar los datos del formulario de registro.

Regresar: acceso directo al inicio de sesión de la aplicación web.

Formulario de registro de usuario: contiene los campos de información que el sistema necesita de un usuario.

2.3.1.3.2 Confirmación de la cuenta

El sistema obliga a los usuarios que se han registrado en la aplicación a que confirmen su cuenta, para esto el sistema envía un correo electrónico usando la información que fue ingresada desde el formulario de registro, el correo electrónico que un usuario recibe tiene el siguiente formato:

Estimado/a \$\$nombrecompleto\$\$,
 Gracias por su registro, para confirmar y activar su cuenta por favor haga clic en el siguiente enlace:
[Confirmar registro](#)

Gráfico 15 – Formato de correo de confirmación de la cuenta

Fuente: (López Yépez & Polo Vásconez, 2016)

\$\$nombrecompleto\$\$: nombre y apellido del usuario que debe confirmar su cuenta.

2.3.1.3.3 Recuperar cuenta

Los usuarios que se han registrado en el sistema o que han sido creados por un administrador, tienen la posibilidad de recuperar su cuenta, para esto se definió la siguiente estructura de modelado:

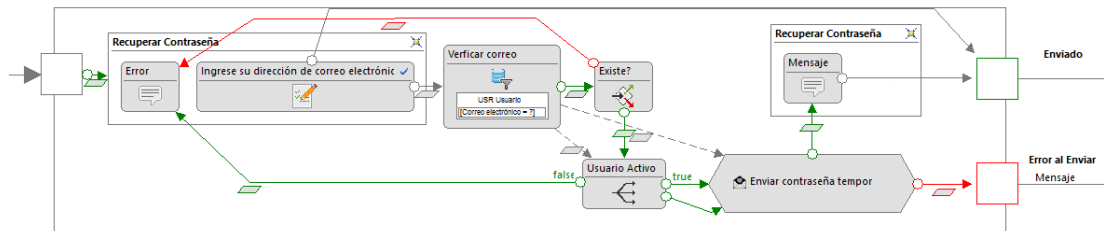


Gráfico 16 – Modelado Recuperar cuenta

Fuente: (López Yépez & Polo Vásconez, 2016)

Esta estructura es representada gráficamente en la siguiente interfaz web:

Gráfico 17 – Interfaz Recuperar cuenta

Fuente: (López Yépez & Polo Vásconez, 2016)

Rastro web: permite al usuario conocer su ubicación dentro de la aplicación web.

Siguiente: permite al sistema verificar el correo electrónico de la cuenta que desea recuperar.

Regresar: acceso directo al inicio de sesión de la aplicación web

Correo electrónico: correo electrónico de la cuenta que va a recuperar.

2.3.1.3.4 Gestión de usuarios

La gestión de usuarios controla a los diferentes grupos usuarios (Administrador del sistema, Responsable de tienda, Despachador y cliente) para permitirle o denegarle el acceso al sistema y configurar los datos que la aplicación necesita de cada usuario, para esto se definió la estructura de modelado presentada en la siguiente imagen:

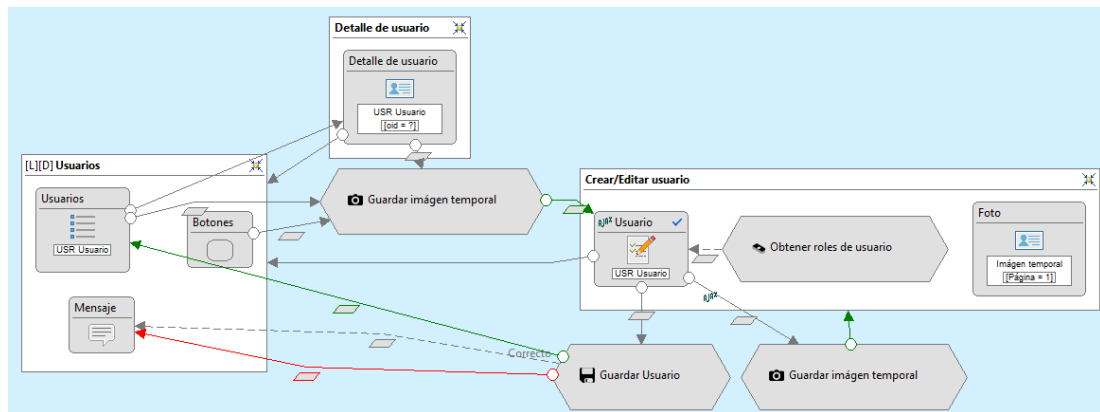


Gráfico 18 – Modelado Gestión de usuarios

Fuente: (López Yépez & Polo Vásconez, 2016)

Esta estructura se representa gráficamente en cada una de sus interfaces web.

2.3.1.3.4.1 Interfaz Usuarios

La interfaz Usuarios contiene el listado de usuarios, desde donde se puede acceder a la creación, modificación, eliminación y detalle de los usuarios ingresados en el sistema.



Gráfico 19 – Interfaz Usuarios

Fuente: (López Yépez & Polo Vásconez, 2016)

Rastro web: permite al usuario conocer su ubicación dentro de la aplicación web.

Nuevo: acceso directo al formulario de “Crear/Editar usuario”

Número de registros por página: permite al usuario modificar el número de registros que se muestran en pantalla (por defecto 10).

Buscador de usuarios: campo que permite al usuario administrador buscar usuarios por todas las columnas del listado.

Detalles: acceso directo a la página “Detalle de usuario”.

Editar: acceso directo al formulario de “Crear/Editar usuario”.

Paginación: la paginación permite al usuario navegar por las diferentes páginas creadas para dividir la cantidad de registros que se muestran en pantalla, cuando estos superan el límite de registros por página (por defecto 10).

2.3.1.3.4.2 Detalle de usuario

La interfaz detalle de usuario, permite al usuario administrador visualizar los datos de configuración de cada uno de los usuarios existentes en el sistema.



Gráfico 20 – Interfaz Detalle de usuario

Fuente: (López Yépez & Polo Vásconez, 2016)

Rastro web: permite al usuario conocer su ubicación dentro de la aplicación web.

Editar: acceso directo al formulario de “Crear/Editar usuario”.

Regresar: acceso directo al Listado de usuarios.

Datos del usuario: permite al usuario administrador visualizar los datos de configuración de un usuario.

2.3.1.3.4.3 Crear/Editar usuario

La interfaz Crear/Editar usuario permite al usuario administrador configurar los datos de un nuevo usuario o uno previamente ingresado en el sistema.

Asistente de compras

Perfil Tiendas Productos Configuración del sistema

Configuración del sistema / Crear/Editar usuario **Rastro web** **Datos del usuario** Guardar

Cédula/RUC * - Foto * user-image-with-black-backgro Elegir archivo

Nombre * Javier

Apellido * Polo

Dirección * -

Teléfono móvil -

Teléfono fijo -

Rol por defecto * Administrador del sistema

Habilitado ☒ Si ☐ No

Nombre de usuario * admin

Contraseña * *****

Correo electrónico * javierpolo@outlook.com

Gráfico 21 – Interfaz Crear/Editar usuario

Fuente: (López Yépez & Polo Vásconez, 2016)

Rastro web: permite al usuario conocer su ubicación dentro de la aplicación web.

Guardar: permite al usuario administrador almacenar los datos del formulario “Crear/Editar usuario”.

Regresar: acceso directo a la Listado de usuarios.

Datos del usuario: permite al usuario administrador configurar los datos de un usuario.

2.3.1.3.5 Perfil de usuario

Los usuarios que se han registrado en la aplicación o que han sido creados por un administrador del sistema, tienen acceso a su perfil de usuario, en donde pueden actualizar sus datos, para esto se definió la estructura de modelado presentada en la siguiente imagen:

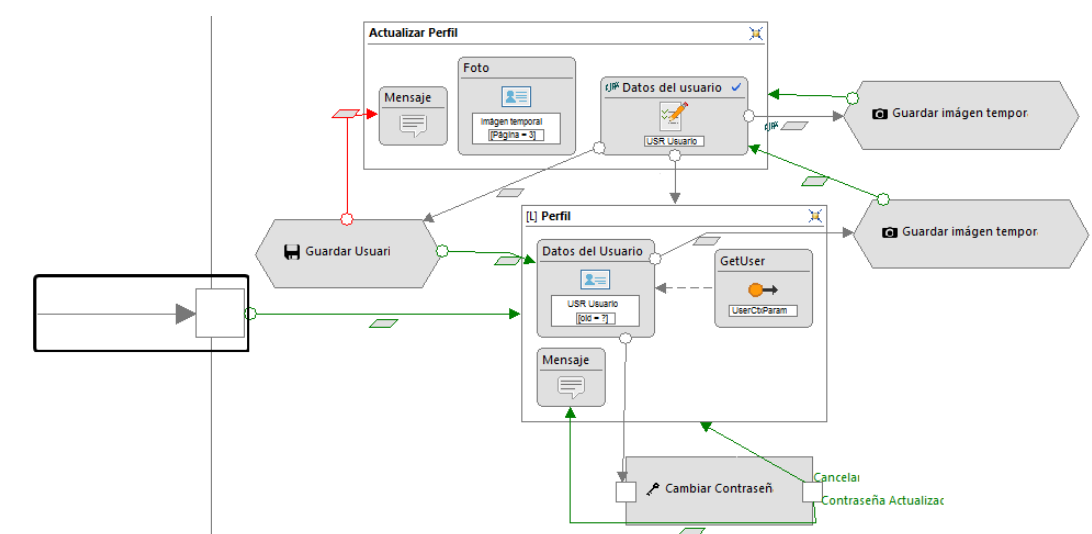


Gráfico 22 – Modelado Perfil de usuario

Fuente: (López Yépez & Polo Vásconez, 2016)

Esta estructura es representada gráficamente en cada una de sus interfaces web.

2.3.1.3.5.1 Interfaz Perfil

La interfaz Perfil contiene los datos del usuario que inició sesión en la aplicación web, desde donde puede actualizar sus datos y contraseña.

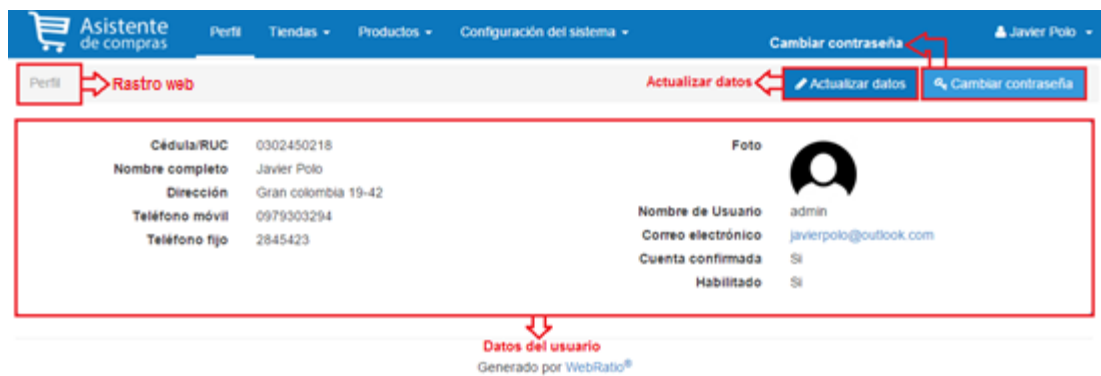


Gráfico 23 – Interfaz Perfil

Fuente: (López Yépez & Polo Vásconez, 2016)

Rastro web: permite al usuario conocer su ubicación dentro de la aplicación web.

Actualizar datos: acceso directo al formulario de “Actualizar datos de perfil”

Cambiar contraseña: acceso directo al formulario “Cambiar contraseña”.

Datos del usuario: permite visualizar los datos del perfil de un usuario.

2.3.1.3.5.2 Actualizar datos

La interfaz Actualizar datos, permite al usuario realizar cambios en sus datos de perfil.

Actualizar datos de perfil → Rastro web

Guardar ← Regresar

Cédula/RUC * Foto * [Elegir archivo](#)

Nombre *

Apellido *

Dirección *

Teléfono móvil

Teléfono fijo

Nombre de usuario *

Correo electrónico *

Datos del usuario
Generado por WebRatio®

Gráfico 24 – Interfaz Actualizar datos

Fuente: (López Yépez & Polo Vásconez, 2016)

Rastro web: permite al usuario conocer su ubicación dentro de la aplicación web.

Guardar: permite al usuario almacenar los datos del formulario “Actualizar datos de perfil”.

Regresar: acceso directo a Perfil.

Datos del usuario: permite al usuario configurar sus datos de perfil.

2.3.1.3.5.3 Cambiar contraseña

La interfaz Cambiar contraseña, permite al usuario cambiar su contraseña actual.

Gráfico 25 – Interfaz Cambiar contraseña

Fuente: (López Yépez & Polo Vásconez, 2016)

Rastro web permite al usuario conocer su ubicación dentro de la aplicación web.

Guardar: permite al usuario almacenar los datos del formulario “Cambiar contraseña”.

Regresar: acceso directo a Perfil.

Datos de contraseña: permite al usuario configurar su nueva contraseña.

2.3.2 Iteración 2

2.3.2.1 US014 – Interfaz del sistema

La interfaz del sistema es una parte esencial en el desarrollo de una aplicación, esta historia de usuario nos permitirá transmitir gráficamente a los usuarios el objetivo principal de la aplicación, para esto se utilizó la infraestructura Bootstrap obtenida desde la tienda de WebRatio.

Los elementos principales que se identifican en el desarrollo de esta historia de usuario son los siguientes:

- **Logo:** el logo de la aplicación permite al usuario identificar a la aplicación distinguiéndola de otras aplicaciones.



Gráfico 26 – Logo

Fuente: (López Yépez & Polo Vásconez, 2016)

- **Color principal:** es el color que se utiliza en la mayoría de los elementos de la interfaz de la aplicación web.

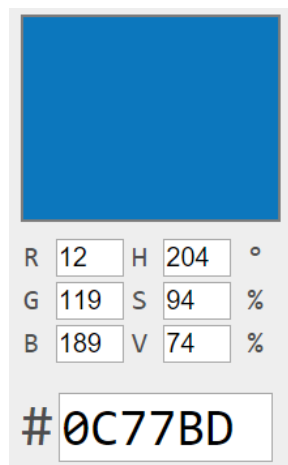


Gráfico 27 – Color principal aplicación web

Fuente: (López Yépez & Polo Vásconez, 2016)

- **Fuente tipográfica principal:** la fuente tipográfica representa el tipo de letra usada en toda la aplicación web, en este caso particular usamos el tipo sans-serif, de tamaño 14px.

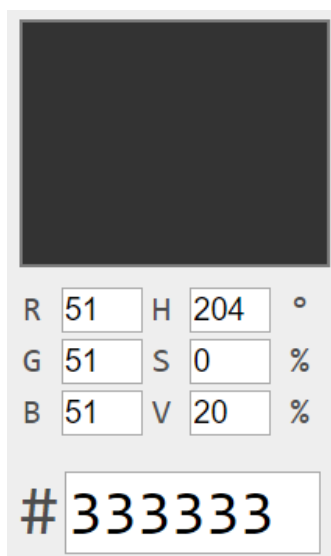


Gráfico 28 – Color fuente principal

Fuente: (López Yépez & Polo Vásconez, 2016)

- **Color principal de botones:** el color principal de los botones nos permite distinguir o resaltar un botón de otro de acuerdo a su importancia dentro de cada uno de los formularios.

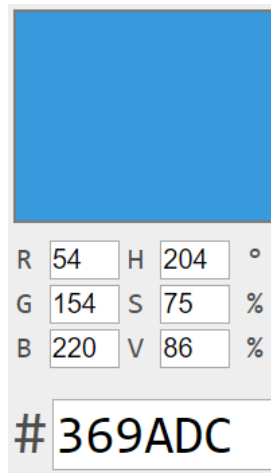


Gráfico 29 – Color principal de botones

Fuente: (López Yépez & Polo Vásconez, 2016)

En el desarrollo de la interfaz en la aplicación web, se elaboran varias plantillas personalizadas entre las cuales podemos destacar las siguientes:

- **Interfaz de inicio de sesión:** esta interfaz permite a los usuarios acceder al sistema.

Gráfico 30 – Interfaz Inicio de sesión

Fuente: (López Yépez & Polo Vásconez, 2016)

- **Plantilla de página principal:** esta plantilla permite que los usuarios puedan navegar entre las diferentes interfaces de la aplicación web.

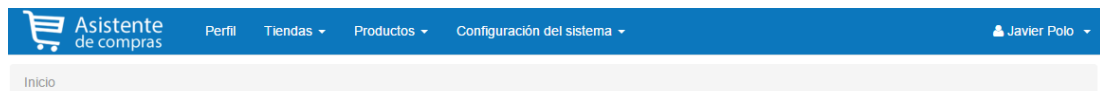


Gráfico 31 – Plantilla de página principal

Fuente: (López Yépez & Polo Vásconez, 2016)

- **Plantilla pantalla responsiva:** se visualiza cuando la resolución de la pantalla es menor a los 800px, es decir en la mayoría de los casos cuando se acceda a la aplicación web desde un teléfono celular, tableta, etc.

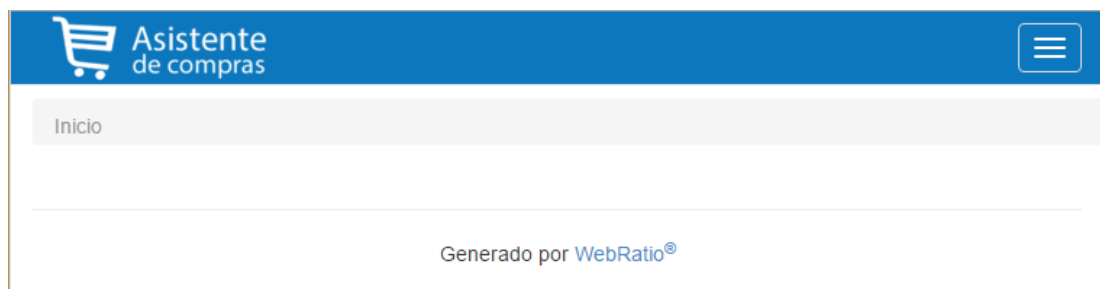


Gráfico 32 – Interfaz pantalla responsiva

Fuente: (López Yépez & Polo Vásconez, 2016)

2.3.3 Iteración 3

2.3.3.1 US004 – Gestión de Tiendas

La gestión de tiendas permite al usuario administrador y responsable de tienda configurar los datos de tiendas afiliadas a las aplicaciones, para esto se definió la estructura de modelado presentada en la siguiente imagen:

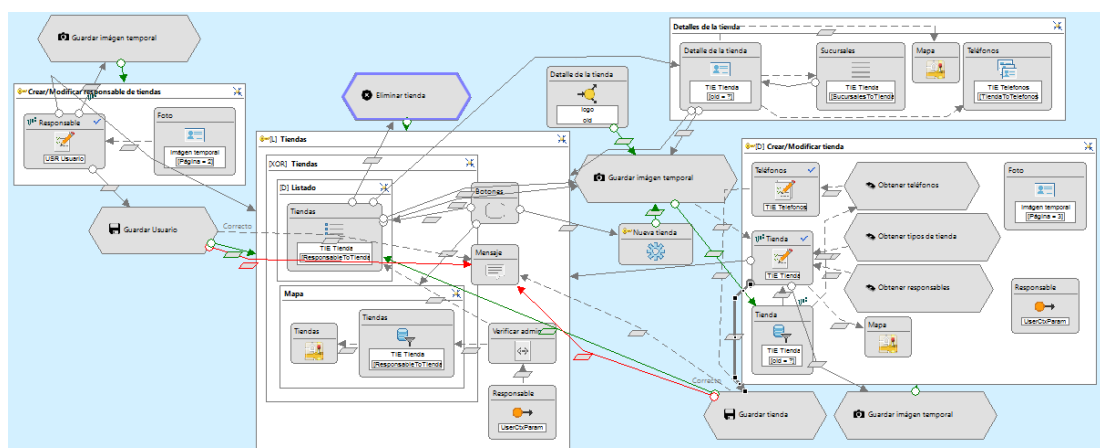


Gráfico 33 – Modelado Gestión de tiendas

Fuente: (López Yépez & Polo Vásconez, 2016)

Esta estructura es representada gráficamente en cada una de sus interfaces web.

2.3.3.1.1 Interfaz tiendas

La interfaz tiendas contiene el listado de tiendas, permite acceder a la creación (solo usuario administrador), modificación, creación de sucursales, detalle y ubicación geográfica de las tiendas ingresadas en el sistema.

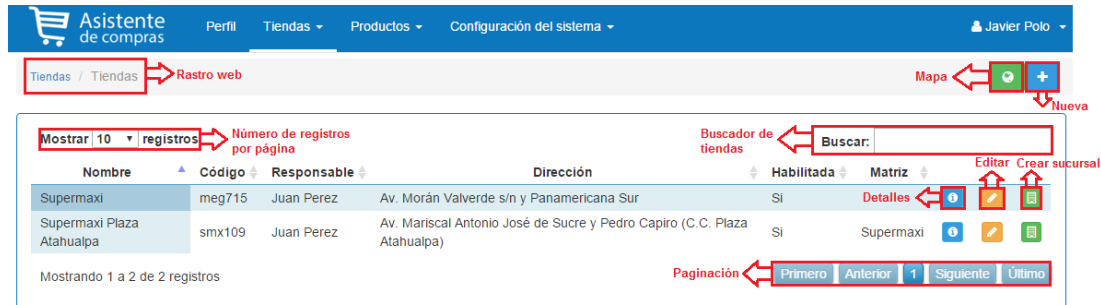


Gráfico 34 – Interfaz tiendas

Fuente: (López Yépez & Polo Vásconez, 2016)

Rastro web: permite al usuario conocer su ubicación dentro de la aplicación web.

Mapa: acceso directo al mapa en donde se puede ubicar las tiendas geográficamente.

Nueva: acceso directo al formulario de “Crear/Editar tienda”

Número de registros por página: permite al usuario modificar el número de registros que se muestran en pantalla (por defecto 10).

Buscador de tiendas: campo que permite al usuario buscar tiendas por todas las columnas del listado.

Detalles: acceso directo a la página “Detalles de la tienda”.

Editar: acceso directo al formulario de “Crear/Editar tienda”.

Crear sucursal: acceso directo al formulario de “Crear/Editar tienda” precargando el valor de la tienda matriz seleccionada.

Paginación: permite al usuario navegar por las diferentes páginas creadas para dividir la cantidad de registros que se muestran en pantalla, cuando estos superan el límite de registros por página (por defecto 10).

2.3.3.1.2 Detalles de la tienda

La interfaz detalles de la tienda, permite al usuario visualizar los datos de configuración de cada una de las tiendas existentes en el sistema.

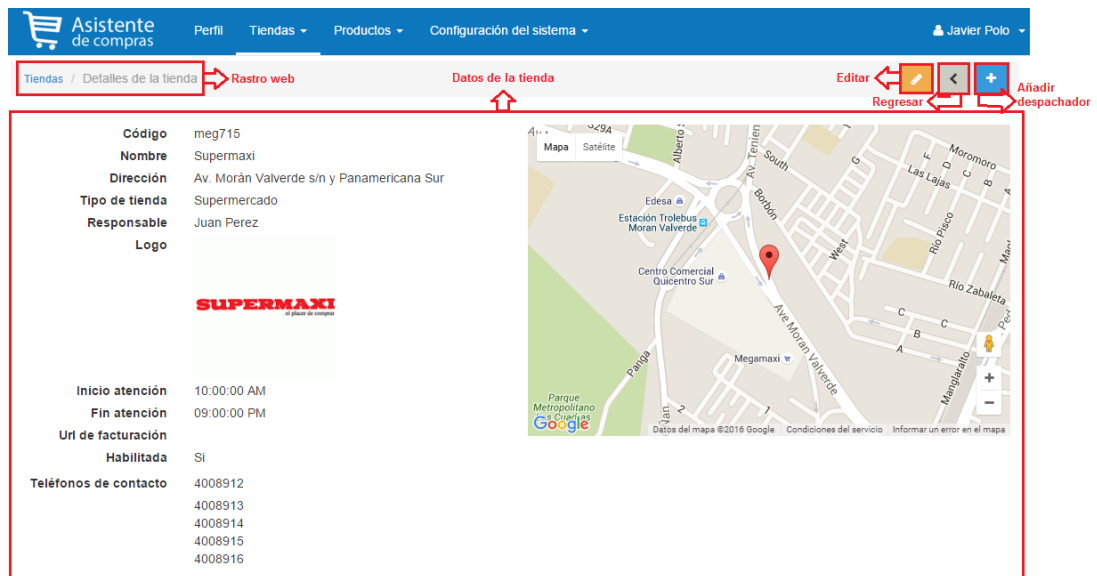


Gráfico 35 – Interfaz Detalles de la tienda

Fuente: (López Yépez & Polo Vásconez, 2016)

Rastro web: permite al usuario conocer su ubicación dentro de la aplicación web.

Editar: acceso directo al formulario de “Crear/Editar tienda”.

Regresar: acceso directo al Listado de tiendas.

Añadir despachador: acceso directo al formulario “Crear/Editar despachador”.

Datos de la tienda: permite al usuario visualizar los datos de configuración de una tienda.

2.3.3.1.3 Crear/Editar tienda

La interfaz Crear/Editar tienda permite al usuario administrador configurar los datos de una nueva tienda o configurar una tienda existente en el sistema.

Asistente de compras

Perfil Tiendas Productos Configuración del sistema

Javier Polo

Tiendas / Crear/Editar tienda Rastro web Datos de la tienda Guardar Regresar

Código * meg715

Nombre * Supermaxi

Dirección * Av. Morán Valverde s/n y Panamericana Sur

Tipo de tienda * Supermercado

Responsable * Juan Perez

Logo * descarga.png Elegir archivo

Google Maps: Quito, Ecuador

Inicio atención * 10:00:00 AM

Fin atención * 09:00:00 PM

Url de facturación

Teléfonos de contacto *

4008912

4008913

Gráfico 36 – Interfaz Crear/Editar usuario

Fuente: (López Yépez & Polo Vásconez, 2016)

Rastro web: permite al usuario conocer su ubicación dentro de la aplicación web.

Guardar: permite al usuario almacenar los datos del formulario “Crear/Editar tienda”.

Regresar: acceso directo a la Listado de tiendas.

Datos de la tienda: permite al usuario visualizar los datos de configuración de una tienda.

2.3.3.1.3 Interfaz mapa de tiendas

La interfaz mapa de tiendas permite al usuario visualizar la ubicación geográfica de las tiendas afiliadas.

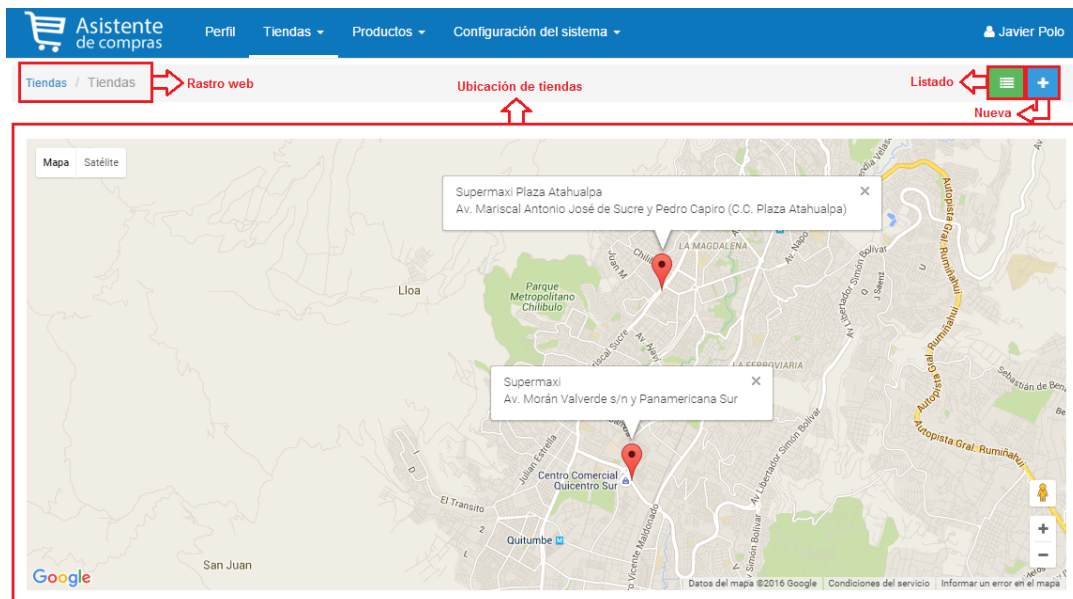


Gráfico 37 – Interfaz Crear/Editar usuario

Fuente: (López Yépez & Polo Vásconez, 2016)

Rastro web: permite al usuario conocer su ubicación dentro de la aplicación web.

Listado: acceso directo a la Listado de tiendas.

Nueva: acceso directo al formulario de “Crear/Editar tienda”

Ubicación de tiendas: permite al usuario visualizar en el mapa los datos de las tiendas afiliadas.

