



**FACULTAD DE INGENIERIA
ESCUELA DE SISTEMAS**

**DISERTACIÓN PREVIA A LA OBTENCIÓN DEL TÍTULO DE
INGENIERO EN SISTEMAS**

**CONSTRUCCIÓN DE UN SISTEMA PARA CANJE
VIRTUAL DE PRODUCTOS A TRAVÉS DE
CÓDIGOS QR COMO MECANISMO DE INGRESO,
UTILIZANDO XP COMO METODOLOGÍA DE
DESARROLLO**

ANTONIO CÓRDOVA

DIRECTORA: ING. ANA MARIA URGILÉS

QUITO, 2016

DEDICATORIA

La presente disertación de grado y posterior obtención del título de Ingeniero en Sistemas se la dedico con todo mi amor y cariño a mi hijo José Martín, quien ha sido mi inspiración para salir adelante y encontrar un mejor mañana.

¡VA POR TI, MARTIN!

AGRADECIMIENTO

Agradezco a Dios, en primer lugar, por permitirme llegar hasta donde estoy y por no dejarme caer en aquellos duros momentos de mi vida.

Agradezco a mis padres, Yolanda y Adolfo, por darme la vida, por todas sus enseñanzas de vida y sobre todo por enseñarme que las cosas se hacen con amor y paciencia.

Agradezco de todo corazón a María Mercedes por creer en mí, por toda la ayuda que me dió en su momento

Agradezco a aquellas personas que me tendieron una mano a lo largo de mi carrera para no decaer y continuar.

Agradezco a mis profesores de la facultad, en especial a aquellos que me enriquecieron con sus experiencias más que con sus conocimientos teóricos. Aquellos que me animaron a emprender y dejar de ser uno más del montón.

Agradezco a mi directora de grado, Anita, de quien aprendí que hay que luchar, para conseguir algo, no desmayar cuando ya falta poco y quien me ayudó a superar esta última etapa de mi carrera.

ÍNDICE CONTENIDO

1. METODOS PARA DECODIFICAR CÓDIGOS QR.....	14
1.1. ORIGEN.....	14
1.2. TIPOS DE CÓDIGOS QR	17
1.2.1. <i>Modelo 1- 2.....</i>	18
1.2.2. <i>Micro código QR</i>	19
1.2.3. <i>Código iQR</i>	20
1.2.4. <i>Código SQRC</i>	21
1.2.5. <i>Frame QR.....</i>	22
1.3. GENERACIÓN DE CÓDIGOS QR	22
1.3.1. <i>Configurar el tamaño del módulo.....</i>	22
1.3.2. <i>Determinar el área de código</i>	25
1.3.3. <i>Procedimiento de codificación.....</i>	25
1.4. DECODIFICACIÓN CÓDIGOS QR	26
1.4.1. <i>Problemas encontrados</i>	27
1.5. APLICACIONES	28
1.5.1. <i>Para la educación</i>	28
1.5.2. <i>Para ayuda social.....</i>	29
1.5.3. <i>Para el trabajo diario.....</i>	29
1.5.4. <i>Para el hogar</i>	30
1.5.5. <i>Para el área de la salud</i>	30
1.6. SISTEMAS QUE UTILIZAN LA GENERACIÓN DE CÓDIGOS QR.....	30
2. ANÁLISIS DE LA METODOLOGÍA ÁGIL.....	32
2.1. INTRODUCCIÓN	32
2.1.1. <i>Ciclo de vida.....</i>	32
2.1.2. <i>El Modelo Evolutivo</i>	33
2.2. METODOLOGÍA DE DESARROLLO: EXTREME PROGRAMMING (XP).....	36
2.2.1. <i>Metodología Ágil</i>	36
2.2.2. <i>Extreme Programming (XP)</i>	39
2.2.3. <i>Justificación de la metodología de desarrollo.....</i>	47
2.2.4. <i>Costo del Proyecto Programado</i>	47
3. CASO DE ESTUDIO: APLICACIÓN DE XP EN LA CONSTRUCCIÓN DEL SISTEMA.....	50
3.1. FASE DE EXPLORACIÓN.....	50
3.1.1. <i>Historias de usuario</i>	51

3.1.2.	<i>Spike arquitectónico</i>	54
3.1.3.	<i>Metáfora del negocio</i>	55
3.2.	FASE DE PLANIFICACIÓN	56
3.2.1.	<i>Estimación de costos</i>	57
3.2.2.	<i>Pruebas de aceptación</i>	58
3.2.3.	<i>Plan de entrega</i>	62
3.3.	FASE DE ITERACIONES	63
3.3.1.	<i>Iteración inicial</i>	65
3.3.2.	<i>Iteración 01 – Primer paso a producción</i>	67
3.3.3.	<i>Iteración 02 – Gestión básica de productos</i>	71
3.3.4.	<i>Iteración 03 – Gestión de usuarios</i>	86
3.3.5.	<i>Iteración 04 - Gestión básica de paquetes</i>	94
3.3.6.	<i>Iteración 05 - Gestión básica de empresas</i>	100
3.3.7.	<i>Iteración 05 - Selección de productos</i>	107
3.3.8.	<i>Iteración 06 - Creación de proceso de canje</i>	113
3.3.9.	<i>Iteración 07 - Proceso de canje – Finalización</i>	123
3.3.10.	<i>Iteración 07 - Zona Administrativa</i>	126
3.3.11.	<i>Iteración 08 - Creación de reporte</i>	131
3.4.	FASE DE MANTENIMIENTO	134
3.5.	FASE POST-MORTEM	134
4.	CONCLUSIONES Y RECOMENDACIONES	135
4.1.	CONCLUSIONES	135
4.2.	RECOMENDACIONES	136

ÍNDICE DE TABLAS

TABLA 1-1 CAPACIDAD DE DATOS DEL CÓDIGO QR.....	16
TABLA 1-2 TAMAÑOS DE PUNTOS POR IMPRESORA	23
TABLA 1-3 TAMAÑOS DE PUNTOS POR RESOLUCIÓN DE PANTALLA.....	24
TABLA 2-1 DIFERENCIAS ENTRE METODOLOGÍAS AGILES Y TRADICIONALES.....	39
TABLA 2-2 DIALOGO ENTRE CLIENTE Y DESARROLLADOR.....	42
TABLA 2-3 COSTO POR HORA POR PUESTO.	48
TABLA 2-4 COSTO PROYECTADO.....	49
TABLA 3-1 ROLES POR PERSONA.....	50
TABLA 3-2 HISTORIA DE USUARIO: PRESENCIA EN INTERNET.	51
TABLA 3-3 HISTORIA DE USUARIO: PAQUETES DE TARJETAS.	51
TABLA 3-4 HISTORIA DE USUARIO: CATÁLOGO DE PRODUCTOS.....	51
TABLA 3-5 HISTORIA DE USUARIO: ORIENTACIÓN DE LA PÁGINA.	52
TABLA 3-6 HISTORIA DE USUARIO: PROCESO DE CANJE.	52
TABLA 3-7 HISTORIA DE USUARIO: PEDIDOS DE PRODUCTOS.....	52
TABLA 3-8 HISTORIA DE USUARIO: EMPRESAS.....	52
TABLA 3-9 HISTORIA DE USUARIO: USUARIOS.....	53
TABLA 3-10 HISTORIA DE USUARIO: ADMINISTRADOR.	53
TABLA 3-11 HISTORIA DE USUARIO: REPORTES.	53
TABLA 3-12 HISTORIA DE USUARIO: COMPRA EN LÍNEA.	53
TABLA 3-13 CARACTERÍSTICAS Web-HOSTING.....	54
TABLA 3-14 HISTORIAS DE USUARIO ORDENADAS.....	56
TABLA 3-15 PLAN DE ENTREGA.	57
TABLA 3-16 ESTIMACIÓN DE COSTOS.	57
TABLA 3-17 PRUEBAS DE ACEPTACIÓN 1 Y 2 – PRIMER PASO A PRODUCCIÓN.....	58
TABLA 3-18 PRUEBA DE ACEPTACIÓN - CATÁLOGO DE PRODUCTOS.	59
TABLA 3-19 PRUEBA DE ACEPTACIÓN 04 - GESTIÓN DE USUARIOS – 2DO. PASO A PRODUCCIÓN.	59
TABLA 3-20 PRUEBAS DE ACEPTACIÓN 05- GESTIÓN BÁSICA DE PAQUETES – 3ER. PASO A PRODUCCIÓN.	60
TABLA 3-21 PRUEBAS DE ACEPTACIÓN - GESTIÓN BÁSICA DE EMPRESAS.....	60
TABLA 3-22 PRUEBAS DE ACEPTACIÓN - CREACIÓN DE PROCESO DE CANJE.....	61
TABLA 3-23 PRUEBAS DE ACEPTACIÓN 08 – CULMINACIÓN DE CANJE.....	61
TABLA 3-24 PRUEBAS DE ACEPTACIÓN 09 - ZONA ADMINISTRATIVA.	62
TABLA 3-25 PRUEBAS DE ACEPTACIÓN - CREACIÓN DE REPORTES.....	62
TABLA 3-26 PRUEBA DE ACEPTACIÓN ADICIONAL - SELECCIÓN DE PRODUCTOS.	63
TABLA 3-27 PLANIFICACIÓN DE ITERACIONES.....	64
TABLA 3-28 ITERACIÓN INICIAL - AMBIENTE DE DESARROLLO.	65

TABLA 3-29 ITERACIÓN 01 – PRIMER PASO A PRODUCCIÓN.....	67
TABLA 3-30 PRUEBAS DE ACEPTACIÓN - ITERACIÓN 01.....	71
TABLA 3-31 PRUEBA DE ACEPTACIÓN 01	71
TABLA 3-32 PRUEBA DE ACEPTACIÓN 02	71
TABLA 3-33 ITERACIÓN 02 - GESTIÓN PRODUCTOS.....	71
TABLA 3-34 ESTANDARIZACIÓN DE LA BDD.	72
TABLA 3-35 PRUEBAS DE ACEPTACIÓN - ITERACIÓN 02. FUENTE (ANTONIO CÓRDOVA, 2016)	86
TABLA 3-36 PRUEBA DE ACEPTACIÓN 03.	86
TABLA 3-37 ITERACIÓN 03 - GESTIÓN USUARIOS.....	86
TABLA 3-38 PRUEBAS DE ACEPTACIÓN - ITERACIÓN 03. FUENTE (ANTONIO CÓRDOVA, 2016)	93
TABLA 3-39 PRUEBA DE ACEPTACIÓN 04.	94
TABLA 3-40 ITERACIÓN 04 - GESTIÓN PAQUETES.....	94
TABLA 3-41 PRUEBAS DE ACEPTACIÓN - ITERACIÓN 04.....	100
TABLA 3-42 PRUEBA DE ACEPTACIÓN 05.	100
TABLA 3-43 ITERACIÓN 05 - GESTIÓN TARJETAS.	100
TABLA 3-44 PRUEBAS DE ACEPTACIÓN - ITERACIÓN 05.....	107
TABLA 3-45 PRUEBA DE ACEPTACIÓN 05.	107
TABLA 3-46 ITERACIÓN 05 - SELECCIÓN DE PRODUCTOS.	107
TABLA 3-47 PRUEBAS DE ACEPTACIÓN - ITERACIÓN 05 – COMPLEMENTARIA.	112
TABLA 3-48 PRUEBA DE ACEPTACIÓN 11.	113
TABLA 3-49 ITERACIÓN 06 - CREACIÓN DE PROCESO CANJE.....	113
TABLA 3-50 PRUEBAS DE ACEPTACIÓN - ITERACIÓN 06.....	121
TABLA 3-51 PRUEBA DE ACEPTACIÓN 07.	121
TABLA 3-52 ITERACIÓN 07 – CULMINACIÓN PROCESO CANJE.....	123
TABLA 3-53 PRUEBAS DE ACEPTACIÓN - CULMINACIÓN DE CANJE - ITERACIÓN 07.....	126
TABLA 3-54 PRUEBA DE ACEPTACIÓN 08.	126
TABLA 3-55 ITERACIÓN 07 – CULMINACIÓN PROCESO CANJE.....	127
TABLA 3-56 PRUEBAS DE ACEPTACIÓN - ZONA ADMINISTRATIVA - ITERACIÓN 07.....	131
TABLA 3-57 PRUEBA DE ACEPTACIÓN 09.	131
TABLA 3-58 ITERACIÓN 08 – CREACIÓN DE REPORTE.	131
TABLA 3-59 PRUEBAS DE ACEPTACIÓN – CREACIÓN DE REPORTES - ITERACIÓN 08.	133
TABLA 3-60 PRUEBA DE ACEPTACIÓN 10.	133

ÍNDICE FIGURAS

GRÁFICO 1-1 LECTURA EN 2 DIMENSIONES.....	16
GRÁFICO 1-2 PATRÓN DE DETECCIÓN DE POSICIÓN.....	16
GRÁFICO 1-3 ESTRUCTURA DE UN SÍMBOLO DE CÓDIGO QR.....	18
GRÁFICO 1-4 CÓDIGO QR - MODELO 1.....	18
GRÁFICO 1-5 CÓDIGO QR – MODELO 2.....	19
GRÁFICO 1-6 COMPARACIÓN DE MICRO CÓDIGO.....	19
GRÁFICO 1-7 COMPARACIÓN DE CÓDIGO IQR	20
GRÁFICO 1-8 TIPOS RECTANGULARES DE CÓDIGOS IQR.....	20
GRÁFICO 1-9 CÓDIGO QR CON SEGURIDAD.....	21
GRÁFICO 1-10 SQRC - COMPONENTES PÚBLICOS Y PRIVADOS.....	21
GRÁFICO 1-11 EJEMPLO DE UN FRAMEQR.	22
GRÁFICO 1-12 COMPARACIÓN DE VISUALIZACIÓN	23
GRÁFICO 1-13 MÓDULO COMPUESTO POR PUNTOS DE IMPRESIÓN.....	23
GRÁFICO 1-14 MARGEN MÍNIMO DE UN CÓDIGO QR.....	25
GRÁFICO 1-15 DISTORSIÓN DEL CÓDIGO.....	28
GRÁFICO 1-16 ERRORES RELACIONADOS CON IMÁGENES O MÁRGENES.....	28
GRÁFICO 1-17 CÓDIGO QR EN WHATSAPP WEB.....	31
GRÁFICO 2-1 MODELO EN CASCADA.	33
GRÁFICO 2-2 DESARROLLO EVOLUTIVO.	34
GRÁFICO 2-3 ESTRUCTURA EN 3 CAPAS DE XP.	39
GRÁFICO 2-4 DIALOGO ENTRE CLIENTE – DESARROLLADOR.	42
GRÁFICO 2-5 FASES DEL CICLO DE VIDA.....	43
GRÁFICO 2-6 ITERACIONES.....	45
GRÁFICO 2-7 REGLAS DE XP.....	46
GRÁFICO 3-1 LOCALHOST EN WAMPSERVER.....	65
GRÁFICO 3-2 ESTRUCTURA DE LARAVEL.....	66
GRÁFICO 3-3 PAQUETE DE PRODUCTO DE JETBRAIN A ANTONIO CÓRDOVA	67
GRÁFICO 3-4 DISEÑO INICIAL.....	68
GRÁFICO 3-5 DISEÑO PÁGINA INTERNA	69
GRÁFICO 3-6 PHPFORM Y MAQUETA DE PAGINA PRINCIPAL.....	70
GRÁFICO 3-7 TABLAS: PEDIDO Y ADMINISTRADORES.....	72
GRÁFICO 3-8 MODELO DE LA BASE DE DATOS	73
GRÁFICO 3-9 MODELO LÓGICO	74
GRÁFICO 3-10 MODELO FÍSICO	75
GRÁFICO 3-11 MYSQL EN LOCALHOST CON LA BDD CREADA	76

GRÁFICO 3-12 CREACIÓN DE MODELOS EN LARAVEL	77
GRÁFICO 3-13 MODELO: PRODUCTOS.PHP	79
GRÁFICO 3-14 CONTROLADOR: PRODUCTOSCONTROLLER.PHP	80
GRÁFICO 3-15 PLANTILLA ADMIN	80
GRÁFICO 3-16 LISTA DE PRODUCTOS	82
GRÁFICO 3-17 AGREGAR PRODUCTO – DATOS	83
GRÁFICO 3-18 ÉXITO EN AGREGAR PRODUCTO.....	83
GRÁFICO 3-19 EDITAR PRODUCTO.....	84
GRÁFICO 3-20 MENSAJE PREVIO A BORRAR PRODUCTO	85
GRÁFICO 3-21 MENSAJE DE ELIMINACIÓN EXITOSO	85
GRÁFICO 3-22 MAQUETA PORTADA DE USUARIOS.....	87
GRÁFICO 3-23 PLANTILLA PARA USUARIO	88
GRÁFICO 3-24 INGRESO DE CÓDIGO	88
GRÁFICO 3-25 MODELO: USUARIOS.PHP.....	89
GRÁFICO 3-26 CONTROLADOR: AUTHCLIENTE.PHP.....	90
GRÁFICO 3-27 CONTROLADOR: DATOSCONTROLLER.PHP	90
GRÁFICO 3-28 PORTADA DE USUARIO	91
GRÁFICO 3-29 ACTUALIZACIÓN DE DATOS DE CLIENTE	92
GRÁFICO 3-30 DATOS ALMACENADOS EXITOSAMENTE	92
GRÁFICO 3-31 INGRESO MEDIANTE CÓDIGO QR	93
GRÁFICO 3-32 VISUALIZACIÓN DE PAQUETES.....	94
GRÁFICO 3-33 CREACIÓN DE PAQUETES	95
GRÁFICO 3-34 MODELO: PAQUETES.PHP	96
GRÁFICO 3-35 LISTA DE PAQUETES	97
GRÁFICO 3-36 EDICIÓN DE PAQUETE	97
GRÁFICO 3-37 ACTUALIZACIÓN EXITOSA DE PAQUETE.....	98
GRÁFICO 3-38 CREACIÓN DE NUEVO PAQUETE.....	98
GRÁFICO 3-39 CREACIÓN DE PAQUETE EXITOSO.....	99
GRÁFICO 3-40 CREACIÓN DE NUEVA EMPRESA.....	101
GRÁFICO 3-41 LISTA DE EMPRESAS.....	101
GRÁFICO 3-42 MODELO: EMPRESAS.PHP.	102
GRÁFICO 3-43 CONTROLADOR: EMPRESASCONTROLLER.PHP.	103
GRÁFICO 3-44 VISTA: LISTA DE EMPRESAS.	103
GRÁFICO 3-45 VISTA: EDITAR EMPRESA.....	104
GRÁFICO 3-46 VISTA: CREAR EMPRESA.....	104
GRÁFICO 3-47 EDICIÓN DE EMPRESA.....	105
GRÁFICO 3-48 CREACIÓN DE EMPRESA.....	105

GRÁFICO 3-49 CREACIÓN DE EMPRESA EXITOSA.....	106
GRÁFICO 3-50 PROCESO DE SELECCIÓN DE PRODUCTOS.....	108
GRÁFICO 3-51 SELECCIÓN DE PRODUCTOS	108
GRÁFICO 3-52 CONTROLADOR PAQUETESCONTROLLER.PHP.....	110
GRÁFICO 3-53 VISTA INDEX.BLADE.PHP.....	110
GRÁFICO 3-54 ZONA EMPRESARIAL - LISTA DE PAQUETES.....	111
GRÁFICO 3-55 SELECCIÓN DE PRODUCTOS	112
GRÁFICO 3-56 PROCESO DE CANJE	113
GRÁFICO 3-57 ESTRUCTURA DE CONTENIDOS – PROCESO DE CANJE	115
GRÁFICO 3-58 MAQUETA - CONFIRMAR PRODUCTOS.....	116
GRÁFICO 3-59 MAQUETA - CONFIRMAR PRODUCTOS.....	116
GRÁFICO 3-60 CONTROLADOR CARRITOCONTROLLER.PHP	118
GRÁFICO 3-61 VISTA CONFIRMAPRODUCTOS.BLADE.PHP.....	118
GRÁFICO 3-62 SELECCIÓN DE PRODUCTO.....	119
GRÁFICO 3-63 CONFIRMAR PRODUCTOS.....	120
GRÁFICO 3-64 CONFIRMAR DATOS.....	121
GRÁFICO 3-65 VISTA MAIL.BLADE.PHP.....	124
GRÁFICO 3-66 CORREO DE CONFIRMACIÓN DE CANJE.....	124
GRÁFICO 3-67 FINALIZACIÓN DE PROCESO DE CANJE.....	125
GRÁFICO 3-68 EMAIL DE COMPROBACIÓN	125
GRÁFICO 3-69 LINK DE INICIO DE SESIÓN	127
GRÁFICO 3-70 MAQUETA DE INICIO DE SESIÓN – ADMINISTRADORES	128
GRÁFICO 3-71 TABLA ADMINISTRADORES.....	128
GRÁFICO 3-72 FORMULARIO DE INICIO DE SESIÓN – ADMINISTRADORES	129
GRÁFICO 3-73 RUTAS DE ADMINISTRACIÓN.....	130
GRÁFICO 3-74 MENÚ DE OPCIONES ADMINISTRACIÓN	130
GRÁFICO 3-75 INICIO DE SESIÓN PARA ADMINISTRADOR	130
GRÁFICO 3-76 MODELO PEDIDOS.PHP.....	132
GRÁFICO 3-77 CONTROLADOR REPORTESCONTROLLER.PHP	133
GRÁFICO 1-1 INGRESO AL SISTEMA – COMPUTADOR	138
GRÁFICO 1-2 INGRESO AL SISTEMA - IPHONE5.....	138
GRÁFICO 1-3 INGRESO DE CÓDIGO	139
GRÁFICO 1-4 INICIO DE SESIÓN ERRÓNEO.....	140
GRÁFICO 1-5 PORTADA PRINCIPAL USUARIOS	140
GRÁFICO 1-6 PRODUCTOS DISPONIBLES PARA EL CANJE	141
GRÁFICO 1-7 BOTÓN: AGREGAR AL CARRITO	142
GRÁFICO 1-8 BOTÓN: QUITAR.....	142

GRÁFICO 1-9 VISUALIZACIÓN DE PRODUCTOS AGREGADOS AL CARRITO	143
GRÁFICO 1-10 CONFIRMAR PRODUCTOS.	144
GRÁFICO 1-11 CONFIRMAR DATOS.	145
GRÁFICO 1-12 FIN PROCESO DE CANJE.....	145
GRÁFICO 2-1 INICIO DE SESIÓN – EMPRESAS.	147
GRÁFICO 2-2 INICIO DE SESIÓN – IPHONE5.	147
GRÁFICO 2-3 INICIO DE SESIÓN ERRÓNEO – EMPRESAS.	148
GRÁFICO 2-4 ZONA EMPRESARIAL.	149
GRÁFICO 2-5 SELECCIÓN DE PRODUCTOS.	150
GRÁFICO 2-6 OPCIONES SOBRE PRODUCTO.	150
GRÁFICO 2-7 COMBO FILTRO DE PRODUCTOS POR TIPO.	151
GRÁFICO 2-8 ACTUALIZACIÓN DE DATOS – EMPRESA.....	152
GRÁFICO 2-9 ERROR EN ACTUALIZACIÓN DE DATOS.....	152
GRÁFICO 3-1 LINK DE ACCESO A LA ADMINISTRACIÓN.....	154
GRÁFICO 3-2 INICIO DE SESIÓN – ADMINISTRACIÓN.	154
GRÁFICO 3-3 INICIO DE SESIÓN ERRÓNEO – ADMINISTRACIÓN.	155
GRÁFICO 3-4 MENÚ PRINCIPAL.....	155
GRÁFICO 3-5 LISTA DE PRODUCTOS.	156
GRÁFICO 3-6 AGREGAR TIPO.....	156
GRÁFICO 3-7 AGREGAR NUEVO PRODUCTO – EJEMPLO.	157
GRÁFICO 3-8 OPCIONES DE PRODUCTO.	158
GRÁFICO 3-9 EDICIÓN DE PRODUCTO.....	158
GRÁFICO 3-10 POP-UP ELIMINAR PRODUCTO.	159
GRÁFICO 3-11 MENSAJE FINAL DE ELIMINACIÓN.	160
GRÁFICO 3-12 LISTA DE PAQUETES.....	160
GRÁFICO 3-13 AGREGAR NUEVO PAQUETE.....	162
GRÁFICO 3-14 PAQUETE AGREGADO CORRECTAMENTE.....	163
GRÁFICO 3-15 OPCIONES DE PAQUETE.....	163
GRÁFICO 3-16 EDITAR PAQUETE.....	164
GRÁFICO 3-17 ELIMINAR PAQUETE.....	165
GRÁFICO 3-18 LISTA DE EMPRESAS.....	166
GRÁFICO 3-19 AGREGAR NUEVA EMPRESA.....	167
GRÁFICO 3-20 EMPRESA CREADA EXITOSAMENTE.....	167
GRÁFICO 3-21 OPCIONES DE EMPRESA.....	168
GRÁFICO 3-22 EDITAR EMPRESA.....	168
GRÁFICO 3-23 ASOCIAR PAQUETE CON EMPRESA.....	169
GRÁFICO 3-24 NUEVO PAQUETE PARA EMPRESA.....	169

GRÁFICO 3-25 ASOCIACIÓN DE PAQUETE EXITOSO.....	170
GRÁFICO 3-26 OPCIONES SOBRE PAQUETES - VER PAQUETES.....	171
GRÁFICO 3-27 LISTA DE PAQUETES ASOCIADOS A EMPRESAS.....	171
GRÁFICO 3-28 CÓDIGOS GENERADOS.....	172
GRÁFICO 3-29 OPCIONES SOBRE USUARIOS.....	172
GRÁFICO 3-30 USUARIOS REGISTRADOS.....	173
GRÁFICO 4-1 ESTRUCTURA DE ARCHIVOS.....	177
GRÁFICO 4-2 ARCHIVOS EN SERVIDOR.....	177
GRÁFICO 4-3 PHPMYADMIN CARGA DE BDD.....	178

ÍNDICE DE ANEXOS

1.	ANEXO: MANUAL DE USUARIO	137
1.1.	INTRODUCCIÓN	137
1.2.	CONCEPTOS IMPORTANTES	137
1.3.	FUNCIONALIDADES	141
1.4.	PROCESO DE USO	142
1.4.1.	<i>Selección de producto/s</i>	142
1.4.2.	<i>Proceso de canje</i>	144
2.	ANEXO: MANUAL DE EMPRESA	146
2.1.	INTRODUCCIÓN	146
2.2.	CONCEPTOS IMPORTANTES	146
2.3.	PROCESO DE USO	149
2.3.1.	<i>Selección de productos</i>	149
2.3.2.	<i>Actualización de datos</i>	151
3.	ANEXO: MANUAL DE ADMINISTRACIÓN	153
3.1.	INTRODUCCIÓN	153
3.2.	CONCEPTOS IMPORTANTES	153
3.3.	PROCESO DE USO	155
3.3.1.	<i>Gestión de Productos</i>	155
3.3.2.	<i>Gestión de Paquetes</i>	160
3.3.3.	<i>Gestión de empresas</i>	165
4.	ANEXO: MANUAL TÉCNICO	174
4.1.	INTRODUCCIÓN	174
4.2.	REQUERIMIENTOS TÉCNICOS	174
4.3.	HERRAMIENTAS UTILIZADAS PARA EL DESARROLLO	175
4.4.	CONFIGURACIÓN DEL SISTEMA	175

RESUMEN

Un código QR es un tipo de código de barras, pero en una matriz de dos dimensiones que pueden mantener mucha más información que cualquier código de su clase. QR viene del inglés “Quick Response” que significa respuesta rápida, lo que hace referencia a la velocidad en la que decodifica los símbolos del código. Este código fue inventado en el año de 1994 en Japón, inicialmente para el uso en empresas que requieren el seguimiento de productos.

Esta tecnología es cada vez más accesible para uso masivo gracias a la facilidad de decodificación y al avance tecnológico de dispositivos móviles como celulares, es decir, en lugar de dar seguimiento a las partes de autos en una empresa automotriz, los usuarios con teléfonos inteligentes pueden escanear los códigos a través de sus cámaras para acceder de manera rápida y eficiente a distintas webs.

Los códigos QR actúan como un enlace entre el usuario y el mundo virtual generando nuevas oportunidades de negocio como enlaces a páginas web, menús on-line, campañas publicitarias, planes de mercadeo, etcétera.

Estas nuevas ideas de negocio, las cuales ponen en su núcleo un sistema informático, generan mayor rentabilidad ya que no se necesita un gran volumen de mano de obra sino por el contrario se requiere únicamente de un mantenimiento preventivo y correctivo, muchas de las veces ni siquiera se necesita de un espacio físico para funcionar ya que actualmente, la tendencia es hacia la nube, es decir la información ya no está en un solo lugar físico, sino en servidores a lo largo del planeta.

Por otro lado, las nuevas ideas están enfocadas al servicio más no a la distribución de software, haciendo que los costos sean más accesibles a usuarios finales o como es el caso de la presente disertación, enfocada a empresas.

En esta disertación de grado se aprovecha este potencial en crecimiento para abrir un nuevo modelo de negocio en donde el usuario hace uso de la tecnología para acceder a la información de manera rápida y en la palma de su mano, permitiendo que el nicho de mercado se expanda y no esté restringida a usuarios con un computador con conexión a internet, sino a personas que están a pie y que cuentan con un dispositivo inteligente.

CAPÍTULO 1

1. METODOS PARA DECODIFICAR CÓDIGOS QR

En el presente capítulo se realizará una revisión sistemática de los orígenes y distintos métodos de generación de códigos que ha surgido a lo largo de las últimas décadas. Asimismo, los métodos para decodificarlos y las diferentes aplicaciones que pueden tener los códigos QR.

1.1. Origen

En la década de los 60's, el mundo se encontraba en pleno desarrollo económico, el comercio empezó a crecer de forma exponencial y las empresas ofrecían sus productos a la población en general, desde autos hasta ropa.

Si bien, los supermercados vendían dichos productos a través de cajas registradoras, las transacciones se realizaban de manera manual, los cajeros escribían los códigos de los productos cada vez que un comprador se acercaba a la caja, convirtiéndose en un grave problema ya que al final del día los cajeros sufrían por entumecimientos en sus muñecas incluso en el túnel carpiano, debido al constante trabajo de la mano con el teclado para el ingreso de datos, adicionalmente, existían muchos problemas al momento de cuadrar caja dado que en caso de que hubiera un error en la escritura de los números eso implicaba una larga y extensa revisión que tomaba más tiempo de lo normal.

Para solucionar este problema, técnicos electrónicos junto con analistas de sistemas desarrollaron “el código de barras”, un sensor electrónico que lee estos códigos y de paso, aceleran el proceso de compra. Este método, ofrece la solución tanto a cajeros como a los encargados de ventas ya que se suprime el uso excesivo del teclado, así como se tiene un control en las máquinas registradoras sobre las ventas realizadas.

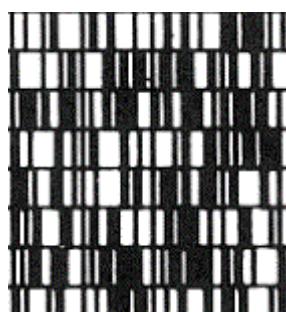
Esta invención se hizo muy popular a nivel mundial, pues no solo se utilizaba para el comercio sino también para inventarios en grandes fábricas, manejo y manipulación de partes o piezas, tarjetas de identidad, entre otras.

Más tarde, otro problema que se presentó, fue la poca capacidad de información que podía almacenar el código de barras, ya que solo contenía 20 o menos caracteres alfanuméricos de información.

Para desarrollar una nueva alternativa, *DENSO WAVE*, a través de su división *DENSO WAVE INCORPORATED*, se propuso desarrollar una nueva alternativa mediante la siguiente frase: “We'd like the capability to code Kanji and Kana¹ characters as well as alphanumeric ones.” (*DENSO WAVE INCORPORATED*, 2014)

Esta división inició los trabajos de investigación y desarrollo de un nuevo código con el fin de guardar la mayor cantidad de información y lograr que éste sea de rápida y fácil lectura.

Masahiro Hara quien era técnico de la empresa, fue la persona encargada de desarrollar este nuevo código, quien se propuso el objetivo de usar códigos de dos dimensiones, es decir, que la información se codifica en dos direcciones, de tal forma que la información se pueda leer de izquierda a derecha y de arriba abajo, como se representa en el gráfico 1-1. El problema de usar dos dimensiones fue cómo identificar el inicio de lectura.



¹ “El idioma japonés escrito es una mezcla (llamada *kanji - kana majiri*) de tres tipos de símbolos, cada uno con su propia función... *Kanj*: Son caracteres pictográficos - ideográficos, adoptados de la Lengua China, se utilizan para las palabras conceptuales... *Kana*: Son símbolos fonéticos desarrollados en Japón. Cada símbolo representa el sonido de una sílaba.” Traducción propia desde (Hadamitzky & Spahn, 1997)

Gráfico 1-1 Lectura en 2 dimensiones.

Fuente: <http://www.wlp.com/bar-code-49.html>

Este problema se solucionó agregando información gráfica adicional sobre el código bidimensional, como se muestra en el gráfico 1-2. Esta ayuda gráfica fue representada por tres cuadrados ubicados en cada esquina del código y un cuadrado pequeño en la esquina sobrante para ayudar en la alineación, formando un patrón de detección de posición, el cual permite una lectura rápida ya que sin importar la posición del código de igual forma el patrón indicaría el inicio y el fin del código.

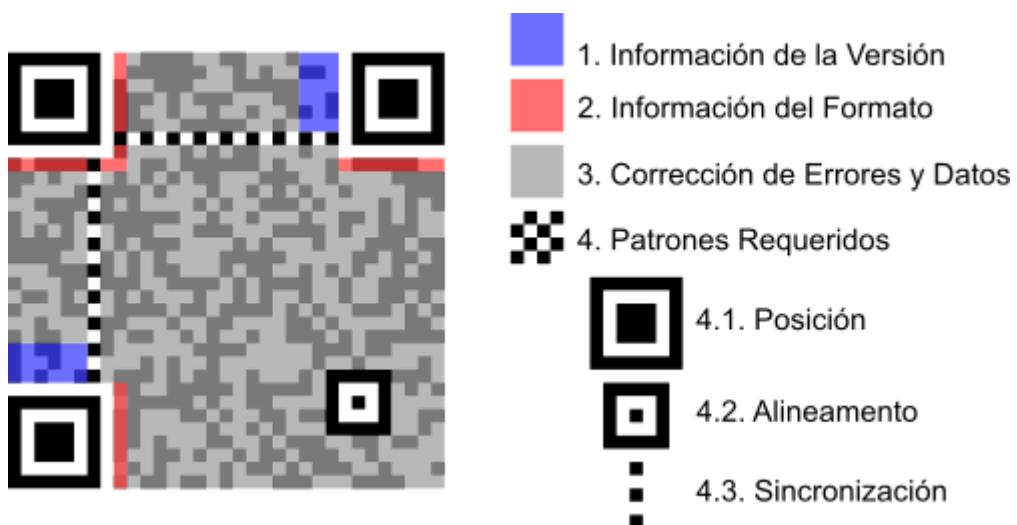


Gráfico 1-2 Patrón de detección de posición.

Fuente (ISO/IEC 18004:2015, 2015)

Después de un año y medio de experimentación, se decidió colocar cuadrados en lugar de otra figura para evitar confusiones o lectura errónea de figuras o logotipos con el patrón de detección de posición, encontrado así un código capaz de codificar aproximadamente unos siete mil números y la capacidad adicional de codificar caracteres Kanji, como se muestra en la tabla 1-1.

Tabla de almacenamiento en códigos QR

Solo numérico	1, 2, 3, 4	Máx. 7.089 caracteres
Alfanumérico	A, b, c, 1, 2, 3	Máx. 4.296 caracteres
Binario	0, 1, 0, 1, 0	Máx. 2.953 bytes
Kanji / Kana	私の人生を導	Máx. 1.817 caracteres

Tabla 1-1 Capacidad de datos del código QR.

Fuente: (DENSO WAVE INCORPORATED, 2014)

Los cuadrados utilizados debían tener una relación 1:1:3: 1:1 y usando áreas negras y blancas en los patrones de detección de posición, de tal forma que no importa la

orientación ni el ángulo de exploración de su código (podría ser cualquier ángulo de 360º).

Finalmente, este código además de poseer una gran cantidad de almacenamiento, también logró mejorar la velocidad de lectura en 10 veces más que cualquier código existente hasta la fecha. Se lo nombró QR por sus siglas en inglés *Quick Response* (Respuesta Rápida)².

Este nuevo invento fue adoptado inicialmente por la industria automotriz para ayudar a la gestión de calidad bajo la metodología Kanban³. Desde entonces muchas industrias como farmacéuticas, empresas de alimentos y bebidas empezaron a usar los códigos para el control de sus bienes, gracias a que DENSO WAVE decidió liberar el código para que cualquier persona o empresa pueda usarlo.

Aunque DENSO WAVE decidió conservar los derechos de patente del código QR, decidió no ejercerlos, esta política dio inicio a una gran corriente de uso por parte de millones de personas alrededor del mundo.

1.2. Tipos de códigos QR

Para conocer los tipos de códigos QR se debe analizar la estructura simbólica del mismo, visualizando ciertas partes principales del código, las cuales se representan en el gráfico 1-3.

² Tomado de la página web oficial (DENSO WAVE INCORPORATED, 2014) de la sección History

³ KANBAN es una herramienta simple para administración de tareas de manera ordenada y efectivamente. KANBAN se deriva de palabras japonesas que significan Sign Card (tarjeta de registro), el cual aumenta dramáticamente mejoras en términos de rendimiento y tiempo de espera.

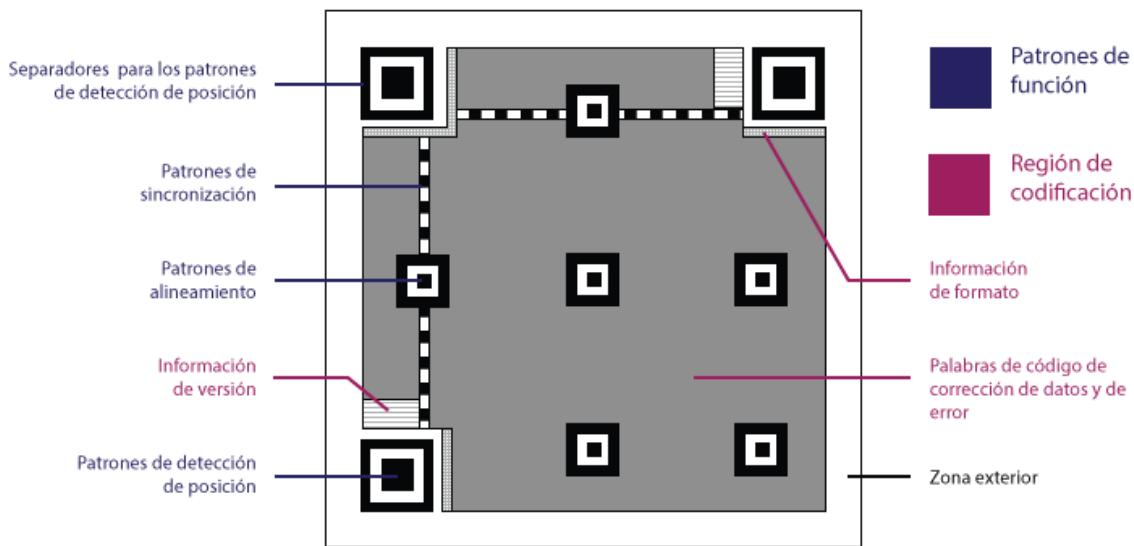


Gráfico 1-3 Estructura de un símbolo de código QR.

Fuente: (ISO/IEC 18004:2015, 2015)

Cada símbolo QR está construido por módulos cuadrados nominalmente establecidos en un arreglo doble que contiene una región de codificación y los patrones de función.

La región de codificación está compuesta por la información de formato, versión y las palabras de código de corrección de datos y de error, mientras que, los patrones de función están compuestos por los patrones de detección de posición y sus respectivos separadores, los patrones de sincronización y alineamiento. Por último, el símbolo o código está rodeado por los cuatro costados de una zona exterior.

Se entiende que, los datos que almacenará el código se encuentran en la región de codificación mientras que los datos de ayuda o cabecera en los patrones de función.

Actualmente existen 5 tipos de códigos QR.

1.2.1. Modelo 1- 2

El modelo 1 es el código original el cual tiene la capacidad de codificar 1167 números y su máxima versión de 73 x 73 módulos. Un ejemplo de éste modelo se muestra en el gráfico 1-4.



Gráfico 1-4 Código QR - Modelo 1

Fuente: (DENSO WAVE INCORPORATED, 2014)

El modelo 2 fue creado como una mejora al modelo 1, de tal forma que pueda ser leído en cualquier posición con la adición de los patrones de alineamiento, de igual forma cuando se imprimen los códigos QR en superficies curvas o impresiones distorsionadas debido al ángulo de lectura.

Este modelo tiene la capacidad de codificar 7089 números y su máxima versión de 177 x 177 módulos. El cual se presenta en el gráfico 1-5.



Gráfico 1-5 Código QR – Modelo 2
Fuente: (DENSO WAVE INCORPORATED, 2014)

1.2.2. Micro código QR

Este código fue creado para un menor almacenamiento de caracteres y para espacios muy reducidos de información.

Comparado con un código QR regular, el micro código tiene solo un patrón de detección de posición y dos módulos de información, tal y como se presenta en el gráfico 1-6.

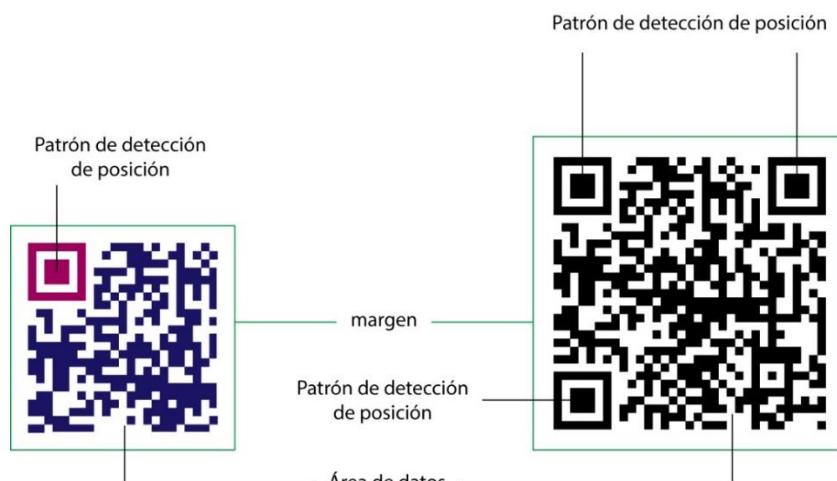


Gráfico 1-6 Comparación de micro código.
Fuente: (DENSO WAVE INCORPORATED, 2014)

La cantidad de datos que puede almacenar un micro código es de 35 números, lo cual le hace más eficiente al momento de su lectura.

1.2.3. Código iQR

Es un código en 2D de tipo matriz de lectura fácil debido a su posición y tamaño. Este código permite una amplia gama de tamaños de códigos desde un pequeño como un código QR tradicional o un micro código, así como un grande que puede almacenar muchos más datos que sus antecesores.⁴

Para hacer un comparativo entre los códigos QR tradicionales, el tamaño mínimo consiste en once módulos, mientras que el iQR tiene nueve, lo que conlleva a una reducción del 60% en comparación entre ambos, tal y como se presenta en el gráfico 1-7.

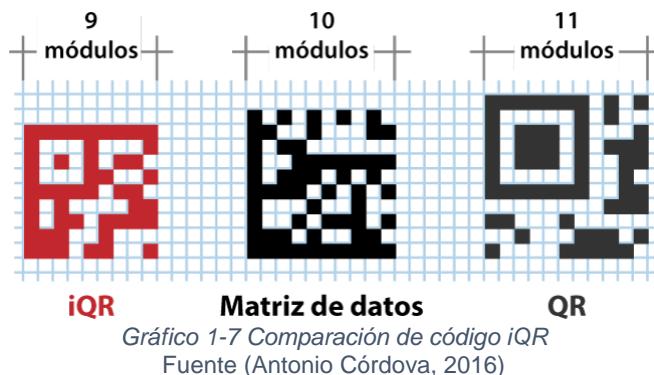


Gráfico 1-7 Comparación de código iQR
Fuente (Antonio Córdova, 2016)

Este código puede ser impreso como un código rectangular (convertido en un código), en un código invertido de blanco y negro o en un código de patrones de puntos (marcación directa de patrones), permitiendo, de esta forma, un incremento de hasta un 80% de codificación de caracteres, o por otro lado se puede reducir hasta un 30% de información. Comparativo exemplificado en el gráfico 1-8.

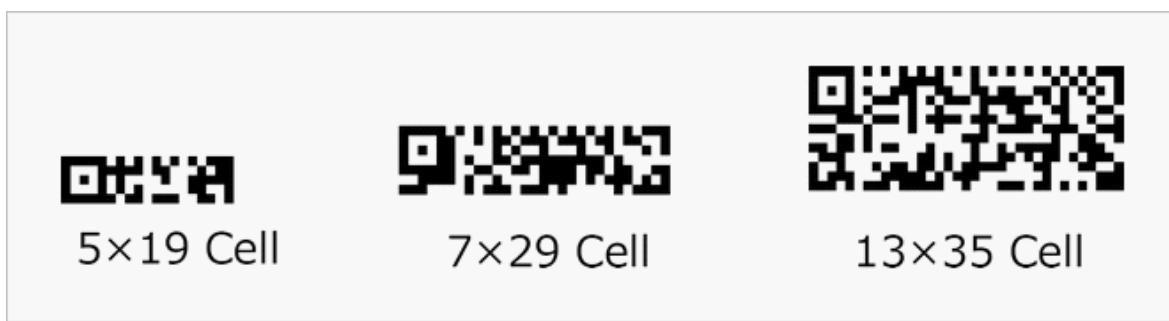


Gráfico 1-8 Tipos rectangulares de códigos iQR.
Fuente: 2d-code.co.uk/iqr-qr-code

Estos códigos pueden ser impresos en superficies cilíndricos, siempre y cuando se mantenga la claridad para la visibilidad y lectura.

⁴ Tomado de la página web oficial (DENSO WAVE INCORPORATED, 2014) de la sección iQR CODE

1.2.4. Código SQRC

SQRC es un tipo de código QR el cual está equipado con una función de lectura restringida, es decir que puede ser usado para almacenar información privada y que deba ser administrada puertas adentro.⁵, éste código es simbolizado en el gráfico 1-9.



Gráfico 1-9 Código QR con seguridad.

Fuente: Antonio

Para poder leer este tipo de código se requiere lectores especiales, como se puede ver en el gráfico 1-10, aunque el propietario de la patente no garantiza la seguridad de la información. El código tiene una parte pública, la cual puede ser leída por cualquier dispositivo; y otra privada, la cual tiene acceso únicamente a lectores con decodificadores creados únicamente para este tipo de códigos.

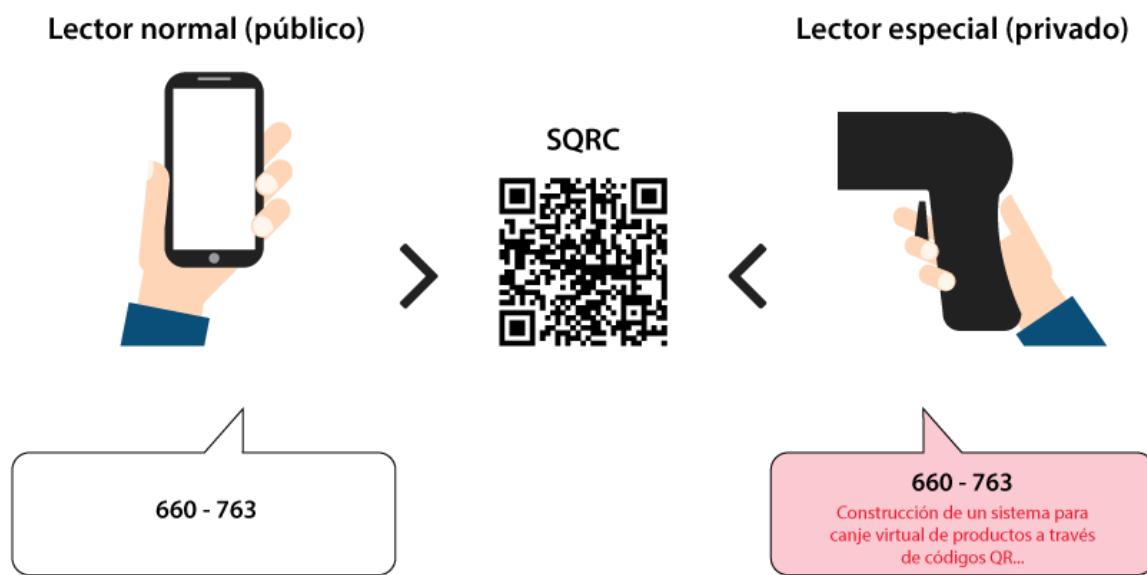


Gráfico 1-10 SQRC - Componentes públicos y privados
Fuente (Antonio Córdova, 2016)

⁵ Tomado de la página web oficial (DENSO WAVE INCORPORATED, 2014) de la sección SQRC CODE

1.2.5. Frame QR



Gráfico 1-11 Ejemplo de un FrameQR.

Fuente: (DENSO WAVE INCORPORATED, 2014)

Este tipo de código tiene un área especial en el centro llamado área de lienzo y que puede ser utilizado de cualquier forma, es decir, se puede incluir figuras especiales, gráficos, fotos, letras, etc.⁶

El área del lienzo no interfiere en la lectura del código ya que parte de una plantilla pre elaborada en la cual se especifica dimensión, forma y ángulo de la figura incrustada.

Se utiliza este tipo de códigos en tarjetas de presentación, verificación de autenticidad, promociones de productos en descuento.

1.3. Generación de códigos QR

Para generar un código QR hay que tomar en cuenta dos características importantes: configurar el tamaño del módulo y determinar el área del código⁷, ya que de ellas depende una correcta lectura.

1.3.1. Configurar el tamaño del módulo

Se debe establecer el tamaño del módulo (la parte más pequeña de color oscuro o negro), teniendo en cuenta las capacidades de la impresora y el dispositivo de lectura. Es decir, si la impresora no tiene una buena capacidad o calidad de impresión se recomienda que el tamaño del módulo sea mayor. Cuanto mayor sea el modulo, más estable y más fácil será la lectura.

Por otra parte, al aumentar el tamaño del módulo, el símbolo del código QR también aumentará su tamaño, por lo tanto, el área de impresión crecerá, es, por lo tanto,

⁶ Tomado de la página web oficial (DENSO WAVE INCORPORATED, 2014) de la sección Frame QR CODE

⁷ Tomado de la página web oficial (DENSO WAVE INCORPORATED, 2014) de la sección Generating a QR code

necesario establecer el tamaño del módulo después de establecer el área de impresión y temas relevantes a esta área.



Gráfico 1-12 Comparación de visualización
Fuente (Antonio Córdova, 2016)

La comparación que se realiza en el gráfico 1-12 entre tres tamaños distintos de un mismo código QR de modelo 1, demuestra que se puede perder el detalle e información que pueda contener el código. Esto se debe a que las impresoras a tinta imprimen puntos y mientras más pequeño es el elemento, se pierde detalle del mismo, a diferencia de una impresora láser que tiene mucha más calidad. Como se muestra en el grafico 1-13.

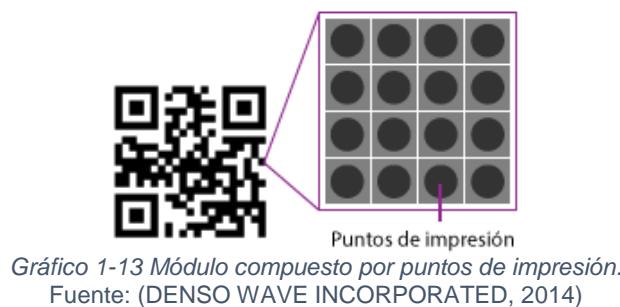


Gráfico 1-13 Módulo compuesto por puntos de impresión.
Fuente: (DENSO WAVE INCORPORATED, 2014)

A continuación, la tabla 1-2 muestra del tamaño de la impresora y el tamaño del módulo, tomada de la página oficial de la empresa DENSO WAVE.

Tamaño de la impresora y el módulo

Impresora	densidad de cabezas	4-punto de configuración	5-punto de configuración	6-punto de configuración
Láser	600 dpi (24 puntos/mm)	0.17mm	0.21mm	0.25mm
	360 dpi (14 puntos/mm)	0.28mm	0.35mm	0.42mm
Térmico	300dpi (12 puntos/mm)	0.33mm	0.42mm	0.5mm
	200 dpi (8 puntos/mm)	0.5mm	0.63 mm	0.75mm

Tabla 1-2 Tamaños de puntos por impresora
Fuente: <http://www.qrcode.com/en/howto/cell.html>

Lo mismo sucede en el caso de lecturas sobre pantallas, que hoy en día existen varias resoluciones desde monitores con tubo de vacío, pantallas planas, pantallas de alta resolución y de alta definición (HDMI), aquellas que tienen Retina (MAC de Apple), o AMOLED. Mientras más detalle o mayor nitidez tenga una pantalla mejor será la lectura del código. A continuación, se presenta la resolución mínima en milímetros en la tabla 1-3.

Tipo de escáner o lector	Resolución
Alta resolución	BHT-800Q Series
	GT20QD-SM
Resolución normal	BHT-710QWB-CE, BHT-600Q Series, Serie QK20, AT10Q-SM / SB
Tipo de cámara	QD25
	Variable en función de la lente

Tabla 1-3 Tamaños de puntos por resolución de pantalla

Fuente: <http://www.qrcode.com/en/howto/cell.html>

1.3.2. Determinar el área de código

Una vez determinado el tamaño de módulo, se debe establecer que la zona del código QR tenga un margen de al menos 4 módulos, es decir un espacio en blanco mínimo de 4 módulos en cada lado del código. El margen en cuestión es una zona clara, limpia sin color (blanco), es decir que no se debe imprimir nada, tal como se indica en el gráfico 1-14

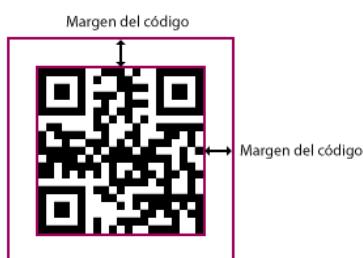


Gráfico 1-14 Margen mínimo de un código QR
Fuente (Antonio Córdova, 2016)

1.3.3. Procedimiento de codificación

En esta sección se especifica de manera general los pasos requeridos para convertir datos de entrada a un símbolo de código QR.

1.3.3.1. Análisis de datos

Para analizar el flujo de datos de entrada se debe identificar la variedad de caracteres que se desea codificar, en otras palabras, identificar si la cadena de datos contiene números, letras, binario o kanji (ver origen en la primera sección). Los códigos QR incluyen varios modos de codificación que dependen de la complejidad de caracteres para lograr una eficiente codificación. Si no se especifica qué tipo de caracteres se almacena, se selecciona la versión más pequeña que se acomode con los datos, las versiones se encuentran especificadas en la (ISO/IEC 18004:2015, 2015).

1.3.3.2. Codificación de datos

Para convertir los caracteres de los datos en un flujo de bits de acuerdo con las reglas establecidas en la (ISO/IEC 18004:2015, 2015) definidos en los subcapítulos del 8.4.1 al 8.4.5. Luego se debe especificar los modos dependiendo de cada segmento nuevo y por último introducir un valor de finalización de la secuencia de datos. Se debe dividir el flujo de bits resultante en palabras código de 8 bits. Finalmente, se debe agregar caracteres de relleno si es necesario para completar el número de palabras código.

1.3.3.3. Codificación de corrección de errores

Se debe dividir la secuencia de palabras código en el número requerido de bloques para permitir que los algoritmos de corrección de error trabajen, esto genera las palabras código de corrección de errores para cada bloque que se añaden al final de la secuencia de palabras código.

1.3.3.4. Estructura del mensaje final

Se debe intercalar los datos y palabras código de corrección de errores de cada bloque y finalmente añadir los bits para llenar si es necesario.

1.3.3.5. Ubicación de los módulos en la matriz

Se debe ubicar los módulos de las palabras código en la matriz junto con los elementos principales del código QR (Patrones de alineación, separación, etc.)

1.3.3.6. Cubrir los símbolos

Se debe aplicar un patrón de enmascaramiento para codificar la región de los símbolos. Posteriormente se evalúa los resultados y se selecciona el patrón que optimiza el equilibrio entre luminosidad y oscuridad para minimizar la aparición de patrones indeseables.

El último paso genera la información del formato y versión para completar el símbolo.

Varios lenguajes de 3ra. Generación ofrecen la codificación de símbolos QR que aplican los pasos anteriores, uno de los que ofrecen esta funcionalidad es PHP, se analizará en los capítulos posteriores.

1.4. Decodificación códigos QR

Para leer el símbolo de un código QR se debe realizar varios pasos de forma que se pueda obtener los datos originales.

1. Ubicar y obtener una imagen del símbolo reconociendo los módulos blancos y negros como un arreglo de 0 y 1 (bits)
2. Leer la información de formato
3. Determinar la versión del símbolo QR.
4. Retirar el patrón de enmascarado
5. Separar los datos y las palabras código de corrección de errores
6. Si se detectan errores, se los corrige, caso contrario se decodifica las palabras código; y
7. Se imprime la respuesta

Existen varios dispositivos (hardware) que se encargan de la lectura de símbolos QR que dependen del sector al cual están orientados, ya que la lectura debe ser clara y rápida, por ejemplo, en una fábrica automotriz donde existe un manejo de millones de piezas que deben ser identificadas en un fragmento de segundo para no parar el proceso de fabricación.

La empresa japonesa creadora del código QR, DENSO WAVE, ofrece una amplia gama de dispositivos de lectura que son orientados al sector empresarial. Por otra parte, los usuarios de Internet están utilizando con mayor frecuencia los códigos QR debido a la evolución del internet a la Web 3.0

“La web 3.0 supone una revolución en la web basada en datos y dispositivos vinculados. Se presentan nuevas formas de integración y combinación de datos para lograr la información y los conocimientos; y nuevas formas de conectarse a la web a través de una variedad de máquinas capaces de la búsqueda, la combinación y el análisis de datos. Las tecnologías emergentes que permiten que la Web 3.0 se convierta en una realidad son: la web semántica, video en la web, web móvil y web ubicua.”⁸.

Se considera como una alternativa válida a la cámara fotográfica de un teléfono inteligente a pesar de ser un dispositivo creado para fines distintos al lector de códigos QR. Esto puede generar problemas al momento de la lectura de códigos, algunos identificados en el siguiente tema.

1.4.1. Problemas encontrados

Los códigos QR pueden leerse de forma legible cuando se siguen las normas adecuadamente, definidas en la (ISO/IEC 18004:2015, 2015) el no hacerlo puede causar problemas determinados con dispositivos de lectura como teléfonos móviles. A continuación, se presenta varios ejemplos de problemas comunes.

1.4.1.1. Distorsión

Cuando un código se modifica en sus dimensiones originales se pierde la claridad y pueden aparecer nuevos módulos inexistentes, lo que produce resultados inesperados en su lectura. Se muestran dos ejemplos con alta y baja resolución en el gráfico 1-15.

⁸ Tomado de (Ramos Martín & Ramos Martín, 2014)

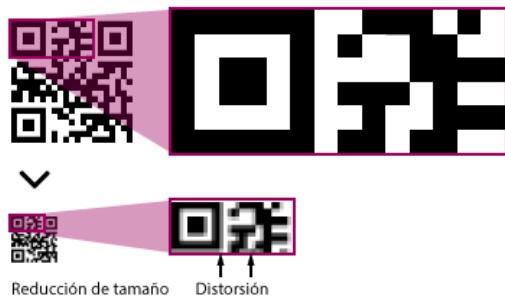


Gráfico 1-15 Distorsión del código.
Fuente (Antonio Córdova, 2016)

1.4.1.2. Código sin margen mínimo

Si colocamos letras, dibujos alrededor del código sin respetar el margen mínimo, será imposible la lectura.

Otro tipo de error es cuando se incluye imágenes superpuestas en el código como marcas de agua, tal como se muestra en el gráfico 1-16



Gráfico 1-16 Errores relacionados con imágenes o márgenes
Fuente (Antonio Córdova, 2016)

1.5. Aplicaciones

Los códigos QR tienen prácticamente un lugar en cualquier negocio, en cualquier industria y en cualquier parte de la vida de usuarios promedio que ansíen conectarse al mundo digital.

Las aplicaciones básicas del código QR han sido fundamentalmente para la manufactura y el control de inventario de productos, pero se puede encontrar nuevas aplicaciones.

1.5.1. Para la educación

Escuelas, colegios, universidades, en fin, en las aulas de clase son un buen lugar para los códigos QR, debido a que actualmente cerca del 81% de estudiantes usan celulares incluso desde tempranas edades.

En un futuro los estudiantes podrán revisar sus tareas que el profesor les envió a través de un código QR, de igual forma podrán acceder a contenido único a través de un código y al alcance de sus manos.

Por otro lado, los padres de familia podrán tener un acceso a los reportes de sus hijos mediante un código protegido por una contraseña.

Los profesores podrán ofrecer un espacio en su agenda para atención a padres o estudiantes simplemente escaneando el código.

En la universidad los maestros podrán compartir su syllabus y la información general sobre la materia impartida, entregando a cada estudiante un código QR para ser escaneado mediante teléfonos inteligentes, de esta forma podrán tener siempre en sus manos la información necesaria para su curso.

Es molesto tanto para personal que trabaja en la institución como para estudiantes solicitar el usuario y contraseña para conectarse con alguna red wi-fi disponible, pues un simple código QR con esta información se necesitará para conectarse.

1.5.2. Para ayuda social

Cuando se quiere contar un proyecto social mediante volantes publicitarios, pero no se cuenta con un espacio suficiente por lo que se podría utilizar un código para poder redirigir al usuario a una web con mayor detalle.

Si el objetivo es recaudar fondos para alguna obra social, los códigos QR pueden servir de vínculo con aquellas personas que deseen colaborar a esta causa.

Firmar o ser parte de alguna iniciativa a través de un acceso rápido a la información de objetivos, enfoques, etc.

1.5.3. Para el trabajo diario

Si se incluye un código QR en la hoja de vida, el empleador puede acceder de primera mano al portafolio, experiencia o a información extra que no se puede incluir en una hoja impresa, por ejemplo, a la web personal.

Si el trabajo es de soporte a usuarios, una buena ayuda sería ofrecer a los usuarios un código QR con acceso a las preguntas frecuentes de tal forma liberar la carga de soporte y enfocarse en resolver problemas críticos.

Mantener un código QR en dispositivos electrónicos para que si en algún momento fallan se pueda buscar rápidamente soluciones para el dispositivo averiado o información de contacto para soporte técnico.

1.5.4. Para el hogar

Parece ser que las aplicaciones de los códigos QR se centran en la industria o actividades fuera del hogar, pero se puede encontrar alguna utilidad en todas las actividades que se realiza diariamente.

Colocar un código QR en todas las cajas para que contengan una lista de ítems que hay en su interior, podría facilitar el empaque y desempaque de estas cajas cuando existe un cambio de domicilio.

Colocar un código QR en el collar de la mascota de tal forma que se pueda almacenar el mapa de residencia, nombre, número telefónico de la familia, posibles enfermedades, alergias, etc.

1.5.5. Para el área de la salud

En un hospital se puede utilizar de diferentes maneras un código QR, además del control de catálogo de medicinas, inventario de bienes inmuebles.

La administración puede crear la historia clínica de un paciente y generar un código QR para que cualquier doctor que esté atendiendo a dicho paciente pueda tener acceso a la historia clínica.

Presentar a los pacientes y público en general el perfil profesional de cualquier médico a través de un código QR colgado en la puerta del despacho.

1.6. Sistemas que utilizan la generación de códigos QR

Actualmente existen varios sistemas que utilizan la generación de códigos QR, especialmente por empresas tecnológicas orientadas a la Web. Tal es el caso de Facebook, que es un gigante del internet que recientemente adquirió WhatsApp⁹, el cual es una aplicación de mensajería instantánea que funciona en dispositivos móviles. Esta empresa ha dado una nueva funcionalidad a los usuarios de esta plataforma, para que puedan continuar usando la mensajería a través de un computador personal que cuente con un navegador web y acceso a internet. Para lograrlo el usuario debe escanear el código QR a través de su celular para permitir el uso.

⁹ "Whatsapp Messenger es una aplicación de mensajería multiplataforma que te permite enviar y recibir mensajes sin pagar por SMS" tomado de (Whatsapp, home, 2016).

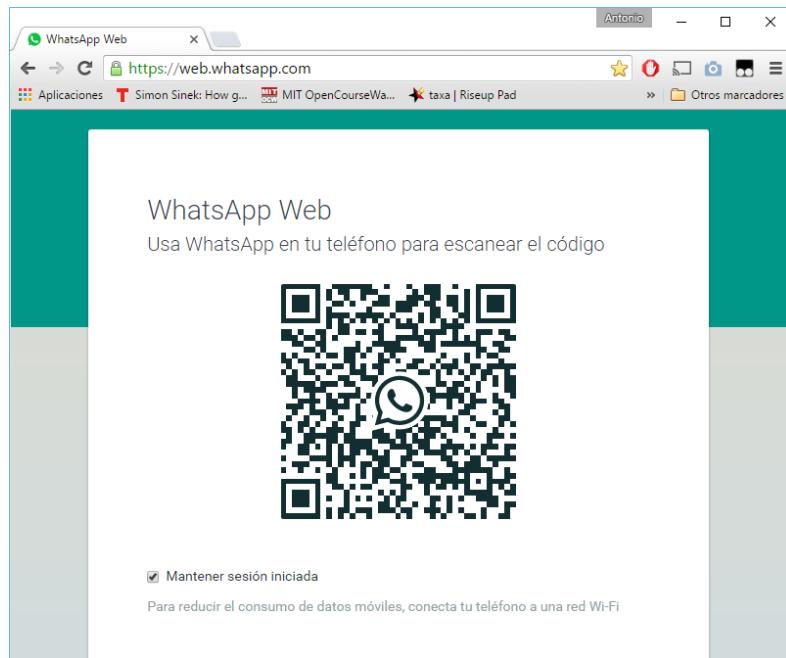


Gráfico 1-17 Código QR en WhatsApp web.
Fuente: web.whatsapp.com

En el gráfico 1-17 se muestra el código QR generado por WhatsApp para acceder a su servicio de mensajería, actualmente, el servicio supera los 600 millones de usuarios, logrando dinamizar su negocio.

Tesco¹⁰, en Corea del Sur instaló una tienda virtual de códigos QR en el 2009 para ofrecer a los clientes una comodidad absoluta en compras, la tienda virtual permite a los consumidores ingresar a la tienda, pero se realiza a través del código. Esta empresa transformó el tiempo de tránsito en el tiempo comercial valioso, logrando un incremento del 76% de clientes, ingresos sobre los 130%, más de 65000 usuarios registrados.

Toys R Us¹¹ inicio una campaña en la cual solicita a sus clientes escoger a su juguete favorito, para ello debieron ingresar al sitio web de compra en línea de la empresa mediante el uso de un código QR colocado en cada producto. Esto genero más de 30000 registros en un solo mes, logro la integración de Google Wallet en la empresa para la venta en línea.

¹⁰ "As one of the world's largest retailers with over 480,000 colleagues, we serve millions of customers a week in our stores and online." Tomado de (Tesco PLC, 2016)

¹¹ "Toys“R”Us, Inc. is the world's leading dedicated toy and baby products retailer, offering a differentiated shopping experience through its family of brands" tomado de (Toys R U, 2016)

CAPÍTULO 2

2. ANÁLISIS DE LA METODOLOGÍA ÁGIL

En el presente capítulo se analizará la metodología ágil, que, a través de su ciclo de vida y el modelo evolutivo constituyen la introducción de este capítulo. Finalmente, se dará a conocer la justificación del uso de esta metodología.

2.1. Introducción

Es importante analizar varios conceptos previos que ayudan a comprender de mejor manera la metodología ágil para el desarrollo de un sistema.

2.1.1. Ciclo de vida

“El proceso de desarrollo de un sistema es conocido también como el ciclo de vida del desarrollo de un sistema de información (...)”¹² En otras palabras un sistema tiene que pasar a través de varias fases de construcción de manera que pueda resolver el problema planteado. El desarrollo de estas fases a menudo puede tomar tiempo y recursos por lo que se requiere de planificación antes de iniciar el trabajo.

Varios expertos han propuesto una serie de métodos que ayudan a culminar satisfactoriamente el desarrollo del sistema.

El método básico es el modelo en cascada el cual describe el ciclo de vida que comprende del Estudio de Factibilidad, Análisis de Requerimientos, Diseño, Codificación y prueba de módulos, Integración y prueba del sistema, Instalación y mantenimiento. Cada una de estas fases tiene un propósito y ofrece un producto en especial.