2.3.3.2 US005 – Gestión de Responsables de tiendas

La interfaz “Crear/Editar usuario” permite al usuario administrador configurar los datos de un usuario con rol “Responsable”, para posteriormente asignarlo como responsable de una tienda en la interfaz “Crear/Editar tienda”, tal como se puede visualizar en la siguiente imagen:

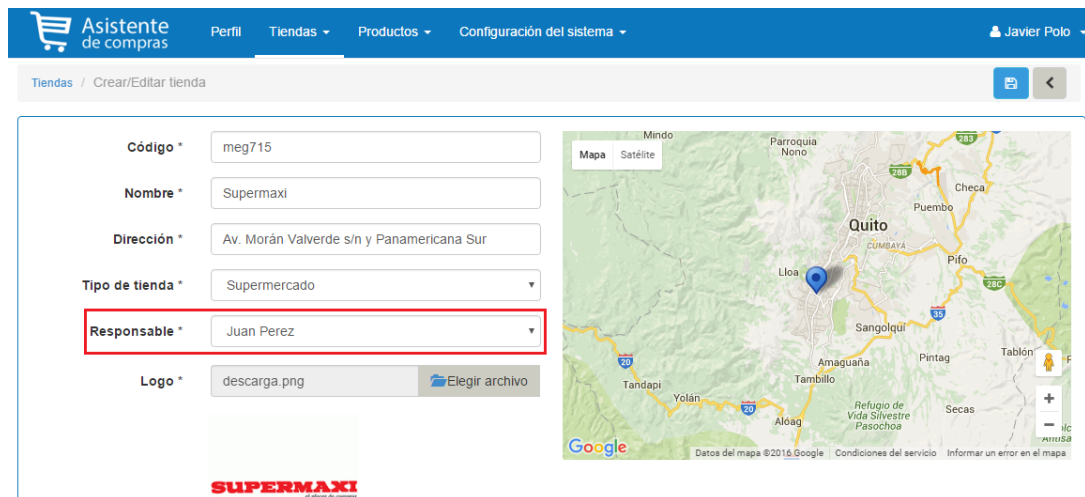


Gráfico 38 – Asignar responsable a tienda

Fuente: (López Yépez & Polo Vásconez, 2016)

2.3.3.3 US006 – Gestión de categorías de productos

La gestión de categorías de productos permite al usuario administrador configurar los datos de las categorías para los productos que serán gestionados por la aplicación web, para esto se definió la estructura de modelado presentada en la siguiente imagen:

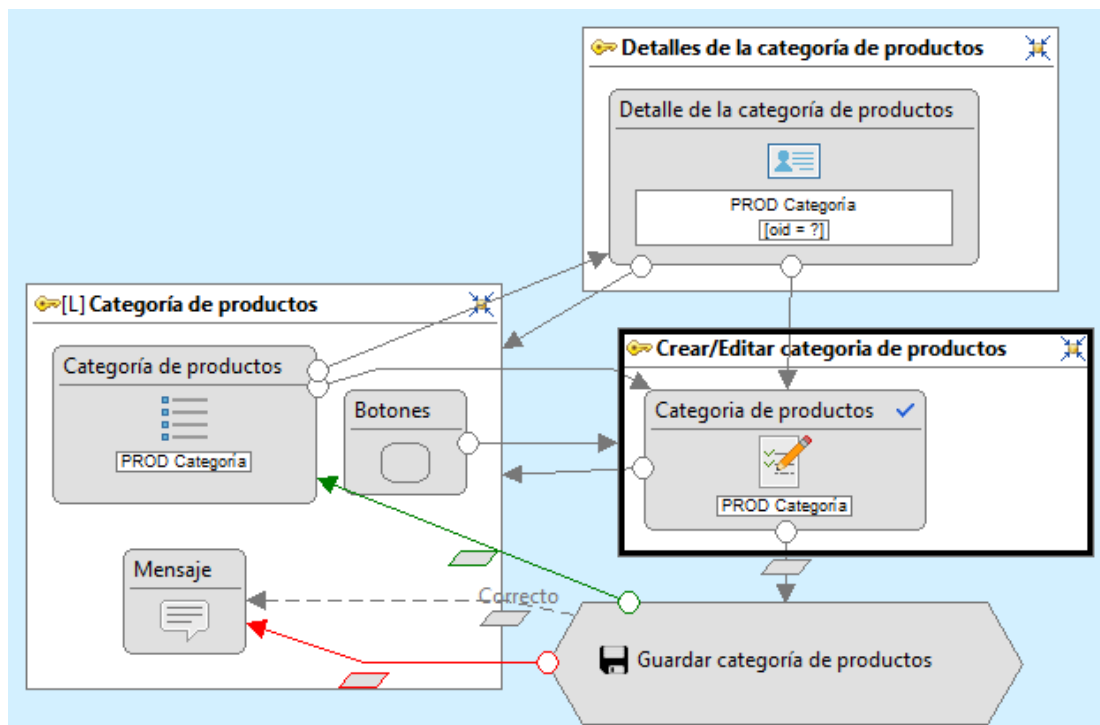


Gráfico 39 – Modelado Gestión de categorías de productos

Fuente: (López Yépez & Polo Vásconez, 2016)

Esta estructura es representada gráficamente en cada una de sus interfaces web.

2.3.3.3.1 Interfaz categoría de productos

La interfaz categoría de productos contiene el listado de categorías de productos existentes en el sistema, permite acceder a la creación, modificación, y detalle de dichas categorías.

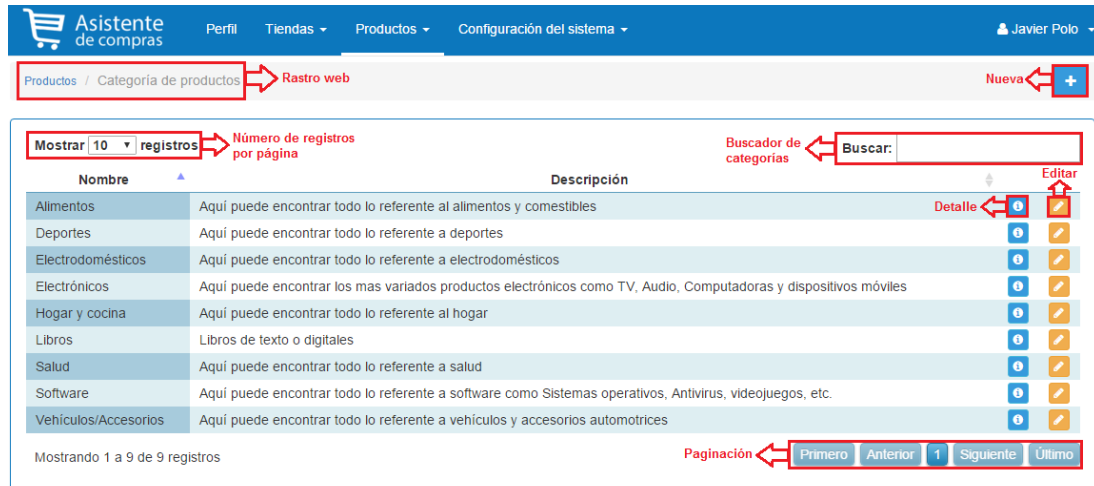


Gráfico 40 – Interfaz categoría de productos

Fuente: (López Yépez & Polo Vásconez, 2016)

Rastro web: permite al usuario conocer su ubicación dentro de la aplicación web.

Nueva: acceso directo al formulario de “Crear/Editar categoría de productos”

Número de registros por página: permite al usuario modificar el número de registros que se muestran en pantalla (por defecto 10).

Buscador de categorías: campo que permite al usuario buscar categorías de productos por todas las columnas del listado.

Detalles: acceso directo a la página “Detalles de la categoría de productos”.

Editar: acceso directo al formulario de “Crear/Editar categoría de productos”.

Paginación: la paginación permite al usuario navegar por las diferentes páginas creadas para dividir la cantidad de registros que se muestran en pantalla, cuando estos superan el límite de registros por página (por defecto 10).

2.3.3.3.2 Detalles de la categoría de productos

La interfaz Detalles de la categoría de productos, permite al usuario visualizar los datos de configuración de cada una de las categorías de productos existentes en el sistema.

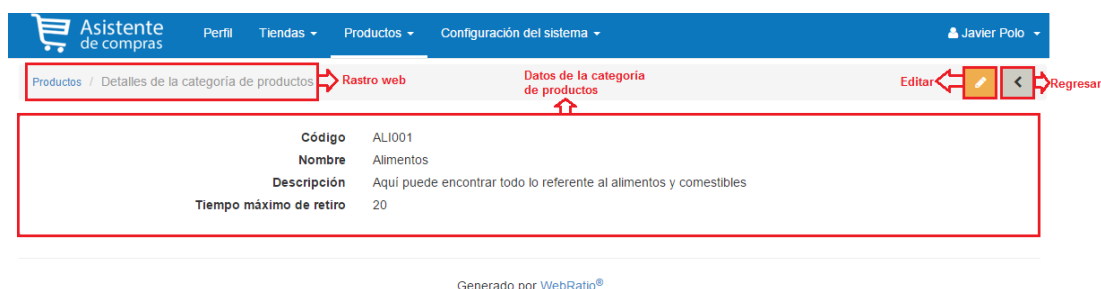


Gráfico 41 – Interfaz Detalles de la categoría de productos

Fuente: (López Yépez & Polo Vásconez, 2016)

Rastro web: permite al usuario conocer su ubicación dentro de la aplicación web.

Editar: acceso directo al formulario de “Crear/Editar categoría de productos”.

Regresar: acceso directo al Listado de Categorías de productos.

Datos de la categoría de productos: permite al usuario visualizar los datos de configuración de una categoría de productos.

2.3.3.3.3 Crear/Editar categoría de productos

La interfaz Crear/Editar categoría de productos permite al usuario administrador configurar los datos de una nueva categoría de productos o una previamente ingresada en el sistema.

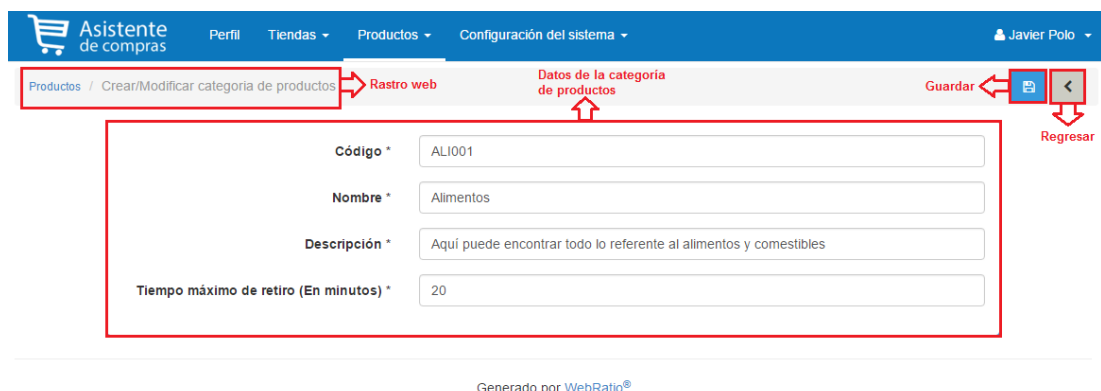


Gráfico 42 – Interfaz Crear/Editar categoría de productos

Fuente: (López Yépez & Polo Vásconez, 2016)

Rastro web: permite al usuario conocer su ubicación dentro de la aplicación web.

Guardar: permite al usuario almacenar los datos del formulario “Crear/Editar categoría de productos”.

Regresar: acceso directo al Listado de Categorías de productos.

Datos de la categoría de productos: permite al usuario visualizar los datos de configuración de una categoría de productos.

2.3.3.4 US007 – Gestión de productos

La gestión de productos permite al usuario responsable de tienda configurar los datos de los productos que serán gestionados por la aplicación web, para esto se definió la estructura de modelado presentada en la siguiente imagen:

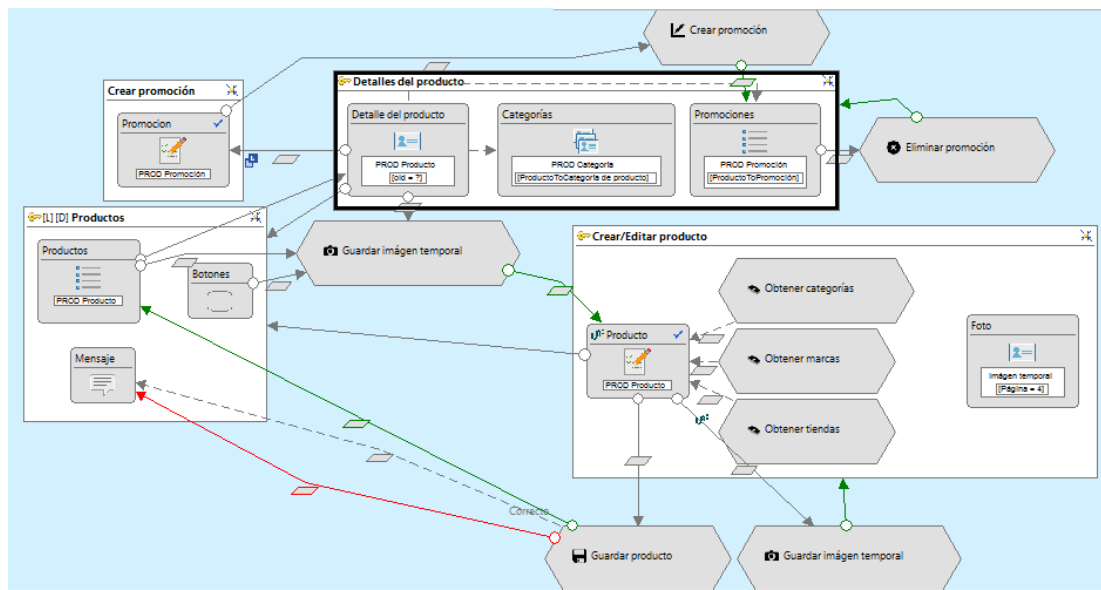


Gráfico 43 – Modelado Gestión de productos

Fuente: (López Yépez & Polo Vásconez, 2016)

Esta estructura es representada gráficamente en cada una de sus interfaces web.

2.3.3.4.1 Interfaz productos

La interfaz productos contiene el listado de productos existentes en el sistema, permite acceder a la creación, modificación, y detalle de dichos productos.

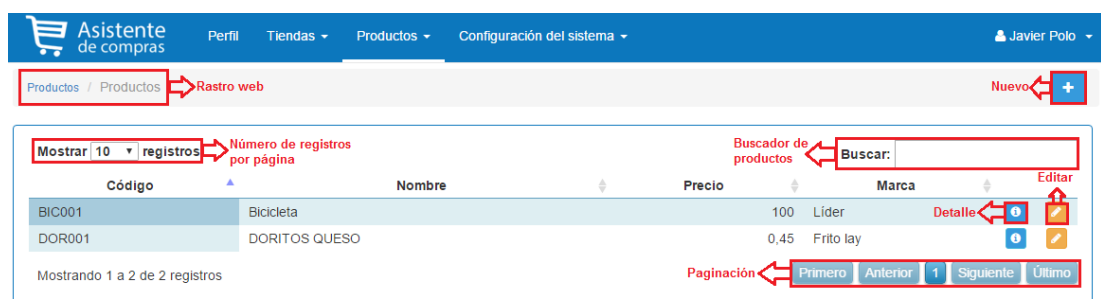


Gráfico 44 – Interfaz productos

Fuente: (López Yépez & Polo Vásconez, 2016)

Rastro web: permite al usuario conocer su ubicación dentro de la aplicación web.

Nuevo: acceso directo al formulario de “Crear/Editar producto”

Número de registros por página: permite al usuario modificar el número de registros que se muestran en pantalla (por defecto 10).

Buscador de productos: campo que permite al usuario buscar productos por todas las columnas del listado.

Detalles: acceso directo a la página “Detalles del producto”.

Editar: acceso directo al formulario de “Crear/Editar producto”.

Paginación: la paginación permite al usuario navegar por las diferentes páginas creadas para dividir la cantidad de registros que se muestran en pantalla, cuando estos superan el límite de registros por página (por defecto 10).

2.3.3.4.2 Detalles del producto

La interfaz detalles del producto, permite al usuario visualizar los datos de configuración de cada uno de los productos existentes en el sistema.



Gráfico 45 – Interfaz detalles del producto

Fuente: (López Yépez & Polo Vásconez, 2016)

Rastro web: permite al usuario conocer su ubicación dentro de la aplicación web.

Editar: acceso directo al formulario de “Crear/Editar producto”.

Promoción: acceso directo al formulario “Crear promoción”

Regresar: acceso directo al Listado de productos.

Datos del producto: permite al usuario visualizar los datos de configuración de un producto.

2.3.3.4.3 Crear/Editar producto

La interfaz Crear/Editar producto permite al usuario responsable de tienda configurar los datos de los productos que los usuarios con rol cliente podrán ordenar desde la aplicación.

Gráfico 46 – Interfaz Crear/Editar producto

Fuente: (López Yépez & Polo Vásconez, 2016)

Rastro web: permite al usuario conocer su ubicación dentro de la aplicación web.

Guardar: permite al usuario almacenar los datos del formulario “Crear/Editar producto”.

Regresar: acceso directo al Listado de productos.

Datos del producto: permite al usuario visualizar los datos de configuración de un producto.

2.3.4 Iteración 4

2.3.4.1 US008 – Búsqueda de productos

La búsqueda de productos permite al usuario un atajo para ubicar productos en la aplicación, para esto se definió la estructura de modelado presentada en la siguiente imagen:

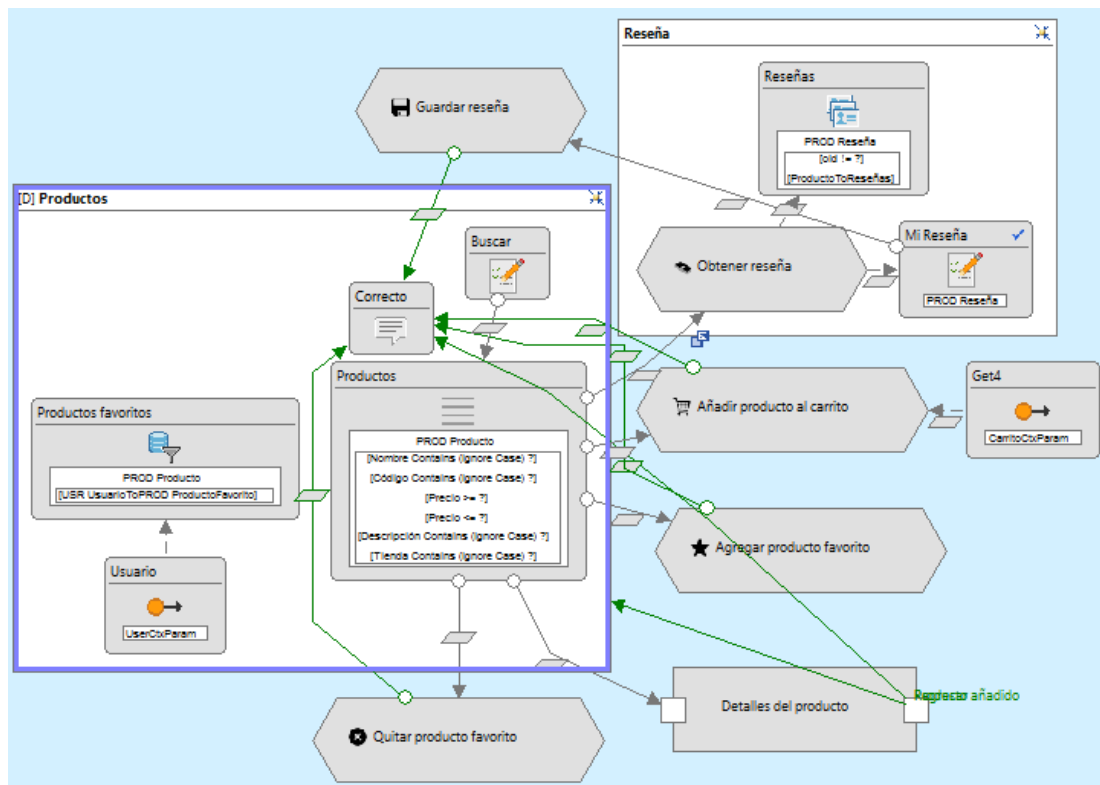


Gráfico 47 – Modelado búsqueda de productos

Fuente: (López Yépez & Polo Vásconez, 2016)

La interfaz búsqueda de productos contiene un listado de productos existentes en el sistema, desde donde se puede realizar reseñas, visualizar el detalle de un producto, marcar productos como favoritos y añadir productos al carrito de compras.



Gráfico 48 – Interfaz búsqueda de productos

Fuente: (López Yépez & Polo Vásconez, 2016)

Rastro web: permite al usuario conocer su ubicación dentro de la aplicación web.

Buscador de productos: campo que permite al usuario buscar productos por nombre, tienda, descripción y categoría.

Detalles: acceso directo a la página “Detalles del producto”.

2.3.4.2 US0011 – Productos y tiendas favoritas

Desde el formulario “búsqueda de productos” se le permite al usuario marcar productos y tiendas como favoritos con el fin de recibir notificaciones sobre promociones, para esto se definió la estructura de modelado presentada en la siguiente imagen:

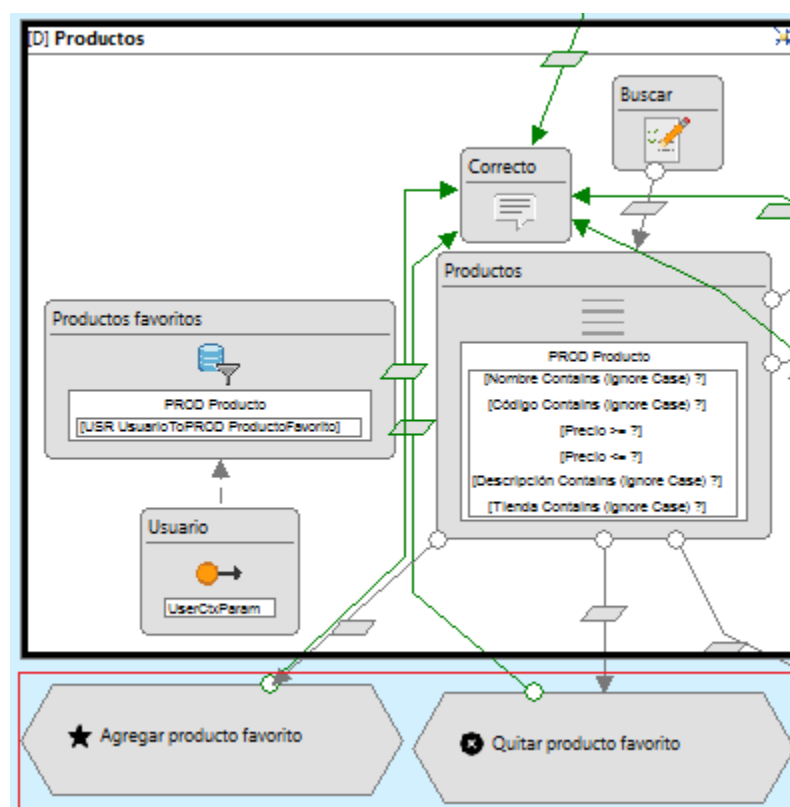


Gráfico 49 – Modelado productos y tiendas favoritas

Fuente: (López Yépez & Polo Vásconez, 2016)

Cuando el usuario marca un producto como favorito visualiza el icono de marcado de color rojo.

Extreme Contact DW



Llantas Extreme Contact DW

Erco tires

\$120

☆☆☆☆☆ (0)



Gráfico 50 – Ejemplo de producto marcado como favorito

Fuente: (López Yépez & Polo Vásconez, 2016)

Un producto no marcado como favorito visualiza el icono de color verde.

Extreme Contact DW



Llantas Extreme Contact DW

Erco tires

\$120

☆☆☆☆☆ (0)



Gráfico 51 – Ejemplo de producto sin marcar como favorito

Fuente: (López Yépez & Polo Vásconez, 2016)

2.3.5 Iteración 5

2.3.5.1 US0013 – Reseñas

Desde el formulario “búsqueda de productos” se le permite al usuario realizar reseñas sobre los productos, para esto se definió la estructura de modelado presentada en la siguiente imagen:

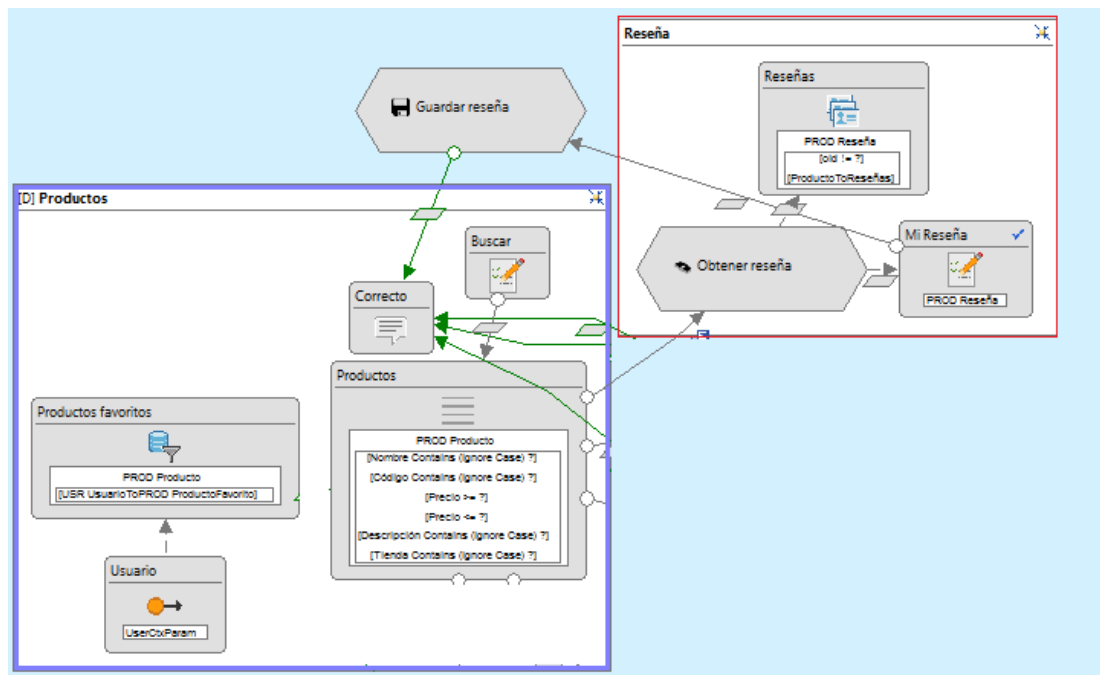


Gráfico 52 – Modelado reseñas

Fuente: (López Yépez & Polo Vásconez, 2016)

El usuario registra la reseña sobre un producto, al tiempo que visualiza las reseñas de los demás usuarios, como se muestra en la imagen:



Gráfico 53 – Ejemplo de crear reseña

Fuente: (López Yépez & Polo Vásconez, 2016)

A los usuarios se les permite crear una reseña por producto.