¹² (El ciclo de vida de los sistemas informáticos, 2006)

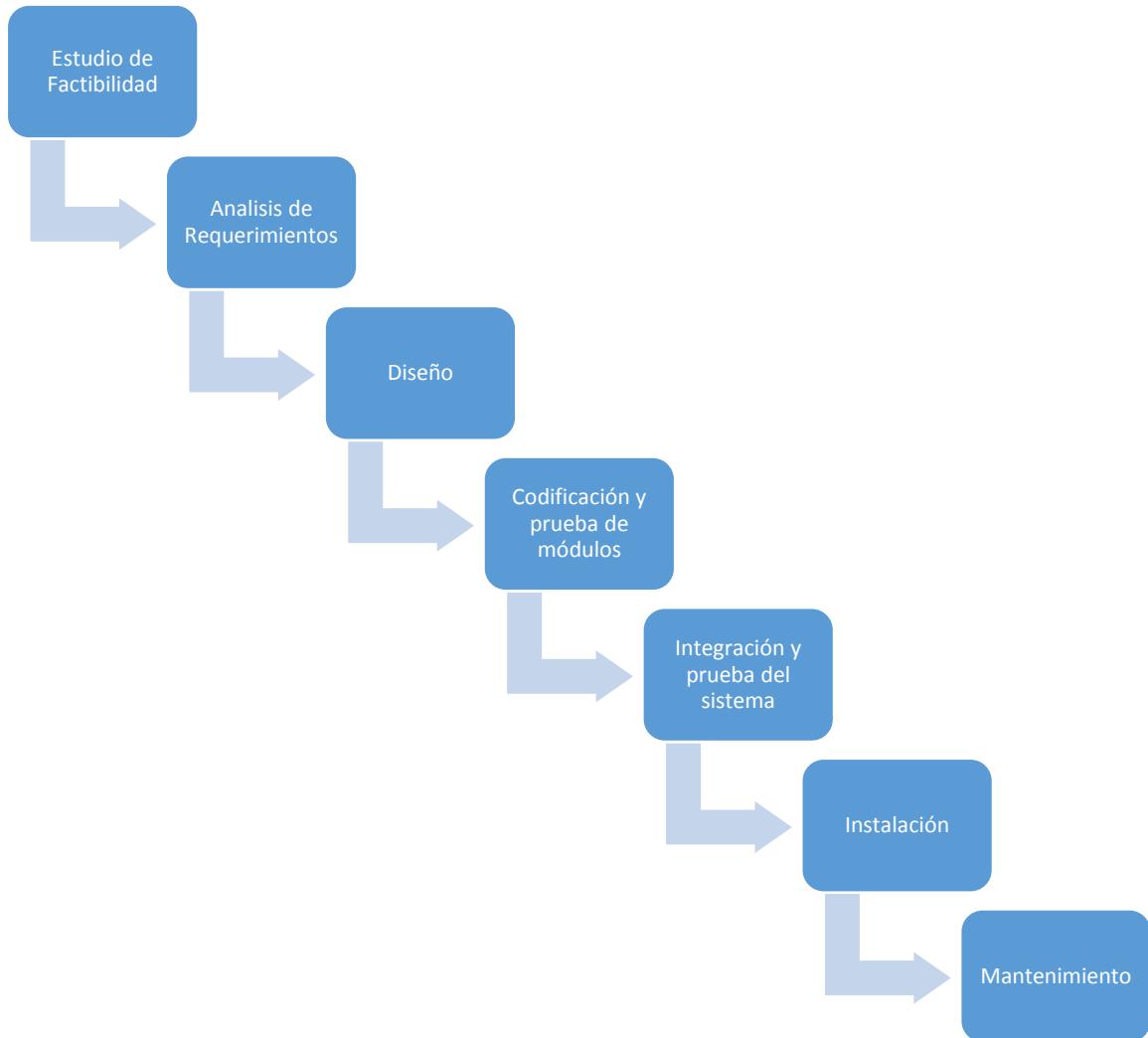


Gráfico 2-1 Modelo en cascada.
Fuente (De la Cuz, 2014)

A partir de este modelo, se crearon dos modelos, el modelo evolutivo y el modelo en espiral. Debido al enfoque de la presente disertación de grado, se explicará el modelo evolutivo.

2.1.2. El Modelo Evolutivo

“El desarrollo evolutivo se basa en la idea de desarrollar una implementación inicial, exponiéndola a los comentarios del usuario y refinándola a través de las diferentes versiones hasta que se desarrolla un sistema adecuado. Las actividades de especificación, desarrollo y validación se entrelazan en vez de separarse, con una rápida retroalimentación entre éstas.”¹³

¹³ Extracto de (Sommerville, 2006)



Gráfico 2-2 Desarrollo Evolutivo.
Fuente: (Sommerville, 2006)

En definitiva, el modelo evolutivo es más flexible que permite incrementos posteriores al inicio del desarrollo. La idea principal de este modelo como podemos apreciar en la figura anterior, es desarrollar el producto de software a través de una serie de versiones hasta lograr la versión final que cumpla su función específica.

La figura nos indica el proceso a seguir:

- Se entrega una versión inicial al cliente
- El cliente analiza la funcionalidad
- Se ajusta la versión basándose en las observaciones dadas por el cliente
- Se entrega la versión final

En la práctica, el ciclo de vida evolutivo, es más efectivo que el modelo en cascada ya que se enfoca en satisfacer las necesidades del cliente con el principio de incrementos futuros.

Finalmente, el modelo evolutivo puede ponerse en práctica a través de la metodología llamada Extreme Programming (XP), debido a que define las actividades y productos a entregar en cada fase.

El siguiente apartado describe la metodología que se aplica en el desarrollo de la presente disertación, y sus principios para las buenas prácticas en el desarrollo de software.

2.2. Metodología de desarrollo: Extreme Programming (XP)

“Una metodología de desarrollo de software es un conjunto de métodos, reglas y postulados que permiten que la forma de producción de software sea más predecible.”¹⁴ Implementar una metodología de desarrollo es cumplir con varios procesos para obtener un producto de software.

2.2.1. Metodología Ágil

La metodología ágil nació en 2001, gracias a varios esfuerzos por parte de un grupo de promotores de nuevas metodologías con el objetivo de proponer una base con principios y valores que permitan a los desarrolladores construir software brindando apertura para cambios que puedan surgir durante el proyecto, evitando la documentación excesiva. En otras palabras, ofrecer una alternativa a las metodologías tradicionales rígidas (que se oponen a los cambios), y que requieren de documentación para cada fase.

Este grupo de impulsores crearon “The Agile Alliance”¹⁵, “una organización, sin ánimo de lucro, dedicada a promover los conceptos relacionados con el desarrollo ágil de software y ayudar a las organizaciones para que adopten dichos conceptos. El punto de partida fue el Manifiesto Ágil, un documento que resume la filosofía “ágil”.”¹⁶

2.2.1.1. Manifiesto ágil¹⁵

El manifiesto ágil no es más que las buenas prácticas que los desarrolladores deben sobrellevar para implementar y ejecutar una mayor agilidad en el proceso. Los valores del manifiesto son:

- **Los Individuos e interacciones están sobre los procesos y herramientas.** El éxito de un proceso de software con una metodología ágil se basa en el factor humano. Es más importante establecer un equipo humano y que éste cree su ambiente laboral, mas no al revés, es decir crear un entorno y que las personas se adapten.
- **Desarrollar software que funcione está sobre una gran documentación.** El desarrollo de software es fundamental y se lo prioriza, solamente cuando es necesario y de manera inmediata se crearán documentos en cuestiones de decisiones importantes sobre el proyecto.

¹⁴ Extracto tomado de documento de ayuda de Ingeniería de Software I (De la Cuz, 2014)

¹⁵ “Agile Alliance is a nonprofit organization with global membership, committed to advancing Agile development principles and practices.” Tomado de la página web de (Agile Alliance, 2016)

¹⁶ Extracto tomado de (Grupo ISSI, 2003)

- **La colaboración con el cliente está sobre la negociación de contratos.** La colaboración del cliente con el equipo desarrollador es continua a lo largo del proceso ya que solo así se garantiza el avance y la culminación exitosa del proyecto; el contrato pasa a segundo plano mientras exista comunicación entre las partes.
- **La respuesta frente a cambios está sobre seguir el plan.** La flexibilidad que tenga el equipo de desarrollo frente a cambios en varios aspectos como pedidos del cliente, nuevos requerimientos, cambios dentro del equipo, tecnología entre otras, logrará que la planificación no sea muy cerrada y estricta.

Estos valores han motivado a diferenciarse de otras metodologías tradicionales mediante los 12 principios (Agile Alliance, 2016). Los dos principios iniciales resumen de buena manera los valores de la metodología ágil, mientras que los siguientes se enfocan en el trabajo del equipo de desarrollo y el proceso que deben seguir.

1. **La máxima prioridad es satisfacer al cliente a través de la entrega a tiempo y software de gran utilidad.** Si el cliente tiene éxito con el producto entregado, el equipo de desarrollo cumplió sus objetivos ya que no solo se trata de crear algo y entregar, sino crecer con el cliente.
2. **Los cambios son bienvenidos, incluso después del desarrollo. Los procesos ágiles aprovechan los cambios para obtener ventajas competitivas del cliente.** Este principio resume el espíritu de la metodología ágil que responde a la flexibilidad del equipo a los cambios que pudieran surgir a lo largo del proyecto.
3. **Entregar el software frecuentemente, desde un par de semanas hasta un par de meses, dando preferencia al corto plazo.** El equipo de desarrollo debe entregar periódicamente productos (prototipos) para que exista la comunicación con el cliente y cumplir con sus expectativas, logrando una sinergia para el éxito del proyecto.
4. **Los clientes y desarrolladores deben trabajar juntos diariamente durante la ejecución del proyecto.** Es la sinergia entre el equipo de desarrollo y el cliente para generar un producto de software de calidad a gusto del cliente.
5. **Construir proyectos en torno a motivaciones individuales, ofreciéndoles el ambiente y el apoyo necesario, confiando en ellos para hacer el trabajo.** El equipo debe estar motivado y enfocado en el proyecto ya que depende de las habilidades individuales de los integrantes del equipo para finalizar el proyecto.

6. **El método más eficiente y efectivo de transmitir información al equipo o hacia el equipo es la conversación directa cara a cara.** Se prioriza reuniones o citas en donde la comunicación fluya en lugar de comunicación escrita dentro del equipo de desarrollo.
7. **Software que funciona es la medida principal de progreso o avance.** Los prototipos creados y posteriormente revisados por el cliente marcan el progreso del proyecto, si un prototipo tiene cambios o mejoras se lo realiza y finalmente se avanza al siguiente prototipo.
8. **Los procesos ágiles promueven el desarrollo sostenible.** Los patrocinadores, desarrolladores y usuarios deben ser capaces de mantener un ritmo constante de forma indefinida. El desarrollo debe ser continuo hasta culminar el proyecto sin interrupciones y manteniendo la comunicación constante.
9. **La atención continua a la excelencia técnica y buen diseño mejora la agilidad.**
10. **Simplicidad es el arte de maximizar la cantidad de trabajo no realizado.**
11. **Las mejores arquitecturas, requisitos y diseños surgen de equipos auto organizados.** El equipo es responsable de ofrecer las mejores soluciones y para ello se requiere de equipos organizados por ellos mismos.
12. **En intervalos regulares, el equipo reflexiona sobre cómo ser más efectivos, luego, se afina y en consecuencia mejora el rendimiento.** Al culminar un prototipo o entregable se realiza una reunión para analizar mejoras que pueden ser implementadas en el siguiente prototipo de tal forma que con cada iteración el rendimiento del equipo sea el más óptimo.

2.2.1.2. Comparación de metodologías

Metodologías Ágiles	Metodologías Tradicionales
Basadas en heurísticas provenientes de prácticas de producción de código	Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo
Especialmente preparados para cambios durante el proyecto	Cierta resistencia a los cambios
Impuestas internamente (por el equipo)	Impuestas externamente
Proceso menos controlado, con pocos principios	Proceso mucho más controlado, con numerosas políticas/normas
No existe contrato tradicional o al menos es bastante flexible	Existe un contrato prefijado
El cliente es parte del equipo de desarrollo	El cliente interactúa con el equipo de desarrollo mediante reuniones
Grupos pequeños (<10 integrantes) y trabajando en el mismo sitio	Grupos grandes y posiblemente distribuidos
Pocos artefactos	Más artefactos

Pocos roles	Más roles
Menos énfasis en la arquitectura del software	La arquitectura del software es esencial y se expresa mediante modelos

Tabla 2-1 Diferencias entre metodologías agiles y tradicionales.

Fuente: (Grupo ISSI, 2003)

2.2.2. Extreme Programming (XP)

La programación extrema (Extreme Programming)¹⁷ nació en 1996, como una metodología ágil en la que se enfoca en la relación cercana del cliente y el equipo de desarrollo e implementado con éxito en muchas compañías en todo el mundo.

XP se define como un método evolutivo¹⁸ ya que está creado para proyectos de alto riesgo con requerimientos imprecisos y cambiantes.

XP se basa en la retroalimentación continua del cliente y el equipo de desarrollo, en la comunicación fluida a la interna del equipo, y a cambios que pudiera sufrir el proyecto, cumpliendo de esta forma los principios de la metodología ágil.

Los autores de XP han decidido agrupar en tres metodologías los principios de la metodología ágil, a continuación, se presenta el gráfico 2-3 con las metodologías.

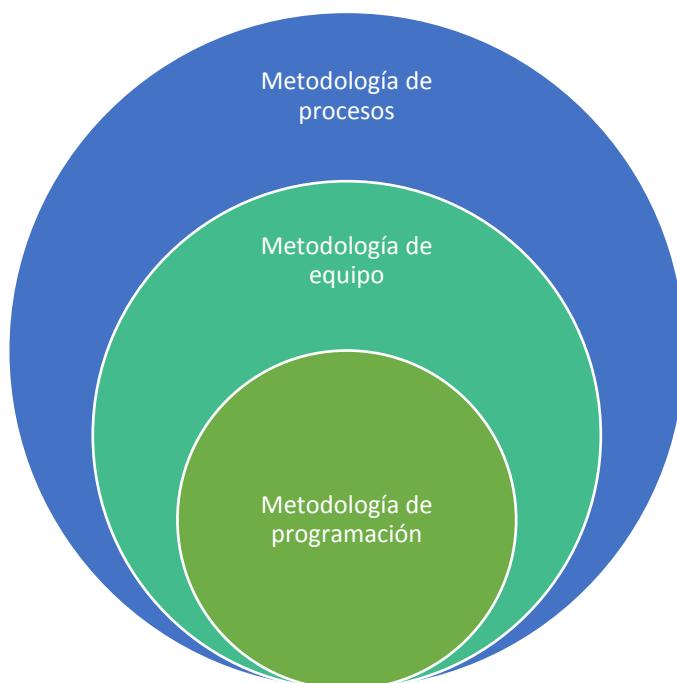


Gráfico 2-3 Estructura en 3 capas de XP.
Fuente: (Fundación para la Universitat Oberta de Catalunya, 2012)

Metodología de programación.

La metodología de programación responde a temas internos del equipo de trabajo.

¹⁷ (Wells, 2009)

¹⁸ (De la Cuz, 2014)

- La excelencia técnica aplicando estándares de desarrollo.
- Simplicidad para reducir el trabajo

Metodología de equipo

Esta metodología se enfoca en el equipo de desarrollo como un grupo humano

- El equipo debe entregar el software de calidad y a tiempo.
- La comunicación interna del equipo es importante para agilitar el proceso.
- Desarrollo sostenible, en donde el equipo codifica, prueba y aplica mejoras aplicando estándares de desarrollo.

Metodología de procesos

La metodología de procesos se apoya en las anteriores para culminar eficientemente el proyecto.

- Ya que el software se lo divide en prototipos, el tiempo de entrega es periódico por lo que se debe distribuir la carga laboral con el tiempo disponible.
- El cliente es parte del equipo para eliminar posibles inconsistencias en el proyecto.
- El proceso va marcado con la entrega de estos prototipos.

2.2.2.1. Valores de XP

XP está basado en valores¹⁹ para encontrar una armonía en el equipo de desarrollo.

- **Simplicidad.** Se desarrollará estrictamente lo que se requiere y no más allá.
- **Comunicación.** Todos los involucrados son parte del equipo, incluso el cliente y todos en conjunto se ofrecerá la mejor solución al requerimiento.
- **Retroalimentación.** Todos los comentarios, sugerencias o mejoras serán tomados en cuenta por parte del equipo de desarrollo
- **Respeto.** Todo el mundo da y recibe respeto como miembros del equipo. Los desarrolladores respetan la experiencia y conocimiento del cliente y viceversa.
- **Coraje.** Se dice la verdad con respecto al avance y estimaciones del proyecto, en donde no existen excusas y la intención es culminar el proyecto exitosamente, adaptándose a cambios cuando éstos se produzcan.

A continuación, se describen las características principales de XP en varios subtemas.

¹⁹ (Wells, 2009)

2.2.2.2. Roles²⁰

XP permite ejecutar varios roles a una sola persona, incluso si se requiere el cambio de rol en diferentes fases del proyecto. XP define varios roles, aunque los más importantes son:

Programador

- Codifica el programa
- Realiza las pruebas unitarias

Cliente

- Escribe las historias de usuario y las jerarquiza
- Diseña las pruebas de aceptación
- Aporta conocimiento del negocio al equipo de desarrollo

Tester (Responsable de las pruebas)

- Ayuda al cliente a diseñar las pruebas de aceptación
- Ejecuta las pruebas de aceptación
- Entrega el reporte de las pruebas al equipo de desarrollo

Tracker (Responsable de seguimiento)

- Se encarga de la administración de la metodología y el proceso XP
- Realiza el seguimiento de cada iteración
- Analiza si es factible incorporar o eliminar las historias de usuario

Boss

- Máximo responsable del proyecto
- Es el enlace entre el cliente con el equipo de desarrollo
- Coordina y garantiza las condiciones necesarias para el trabajo

Existen otros roles como:

Coach (Entrenador)

Se responsabilizan de ayudar al equipo con respecto a XP de forma que se aproveche al máximo esta metodología.

Consultor

No pertenece al equipo, es de gran ayuda en ciertos temas donde el equipo requiera un trabajo externo.

²⁰ (Fundación para la Universitat Oberta de Catalunya, 2012)

2.2.2.3. Ciclo de vida

XP aplica su modelo de simplicidad en el ciclo de vida, como una comunicación entre programador y cliente, asimilando el dialogo en el grafico 2-4.



Gráfico 2-4 Dialogo entre cliente – desarrollador.

Fuente www.paredro.com/5-maneras-de-optimizar-la-comunicacion-entre-los-clientes-y-disenadores/

En la tabla 2-2, se describe el dialogo cliente-desarrollador conjuntamente con el ciclo de vida.

Ciclo de vida	Cliente	Desarrollador
Exploración	Tengo una necesidad.	
		Puedo ayudarte, ¿Cuál es tu necesidad?
	Necesito esto...	
Planificación		Creo que lo entiendo, ¿Si construyo esto te sirve?
	Pues sí, ¿Podrías hacerlo?	
		Ahora mismo me pongo a trabajar.
Iteraciones	¿Has acabado?	
		Estoy en ello.
	Necesitaría también esto.	
		¿Es importante? Tardaría mucho tiempo.
	Entonces no lo pongas.	
Producción		Ya tengo esto. ¿Te sirve?
	No del todo.	
Mantenimiento		¿Y esta nueva versión?
	Sí, es justo lo que necesito. Gracias	
Mantenimiento		De nada. Si tienes problemas, avísame

Tabla 2-2 Dialogo entre cliente y desarrollador.

Fuente: (Fundación para la Universitat Oberta de Catalunya, 2012)

Las fases del ciclo de vida son las que se exponen en el grafico 2-5

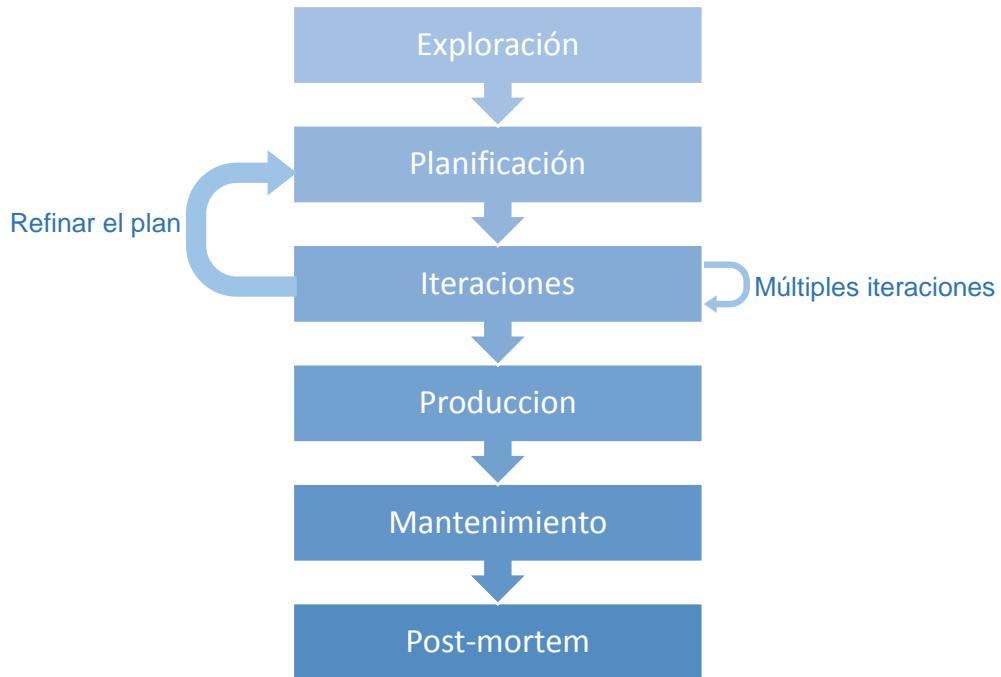


Gráfico 2-5 Fases del ciclo de vida
Fuente (Antonio Córdova, 2016)

2.2.2.3.1. Fase de Exploración²¹

Es la fase inicial del ciclo de vida de XP en donde se desarrollan los siguientes procesos

Historias de usuario

Tienen el mismo propósito de los casos de uso²² aunque no son los mismos ya que son los clientes quienes escriben en su propio lenguaje sin terminología técnica, las funcionalidades que desean obtener del aplicativo.

Se utilizan para realizar estimaciones de tiempo, para documentar los requerimientos, para la realización de pruebas de aceptación de las iteraciones. Gracias a las historias de usuarios se podrá conocer si se ha implementado correctamente tal y como lo solicitó el cliente.

No se debe confundir a las historias de usuarios con el análisis de requerimientos, pues la diferencia está en la profundidad del análisis, en otras palabras, el análisis de requerimientos está realizado por un analista con un alto enfoque técnico, mientras que

²¹ (Wells, 2009)

²² "Los casos de uso se basan en la descripción de escenarios (casos) en los cuales los usuarios interactúan con el sistema que se ha definido para alcanzar un objetivo o para cumplir una tarea particular" (De la Cruz, 2014)

las historias de usuario están redactadas por los clientes que generalmente no tienen conocimientos técnicos.

Las historias de usuario pueden tener una duración de 1, 2, y hasta 3 semanas para desarrollar lo descrito en la historia y es considerado como el tiempo estimado ideal, si se requiere más tiempo se deberá dividir la historia, o por el contrario si la historia es demasiado corta se la debe agrupar en una historia.

Spike arquitectónico

El equipo de desarrollo debe familiarizarse con la metodología, herramientas, lenguaje, en fin, el spike arquitectónico es todo el entorno de trabajo con el cual se va a realizar el proyecto, mientras el cliente describe las historias de usuario.

La metáfora del negocio

La metáfora del negocio es algo que se utiliza para explicar de manera clara y simple, pero con detalles, qué es lo que se va a desarrollar. Es importante este apartado, ya que sirve de ayuda en la comunicación entre el cliente y el equipo de desarrollo, pues el cliente se siente a gusto refiriéndose al proyecto (sistema) en sus propios términos, mientras que, para el equipo de desarrollo la metáfora es útil para implementar las clases y objetos con el cual se está trabajando.

2.2.2.3.2. Fase de Planificación

El cliente debe entregar las historias de usuario jerarquizándolas según su importancia, ya que, si el alcance es modificado, solo se afectarán aquellas iteraciones que no sean importantes. Luego, el equipo realiza un estudio de las mismas y estima el costo por historia. Finalmente, se convoca a una reunión para planificar la entrega del producto dividido en iteraciones, es decir la entrega del producto será dividido en partes.

Estas partes son los planes de iteraciones en las cuales se especifica las funcionalidades que se implementarán en cada iteración. Éstas a su vez asignan un tiempo de entrega que deben ser más o menos similares entre ellos. De esta forma se determina el alcance del proyecto.

Pruebas de aceptación

Para aceptar el alcance del proyecto se deberá crear las pruebas de aceptación que son creadas a partir de las historias de usuario, en otras palabras, el cliente crea las historias de usuario al igual que los escenarios posibles para probar el correcto funcionamiento de dicha historia de usuario. El cliente podría proponer una o varias pruebas por cada historia de usuario.

Cada prueba de aceptación representa un resultado esperado del sistema y el cliente es el responsable de verificar que los resultados sean positivos, si existen resultados negativos se debe crear una lista de ellos, organizados en orden de importancia. Estas pruebas, al ser creadas por el cliente y en la etapa de planificación, no deben tener mucho detalle, pero lo que se busca es garantizar la calidad y correcto funcionamiento según lo que busca el cliente.

La responsabilidad del equipo de desarrollo es programar el tiempo de cada iteración para solucionar las pruebas negativas. Finalmente, las pruebas aceptadas o positivas cambian de nombre a pruebas funcionales, que garantiza la calidad del producto desarrollado.

2.2.2.3. Fase de Iteraciones²³

Al inicio de cada iteración siempre debe existir una reunión para planificar el trabajo. Las iteraciones se subdividen en tareas de dos o tres días según su tamaño, que se entregan al equipo para desarrollarlas. Las que sobran, no tienen asignación del equipo y conforme se avance el desarrollo, se irán tomando. Esta fase se muestra en la figura 2-6.

Debe existir una reunión diaria para dar seguimiento al avance del proyecto de tal forma que se cumplan las estimaciones, para ello existe el rol del Tracker.



Gráfico 2-6 Iteraciones.
Fuente: (Wells, 2009)

²³ (Wells, 2009)

2.2.2.3.4. Fase de Producción

Cuando la iteración es aceptada por el cliente (es decir si la iteración tiene valor real para el cliente) se la pone en producción, sin esperar a que se termine todas las iteraciones manteniendo esta fase hasta culminar con todas las iteraciones. En ese momento, se cambia a la siguiente fase de mantenimiento.

2.2.2.3.5. Fase de Mantenimiento

Cuando ya se tienen todas las iteraciones en producción, y el alcance se ha completado, se revisa conjuntamente con el cliente nuevas historias de usuario que hayan quedado por realizar tras la puesta en producción. Estas historias se incorporarán si se dispone del presupuesto necesario.

2.2.2.3.6. Fase de post-mortem

Cuando el cliente ya no cuenta con más historias de usuario y el trabajo se ve reducido al mínimo entra en fase de muerte.

2.2.2.4. Reglas de XP²⁴

Las reglas de XP se definen gráficamente como un autómata finito²⁵ en donde interviene todo el ciclo de vida expuesto anteriormente. En este modelo se representa a través del grafico 2-7 los estados que atraviesa un proyecto hasta obtener un prototipo, es decir al culminar una iteración.

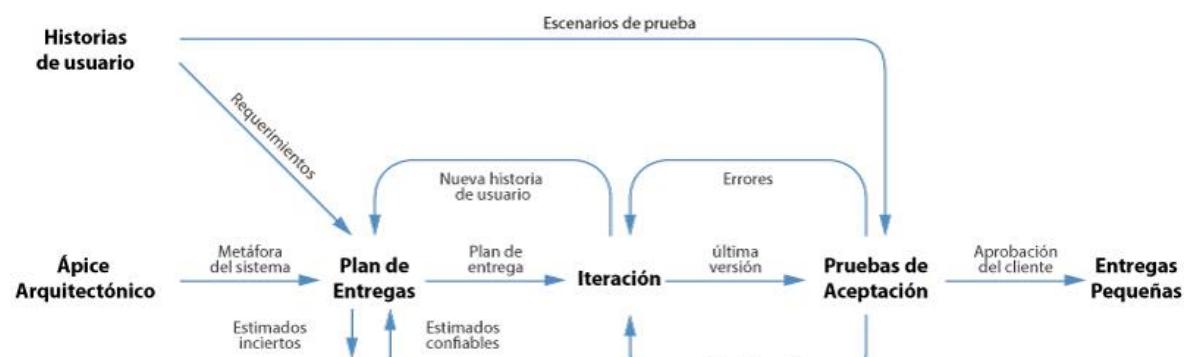


Gráfico 2-7 Reglas de XP.
Fuente: (Wells, 2009)

²⁴ (Wells, 2009)

²⁵ Definir autómata finito

Se visualizan 5 reglas principales:

- **Planificación.** Contiene las historias de usuario, la planificación de iteraciones y el calendario de entrega
- **Administración.** Se refiere a las reuniones periódicas de entrega, reuniones diarias para el control de avance de la iteración. También se resuelven problemas.
- **Diseño.** Donde se especifica la metáfora del negocio, se crean soluciones frente a la historia de usuario.
- **Codificación.** El desarrollo en sí de los prototipos los cuales se ejecutan en pareja.
- **Pruebas de aceptación.** Se prueba el producto desarrollado mediante la unidad de pruebas. Finalmente, si todas las pruebas son favorables, el producto pasa a producción.

2.2.3. Justificación de la metodología de desarrollo

Una metodología ágil se basa en el modelo evolutivo, es decir, es un modelo flexible el cual permite incrementos futuros de nuevas funcionalidades o mejoras al planificado. (De la Cuz, 2014)

Se implementará el modelo evolutivo debido a que es probable que haya cambios en algunas fases del proceso. Por otra parte, la idea de desarrollar este sistema ha surgido por iniciativa personal, lo que conlleva a que esté directamente involucrado en todo el ciclo de vida.

El plan de disertación de grado define en su cronograma un tiempo de ejecución de 5 meses, asimismo, la metodología evolutiva a través de XP está diseñada para desarrollos cortos y rápidos por lo que esta metodología se presenta como la mejor alternativa para aplicar la construcción del sistema de esta disertación de grado.

2.2.4. Costo del Proyecto Programado

El costo preliminar del proyecto se ha realizado en base al cronograma fijado en el plan de disertación de grado, el cual especifica un trabajo de 5 meses a tiempo completo por parte de todo el equipo, en el cual se especifica las etapas de análisis, diseño, desarrollo, documentación y entrega.

Se prevé 3 etapas generales para el presente proyecto:

- Diseño
- Desarrollo

- Entrega Final

En cuanto al valor por área está especificado en la tabla 2-3:

Puesto	Costo por hora (\$)
Diseñador	8
Programador	15
Tester	10
Otros	Entre 5 y 8

Tabla 2-3 Costo por hora por puesto.

Fuente (Antonio Córdova, 2016)

En la tabla 2-4, se cotiza la mano de obra por hora dependiendo de las tareas del proyecto.

Costo Proyecto Programado				
	TAREAS DEL PROYECTO	HORAS MANO OBRA	COSTO MANO OBRA (\$)	TOTAL POR TAREA
DISEÑO DEL PROYECTO	Desarrollar especificaciones funcionales	40.0	8.00	320.00
	Desarrollar arquitectura del sistema	10.0	8.00	80.00
	Desarrollar especificaciones de diseño preliminares	40.0	8.00	320.00
	Desarrollar especificaciones de diseño detalladas	20.0	8.00	160.00
	Desarrollar plan de prueba de aceptación	30.0	8.00	240.00
	Subtotal	140.0	40.00	1120.00
DESARROLLO DEL PROYECTO	Desarrollar componentes	400.0	15.00	6000.00
	Obtener software	0.0	0.00	0.00
	Obtener hardware	0.0	0.00	0.00
	Desarrollar paquete de prueba de aceptación	20.0	10.00	200.00
	Realizar prueba de unidad / integración	60.0	10.00	600.00
	Subtotal	480.0	35.00	6800.00
ENTREGA DEL PROYECTO	Instalar el sistema	5.0	8.00	13.00
	Entrenar a los clientes	4.0	8.00	12.00
	Realizar prueba de aceptación	20.0	8.00	28.00
	Realizar revisión posterior al proyecto	10.0	8.00	18.00
	Documentación Técnica y del usuario del sistema	32.0	5.00	37.00
	Subtotal	71.0	37.00	108.00

Subtotales	691.0	112.00	8028.00
Total (programado)	691.0	112.00	8028.00

Tabla 2-4 Costo proyectado.
Fuente (Antonio Córdova, 2016)

CAPÍTULO 3

3. CASO DE ESTUDIO: APLICACIÓN DE XP EN LA CONSTRUCCIÓN DEL SISTEMA

A continuación, se aplicará la metodología XP en el desarrollo del presente proyecto a través de sus fases analizadas en el capítulo anterior, de forma que se desarrolle el producto a través de varias iteraciones hasta conseguir la versión final para ponerla en producción.

3.1. Fase de exploración

La fase de exploración es la fase inicial en la cual interviene el cliente de manera directa quien detalla las funcionalidades del sistema, en este caso el cliente tiene una idea de negocio que desea poner en marcha con la creación de una página web para ofrecer intercambios virtuales de productos a través de códigos QR.

Por motivos académicos y la presente disertación de grado, se ha definido las siguientes personas en la tabla 3-1.

Persona (ROLES)			Nombre
Programador 1 y 2	Cliente	Tracker	Antonio Córdova
	Coach		Ing. Anita Urgilés
Tester 1 y 2			Estudiantes seleccionados por tracker

Tabla 3-1 Roles por persona.
Fuente (Antonio Córdova, 2016)

El proyecto se lo ha denominado “**Find Ur Gift**” que en español significa encuentra tu regalo.

3.1.1. Historias de usuario

Se solicita al cliente las historias de usuario que deben ser escritas con ayuda del equipo de desarrollo, las mismas que se especifican a continuación.

Historia de usuario 01	Presencia en internet
Quiero que la gente pueda acceder a mi sistema de forma fácil y encontrar mis datos personales para que me contacten.	

Tabla 3-2 Historia de usuario: Presencia en internet.