2.3.5.2 US009 – Carrito de compras

Desde el formulario “búsqueda de productos” se le permite al usuario añadir productos al carrito de compras. Los productos que se añadan al carrito de compras se visualizaran en el listado de productos del carrito, para esto se definió la estructura de modelado presentada en la siguiente imagen:

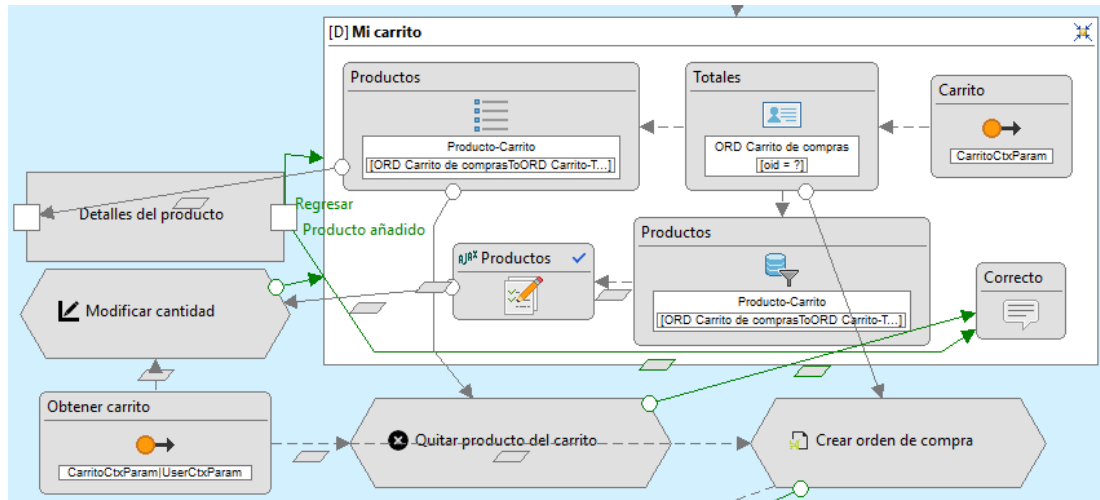


Gráfico 54 – Modelado carrito de compras

Fuente: (López Yépez & Polo Vásconez, 2016)

La Interfaz carrito de compras permite al usuario modificar la cantidad de productos a adquirir, acceder al detalle del producto y remover productos del carrito de compras:

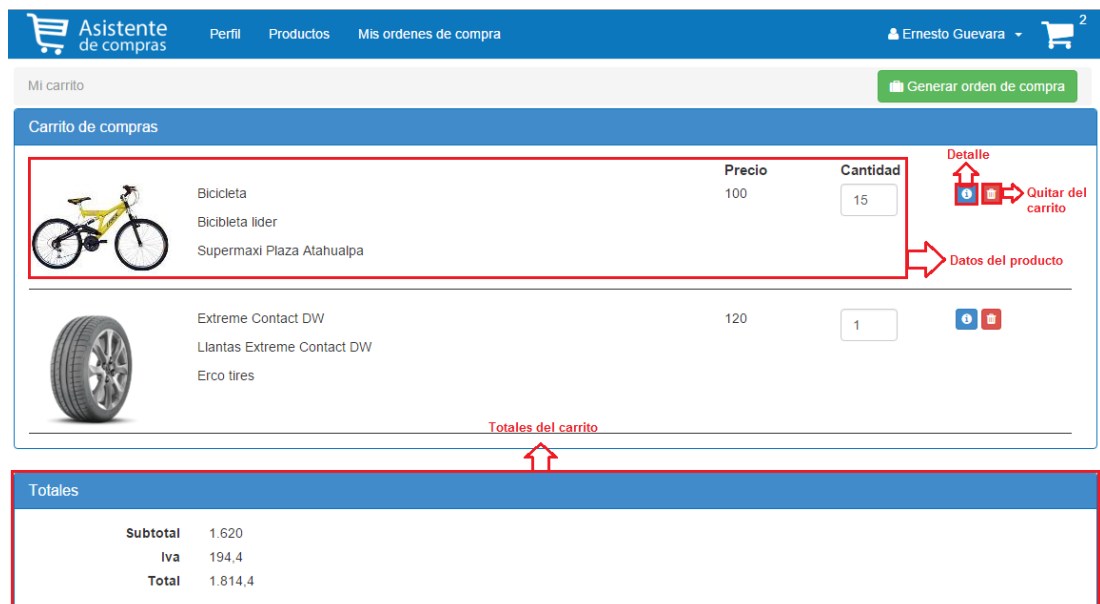


Gráfico 55 – Interfaz carrito de compras

Fuente: (López Yépez & Polo Vásconez, 2016)

Datos del producto: permite al usuario visualizar los datos del producto que está en el carrito de compras.

Detalle: acceso directo al formulario “Detalles del producto”.

Quitar del carrito: permite al usuario quitar productos del carrito de compras.

Totales del carrito: permite al usuario visualizar el total que pagaría por la orden de compra.

2.3.6 Iteración 6

2.3.6.1 US010 – Orden de compra

Desde el formulario “Mis órdenes de compra” se le permite al usuario visualizar las órdenes de compra que han sido emitidas por él, para esto se definió la estructura de modelado presentada en la siguiente imagen:

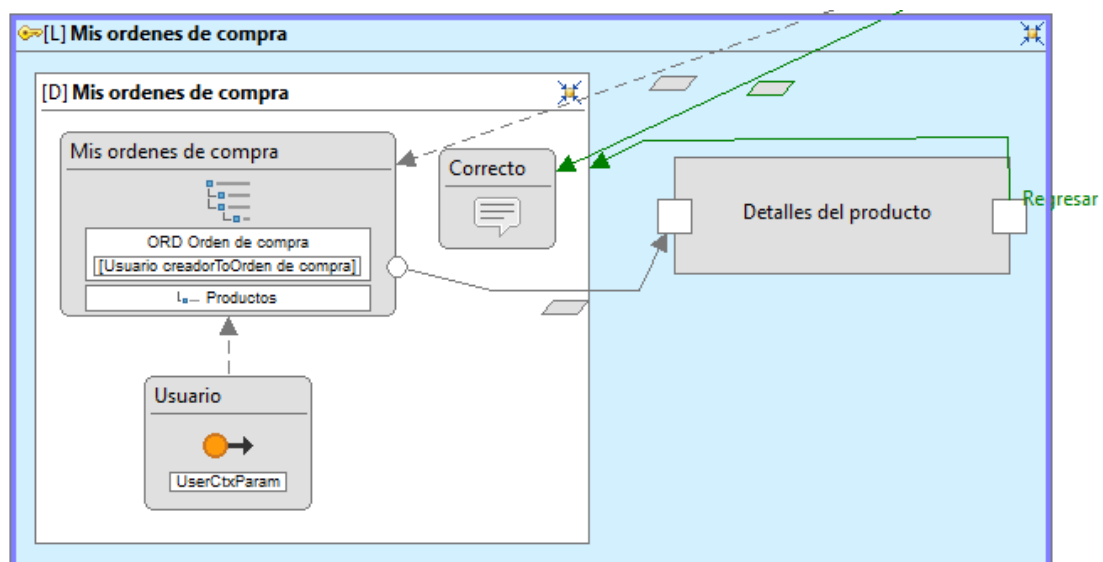


Gráfico 56 – Modelado mis órdenes de compra

Fuente: (López Yépez & Polo Vásconez, 2016)

La interfaz mis órdenes de compra permite al usuario visualizar el estado de sus órdenes de compra, así como el total a pagar por cada una de las tiendas:

CAPÍTULO 3: APLICACIÓN MÓVIL CON WEBRATIO

Introducción

En este capítulo se definen los requisitos y los casos de uso que se involucran con el desarrollo de la aplicación móvil. Además, se explica el resultado de cada una de las historias de usuario desarrolladas, el modelado del caso de uso y su resultado en la interfaz móvil.

3.1 Análisis de requisitos del sistema

El “Asistente de compras” es una aplicación móvil que permitirá el control de solicitudes de compra para productos en tiendas que se encuentren registradas en el sistema.

Luego de visitar tiendas en línea como (Amazon, 2015), (OLX, 2015) y (Mercado Libre, 2015) hemos determinado que es necesario contar con:

- 2.5 Búsqueda de productos
- 2.6 Registro de usuarios
- 2.7 Roles de usuario
- 2.7.4 Cliente
- 2.8 Órdenes de compra
- 2.9 Carrito de compras
- 2.11 Productos y tiendas Favoritas
- 2.13 Envío de notificaciones
- 2.14 Reseñas

Debido a que la especificación de requisitos para la aplicación web es similar para la móvil, se puede observar el detalle en el [anexo 2](#)

3.2 Historias de usuario

Las historias de usuario necesarias para el desarrollo de la aplicación móvil nombrada “Asistente de compras” se definen a continuación:

- 3.1 US000 – Configuración de ambiente de desarrollo
- 3.2 US001 – Modelo de dominio

- 3.4 US003 – Registro de usuarios
- 3.9 US008 – Búsqueda de productos
- 3.10 US009 – Carrito de compras
- 3.11 US010 – Orden de compra
- 3.12 US011 – Productos y tiendas favoritas
- 3.13 US012 – Notificaciones automáticas por ofertas en las tiendas
- 3.14 US013 – Reseñas
- 3.15 US014 – Interfaz del sistema

Debido a que las historias de usuario son similares en cuanto a especificación de requisitos para la aplicación web, se puede observar el detalle en el [anexo 3](#).

3.3 Desarrollo de las historias de usuario

Posterior a la definición de las historias de usuario se inició con el desarrollo de las mismas, para las cuales se realizó la planificación de los Iteraciones necesarios para finalizar la aplicación web, dichas Iteraciones se encuentra detallados en la lista de tareas del producto que se explican en el [anexo 4](#).

El resultado de cada una de las historias de usuario es el siguiente:

3.3.1 Iteración 1

3.3.1.1 US001 – Modelo de dominio

El modelo de dominio maneja la estructura de los datos que la aplicación móvil necesita, a través del modelo Entidad – Relación y esta descrito en las siguientes imágenes:

3.3.1.1.1 Modelo Entidad – Relación DataService

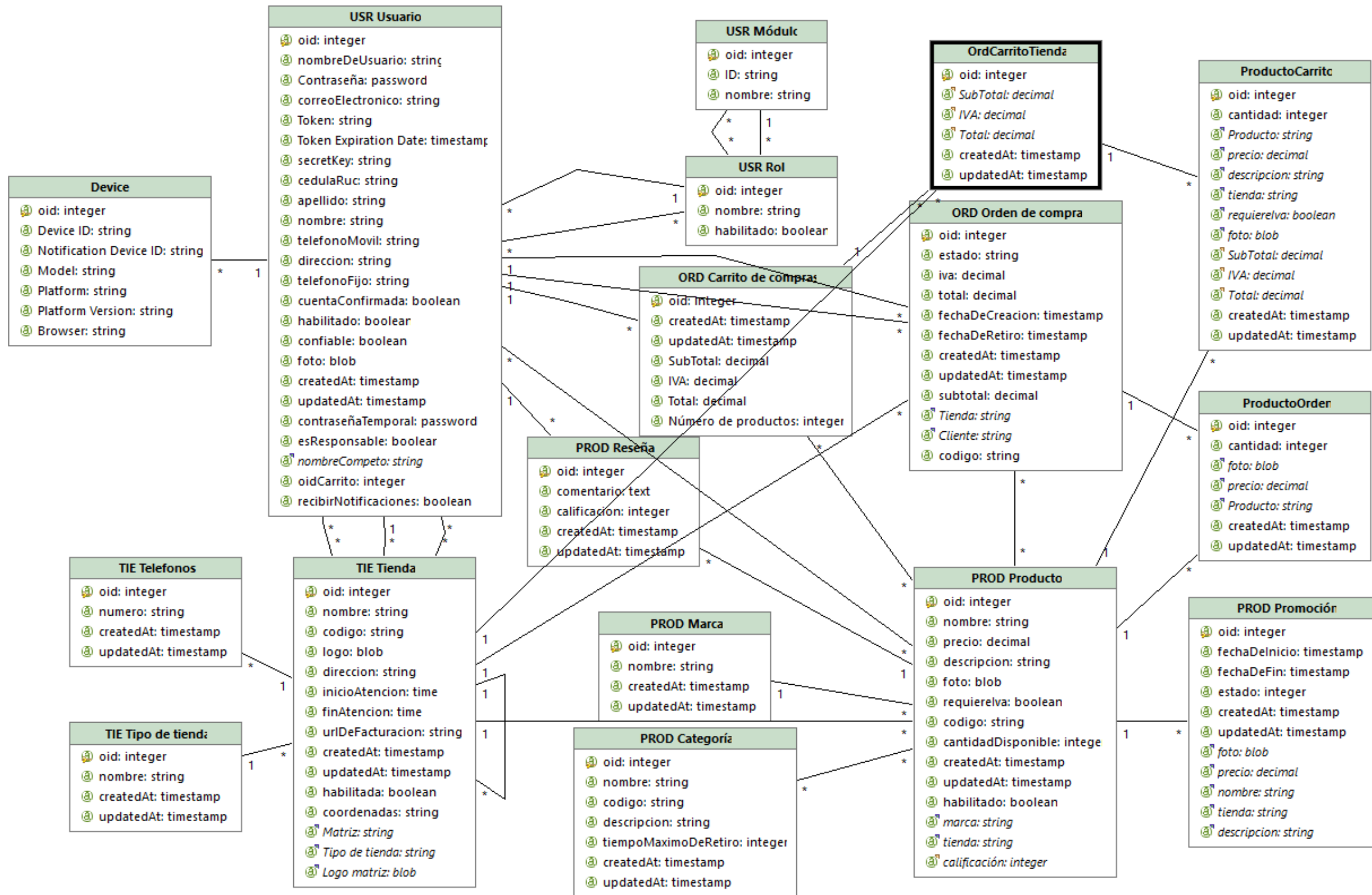


Gráfico 58 – Modelo Entidad – Relación DataService

Fuente: (López Yépez & Polo Vásconez, 2016)

3.3.1.1.2 Modelo Entidad – Relación Aplicación Móvil

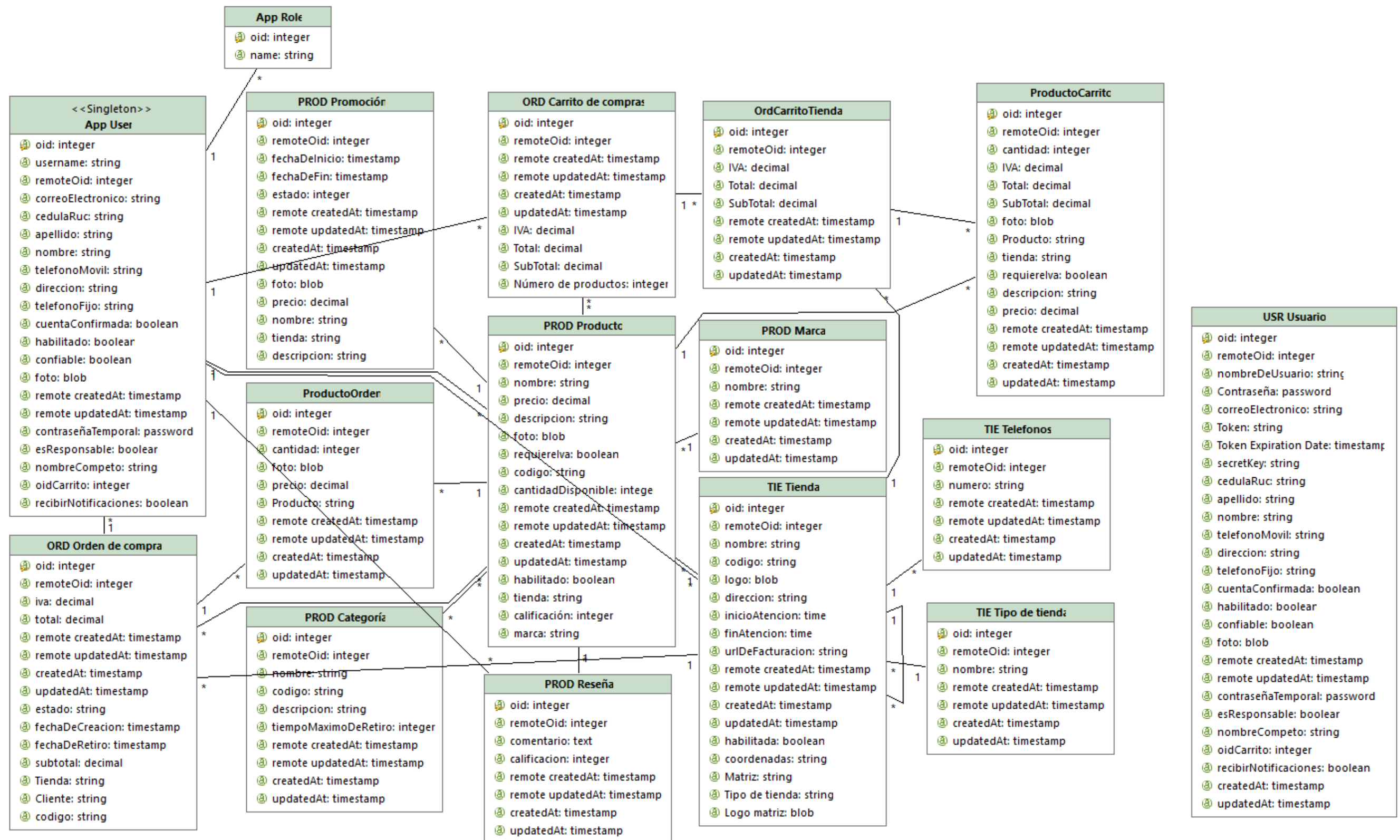


Gráfico 59 – Modelo Entidad – Relación Aplicación Móvil

Fuente: (López Yépez & Polo Vásconez, 2016)

3.3.1.2 US003 – Registro de usuarios

El registro de usuarios permite a nuevos usuarios acceder al sistema y usar todas las funcionalidades definidas para el rol cliente. Para su manejo se utilizó la estructura creada para la aplicación web, y el acceso a los servicios de registro, autenticación, se lo realiza por medio de consultas a servicios web. El modelado IFML para la mencionada sección es el descrito en la siguiente imagen:

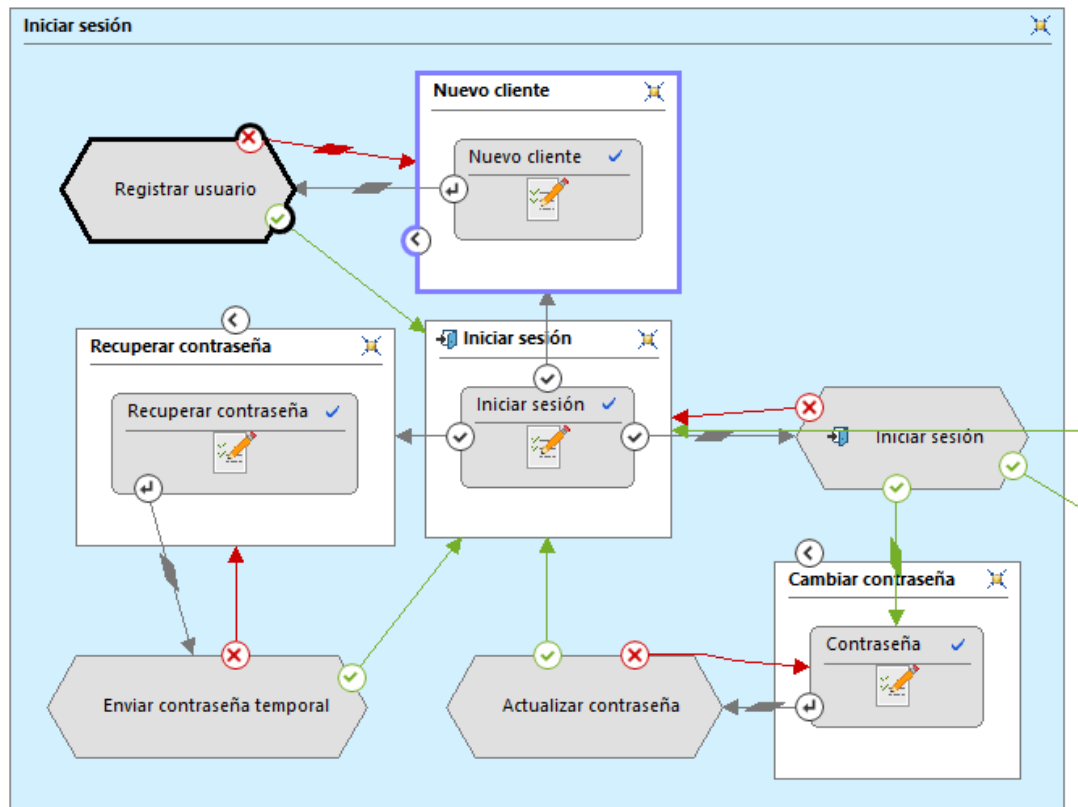


Gráfico 60 – IFML Móvil Pantalla Iniciar sesión

Fuente: (López Yépez & Polo Vásconez, 2016)

Dichas pantallas han sido representadas por las siguientes ilustraciones, las mismas que representan correctamente cada una de las funcionalidades para las cuales fueron desarrolladas.

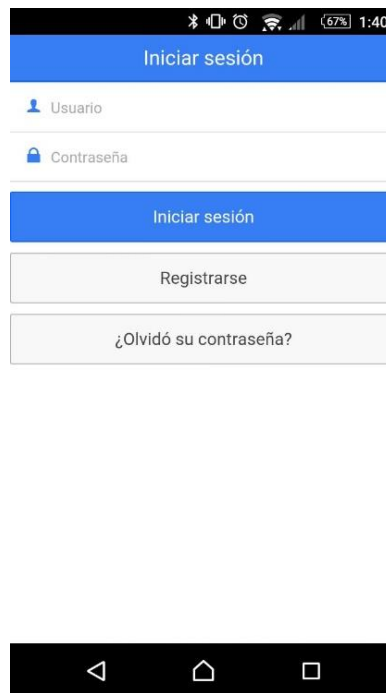


Gráfico 61 – Interfaz inicio de sesión

Fuente: (López Yépez & Polo Vásconez, 2016)

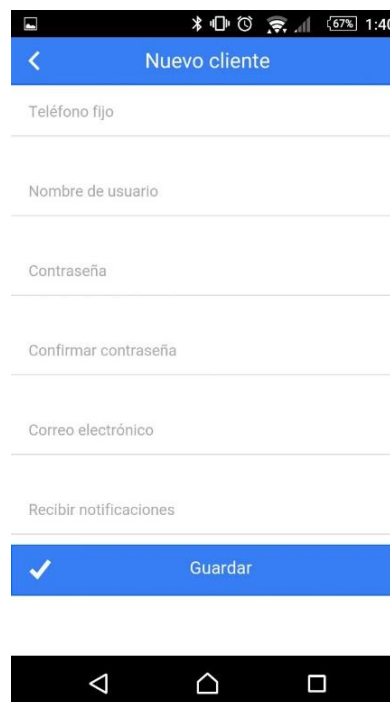


Gráfico 62 – Interfaz nuevo cliente

Fuente: (López Yépez & Polo Vásconez, 2016)

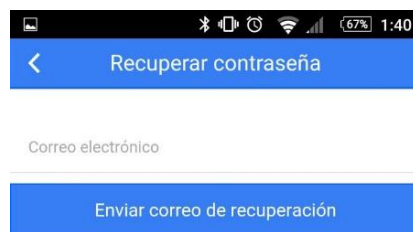


Gráfico 63 – Interfaz recuperar contraseña

Fuente: (López Yépez & Polo Vásconez, 2016)

3.3.2 Iteración 2

3.3.2.1 US014 – Interfaz del sistema

La interfaz del sistema es una parte esencial en el desarrollo de una aplicación, esta historia de usuario nos permite transmitir gráficamente a los usuarios el objetivo principal de la aplicación, para esto se utiliza como base el proyecto de estilos “Partner Network Style” obtenida desde la tienda de WebRatio.

Los elementos principales que se identificaron en el desarrollo de esta historia de usuario son los siguientes:

- **Logo:** el logo de la aplicación permite al usuario identificar a la aplicación distinguiéndola de otras aplicaciones.



Gráfico 64 – Logo

Fuente: (López Yépez & Polo Vásconez, 2016)

- **Color principal:** es el color que se utiliza en la mayoría de los elementos de la interfaz de la aplicación web.

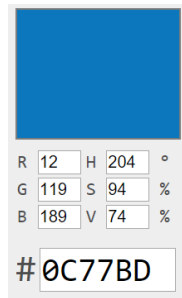


Gráfico 65 – Color principal aplicación móvil

Fuente: (López Yépez & Polo Vásconez, 2016)

- **Fuente tipográfica principal:** la fuente tipográfica representa el tipo de letra usada en toda la aplicación web, en este caso particular usamos el tipo sans-serif, de tamaño 14px.

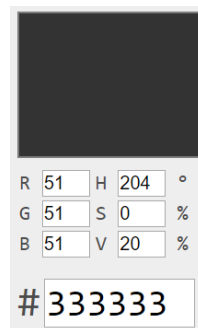


Gráfico 66 – Color fuente principal

Fuente: (López Yépez & Polo Vásconez, 2016)

- **Color principal de botones:** el color principal de los botones nos permite distinguir o resaltar un botón de otro de acuerdo a su importancia dentro de cada uno de los formularios.

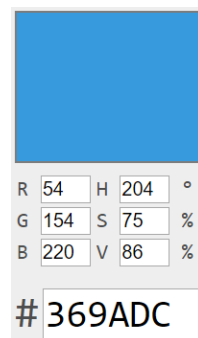


Gráfico 67 – Color principal de botones

Fuente: (López Yépez & Polo Vásconez, 2016)

Dentro del desarrollo de la interfaz en la aplicación móvil, se elaboran plantillas personalizadas, sin embargo, la más importante es la siguiente:

Menú lateral: permite una mejor navegación dentro de los diferentes elementos de la aplicación móvil, es de fácil acceso e intuitiva ya que contiene los accesos directos más relevantes de la aplicación

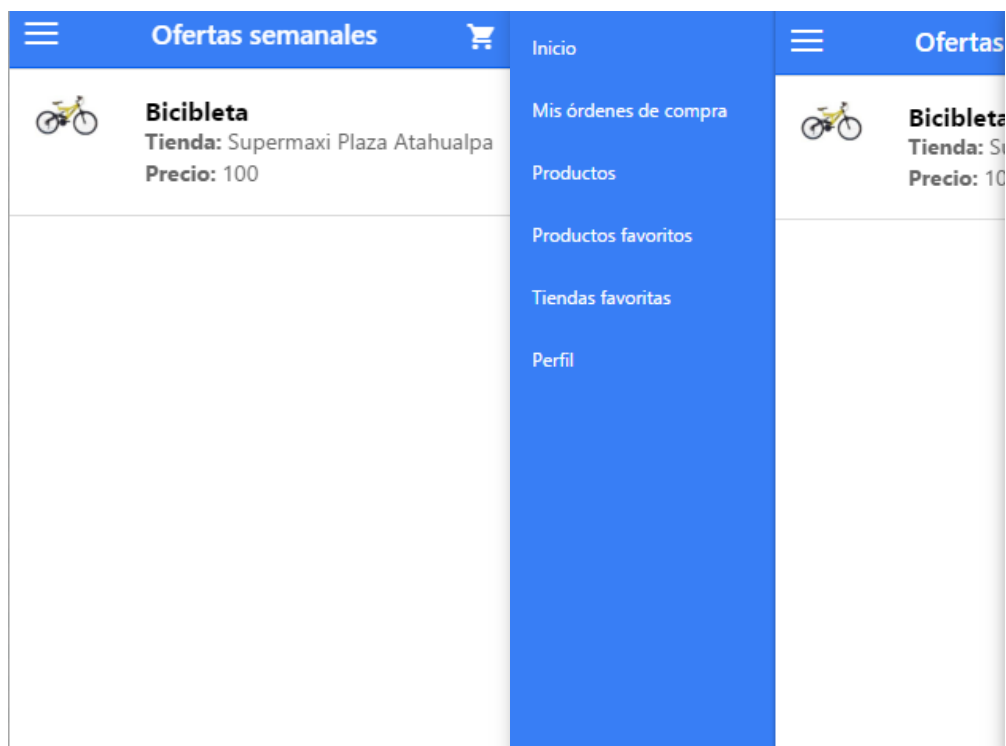


Gráfico 68 – Menú lateral oculto & Menú Lateral desplegado

Fuente: (López Yépez & Polo Vásconez, 2016)

3.3.3 Iteración 3

3.3.3.1 US008 – Búsqueda de productos

La búsqueda de productos permite al usuario visualizar los datos de los productos luego de que se ha realizado una búsqueda sobre el formulario, para esto se define la estructura de modelado presentada en la siguiente imagen:

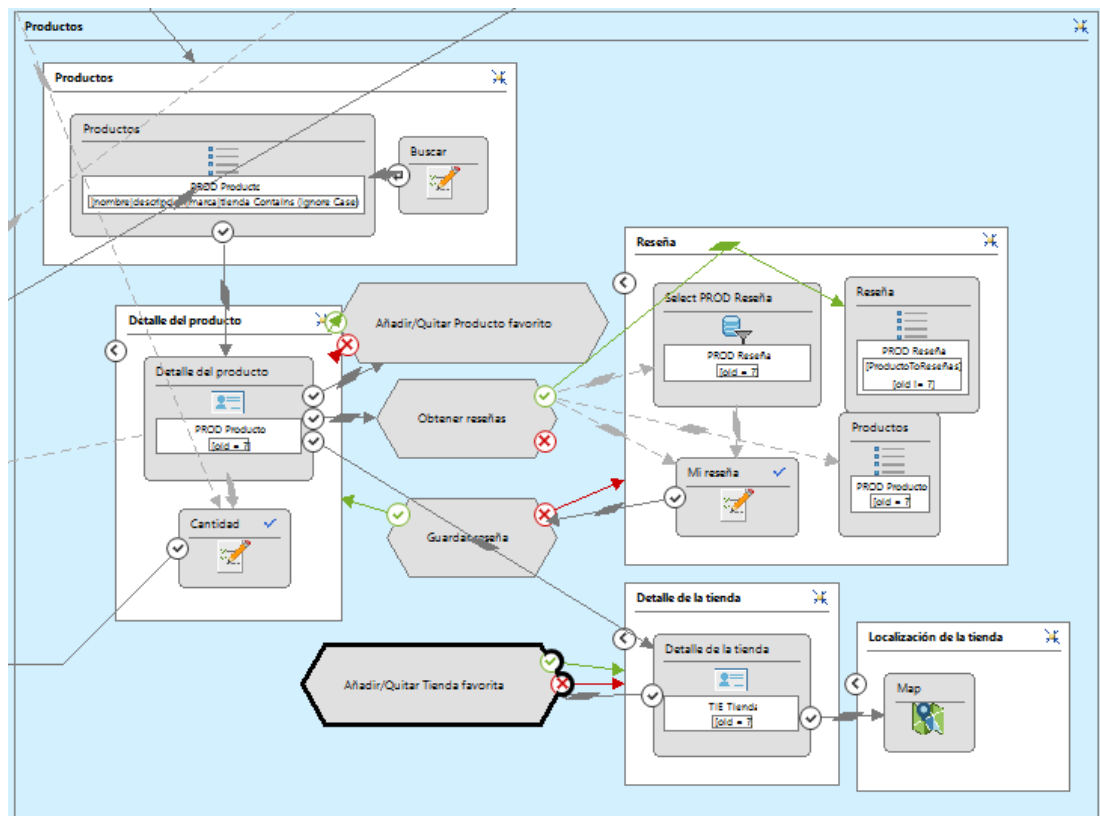


Gráfico 69 – Modelo IFML Sección Productos

Fuente: (López Yépez & Polo Vásconez, 2016)

Los resultados del modelo IFML presentado, permiten contar con un listado de productos que facilitan al usuario interactuar con los diferentes productos disponibles de las varias tiendas, realizar búsquedas por varios criterios y decidir la mejor opción para adquirir, adicional podrán visualizar un detalle del producto e interactuar con el mismo, por ejemplo, agregarlo como favorito o visualizar la tienda que lo está ofertando.

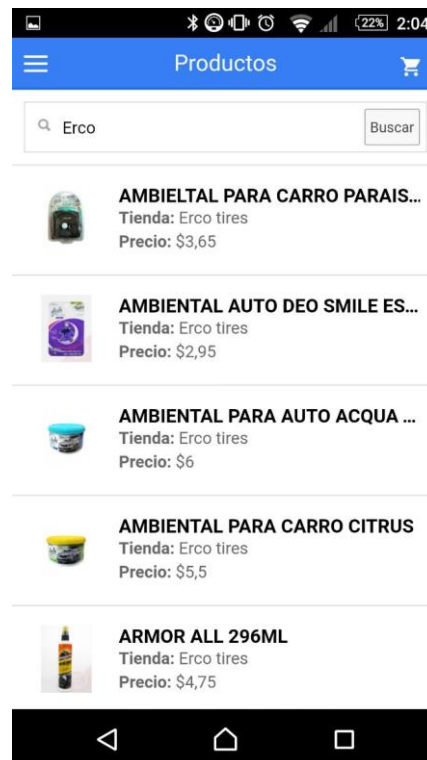


Gráfico 70 – Interfaz búsqueda de productos

Fuente: (López Yépez & Polo Vásconez, 2016)

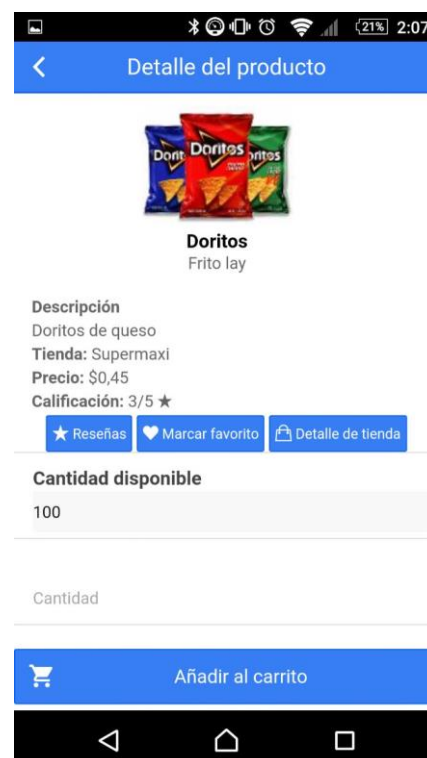


Gráfico 71 – Interfaz detalle de producto

Fuente: (López Yépez & Polo Vásconez, 2016)

3.3.3.2 US011 – Productos y tiendas favoritas

Desde el detalle de los productos, se le permite al usuario marcar productos como favoritos con el fin de recibir notificaciones sobre promociones que se estén ofertando sobre los productos, para esto se definió la estructura de modelado presentada en la siguiente imagen:

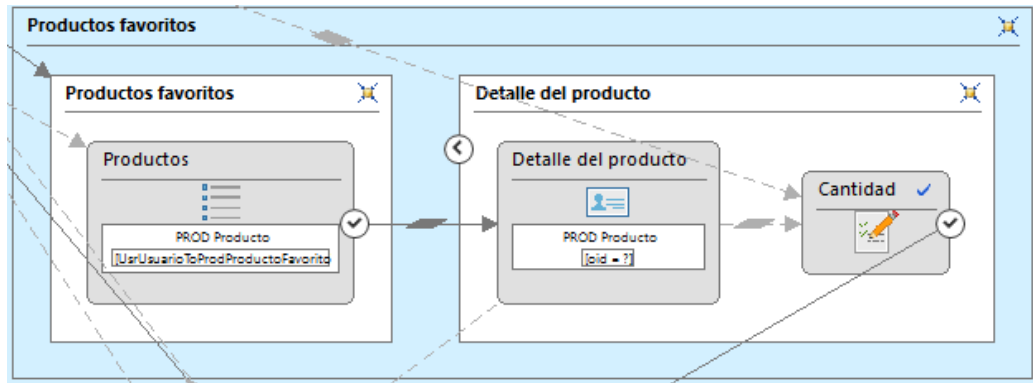


Gráfico 72 – Modelo IFML Productos favoritos

Fuente: (López Yépez & Polo Vásconez, 2016)

Cuando el usuario ha marcado como favorito un producto desde la pantalla de detalles del producto, este podrá acceder rápidamente a sus productos favoritos desde el menú principal.



Gráfico 73 – Interfaz productos favoritos

Fuente: (López Yépez & Polo Vásconez, 2016)

Desde el detalle de los productos, se le permite al usuario acceder al detalle de una tienda, y desde esta pantalla se la puede marcar como favorita, para esto se definió la estructura de modelado presentada en la siguiente imagen:

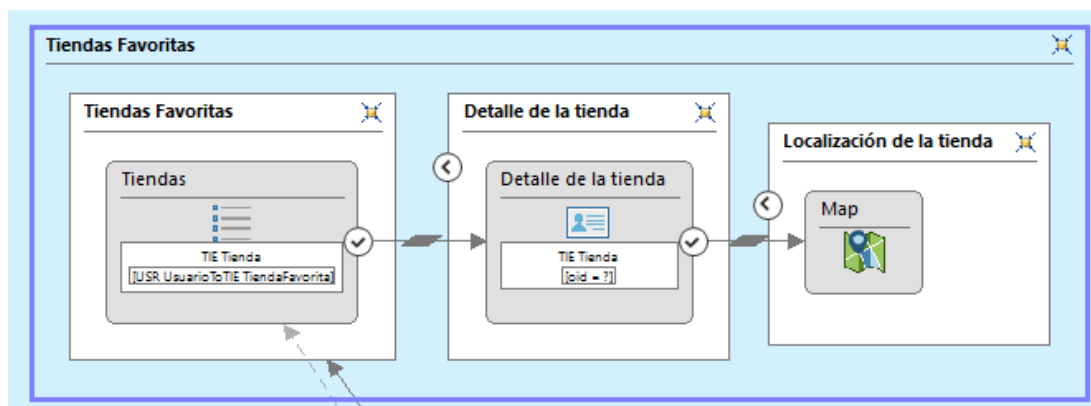


Gráfico 74 – Modelo IFML tiendas favoritas

Fuente: (López Yépez & Polo Vásquez, 2016)

Cuando el usuario ha marcado como favorito una tienda desde la pantalla de detalles de la tienda, este podrá acceder rápidamente a sus tiendas favoritas desde el menú principal.



Gráfico 75 – Interfaz tiendas favoritas

Fuente: (López Yépez & Polo Vásquez, 2016)

Desde el detalle de la tienda podremos ver información básica de la misma y adicional tenemos disponible un enlace para visualizar a la tienda en el mapa.



Gráfico 76 – Interfaz detalle de tienda

Fuente: (López Yépez & Polo Vásconez, 2016)

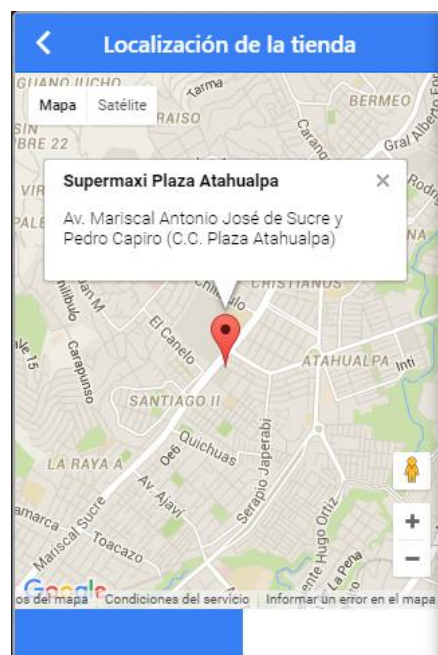


Gráfico 77 – Interfaz Localización de tienda

Fuente: (López Yépez & Polo Vásconez, 2016)

3.3.3.3 US013 – Reseñas

Desde el listado de productos se le permite al usuario acceder al detalle de cada uno de los productos, desde aquí el usuario puede visualizar la calificación del producto

en caso de haberla y puede visualizar mediante un enlace las reseñas sobre los productos y realizar una nueva en caso de requerirlo, para esto se definió la estructura de modelado presentada en la siguiente imagen:

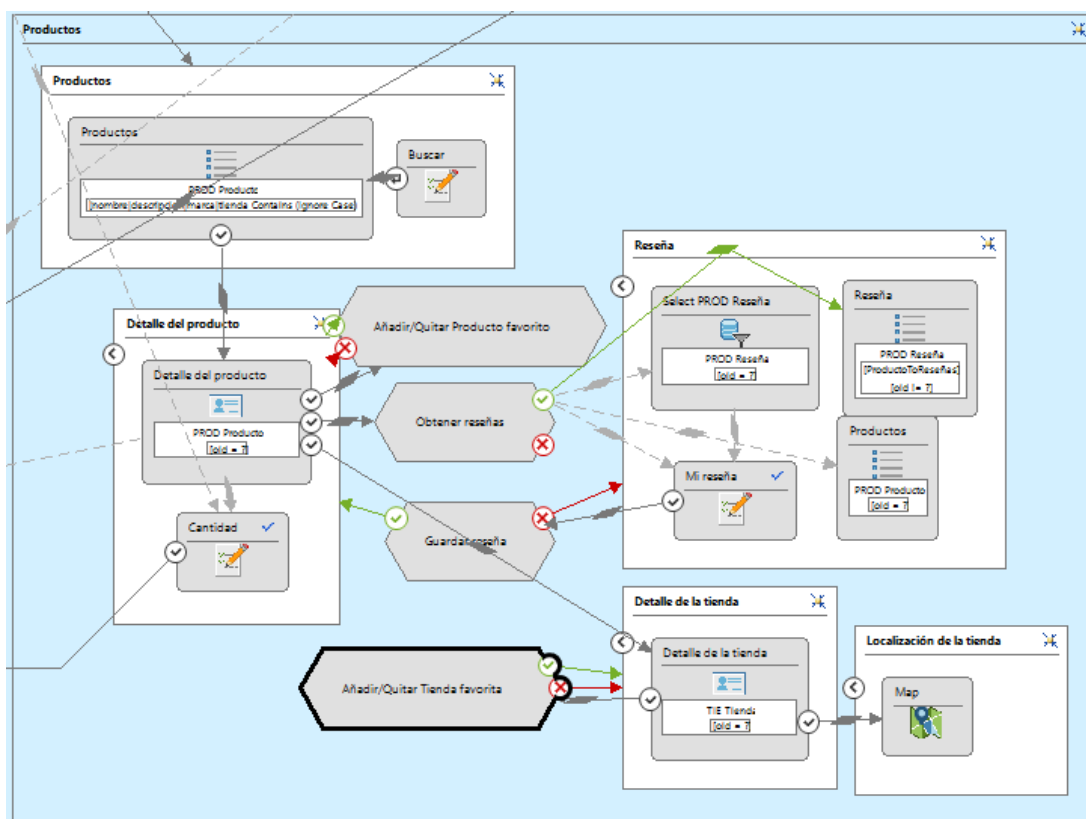


Gráfico 78 – Modelo IFML Productos – Reseñas

Fuente: (López Yépez & Polo Vásconez, 2016)

El resultado del modelo IFML presentado anteriormente nos permite visualizar la siguiente pantalla:

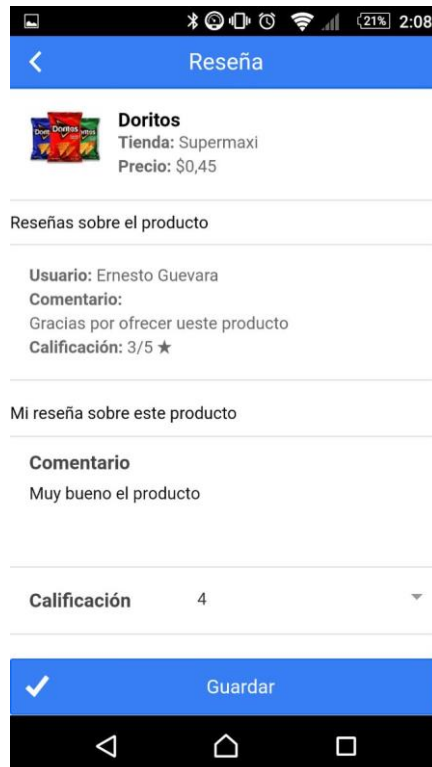


Gráfico 79 – Interfaz listado – ingreso de reseñas

Fuente: (López Yépez & Polo Vásconez, 2016)

Cabe destacar que el usuario únicamente puede realizar una reseña por producto.

3.3.4 Iteración 4

3.3.4.1 US012 – Notificaciones automáticas por ofertas en las tiendas

Las notificaciones automáticas son una parte muy importante de nuestra aplicación móvil, en esta sección se desarrollaron servicios web, que funcionan dependiendo el caso, órdenes de compra o promociones de productos, dichas notificaciones serán enviadas a los usuarios, siempre y cuando estos hayan permitido la recepción de notificaciones, y serán enviadas vía email y mediante el evento *push* en dispositivos móviles.

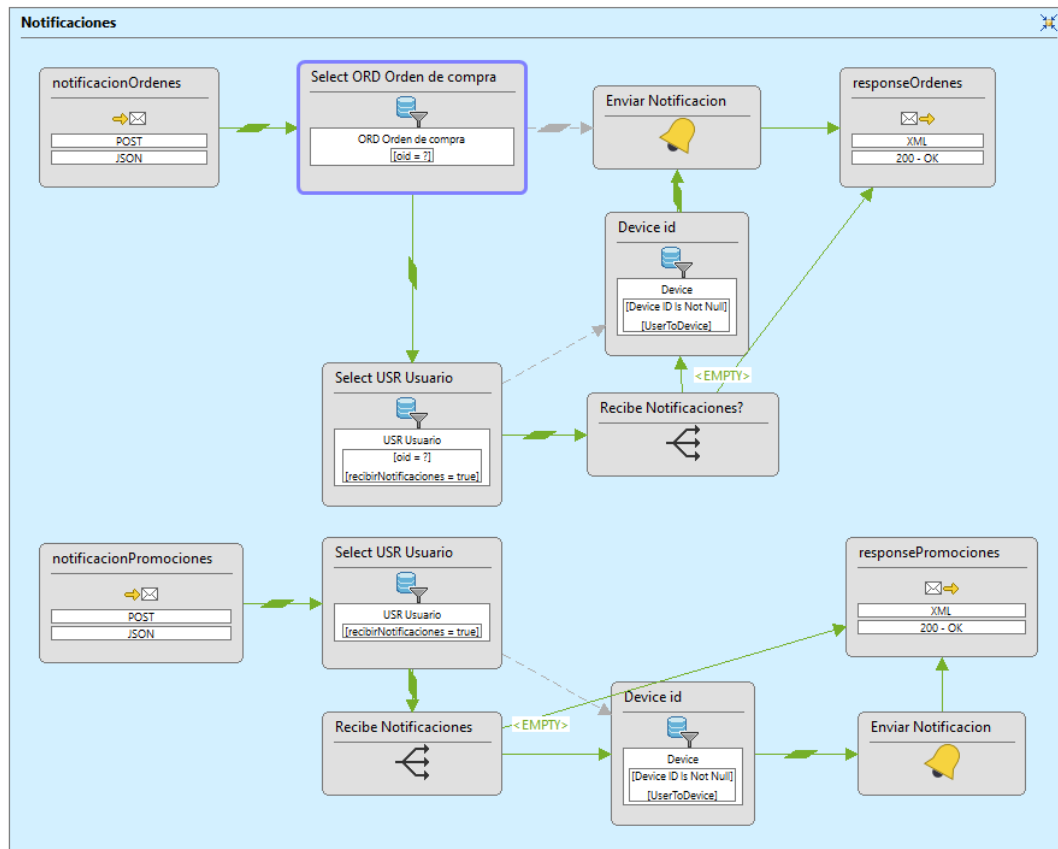


Gráfico 80 – Modelo IFML Servicio Web Notificaciones

Fuente: (López Yépez & Polo Vásconez, 2016)

Como se puede observar en la siguiente imagen, se presenta una nueva notificación que informa la presencia de nuevas ofertas dentro de la aplicación

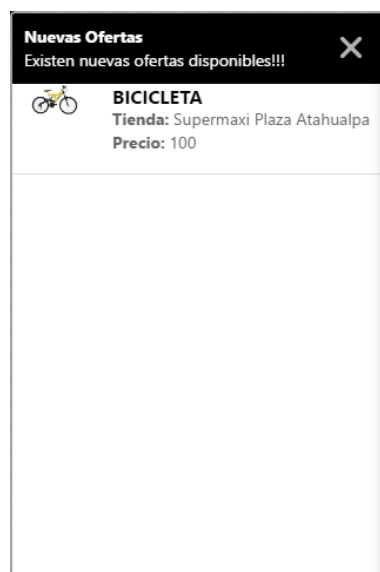


Gráfico 81 – Notificación Nuevas Ofertas

Fuente: (López Yépez & Polo Vásconez, 2016)

En la siguiente imagen se puede visualizar una notificación que indica el cambio de estado ha actualizado de una orden de compra.

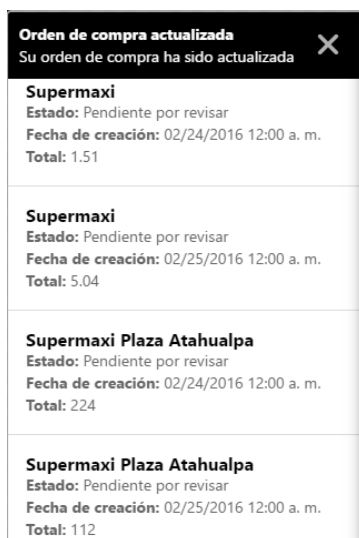


Gráfico 82 – Notificación órdenes de compra

Fuente: (López Yépez & Polo Vásconez, 2016)

3.3.5 Iteración 5

3.3.5.1 US009 – Carrito de compras

Desde cualquier pantalla de detalles de productos se le permite al usuario añadir productos al carrito de compras. Los productos que se añadan al carrito de compras se visualizan en el listado de productos del carrito, para esto se definió la estructura de modelado presentada en la siguiente imagen:

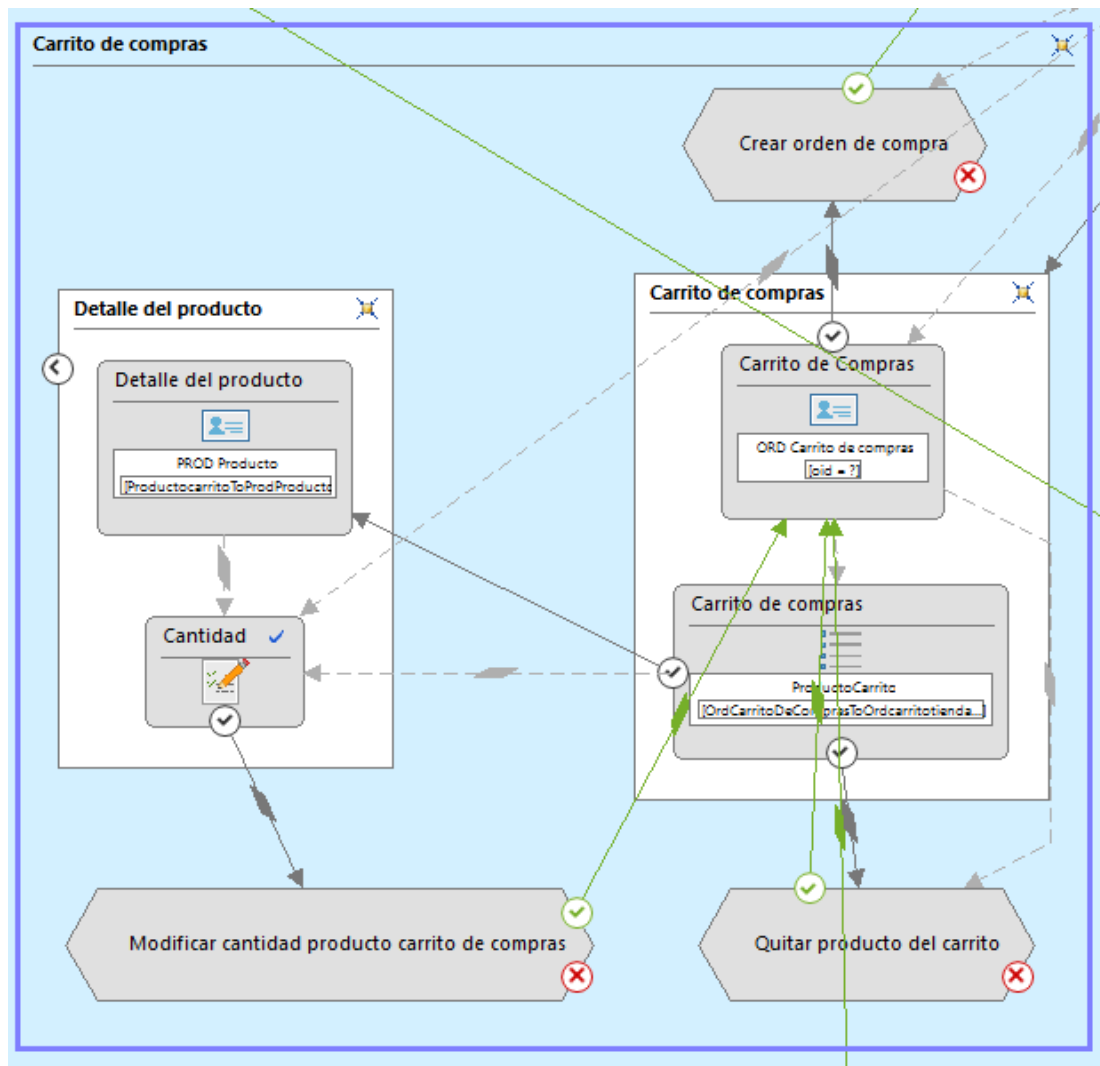


Gráfico 83 – Modelo IFML Carrito de Compras

Fuente: (López Yépez & Polo Vásconez, 2016)

Podemos acceder al carrito de compras desde la barra superior de nuestra aplicación, dentro de nuestro carrito de compras podremos observar los detalles de los productos que estamos adquiriendo, modificar la cantidad de productos requeridos cuando accedamos al detalle de producto que ya conocemos y de ser necesario podemos eliminar el producto del carrito de compras.

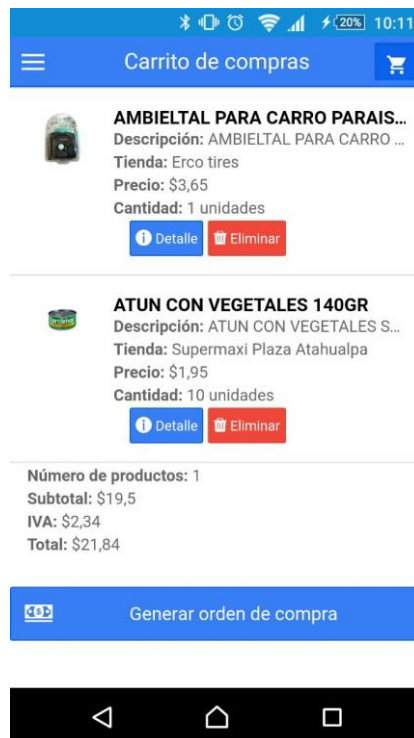


Gráfico 84 – Interfaz Carrito de Compras

Fuente: (López Yépez & Polo Vásconez, 2016)

3.3.6 Iteración 6

3.3.6.1 US010 – Orden de compra

Desde el listado “Mis órdenes de compra” se le permite al usuario visualizar las órdenes de compra generadas por él, para esto se definió la estructura de modelado presentada en la siguiente imagen:

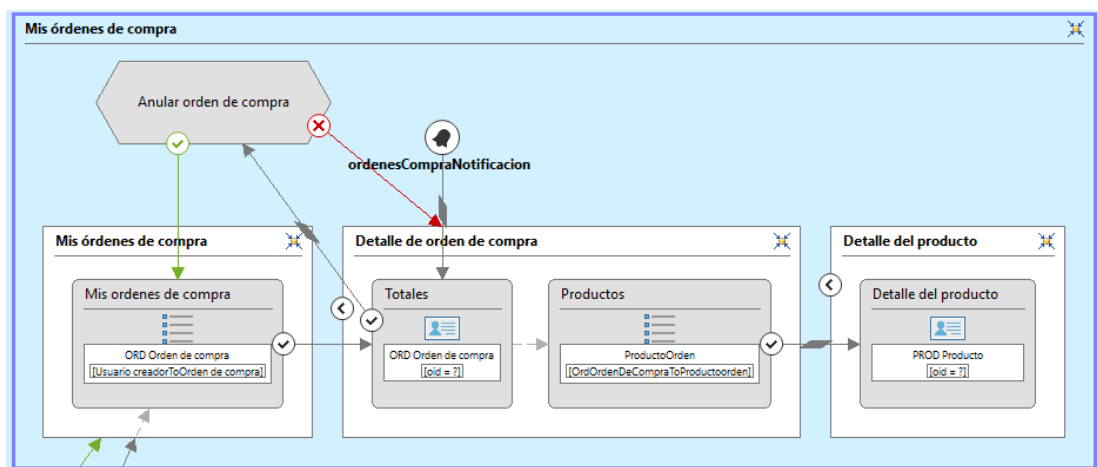


Gráfico 85 – Modelo IFML Mis órdenes de compra

Fuente: (López Yépez & Polo Vásconez, 2016)

Desde la pantalla mis órdenes de compra se permiten visualizar el estado de las órdenes de compra, así como el total a pagar por cada una de las tiendas, si el usuario requiere más detalle, podrá visualizar la orden de compra para consultar cuales fueron los productos que adquirió y dependiendo del estado de la orden de compra podrá cancelarla.

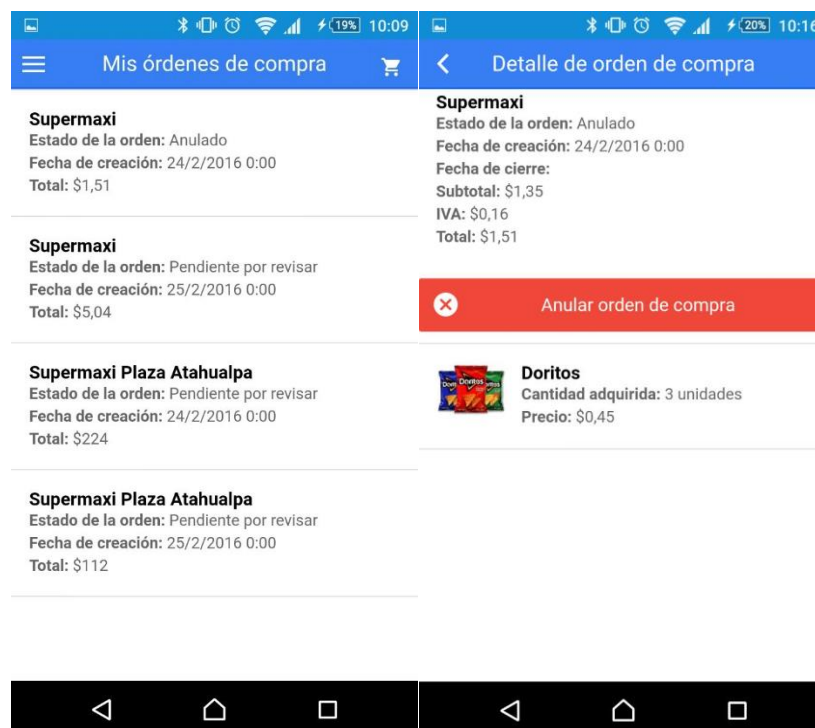


Gráfico 86 – Interfaz Mis órdenes de compra – Detalle de Orden de Compra

Fuente: (López Yépez & Polo Vásconez, 2016)

3.3.7 Iteración 7

En esta última iteración, se realizaron ajustes hasta tener la interfaz final de la aplicación móvil, además se probó la instalación en diferentes dispositivos Android e iOS en los cuales se pudo verificar el correcto funcionamiento de la aplicación, adicional se realizaron correcciones en cuanto a visualización de elementos que presentaban inconvenientes.