Fuente (Antonio Córdova, 2016)

Historia de usuario 02	Paquetes de tarjetas
<p>Los paquetes representan las ventas, es decir, se vende a mis clientes un tipo de tarjeta que puede contener mínimo 5 códigos para que sean repartidos a sus usuarios y estos a su vez puedan acceder a la página para canjear los códigos por productos.</p> <p>Los paquetes deben estar en un sitio visible de la página para promocionarlas y venderlas a través de ella.</p>	

Tabla 3-3 Historia de usuario: Paquetes de tarjetas.

Fuente (Antonio Córdova, 2016)

Historia de usuario 03	Catálogo de productos
<p>Los productos no deben estar visibles en la página web, ya que no se venden los productos sino tarjetas. Únicamente, cuando un usuario ingresa el código puede visualizar los productos para que escoja uno de ellos.</p> <p>El catálogo es el total de los productos creados en la página web</p>	

Tabla 3-4 Historia de usuario: Catálogo de productos.

Fuente (Antonio Córdova, 2016)

Historia de usuario 04	Orientación de la página
<p>Esta página está orientada a empresas que quieran comprar mis tarjetas, por otro lado, la página debe ofrecer un espacio para que los usuarios que tengan una especie de código, puedan ingresar al sistema y ver los productos</p>	

Tabla 3-5 Historia de usuario: Orientación de la página.
Fuente (Antonio Córdova, 2016)

Historia de usuario 05	Proceso de canje
<p>Después de que el usuario escoge el producto deseado, debe seguir un proceso para que yo pueda identificarlo a través de sus datos personales y además conocer su dirección para el envío de su regalo.</p>	

Tabla 3-6 Historia de usuario: Proceso de canje.
Fuente (Antonio Córdova, 2016)

Historia de usuario 06	Pedidos de productos
<p>El sistema debe mostrarme una lista de todos los pedidos que han sido realizados para que pueda organizar los envíos.</p>	

Tabla 3-7 Historia de usuario: Pedidos de productos.
Fuente (Antonio Córdova, 2016)

Historia de usuario 07	Empresas
<p>Mis clientes directos son empresas que van a comprar mis tarjetas, ellos deben tener un acceso para gestionar los paquetes que me compraron. Mis clientes deben seleccionar por ellos mismos que productos mostrar a cada uno de sus usuarios.</p>	

Tabla 3-8 Historia de usuario: Empresas.
Fuente (Antonio Córdova, 2016)

Historia de usuario 08	Usuarios
Los usuarios finales son aquellas personas que les entregaron los códigos para que puedan canjearlos por productos. Quiero que los usuarios puedan ingresar al sistema y seleccionar los productos.	

Tabla 3-9 Historia de usuario: Usuarios.
Fuente (Antonio Córdova, 2016)

Historia de usuario 09	Administrador
Para administrar el sitio quiero tener la posibilidad de crear los paquetes, los códigos, las empresas, es decir administrar todo el sitio.	

Tabla 3-10 Historia de usuario: Administrador.
Fuente (Antonio Córdova, 2016)

Historia de usuario 10	Reportes
Se debe obtener la lista de los pedidos como administrador y que las empresas puedan obtener la lista de quienes han canjeado sus códigos.	

Tabla 3-11 Historia de usuario: Reportes.
Fuente (Antonio Córdova, 2016)

Historia de usuario 11	Compra en línea
Los clientes podrán comprar mis tarjetas en línea para evitar que vengan a mi oficina a cancelar	

Tabla 3-12 Historia de usuario: Compra en línea.
Fuente (Antonio Córdova, 2016)

3.1.2. Spike arquitectónico

El desarrollo de este sistema se basará en PHP a través del framework Laravel 5.0, ya que el equipo de desarrollo ha decidido utilizar la arquitectura clásica en tres capas.

Existía una duda si utilizar PHP puro o Laravel. Se optó por Laravel debido a que cuenta con un manejo bastante simple de variables de sesión necesario para el sistema.

Los pagos en línea en el país aún es un proceso complicado por lo que se le ofrecerá al cliente pagos a través de PayPal.

El equipo de desarrollo está familiarizado con herramientas de código abierto y que son compatibles entre sí y están plenamente integradas. Para ello se ha dividido la estación de trabajo en dos entornos:

Entorno local

WampServer²⁶. Es un entorno de desarrollo web para Windows que permite crear páginas web.

Entorno Web (producción)

Web hosting contratado con las siguientes descritas en la tabla 3-13.

Almacenamiento	3000 MB
Transferencia mensual de información	10000 MB
Cantidad de subdominios	libre
Número de cuentas email	libre
Cantidad de envíos por hora	50
Cantidad de Bases de Datos	10
Panel de control	Hepsia
FTP	libre
Php 5	sí
Estadísticas	sí
Backups auto administrados	sí

Tabla 3-13 Características Web-Hosting.

Fuente uk.cloudlogin.co

²⁶ (Alter Way, 2016)

Los dos ambientes trabajan con los siguientes datos:

- **Servidor:** Apache v. 2.4
- **Lenguaje:** PHP v. 5.5.9
- **Motor de BD:** MySQL libre o MariaDB 10.0.20
- **Administrador de base de datos:** PhpMyAdmin v. 4.3.7

3.1.3. Metáfora del negocio

El último paso de esta fase es definir la metáfora del negocio para que el cliente y el equipo de desarrollo se comuniquen en los mismos términos.

- **Dominio.** Es el nombre con el cual se reconoce al sistema en internet, en este caso será “*findurgift.webcordova.com*”.
- **BDD.** Siglas de Base de Datos
- **Ambiente de desarrollo.** *Espacio o área de trabajo en donde los programadores realizarán la construcción del sistema.*
- **Menú.** Listado de links o botones que sirven para la navegación entre las diferentes opciones que puede tener el sistema.
- **Tarjetas:** Al igual que una tarjeta física con un valor monetario máximo para consumo, la diferencia en el sistema es que se crea una tarjeta con precio máximo y en base a ella se crearán los códigos de acceso para los usuarios
- **Código de acceso.** Son unos caracteres que el usuario deberá ingresar para conocer que premios puede escoger.
- **Catálogo de tarjetas.** Es la reunión de todos los paquetes creadas en el sistema. Sería como un catálogo de productos, pero al ser digital, puede cambiar dinámicamente.
- **Proceso de canje.** El proceso de canje funcionará de esta forma: Un cliente solicita servicio a domicilio en un restaurante de la ciudad, para ello busca el número telefónico del restaurante, realiza la llamada, selecciona a su gusto los productos del menú, el cliente deberá entregar sus datos para la factura y el lugar a donde se le envíe el pedido.
- **Pedidos de productos.** Para continuar con la analogía anterior, pero desde el lado del restaurante, el encargado de los pedidos recibe una lista con todos los pedidos detallando los ítems del menú, datos del cliente y las direcciones de entrega, este documento se le entrega al motorizado con los productos para proceder a su entrega a domicilio.

- **Empresas.** Son los usuarios administradores de los paquetes que compraron. La idea es similar a una tienda de regalos, en donde el administrador de la tienda exhibirá los productos que crea necesario para sus clientes.
- **Usuarios finales.** Son los clientes poseedores de los códigos de acceso.
- **Administrador.** Es la persona quien gestionará todos los procesos del sistema. Es como el gerente de una tienda quien propone los precios, coordina el trabajo, etc.

3.2. Fase de planificación

El cliente entrega las historias de usuario y el equipo de desarrollo evalúa costos, distribución del trabajo y tiempos de desarrollo en jornadas de 8 horas cada una, como se describe en la tabla 3-14.

Orden	Historia de usuario	Título	Tiempo (jornadas)
1	01	Presencia en internet	2
2	04	Orientación de la página	3
3	03	Catálogo de productos	5
4	08	Usuarios	5
5	02	Paquetes de tarjetas	5
6	07	Empresas	5
7	05	Proceso de canje	6
8	09	Administrador	4
9	10	Reportes	5
10	06	Pedidos de productos	5
11	11	Compra en línea	5
TOTAL			50

Tabla 3-14 Historias de usuario ordenadas.

Fuente (Antonio Córdova, 2016)

El equipo se reúne para planificar las fechas para la entrega de cada iteración, estableciendo como iteración cada 5 jornadas es decir entrega semanal.

Como vemos en la tabla siguiente, se crearon nueve iteraciones (8 semanas) con cinco pasos a producción. Cada paso a producción representa un producto visible que puede entrar en funcionamiento

Cabe resaltar que desde la primera semana del proyecto se entregará valor al negocio. Estos datos se presentan en la tabla 3-15.

Historia No.	Título	Tiempo (J)	Plan de entregas (iteraciones)	
			Iter.	Qué se entrega
	Instalación de ambiente de desarrollo	0.5	0	Ambiente de desarrollo preparado para iniciar trabajo
01	Presencia en internet	4	1	1er. Paso a producción
04	Orientación de la página	1		
03	Catálogo de productos	5	2	Gestión básica de productos
08	Usuarios	5	3	Gestión de usuarios. 2do paso a producción
02	Paquetes de tarjetas	5	4	Gestión básica de paquetes. 3ro. paso a producción
07	Empresas	5	5	Gestión básica de empresas. 4to. paso a producción
05	Proceso de canje	5	6	Creación del proceso
05	Proceso de canje	1	7	Culminación del proceso
09	Administrador	4		Creación de zona administrativa. 5to. paso a producción
10	Reportes	5	8	Creación de reportes
06	Pedidos de productos	5		Fuera del ámbito del proyecto
11	Compra en línea	5		

Tabla 3-15 Plan de entrega.

Fuente (Antonio Córdova, 2016)

3.2.1. Estimación de costos

En el capítulo anterior se realizó una estimación de costo del proyecto en base al plan de disertación de grado, en el cual se concluye un presupuesto de \$ 8.028,00. En esta ocasión, la estimación se realiza en base a las historias de usuario y el tiempo establecido que es de 50 jornadas.

Rol	Costo (Hora)	Costo (jornadas)	Tiempo (jornadas)	Total (\$)
Diseñador	8.00			
Programador	15.00	120.00	50.00	6000.00
Tester	10.00	80.00	25.00	2000.00
Apoyo	5.00	40.00	2	80.00
TOTAL	30.00	240.00		8080.00

Tabla 3-16 Estimación de costos.

Fuente (Antonio Córdova, 2016)

La tabla 3-16 indica el valor más aproximado a la realidad después de conocer las historias de usuario proporcionado por el cliente. Esta tabla muestra un valor muy aproximado a la estimación inicial, lo que ratifica una acertada visión del tamaño del proyecto con respecto al tiempo de desarrollo del equipo.

3.2.2. Pruebas de aceptación

Las pruebas de aceptación han sido ordenadas por iteraciones, las cuales son descritas en las siguientes tablas:

3.2.2.1. Iteración 01. Primer paso a producción

Prueba de aceptación 01	- Historia de usuario 01 - PRESENCIA EN INTERNET	Iteración 01
Precondiciones	El ambiente de desarrollo debe estar listo para dar inicio al desarrollo	
Entradas	1. El usuario ingresa a google y busca a través del dominio o ingresa al sitio web. 2. El usuario ubica la página y da clic en el enlace	
Resultados esperados	La página debe estar habilitada y mostrar los datos de contacto	
Resultado final	La página web muestra de manera destacada los datos indicados	
Prueba de aceptación 02	- Historia de usuario 04 - ORIENTACIÓN DE LA PÁGINA	Iteración 01
Precondiciones	Los paquetes y productos deben ser creadas previamente.	
Entradas	1. El usuario ingresa a la página web	
Resultados esperados	La página debe mostrar los paquetes y mostrar un espacio para ingresar el código.	
Resultado final	La página web muestra de manera destacada los datos indicados	

Tabla 3-17 Pruebas de aceptación 1 y 2 – Primer paso a producción.

Fuente (Antonio Córdova, 2016)

3.2.2.2. Iteración 02. Gestión básica de productos

Prueba de aceptación 03	- Historia de usuario 03 - CATÁLOGO DE PRODUCTOS	Iteración 02
Precondiciones	<ol style="list-style-type: none"> Para visualizar los productos el administrador debe crear los productos. La empresa debe seleccionar los productos que se muestran a los usuarios. 	
Entradas	<ol style="list-style-type: none"> El usuario ingresa a la página web Digita el código o escanea el código con su teléfono móvil 	
Resultados esperados	El sistema debe mostrar los productos disponibles para el usuario.	

Tabla 3-18 Prueba de aceptación - Catálogo de productos.

Fuente (Antonio Córdova, 2016)

3.2.2.3. Iteración 03. Gestión de usuarios

Esta iteración culmina con el segundo paso a producción.

Prueba de aceptación 04	- Historia de usuario 08 - GESTIÓN DE USUARIOS	Iteración 03
Precondiciones	<p>Para que el usuario pueda ingresar al sistema, se debe crear códigos de acceso.</p> <p>Para visualizar los productos, la empresa debe seleccionar productos.</p>	
Entradas	<ol style="list-style-type: none"> El usuario ingresa a la página web Digita el código o escanea el código con su teléfono móvil 	
Resultados esperados	<p>El sistema debe presentar la página principal del usuario.</p> <p>El sistema debe mostrar los productos disponibles para el usuario.</p>	

Tabla 3-19 Prueba de aceptación 04 - Gestión de usuarios – 2do. Paso a producción.

Fuente (Antonio Córdova, 2016)

3.2.2.4. Iteración 04. Gestión básica de paquetes

Esta iteración culmina con el tercer paso a producción.

Prueba de aceptación 05	- Historia de usuario 02 - GESTIÓN BÁSICA DE PAQUETES	Iteración 04
Precondiciones	El administrador del sistema debe crear los paquetes previamente	
Entradas	1. El usuario empresarial ingresa a la página web	
Resultados esperados	La página debe mostrar los paquetes habilitadas y mostrar características de los paquetes	

Tabla 3-20 Pruebas de aceptación 05- Gestión básica de paquetes – 3er. Paso a producción.

Fuente (Antonio Córdova, 2016)

3.2.2.5. Iteración 05. Gestión básica de empresas

Esta iteración culmina con el cuarto paso a producción.

Prueba de aceptación 06	- Historia de usuario 07 - GESTIÓN BÁSICA DE EMPRESAS	Iteración 05
Precondiciones	El administrador debe crear las empresas.	
Entradas	1. La empresa ingresa a la página web 2. La empresa inicia sesión	
Resultados esperados	El sistema debe mostrar la página principal de empresas	

Tabla 3-21 Pruebas de aceptación - Gestión básica de empresas.

Fuente (Antonio Córdova, 2016)

3.2.2.6. Iteración 06. Creación de proceso de canje

Es necesario dividir la historia de usuario número 5 ya que es un proceso que toma más de 5 jornadas de trabajo, por lo tanto, en la presente iteración se concentra en los formularios necesarios para recabar información del cliente como datos personales y dirección de entrega.

Prueba de aceptación 07	- Historia de usuario 05 - CREACIÓN DE PROCESO DE CANJE	Iteración 06
Precondiciones	El usuario debe iniciar sesión.	
Entradas	<ol style="list-style-type: none"> 1. El usuario selecciona el producto deseado. 2. El usuario ingresa datos personales 3. El usuario ingresa dirección de entrega 	
Resultados esperados	El sistema debe mostrar los datos que ingresó el usuario para que confirme si están correctos.	

Tabla 3-22 Pruebas de aceptación - Creación de proceso de canje.

Fuente (Antonio Córdova, 2016)

3.2.2.7. Iteración 07. Creación de zona administrativa

En la presente iteración se culmina el proceso de canje, es decir, la confirmación y posterior procesamiento de datos ingresados por el usuario, finalmente el envío de correo electrónico al usuario con los datos enviados.

Por otra parte, se crea la zona administrativa que controlará el sistema completo. De esta forma se concluye con el quinto paso a producción.

Prueba de aceptación 08	- Historia de usuario 05 - CULMINACIÓN DE PROCESO DE CANJE	Iteración 07
Precondiciones	El usuario debe seguir el proceso de canje inicial	
Entradas	<ol style="list-style-type: none"> 1. El usuario confirma los datos personales y dirección de entrega 	
Resultados esperados	El sistema debe enviar un correo al administrador y un correo al usuario confirmando su canje.	

Tabla 3-23 Pruebas de aceptación 08 – Culminación de canje.

Fuente (Antonio Córdova, 2016)

Prueba de aceptación 09	- Historia de usuario 09 - ZONA ADMINISTRATIVA	Iteración 07
Precondiciones	Se debe crear un usuario administrador	
Entradas	1. El administrador inicia sesión	
Resultados esperados	El sistema debe presentar todas las opciones administrativas	

Tabla 3-24 Pruebas de aceptación 09 - Zona Administrativa.
Fuente (Antonio Córdova, 2016)

3.2.2.8. Iteración 08. Creación de reportes

Prueba de aceptación 10	- Historia de usuario 10 - CREACIÓN DE REPORTES	Iteración 08
Precondiciones	El administrador debe iniciar sesión	
Entradas	1. El administrador hace clic en reportes 2. Selecciona el reporte	
Resultados esperados	El sistema debe exportar los datos requeridos en un archivo de Excel	

Tabla 3-25 Pruebas de aceptación - Creación de reportes.
Fuente (Antonio Córdova, 2016)

3.2.3. Plan de entrega

Se había especificado que cada iteración comprende 5 jornadas (semanal) en las cuales se entregará semanalmente cada iteración, estableciendo así el alcance de esta presente disertación de grado.

3.3. Fase de iteraciones

Se realiza la reunión de trabajo con el equipo de desarrollo y el cliente para describir las tareas que serán desarrolladas. En esta reunión se analiza una nueva prueba de aceptación como alcance a la historia de usuario 06, de tal forma que se pueda dividir la iteración 05 en dos, manteniendo el mismo tiempo de trabajo, pero dos diferentes pruebas de aceptación.

Prueba de aceptación 11	- Historia de usuario 06 - SELECCIÓN DE PRODUCTOS	Iteración 05
Precondiciones	La empresa debe iniciar sesión. El administrador debe crear productos previamente para que las empresas puedan seleccionarlos.	
Entradas	1. La empresa hace clic en ver productos 2. La empresa selecciona los productos interesados	
Resultados esperados	El sistema debe mostrar todos los productos con: <ul style="list-style-type: none"> • Un botón para seleccionar o quitar el producto • Un botón para ver detalles del producto 	

Tabla 3-26 Prueba de aceptación adicional - Selección de productos.

Fuente (Antonio Córdova, 2016)

A continuación, se describe todas las tareas programadas:

IT.	TAREAS	TIEMPO PLAN.	FIN	DIAS
0	Instalación de ambiente de desarrollo	4:00:00	7/12/2015	1
	PHPStorm 2016.1	1:00:00	7/12/2015	
	Instalación de Laravel para el proyecto	1:00:00	7/12/2015	
	Herramienta para el modelado de Base de Datos: Power Designer	1:00:00	7/12/2015	
	Herramienta para administración del sitio local: WAMP	1:00:00	7/12/2015	
1	Presencia en internet	28:00:00	10/12/2015	4
	Diseño de Experiencia de usuario en general	8:00:00	8/12/2015	
	Maquetación de las páginas	8:00:00	9/12/2015	
	Creación de espacio para contacto	2:00:00	9/12/2015	
	Implementación de las páginas en ambiente de trabajo	8:00:00	10/12/2015	
	Pruebas de aceptación	2:00:00	10/12/2015	
1	Orientación de la página	8:00:00	11/12/2015	1
	Creación de espacio para ingreso de usuarios	6:00:00	11/12/2015	
	Creación de espacio para visualización de tarjetas	1:00:00	11/12/2015	
	Pruebas de aceptación	1:00:00	11/12/2015	
2	Gestión básica de productos	40:00:00	18/12/2015	5
	Modelamiento de base de datos	10:00:00	15/12/2015	

	Implementación de la Base de datos en el framework Laravel	4:00:00	15/12/2015	
	Programación del CRUD de productos	20:00:00	18/12/2015	
	Pruebas de aceptación	6:00:00	18/12/2015	
3	Gestión de usuarios	40:00:00	25/12/2015	5
	Diseño de Experiencia de usuario para clientes	8:00:00	21/12/2015	
	Programación CRUD de usuarios	13:00:00	23/12/2015	
	Implementación de ingreso al sistema a través de códigos QR, código	13:00:00	25/12/2015	
	Pruebas de aceptación	6:00:00	25/12/2015	
4	Gestión básica de paquetes	40:00:00	8/1/2016	5
	Diseño de Experiencia de usuario para paquetes	19:00:00	6/1/2016	
	Programación CRUD de tarjetas	15:00:00	8/1/2016	
	Pruebas de aceptación	6:00:00	8/1/2016	
5	Gestión básica de empresas	40:00:00	15/1/2016	5
	Diseño de Experiencia de usuario para empresas	8:00:00	11/1/2016	
	Programación CRUD de empresas	13:00:00	13/1/2016	
	Implementación de ingreso al sistema a través usuario y contraseña	13:00:00	15/1/2016	
	Pruebas de aceptación	6:00:00	15/1/2016	
5	Selección de productos	40:00:00	22/1/2016	5
	Diseño de Experiencia de usuario para la selección de productos	16:00:00	19/1/2016	
	Implementación de la selección de productos	18:00:00	22/1/2016	
	Pruebas de aceptación	6:00:00	22/1/2016	
6	Creación de proceso de canje	40:00:00	29/1/2016	5
	Diseño de Experiencia de usuario para el proceso de canje	16:00:00	26/1/2016	
	Implementación del proceso de canje	20:00:00	29/1/2016	
	Pruebas de aceptación	4:00:00	29/1/2016	
7	Culminación de proceso de canje	8:00:00	1/2/2016	1
	Implementación de envío de correos electrónicos a usuario y administrador	4:00:00	1/2/2016	
	Pruebas de aceptación	4:00:00	1/2/2016	
7	Zona administrativa	32:00:00	5/2/2016	4
	Diseño de Experiencia de usuario para administrador	18:00:00	4/2/2016	
	Implementación de ingreso al sistema a través usuario y contraseña	8:00:00	5/2/2016	
	Pruebas de aceptación	6:00:00	5/2/2016	
8	Creación de reportes	40:00:00	12/2/2016	5
	Diseño de documento Excel de reporte	14:00:00	9/2/2016	
	Implementación de reporte	20:00:00	12/2/2016	
	Pruebas de aceptación	6:00:00	12/2/2016	
	TOTAL	360:00:00		46

Tabla 3-27 Planificación de iteraciones.

Fuente (Antonio Córdova, 2016)

El *tracker* da inicio al seguimiento de las iteraciones mediante la herramienta **ProjectLibre**²⁷,

3.3.1. Iteración inicial

En esta iteración se prepara el ambiente de desarrollo para iniciar el trabajo.

ITER.	TAREAS	TIEMPO PLANIFICADO	PROPIETARIO
0	Instalación de ambiente de desarrollo	4:00:00	Antonio Córdova
	PhpSorm 8.2 y PHP	1:00:00	
	Instalación de Laravel para el proyecto	1:00:00	
	Herramienta para el modelado de Base de Datos: Power Designer	1:00:00	
	Herramienta para administración del sitio local: WAMP	1:00:00	

Tabla 3-28 Iteración inicial - Ambiente de desarrollo.

Fuente (Antonio Córdova, 2016)

El ambiente de desarrollo será local por lo que es indispensable crear un entorno local a través de la aplicación WAMP.

The screenshot shows the WampServer control panel interface. At the top, it displays the WampServer logo and version information (Version 2.5 Version Française). Below this, the "Server Configuration" section provides details about Apache, PHP, and MySQL versions. It also lists loaded extensions such as apache2handler, Core, ctype, curl, fileinfo, filter, hash, ldap, libxml, mysqli, pcntl, PDO, session, SPL, and xdebug. The "Tools" section includes links for phpinfo() and phpmyadmin. The "Your Projects" section lists various projects like awesome, chocolates, email, exalta, galapagos, masipi, phpcode, PHP_Scripts, portal, procesamiento, Proyecto, pruebas, regalospremios, teregalo, webcordova, and weekendsout. The "Your Aliases" section lists aliases for phpmyadmin, phpsinfo, sqbluddy, and webgrind. At the bottom, there are links for WampServer, Donate, and Alter Way.

Gráfico 3-1 LocalHost en WampServer
Fuente (Antonio Córdova, 2016)

²⁷ ProjectLibre fue concebido como una alternativa de código libre en respuesta a Microsoft Project Desktop (ProjectLibre, 2015)

Ya con el servidor encendido, se instala el framework Laravel a través de la consola de Windows, para ello es necesario la herramienta Composer²⁸, con ella se puede instalar o actualizar dependencias de PHP, para ello realizamos los siguientes pasos:

1. Desde la consola de Windows, se dirige al directorio donde se encuentran los proyectos web del servidor, en este caso es el siguiente:

```
cd C:\wamp\www
```

2. Se crea el proyecto Laravel:

```
composer create-project laravel/laravel webcordova --prefer-dist
```

Con el comando anterior se descarga un proyecto de Laravel nuevo con la última versión disponible, con la siguiente estructura:

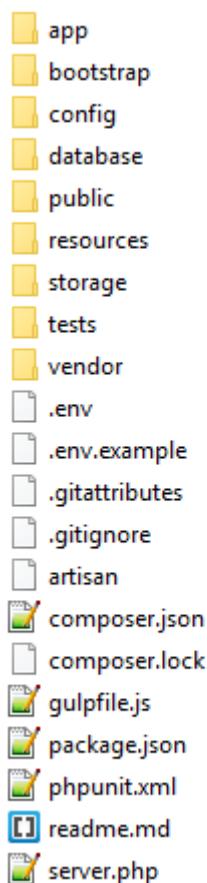


Gráfico 3-2 Estructura de Laravel
Fuente (Antonio Córdova, 2016)

²⁸ Composer es una herramienta que gestiona dependencias de PHP y ayuda a instalar nuevos componentes para PHP (Adermann & Boggiano, 2015)

Esta estructura descargada contiene archivos, principalmente, de PHP los cuales se modifican o se crean dependiendo de la funcionalidad que se requiera, para ello es necesario una herramienta de desarrollo que facilite la programación.

La herramienta seleccionada es PHPStorm el cual es un IDE de desarrollo exclusivo para PHP de pago, pero gracias al convenio de la Pontificia Universidad Católica del Ecuador, se obtuvo una licencia estudiantil para esta herramienta. La captura de pantalla de la licencia se muestra en el grafico 3-3.

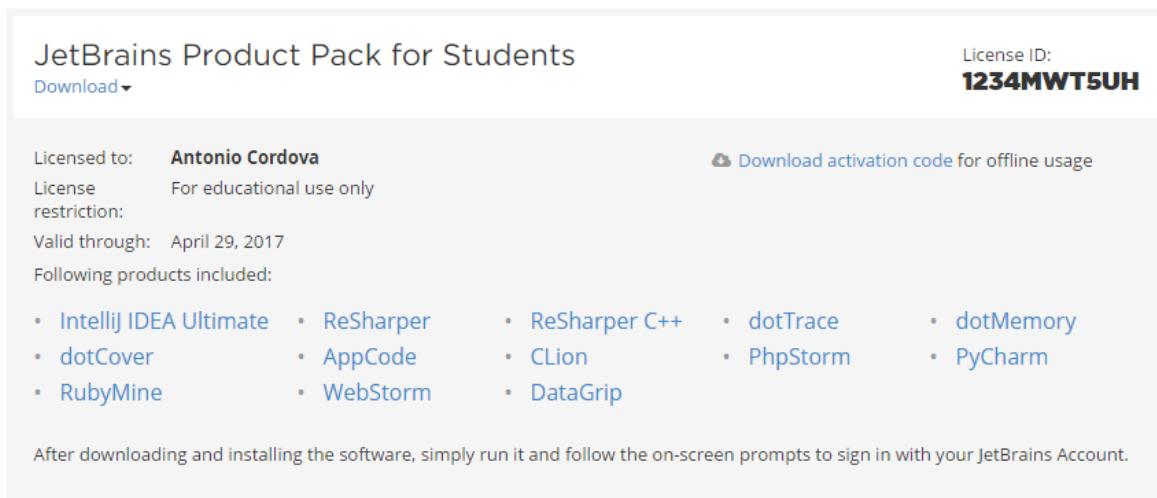


Gráfico 3-3 Paquete de producto de JetBrain a Antonio Córdova
Fuente (Antonio Córdova, 2016)

Finalmente, el modelado de la base de datos se realiza con Power Designer, posteriormente, se creará la base de datos en MySQL

3.3.2. Iteración 01 – Primer paso a producción

IT.	TAREAS	TIEMPO PLANIFICADO	PROPIETARIO
1	Presencia en internet	28:00:00	Antonio Córdova
	Diseño de Experiencia de usuario en general	8:00:00	
	Maquetación de las páginas	8:00:00	
	Creación de espacio para contacto	2:00:00	
	Implementación de las páginas en ambiente de trabajo	8:00:00	
1	Orientación de la página	8:00:00	Antonio Córdova
	Creación de espacio para ingreso de usuarios	6:00:00	
	Creación de espacio para visualización de tarjetas	1:00:00	
	Pruebas de aceptación	1:00:00	

Tabla 3-29 Iteración 01 – Primer paso a producción.
Fuente (Antonio Córdova, 2016)

El diseño enfocado en el usuario es tener en cuenta para quién y para qué está construido este sitio de modo que sea una web amigable y acorde con lo solicitado. Tal como se presenta en el grafico 3-4.

- Encontrar datos personales (empresa) para contacto
- Sistema de contacto simple a través de un formulario simple

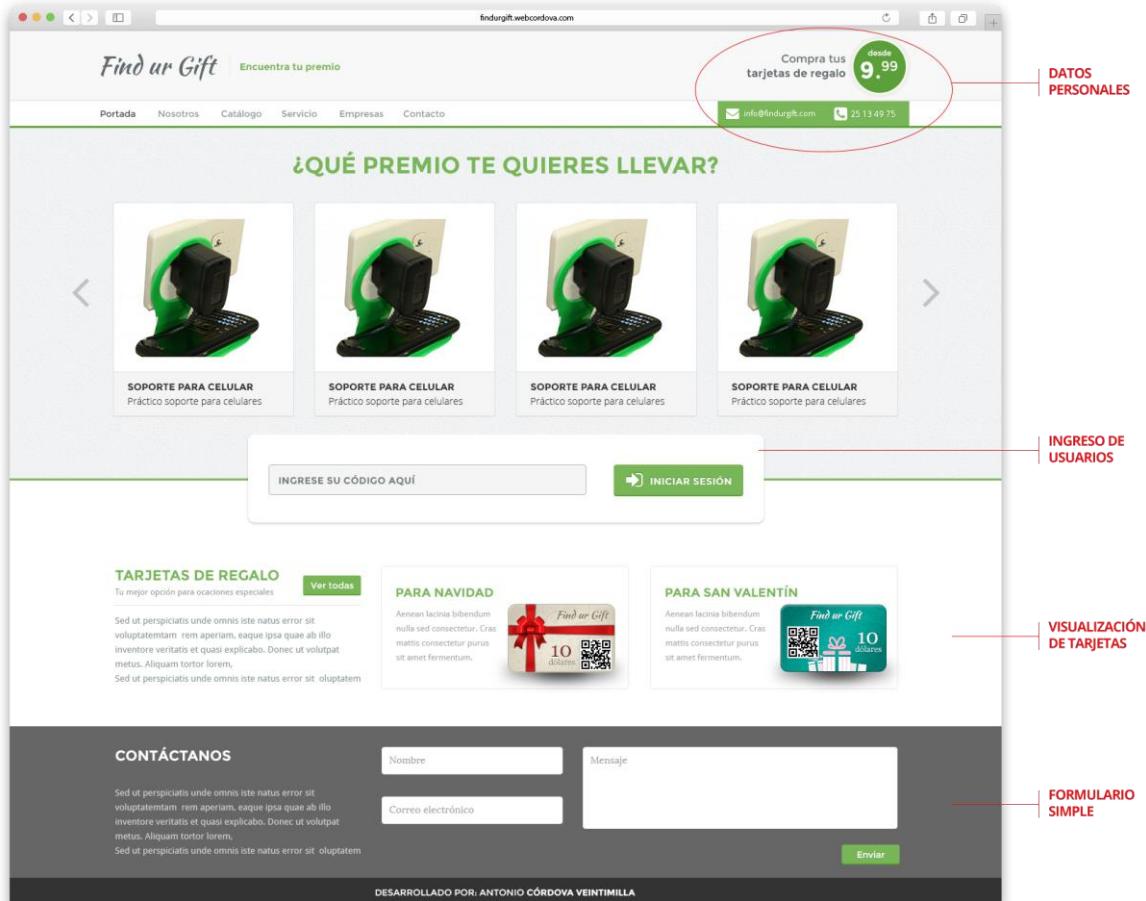


Gráfico 3-4 Diseño inicial
Fuente (Antonio Córdova, 2016)

En el grafico 3-5 se muestra el diseño web elaborado para una página interna del sitio

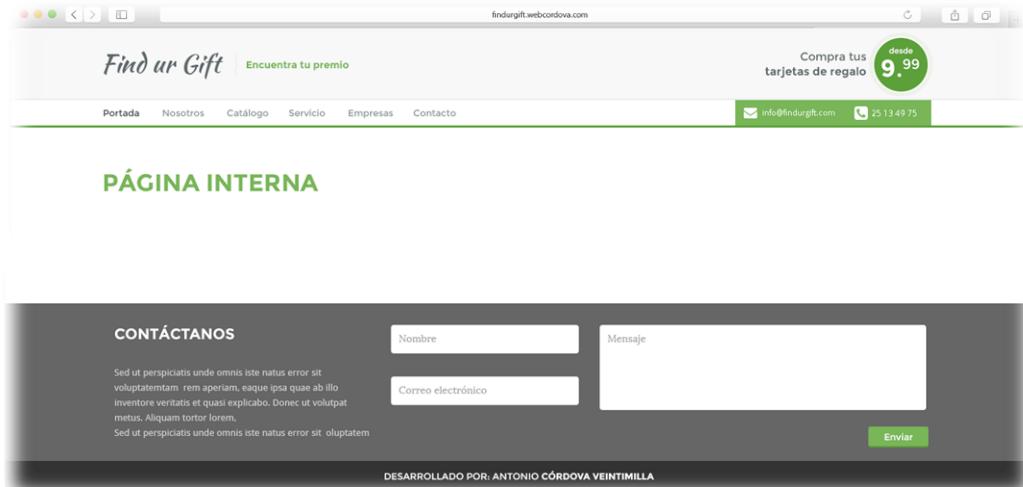


Gráfico 3-5 Diseño página interna
Fuente (Antonio Córdova, 2016)

Un arquitecto diseña maquetas de sus planos como una versión simulada a escala de lo que será su construcción, asimismo, en el diseño web se realiza una maqueta que es el paso previo a la implementación del sistema, de manera que se pueda palpar el sitio de manera interactiva, aunque no tenga una base de datos y/o controladores.