CONCLUSIONES

En el desarrollo del presente trabajo de titulación, se propuso la realización de aplicaciones móviles (Android – iOS) y web a la cual hemos denominado “Asistente de Compras”, utilizando la herramienta de desarrollo WebRatio y el marco de trabajo ágil *Scrum*.

Utilizar *Scrum* nos ayudó a comprender la importancia que tiene manejar un proyecto, desde el análisis y recopilación de requisitos hasta la correcta planificación de los productos que componen el proyecto, esto en función del tiempo y el compromiso que como profesionales tenemos que demostrar hacia nuestros clientes. Una buena relación entre los miembros del equipo *Scrum* y el cliente se vuelve, en nuestra opinión, el factor más importante, tanto para el desarrollo como para la culminación satisfactoria de un proyecto.

En cuanto a la gestión de los requisitos y análisis de los mismos también comprendimos que si un proyecto parte de buenos cimientos, es decir tenemos muy bien definidos y claros los requisitos, su desarrollo se volverá menos complicado conforme se avance en el tiempo hasta llegar a sus fases finales.

Si bien *Scrum* es una metodología de desarrollo ágil que nos ayuda a realizar proyectos definidos en tiempos cortos. Es importante que estos proyectos tengan la planificación y organización adecuada, en donde se pueda evidenciar una estructura coherente de la aplicación entre los productos a lo largo del desarrollo del proyecto. Esto debido a que si los productos culminados luego de cada iteración no demuestran que el proyecto tiene el concepto que nuestros clientes pretenden de su aplicación, damos la impresión de que no se está realizando un trabajo adecuado o que no se entiende lo que el cliente busca con la aplicación.

Conforme el proyecto fue desarrollándose progresivamente, pudimos evidenciar ciertas falencias en la tecnología de desarrollo WebRatio en su versión para dispositivos móviles. Inicialmente se utilizó para el desarrollo la versión WebRatio Mobile Platform 8.4.4 pero debido a las deficiencias presentadas, principalmente cuando se realizó la integración entre las aplicaciones Web y Móvil, específicamente cuando se realiza la sincronización de la base de datos MySQL con la base de datos del dispositivo móvil SQLite, dichos problemas derivaron en la demora del desarrollo

de la aplicación móvil y por lo tanto retrasos en la producción, por lo cual podemos concluir que, aunque se migró de la versión inicial utilizada para el desarrollo a la versión más actual disponible hasta la fecha (WebRatio Mobile Platform 8.7.4), la tecnología WebRatio no es todavía lo suficiente madura para el uso intensivo de datos.

De igual manera, mientras se avanzaba con el desarrollo del proyecto y luego de cada reunión de revisión de los productos con el cliente, casi siempre, éste tiene una o varias observaciones. Si bien estas observaciones nos han ayudado para mejorar las funcionalidades desarrolladas, es importante encausar al cliente para que se concentre en los aspectos significativos de las mejoras y los detalles queden a cargo del grupo de trabajo.

Un cliente puede considerar en ocasiones que algo que nos solicita es simplemente una mejora a los prototipos desarrollados, muchas veces realizada únicamente como una aclaración a un requerimiento, pero no siempre es así. En ocasiones estas “mejoras” no son más que nuevas funcionalidades, por eso recomendamos que se proponga desde el inicio del proyecto que los productos sean desarrollados como se los definieron y en caso de encontrar mejoras o nuevas funcionalidades, estas sean estimadas como nuevos productos y situadas dentro de una iteración diferente, debido a que esto suele modificar el tiempo de entrega y el costo del proyecto.

En cuanto a las mejoras que el cliente generalmente solicita, el número de mejoras solicitadas también deberían limitarse como se mencionó anteriormente ya que la excesiva cantidad de mejoras solicitadas dentro de un proyecto pueden conllevar a que el mismo se introduzca en un ciclo de nunca acabar y nos ate a un cliente por más tiempo del que tenemos planificado, generando incomodidad tanto para el cliente como para el equipo de desarrollo.

Al finalizar el desarrollo de las aplicaciones móvil (Android – iOS) y web se procedió a realizar las diferentes pruebas de integración para verificar el comportamiento en los diferentes dispositivos, lo cual también derivó en inconvenientes al momento de generar los instaladores para las aplicaciones móviles para dispositivos Apple, debido a que las aplicaciones únicamente podían ser probadas en los dispositivos registrados (iPhone/iPad/iPod Touch) y asociados a la cuenta del desarrollador. De igual forma, para poder generar los instaladores para dispositivos Apple se debía generar certificados de seguridad para que la aplicación pueda ser instalada. Esta

parte crucial en el desarrollo solo puede ser generada en equipos Apple y que cuenten con el sistema operativo OSX en su última versión por lo tanto también es requerido un equipo (MacBook, iMac) de este tipo y no solo bastaba con la adquisición de una cuenta de desarrollador y tener habilitado el acceso a subir aplicaciones al iTunes Store de Apple. Luego de superar todos estos inconvenientes se puede generar una aplicación para dispositivos móviles con sistema operativo iOS.

En contraparte, generar el instalador de una aplicación para dispositivos con sistema operativo Android tiene un proceso más sencillo y menos restrictivo ya que puede ser generado sin mayores inconvenientes y debe ser configurado únicamente con las *API Keys* de la cuenta de desarrollador de Google dependiendo de las funcionalidades que se requiera para la aplicación móvil (acceso a mapas, notificaciones).

En cuanto a la instalación de la aplicación web, esta puede ser desplegada en un servidor de aplicaciones compatible con JSP (*Java Server Pages*), por ejemplo, Apache Tomcat o JBoss.

Finalmente podemos concluir que se desarrollaron satisfactoriamente las aplicaciones tanto web como móviles llamadas “Asistente de compras” las mismas que al usuario final o cliente le dan la posibilidad de realizar búsquedas de productos de acuerdo a sus necesidades, añadirlos a un carrito de compras, revisar los productos añadidos y crear órdenes de compra para que sean atendidas por las diferentes tiendas afiliadas, dándole así la posibilidad de revisar y realizar sus compras desde la comodidad de sus hogares o donde sea que se encuentren y únicamente recibir la notificación de que sus productos están listos para ser retirados.

BIBLIOGRAFÍA

- Alarcon, R., & Wilde, E. (2010). Linking Data from RESTful Services.
- Alarcon, R., Wilde, E., & Bellido, J. (2011). Hypermedia-Driven RESTful Service. En R. Alarcon, E. Wilde, & J. Bellido, *Service-Oriented Computing* (págs. 111-120). Obtenido de <http://dret.net/netdret/docs/wilde-wesoa2010-restful-composition.pdf>
- Amazon. (Diciembre de 2015). *Amazon*. Recuperado el 30 de Noviembre de 2015, de <http://www.amazon.com/>
- Brambilla, M., & Butti, S. (Septiembre de 2014). *FROM WEBML TO WEBRATIO AND IFML*. Obtenido de <http://dbgroup.como.polimi.it/brambilla/sites/dbgroup.como.polimi.it/brambilla/files/referencematerials/WebML-IFML-WebRatio-Novatica-EN.pdf>
- Deemer, P., Benefield, G., Larman, C., & Vodde, B. (2012). *The Scrum Primer*.
- Fernández Romero, Y., & Díaz González, Y. (2012). Patrón Modelo-Vista-Controlador. *Revista Telem@tica*, 47-57.
- Garcia, R. (Enero de 2012). *Gitbooks*. Recuperado el 10 de Noviembre de 2015, de <https://rdgarcia.gitbooks.io/generacion-de-aplicaciones-web-confiml/content/Caracter%C3%ADsticas.html>
- Hundermark, P. (2009). *Do Better Scrum Second Edition*.
- López Yépez, P. X., & Polo Váscquez, J. A. (2016). Asistente de Compras.
- Mercado Libre. (Diciembre de 2015). *Mercado Libre*. Recuperado el 1 de Diciembre de 2015, de <http://www.mercadolibre.com.ec/>
- Object Managment Group. (Febrero de 2015). *Interaction Flow Modeling Language*. Obtenido de <http://www.omg.org/spec/IFML/1.0/PDF/>
- OLX. (Diciembre de 2015). *OLX*. Recuperado el 30 de Noviembre de 2015, de <http://www.olx.com.ec/>
- Palacio, J. (2014). Gestión de proyectos Scrum Manager. Scrum Manager.
- Qureshi, R. J., & Sabir, F. (2013). A COMPARISON OF MODEL VIEW CONTROLLER AND MODEL. *Sci.Int.(Lahore)*, 25(1), 7-9.
- Rossi, G. (Agosto de 2013). *Web Modeling Languages Strike Back*. Obtenido de <https://www.computer.org/csdl/mags/ic/2013/04/mic2013040004.pdf>
- Rubin, K. S. (2013). *Essential Scrum A Practical Guide to the Most Popular Agile Process*. Pearson Education, Inc.
- Universidad de Alicante. (2015). *Modelo vista controlador (MVC)*. Obtenido de <http://si.ua.es/es/documentacion/asp-net-mvc-3/1-dia/modelo-vista-controlador-mvc.html>

WebRatio. (2014). Webratio Enterprise Platform. Italia. Recuperado el 10 de Noviembre de 2015, de <http://www.webratio.com>

WebRatio. (2015). Webratio Mobile Platform. Italia. Recuperado el 12 de Noviembre de 2015, de <http://www.webratio.com>

ANEXOS

Anexo 1

Terminología

Termino propio	Termino original
Acción	Action definition
Ancla	Anchor
Área	Area
Asociación	Association
Atributo	Attribute
Atributo calculado	Calculated Attribute
Atributo clave	Key Attribute
Atributo constante	Constant Attribute
Atributo derivado manual	Manually Derived Attribute
Atributo importado	Imported Attribute
Atributos condición de restricción	Attributes Restriction Condition
Atributos derivados	Derived attributes
Barra de herramientas	Toolbar
Campo	Field
Campo de selección	ComboBox
Celda	Celda
Clase	Class
Clase de servicio	Service Class
Clases	Classes
Cliente	Product Owner
Componente	Component
Componente de vista	View Component
Componentes	Components
Condición de atributos	Attributes Condition
Condición de clave	Key Condition
Condición de relación	RelationshipRole Condition
Contenedor de vista	View Container
Contenedor de vista destacado	Landmark view container
Definiciones de módulo	Module Definitions
Disparador	Trigger
Elemento de vista	View Element

Termino propio	Termino original
Enlace de parámetros	Parameter Binding
Entidad	Entity
Entidades	Entities
Estilo	Style Definition
Evento	Event
Evento de salto	Jump Event
Evento enviar	On Submit Event
Evento regresar	Back Event
Evento selección	On Select Event
Evento seleccionar	Select Event
Expresión de activación	Activation Expression
Expresión de un flujo de interacción	Interaction Flow Expression
Facilitador	<i>Scrum</i> Master
Flujo	Flow
Flujo correcto	OK Flow
Flujo de datos	Data Flow
Flujo de navegación	Navigation Flow
Flujo incorrecto	KO Flow
Flujos	Flows
Flujos de interacción	Interaction Flows
Formulario	Form
Fuente de datos	Database
Fuera de línea	Offline
Generalización	Generalization
Gráficos de trabajo pendiente	Burndown del Sprint
Grilla	Grid
Grupo de operaciones	Operation Group
Infraestructura	Framework
Interfaz	Front–end
Iteración	Sprint
Lista de impedimentos	Backlog de impedimentos
Lista de tareas	Backlog
Lista de tareas de la iteración	Sprint Backlog
Lista de tareas del producto	Product Backlog

Termino propio	Termino original
Lista Simple	Simple List
Marco	Frame
Método de servicio	Service Method
Rastro web	Breadcrumb
Modelo de dominio	Domain Model
Modelo Web	Web Model
Módulo	Module
Módulos	Modules
Motor	Back-end
Operación	Operation
Operación de servicio de usuario	User service operation
Página	Page
Página de inicio	Home Page
Página por defecto	Default Page
Página principal	Master Page
Páginas	Pages
Páginas alternativas	Alternative page
Páginas principales	Master Pages
Pantalla	Screen
Pantalla de ajuste	Screen Set
Pantallas dedicadas	Screens devoted
Parámetro	Parameter
Parámetro de contexto	Context Parameter
Parte de un componente de vista	View Component Part
Plantilla de diseño	Layout template
Proveedores de datos	Data Providers
Proyecto de servicio de datos	Data Service Project
Proyecto móvil	Mobile Project
Proyecto WebRatio	WebRatio Project
Puerto	Port
Puerto de entrada	Input Port
Puerto de salida	Output Port
Punto prominente	Landmark
Regla de validación	Validation Rule

Termino propio	Termino original
Relación	Relationship
Reuniones diarias	Daily <i>Scrum</i>
Rol	Role
Rol de asociación	Association Role
Salto	Jump
Servicio	Service
Servicio de datos	Data Service
Servicios de usuario	User Services
Servicios web REST	Restful Services
Tarea	Job
Tareas	Jobs
Valor	Value
Ventana	Window
Ventana de bloqueo	Modal Window
Ventana superpuesta	Modeless Window
Vista de aplicación	App View
Vista del servicio	Service View
Vista del sitio	Site View

Análisis de requisitos

El análisis de requisitos nos permite identificar y definir las funcionalidades que necesita un sistema

2.1 Gestión de tiendas

Una tienda es una entidad que ofrece productos a sus clientes por medio de las aplicaciones web y móvil, estas tiendas pueden ser almacenes, supermercados, distribuidoras, etc. Cada una de las tiendas tendrá las siguientes características:

- Nombre: es el nombre de la tienda
- Tipo de tienda: permite identificar si es un supermercado, ferretería, mueblería, etc.
- Código: código único que identificará a una tienda.
- Logo: imagen que permite identificar a una tienda.
- Responsable: persona encargada de gestionar los datos de una tienda en las aplicaciones.
- Dirección: dirección física de una tienda (Ubicación)
- Coordenadas geográficas: latitud y Longitud de ubicación de una tienda.
- Teléfonos: teléfonos de contacto de una tienda, pueden ser uno o varios.
- Sucursales: una tienda puede tener varias sucursales
- Horario de atención
- URL de facturación

2.2 Gestión de responsables de tiendas

Un responsable de tiendas es una persona que accede al sistema web para gestionar los datos de una tienda, gestionar los productos que estas ofertan a sus clientes y supervisar las órdenes de compra que se han creado. Cada responsable cuenta con las siguientes características:

- Nombre del responsable
- Apellido del responsable
- Correo electrónico
- Teléfono fijo
- Teléfono móvil

- Dirección de localización: dirección en donde se puede localizar al responsable.

2.3 Gestión de productos

Un producto es un bien tangible que las tiendas ponen a disposición de los clientes. Una tienda puede ofrecer uno o varios productos, los cuales tendrán las siguientes características:

- Código
- Nombre: nombre que identificará brevemente a un producto
- Precio: valor de adquisición de un producto
- Requiere IVA: permite identificar si a un producto se le aumentará el valor del IVA (Impuesto al valor agregado) para la emisión de la orden de compra.
- Descripción del producto: porción de texto que explica la utilidad y características de un producto.
- Foto: imagen que permite visualizar el producto que un cliente pretende adquirir.
- Cantidad disponible: permite identificar si una tienda tiene en stock o no un producto.
- Categoría del producto
- Marca: define el fabricante del producto
- En promoción: define si un producto se encuentra en promoción o no, para permitir el envío de notificaciones automáticas.

2.4 Gestión de categorías de productos

Una categoría de producto es una característica que permite diferenciar a los productos dentro de una tienda, ejemplos de categorías son: Herramientas, maderas, lácteos, tecnología, etc. Estas categorías están compuestas por:

- Nombre de la categoría
- Código de la categoría: código único que permite identificar una categoría
- Descripción de la categoría: porción de texto que explica que productos se incluyen dentro de cada categoría.
- Por parámetro, cada categoría de producto tendrá un tiempo máximo de retiro por el cliente, el cual se usará para calcular el tiempo en que se dará de baja una orden de compra, esto debido a que existe productos que pueden verse

afectados por haber sido despachados, como por ejemplo la leche, queso, etc.

2.5 Búsqueda de productos

La búsqueda de productos, permitirá al usuario comparar productos por su categoría, marca, nombre, código, precio y en qué tiendas se los puede encontrar. Esta búsqueda de productos tiene como finalidad brindar a los usuarios la posibilidad de ayudarlo a decidirse por productos que se encuentren en ofertas, priorizándolos entre sus similares.

2.6 Registro de usuarios

El registro de usuarios permitirá controlar a las personas que acceden a las aplicaciones, restringiéndoles el acceso a ciertas áreas del sistema, como por ejemplo el gestor de tiendas. Cada usuario que se registre en el sistema debe proveer los siguientes datos:

- Nombre de la persona
- Apellido de la persona
- Dirección de la persona
- Teléfono fijo
- Teléfono móvil
- Nombre de usuario: el nombre de usuario será usado para acceder al sistema.
- Contraseña: la contraseña será usada para acceder al sistema. Esta contraseña deberá tener un mínimo de 8 caracteres entre por lo menos números y letras.
- Cedula/Ruc de la persona: este campo será usado para que el usuario despachador puede llenar los datos de facturación.
- Estado: el estado permitirá identificar si un usuario está activo o no. Un usuario activo es aquel que se le considera confiable para realizar las órdenes de compra, en cambio un usuario inactivo es aquel que ha realizado órdenes de compra pero que no se le considera confiable debido a que no ha retirado al menos 3 órdenes de compra indistintamente de la tienda.
- Recibir notificaciones: este campo permitirá a los usuarios evitar o permitir que el sistema le envíe notificaciones.

2.7 Roles de usuario

Los roles de usuario son restricciones de seguridad que se usan para controlar el acceso a ciertas áreas del sistema para cada diferente tipo de usuario, esto lo gestiona la aplicación.

2.7.1 Administrador del sistema

El administrador del sistema estará en la capacidad de gestionar usuarios, roles, tiendas, tipos de tienda y categoría de productos.

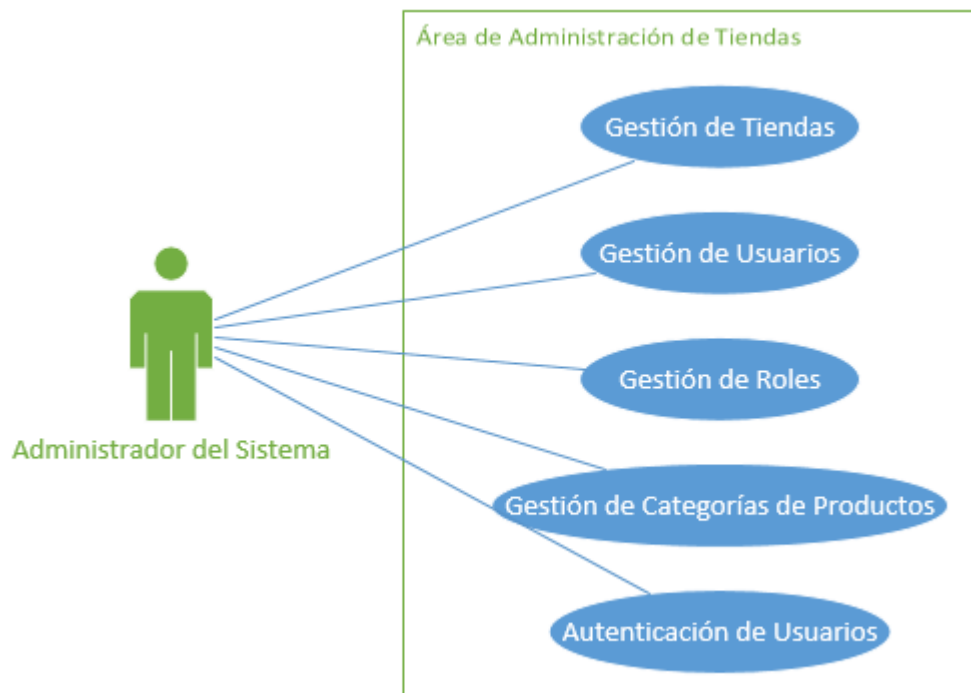


Gráfico 87 – Caso de uso usuario administrador

Fuente: (López Yépez & Polo Vásconez, 2016)

2.7.2 Responsable de tienda

Un usuario supervisor de tienda gestionará datos únicamente de su tienda asignada por el administrador del sistema. Este usuario podrá gestionar datos de su tienda, sucursales de la tienda, productos, administrar usuarios despachadores y revisar órdenes de compra despachadas.

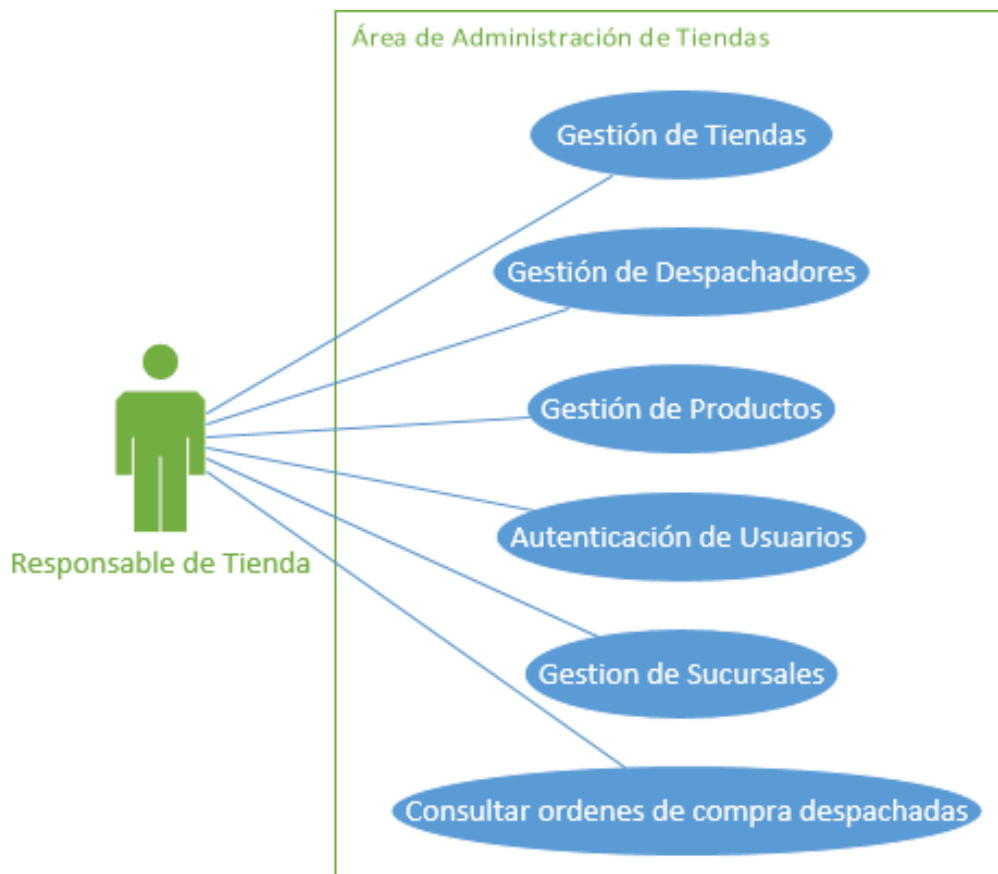


Gráfico 88 – Caso de uso usuario responsable de tienda

Fuente: (López Yépez & Polo Vásconez, 2016)

2.7.3 Despachador

El usuario despachador revisará las órdenes de compra emitidas por los clientes del sistema y cambiará su estado, con esto el sistema notificará a los clientes cuando sus órdenes han sido despachadas.

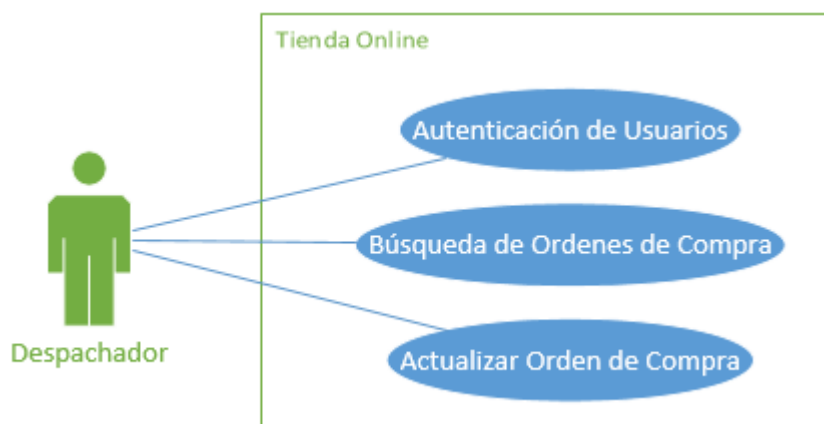


Gráfico 89 – Caso de uso usuario despachador

Fuente: (López Yépez & Polo Vásconez, 2016)

2.7.4 Cliente

Un cliente podrá registrarse en el sistema, consultar productos, agregar productos al carrito de compras, emitir varias órdenes de compra (una a la vez) sin importar la tienda del producto, consultar los datos de las diferentes tiendas, marcar tiendas y/o productos como favoritos, consular las facturas electrónicas emitidas por sus órdenes de compra recibidas y crear reseñas para productos que han sido adquiridos por el cliente.

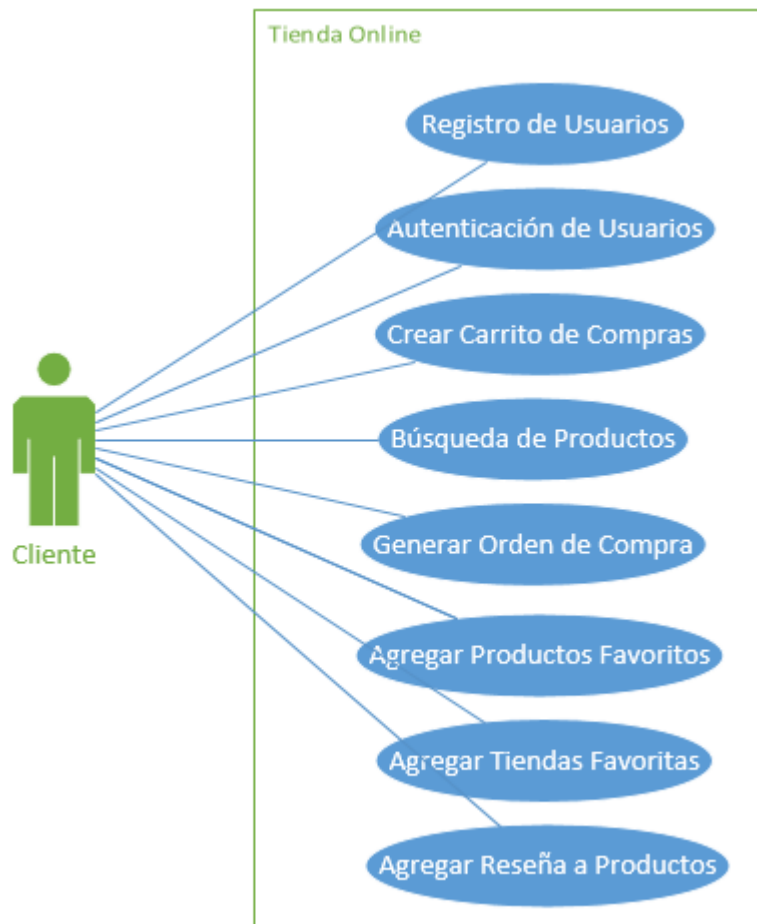


Gráfico 90 – Caso de uso cliente

Fuente: (López Yépez & Polo Vásconez, 2016)

2.8 Órdenes de compra

Una orden de compra es un comprobante digital que permite visualizar todos los productos que se han añadido al carrito de compras para ser adquiridos y finalmente emitidos a las tiendas en las que se realizarán las compras.

Para el manejo de las órdenes de compra, es necesario que los usuarios puedan realizar las siguientes acciones:

- Emitir una orden de compra final: una vez que el usuario confirma la orden de compra
- Cancelar una orden de compra: un usuario puede cancelar una orden de compra dentro de los primeros 15 minutos luego de que esta ha sido emitida.
- Fecha y hora de retiro: esta fecha y hora permitirá cerrar las órdenes de compra una vez que este campo sea actualizado. Este campo puede ser actualizado únicamente por un usuario despachador.
- Estado de las órdenes de compra: una orden de compra puede tener los estados de Creada (al momento de ingresar productos al carrito), Pendiente por revisar (Despachador), Esperando retiro del cliente, Retirado por el cliente y Anulado por el cliente.
- Una vez que se actualiza este campo se emite una notificación al teléfono del cliente y su correo electrónico con los datos y estado de la orden de compra.

2.9 Carrito de compras

Un carrito de compras permite a los usuarios marcar los productos que pueden adquirir y consultar el total del valor a pagar por todos esos productos agregados por cada compra diferente que se haga. Las características principales de un carrito de compras son:

- Agregar productos al carrito.
- Quitar productos del carrito de compras.
- Consultar los productos añadidos y verificar el total a pagar por la compra.
- Confirmación del carrito de compras (emisión de la orden de compra).

2.10 Despachadores

Es la persona que se encarga de verificar las órdenes de compra, modificar su estado y poner la fecha de cuándo puede acercarse a retirar los productos.

2.11 Productos y tiendas favoritas

El sistema tendrá parametrizado el número de compras realizadas en una determinada tienda y/o producto para asignar de manera automática como favoritos del cliente, así como también, el cliente tendrá la opción de agregar a su criterio sus productos y/o tiendas favoritas. Estará disponible la opción de consultar los favoritos.

2.12 Facturación electrónica

Un cliente puede consultar las facturas de órdenes de compra emitidas por el sistema, esto desde la web de la tienda o desde un enlace directo a la página del SRI.

2.13 Envío de notificaciones

El sistema envía notificaciones por correo electrónico y notificaciones push de los productos que se encuentran en promoción de las tiendas favoritas de los clientes.

2.14 Reseñas

El cliente que tuvo la experiencia de comprar un producto, puede añadir reseñas indicando su experiencia (En un rango de 1 a 5 siendo 1 poco satisfactoria y 5 muy satisfactoria), la misma que podrá ser visualizada por otros usuarios. Además de consultar el ranking general por cada producto, es decir un promedio de las calificaciones realizadas por otros usuarios.

Definición de las historias de usuario

A continuación, el detalle de las Historias de usuario para el “Asistente de compras”

3.1 US000 – Configuración de ambiente de desarrollo
ID: US000 Complejidad de Desarrollo: Baja Nombre: Configuración de ambiente de desarrollo Dependencia: Ninguna Prioridad: Urgente Tiempo estimado: 3 horas
Descripción Teniendo en Cuenta: Un miembro del equipo de desarrollo <i>SCRUM</i> requiere configurar el ambiente de bases de datos, CVS y Carpeta de trabajo (<i>Workspace</i>) de WebRatio Enterprise Platform. Cuándo: Previo al proceso de desarrollo del proyecto. Entonces: Define las rutas de acceso (CVS y <i>Workspace</i>) y cadenas de conexión con la base de datos.
Criterio de Aceptación: El entorno está configurado correctamente y se encuentra a disposición de los desarrolladores.

Tabla 2 - US000 – Configuración de ambiente de desarrollo

Fuente: (López Yépez & Polo Vásconez, 2016)

3.2 US001 – Modelo de dominio
ID: US001 Complejidad de Desarrollo: Baja Nombre: Modelo de dominio Dependencia: US000 Prioridad: Urgente Tiempo estimado: 3 horas

Descripción Teniendo en Cuenta: La aplicación Web requiere de una base de datos relacional. Cuándo: Los ambientes de desarrollo se encuentran configurados correctamente. Entonces: Defino el modelo Entidad–Relación (E–R) que permita el manejo de la base de datos de la aplicación Web, la misma que será consumida por la aplicación móvil.
Criterio de Aceptación: El modelo entidad–relación contiene las tablas y relaciones necesarias para el manejo persistente de los datos que manejaran las dos aplicaciones.

Tabla 3 - US001 – Modelo de dominio

Fuente: (López Yépez & Polo Vásconez, 2016)

3.3 US002 – Gestión de Roles de usuario
ID: US002 Complejidad de Desarrollo: Media Nombre: Gestión de Roles de usuario Dependencia: US001 Prioridad: Alta Tiempo estimado: 4 horas
Descripción Teniendo en Cuenta: La aplicación Web requiere un área de administración de roles de usuario para la seguridad del sistema. Cuándo: El Modelo de dominio ha sido desarrollado. Entonces: Defino un área de administración de roles de usuario.
Criterio de Aceptación: El área desarrollada permite crear, modificar, eliminar y consultar roles de usuario de la aplicación Permite el acceso a esta área únicamente al Rol “Administrador del sistema”. Los roles creados por defecto en la aplicación son:

- Administrador del sistema
- Responsable de tienda
- Despachador
- Cliente

Tabla 4 - US002 – Gestión de Roles de usuario

Fuente: (López Yépez & Polo Vásconez, 2016)

3.4 US003 – Registro de usuarios

ID: US003

Complejidad de Desarrollo: Media

Nombre: Registro de usuarios

Dependencia: US002

Prioridad: Alta

Tiempo estimado: 11 horas

Descripción:

Teniendo en Cuenta:

La aplicación Web requiere un área de registro de usuarios y un área de administración de usuarios.

Cuándo:

El área de gestión de roles de usuario ha sido desarrollada.

Entonces:

Defino un área de registro de usuarios y un área de administración de usuarios.

Criterio de Aceptación:

Registro de usuarios

Existe una página de registro de usuarios

El registro de usuarios obliga a confirmar la cuenta para acceder al sistema.

Los usuarios registrados tienen la opción de recuperación de su cuenta (Sin importar su rol en el sistema).

El área de registro de usuarios únicamente creará usuarios con el rol de “cliente”

Área de gestión de usuarios

Existe un área donde se puede crear, modificar, inactivar y consultar los usuarios del sistema.

Permite el acceso a esta área únicamente al Rol “Administrador del sistema”.

Perfil de usuario

Crear una página que permita a los usuarios que han accedido a la aplicación, modificar sus datos.

Tabla 5 - US003 – Registro de usuarios

Fuente: (López Yépez & Polo Vásconez, 2016)

3.5 US004 – Gestión de Tiendas

ID: US004

Complejidad de Desarrollo: Media

Nombre: Gestión de Tiendas

Dependencia: Ninguna

Prioridad: Alta

Tiempo estimado: 7 horas

Descripción:

Teniendo en Cuenta:

La aplicación Web requiere un área de administración de tiendas.

Cuándo:

Se han definido los roles de usuario de la aplicación.

Entonces:

Defino un área de administración de tiendas.

Criterio de Aceptación:

El área desarrollada permite crear, modificar, eliminar y consultar datos de las tiendas afiliadas a la aplicación

Permite el acceso a esta área únicamente al Rol “Administrador del sistema”.

Cuenta con una página de consulta de tiendas disponibles a través de un mapa, en donde se puede acceder a la información de detalle completo por cada tienda.

El área de consulta de tiendas estará disponible para todos los roles de usuario definidos en la US002

Tabla 6 - US004 – Gestión de Tiendas

Fuente: (López Yépez & Polo Vásconez, 2016)

3.6 US005 – Gestión de responsables de tiendas

ID: US005

Complejidad de Desarrollo: Baja

Nombre: Gestión de responsables de tiendas

Dependencia: US004

Prioridad: Normal Tiempo estimado: 4 horas
Descripción: Teniendo en Cuenta: La aplicación Web requiere un área de administración de responsables de tiendas. Cuándo: El área de gestión de tiendas ha sido desarrollada Entonces: Definimos un área de administración de responsables de tiendas.
Criterio de Aceptación: El área desarrollada permite asignar responsables a las tiendas afiliadas a la aplicación Permite el acceso a esta área únicamente al Rol “Administrador del sistema”. Cuenta con un acceso directo para crear un usuario responsable a una tienda específica.

Tabla 7 - US005 – Gestión de responsables de tiendas

Fuente: (López Yépez & Polo Vásconez, 2016)

3.7 US006 – Gestión de categorías de productos
ID: US006 Complejidad de Desarrollo: Baja Nombre: Gestión de categorías de productos Dependencia: ninguna Prioridad: Normal Tiempo estimado: 4 horas
Descripción: Teniendo en Cuenta: La aplicación Web requiere un área de gestión de categorías de productos. Cuándo: Han sido definidas las categorías a usarse en el sistema Entonces: Defino el área de gestión de categorías de productos.

Criterio de Aceptación:

El área desarrollada permite crear, modificar, eliminar y consultar categorías de productos.

Permite el acceso a esta área únicamente al Rol “Administrador del sistema”.

Las categorías de producto creadas por defecto son:

- Libros
- Hogar y cocina
- Alimentos
- Electrodomésticos
- Electrónicos (TV, Audio, Computadoras y dispositivos móviles)
- Deportes
- Salud
- Software (Sistemas operativos, Antivirus, videojuegos)

Tabla 8 - US006 – Gestión de categorías de productos

Fuente: (López Yépez & Polo Vásquez, 2016)

3.8 US007 – Gestión de productos

ID: US007

Complejidad de Desarrollo: Baja

Nombre: Gestión de productos

Dependencia: US006

Prioridad: Normal

Tiempo estimado: 4 horas

Descripción:**Teniendo en Cuenta:**

La aplicación Web requiere un área de gestión de productos.

Cuándo:

Han sido definidos las categorías de productos que podrán usar las tiendas

Entonces:

Definimos el área de gestión de productos.

Criterio de Aceptación:

El área desarrollada permite crear, modificar, eliminar y consultar productos de una tienda específica.

Permite el acceso a esta área únicamente al Rol “Responsable de tienda”.

Existe una opción que permite marcar a un producto como “en promoción” o no.

Tabla 9 - US007 – Gestión de productos

Fuente: (López Yépez & Polo Vásconez, 2016)

3.9 US008 – Búsqueda de productos
ID: US008 Complejidad de Desarrollo: Alta Nombre: Búsqueda de productos Dependencia: US007 Prioridad: Normal Tiempo estimado: 5 horas
Descripción: Teniendo en Cuenta: La aplicación Web requiere un buscador productos. Cuándo: El área de gestión de productos ha sido creada Entonces: Definimos el área de búsqueda de productos.
Criterio de Aceptación: El área desarrollada permite buscar productos por: - Categoría - Marca - Nombre - Código - Precio - Tienda Permite visualizar el buscador únicamente a los Roles “Responsable de tienda” y “cliente”. Permite añadir los productos consultados al carrito de compras.

Tabla 10 - US008 – Búsqueda de productos

Fuente: (López Yépez & Polo Vásconez, 2016)

3.10 US009 – Carrito de compras
ID: US009 Complejidad de Desarrollo: Alta Nombre: Carrito de compras Dependencia: US008 Prioridad: Normal Tiempo estimado: 12 horas
Descripción: Teniendo en Cuenta: La aplicación Web requiere de un carrito de compras. Cuándo: El buscador de productos ha sido creado Entonces: Definimos un carrito de compras que me permita añadir uno o varios productos de diferentes tiendas.
Criterio de Aceptación: Un usuario con rol de “cliente” tiene la opción de añadir y quitar productos al carrito de compras. Permite usar el carrito de compras únicamente al Rol “cliente”. Se puede aceptar la compra de los productos que están en el carrito de compras.

Tabla 11 - US009 – Carrito de compras

Fuente: (López Yépez & Polo Vásquez, 2016)

3.11 US010 – Orden de compra
ID: US010 Complejidad de Desarrollo: Alta Nombre: Carrito de compras Dependencia: US009 Prioridad: Normal Tiempo estimado: 12 horas

Descripción: Teniendo en Cuenta: La aplicación Web requiere de la emisión de órdenes de compra. Cuándo: El carrito de compras ha sido creado Entonces: Definimos una página que me permita manejar las órdenes de compra.
Criterio de Aceptación: Un usuario con rol de “cliente” tiene la opción de emitir órdenes de compra a través de la aceptación de los productos de su carrito de compras. El sistema calcula el tiempo máximo para cancelar una orden de compra, en base a los productos que se añadieron al carrito de compras. Permite anular una orden de compra al usuario “cliente” que lo creó, esto dentro del tiempo calculado. El usuario con rol “Despachador” puede consultar las órdenes de compra con estado “Pendiente por revisar (Despachador)” y actualizar las órdenes de compra despachadas con el estado de “Esperando retiro del cliente”. Cuando el usuario con rol “cliente” se acerque a retirar su orden de compra a la tienda, entonces el usuario despachador actualiza el estado de la orden a “Retirado por el cliente”. Una vez que se actualiza el campo “estado” de la orden de compra, el sistema emite una notificación al teléfono del cliente y su correo electrónico con los datos de la compra como los productos, el total y el estado de la orden de compra.

Tabla 12 - US010 – Orden de compra

Fuente: (López Yépez & Polo Vásconez, 2016)

3.12 US011 – Productos y tiendas favoritas
ID: US011 Complejidad de Desarrollo: Media Nombre: Productos y tiendas favoritas Dependencia: Ninguna Prioridad: Normal Tiempo estimado: 6 horas

Descripción: Teniendo en Cuenta: La aplicación Web y Móvil manejará una o varias tiendas y productos como favoritos, estos favoritos pueden ser marcados automáticamente por el sistema y manual por el cliente. Cuándo: El usuario ha realizado n (valor por parámetro del sistema) compras en una tienda o cuando el cliente lo requiera. Entonces: Definimos una opción en cada producto para poder marcarlo como favorito
Criterio de Aceptación: Un usuario con rol de “cliente” tiene la opción marcar a una o varias tiendas como favoritas. Un usuario con rol de “cliente” tiene la opción marcar a uno o varios productos como favoritos. El sistema tiene parametrizado el número de compras realizadas en una determinada tienda y/o producto para asignar de manera automática como favoritos del usuario con rol “cliente”. Existe una página de consulta de los favoritos para cada usuario.

Tabla 13 - US011 – Productos y tiendas favoritas

Fuente: (López Yépez & Polo Vásquez, 2016)

3.13 US012 – Notificaciones automáticas por ofertas en las tiendas
ID: US012 Complejidad de Desarrollo: Media Nombre: Notificaciones automáticas por ofertas en las tiendas Dependencia: US011 Prioridad: Normal Tiempo estimado: 5 horas

Descripción: Teniendo en Cuenta: La aplicación Web enviará notificaciones automáticas por correo electrónico y la aplicación móvil mediante notificaciones push. Cuándo: El usuario con rol “Responsable de tienda” marque a un producto como “en promoción” Entonces: El sistema enviará automáticamente notificaciones a los usuarios con rol “cliente”
Criterio de Aceptación: Un usuario con rol de “cliente” recibe notificaciones por correo electrónico de los productos y tiendas marcadas como favoritas que se encuentran en promoción o que tienen productos en promoción. Un usuario con rol de “cliente” tiene la opción de evitar que se le envíe notificaciones automáticas para productos marcados como favoritos.

Tabla 14 - US012 – Notificaciones automáticas por ofertas en las tiendas

Fuente: (López Yépez & Polo Vásconez, 2016)

3.14 US013 – Reseñas
ID: US013 Complejidad de Desarrollo: Media Nombre: Reseñas Dependencia: Ninguna Prioridad: Normal Tiempo estimado: 4 horas
Descripción: Teniendo en Cuenta: La aplicación Web y móvil manejarán la opción para agregar reseñas a productos. Cuándo: El usuario con rol “cliente” adquiera un producto a través de una orden de compra Entonces: El sistema le permitirá crear reseñas para esos productos
Criterio de Aceptación:

Un usuario con rol de “cliente” luego de comprar un producto, puede añadir reseñas indicando su experiencia (En un rango de 1 a 5 siendo 1 poco satisfactoria y 5 muy satisfactoria).

Las reseñas son visualizadas por los usuarios con rol “Responsable de tienda” y “cliente”

Las reseñas pueden ser editadas por el usuario que las creó.

El sistema calcula el ranking general de cada producto, en base al promedio de calificaciones de los usuarios con rol “cliente”.

Tabla 15 - US013 – Reseñas

Fuente: (López Yépez & Polo Vásconez, 2016)

3.15 US014 – Interfaz del sistema
ID: US014 Complejidad de Desarrollo: Media Nombre: Interfaz del sistema Dependencia: Ninguna Prioridad: Normal Tiempo estimado: 20 horas
Descripción: Teniendo en Cuenta: La aplicación Web y móvil tendrá una interfaz personalizada Cuándo: Los desarrolladores han finalizado los aspectos funcionales del sistema Entonces: Crearán plantillas (<i>Templates</i>) personalizadas para las aplicaciones web y móvil
Criterio de Aceptación: Las plantillas creadas cumplen con los conceptos de IX (Diseño de Interfaces), UX (Experiencia de Usuario) y IxD (Diseño de Interacción).

Tabla 16 - US014 – Interfaz del sistema

Fuente: (López Yépez & Polo Vásconez, 2016)

4.1 Lista de tareas del Producto

El “Asistente de compras” se desarrolló con la siguiente planificación:

Especificaciones															
ID	Iteración	Estado	Porcentaje	Funcionalidad	Plataforma	Prioridad	Historias de usuario	Criterio de aceptación	Carpeta (Documentación)	Referencias (Documentación)	Comentarios (Especificación)	Tamaño de la historia de usuario (1-5)	Complejidad (1-5)	Listo para trabajar en	US
PBI01				Administración	Web										
PBI01_US000	IT1	Aceptado y Cerrado	100			Urgente	Configuración de ambiente de desarrollo	El entorno está configurado correctamente y se encuentra a disposición de los desarrolladores.	One Drive/ Tesis	Tesisv2.0.1 .docx	Revisar sección anexos del documento para más detalle de las historias de usuario	2	2	5	PBI01_US000 - Configuración de ambiente de desarrollo - (Aceptado y Cerrado)
PBI01_US001	IT1	Aceptado y Cerrado	100			Urgente	*Modelo de dominio	El modelo entidad-relación contiene las tablas y relaciones necesarias para el manejo persistente de los datos que manejaran las dos aplicaciones.	One Drive/ Tesis	Tesisv2.0.1 .docx	Revisar sección anexos del documento para más detalle de las historias de usuario	2	2	5	PBI01_US001 - *Modelo de dominio - (Aceptado y Cerrado)
PBI01_US002	IT1	Aceptado y Cerrado	100			Alta	Gestión de Roles de usuario	1. El área desarrollada permite crear, modificar, eliminar y consultar roles de usuario de la aplicación 2. Permite el acceso a esta área únicamente al Rol “Administrador del sistema”. 3. Los roles creados por defecto en la aplicación son: - Administrador del sistema - Responsable de tienda - Despachador - Cliente	One Drive/ Tesis	Tesisv2.0.1 .docx	Revisar sección apéndice del documento para más detalle de las historias de usuario	3	3	5	PBI01_US002 - Gestión de Roles de usuario - (Aceptado y Cerrado)
PBI01_US003	IT1	Aceptado y Cerrado	100			Alta	Registro de usuarios	Registro de usuarios 1. Existe una página de registro de usuarios 2. El registro de usuarios obliga a confirmar la cuenta para acceder al sistema. 3. Los usuarios registrados tienen la opción de recuperación de su cuenta (Sin importar su rol en el sistema). 4. El área de registro de usuarios únicamente creará usuarios con el rol de “Cliente” Área de gestión de usuarios	One Drive/ Tesis	Tesisv2.0.1 .docx	Revisar sección apéndice del documento para más detalle de las historias de usuario	4	3	5	PBI01_US003 - Registro de usuarios - (Aceptado y Cerrado)

							5. Existe un área donde se puede crear, modificar, inactivar y consultar los usuarios del sistema. 6. Permite el acceso a esta área únicamente al Rol “Administrador del sistema”. Perfil de usuario 7. Crear una página que permita a los usuarios que han accedido a la aplicación, modificar sus datos.							
PBI01_US004	IT3	Aceptado y Cerrado	100			Alta	Gestión de Tiendas 1. El área desarrollada permite crear, modificar, eliminar y consultar datos de las tiendas afiliadas a la aplicación 2. Permite el acceso a esta área únicamente al Rol “Administrador del sistema”. 3. Cuenta con una página de consulta de tiendas disponibles a través de un mapa, en donde se puede acceder a la información de detalle completo por cada tienda. 4. El área de consulta de tiendas estará disponible para todos los roles de usuario definidos en la US002	One Drive/ Tesis	Tesisv2.0.1 .docx	Revisar sección apéndice del documento para más detalle de las historias de usuario	3	3	5	PBI01_US004 - Gestión de Tiendas - (Aceptado y Cerrado)
PBI01_US005	IT3	Aceptado y Cerrado	100			Normal	Gestión de responsables de tiendas 1. El área desarrollada permite asignar responsables a las tiendas afiliadas a la aplicación 2. Permite el acceso a esta área únicamente al Rol “Administrador del sistema”. 3. Cuenta con un acceso directo para crear un usuario responsable a una tienda específica.	One Drive/ Tesis	Tesisv2.0.1 .docx	Revisar sección apéndice del documento para más detalle de las historias de usuario	3	2	5	PBI01_US005 - Gestión de responsables de tiendas - (Aceptado y Cerrado)
PBI01_US006	IT3	Aceptado y Cerrado	100			Normal	Gestión de categorías de productos 1. El área desarrollada permite crear, modificar, eliminar y consultar categorías de productos. 2. Permite el acceso a esta área únicamente al Rol “Administrador del sistema”. 3. Las categorías de producto creadas por defecto son: - Libros - Hogar y cocina - Alimentos - Electrodomésticos - Electrónicos (TV, Audio, Computadoras y dispositivos móviles) - Deportes - Salud - Software (Sistemas operativos, Antivirus, videojuegos)	One Drive/ Tesis	Tesisv2.0.1 .docx	Revisar sección apéndice del documento para más detalle de las historias de usuario	2	1	5	PBI01_US006 - Gestión de categorías de productos - (Aceptado y Cerrado)

PBI02				Productos	Web										
PBI02_US007	IT3	Aceptado y Cerrado	100			Normal	Gestión de productos	Criterio de Aceptación: 1. El área desarrollada permite crear, modificar, eliminar y consultar productos de una tienda específica. 2. Permite el acceso a esta área únicamente al Rol "Responsable de tienda". 3. Existe una opción que permite marcar a un producto como "en promoción" o no.	One Drive/ Tesis	Tesisv2.0.1 .docx	Revisar sección apéndice del documento para más detalle de las historias de usuario	3	2	5	PBI02_US007 - Gestión de productos - (Aceptado y Cerrado)
PBI02_US008	IT4	Aceptado y Cerrado	0			Normal	Búsqueda de productos	1. El área desarrollada permite buscar productos por: - Categoría - Marca - Nombre - Código - Precio - Tienda 2. Permite visualizar el buscador únicamente a los Roles "Responsable de tienda" y "Cliente". 3. Permite añadir los productos consultados al carrito de compras.	One Drive/ Tesis	Tesisv2.0.1 .docx	Revisar sección apéndice del documento para más detalle de las historias de usuario	4	5	5	PBI02_US008 - Búsqueda de productos - (Aceptado y Cerrado)
PBI02_US011	IT4	Aceptado y Cerrado	0			Normal	Productos y tiendas favoritas	1. Un usuario con rol de "Cliente" tiene la opción marcar a una o varias tiendas como favoritas. 2. Un usuario con rol de "Cliente" tiene la opción marcar a uno o varios productos como favoritos. 3. El sistema tiene parametrizado el número de compras realizadas en una determinada tienda y/o producto para asignar de manera automática como favoritos del usuario con rol "Cliente". 4. Existe una página de consulta de los favoritos para cada usuario.	One Drive/ Tesis	Tesisv2.0.1 .docx	Revisar sección apéndice del documento para más detalle de las historias de usuario	2	3	5	PBI02_US011 - Productos y tiendas favoritas - (Aceptado y Cerrado)
PBI02_US013	IT5	Aceptado y Cerrado	0			Normal	Reseñas	1. Un usuario con rol de "Cliente" luego de comprar un producto, puede añadir reseñas indicando su experiencia (En un rango de 1 a 5 siendo 1 poco satisfactoria y 5 muy satisfactoria). 2. Las reseñas son visualizadas por los usuarios con rol "Responsable de tienda" y "Cliente" 3. Las reseñas pueden ser editadas por el usuario que las creó. 4. El sistema calcula el ranking general de cada producto, en base al	One Drive/ Tesis	Tesisv2.0.1 .docx	Revisar sección apéndice del documento para más detalle de las historias de usuario	2	3	5	PBI02_US013 - Reseñas - (Aceptado y Cerrado)

								promedio de calificaciones de los usuarios con rol "Cliente".							
PBI03				Compras y notificaciones	Web										
PBI03_US009	IT5	Aceptado y Cerrado	0			Normal	Carrito de compras	1. Un usuario con rol de "Cliente" tiene la opción de añadir y quitar productos al carrito de compras. 2. Permite usar el carrito de compras únicamente al Rol "Cliente". Se puede aceptar la compra de los productos que están en el carrito de compras.	One Drive/ Tesis	Tesisv2.0.1.docx	Revisar sección apéndice del documento para más detalle de las historias de usuario	5	5	5	PBI03_US009 - Carrito de compras - (Aceptado y Cerrado)
PBI03_US010	IT6	Aceptado y Cerrado	0			Normal	Orden de compra	1. Un usuario con rol de "Cliente" tiene la opción de emitir órdenes de compra a través de la aceptación de los productos de su carrito de compras. 2. El sistema calcula el tiempo máximo para cancelar una orden de compra, en base a los productos que se añadieron al carrito de compras. 3. Permite anular una orden de compra al usuario "cliente" que lo creó, esto dentro del tiempo calculado. 4. El usuario con rol "Despachador" puede consultar las órdenes de compra con estado "Pendiente por revisar (Despachador)" y actualizar las órdenes de compra despachadas con el estado de "Esperando retiro del cliente". 5. Cuando el usuario con rol "Cliente" se acerque a retirar su orden de compra a la tienda, entonces el usuario despachador actualiza el estado de la orden a "Retirado por el cliente". 6. Una vez que se actualiza el campo "estado" de la orden de compra, el sistema emite una notificación al teléfono del cliente y su correo electrónico con los datos de la compra como los productos, el total y el estado de la orden de compra.	One Drive/ Tesis	Tesisv2.0.1.docx	Revisar sección apéndice del documento para más detalle de las historias de usuario	5	5	5	PBI03_US010 - Orden de compra - (Aceptado y Cerrado)

PBI03_US012	IT7	Aceptado y Cerrado	0			Normal	Notificaciones automáticas por ofertas en las tiendas	1. Un usuario con rol de "Cliente" recibe notificaciones por correo electrónico de los productos y tiendas marcadas como favoritas que se encuentran en promoción o que tienen productos en promoción. 2. Un usuario con rol de "Cliente" tiene la opción de evitar que se le envíe notificaciones automáticas para productos marcados como favoritos.	One Drive/ Tesis	Tesisv2.0.1 .docx	Revisar sección apéndice del documento para más detalle de las historias de usuario	3	3	5	PBI03_US012 - Notificaciones automáticas por ofertas en las tiendas - (Aceptado y Cerrado)
PBI04				Interfaz	Web										
PBI04_US014	IT2	Aceptado y Cerrado	0			Normal	Interfaz del sistema	1. Las plantillas creadas cumplen con los conceptos de IX (Diseño de Interfaces), UX (Experiencia de Usuario) y IxD (Diseño de Interacción).	One Drive/ Tesis	Tesisv2.0.1 .docx	Revisar sección apéndice del documento para más detalle de las historias de usuario	5	3	5	PBI04_US014 - Interfaz del sistema - (Aceptado y Cerrado)
PBI05				Administración	Móvil										
PBI05_US000	IT1	Aceptado y Cerrado	0			Normal	Configuración de ambiente de desarrollo	El entorno está configurado correctamente y se encuentra a disposición de los desarrolladores.	One Drive/ Tesis	Tesisv2.0.1 .docx	Revisar sección apéndice del documento para más detalle de las historias de usuario	2	3	5	PBI05_US000 - Configuración de ambiente de desarrollo - (Aceptado y Cerrado)
PBI05_US001	IT1	Aceptado y Cerrado	0			Urgente	Domain Model	El modelo entidad-relación contiene las tablas y relaciones necesarias para el manejo persistente de los datos que manejen las dos aplicaciones.	One Drive/ Tesis	Tesisv2.0.1 .docx	Revisar sección apéndice del documento para más detalle de las historias de usuario	2	2	5	PBI05_US001 - Domain Model - (Aceptado y Cerrado)
PBI05_US003	IT1	Aceptado y Cerrado	0			Alta	Registro de usuarios	Registro de usuarios 1. Existe una página de registro de usuarios 2. El registro de usuarios obliga a confirmar la cuenta para acceder al sistema. 3. Los usuarios registrados tienen la opción de recuperación de su cuenta (Sin importar su rol en el sistema). 4. El área de registro de usuarios únicamente creará usuarios con el rol de "Cliente" Área de gestión de usuarios 5. Existe un área donde se puede crear, modificar, inactivar y consultar los usuarios del sistema. 6. Permite el acceso a esta área únicamente al Rol "Administrador del sistema". Perfil de usuario 7. Crear una página que permita a los usuarios que han accedido a la aplicación, modificar sus datos.	One Drive/ Tesis	Tesisv2.0.1 .docx	Revisar sección apéndice del documento para más detalle de las historias de usuario	4	3	5	PBI05_US003 - Registro de usuarios - (Aceptado y Cerrado)
PBI06				Productos	Móvil										