Una maqueta no significa que sea la versión final del front end, sino, una muestra de los la estructura y posición de las diferentes secciones del sitio, ayuda a mostrar un avance del sistema al cliente y detectar posibles desarrollos adicionales no mencionados en las historias de usuario.

El desarrollo de las maquetas se realiza a través del IDE PhpStorm para la programación de código HTML5, CSS3 y jQuery

```

<!DOCTYPE html>
<html lang="es">
    <head>
        <meta charset="utf-8">
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <meta name="viewport" content="width=device-width, initial-scale=1">
        <title>Find Ur Gift :: Encuentra tu premio</title>

        <link href="css/bootstrap.css" rel="stylesheet"/>
        <link href="css/fontawesome-all.min.css" rel="stylesheet" type="text/css"/>
        <link href="css/tipicons.min.css" rel="stylesheet" type="text/css"/>
        <link href="https://fonts.googleapis.com/css?family=Droid+Sans+400,700|Kaushan+Script" rel="stylesheet" type="text/css"/>
        <link href="css/site.css" rel="stylesheet" type="text/css"/>
        <link href="css/owl.carousel.css" rel="stylesheet" type="text/css"/>
    </head>
    <body>
        <div class="preloader">
            <div class="preloader_image"></div>
        </div>
        <header class="main">
            <div class="top">
                <div class="container">
                    <div class="row">
                        <div class="col-md-6">
                            <h1 id="logo">Find ur Gift</h1>
                            <strong>Encuentra tu premio</strong>
                        </div>
                        <div class="col-md-3 col-md-offset-3">
                            <div class="promo clearfix">
                                <div class="pull-right">
                                    <a href="#">Ver más</a>
                                    <strong>99</strong>
                                </div>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        </header>
        <div class="content">
            <div class="container">
                <div class="row">
                    <div class="col-md-12">
                        <div class="text">
                            <p>Encontrá tu premio con Find ur Gift, la aplicación que te permite encontrar los mejores regalos para tus seres queridos. Nuestra misión es facilitar la vida de los padres y amantes de las compras en línea. Somos una startup argentina que nació con el sueño de hacer que la experiencia de compra sea más fácil y divertida. Nuestro equipo trabaja día y noche para ofrecerte lo mejor de la web y las tiendas físicas en un solo lugar. ¡Únete a nosotros y descubre todo lo que tenemos para ofrecerte!</p>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </body>
</html>

```

Gráfico 3-6 PhpStorm y maqueta de pagina principal
Fuente (Antonio Córdova, 2016)

3.3.2.1. Pruebas de aceptación

Al término de la iteración 01 se obtuvieron los siguientes resultados durante 4 días de trabajo:

IT.	TAREAS	TIEMPO PLANIFICADO	COMIENZO	FIN
1	Presencia en internet	28:00:00	7/12/2015	10/12/2015
	Diseño de Experiencia de usuario en general	8:00:00	7/12/2015	8/12/2015
	Maquetación de las paginas	8:00:00	8/12/2015	9/12/2015
	Creación de espacio para contacto	2:00:00	9/12/2015	9/12/2015
	Implementación de las paginas en ambiente de trabajo	8:00:00	9/12/2015	10/12/2015
	Pruebas de aceptación	2:00:00	10/12/2015	10/12/2015
1	Orientación de la página	8:00:00	11/12/2015	11/12/2015
	Creación de espacio para ingreso de usuarios	6:00:00	11/12/2015	11/12/2015
	Creación de espacio para visualización de tarjetas	1:00:00	11/12/2015	11/12/2015
	Pruebas de aceptación	1:00:00	11/12/2015	11/12/2015

Tabla 3-30 Pruebas de aceptación - Iteración 01.

Fuente (Antonio Córdova, 2016)

PRUEBA ACEPTACIÓN 01	PRESENCIA EN INTERNET	CUMPLIMIENTO DE RESULTADOS ESPERADOS	APROBADO POR TRACKER	APROBADO POR CLIENTE
RESULTADOS ESPERADOS	La página debe estar habilitada y mostrar los datos de contacto	X	X	X
RESULTADO FINAL	La página web muestra de manera destacada los datos indicados	X	X	X

Tabla 3-31 Prueba de aceptación 01.

Fuente (Antonio Córdova, 2016)

PRUEBA ACEPTACIÓN 02	ORIENTACIÓN DE LA PÁGINA	CUMPLIMIENTO DE RESULTADOS ESPERADOS	APROBADO POR TRACKER	APROBADO POR CLIENTE
RESULTADOS ESPERADOS	La página debe mostrar los paquetes y mostrar un espacio para ingresar el código.	X	X	X
RESULTADO FINAL	La página web muestra de manera destacada los datos indicados	X	X	X

Tabla 3-32 Prueba de aceptación 02.

Fuente (Antonio Córdova, 2016)

3.3.3. Iteración 02 – Gestión básica de productos

IT.	TAREAS	TIEMPO PLANIFICADO	PROPIETARIO
2	Gestión básica de productos	40:00:00	Antonio Córdova
	Modelamiento de base de datos	10:00:00	
	Implementación de la Base de datos en el framework Laravel	4:00:00	
	Programación del CRUD de productos	20:00:00	
	Pruebas de aceptación	6:00:00	

Tabla 3-33 Iteración 02 - Gestión productos.

Fuente (Antonio Córdova, 2016)

Es importante diseñar y construir el modelo de la base de datos ya que se requiere de las tablas para el desarrollo de esta y futuras iteraciones. De esta forma se cumple con el cronograma establecido por el Tracker

3.3.3.1. Modelamiento de la base de datos

El diseño inicial parte del modelado conceptual de datos con sus tablas principales. Asimismo, se normalizó los campos utilizando la siguiente estandarización:

Tabla	Inicial de campo	Clave Principal
Códigos	COD	CODID
Usuarios	USR	USRID
Empresas	EMP	EMPID
Paquetes	PAQ	PAQID
Productos	PRO	PROID
Zona Geográfica	ZNA	ZNAID
Tipo de Producto	TIP	TIPID
Administradores	ADM	ADMID
Pedidos	PED	PEDID

Tabla 3-34 Estandarización de la BDD.
Fuente (Antonio Córdova, 2016)

PEDIDOS			
pedId	<pi>	Serial	<M>
pedEmplId		Integer	<M>
pedUsrId		Integer	<M>
pedCodId		Integer	<M>
pedProId		Integer	<M>
pedFhaCreacion		Timestamp	<M>
pedFhaModificacion		Timestamp	<M>
pedId	<pi>		

ADMINISTRADORES			
admId	<pi,ai>	Serial	<M>
admNombre		Variable characters (255)	<M>
admCorreo		Variable characters (100)	<M>
admClave		Variable characters (60)	<M>
ADMID2	<ai>		
admId	<pi>		

Gráfico 3-7 Tablas: pedido y administradores
Fuente (Antonio Córdova, 2016)

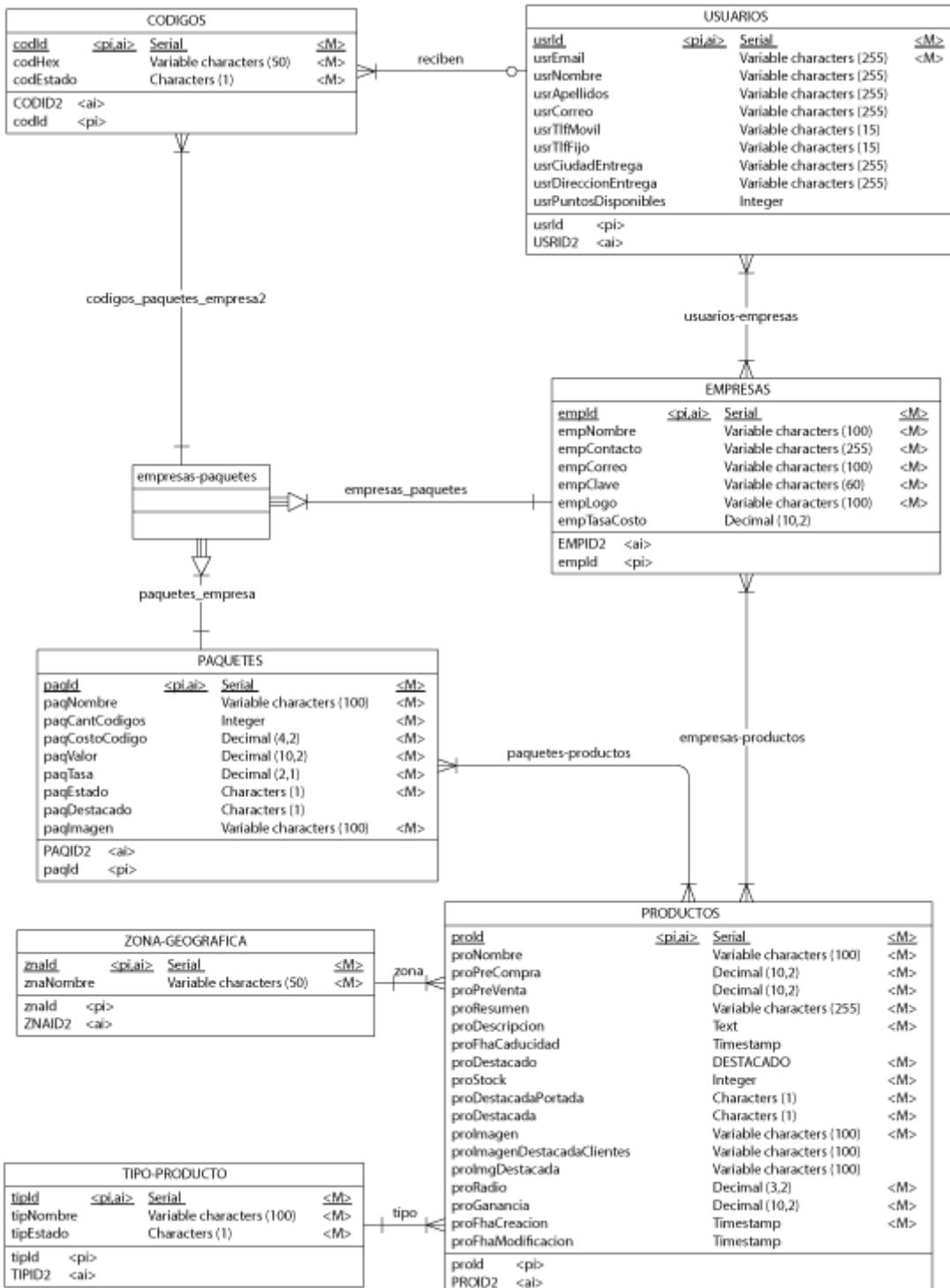


Gráfico 3-8 Modelo de la base de datos
Fuente (Antonio Córdova, 2016)

El programa genera el modelo lógico y finalmente, crea el modelo físico con el cual se implementa la base de datos.

En el modelo lógico aparecen nuevas tablas para romper la relación muchos a muchos, estas nuevas tablas son:

- Usuarios – Empresas
- Empresas – Productos
- Paquetes – Productos
- Empresas – Paquetes

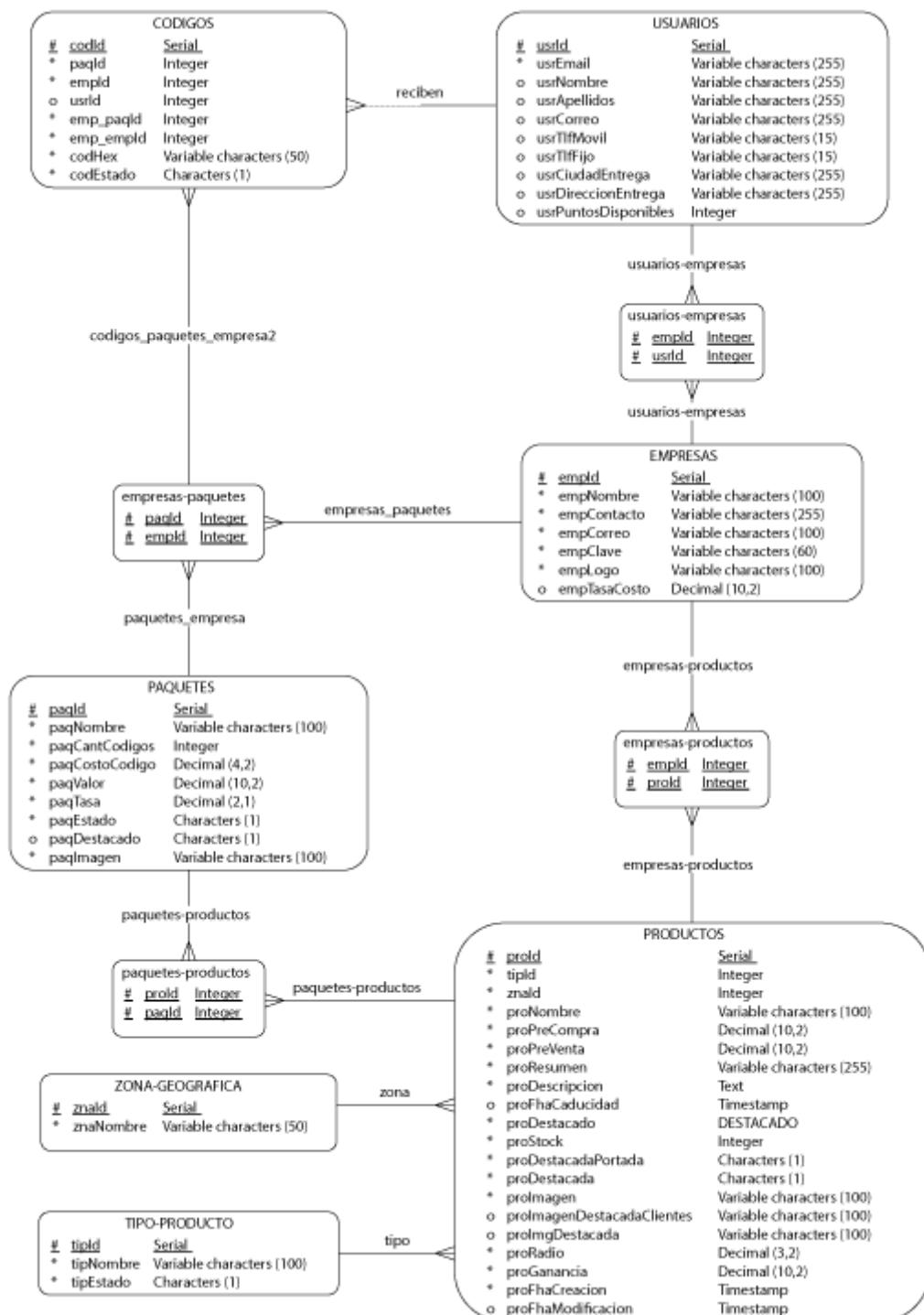


Gráfico 3-9 Modelo Lógico
Fuente (Antonio Córdova, 2016)

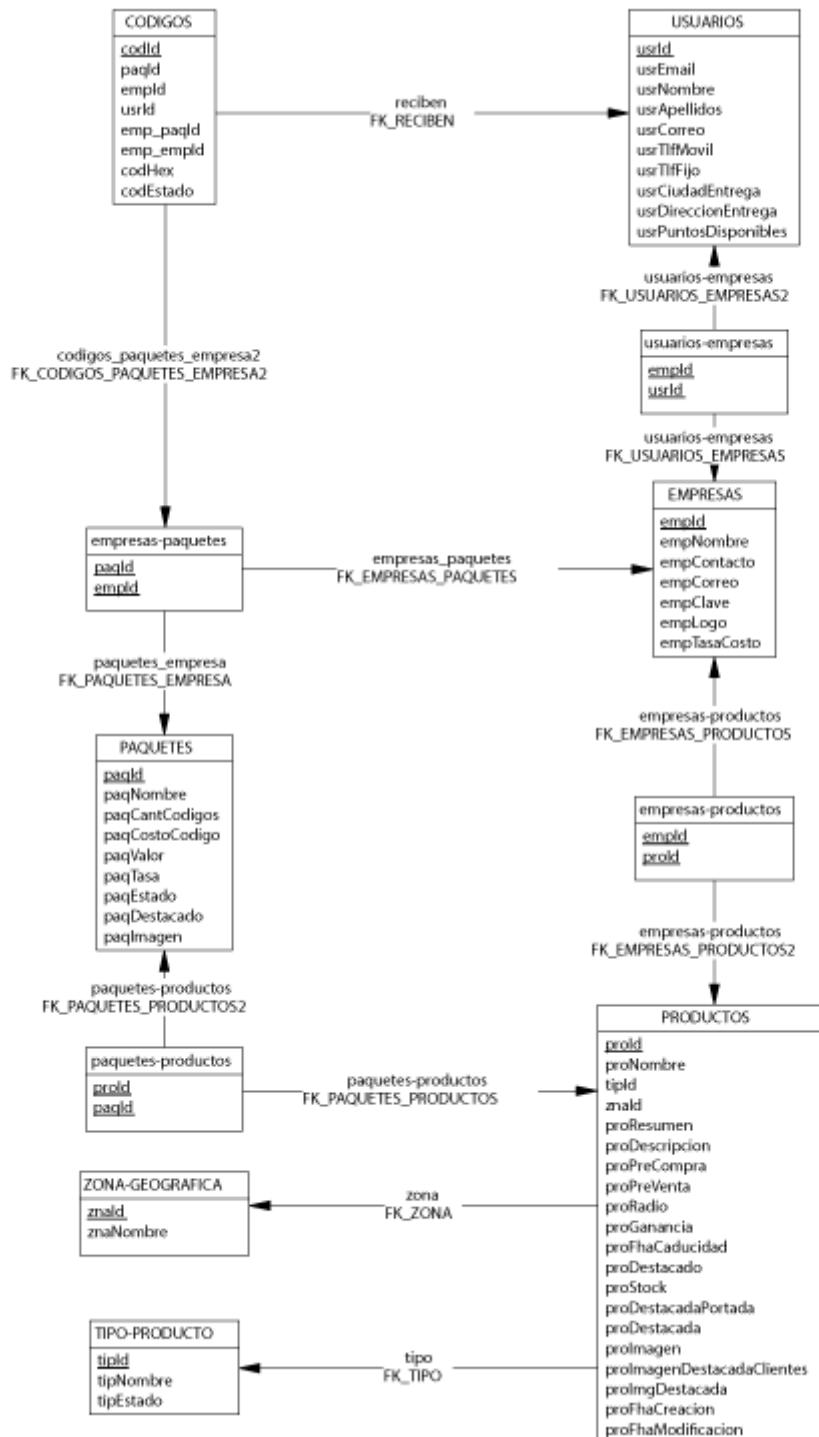


Gráfico 3-10 Modelo Físico
Fuente (Antonio Córdova, 2016)

Se genera la base de datos a partir de este modelo físico, obteniéndose el código SQL para MySQL, para ello se utiliza PhpMyAdmin de manera que se importa el script de creación. Finalmente, obtenemos el siguiente resultado:

Tabla	Acción	Filas	Tipo	Cotejamiento	Tamaño	Residuo a depurar
administradores	Examinar Estructura Buscar Insertar Vaciar Eliminar	-8	InnoDB	ucs2_general_ci	32 KB	-
codigos	Examinar Estructura Buscar Insertar Vaciar Eliminar	-56	InnoDB	ucs2_general_ci	64 KB	-
empresas	Examinar Estructura Buscar Insertar Vaciar Eliminar	-8	InnoDB	ucs2_general_ci	32 KB	-
empresas_paquetes	Examinar Estructura Buscar Insertar Vaciar Eliminar	-6	InnoDB	ucs2_general_ci	48 KB	-
empresas_productos	Examinar Estructura Buscar Insertar Vaciar Eliminar	-4	InnoDB	ucs2_general_ci	32 KB	-
paquetes	Examinar Estructura Buscar Insertar Vaciar Eliminar	-1	InnoDB	ucs2_general_ci	32 KB	-
paquetes_productos	Examinar Estructura Buscar Insertar Vaciar Eliminar	-8	InnoDB	ucs2_general_ci	32 KB	-
pedidos	Examinar Estructura Buscar Insertar Vaciar Eliminar	-4	InnoDB	ucs2_general_ci	16 KB	-
productos	Examinar Estructura Buscar Insertar Vaciar Eliminar	-3	InnoDB	ucs2_general_ci	64 KB	-
tipo_producto	Examinar Estructura Buscar Insertar Vaciar Eliminar	-5	InnoDB	ucs2_general_ci	32 KB	-
usuarios	Examinar Estructura Buscar Insertar Vaciar Eliminar	-4	InnoDB	ucs2_general_ci	32 KB	-
usuarios_empresas	Examinar Estructura Buscar Insertar Vaciar Eliminar	-4	InnoDB	ucs2_general_ci	32 KB	-
zona_geografica	Examinar Estructura Buscar Insertar Vaciar Eliminar	-4	InnoDB	ucs2_general_ci	32 KB	-

13 tablas Número de filas 79 InnoDB ucs2_general_ci 480 KB 0 B

Vista de impresión Diccionario de datos

Crear tabla

Nombre: Número de columnas:

Continuar

Gráfico 3-11 MySQL en LocalHost con la BDD creada
Fuente (Antonio Córdova, 2016)

3.3.3.2. Implementación de modelos en Laravel

En Laravel, se implementa las tablas recientemente creadas con el fin de gestionar el CRUD de todas ellas; se utiliza el siguiente comando para la creación de los modelos de las tablas:

```
php artisan make:model Nombre-de-tabla
```

Este comando genera los modelos dentro de la carpeta **app** como se visualiza en el siguiente gráfico:

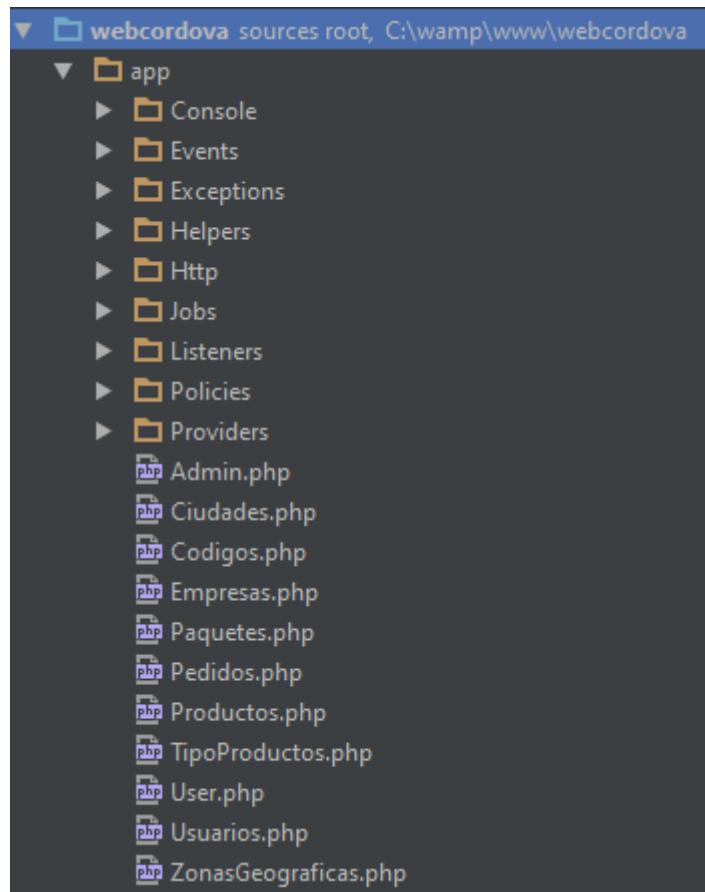


Gráfico 3-12 Creación de modelos en Laravel

Fuente (Antonio Córdova, 2016)

Se han creado los modelos de las tablas principales y no de las tablas secundarias ya que Laravel gestiona estos modelos de forma interna. Más adelante se profundizará en este tema cuando exista la relación uno a muchos o muchos a muchos.

Es necesario referirse a Laravel como un framework que utiliza un patrón de arquitectura de software multicapas, tal y como se explica en el sitio web **styde.net** a través de su autor Dulio Palacios, quien es profesor universitario y autor de este sitio:

“De hecho si bien Laravel 4 incluía las 3 famosas carpetas controllers, models, views, en Laravel 5:

Ya no encontrarás una carpeta “models”, en vez de eso tienes una carpeta app/ donde puedes estructurar tu aplicación de la forma que tenga más sentido para tu proyecto.

La carpeta Controllers es una pequeña parte de la capa “Http” que se encuentra dentro de app/. Junto a Controllers, tienes Middleware/

(middleware no se pluraliza), tienes el directorio Requests/ donde se albergan los FormRequests y tienes el archivo “routes.php”

La carpeta “views/” se encuentra ahora dentro de resources/ y forma parte de los “recursos” para presentarle los datos al usuario (assets/ lang/), creo que esta capa es la que menos cambió, por así decirlo.

En la carpeta app/ también encontrarás otras capas como Eventos, Listeners, Excepciones, Jobs, etc.” (Palacios, 2015)

Es decir, Laravel es más que un framework en 3 capas **MVC** (Modelo – vista – controlador) – “El patrón de arquitectura MVC (Modelo Vista Controlador) es un patrón que define la organización independiente del Modelo (Objetos de Negocio), la Vista (interfaz con el usuario u otro sistema) y el Controlador (controlador del workflow de la aplicación).” (Estudiantes de la Universidad Carlos III de Madrid, 2015), tal y como Palacios explicó anteriormente, pero depende de que es lo que se quiere lograr como producto final.

Por temas académicos y el fin de esta disertación de grado, se aplicará el modelo a 3 capas con un nivel de acceso mediante el sistema de enrutamiento de Laravel, para ello se utilizarán los siguientes directorios:

App

Ubicada en la raíz del sitio. En este directorio almacenaremos todos los modelos, es decir los archivos que contienen la información de las tablas de la base de datos.

Controllers

Ubicada en app > Http. > Controllers. En este directorio se guardarán todos los controladores o la lógica del negocio de esta presente disertación de grado.

Views

Ubicada en resources > views. En este directorio estarán las vistas y las plantillas que son necesarias para que el usuario visualice toda la información.

Public

Ubicada en la raíz del sitio. Los archivos de esta carpeta son de acceso público y se guardan los archivos creados previamente en la maquetación tales como CSS, JS, fuentes e imágenes.

3.3.3.3. Programación CRUD²⁹ de productos

Modelo

Productos.php ya fue creado anteriormente y es el modelo donde asocia con la tabla productos de base de datos.

En este archivo encontramos las relaciones entre tablas:

- Productos – Tipo de productos
- Productos – Zona geográfica

Esta relación es de uno a muchos por lo que no es necesario crear una nueva tabla o nuevo modelo.

```

<?php
namespace App;
use ...
class Productos extends Model
{
    /**
     * The table associated with the model.
     *
     * @var string
     */
    protected $table = 'productos';

    /**
     * The Primary key for this model
     *
     * @var string
     */
    protected $primaryKey = 'PROID';

    /**
     * Paquetes que pertenecen a Productos
     */
    public function paquetes()
    {
        return $this->belongsToMany('App\Paquetes', 'paquetes_productos', 'PROID', 'PAQID');
    }

    /**
     * Empresas que pertenecen a Productos
     */
    public function empresas()
    {
        return $this->belongsToMany('App\Empresas', 'empresas_productos', 'PROID', 'EMPID');
    }

    /**
     * Obtención del tipo de producto que contiene productos
     */
}

```

Gráfico 3-13 Modelo: *Productos.php*
Fuente (Antonio Córdova, 2016)

Controlador

Para crear o modificar se aplica el principio de reutilización de código de XP, es decir se utiliza el mismo código para ambos casos:

Para leer los datos del modelo, se utilizan varios métodos, el principal es el que muestra todos los productos, haciendo una llamada a la base para obtener la lista completa paginada en 50 ítems, es decir se mostrará únicamente los 50 primeros ítems y mediante un paginador irá mostrando los siguientes ítems.

²⁹ El CRUD (acrónimo en inglés de Create, Read, Update, Delete) son operaciones básicas de manipulación de datos para la base de datos (Rochkind, 2013)

Eliminar Producto

Se requiere de un identificador de producto para conocer que producto se desea eliminar por lo que se envía como parámetro la variable **\$id** e internamente el método busca el producto y lo destruye, no sin antes validar la existencia del mismo.

```

webcordova - [C:\wamp\www\webcordova] - ...app\Http\Controllers\Admin\ProductsController.php - PhpStorm 2016.1
Edit View Navigate Code Refactor Run Tools VCS Window Help
File webcordova app Http Controllers Admin ProductsController.php
Project webcordova sources root: C:\wamp\www\webcordova
  app
    Console
    Events
    Exceptions
    Helpers
    Http
      Controllers
        Admin
          AuthAdminController.php
          EmpresaController.php
          EmpresaPaquetesController.php
          MobileEmpresaController.php
          PaquetesController.php
          ProductsController.php
        Auth
        Cliente
        Empresas
        Controller.php
        Libraries
        Middleware
        Requests
        Kernel.php
        routes.php
      Jobs
      Listeners
      Policies
      Providers
        Admin.php
        Ciudades.php
        Codigos.php
        Empresas.php
        Paquetes.php
        Pedidos.php
        Products.php
        TipoProductos.php
  App\Http\Controllers\Admin\ProductsController.php
  App\Http\Controllers\Admin\EmpresasController.php
  empresas\index.blade.php
  empresas\edit.blade.php
  empresas\add.blade.php
  Products.php
  Admin\ProductsController.php
  prueba.html
  Remote Host Database
  Event Log
  28 chars 125:17 LF+ UTF-8+
  Favorites Terminal
  
```

Gráfico 3-14 Controlador: ProductsController.php
Fuente (Antonio Córdova, 2016)

Vista

Para la vista se utiliza la maqueta inicial en donde se realiza las respectivas modificaciones y dar el paso de HTML a PHP. Adicionalmente, se requiere de la creación inicial de una plantilla para que funcione de manera genérica en cualquier vista nueva.

Debido a que el manejo de datos de productos es exclusivo del administrador, se ha creado la plantilla **Admin**, la cual contiene la estructura básica (cabecera, menú principal y pie de página), y de un espacio para el contenido específico:

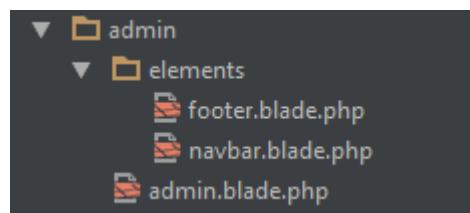


Gráfico 3-15 Plantilla Admin
Fuente (Antonio Córdova, 2016)

Esta plantilla está basada en la maqueta realizada anteriormente, con varias modificaciones:

- **Llamadas a archivos externos como CSS o JS**

En la maqueta tenemos la siguiente línea de ejemplo:

```
<link href="css/bootstrap.css" rel="stylesheet">
```

En la cual se llama a un archivo externo llamado bootstrap.css. En la plantilla se reemplaza el código por el siguiente

```
<link href="{{ asset('css/bootstrap.css') }}" rel="stylesheet">
```

Este código hace referencia a la ubicación del archivo de manera relativa al proyecto.

- **Inclusión de espacio para contenido específico**

Todo el contenido de ejemplo creado en la maqueta es reemplazado por:

```
@yield('content')
```

- **Condiciones de muestreo de hojas de estilo o scripts adicionales**

```
@if(isset($styles))
    @foreach($styles as $style)
        <link rel="stylesheet" href="{{ asset('/'.$style) }}"/>
    @endforeach
@endif
```

Este código incluye hojas de estilo adicionales si se envía desde el controlador.

Finalmente, para llamar a esta plantilla en cualquier vista se lo realiza de la siguiente manera:

```
@extends('templates.admin.admin')
@section('title', 'Lista de empresas')
@section('content')
CONTENIDO ESPECIFICO
@endsection
```

La primera línea llama a la plantilla mediante su ubicación templates > admin > admin. La segunda línea indica el inicio del contenido específico, asimismo, la última línea indica el fin de este contenido, todo lo que está entre estas dos líneas se mostrará al usuario.

Mostrar contenido

Para mostrar el contenido, particularmente la lista de productos, se utiliza de igual manera la maqueta con los cambios respectivos, en donde se encuentran 2 partes: si existe productos o no productos.

Si existen productos, los muestra en una tabla con todos sus datos traídos desde la base; por otro lado, si no existen productos, muestra un mensaje al usuario que no existen productos.

Adicionalmente, se implementa una línea para paginar los productos, es decir se mostrará 50 productos y el paginador automáticamente creará botones para la cantidad de productos creados.

Crear o editar un producto

La creación y edición de un producto es un proceso similar ya que se muestra los mismos campos, la diferencia se encuentra en la edición del producto a través de un método previo que busca el producto a editar para obtener sus datos de tal forma que se los pueda modificar.

3.3.3.4. Resultados

LISTA DE PRODUCTOS									
Nombre	Imagen	Tipo	Zona	Compra Venta	Ganancia/Radio	Fecha caducidad	Destacado Portada	Stock	Opciones
Samsung S6 Edge		Tecnología	nacional	850.00 980.00	130.00 1.15	no aplica		5	editar eliminar detalles
CONCHA Y TORO TRIO MERLOT		Licores	nacional	17.75 30.00	12.25 1.69	no aplica		5	editar eliminar detalles
Audifonos RCA		Tecnología	nacional	4.95 7.10	2.15 1.43	no aplica		10	editar eliminar detalles
Paquete BIC		Utiles Escolares	nacional	4.25 6.50	2.25 1.53	no aplica		19	editar eliminar detalles

Gráfico 3-16 Lista de productos
Fuente (Antonio Córdova, 2016)

En el gráfico 3-16 se muestra la lista completa de productos creados en el sistema satisfactoriamente, en la cual se crearon 4 productos de prueba con su respectiva información, adicionalmente cada producto tiene sus opciones del CRUD: Editar, Eliminar, Ver detalles.

El gráfico 3-17 muestra la pantalla para agregar un nuevo producto.

AGREGAR NUEVO PRODUCTO

*Campos obligatorios

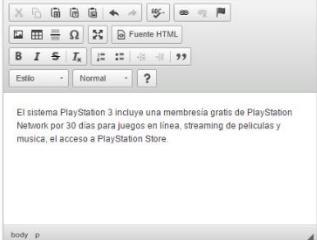
Nombre*:	Play Station 3	
Tipo*:	Tecnología	Agregar Tipo
Zona geográfica*:	Quito	Agregar zona
Resumen*:	PlayStation 3 Super Slim de 500GB de capacidad para almacenar videos, fotos y juegos.	
Descripción*:	 <p>El sistema PlayStation 3 incluye una membresía gratis de PlayStation Network por 30 días para juegos en línea, streaming de películas y música, el acceso a PlayStation Store</p>	
Precio de compra*:	380	
Precio de venta*:	450	
Fecha de caducidad:	2016-11-06	
Stock:	2	
<input checked="" type="checkbox"/> ¿Producto en portada?		
<input type="button" value="Guardar"/> <input type="button" value="Cancelar"/>		

IMAGEN GENÉRICA*



Seleccionar Imagen de producto

IMAGEN DESTACADA

Gráfico 3-17 Agregar producto – datos
Fuente (Antonio Córdova, 2016)

Una vez guardada la información ingresada en la página anterior, el sistema muestra un mensaje de éxito en el proceso: **Se agregó el producto Play Station 3 correctamente**

Find Ur Gift :: Lista de prc

Antonio

findurgift.webcordova.com/admin/productos

Productos Empresas Paquetes Reportes Pedidos Salir Find Ur Gift

LISTA DE PRODUCTOS

Agregar producto Reporte

Se agregó el producto Play Station 3 correctamente

Nombre	Imagen	Tipo	Zona	Compra Venta	Ganancia/Radio	Fecha caducidad	Destacado	Portada	Stock	Opciones
Samsung S6 Edge		Tecnología	nacional	850.00 980.00	130.00 1.15	no aplica			5	editar eliminar detalles
CONCHA Y TORO TRIO MERLOT		Licores	nacional	17.75 30.00	12.25 1.69	no aplica			5	editar eliminar detalles
Audifonos RCA		Tecnología	nacional	4.95 7.10	2.15 1.43	no aplica			10	editar eliminar detalles
Paquete BIC		Utiles Escolares	nacional	4.25 6.50	2.25 1.53	no aplica			19	editar eliminar detalles
Play Station 3		Tecnología	Quito	380.00 450.00	70.00 1.18	2016-11-06 00:00:00			2	editar eliminar detalles

DESARROLLADO POR: ANTONIO CÓRDOWA

Gráfico 3-18 Éxito en agregar producto.
Fuente (Antonio Córdova, 2016)

Ahora se edita el producto ingresado anteriormente haciendo clic en editar:

The screenshot shows the product editing form for a PlayStation 3. The fields filled in are:

- Nombre***: Play Station 3
- Tipo***: Tecnología
- Zona geográfica***: Quito
- Resumen***: PlayStation 3 Super Slim de 500Gb de capacidad para almacenar videos, fotos y juegos.
- Descripción***: (Contains rich text editor content about PlayStation Network membership.)
- Precio de compra***: 380.00
- Precio de venta***: 450.00
- Fecha de caducidad**: (Empty field)
- Stock**: 2
- ¿Producto en portada?**:

On the right side, there are sections for "IMAGEN GENÉRICA*" and "IMAGEN DESTACADA". The "IMAGEN DESTACADA" section displays a thumbnail of a PlayStation 3 console with the text "PlayStation 3" and "PS3" visible. Below it are buttons for "Seleccionar imagen destacada para clientes" and "Subir Imagen".

At the bottom of the form are "Guardar" and "Cancelar" buttons.

Gráfico 3-19 Editar producto
Fuente (Antonio Córdova, 2016)

Se agrega una imagen destacada para que se muestre en portada del usuario. A continuación, se guarda la información y el sistema muestra el mensaje: **Se actualizó el producto Play Station 3 correctamente**

Finalmente hacemos clic en borrar cualquier producto para probar su funcionalidad, el sistema nos muestra una ventana emergente para aprobar el borrado del producto, y por último el mensaje de eliminación exitosa:

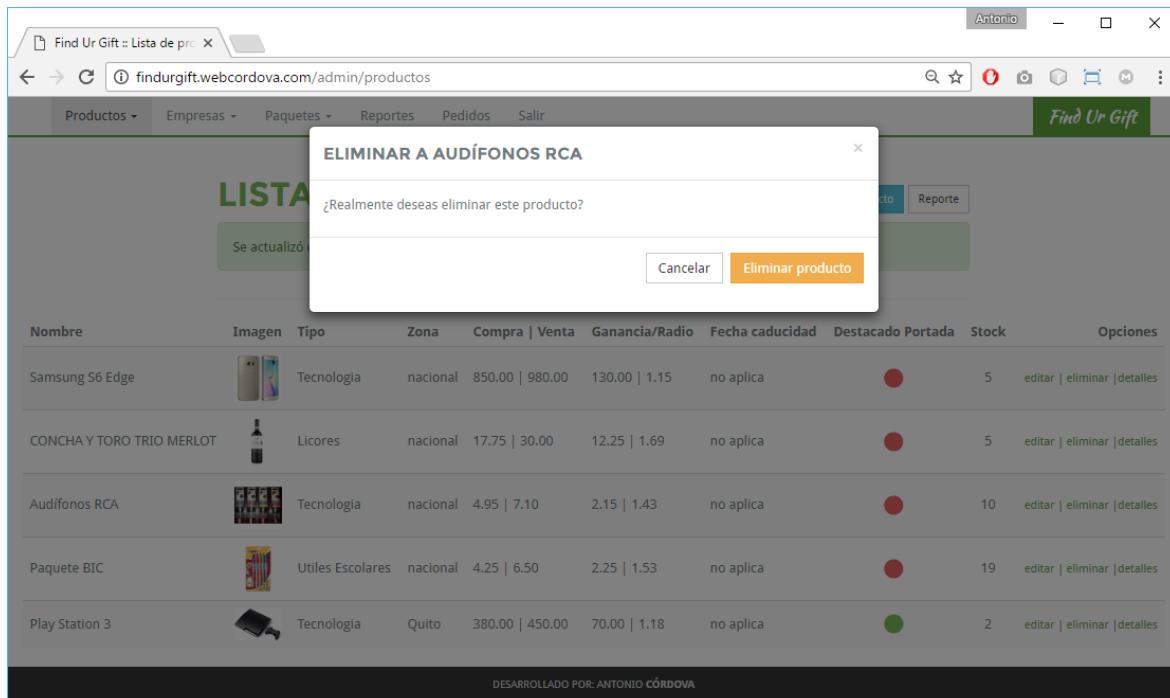


Gráfico 3-20 Mensaje previo a borrar producto
Fuente (Antonio Córdova, 2016)

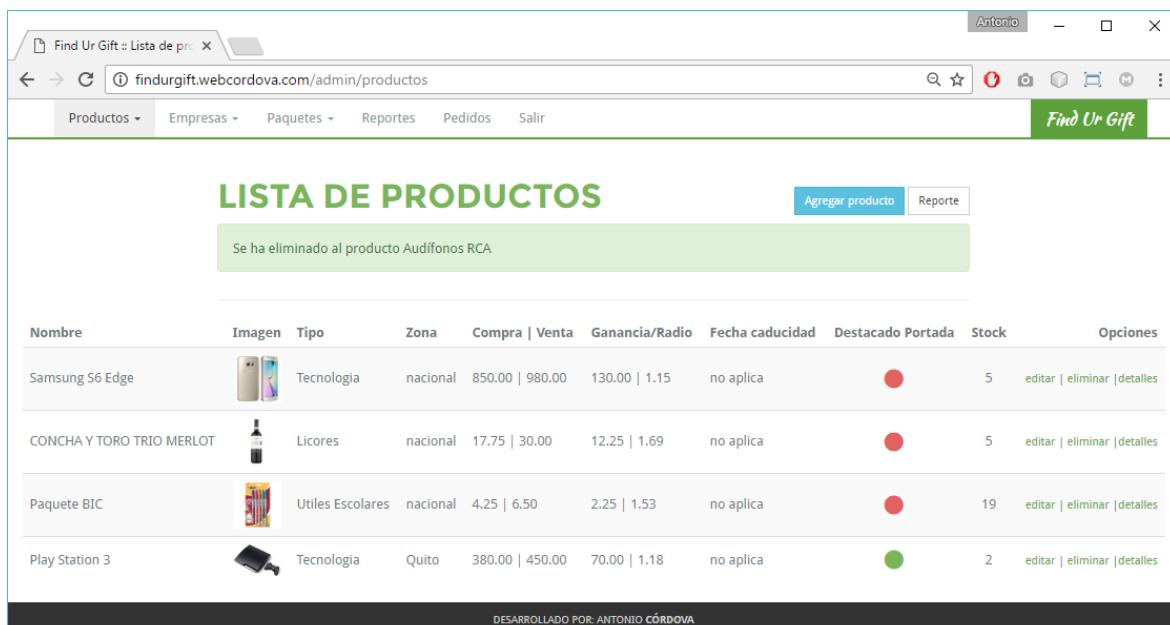


Gráfico 3-21 Mensaje de eliminación exitoso
Fuente (Antonio Córdova, 2016)

3.3.3.5. Pruebas de aceptación

Al término de la iteración 02 se obtuvieron los siguientes resultados durante 5 días de trabajo:

IT.	TAREAS	TIEMPO PLANIFICADO	COMIENZO	FIN
2	Gestión básica de productos	40:00:00	14/12/2015	18/12/2015
	Modelamiento de base de datos	10:00:00	14/12/2015	15/12/2015
	Implementación de la Base de datos en el framework Laravel	4:00:00	15/12/2015	15/12/2015
	Programación del CRUD de productos	20:00:00	15/12/2015	18/12/2015
	Pruebas de aceptación	6:00:00	18/12/2015	18/12/2015
	Gestión básica de productos	40:00:00	14/12/2015	18/12/2015

Tabla 3-35 Pruebas de aceptación - iteración 02. Fuente (Antonio Córdova, 2016)

PRUEBA ACEPTACIÓN 03	CATÁLOGO DE PRODUCTOS	CUMPLIMIENTO DE RESULTADOS ESPERADOS	APROBADO POR TRACKER	APROBADO POR CLIENTE
RESULTADOS ESPERADOS	El sistema debe mostrar los productos disponibles para el usuario.	X	X	X

Tabla 3-36 Prueba de aceptación 03.

Fuente (Antonio Córdova, 2016)

3.3.4. Iteración 03 – Gestión de usuarios

En esta iteración se desarrollará la nueva plantilla para los usuarios registrados o quienes tengan el código de acceso al sistema para canje de productos.

Además, se realizará el CRUD de usuarios (crear, modificar, eliminar o ver usuarios)

El ingreso al sistema a través de códigos QR mediante un dispositivo móvil o a través del código en la página web.

IT.	TAREAS	TIEMPO PLANIFICADO	PROPIETARIO
3	Gestión de usuarios	40:00:00	Antonio Córdova
	Diseño de Experiencia de usuario para clientes	8:00:00	
	Programación CRUD de usuarios	13:00:00	
	Implementación de ingreso al sistema a través de códigos QR, código	13:00:00	
	Pruebas de aceptación	6:00:00	

Tabla 3-37 Iteración 03 - Gestión usuarios.

Fuente (Antonio Córdova, 2016)

3.3.4.1. Diseño de experiencia de usuario para clientes

La historia de usuario 08 solicita que los productos se muestren a los usuarios para que puedan seleccionarlos y realizar el proceso de canje de producto.

Para iniciar sesión, el cliente debe tener un código de acceso puede ser QR para dispositivos móviles o Hexadecimal para acceso a través de un navegador web.

El grafico 3-20, muestra la pantalla inicial después del proceso de inicio de sesión.

The image displays a user interface mockup for a gift catalog application, likely built using the XP methodology. The interface is organized into several main sections:

- Header:** Features the PRONACA logo, navigation links for 'Lista de premios', 'Datos de usuario', 'Carrito', 'Salir', and email contact information ('info@findugift.com'), and a timestamp ('25/11/49 75').
- Productos Destacados:** Shows a large image of a Samsung Galaxy S6 Edge smartphone with a gold case. Below it are three categories of gifts: 'Regalos para ELLA' (pink gift bag), 'Regalos para EL' (black sunglasses), and 'Regalos para NIÑOS' (red toy truck). Each category includes a placeholder text block and a 'ver ahora' button.
- CATEGORÍAS DE PRODUCTOS:** A horizontal list of product categories: 'NUEVOS PRODUCTOS', followed by arrows pointing left and right, and then 'PRODUCTOS AGREGADOS RECENTEMENTE'.
- Productos Recientemente Agregados:** Displays four recent additions: four white Sony batteries labeled '654 Nombre Producto'. Each item has an 'Agregar al carrito' button and a 'detalles' link.
- Visualización de Productos:** Shows two rows of eight Sony batteries each, all labeled '654 Nombre Producto'. Each item has an 'Agregar al carrito' button and a 'detalles' link.
- Contact Us:** A footer section titled 'CONTÁCTANOS' with a form for entering Name, Last Name, and Email, along with a 'Mensaje' text area and an 'Enviar' button.

Gráfico 3-22 maqueta portada de usuarios
Fuente (Antonio Córdova, 2016)

La nueva plantilla generada para usuarios registrados se ubica en el directorio resources > views > templates > usuario, esta plantilla tiene la misma estructura que administración, se agrega un nuevo archivo php para el proceso de canje

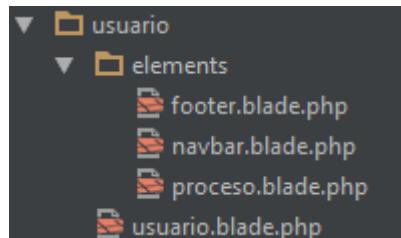


Gráfico 3-23 Plantilla para usuario
Fuente (Antonio Córdova, 2016)

El usuario debe iniciar sesión a través de los dos métodos disponibles: código QR o ingresando el código en la página web para poder visualizar el contenido.

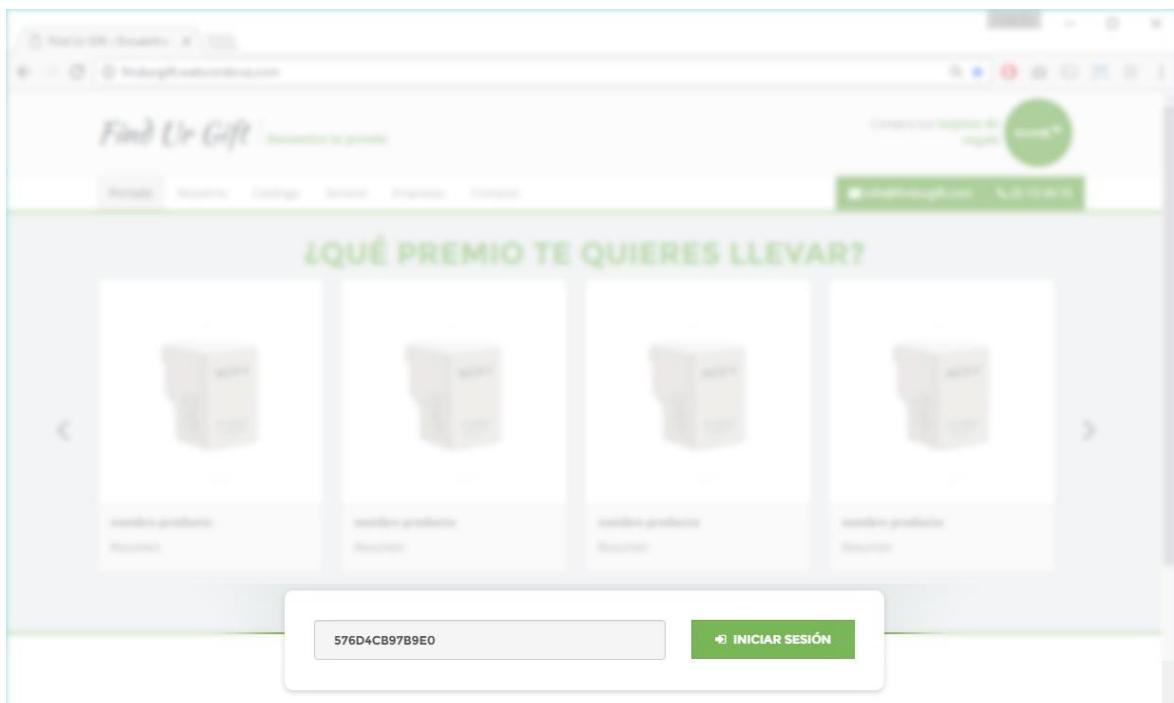


Gráfico 3-24 Ingreso de código
Fuente (Antonio Córdova, 2016)

3.3.4.2. Programación CRUD de usuarios

Modelo

La clase PHP llamada Usuarios es aquella que se contiene información directa sobre la tabla usuarios y sus respectivos campos. Las relaciones uno a muchos entre esta tabla son códigos y empresas.

```

webcordova - [C:\wamp\www\webcordova]\app\Usuarios.php - PhpStorm 2016.1
File Edit View Navigate Code Refactor Run Tools VCS Window Help
webcordova \ app \ Usuarios.php
Project 1 Project
  - EmpresasPaquetesController.php
  - MobileEmpresasController.php
  - PaquetesController.php
  - ProductosController.php
  - Auth
  - Cliente
  - Empresas
    - Controller.php
  - Librarias
  - Middleware
  - Requests
  - Kernel.php
  - routes.php
  - Jobs
  - Listeners
  - Policies
  - Providers
  - Admin.php
  - Ciudades.php
  - Codigos.php
  - Empresas.php
  - Paquetes.php
  - Pedidos.php
  - Productos.php
  - TipoProductos.php
  - User.php
  - Usuarios.php
  - ZonasGeograficas.php
  - bootstrap
  - config
  - database
  - public
  - resources
    - assets
    - lang
  - views
  Favorites
  Guia
  Event Log
  Terminal
  App\Usuarios | seleccionarPorEmpresaPaquete
  1
  2
  3
  4
  5
  6
  7
  8
  9
  10
  11
  12
  13
  14
  15
  16
  17
  18
  19
  20
  21
  22
  23
  24
  25
  26
  27
  28
  29
  30
  31
  32
  33
  34
  35
  36
  37
  38
  39
  40
  41
  42
  43
  44
  45
  protected $table = 'usuarios';
  /**
   * The Primary key for this model
   *
   * @var string
   */
  protected $primaryKey = 'USRID';

  /**
   * The attributes excluded from the model's JSON form.
   *
   * @var array
   */
  protected $hidden = ['password', 'remember_token'];

  /**
   * Obtención de códigos para el usuario
   */
  public function codigos(){
    return $this->hasMany('App\Codigos', 'USRID');
  }

  /**
   * Empresas que pertenecen a usuarios
   */
  public function empresas(){
  }
  
```

Gráfico 3-25 Modelo: Usuarios.php
Fuente (Antonio Córdova, 2016)

Controlador

Existen dos controladores generados para esta iteración: AuthCliente.php y DatosController.php, los cuales están ubicados en Controllers > Cliente

En el gráfico 3-24 se muestra el archivo AuthCliente.php, el cual controla el acceso al sistema, en otras palabras, verifica que el código ingresado sea el correcto. Si el código ingresado es el correcto, entonces se crea un nuevo cliente con los puntos disponibles del usuario, provenientes de la tarjeta creada, el resto de datos será vacío para que el usuario sea quien llene estos campos.

Cuando el usuario utiliza el método de acceso a través del código QR, hace una llamada al método ***doLoginQR***.

Básicamente, es igual al anterior método de ingreso, la diferencia se encuentra en la forma de su llamado a través del método GET debido a que la información enviada es una ruta URL para ingresar al sistema sin ningún código ni contraseña, es decir, el usuario solo deberá escanear el código y con ello habrá ingresado al sistema

Para salir del sistema se utiliza el método para eliminar la variable de sesión ***usuario_login_id***

```

webcordova - [C:\wamp\www\webcordova] - ...app\Http\Controllers\Cliente\AuthCliente.php - PhpStorm 2016.1
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Project webcordova sources root, C:\wamp\www\webcordova
  app
    Console
    Events
    Exceptions
    Helpers
    Http
      Controllers
        Admin
        Auth
        Cliente
          AuthCliente.php
          CarritoController.php
          DatosController.php
          ProductosController.php
        Controller.php
        Empresas
        Libraries
        Middleware
        Requests
        Kernel.php
        routes.php
      Jobs
      Listeners
      Policies
      Providers
        Admin.php
        Ciudades.php
        Codigos.php
        Empresas.php
        Paquetes.php
        Pedidos.php
        Productos.php
        TipoProductos.php
        User.php
        Usuarios.php
  Favorites  Gulp
  Event Log Terminal
AuthCliente.php
1 <?php
2
3 namespace App\Http\Controllers\Cliente;
4
5 use ...
6
7 class AuthCliente extends Controller
8 {
9     /**
10      * Session mixed
11      * Método para iniciar sesión
12      */
13
14     public function doLogin()
15     {
16         $hexCodigo = Input::get('codigo');
17         $codigo = Codigos::query()->where('CODHEX', '=', $hexCodigo)->first();
18         if (!$codigo) {
19             return Redirect::back()->withSession(['error_auth_msg' , 'código incorrecto']);
20         }
21         if ($codigo->ESTADO == 0) {
22             return Redirect::back()->withSession(['error_auth_msg' , 'Ya no tiene más puntos para participar']);
23         }
24
25         if ($codigo->USRID == null) {
26             $punte = Punte::find($codigo->PAQID);
27             $pptaDisponible = $punte->PAQDISPONIBLE;
28             $usuariocreado = new Usuario();
29             $usuariocreado->USRPUNTOSDISPONIBLES = $pptaDisponible;
30             $usuariocreado->save();
31             $codigo->USRID = $usuariocreado->USRID;
32             $codigo->save();
33             $empresa = Empresas::find($codigo->EMPID);
34             $usuario = $empresa->empresas()->save($empresa);
35             $usuario->save();
36             Session::put('usuario_login_id' , $codigo->USRID);
37             return Redirect::to('/usuarios/productos');
38         }
39
40         Session::put('usuario_login_id' , $codigo->USRID);
41         return Redirect::to('/usuarios/productos');
42     }
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
99

```

Gráfico 3-26 Controlador: AuthCliente.php
Fuente (Antonio Córdova, 2016)

En el gráfico 3-25 se muestra el archivo DatosController.php está enfocado a la edición del usuario, naturalmente estos datos no se podrán borrar para futuros controles o fiscalizaciones.

```

webcordova - [C:\wamp\www\webcordova] - ...app\Http\Controllers\Cliente\DatosController.php - PhpStorm 2016.1
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Project webcordova sources root, C:\wamp\www\webcordova
  app
    Console
    Events
    Exceptions
    Helpers
    Http
      Controllers
        Admin
        Auth
        Cliente
          AuthCliente.php
          CarritoController.php
          DatosController.php
          ProductosController.php
        Controller.php
        Empresas
        Libraries
        Middleware
        Requests
        Kernel.php
        routes.php
      Jobs
      Listeners
      Policies
      Providers
        Admin.php
        Ciudades.php
        Codigos.php
        Empresas.php
        Paquetes.php
        Pedidos.php
        Productos.php
        TipoProductos.php
        User.php
        Usuarios.php
  Favorites  Gulp
  Event Log Terminal
DatosController.php
1 <?php
2
3 namespace App\Http\Controllers\Cliente;
4
5 use ...
6
7 class DatosController extends Controller
8 {
9     public function __construct()
10     {
11         if (!Session::has('usuario_login_id')) {
12             Redirect::to('/')->send();
13         }
14     }
15
16     private $rules = array(
17         'USRNOMBRE' => 'required',
18         'USRAPELLIDOS' => 'required',
19         'USRCORREO' => 'required|email',
20         'USRTELEFONIL' => 'required',
21         'USRTELEFONO' => 'required',
22         'USRCEDADENTRSGA' => 'required',
23         'USRDIRRECICLACIONTRSGA' => 'required'
24     );
25
26     private $rulesMessages = array(
27         'required' => 'campo obligatorio',
28         'email' => 'campo debe ser correo electrónico válido',
29     );
30
31     public function datos()
32     {
33         $usuario = Usuarios::find(Session::get('usuario_login_id'));
34         $empresa = Empresas::whereHas('usuarios', function ($query) {
35             $query->where('usuarios.USRID', Session::get('usuario_login_id'));
36         })->first();
37
38         if (Input::exists()) {
39             $usuario = Input::all();
40         }
41     }
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
59
60
61
62
63
64
65
66
67
68
69
69
70
71
72
73
74
75
76
77
78
79
79
80
81
82
83
84
85
86
87
88
89
89
90
91
92
93
94
95
96
97
98
99
99

```

Gráfico 3-27 Controlador: DatosController.php
Fuente (Antonio Córdova, 2016)

Vista

Se crea un formulario para el ingreso de datos, utilizando la plantilla usuario.

3.3.4.3. Resultados

Cuando el cliente inicia sesión, aparece los productos seleccionados y se crea automáticamente el usuario

BIENVENIDO! YA PUEDES ESCOGER TUS PREMIOS

Tienes disponibles 10 puntos para canjearlos cuando quieras

ESCOJE TUS PREMIOS

Paquete BIC
10 pts.
Paquete BIC

● Agregar al carrito ver más

CONTÁCTANOS

Sed ut perspiciatis unde omnis iste natus error sit voluptatem rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi explicabo. Donec ut voluptatum metus. Aliquam tortor lorem, Sed ut perspiciatis unde omnis iste natus error sit voluptatem

Nombre _____ Mensaje _____

Apellidos _____

Correo Electrónico _____

Enviar

DESARROLLADO POR ANTONIO CÓRDOVA

Gráfico 3-28 Portada de usuario
Fuente (Antonio Córdova, 2016)

El usuario puede hacer clic en Datos de usuario para actualizar su información en donde podrá modificar sus datos.

DATOS DE USUARIO

Ingrésa datos personales para el envío

Nombres:

Apellidos:

Email:

Teléfono Fijo:

Teléfono Móvil:

Dirección de entrega:

Ciudad de entrega:

CONTÁCTANOS

Sed ut perspiciatis unde omnis iste natus error sit voluptatem rem apriam, eaque ipsa quae ab illo inventore veritatis et quasi explicabo. Donec ut voluptat metus. Aliquam tortor lorem, Sed ut perspiciatis unde omnis iste natus error sit voluptatem

Nombre

Apellidos

Correo Electrónico

Mensaje

Enviar

Gráfico 3-29 Actualización de datos de cliente
Fuente (Antonio Córdova, 2016)

Cuando se almacena los datos, aparece el mensaje de éxito.

DATOS DE USUARIO

Ingrésa datos personales para el envío

Se actualizaron tus datos satisfactoriamente

Nombres:

Apellidos:

Email:

Teléfono Fijo:

Teléfono Móvil:

Dirección de entrega:

Ciudad de entrega:

Guarda Regresar a productos

Gráfico 3-30 Datos almacenados exitosamente
Fuente (Antonio Córdova, 2016)



Gráfico 3-31 Ingreso mediante código QR
Fuente (Antonio Córdova, 2016)

3.3.4.4. Pruebas de aceptación

Al término de la iteración 03 se obtuvieron los siguientes resultados durante 5 días de trabajo:

IT.	TAREAS	TIEMPO PLANIFICADO	COMIENZO	FIN
3	Gestión de usuarios	40:00:00	21/12/2015	25/12/2015
	Diseño de Experiencia de usuario para clientes	8:00:00	21/12/2015	21/12/2015
	Programación CRUD de usuarios	13:00:00	22/12/2015	23/12/2015
	Implementación de ingreso al sistema a través de códigos QR, código	13:00:00	23/12/2015	25/12/2015
	Pruebas de aceptación	6:00:00	25/12/2015	25/12/2015

Tabla 3-38 Pruebas de aceptación - iteración 03. Fuente (Antonio Córdova, 2016)

PRUEBA ACEPTACIÓN 04	GESTIÓN DE USUARIOS	CUMPLIMIENTO DE RESULTADOS ESPERADOS	APROBADO POR TRACKER	APROBADO POR CLIENTE
RESULTADOS ESPERADOS	El sistema debe presentar la página principal del usuario.	X	X	X
	El sistema debe mostrar los productos disponibles para el usuario.	X	X	X

Tabla 3-39 Prueba de aceptación 04.

Fuente (Antonio Córdova, 2016)

3.3.5. Iteración 04 - Gestión básica de paquetes

En la presente iteración se trabajará tanto en las empresas como en el CRUD de paquetes, esto se debe a que una empresa debe asociarse con uno o más paquetes.

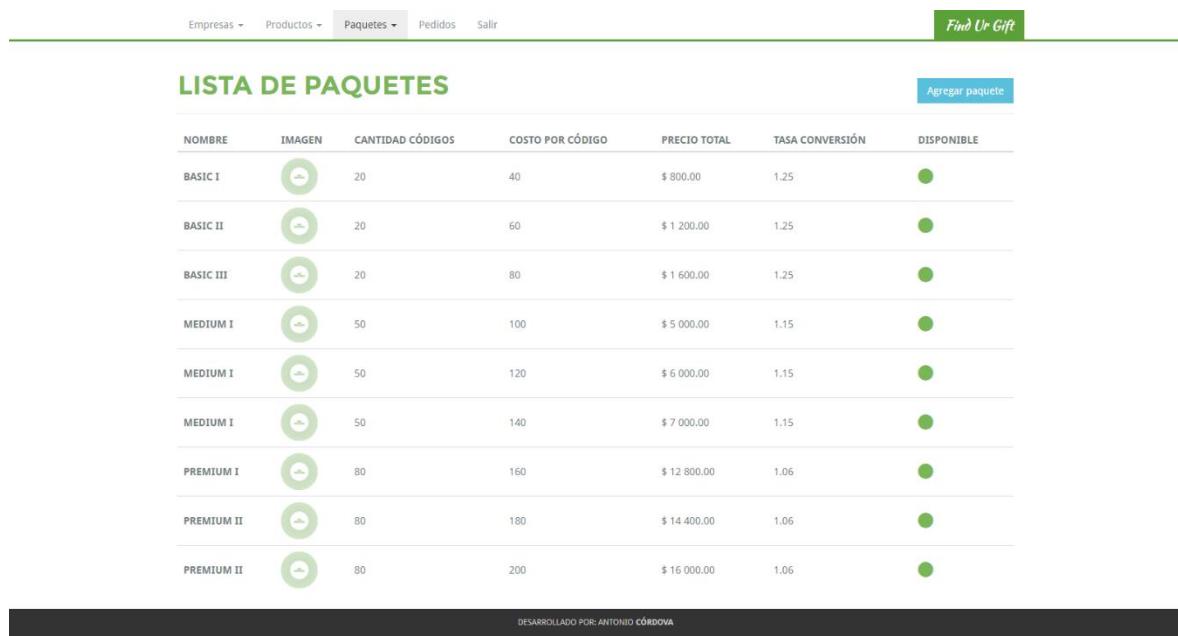
IT.	TAREAS	TIEMPO PLANIFICADO	PROPIETARIO
4	Gestión básica de paquetes	40:00:00	Antonio Córdova
	Diseño de Experiencia de usuario para paquetes	19:00:00	
	Programación CRUD de tarjetas	15:00:00	
	Pruebas de aceptación	6:00:00	

Tabla 3-40 Iteración 04 - Gestión paquetes.

Fuente (Antonio Córdova, 2016)

3.3.5.1. Diseño de Experiencia de usuario para paquetes

Se requiere la creación de paquetes y que a su vez generen códigos para realizar el canje respectivo por productos. Para ello se realizó la maqueta siguiente:



La maqueta muestra una interfaz web con la siguiente estructura:

- Barra superior con menús: Empresas, Productos, Paquetes, Pedidos, Salir y un botón "Find Ur Gift".
- Título "LISTA DE PAQUETES" en verde.
- Boton "Agregar paquete" en azul.
- Tabla de datos con las siguientes columnas: NOMBRE, IMAGEN, CANTIDAD CÓDIGOS, COSTO POR CÓDIGO, PRECIO TOTAL, TASA CONVERSIÓN y DISPONIBLE.
- Los datos de la tabla son:

NOMBRE	IMAGEN	CANTIDAD CÓDIGOS	COSTO POR CÓDIGO	PRECIO TOTAL	TASA CONVERSIÓN	DISPONIBLE
BASIC I		20	40	\$ 800.00	1.25	
BASIC II		20	60	\$ 1 200.00	1.25	
BASIC III		20	80	\$ 1 600.00	1.25	
MEDIUM I		50	100	\$ 5 000.00	1.15	
MEDIUM I		50	120	\$ 6 000.00	1.15	
MEDIUM I		50	140	\$ 7 000.00	1.15	
PREMIUM I		80	160	\$ 12 800.00	1.06	
PREMIUM II		80	180	\$ 14 400.00	1.06	
PREMIUM II		80	200	\$ 16 000.00	1.06	

- Nota en la parte inferior: DESARROLLADO POR: ANTONIO CÓRDOVA

Gráfico 3-32 Visualización de paquetes

Fuente (Antonio Córdova, 2016)

AGREGAR NUEVO PAQUETE

Nombre de paquete:

Cantidad de códigos:

Costo por código:

Precio Total:

Tasa de conversión:

¿Paquete Disponible?

NO IMAGEN

Seleccionar imagen de paquete
[Seleccionar archivo] Ningún archivo seleccionado

DESARROLLADO POR: ANTONIO CÓRDOVA

Gráfico 3-33 Creación de paquetes
Fuente (Antonio Córdova, 2016)

Con estas dos páginas se puede implementar la sección de paquetes

3.3.5.2. Programación CRUD de paquetes

Modelo

Paquetes.php es la clase que almacena los datos de la tabla Paquetes y en la cual se crea las relaciones con otras tablas:

- **Productos:** Relación muchos a muchos, donde la tabla intermedia es paquetes_productos
- **Empresas:** Relación muchos a muchos, donde la tabla intermedia es empresas_paquetes
- **Códigos:** Relación uno a muchos

```

<?php
namespace App;
use ...

class Paquetes extends Model
{
    /**
     * The table associated with the model.
     *
     * @var string
     */
    protected $table = 'paquetes';

    /**
     * The Primary key for this model
     *
     * @var string
     */
    protected $primaryKey = 'PAQID';

    /**
     * Productos que pertenecen a paquetes
     */
    public function productos(){
        return $this->belongsTo('App\Productos', 'paquetes_productos', 'PAQID', 'PROID');
    }

    /**
     * Empresas que perteneceon a productos
     */
    public function empresas(){
        return $this->belongsToMany('App\Empresas', 'empresas_paquetes', 'PAQID', 'EMPID');
    }

    /**
     * Obtención de códigos para el paquete
     */
    public function codigos(){
        return $this->hasMany('App\Codigos');
    }
}

```

Gráfico 3-34 Modelo: Paquetes.php

Fuente (Antonio Córdova, 2016)

Controlador

La clase **PaquetesController.php** ubicada en app > Http > Controllers > Admin, es quien controla el CRUD, a través de los métodos:

- index
- process
- delete

Vista

Se requiere de 3 vistas para poder realizar el CRUD:

- **Lista de Paquetes y eliminación:** Una tabla que contiene todos los datos de los paquetes y las opciones cara cada uno.
- **Edición:** Un formulario para editar algún paquete:
- **Creación:** Similar al formulario de edición sin datos iniciales

3.3.5.3. Resultados

NOMBRE	IMAGEN	CANTIDAD CÓDIGOS	COSTO POR CÓDIGO	PRECIO TOTAL	TASA CONVERSIÓN	ESTADO	OPCIONES
Basic I		50	10.00	500.00	1.2	●	editar eliminar

DESARROLLADO POR: ANTONIO CÓRDOVA

Gráfico 3-35 Lista de paquetes
Fuente (Antonio Córdova, 2016)

Nombre de paquete: Basic I

Cantidad de códigos: 50

Costo por código: 10.00

Precio Total: 500.00

Tasa de conversión: 1.2

Opciones: ¿Paquete Disponible? ¿Destacado?