PBI06_US008	IT3	Aceptado y Cerrado	0			Normal	Búsqueda de productos	1. El área desarrollada permite buscar productos por: - Marca - Nombre - Descripción - Tienda 2. Permite visualizar el buscador únicamente a los Roles “Cliente”. 3. Permite añadir los productos consultados al carrito de compras.	One Drive/ Tesis	Tesisv2.0.1 .docx	Revisar sección apéndice del documento para más detalle de las historias de usuario	3	5	5	PBI06_US008 - Búsqueda de productos - (Aceptado y Cerrado)
PBI06_US011	IT3	Aceptado y Cerrado	0			Normal	Productos y tiendas favoritas	1. Un usuario con rol de “Cliente” tiene la opción marcar a una o varias tiendas como favoritas. 2. Un usuario con rol de “Cliente” tiene la opción marcar a uno o varios productos como favoritos. 3. El sistema tiene parametrizado el número de compras realizadas en una determinada tienda y/o producto para asignar de manera automática como favoritos del usuario con rol “Cliente”. 4. Existe una página de consulta de los favoritos para cada usuario.	One Drive/ Tesis	Tesisv2.0.1 .docx	Revisar sección apéndice del documento para más detalle de las historias de usuario	2	3	5	PBI06_US011 - Productos y tiendas favoritas - (Aceptado y Cerrado)
PBI06_US013	IT3	Aceptado y Cerrado	0			Normal	Reseñas	1. Un usuario con rol de “Cliente” luego de comprar un producto, puede añadir reseñas indicando su experiencia (En un rango de 1 a 5 siendo 1 poco satisfactoria y 5 muy satisfactoria). 2. Las reseñas son visualizadas por los usuarios con rol “Cliente” 3. Las reseñas pueden ser editadas por el usuario que las creó. 4. El sistema calcula el ranking general de cada producto, en base al promedio de calificaciones de los usuarios con rol “Cliente”	One Drive/ Tesis	Tesisv2.0.1 .docx	Revisar sección apéndice del documento para más detalle de las historias de usuario	2	3	5	PBI06_US013 - Reseñas - (Aceptado y Cerrado)
PBI07				Compras y notificaciones	Móvil										
PBI07_US009	IT4	Aceptado y Cerrado	0			Normal	Carrito de compras	1. Un usuario con rol de “Cliente” tiene la opción de añadir y quitar productos al carrito de compras. 2. Permite usar el carrito de compras únicamente al Rol “Cliente”. Se puede aceptar la compra de los productos que están en el carrito de compras.	One Drive/ Tesis	Tesisv2.0.1 .docx	Revisar sección apéndice del documento para más detalle de las historias de usuario	5	5	5	PBI07_US009 - Carrito de compras - (Aceptado y Cerrado)

PBI07_US010	IT4	Aceptado y Cerrado	0			Normal	Orden de compra	1. Un usuario con rol de “Cliente” tiene la opción de emitir órdenes de compra a través de la aceptación de los productos de su carrito de compras. 2. El sistema calcula el tiempo máximo para cancelar una orden de compra, en base a los productos que se añadieron al carrito de compras. 3. Permite anular una orden de compra al usuario “cliente” que lo creó, esto dentro del tiempo calculado. 4. El usuario con rol “Despachador” puede consultar las órdenes de compra con estado “Pendiente por revisar (Despachador)” y actualizar las órdenes de compra despachadas con el estado de “Esperando retiro del cliente”. 5. Cuando el usuario con rol “Cliente” se acerque a retirar su orden de compra a la tienda, entonces el usuario despachador actualiza el estado de la orden a “Retirado por el cliente”. 6. Una vez que se actualiza el campo “estado” de la orden de compra, el sistema emite una notificación al teléfono del cliente y su correo electrónico con los datos de la compra como los productos, el total y el estado de la orden de compra.	One Drive/ Tesis	Tesisv2.0.1 .docx	Revisar sección apéndice del documento para más detalle de las historias de usuario	5	5	5	PBI07_US010 - Orden de compra - (Aceptado y Cerrado)
PBI07_US012	IT5	Aceptado y Cerrado	0			Normal	Notificaciones automáticas por ofertas en las tiendas	1. Un usuario con rol de “Cliente” recibe notificaciones por correo electrónico de los productos y tiendas marcadas como favoritas que se encuentran en promoción o que tienen productos en promoción. 2. Un usuario con rol de “Cliente” tiene la opción de evitar que se le envíe notificaciones automáticas para productos marcados como favoritos.	One Drive/ Tesis	Tesisv2.0.1 .docx	Revisar sección apéndice del documento para más detalle de las historias de usuario	3	3	5	PBI07_US012 - Notificaciones automáticas por ofertas en las tiendas - (Aceptado y Cerrado)
PBI08				Interfaz	Móvil										
PBI08_US014	IT2	Aceptado y Cerrado	0			Urgente	Interfaz del sistema	1. Las plantillas creadas cumplen con los conceptos de IX (Diseño de Interfaces), UX (Experiencia de Usuario) y IxD (Diseño de Interacción).	One Drive/ Tesis	Tesisv2.0.1 .docx	Revisar sección apéndice del documento para más detalle de las historias de usuario	5	3	5	PBI08_US014 - Interfaz del sistema - (Aceptado y Cerrado)

Tabla 17 - Lista de tareas del producto

Fuente: (López Yépez & Polo Vásconez, 2016)

4.2 Iteración 1

La iteración 1 se desarrolló de acuerdo a los requerimientos iniciales de las aplicaciones, siendo estas las tareas que se ejecutaron:

Funcionalidad	Historia de usuario ID	Plataforma	Historia de usuario	Tarea ID	Tarea	Asignado a	Estado	Porcentaje (%)	Esfuerzo estimado (h)	Tiempo por hacer (h)
Administración	PBI01	Web								
	PBI01_US000		PBI01_US000 - Configuración de ambiente de desarrollo - (Aceptado y Cerrado)			Javier Polo	Aceptado y Cerrado	100		
				PBI01_US000_TA001	Instalación de JDK y configuración de variables de entorno java	Javier Polo			1	0,00
				PBI01_US000_TA002	Instalación de WebRatio Enterprise Platform	Javier Polo			0,5	0,00
				PBI01_US000_TA003	Instalación de MySQL Server	Javier Polo			0,5	0,00
				PBI01_US000_TA004	Creación del proyecto	Javier Polo			0,5	0,00
				PBI01_US000_TA005	Configuración de CVS y publicación del proyecto web	Javier Polo			0,5	0,00
	PBI01_US001		PBI01_US001 - *Modelo de dominio - (Aceptado y Cerrado)			Javier Polo	Aceptado y Cerrado	100		
				PBI01_US001_TA001	*Modelo de dominio del proyecto (E-R)	Javier Polo			2,50	0,00
				PBI01_US001_TA002	Sincronización del modelo de dominio con MySQL server	Javier Polo			0,50	0,00
	PBI01_US002		PBI01_US002 - Gestión de Roles de usuario - (Aceptado y Cerrado)			Javier Polo	Aceptado y Cerrado	100		
				PBI01_US002_TA001	Formulario para crear roles de usuario	Javier Polo			0,50	0,00
				PBI01_US002_TA002	Módulo para guardar los roles de usuario	Javier Polo			0,50	0,00
				PBI01_US002_TA003	Formulario para modificar roles de usuario	Javier Polo			0,50	0,00
				PBI01_US002_TA004	Listado de roles de usuario	Javier Polo			0,50	0,00
				PBI01_US002_TA005	Formulario de detalle para roles de usuario	Javier Polo			0,50	0,00
				PBI01_US002_TA006	Módulo para eliminar lógicamente roles de usuario	Javier Polo			0,50	0,00
				PBI01_US002_TA007	Control de permisos al área de roles de usuario	Javier Polo			0,50	0,00
				PBI01_US002_TA008	Crear los roles por defecto	Javier Polo			0,50	0,00
	PBI01_US003		PBI01_US003 - Registro de usuarios - (Aceptado y Cerrado)			Javier Polo	Aceptado y Cerrado	100		
				PBI01_US003_TA001	Master page con enlace para cerrar sesión	Javier Polo			1,00	0,00
				PBI01_US003_TA002	Formulario para el registro de usuarios	Javier Polo			1,00	0,00
				PBI01_US003_TA003	Módulo para guardar los datos de usuario	Javier Polo			0,50	0,00
				PBI01_US003_TA004	Módulo para el envío de correo electrónico para la confirmación de la cuenta	Javier Polo			2,00	0,00
				PBI01_US003_TA005	Formulario y módulo de recuperación de la cuenta	Javier Polo			1,50	0,00
				PBI01_US003_TA006	Formulario para crear usuarios	Javier Polo			0,50	0,00
				PBI01_US003_TA007	Módulo para guardar usuarios	Javier Polo			0,50	0,00
				PBI01_US003_TA008	Formulario para modificar usuarios	Javier Polo			0,50	0,00

Funcionalidad	Historia de usuario ID	Plataforma	Historia de usuario	Tarea ID	Tarea	Asignado a	Estado	Porcentaje (%)	Esfuerzo estimado (h)	Tiempo por hacer (h)
				PBI01_US003_TA009	Listado de usuarios	Javier Polo			0,50	0,00
				PBI01_US003_TA010	Formulario de detalle de usuarios	Javier Polo			0,50	0,00
				PBI01_US003_TA011	Módulo para inactivar usuarios	Javier Polo			0,50	0,00
				PBI01_US003_TA012	Formulario de consulta de perfil de usuario	Javier Polo			1,00	0,00
				PBI01_US003_TA013	Control de permisos al área de gestión de usuarios	Javier Polo			1,00	0,00
Administración	PBI05	Móvil								
	PBI05_US000		PBI05_US000 - Configuración de ambiente de desarrollo - (Aceptado y Cerrado)			Xavier López	Aceptado y Cerrado	100		
				PBI05_US000_TA001	Instalación de JDK y configuración de variables de entorno java	Xavier López			1	0,00
				PBI05_US000_TA002	Instalación de WebRatio Professional Platform	Xavier López			0,5	0,00
				PBI05_US000_TA003	Instalación de MySQL Server	Xavier López			0,5	0,00
				PBI05_US000_TA004	Creación del proyecto móvil	Xavier López			0,5	0,00
				PBI05_US000_TA005	Creación del proyecto data service	Xavier López			0,5	0,00
				PBI05_US000_TA006	Configuración de CVS y publicación del proyecto móvil	Xavier López			1	0,00
				PBI05_US000_TA007	Configuración de CVS y publicación del proyecto data service	Xavier López			1	0,00
	PBI05_US001		PBI05_US001 - Domain Model - (Aceptado y Cerrado)			Xavier López	Aceptado y Cerrado	100		
				PBI05_US001_TA001	Sincronización del BD con proyecto Data Service	Xavier López			1,00	0,00
				PBI05_US001_TA002	Sincronización del modelo de dominio con MySQL server	Xavier López			1,00	0,00
	PBI05_US003		PBI05_US003 - Registro de usuarios - (Aceptado y Cerrado)			Xavier López	Aceptado y Cerrado	100		
				PBI05_US003_TA001	Toolbar con enlace al registro de usuarios	Xavier López			1,00	0,00
				PBI05_US003_TA002	Formulario para el registro de usuarios	Xavier López			2,00	0,00
				PBI05_US003_TA003	Módulo para guardar los datos de usuario	Xavier López			2,00	0,00
				PBI05_US003_TA004	Módulo para el envío de correo electrónico para la confirmación de la cuenta	Xavier López			2,00	0,00
				PBI05_US003_TA005	Formulario y módulo de recuperación de la cuenta de usuario	Xavier López			2,00	0,00
				PBI05_US003_TA006	Formulario de consulta de perfil de usuario	Xavier López			1,00	0,00
				PBI05_US003_TA007	Formulario de edición de datos de usuario	Xavier López			2,00	0,00
				PBI05_US003_TA008	Validaciones formulario registro de usuarios	Xavier López			2,00	0,00

Tabla 18 - Iteración 1

Fuente: (López Yépez & Polo Vásconez, 2016)

Estas tareas de acuerdo a su tiempo de esfuerzo estimado, contrario al tiempo ejecutado produjo la siguiente quema de horas (Gráfico de trabajo pendiente):

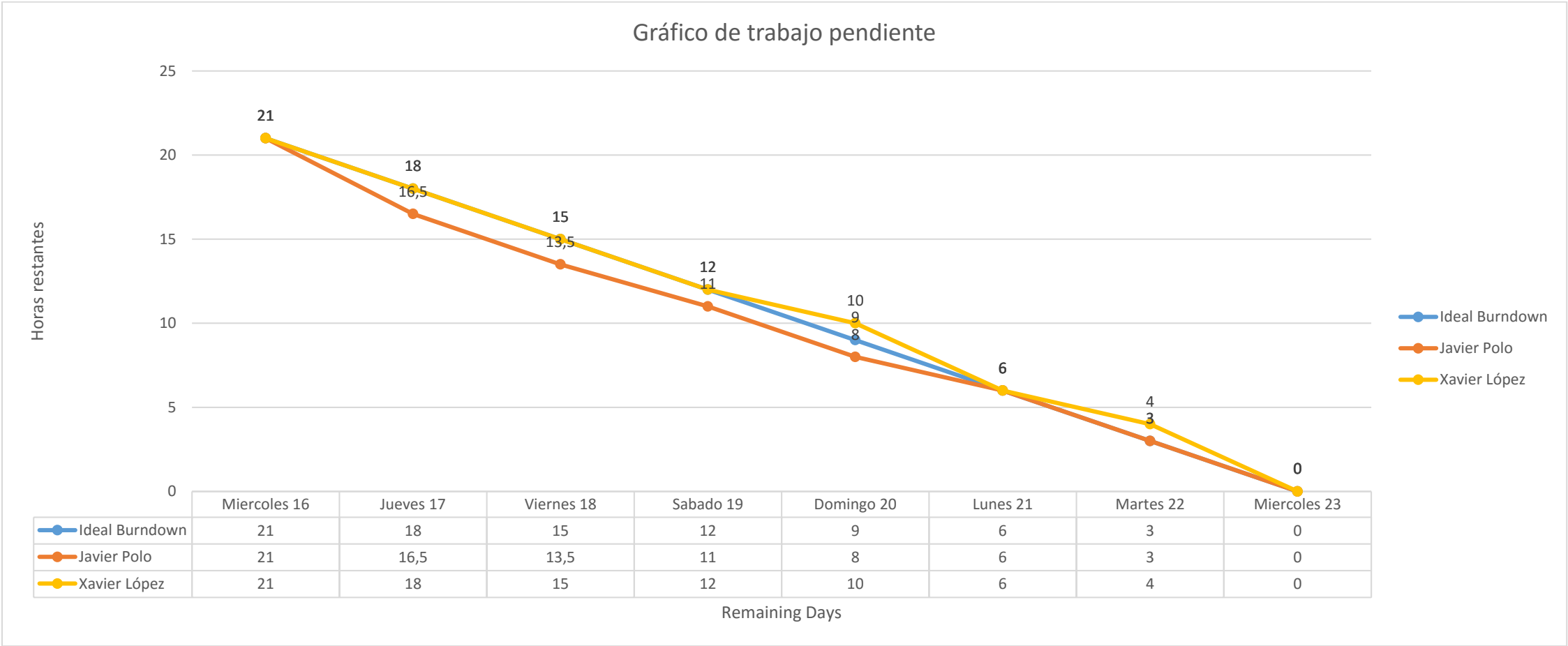


Gráfico 91 - Gráfico de trabajo pendiente de la iteración 1

Fuente: (López Yépez & Polo Vásconez, 2016)

4.3 Iteración 2

La iteración 2 se desarrolló de acuerdo a los requerimientos iniciales de las aplicaciones, siendo estas las tareas que se ejecutaron:

Funcionalidad	Historia de usuario ID	Plataforma	Historia de usuario	Tarea ID	Tarea	Asignado a	Estado	Porcentaje (%)	Esfuerzo estimado (h)	Tiempo por hacer (h)
Interfaz	PBI04	Web								
	PBI04_US014		PBI04_US014 - Interfaz del sistema - (Aceptado y Cerrado)			Javier Polo	Aceptado y Cerrado	100		
				PBI04_US014_TA001	Configuración del proyecto de estilos bootstrap	Javier Polo			4	0,00
				PBI04_US014_TA002	Plantilla de página general	Javier Polo			6	0,00
				PBI04_US014_TA003	Plantilla de página inicio de sesión	Javier Polo			5	0,00
				PBI04_US014_TA004	Plantilla de rastro de la web	Javier Polo			3	0,00
				PBI04_US014_TA005	Plantilla de usuario actual	Javier Polo			2	0,00
Interfaz	PBI08	Móvil								
	PBI08_US014		PBI08_US014 - Interfaz del sistema - (Aceptado y Cerrado)			Xavier López	Aceptado y Cerrado	100		
				PBI08_US014_TA001	Configuración del proyecto de estilos	Xavier López			2	0,00
				PBI08_US014_TA002	Plantilla de pantalla de inicio de sesión	Xavier López			2	0,00
				PBI08_US014_TA003	Plantilla de usuario actual	Xavier López			2	0,00
				PBI08_US014_TA004	Plantilla de pantalla de registro/edición de usuarios	Xavier López			2	0,00
				PBI08_US014_TA005	Plantilla de pantalla de recuperación de contraseña	Xavier López			2	0,00
				PBI08_US014_TA006	Plantilla de pantalla de detalle de producto	Xavier López			2	0,00
				PBI08_US014_TA007	Plantilla de pantalla de detalle de tienda	Xavier López			2	0,00
				PBI08_US014_TA008	Plantilla de pantalla de listado de productos	Xavier López			2	0,00
				PBI08_US014_TA009	Plantilla de pantalla de listado de reseñas	Xavier López			2	0,00
				PBI08_US014_TA010	Plantilla de pantalla de detalle Órdenes de Compra	Xavier López			2	0,00
				PBI08_US014_TA011	Plantilla de menú lateral	Xavier López			2	0,00

Tabla 19 - Iteración 2

Fuente: (López Yépez & Polo Vásconez, 2016)

Estas tareas de acuerdo a su tiempo de esfuerzo estimado, contrario al tiempo ejecutado produjo la siguiente quema de horas (Gráfico de trabajo pendiente):

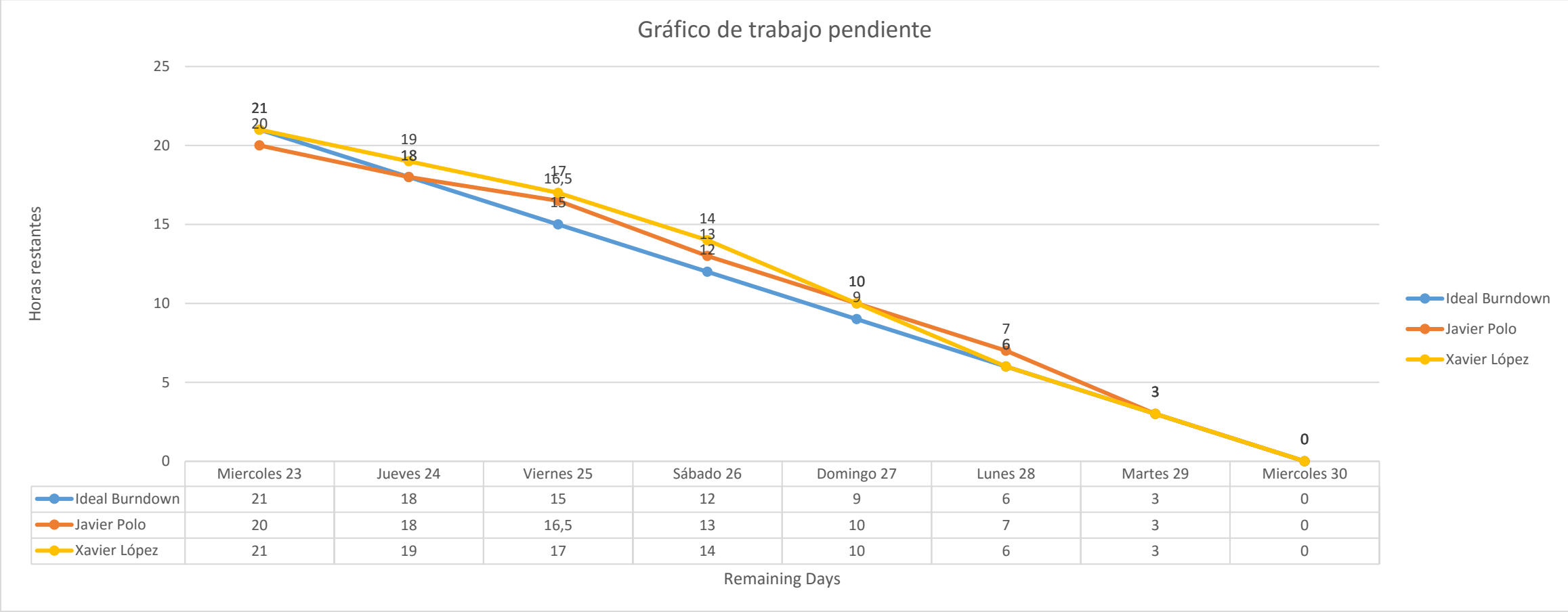


Gráfico 92 – Gráfico de trabajo pendiente de la iteración 2

Fuente: (López Yépez & Polo Vásconez, 2016)

4.4 Iteración 3

La iteración 3 se desarrolló de acuerdo a los requerimientos iniciales de las aplicaciones, siendo estas las tareas que se ejecutaron:

Funcionalidad	Historia de usuario ID	Plataforma	Historia de usuario	Tarea ID	Tarea	Asignado a	Estado	Porcentaje (%)	Esfuerzo estimado (h)	Tiempo por hacer (h)
Administración	PBI01	Web								
	PBI01_US004		PBI01_US004 - Gestión de Tiendas - (Aceptado y Cerrado)			Javier Polo	Aceptado y Cerrado	100		
				PBI01_US004_TA001	Formulario para crear tiendas	Javier Polo			1,00	0,00
				PBI01_US004_TA002	Módulo para guardar las tiendas	Javier Polo			0,50	0,00
				PBI01_US004_TA003	Formulario para modificar tiendas	Javier Polo			0,50	0,00
				PBI01_US004_TA004	Listado de tiendas	Javier Polo			1,00	0,00
				PBI01_US004_TA005	Formulario de detalle para tiendas	Javier Polo			1,00	0,00
				PBI01_US004_TA006	Visualización de tiendas desde mapa	Javier Polo			4,00	0,00
				PBI01_US004_TA007	Control de permisos al área de tiendas	Javier Polo			0,50	0,00
				PBI01_US004_TA008	Formulario para crear sucursales	Javier Polo			1,00	0,00
	PBI01_US005		PBI01_US005 - Gestión de responsables de tiendas - (Aceptado y Cerrado)			Javier Polo	Aceptado y Cerrado	100		
				PBI01_US005_TA001	Modificar formulario Crear/Editar usuarios para permitir guardar responsables de tiendas	Javier Polo			0,50	0,00
				PBI01_US005_TA002	Modificar formulario detalle de usuarios para permitir guardar responsables de tiendas	Javier Polo			0,50	0,00
				PBI01_US005_TA003	Pruebas funcionales	Javier Polo			1,00	0,00
	PBI01_US006		PBI01_US006 - Gestión de categorías de productos - (Aceptado y Cerrado)			Javier Polo	Aceptado y Cerrado	100		
				PBI01_US006_TA001	Formulario para crear categorías de productos	Javier Polo			1,00	0,00
				PBI01_US006_TA002	Módulo para guardar las categorías de productos	Javier Polo			0,50	0,00
				PBI01_US006_TA003	Formulario para modificar categorías de productos	Javier Polo			0,50	0,00
				PBI01_US006_TA004	Listado de categorías de productos	Javier Polo			0,50	0,00
				PBI01_US006_TA005	Formulario de detalle para categorías de productos	Javier Polo			0,50	0,00
				PBI01_US006_TA006	Control de permisos al área de categorías de productos	Javier Polo			0,50	0,00
				PBI01_US006_TA007	Crear las categorías de productos por defecto	Javier Polo			1,00	0,00
Productos	PBI02	Web								
	PBI02_US007		PBI02_US007 - Gestión de productos - (Aceptado y Cerrado)			Javier Polo	Aceptado y Cerrado	100		
				PBI02_US007_TA001	Formulario para crear productos	Javier Polo			1,00	0,00
				PBI02_US007_TA002	Módulo para guardar productos	Javier Polo			1,00	0,00
				PBI02_US007_TA003	Formulario para modificar productos	Javier Polo			0,50	0,00
				PBI02_US007_TA004	Listado de productos	Javier Polo			1,00	0,00
				PBI02_US007_TA005	Formulario de detalle de productos	Javier Polo			1,00	0,00
				PBI02_US007_TA006	Control de permisos al área de productos	Javier Polo			0,50	0,00
Productos	PBI06	Móvil								

	PBI06_US008		PBI06_US008 - Búsqueda de productos - (Aceptado y Cerrado)			Xavier López	Aceptado y Cerrado	100		
			PBI06_US008_TA011	Lista de productos	Xavier López				1,00	0,00
			PBI06_US008_TA012	Plantilla para componente de lista para mostrar los productos	Xavier López				4,00	0,00
			PBI06_US008_TA013	Modelado búsqueda de productos	Xavier López				2,00	0,00
	PBI06_US011		PBI06_US011 - Productos y tiendas favoritas - (Aceptado y Cerrado)			Xavier López	Aceptado y Cerrado	100		
			PBI06_US011_TA001	Incluir opciones de Productos y Tiendas Favoritas al menú	Xavier López				1,00	0,00
			PBI06_US011_TA002	Módulo para añadir productos como favoritos	Xavier López				1,00	0,00
			PBI06_US011_TA003	Módulo para añadir tiendas como favoritas	Xavier López				1,00	0,00
			PBI06_US011_TA004	Módulo para quitar productos como favoritos	Xavier López				1,00	0,00
			PBI06_US011_TA005	Módulo para quitar tiendas como favoritas	Xavier López				1,00	0,00
			PBI06_US011_TA006	Incluir botones de Añadir y Quitar productos como favoritos desde detalle de producto	Xavier López				1,00	0,00
			PBI06_US011_TA007	Incluir botones de Añadir y Quitar tiendas como favoritas desde detalla del producto	Xavier López				1,00	0,00
			PBI06_US011_TA008	Lista de productos favoritos	Xavier López				1,00	0,00
			PBI06_US011_TA009	Lista de tiendas favoritas	Xavier López				1,00	0,00
	PBI06_US013		PBI06_US013 - Reseñas - (Aceptado y Cerrado)			Xavier López	Aceptado y Cerrado	100		
			PBI06_US013_TA001	Formulario para crear reseñas	Xavier López				2,00	0,00
			PBI06_US013_TA002	Formulario para editar reseñas	Xavier López				1,00	0,00
			PBI06_US013_TA003	Módulo para guardar reseñas	Xavier López				2,00	0,00

Tabla 20 - Iteración 3

Fuente: (López Yépez & Polo Vásconez, 2016)

Estas tareas de acuerdo a su tiempo de esfuerzo estimado, contrario al tiempo ejecutado produjo la siguiente quema de horas (Gráfico de trabajo pendiente):

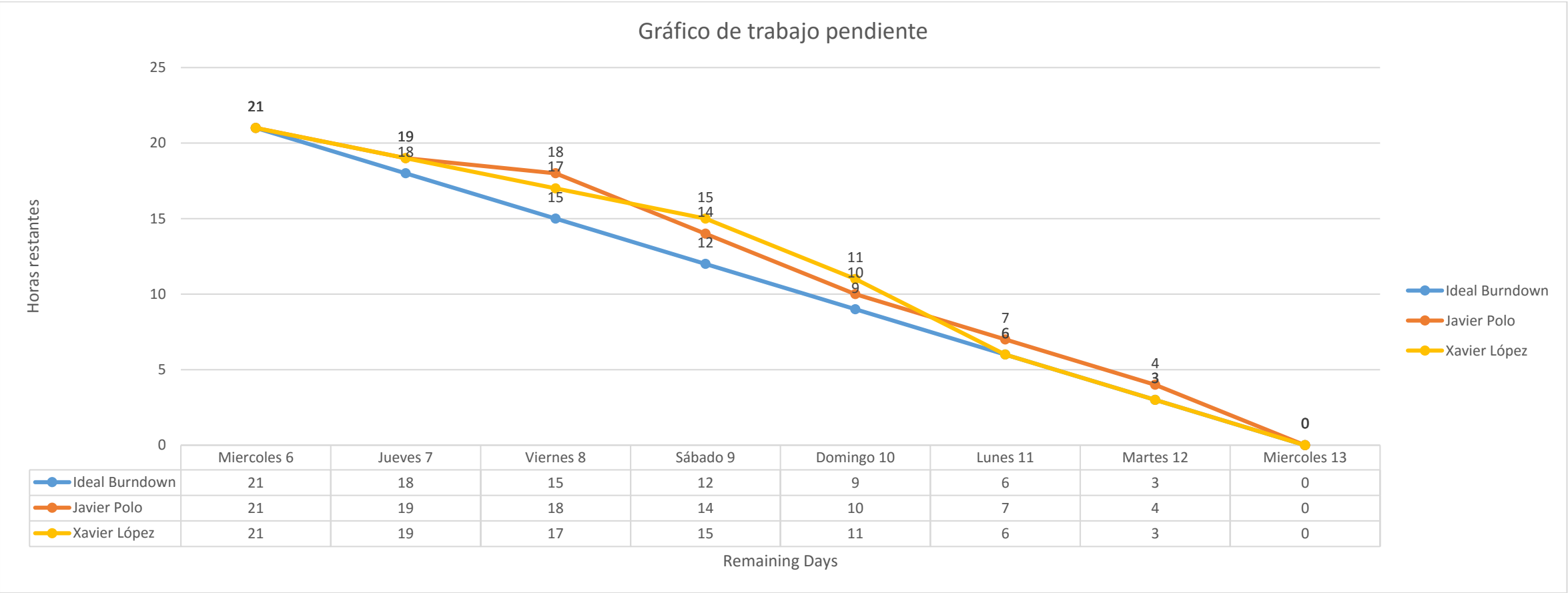


Gráfico 93 – Gráfico de trabajo pendiente de la iteración 3

Fuente: (López Yépez & Polo Vásconez, 2016)