Guardar Cancelar

Seleccionar imagen del paquete

Subir imagen

DESARROLLADO POR: ANTONIO CÓRDOVA

Gráfico 3-36 Edición de paquete
Fuente (Antonio Córdova, 2016)

Se cambió el nombre del paquete a Básico Prueba, obteniendo el resultado: Se actualizó el paquete Básico Prueba correctamente

The screenshot shows a web-based application interface titled 'Find Ur Gift :: Lista de paquetes'. The top navigation bar includes links for 'Productos', 'Empresas', 'Paquetes', 'Reportes', and 'Salir'. On the right, there's a 'Find Ur Gift' logo. Below the navigation, a green banner displays the message 'Se actualizó el paquete Básico Prueba correctamente'. The main content area is titled 'LISTA DE PAQUETES' and contains a table with one row of data:

NOMBRE	IMAGEN	CANTIDAD CÓDIGOS	COSTO POR CÓDIGO	PRECIO TOTAL	TASA CONVERSIÓN	ESTADO	OPCIONES
Básico Prueba		50	10.00	500.00	1.2	(green circle)	editar eliminar

A dark footer bar at the bottom states 'DESARROLLADO POR: ANTONIO CÓRDOVA'.

Gráfico 3-37 Actualización exitosa de paquete
Fuente (Antonio Córdova, 2016)

Para crear un paquete nuevo se hace clic en agregar paquete y llenamos los campos, con la cantidad mínima requerida (5 códigos)

The screenshot shows the 'Find Ur Gift :: Crear paquete' page. The top navigation bar is identical to the previous screenshot. The main content area is titled 'AGREGAR NUEVO PAQUETE'. On the left, there are input fields for package details:

- Nombre de paquete: Paquete nuevo
- Cantidad de códigos: 5
- Costo por código: 10
- Precio Total: 50
- Tasa de conversión: 1.2
- Opciones: ¿Paquete Disponible? ¿Destacado?

On the right, there's a 'Seleccionar imagen del paquete' section containing a preview image of a QR code labeled 'Find ur Gift' with '10 dólares' and some gift icons. Below it is another 'Seleccionar imagen del paquete' section with a 'Subir imagen' button.

A dark footer bar at the bottom states 'DESARROLLADO POR: ANTONIO CÓRDOVA'.

Gráfico 3-38 Creación de nuevo paquete
Fuente (Antonio Córdova, 2016)

El resultado es: Se agregó el paquete Paquete nuevo correctamente

The screenshot shows a web application interface for managing packages. At the top, there's a header with the title 'Find Ur Gift :: Lista de paquetes' and a sub-header 'webcordova.dev/admin/paquetes'. Below the header is a navigation bar with links for 'Productos', 'Empresas', 'Paquetes', 'Reportes', and 'Salir'. On the right side of the header is a green button labeled 'Find Ur Gift'. The main content area has a green header bar with the text 'Se agregó el paquete Paquete nuevo correctamente'. Below this is a table with the following data:

NOMBRE	IMAGEN	CANTIDAD CÓDIGOS	COSTO POR CÓDIGO	PRECIO TOTAL	TASA CONVERSIÓN	ESTADO	OPCIONES
Básico Prueba		50	10.00	500.00	1.2	●	editar eliminar
Paquete nuevo		5	10.00	50.00	1.2	●	editar eliminar

At the bottom of the page, there's a footer with the text 'DESARROLLADO POR: ANTONIO CÓRDOVA' and some technical details: 'Window size: 1139 x 513' and 'Viewport size: 1123 x 420'.

Gráfico 3-39 Creación de paquete exitoso
Fuente (Antonio Córdova, 2016)

3.3.5.4. Pruebas de aceptación

Al término de la iteración 04 se obtuvieron los siguientes resultados durante 5 días de trabajo:

IT.	TAREAS	TIEMPO PLANIFICADO	COMIENZO	FIN
4	Gestión básica de tarjetas	40:00:00	4/1/2016	8/1/2016
	Diseño de Experiencia de usuario para paquetes	19:00:00	4/1/2016	6/1/2016
	Programación CRUD de paquetes	15:00:00	6/1/2016	8/1/2016
	Pruebas de aceptación	6:00:00	8/1/2016	8/1/2016

Tabla 3-41 Pruebas de aceptación - iteración 04.

Fuente (Antonio Córdova, 2016)

PRUEBA ACEPTACIÓN 05	GESTIÓN BÁSICA DE PAQUETES	CUMPLIMIENTO DE RESULTADOS ESPERADOS	APROBADO POR TRACKER	APROBADO POR CLIENTE
RESULTADOS ESPERADOS	La página debe mostrar las tarjetas habilitadas y mostrar características de las tarjetas	X	X	X

Tabla 3-42 Prueba de aceptación 05.

Fuente (Antonio Córdova, 2016)

3.3.6. Iteración 05 - Gestión básica de empresas

El trabajo de esta iteración es la construcción de la plantilla para las empresas, conjuntamente con la selección de productos para cada paquete asignado a la empresa.

Esta iteración se completará con dos administraciones:

- **Básica** que comprende la creación, modificación, eliminación y vista de los paquetes.
- **Complementaria** que comprende la selección de los productos por paquete adquirido, en otras palabras, cada paquete adquirido por una empresa debe ser asociado por varios productos seleccionados, caso contrario los paquetes permanecerán vacíos.

IT.	TAREAS	TIEMPO PLANIFICADO	PROPIETARIO
5	Gestión básica de empresas	40:00:00	Antonio Córdova
	Diseño de Experiencia de usuario para empresas	8:00:00	
	Programación CRUD de empresas	13:00:00	
	Implementación de ingreso al sistema a través usuario y contraseña	13:00:00	
	Pruebas de aceptación	6:00:00	

Tabla 3-43 Iteración 05 - Gestión tarjetas.

Fuente (Antonio Córdova, 2016)

Debido al tamaño de esta iteración y lo solicitado por el cliente, se extiende esta iteración por una semana más y pasa a formar una nueva prueba de aceptación.

Gracias a la metodología XP se pudo continuar con el trabajo a pesar de este nuevo ciclo.

3.3.6.1. Diseño de Experiencia de usuario para paquetes

Se requiere la creación de maquetas para el CRUD de empresas, de igual manera para la selección de productos por cada paquete.

Gráfico 3-40 Creación de nueva empresa.
Fuente (Antonio Córdova, 2016)

Nombre Empresa	Contacto	E-mail	Notificación por canje	Notificación por puntos	Tasa de conversión	Opciones
Yakindu	Mario Moroho	mario@yakindu.ec	semanal	semanal	1.2	
Yakindu	Mario Moroho	mario@yakindu.ec	semanal	semanal	1.2	
Yakindu	Mario Moroho	mario@yakindu.ec	semanal	semanal	1.2	

Gráfico 3-41 Lista de empresas.
Fuente (Antonio Córdova, 2016)

Las vistas presentadas son de uso para el administrador por lo que se aplicará la plantilla de administrador creada en la iteración 02.

Con las vistas anteriores se da inicio a la implementación en 3 capas del CRUD de empresas.

3.3.6.2. Programación CRUD de empresas

Modelo

Empresas.php es la clase que almacena los datos de la tabla Empresas y en la cual se crea las relaciones con otras tablas:

- **Usuarios:** Relación muchos a muchos, donde la tabla intermedia es `usuarios_empresas`
- **Productos:** Relación muchos a muchos, donde la tabla intermedia es `empresas_productos`
- **Paquetes:** Relación muchos a muchos, donde la tabla intermedia es `empresas_paquetes`

```

<?php
namespace App;
use Illuminate\Database\Eloquent\Model;

class Empresas extends Model
{
    /**
     * The table associated with the model.
     */
    protected $table = 'empresas';

    /**
     * The Primary key for this model
     */
    protected $primaryKey = 'EMPID';

    /**
     * Usuarios que pertenecen a empresas
     */
    public function usuarios()
    {
        return $this->belongsToMany('App\Usuarios', 'usuarios_empresas', 'EMPID', 'USRID');
    }

    /**
     * Productos que pertenecen a Empresas
     */
    public function productos()
    {
        return $this->belongsToMany('App\Productos', 'empresas_productos', 'EMPID', 'PROID');
    }

    /**
     * Paquetes que pertenecen a Empresas
     */
    public function paquetes()
    {
    }
}

```

Gráfico 3-42 Modelo: *Empresas.php*.
Fuente (Antonio Córdova, 2016)

Controlador

La clase ***EmpresasController.php*** ubicada en `app > Http > Controllers > Admin`, es quien controla el CRUD, a través de los siguientes métodos:

- Index
- Process

El cliente ha mencionado la idea de no eliminar las empresas por temas de fiscalización y control de ventas, por lo que no se implementará este método.

The screenshot shows the PhpStorm IDE interface with the following details:

- Project Tree:** The left sidebar displays the project structure under "webcordova". It includes the "app" folder, which contains "Console", "Events", "Exceptions", "Helpers", "Http" (with "Controllers" and "Admin" subfolders), "Auth", "Client" (with "AuthClient", "Carto", "Datos", "Productos" subfolders), "Empresas" (with "Controller" subfolder), "Libraries", "Middleware", "Requests", "Kernel", "routes.php", "Jobs", "Listeners", "Policies", "Providers", and "Empleados" (with "Admin", "Ciudades", "Codigos", and "Empresas" subfolders).
- Code Editor:** The main editor window shows the code for `Admin\EmpresasController.php`. The code defines a class `EmpresasController` that extends `Controller`. It includes validation rules for `$rules` and messages for `$rulesMessages`, and an `index` method that paginates results.
- Toolbars and Status Bar:** The top bar includes standard file operations like File, Edit, View, Navigate, Code, Refactor, Run, Tools, VCS, Window, Help. The status bar at the bottom right shows "816 IFCs UTE-82 3s".

Gráfico 3-43 Controlador: EmpresasController.php.

Fuente (Antonio Córdova, 2016)

Vista

Se requiere de 3 vistas para poder realizar el CRUD:

- **Lista de Empresas:** Una tabla que contiene todos los datos de las empresas y las opciones para cada una.
 - **Edición de Empresas:** Un formulario para editar la empresa seleccionada.
 - **Creación de Empresas:** Un formulario para crear una empresa nueva

3.3.6.3. Resultados

The screenshot shows a web browser window for 'Find Ur Gift : Lista de empresas'. The URL is 'webcordova.dev/admin/empresas'. The top navigation bar includes links for 'Productos', 'Empresas' (highlighted in green), 'Paquetes', 'Reportes', and 'Salir'. A green header bar on the right contains the 'Find Ur Gift' logo. The main content area has a title 'LISTA DE EMPRESAS' and two buttons: 'Agregar empresa' (Add company) and 'Reporte' (Report). Below is a table listing two companies:

	Nombre Empresa	Contacto	E-mail	Tasa costo	Paquetes	Usuarios	Opciones
	Pronaca	Jaime Herrera	jaime@pronaca.com	2.10	 		editar
	Nestle	Margarita del Castillo	mcastillo@nestle.com	1.40	 		editar

Gráfico 3-44 Vista: lista de empresas.

Fuente (Antonio Córdova, 2016)

EDITAR EMPRESA

*campos obligatorios

Nombre*:	Pronaca
Contacto*:	Jaime Herrera
Email*:	jaime@pronaca.com
Contraseña*:	*****
Tasa de puntos*:	2.10

Guardar **Cancelar**

Seleccionar imagen de la empresa
PRONACA
Subir imagen

DESARROLLADO POR: ANTONIO CÓRDOVA

Gráfico 3-45 Vista: editar empresa.
Fuente (Antonio Córdova, 2016)

AGREGAR NUEVA EMPRESA

*campos obligatorios

Nombre*:	
Contacto*:	
Email*:	
Contraseña*:	
Tasa de puntos*:	

Guardar **Cancelar**

Seleccionar imagen de la empresa
NO IMAGEN
Subir imagen

DESARROLLADO POR: ANTONIO CÓRDOVA

Gráfico 3-46 Vista: crear empresa.
Fuente (Antonio Córdova, 2016)

Se cambió el nombre de contacto de la empresa, obteniendo como resultado: **Se actualizó la empresa Pronaca correctamente**

EDITAR EMPRESA

*campos obligatorios

Nombre*:	Pronaca
Contacto*:	Hector Ortiz
Email*:	hortiz@pronaca.com
Contraseña*:	*****
Tasa de puntos*:	2.10

Seleccionar imagen de la empresa

Subir imagen

Guardar **Cancelar**

DESARROLLADO POR: ANTONIO CÓRDOVA

Gráfico 3-47 Edición de empresa.
Fuente (Antonio Córdova, 2016)

AGREGAR NUEVA EMPRESA

*campos obligatorios

Nombre*:	Moderna
Contacto*:	Marielisa Gomez
Email*:	mgomez@moderna.com
Contraseña*:	*****
Tasa de puntos*:	1.5

Seleccionar imagen de la empresa

Subir imagen

Guardar **Cancelar**

DESARROLLADO POR: ANTONIO CÓRDOVA

Gráfico 3-48 Creación de empresa.
Fuente (Antonio Córdova, 2016)

Una vez creada la empresa, se muestra el mensaje de éxito: **Se agregó la empresa Moderna correctamente**

The screenshot shows a web application interface titled "Find Ur Gift :: Lista de empresas". The URL in the address bar is "webcordova.dev/admin/empresas". The top navigation menu includes "Productos", "Empresas" (selected), "Paquetes", "Reportes", and "Salir". On the right side of the header is a green button labeled "Find Ur Gift". Below the header, there is a success message: "Se agregó la empresa Moderna correctamente". A blue button labeled "Agregar empresa" and a link labeled "Reporte" are also present. The main content area displays a table titled "LISTA DE EMPRESAS" with the following data:

	Nombre Empresa	Contacto	E-mail	Tasa costo	Paquetes	Usuarios	Opciones
	Pronaca	Hector Ortiz	hortiz@pronaca.com	2.10			editar
	Nestle	Margarita del Castillo	mcastillo@nestle.com	1.40			editar
	Moderna	Marielisa Gomez	mgomez@moderna.com	1.50			editar

At the bottom of the page, a black footer bar contains the text "DESARROLLADO POR: ANTONIO CÓRDOVA".

Gráfico 3-49 Creación de empresa exitosa.

Fuente (Antonio Córdova, 2016)

3.3.6.4. Pruebas de aceptación

En la primera parte de la presente iteración se obtuvieron los siguientes resultados durante los 5 días de trabajo

IT.	TAREAS	TIEMPO PLANIFICADO	COMIENZO	FIN
5	Gestión básica de empresas	40:00:00	11/1/2016	15/1/2016
	Diseño de Experiencia de usuario para empresas	8:00:00	11/1/2016	11/1/2016
	Programación CRUD de empresas	13:00:00	12/1/2016	13/1/2016
	Implementación de ingreso al sistema a través usuario y contraseña	13:00:00	13/1/2016	15/1/2016
	Pruebas de aceptación	6:00:00	15/1/2016	15/1/2016

Tabla 3-44 Pruebas de aceptación - iteración 05.

Fuente (Antonio Córdova, 2016)

PRUEBA ACEPTACIÓN 06-A	GESTIÓN BÁSICA DE EMPRESAS	CUMPLIMIENTO DE RESULTADOS ESPERADOS	APROBADO POR TRACKER	APROBADO POR CLIENTE
RESULTADOS ESPERADOS	El sistema debe mostrar la página principal de empresas	X	X	X

Tabla 3-45 Prueba de aceptación 05.

Fuente (Antonio Córdova, 2016)

3.3.7. Iteración 05 - Selección de productos

El cliente solicita, a través de la historia de usuario 07, que sean los usuarios tipo empresa quienes seleccionen por ellos mismos los productos y por cada paquete.

IT.	TAREAS	TIEMPO PLANIFICADO	PROPIETARIO
5	Selección de productos	40:00:00	Antonio Córdova
	Diseño de Experiencia de usuario para la selección de productos	16:00:00	
	Implementación de la selección de productos	18:00:00	
	Pruebas de aceptación	6:00:00	

Tabla 3-46 Iteración 05 - Selección de productos.

Fuente (Antonio Córdova, 2016)

3.3.7.1. Diseño de experiencia de usuario para selección de productos

El objetivo de esta iteración es ofrecer al usuario una herramienta simple de uso que permita la selección de uno o varios productos para que sean visualizados a su vez por sus clientes, de manera que puedan ser canjeados por sus puntos.

En el gráfico 3-48 se explicará de mejor manera el funcionamiento de esta historia de usuario:

1. Creación de productos [Administración]



2. Selección de productos [Empresas]



3. Productos mostrados [Clientes]

Gráfico 3-50 Proceso de selección de productos
Fuente (Antonio Córdova, 2016)

El diseño de la maqueta es el mostrado en el siguiente gráfico, el cual tuvo varias modificaciones para mejorar el desempeño y la usabilidad del usuario, en este caso empresarial.

The screenshot shows a user interface titled "SELECCIÓN DE PRODUCTOS". At the top, there is a navigation bar with links: Inicio, Paquetes, Actualizar datos, Reporte, and Salir. On the right side, there is an email link (info@findurgift.com) and a phone number (25 13 49 75). Below the title, it says "Paquete: Premium I".

TECNOLOGÍA

	Elegir	Elegir	Elegir	Elegir
654.00	654.00	654.00	654.00	
nombre producto	nombre producto	nombre producto	nombre producto	
Resumen	Resumen	Resumen	Resumen	

To the right of the table, there is a "Filtrar productos por tipo" dropdown set to "Todos" and a "Confirmar productos seleccionados" button. Below the table, there is a red button labeled "Seleccionados todos".

At the bottom of the interface, it says "DESARROLLADO POR: ANTONIO CÓRDOVA".

Gráfico 3-51 Selección de productos.
Fuente (Antonio Córdova, 2016)

Con la vista anterior se da inicio a la Implementación de la selección de productos.

3.3.7.2. Implementación de la selección de productos

La implementación comprende el trabajo en las tres capas (modelo, vista y controlador), para esta tarea se requiere el uso de varios archivos creados en anteriores iteraciones debido a que esta funcionalidad requiere de procesos anteriores.

Modelo

No se requiere generar un nuevo modelo sino más bien llamar a los ya creados, estos son:

- **Empresas:** Es necesario conocer cuál es la empresa que está realizando el proceso de selección.
- **Paquetes:** Se requiere conocer qué paquete va a recibir los productos seleccionados por la empresa.
- **Productos:** Se obtiene la lista de productos disponibles.
- **Tipo de producto:** Se obtiene una lista de tipos de producto útil para filtrar los productos

Controlador

La clase ***PaquetesController.php*** ubicada en app > Http > Controllers > Empresas, es quien controla la actividad sobre los productos, a través de los siguientes métodos:

- **Index:** Muestra todos los paquetes disponibles a la empresa
- **Productos:** Muestra todos los productos disponibles a la empresa
- **getProductos:** Muestra los productos filtrados por tipo
- **setProductos:** Activa o desactiva el producto en un paquete
- **remAll:** Quita todos los productos seleccionados

```

webcordova - [D:\Google Drive\PUCE\TESIS\02 Implementación y Desarrollo\Implementación\webcordova] - ...app\Http\Controllers\Empresas\PaquetesController.php - PhpStorm 2016.1
File Edit View Navigate Code Refactor Run Tools VCS Window Help
webcordova app Http Controllers Empresas PaquetesController.php
Project Structure Favorites Gulp
src Project app
  -> Controllers
    -> Admin
    -> Auth
    -> Cliente
    -> Empresas
      -> AuthEmpresas.php
      -> EmpresasController.php
      -> PaquetesController.php
    -> Http
      -> Controllers
        -> Admin
        -> Auth
        -> Cliente
        -> Empresas
          -> AuthEmpresas.php
          -> EmpresasController.php
          -> PaquetesController.php
      -> Libraries
      -> Middleware
      -> Requests
        -> Kernel.php
        -> routes.php
      -> Jobs
      -> Listeners
      -> Policies
      -> Providers
        -> Admin.php
        -> Ciudades.php
        -> Codigos.php
        -> Empresas.php
        -> Paquetes.php
        -> Pedidos.php
        -> Productos.php
        -> Tipoproductos.php
        -> User.php
        -> Usuarios.php
        -> ZonasGeograficas.php
      -> bootstrap
      -> config
  -> Favorites
  -> Gulp
  -> Terminal File Transfer
  -> Event Log
  -> Remote Host Database
PaquetesController.php
1 <?php
2 namespace App\Http\Controllers\Empresas;
3
4 use ...
5
6 class PaquetesController extends Controller
7 {
8     public function __construct()
9     {
10         if (!Session::has('empresa_login_id')) {
11             Redirect::to('/empresas/login')->send();
12         }
13     }
14
15     public $scripts_after;
16     public $styles;
17
18     public function index()
19     {
20         $this->scripts_after = array(
21             'js/owl.carousel.min.js',
22             'js/empresa/empresa-owl.js',
23         );
24         view()->share(array(
25             'scripts_after' => $this->scripts_after
26         ));
27         $empresa = Empresas::find(Session::get('empresa_login_id'));
28
29         $paquetes = $empresa->paquetes();
30
31         return view('empresa/paquetes/index', array('currPage' => 'paquetes', 'empresa' => $empresa, 'paquetes' => $paquetes));
32     }
33
34     public function productos($id)
35     {
36         $this->scripts_after = array(
37             'js/empresa/productos.js',
38             'js/jquery.matchHeight-min.js',
39         );
40     }
41
42     public function paquetes($id)
43     {
44         $this->scripts_after = array(
45             'js/empresa/Carousel.js',
46             'js/owl.carousel.min.js',
47         );
48     }
49 }

```

Gráfico 3-52 Controlador PaquetesController.php.
Fuente (Antonio Córdova, 2016)

Vista

Se han generado 3 vistas importantes:

- Inicial:** Para seleccionar el paquete al que se desea asignar los productos
- Productos:** Para seleccionar los productos al paquete seleccionado anteriormente.
- Detalle producto:** Para realizar el bucle de productos, útil para el filtrado de productos.

```

webcordova - [D:\Google Drive\PUCE\TESIS\02 Implementación y Desarrollo\Implementación\webcordova] - ...resources\views\empresa\paquetes\index.blade.php - PhpStorm 2016.1
File Edit View Navigate Code Refactor Run Tools VCS Window Help
webcordova resources views empresa paquetes index.blade.php
Project Structure Favorites Gulp
src Project assets lang views
  -> admin
  -> empresas
  -> mobile
  -> paquetes
    -> add.blade.php
    -> edit.blade.php
    -> index.blade.php
  -> productos
    -> paquetes
      -> detalle_producto.blade.php
      -> index.blade.php
        -> productos.blade.php
        -> login.blade.php
  -> errors
  -> templates
    -> admin
    -> basic
    -> emails
    -> empresa
      -> elements
        -> empresa.blade.php
    -> mobile
    -> usuario
      -> admin.blade.php
      -> empresa.blade.php
      -> user.blade.php
    -> usuarios
    -> vendor
  -> Event Log
  -> Remote Host Database
index.blade.php
1 << div.container >>
2   <div>@extends('template.admin.empresase')

```

Gráfico 3-53 Vista index.blade.php.
Fuente (Antonio Córdova, 2016)

Estas vistas se encuentran en el directorio: views > empresa > paquetes. Asimismo, utilizan la plantilla empresarial creada anteriormente.

3.3.7.3. Resultados

The screenshot shows a web application interface titled 'ZONA EMPRESARIAL'. At the top right, there is a green button labeled 'INICIAR SESIÓN'. Below the title, there is a large circular logo with '150' in the center, surrounded by 'AÑOS' and 'ESTADOS UNIDOS'. To the right of the logo, there is a section for 'INVESTIGACIÓN' with a 'VER' button. Below this, there is a section titled 'PAQUETES ADQUIRIDOS' featuring a gift card icon with 'Find ur Gift' and '10 dólares'. A QR code is also present. Below the icon, there is a table with the following data:

Paquete:	Basic I
Códigos disponibles:	50
Selección productos	

Gráfico 3-54 Zona empresarial - lista de paquetes.
Fuente (Antonio Córdova, 2016)

Cada paquete tiene un botón para seleccionar sus productos individualmente. Al hacer clic en Selección de productos, se muestra todos los productos.

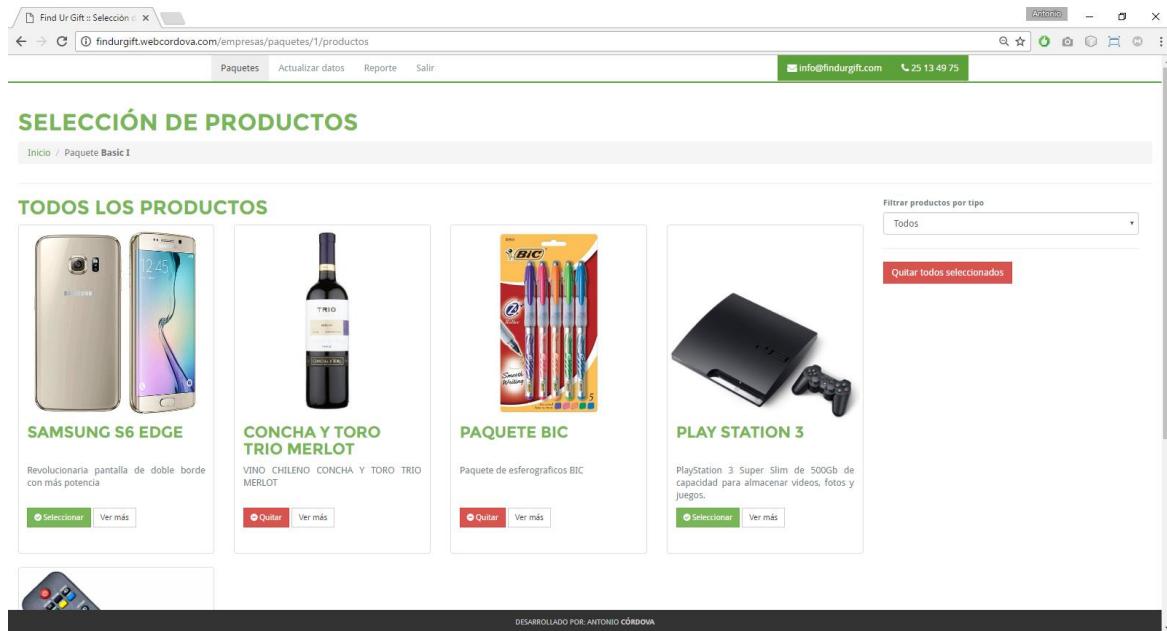


Gráfico 3-55 Selección de productos.

Fuente (Antonio Córdova, 2016)

En la imagen anterior, se visualizan 2 productos seleccionados (Botones rojos “Quitar”) ya que el botón de selección cambio a rojo con el texto “Quitar”, si el usuario desea quitar o deseleccionar se debe hacer clic nuevamente en dicho botón.

Al lado derecho se encuentra el filtro de productos el cual selecciona los productos por tipo. Finalmente, en la parte inferior del filtro se ubica el botón para quitar todos los productos seleccionados.

3.3.7.4. Pruebas de aceptación

Al término de la iteración 05 complementaria se obtuvieron los siguientes resultados durante 5 días de trabajo:

IT.	TAREAS	TIEMPO PLANIFICADO	COMIENZO FIN	
			COMIENZO	FIN
5	Selección de productos	40:00:00	18/1/2016	22/1/2016
	Diseño de Experiencia de usuario para la selección de productos	16:00:00	18/1/2016	19/1/2016
	Implementación de la selección de productos	18:00:00	20/1/2016	22/1/2016
	Pruebas de aceptación	6:00:00	22/1/2016	22/1/2016

Tabla 3-47 Pruebas de aceptación - iteración 05 – Complementaria.

Fuente (Antonio Córdova, 2016)

La prueba de aceptación agregada es la onceava ya que al inicio de las iteraciones se tomó la decisión de ampliar la iteración 5 con dos pruebas de aceptación.

PRUEBA ACEPTACIÓN 11	SELECCIÓN DE PRODUCTOS	CUMPLIMIENTO DE RESULTADOS ESPERADOS	APROBADO POR TRACKER	APROBADO POR CLIENTE
RESULTADOS ESPERADOS	El sistema debe mostrar todos los productos con: Un botón para seleccionar o quitar el producto Un botón para ver detalles del producto	X	X	X
RESULTADO FINAL	Productos agregados a cada paquete	X	X	X

Tabla 3-48 Prueba de aceptación 11.

Fuente (Antonio Córdova, 2016)

3.3.8. Iteración 06 - Creación de proceso de canje

Esta iteración inicia el proceso de canje de productos, una vez el cliente haya seleccionado el o los productos hasta la confirmación de los datos ingresados. Para lo cual se requiere que el diseño ayude al usuario a identificar claramente los productos para que puedan ser agregados al carrito de compra y finalmente, dar inicio al proceso de canje por los puntos.

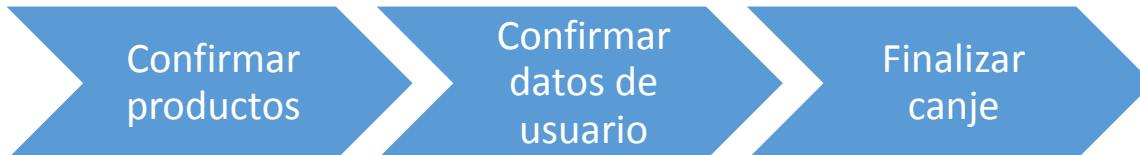
IT.	TAREAS	TIEMPO PLANIFICADO	PROPIETARIO
6	Creación de proceso de canje	40:00:00	Antonio Córdova
	Diseño de Experiencia de usuario para el proceso de canje	16:00:00	
	Implementación del proceso de canje	20:00:00	
	Pruebas de aceptación	4:00:00	

Tabla 3-49 Iteración 06 - Creación de proceso canje.

Fuente (Antonio Córdova, 2016)

3.3.8.1. Diseño de Experiencia de usuario para la selección de productos

Se requiere generar el proceso de canje de productos por los puntos del cliente, para lo cual, se estableció el siguiente proceso:

Gráfico 3-56 Proceso de canje.
Fuente (Antonio Córdova, 2016)

Confirmar productos

En esta sección aparecerá la lista de productos seleccionados por el cliente, quien debe ratificar que los productos mostrados son los que eligió.

Confirmar datos de usuario

En esta sección debe verificar los datos del cliente, los cuales son obligatorios debido a que son necesarios para la entrega del producto. Este formulario es el mismo que se presenta en los datos de usuario en caso que el cliente decida ingresar los datos con anterioridad.

Finalizar canje

El sistema finaliza el canje almacenando los datos del usuario y finalmente, envía el correo electrónico tanto al cliente como al administrador con los datos de los productos canjeados.

Se plantea la siguiente estructura de contenidos, en donde se debe destacar el proceso de canje, el área de contenido, el resto del contenido visual se toma de la plantilla de usuarios.

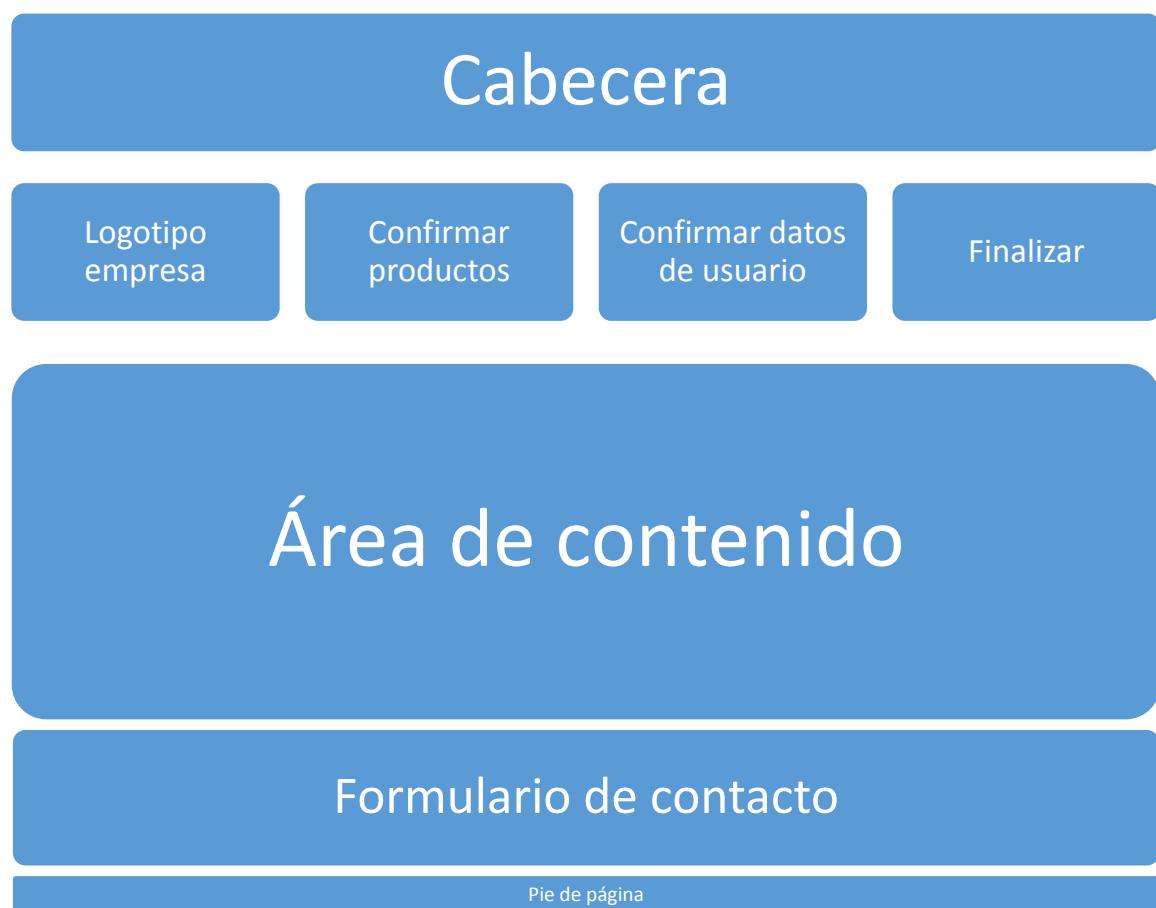


Gráfico 3-57 Estructura de contenidos – proceso de canje.
Fuente (Antonio Córdova, 2016)

Finalmente, la maqueta propuesta es la siguiente:

Continua el proceso si los productos en pantalla son los elegidos

CONFIRMAR PRODUCTOS

Continua el proceso si los productos en pantalla son los elegidos

Confirmar Productos **Confirmar Datos de Usuario** **Finalizar**

10 pts.
Paquete BIC

Confirmar productos **Volver a la lista de productos**

CONTÁCTANOS

Sed ut perspiciatis unde omnis iste natus error sit voluptatem rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi explicabo. Donec ut voluptat metus. Aliquam tortor lorem. Sed ut perspiciatis unde omnis iste natus error sit voluptatem

Nombre:
Apellidos:
Correo Electrónico:

Mensaje:

Enviar

DESARROLLADO POR: ANTONIO CÓRDובה

Gráfico 3-58 Maqueta - confirmar productos
Fuente (Antonio Córdova, 2016)

Continua el proceso una vez hayas ingresado todos los datos siguientes

CONFIRMAR DATOS DE CLIENTE

Nombres: Marcela
Apellidos: Cifuentes
Email: marcecifuentes@hotmail.com
Teléfono Fijo: 2252423
Teléfono Móvil: 0998877665

DATOS DIRECCIÓN DE ENTREGA

Dirección de entrega: De los Duraznos y Manzanos
Ciudad de entrega: Quito

Guardar **Regresar a productos**

CONTÁCTANOS

Sed ut perspiciatis unde omnis iste natus error sit voluptatem rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi explicabo. Donec ut voluptat metus. Aliquam tortor lorem. Sed ut perspiciatis unde omnis iste natus error sit voluptatem

Nombre:
Apellidos:
Correo Electrónico:

Mensaje:

Enviar

DESARROLLADO POR: ANTONIO CÓRDובה

Gráfico 3-59 Maqueta - confirmar productos
Fuente (Antonio Córdova, 2016)

3.3.8.2. Implementación del proceso de canje

La implementación del proceso de canje inicia con el trabajo en las tres capas (modelo, vista y controlador), para esta tarea se requiere de tres modelos importantes Empresas, Productos y Usuarios, cada uno de ellos entrega cierta información para realizar el canje.