4.5 Iteración 4

La iteración 4 se desarrolló de acuerdo a los requerimientos iniciales de las aplicaciones, siendo estas las tareas que se ejecutaron:

Funcionalidad	Historia de usuario ID	Plataforma	Historia de usuario	Tarea ID	Tarea	Asignado a	Estado	Porcentaje (%)	Esfuerzo estimado (h)	Tiempo por hacer (h)
Productos	PBI02	Web								
	PBI02_US008		PBI02_US008 - Búsqueda de productos - (Aceptado y Cerrado)			Javier Polo	Aceptado y Cerrado	100		
				PBI02_US008_TA001	Lista de productos	Javier Polo			1,00	0,00
				PBI02_US008_TA002	Plantilla para componente de lista para mostrar los productos	Javier Polo			10,00	0,00
				PBI02_US008_TA003	Modelado búsqueda de productos	Javier Polo			2,00	0,00
	PBI02_US011		PBI02_US011 - Productos y tiendas favoritas - (Aceptado y Cerrado)			Javier Polo	Aceptado y Cerrado	100		
				PBI02_US011_TA001	Módulo para añadir productos como favoritos	Javier Polo			1,00	0,00
				PBI02_US011_TA002	Módulo para añadir tiendas como favoritas	Javier Polo			1,00	0,00
				PBI02_US011_TA003	Módulo para quitar productos como favoritos	Javier Polo			1,00	0,00
				PBI02_US011_TA004	Módulo para quitar tiendas como favoritas	Javier Polo			1,00	0,00
				PBI02_US011_TA005	Condiciones de visibilidad de los botones Añadir y Quitar productos como favoritos	Javier Polo			2,00	0,00
				PBI02_US011_TA006	Condiciones de visibilidad de los botones Añadir y Quitar tiendas como favoritas	Javier Polo			2,00	0,00
Compras y notificaciones	PBI07	Móvil								
	PBI07_US012		PBI07_US012 - Notificaciones automáticas por ofertas en las tiendas - (Aceptado y Cerrado)			Xavier López	Aceptado y Cerrado	100		
				PBI07_US012_TA001	Adquisición Cuenta Apple Store para desarrolladores	Xavier López			2,00	0,00
				PBI07_US012_TA002	Configuración de equipos de prueba para aplicaciones iOS	Xavier López			3,00	0,00
				PBI07_US012_TA003	Configuración de certificados para uso de notificaciones push en aplicaciones iOS	Xavier López			3,00	0,00
				PBI07_US012_TA004	Configuración de cuenta Google para desarrolladores	Xavier López			2,00	0,00
				PBI07_US012_TA005	Generación de Api Key para uso de notificaciones push en aplicaciones Android	Xavier López			2,00	0,00
				PBI07_US012_TA006	Creación de servicio web para notificaciones de productos y promociones	Xavier López			2,00	0,00
				PBI07_US012_TA007	Creación de servicio web para notificaciones de órdenes de compra	Xavier López			2,00	0,00
				PBI07_US012_TA008	Implementación de componente de notificación en aplicación móvil	Xavier López			2,00	0,00
				PBI07_US012_TA009	Pruebas de funcionamiento en dispositivos móviles	Xavier López			2,00	0,00

Tabla 21 - Iteración 4

Fuente: (López Yépez & Polo Vásconez, 2016)

Estas tareas de acuerdo a su tiempo de esfuerzo estimado, contrario al tiempo ejecutado produjo la siguiente quema de horas (Gráfico de trabajo pendiente):

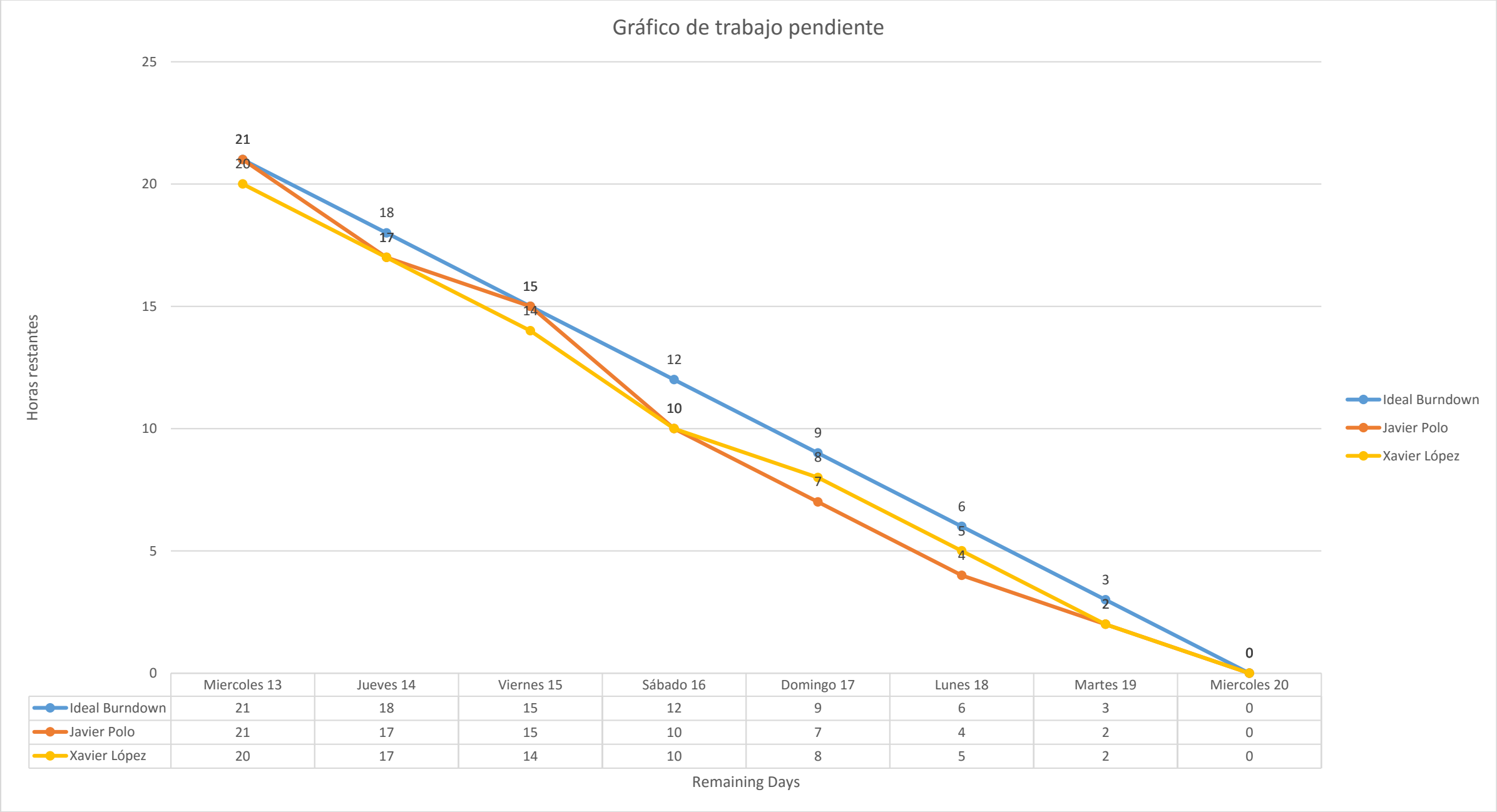


Gráfico 94 – Gráfico de trabajo pendiente de la iteración 4
Fuente: (López Yépez & Polo Vásconez, 2016)

4.6 Iteración 5

La iteración 5 se desarrolló de acuerdo a los requerimientos iniciales de las aplicaciones, siendo estas las tareas que se ejecutaron:

Funcionalidad	Historia de usuario ID	Plataforma	Historia de usuario	Tarea ID	Tarea	Asignado a	Estado	Porcentaje (%)	Esfuerzo estimado (h)	Tiempo por hacer (h)
Productos	PBI02	Web								
	PBI02_US013		PBI02_US013 - Reseñas - (Aceptado y Cerrado)			Javier Polo	Aceptado y Cerrado	100		
				PBI02_US013_TA001	Lista de reseñas	Javier Polo			3,00	0,00
				PBI02_US013_TA002	Template de atributo para mostrar la calificación del producto	Javier Polo			2,00	0,00
				PBI02_US013_TA003	Template de campo para editar la calificación del producto	Javier Polo			2,00	0,00
				PBI02_US013_TA004	Formulario para crear reseñas	Javier Polo			0,50	0,00
				PBI02_US013_TA005	Formulario para editar reseñas	Javier Polo			0,50	0,00
				PBI02_US013_TA006	Módulo para guardar reseñas	Javier Polo			1,00	0,00
Compras y notificaciones	PBI03	Web								
	PBI03_US009		PBI03_US009 - Carrito de compras - (Aceptado y Cerrado)			Javier Polo	Aceptado y Cerrado	100		
				PBI03_US009_TA001	Módulo para añadir productos al carrito de compras	Javier Polo			4,00	0,00
				PBI03_US009_TA002	Módulo para quitar productos del carrito de compras	Javier Polo			3,00	0,00
				PBI03_US009_TA003	Formulario de detalle del carrito de compras	Javier Polo			2,00	0,00
				PBI03_US009_TA004	Módulo para modificar la cantidad de un producto del carrito de compras	Javier Polo			3,00	0,00
Compras y notificaciones	PBI07	Móvil								
	PBI07_US009		PBI07_US009 - Carrito de compras - (Aceptado y Cerrado)			Xavier López	Aceptado y Cerrado	100		
				PBI07_US009_TA001	Módulo para añadir productos al carrito de compras	Xavier López			4,00	0,00
				PBI07_US009_TA002	Módulo para quitar productos del carrito de compras	Xavier López			3,00	0,00
				PBI07_US009_TA003	Servicio web para añadir productos al carrito de compras	Xavier López			2,00	0,00
				PBI07_US009_TA004	Servicio web para quitar productos al carrito de compras	Xavier López			2,00	0,00
				PBI07_US009_TA005	Formulario de detalle del carrito de compras	Xavier López			2,00	0,00
				PBI07_US009_TA006	Módulo para modificar la cantidad de un producto del carrito de compras	Xavier López			3,00	0,00
				PBI07_US009_TA007	Servicio web para modificar la cantidad de un producto del carrito de compras	Xavier López			2,00	0,00
				PBI07_US009_TA008	Servicio web para actualizar totales cuando modifica el carrito de compras	Xavier López			2,00	0,00
				PBI07_US009_TA009	Pruebas de funcionamiento de Servicio web y componente en general	Xavier López			1,00	0,00

Tabla 22 - Iteración 5

Fuente: (López Yépez & Polo Vásconez, 2016)

Estas tareas de acuerdo a su tiempo de esfuerzo estimado, contrario al tiempo ejecutado produjo la siguiente quema de horas (Gráfico de trabajo pendiente):

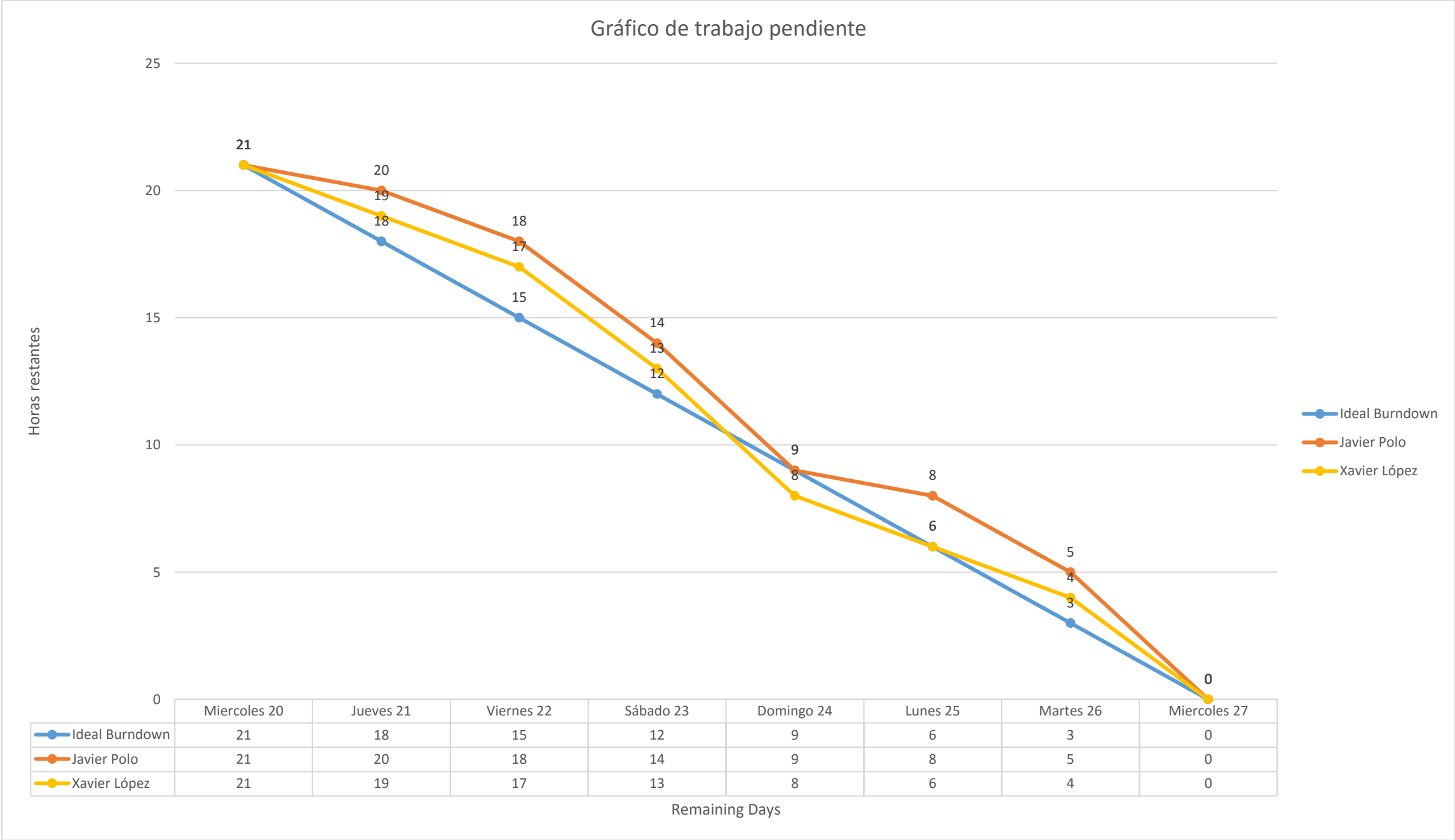


Gráfico 95 – Gráfico de trabajo pendiente de la iteración 5

Fuente: (López Yépez & Polo Vásconez, 2016)

4.7 Iteración 6

La iteración 6 se desarrolló de acuerdo a los requerimientos iniciales de las aplicaciones, siendo estas las tareas que se ejecutaron:

Funcionalidad	Historia de usuario ID	Plataforma	Historia de usuario	Tarea ID	Tarea	Asignado a	Estado	Porcentaje (%)	Esfuerzo estimado (h)	Tiempo por hacer (h)
Compras y notificaciones	PBI03	Web								
	PBI03_US009		PBI03_US009 - Carrito de compras - (Aceptado y Cerrado)			Javier Polo	Aceptado y Cerrado	100		
				PBI03_US009_TA005	Módulo de detalle del producto del carrito de compras	Javier Polo			2,00	0,00
				PBI03_US009_TA006	Plantilla de lista para el carrito de compras	Javier Polo			6,00	0,00
				PBI03_US009_TA007	Pruebas funcionales	Javier Polo			2,00	0,00
	PBI03_US010		PBI03_US010 - Orden de compra - (Aceptado y Cerrado)			Javier Polo	Aceptado y Cerrado	100		
				PBI03_US010_TA001	Módulo para generar órdenes de compra	Javier Polo			5,00	0,00
				PBI03_US010_TA002	Listado de mis órdenes de compra	Javier Polo			3,00	0,00
				PBI03_US010_TA003	Módulo para cancelar(eliminar) órdenes de compra	Javier Polo			2,00	0,00
				PBI03_US010_TA004	Pruebas funcionales	Javier Polo			1,00	0,00
Compras y notificaciones	PBI07	Móvil								
	PBI07_US010		PBI07_US010 - Orden de compra - (Aceptado y Cerrado)			Xavier López	Aceptado y Cerrado	100		
				PBI07_US010_TA001	Módulo para generar órdenes de compra	Xavier López			5,00	0,00
				PBI07_US010_TA002	Listado de mis órdenes de compra				3,00	0,00
				PBI07_US010_TA003	Módulo para cancelar(eliminar) órdenes de compra				3,00	0,00
				PBI07_US010_TA004	Servicio web para generar órdenes de compra				2,00	0,00
				PBI07_US010_TA005	Servicio web para cancelar órdenes de compra				2,00	0,00
				PBI07_US010_TA006	Pantalla para detalle de las órdenes de compra				2,00	0,00
				PBI07_US010_TA007	Pantalla para detalle de los productos de la orden de compra				2,00	0,00
				PBI07_US010_TA008	Pruebas funcionales				2,00	0,00

Tabla 23 - Iteración 6

Fuente: (López Yépez & Polo Vásconez, 2016)

Estas tareas de acuerdo a su tiempo de esfuerzo estimado, contrario al tiempo ejecutado produjo la siguiente quema de horas (Gráfico de trabajo pendiente):

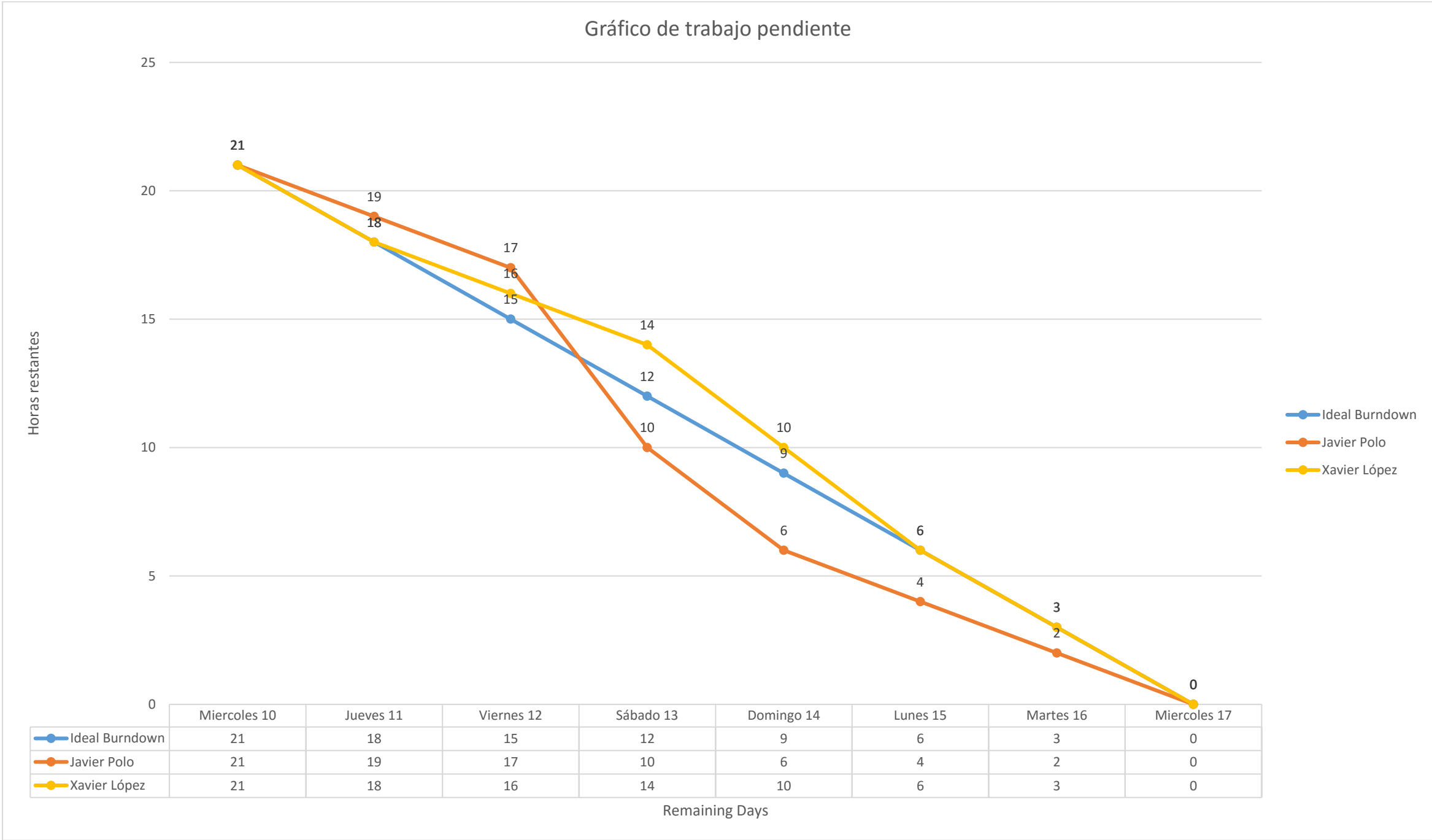


Gráfico 96 – Gráfico de trabajo pendiente de la iteración 6

Fuente: (López Yépez & Polo Vásconez, 2016)

4.8 Iteración 7

La iteración 7 se desarrolló de acuerdo a los requerimientos iniciales de las aplicaciones, siendo estas las tareas que se ejecutaron:

Funcionalidad	Historia de usuario ID	Plataforma	Historia de usuario	Tarea ID	Tarea	Asignado a	Estado	Porcentaje (%)	Esfuerzo estimado (h)	Tiempo por hacer (h)
Compras y notificaciones	PBI03	Web								
	PBI03_US010		PBI03_US010 - Orden de compra - (Aceptado y Cerrado)			Javier Polo	Aceptado y Cerrado	100		
				PBI03_US010_TA001	Módulo para actualizar el estado de las órdenes de compra	Javier Polo			2,00	0,00
				PBI03_US010_TA002	Plantilla de lista para las órdenes de compra	Javier Polo			5,00	0,00
				PBI03_US010_TA003	Listado de órdenes de compra (usuario despachador)	Javier Polo			3,00	0,00
				PBI03_US010_TA004	Pruebas funcionales	Javier Polo			1,00	0,00
	PBI02_US013		PBI03_US012 - Notificaciones automáticas por ofertas en las tiendas - (Aceptado y Cerrado)			Javier Polo	Aceptado y Cerrado	100		
				PBI03_US012_TA001	Módulo para enviar notificaciones	Javier Polo			4,00	0,00
				PBI03_US012_TA002	Configuración de la tarea programada	Javier Polo			1,00	0,00
				PBI03_US012_TA003	Plantilla .mt para la notificación por correo electrónico	Javier Polo			3,00	0,00
				PBI03_US012_TA004	Pruebas funcionales	Javier Polo			2,00	0,00
Interfaz	PBI08	Móvil								
	PBI08_US014		PBI08_US014 - Interfaz del sistema - (Aceptado y Cerrado)			Xavier López	Completado	100		
				PBI08_US014_TA012	Plantilla Componente Detalles - Usuario	Xavier López			2,00	0,00
				PBI08_US014_TA013	Plantilla Componente Lista - Productos	Xavier López			3,00	0,00
				PBI08_US014_TA014	Plantilla Componente Detalle - Tienda	Xavier López			4,00	0,00
				PBI08_US014_TA015	Plantilla Componente Detalle - Reseña	Xavier López			3,00	0,00
				PBI08_US014_TA016	Plantilla Componente Formulario - Reseña	Xavier López			2,00	0,00
				PBI08_US014_TA017	Plantilla Componente Detalle - Localización tiendas	Xavier López			4,00	0,00
				PBI08_US014_TA018	Pruebas funcionales	Xavier López			3,00	0,00

Tabla 24 - Iteración 7

Fuente: (López Yépez & Polo Vásconez, 2016)

Estas tareas de acuerdo a su tiempo de esfuerzo estimado, contrario al tiempo ejecutado produjo la siguiente quema de horas (Gráfico de trabajo pendiente):

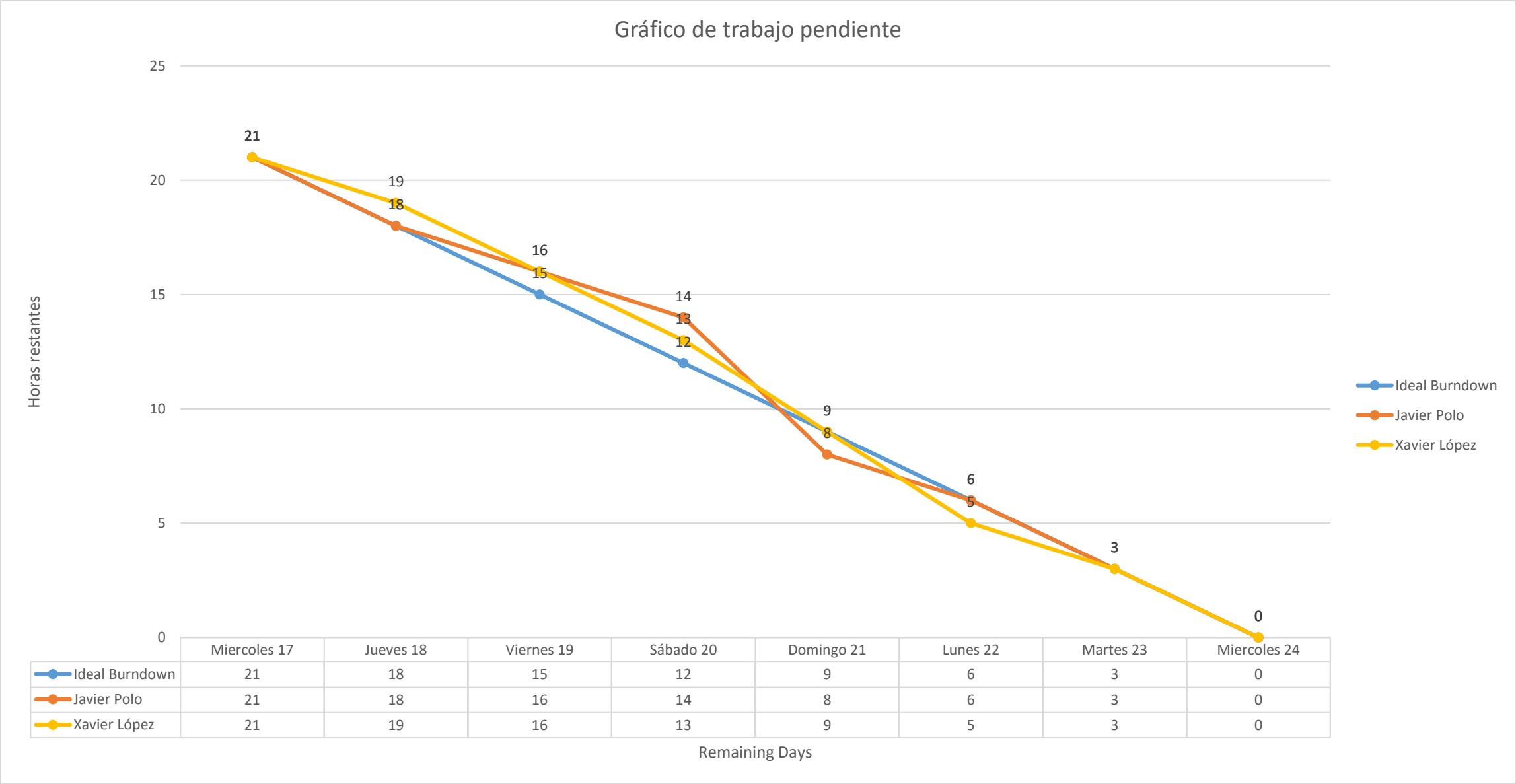


Gráfico 97 – Gráfico de trabajo pendiente de la iteración 7

Fuente: (López Yépez & Polo Vásconez, 2016)