Modelo

No se requiere generar un nuevo modelo sino más bien llamar a los mencionados anteriormente, estos son:

- **Empresas:** Es necesario conocer cuál es la empresa que está asociado al usuario del canje.
- **Productos:** Se obtiene la lista de productos disponibles para el usuario.
- **Usuario:** Se obtiene datos esenciales del mismo que realiza el canje.

Controlador

La clase ***CarritoController.php*** ubicada en app > Http > Controllers > Cliente, es quien controla el proceso de canje de los productos seleccionados, a través de los siguientes métodos:

- **Process:** Procesa el ingreso de un producto al carrito de canje, en otras palabras, cuando un usuario haga clic en agregar al carrito, éste método se encarga de controlar si se agrega o se quita un producto del arreglo de productos del carrito.
- **confirmaProductos:** Proceso de aceptación por parte del cliente en donde está de acuerdo con los productos seleccionados.
- **procesoDatos:** Muestra el formulario de datos de usuario, si el usuario ya llenó los datos con anterioridad, los muestra, caso contrario, presenta los campos vacíos.
- **ConfirmaDatos:** Cuando el usuario hace clic en guardar información, este método, almacena los datos ingresados por el usuario.
- **Trash:** Método que elimina todos los datos de productos del carrito.

```

<?php
namespace App\Http\Controllers\Cliente;
use ...

class CarritoController extends Controller
{
    public function __construct() {...}

    private $rules = array(...);
    private $rulesMessages = array(...);

    /**
     * @param $idPro
     * @param $action
     * @return mixed
     * Procesa el ingreso de un producto al carrito
     */
    public function process($idPro, $action)
    {
        $carrito = Session::get('carrito');
        $producto = Productos::find($idPro);
        if ($action == 'add') {
            if ($producto) {
                $producto->PROPUNTOS = Input::get('puntos');
                $carrito['productos'][$producto->PROID] = $producto;
                $carrito['total'] += Input::get('puntos');
            }
        }
    }
}

```

MySQL - webcordova_fug@mysql.uk.cloudlogin.co (6s): Connection to MySQL - webcordova_fug@mysql.uk.cloudlogin.co failed // C... (1 hour ago) 5:5 LF+ UTF-8+ Event Log

Gráfico 3-60 Controlador CarritoController.php.

Fuente (Antonio Córdova, 2016)

Vista

Se han generado 2 vistas importantes:

- **confirmaProductos:** Para visualizar los productos seleccionados por el usuario
- **confirmaDatos:** Para mostrar el formulario de ingreso de datos del usuario.

```

<?php $carrito = Session::get('carrito') ?>
@extends('templates.usuario.usuario')
@section('title', 'Confirmar Datos de Usuario')
@section('content')
    <section id="canje">
        <div class="container">
            <div class="row">
                <div class="col-lg-12">
                    <div class="page-header">
                        @include('templates.usuario.elements.proceso')
                        <h1>Confirmar Datos de Cliente</h1>
                        <p>Continua el proceso una vez hayas ingresado todos tus datos</p>
                    </div>
                </div>
            </div>
        </div>
    </section>
    <section>
        <div class="container"><br/>
            @if (Session::has('success'))
                <div class="row">
                    <div class="col-md-12">
                        <div class="alert alert-success">
                            <p class="text-success">{{Session::get('success')}}</p>
                        </div>
                    </div>
                </div>
            @endif
        </div>
    </section>
</div>

```

MySQL - webcordova_fug@mysql.uk.cloudlogin.co (6s): Connection to MySQL - webcordova_fug@mysql.uk.cloudlogin.co failed /... (today 11:24) 1:1 CRLF+ UTF-8+ Event Log

Gráfico 3-61 Vista confirmaProductos.blade.php.

Fuente (Antonio Córdova, 2016)

Estas vistas se encuentran en el directorio: views > usuarios > canje. Utilizan la plantilla de usuarios para realizar el canje.

3.3.8.3. Resultados

El proceso de canje inicia cuando el cliente selecciona el producto de su preferencia, de modo que el producto se agrega al carrito de canje.

The screenshot shows a user interface for a swap program. At the top, there's a navigation bar with links for 'Lista de premios', 'Datos de usuario', 'Carrito', 'Salir', and contact information ('info@findurgift.com', '25 13 49 75'). The main area displays a shopping cart with one item: 'Paquete BIC' worth 10 points. Below the cart are two buttons: 'Finalizar canje' (Finish swap) and 'Vaciar carrito' (Empty cart). To the left and right of the cart, there are promotional sections for children's gifts, each featuring a placeholder text block and a 'ver ahora!' button.

BIENVENIDO! YA PUEDES ESCOGER TUS PREMIOS

Tienes disponibles 10 puntos para cangearlos cuando quieras

ESCOJE TUS PREMIOS

This screenshot shows a product selection step. The title 'ESCOJE TUS PREMIOS' is centered above a list of items. The first item shown is 'Paquete BIC' worth 10 pts. Below the item are two buttons: 'Quitar' (Remove) and 'ver más' (see more). The background has a light gray gradient.

CONTÁCTANOS

Sed ut perspiciatis unde omnis iste natus error sit voluptatem tam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi explicabo. Donec ut volutpat metus. Aliquam tortor lorem, Sed ut perspiciatis unde omnis iste natus error sit oluptatem

Nombre

Apellidos

Correo Electrónico

Mensaje

DESARROLLADO POR: ANTONIO CÓRDOVA

Gráfico 3-62 Selección de producto.
Fuente (Antonio Córdova, 2016)

Cuando el cliente ya no desea más productos, hace clic en finalizar canje, el cual da inicio al proceso de canje.

Gráfico 3-63 Confirmar Productos.
Fuente (Antonio Córdova, 2016)

El siguiente y último paso de esta iteración es la confirmación de los datos del cliente en el cual puede editarlos o si el cliente es nuevo puede llenar todos los casilleros.



CONFIRMAR DATOS DE CLIENTE

Continua el proceso una vez hayas ingresado todos los datos siguientes

Nombres:	Antonio
Apellidos:	Cordova
Email:	cordovantonio@gmail.com
Teléfono Fijo:	+593996445188
Teléfono Móvil:	+593996445188

DATOS DIRECCIÓN DE ENTREGA

Dirección de entrega:	Barcelona E13-53
Ciudad de entrega:	Quito

Guardar Regresar a productos

Gráfico 3-64 Confirmar Datos.
Fuente (Antonio Córdova, 2016)

3.3.8.4. Pruebas de aceptación

Durante los 5 días de trabajo en esta sexta iteración se obtuvieron los siguientes resultados:

IT.	TAREAS	TIEMPO PLANIFICADO	COMIENZO	FIN
6	Creación de proceso de canje	40:00:00	25/1/2016	29/1/2016
	Diseño de Experiencia de usuario para el proceso de canje	16:00:00	25/1/2016	26/1/2016
	Implementación del proceso de canje	20:00:00	27/1/2016	29/1/2016
	Pruebas de aceptación	4:00:00	29/1/2016	29/1/2016

Tabla 3-50 Pruebas de aceptación - iteración 06.

Fuente (Antonio Córdova, 2016)

PRUEBA ACEPTACIÓN 07	CREACIÓN DE PROCESO DE CANJE	CUMPLIMIENTO DE RESULTADOS ESPERADOS	APROBADO POR TRACKER	APROBADO POR CLIENTE
RESULTADOS ESPERADOS	El sistema debe mostrar los datos que ingreso el usuario para que confirmarme si están correctos.	X	X	X
RESULTADO FINAL				

Tabla 3-51 Prueba de aceptación 07.

Fuente (Antonio Córdova, 2016)

3.3.9. Iteración 07 - Proceso de canje – Finalización

Esta nueva iteración comprende la finalización del proceso de canje, es en esta etapa en la que se almacena los datos en base de datos, se genera los correos electrónicos de confirmación y el mensaje final que se culminó con éxito el canje.

IT.	TAREAS	TIEMPO PLANIFICADO	PROPIETARIO
7	Culminación de proceso de canje	8:00:00	Antonio Córdova
	Implementación de envío de correos electrónicos a usuario y administrador	4:00:00	
	Pruebas de aceptación	4:00:00	

Tabla 3-52 Iteración 07 – Culminación proceso canje.

Fuente (Antonio Córdova, 2016)

3.3.9.1. Implementación de envío de E-mails

El trabajo de esta iteración es orientado al correo electrónico el cual requiere un diseño especial tipo plantilla única para los clientes, el cual recibe la lista de productos canjeados y los procesa para mostrar en el cuerpo del correo con su respectiva foto y valor en puntos.

Modelo

Se requieren de los modelos:

- **Usuarios:** Se obtiene datos esenciales del mismo que realiza el canje
- **Códigos:** Se requiere conocer el código con el cual está ejecutando en canje.
- **Empresas:** Se necesita conocer la empresa quien le proporciona el código.

Controlador

Se utilizará la misma clase creada en la iteración seis, la cual es CarritoController.php a través del método finalizar, el cual realizará múltiples acciones:

- Almacenar datos enviados por el cliente
- Enviara el correo electrónico tanto al cliente como a la administración.

Vista

Se construyó una vista única para el correo electrónico, el cual contiene la información del cliente, pero poniendo énfasis a los productos canjeados.

The screenshot shows the PhpStorm IDE interface. The left sidebar displays the project structure under 'Project' with a tree view of files and folders. The main editor area shows the code for 'mail.blade.php'. The code is a Blade template with HTML, head, and style sections. The right sidebar contains toolbars for 'Run', 'Database', and 'Remote Host'.

```

<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>FindUrGift</title>
    <style type="text/css">
        #outlook a {
            padding: 0;
        }

        body {
            width: 100% !important;
            -webkit-text-size-adjust: 100%;
            -ms-text-size-adjust: 100%;
            margin: 0;
            padding: 0;
        }

        .ExternalClass {
            width: 100%;
        }

        .ExternalClass, .ExternalClass p, .ExternalClass span, .ExternalClass font, .ExternalClass td, .ExternalClass div {
            line-height: 100%;
        }

        #backgroundTable {
    
```

Gráfico 3-65 Vista mail.blade.php.

Fuente (Antonio Córdova, 2016)

Esta vista se encuentra en el directorio: views > templates > emails.

Se propone el siguiente correo para confirmación de canje:



Gráfico 3-66 Correo de confirmación de canje.

Fuente (Antonio Córdova, 2016)

3.3.9.2. Resultados

Una vez confirmado los dos pasos (confirmar productos y confirmar datos), la pantalla final es la felicitación al cliente por el canje realizado, esta pantalla se muestra una vez que el sistema haya enviado el correo electrónico al cliente.



Gráfico 3-67 Finalización de proceso de canje.
Fuente (Antonio Córdova, 2016)

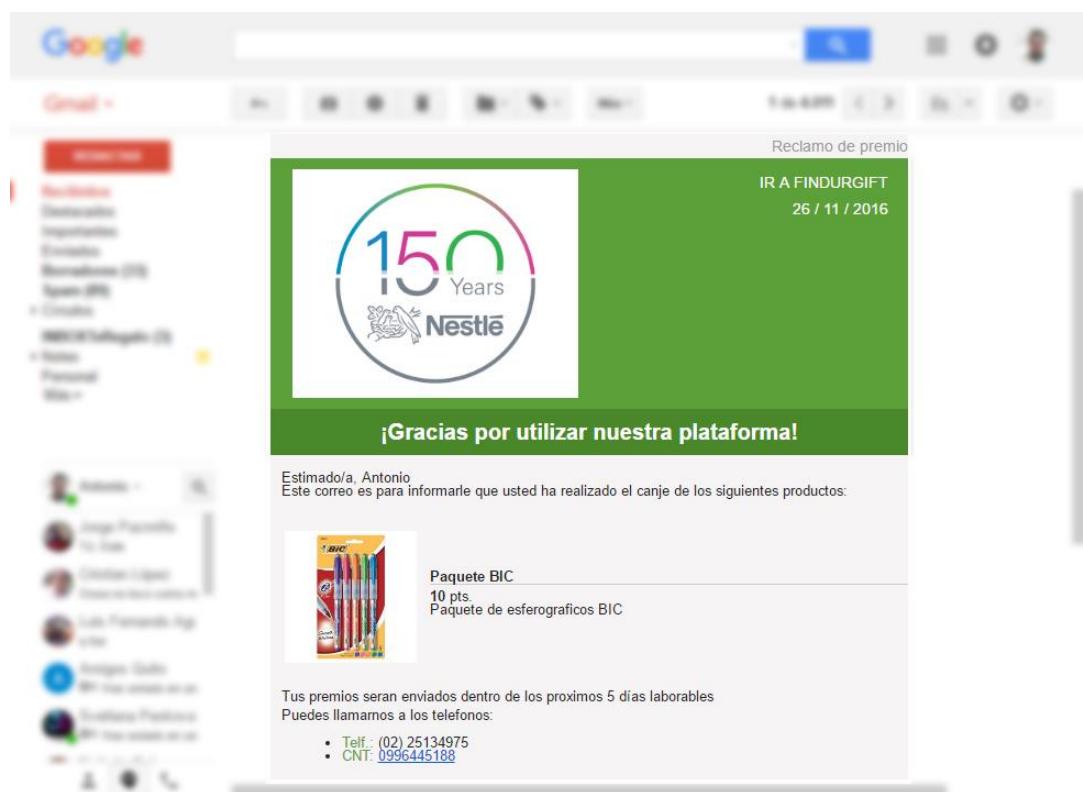


Gráfico 3-68 Email de comprobación
Fuente (Antonio Córdova, 2016)

3.3.9.3. Pruebas de aceptación

En esta primera parte de la iteración 07 se concluye el proceso de canje a través del envío de correo electrónico al cliente y al administrativo, así como el mensaje final al cliente cuando el sistema almacena los datos necesarios.

IT.	TAREAS	TIEMPO PLANIFICADO	COMIENZO	FIN
7	Culminación de proceso de canje	8:00:00	1/2/2016	1/2/2016
	Implementación de envío de correos electrónicos a usuario y administrador	4:00:00	1/2/2016	1/2/2016
	Pruebas de aceptación	4:00:00	1/2/2016	1/2/2016

Tabla 3-53 Pruebas de aceptación - culminación de canje - iteración 07.

Fuente (Antonio Córdova, 2016)

PRUEBA ACEPTACIÓN 08	CULMINACIÓN DE PROCESO DE CANJE	CUMPLIMIENTO DE RESULTADOS ESPERADOS	APROBADO POR TRACKER	APROBADO POR CLIENTE
RESULTADOS ESPERADOS	El sistema debe enviar un correo al administrador y un correo al usuario confirmando su canje.	X	X	X
RESULTADO FINAL				

Tabla 3-54 Prueba de aceptación 08.

Fuente (Antonio Córdova, 2016)

3.3.10. Iteración 07 - Zona Administrativa

Se requiere la zona administrativa, es decir, la zona que controla absolutamente todo el sistema y que permita crear, editar, entre otras opciones más, administrar a:

- Códigos
- Paquetes
- Empresas
- Reportes
- Productos

A lo largo de las iteraciones anteriores, se han construido partes de esta administración como, por ejemplo, la administración de productos en la iteración 02, por lo que en la presente iteración se realizará el inicio de sesión y la integración de las iteraciones anteriores que tengan que ver con la administración general, de manera que la navegabilidad sea adecuada al usuario administrador y se presente todas las opciones administrativas.

IT.	TAREAS	TIEMPO PLANIFICADO	PROPIETARIO
7	Zona administrativa	32:00:00	Antonio Córdova
	Diseño de Experiencia de usuario para administrador	18:00:00	
	Implementación de ingreso al sistema a través usuario y contraseña	8:00:00	
	Pruebas de aceptación	6:00:00	

Tabla 3-55 Iteración 07 – Culminación proceso canje.

Fuente (Antonio Córdova, 2016)

3.3.10.1. Diseño de Experiencia de usuario para administrador

Se requiere de un formulario básico para inicio de sesión del usuario administrador quien controlará todo el sistema, para ello se requiere de una tabla básica para registrar mediante usuario (correo) y una clave que permita el control de autenticación al sistema.

El link de acceso se encontrará en el pie de página del sistema, tanto para clientes, empresas, así como para visitantes en general del sitio.

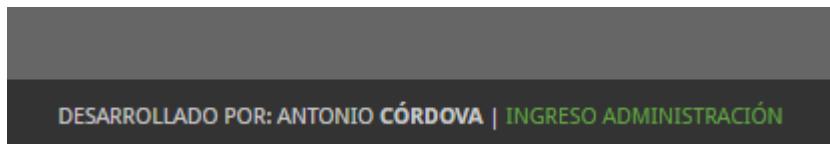


Gráfico 3-69 Link de inicio de sesión.

Fuente (Antonio Córdova, 2016)

Se plantea el siguiente formulario construido en base a la plantilla de visitante del sitio, el cual tendrá dos campos de ingreso de texto, el primero servirá para escribir el usuario a través de su correo electrónico, y el segundo, para ingresar la contraseña. Por último, se utiliza un casillero para recordar la contraseña.

Find Ur Gift | Encuentra tu premio

Compra tus tarjetas de regalo desde 9.99

Portada Nosotros Catálogo Servicio Empresas Contacto info@findurgift.com 25 13 49 75

INICIO DE SESIÓN ADMINISTRACIÓN

Correo Electrónico: example@gmail.com

Contraseña: clave de acceso

Recordarme

Ingresar Olvidó su contraseña?

CONTÁCTANOS

Sed ut perspiciatis unde omnis iste natus error sit voluptatem tam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi explicabo. Donec ut voluptat metus. Aliquam tortor lorem, Sed ut perspiciatis unde omnis iste natus error sit oluptatem

Nombre
Apellido
Correo Electrónico

Mensaje

Enviar

DESARROLLADO POR: ANTONIO CÓRDOVA | INGRESO ADMINISTRACIÓN

Gráfico 3-70 Maqueta de inicio de sesión – administradores
Fuente (Antonio Córdova, 2016)

3.3.10.2. Implementación de ingreso al sistema - Administrador

En la iteración 02 se planteó la estructura de la tabla de administradores, la cual se la utilizará para el inicio de sesión. Se requiere trabajar en las tres áreas principales Modelo, Vista y Controlador.

Modelo

El modelo de esta tabla es bien simple y únicamente se requiere principalmente de dos campos, correo y clave. Al ser un proyecto académico, se realizó la implementación del inicio de sesión sin administración de usuario administrador.

ADMINISTRADORES			
admId	<pi,ai>	Serial	<M>
admNombre		Variable characters (255)	<M>
admCorreo		Variable characters (100)	<M>
admClave		Variable characters (80)	<M>
ADMID2	<ai>		
admId	<pi>		

Gráfico 3-71 Tabla Administradores
Fuente (Antonio Córdova, 2016)

El modelo creado es Administradores.php el cual se conecta a la tabla que lleva el mismo nombre.

Controlador

Se crea el archivo AuthAdmin.php localizado en app > Http > Controllers > Admin. Los métodos básicos son:

- **doLogin**, Utilizado para recibir los datos del usuario (correo y clave), chequear en la tabla si existe el usuario con su respectiva contraseña y redirigir en caso de éxito. Se define como página inicial una vez que el usuario haya iniciado sesión a la lista de productos.
- **Logout**, utilizado para terminar y borrar los datos de sesión en el sistema.

Vista

Se crea la vista para iniciar sesión, utilizando la plantilla de visitantes.

Gráfico 3-72 Formulario de inicio de sesión – administradores.
Fuente (Antonio Córdova, 2016)

Finalmente se crea la navegabilidad a través del menú superior en la vista, mediante las rutas creadas en el archivo routes.php mostrado en el gráfico 3-73. El usuario administrador podrá navegar entre las distintas zonas productos, empresas, paquetes, entre otros.

The screenshot shows the PhpStorm IDE interface with the routes.php file open. The code defines several route groups for administration, including authentication, products, and company management. The code is well-commented in Spanish.

```

routes.php
/*
 * Rutas para Administración
 * Contiene rutas para proceso de login, administración de empresas,
 * paquetes, productos y gestión de pedidos
 */
Route::group(['middleware' => ['web']], function () {
    Route::group(['prefix' => 'admin'], function () {
        //AUTENTICACIÓN
        Route::get('/login', 'Admin\AuthAdmin@login');
        Route::post('/login/process', 'Admin\AuthAdmin@doLogin');
        Route::get('/logout', 'Admin\AuthAdmin@logout');

        //PRODUCTOS
        Route::group(['prefix' => 'productos'], function () {
            Route::get('/', 'Admin\ProductosController@index');
            Route::get('/add', 'Admin\ProductosController@register');
            Route::post('/addType', 'Admin\ProductosController@registerType');
            Route::post('/addZone', 'Admin\ProductosController@registerZone');
            Route::post('/process/{tipo}', 'Admin\ProductosController@registerProcess');
            Route::post('/addImage', 'Admin\ProductosController@registerImage');
            Route::get('/edit/{id}', 'Admin\ProductosController@registerEdit');
            Route::get('/delete/{id}', 'Admin\ProductosController@registerDelete');
        });
    });
});

```

Gráfico 3-73 Rutas de administración.

Fuente (Antonio Córdova, 2016)

3.3.10.3. Resultados

La navegabilidad entre secciones de la administración es evidente a través del menú de opciones superior, desde el cual se puede acceder a productos, empresas, paquetes, reportes y salir del sistema.

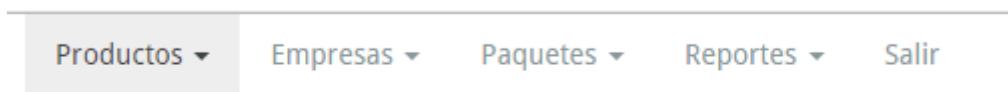


Gráfico 3-74 Menú de opciones administración

El inicio de sesión se encuentra activo y consulta a la base de datos

INICIO DE SESIÓN ADMINISTRACIÓN

The login form fields are:

- Correo Electrónico: mcastillo@nestle.com
- Contraseña: (redacted)
- Recordarme
-

Gráfico 3-75 Inicio de sesión para administrador.

Fuente (Antonio Córdova, 2016)

3.3.10.4. Pruebas de aceptación

La iteración 07 concluye con la implementación del inicio de sesión del administrador, así como su navegabilidad.

IT.	TAREAS	TIEMPO PLANIFICADO	COMIENZO	FIN
7	Zona administrativa	32:00:00	2/2/2016	5/2/2016
	Diseño de Experiencia de usuario para administrador	18:00:00	2/2/2016	4/2/2016
	Implementación de ingreso al sistema a través usuario y contraseña	8:00:00	4/2/2016	5/2/2016
	Pruebas de aceptación	6:00:00	5/2/2016	5/2/2016

Tabla 3-56 Pruebas de aceptación - Zona administrativa - iteración 07.

Fuente (Antonio Córdova, 2016)

PRUEBA ACEPTACIÓN 09	ZONA ADMINISTRATIVA	CUMPLIMIENTO DE RESULTADOS ESPERADOS	APROBADO POR TRACKER	APROBADO POR CLIENTE
RESULTADOS ESPERADOS	El sistema debe presentar todas las opciones administrativas	X	X	X
RESULTADO FINAL				

Tabla 3-57 Prueba de aceptación 09.

Fuente (Antonio Córdova, 2016)

3.3.11. Iteración 08 - Creación de reporte

En la última iteración se generará un reporte en archivo XLS el cual contendrá el reporte de productos canjeados, quien realizó el canje y finalmente la dirección de entrega de producto.

IT.	TAREAS	TIEMPO PLANIFICADO	PROPIETARIO
8	Creación de reportes	40:00:00	Antonio Córdova
	Diseño de documento Excel de reporte	14:00:00	
	Implementación de reporte	20:00:00	
	Pruebas de aceptación	6:00:00	

Tabla 3-58 Iteración 08 – Creación de reporte.

Fuente (Antonio Córdova, 2016)

3.3.11.1. Diseño de documento Excel de reporte

Se requiere de datos específicos en el reporte, los cuales son los siguientes:

- Empresa auspiciante
- Identificador de usuario
- Código con el cual se realizó el canje
- Que producto se escogió

- Fecha en la cual fue realizado el pedido

El diseño es básico ya que es un archivo xls, por lo que se requiere únicamente de una cabecera con fecha de exportación del reporte y el título “**FindUrGift**”

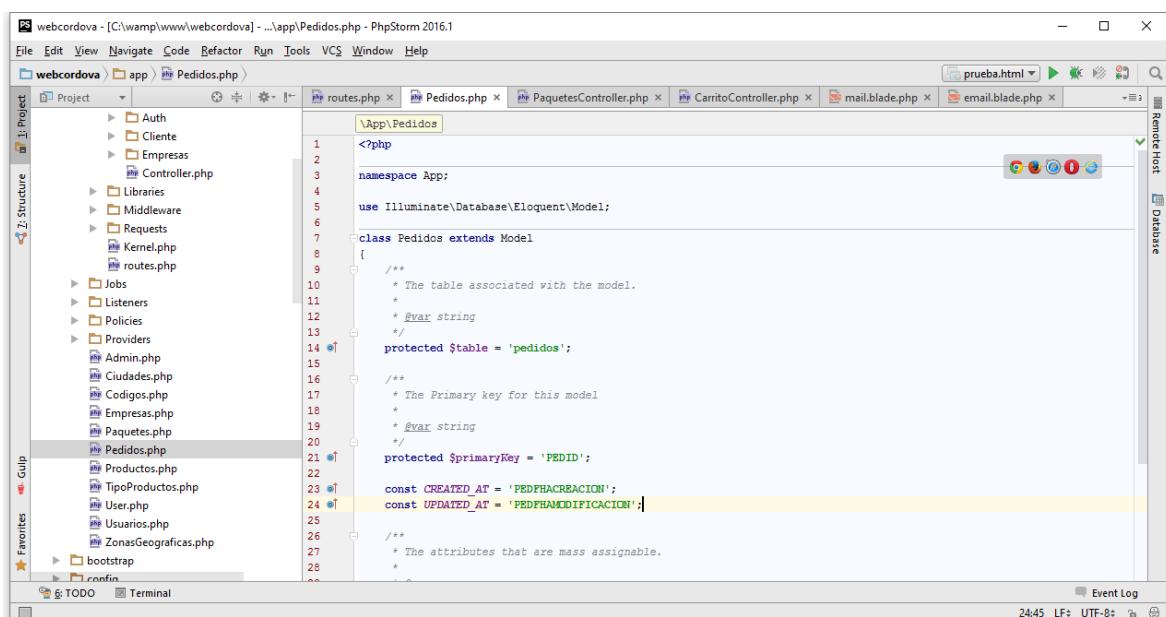
En la siguiente línea aparece el encabezado de cada columna y finalmente los registros de todos los pedidos registrados en la base de datos.

3.3.11.2. Implementación del reporte

La implementación de este reporte requiere de dos capas, el modelo y el controlador ya que no se necesita ningún filtro en especial.

Modelo

El archivo Pedidos.php almacena la información de la tabla de pedidos y establece los campos requeridos.



```

    <?php
namespace App;

use Illuminate\Database\Eloquent\Model;

class Pedidos extends Model
{
    /**
     * The table associated with the model.
     *
     * @var string
     */
    protected $table = 'pedidos';

    /**
     * The Primary key for this model
     *
     * @var string
     */
    protected $primaryKey = 'PEDIDID';

    const CREATED_AT = 'PEDFICCREACION';
    const UPDATED_AT = 'PEDFAMIFICACION';

    /**
     * The attributes that are mass assignable.
     *
     */
}

```

Gráfico 3-76 Modelo Pedidos.php.
Fuente (Antonio Córdova, 2016)

Controlador

Se ha creado un archivo ReportesController.php en app > Http > Controllers > Admin el cual contiene un único método llamado pedidos, en donde se realiza la consulta nativa de Laravel “all” para obtener todos los registros de la tabla de pedidos, todo este proceso se realiza a través del modelo creado anteriormente.

Finalmente se construye una tabla básica HTML que será codificado para que sea exportado a XLS es decir un formato legible para usuarios que trabajan con tablas.

```

<?php
namespace App\Http\Controllers\Admin;

use App\Empresas;
use App\Http\Requests;
use App\Http\Controllers\Controller;

class ReportsController extends Controller
{
    public function pedidos()
    {
        $pedidos = Pedido::all();
        $html = "<html>
            <table border='1'>
                <tr bgcolor=#a9a9a9>
                    <td align='right'><font face='Haettenschweiler' size=50 color='white'>Te Regalo</font></td>
                </tr>
                <tr>
                    <td align='right'>Reporte:</td>
                    <td align='left'>" . utf8_decode('Reporte de pedidos') . "</td>
                </tr>
                <tr>
                    <td align='right'>Fecha:</td>
                    <td align='left'>" . date('d/m/Y') . "</td>
                </tr>
            </table>";
    }
}

```

Gráfico 3-77 Controlador ReportsController.php.

Fuente (Antonio Córdova, 2016)

3.3.11.3. Pruebas de aceptación

El resultado final es obtener un reporte de los canjes realizados, en versión XLS, es decir un archivo físico descargado de navegador.

IT.	TAREAS	TIEMPO PLANIFICADO	COMIENZO	FIN
8	Creación de reportes	40:00:00	8/2/2016	12/2/2016
	Diseño de documento Excel de reporte	14:00:00	8/2/2016	9/2/2016
	Implementación de reporte	20:00:00	9/2/2016	12/2/2016
	Pruebas de aceptación	6:00:00	12/2/2016	12/2/2016

Tabla 3-59 Pruebas de aceptación – Creación de reportes - iteración 08.

Fuente (Antonio Córdova, 2016)

PRUEBA ACEPTACIÓN 10	CREACIÓN DE REPORTES	CUMPLIMIENTO DE RESULTADOS ESPERADOS	APROBADO POR TRACKER	APROBADO POR CLIENTE
RESULTADOS ESPERADOS	El sistema debe exportar los datos requeridos en un archivo de Excel	X	X	X
RESULTADO FINAL				

Tabla 3-60 Prueba de aceptación 10.

Fuente (Antonio Córdova, 2016)

3.4. Fase de Mantenimiento

Se cumplió con todas las iteraciones planificadas, quedando fuera de ámbito a la administración de los pedidos por canje y a la compra en línea, los cuales quedarían para un futuro mantenimiento o actualización del sistema.

3.5. Fase Post-Mortem

El presente proyecto entra en fase de Post-Mortem debido a que ya no se cuenta con más historias de usuario, dando como resultado el acta de cierre y entrega del proyecto.

CAPÍTULO 4

4. CONCLUSIONES Y RECOMENDACIONES

4.1. Conclusiones

De la Disertación

- La implementación de este sistema a través de códigos QR para el inicio de sesión, permite a usuarios que están en constante movimiento, el acceso a la información de manera ágil y cómoda desde su dispositivo móvil.
- Gracias a la planificación global del proyecto se pudo cumplir con el cronograma a pesar de la aparición de una nueva iteración.
- Realizar una investigación previa sobre los problemas propuestos por el cliente, asegura un mayor entendimiento del equipo a lo que se va a enfrentar, garantizando el correcto funcionamiento del sistema.
- La definición del proyecto a través de historias de usuario redactadas en el lenguaje del cliente, favoreció al equipo para encontrar la mejor solución posible al problema propuesto.
- Las historias de usuario permiten generar tareas puntuales las cuales se distribuyen en el equipo permitiendo así la priorización o jerarquización de las mismas, así también como desechar aquellas que no aportan a la resolución del problema.
- Definir claramente el alcance final permite al equipo organizarse de mejor manera en tiempos y calidad en el desarrollo de manera tal que exista armonía en la relación entre el cliente y el equipo XP.

- El IDE de desarrollo PHPStorm permitió al equipo el trabajo óptimo del sistema debido a sus diversas herramientas.
- El Framework Laravel, permitió la agilidad en el desarrollo del sistema mejorando la productividad del equipo, gracias a sus librerías.

4.2. Recomendaciones

- Cuando se realiza una lectura de un código QR, pueden aparecer varios problemas determinados por el tamaño del código o cambio de resolución, es decir cuando se pierde la claridad del código. Se recomienda seguir las normas establecidas en la (ISO/IEC 18004:2015, 2015)
- Para utilizar códigos QR en cualquier aplicación, se recomienda analizar librerías que puedan construir los códigos, de modo que la lógica del negocio se concentre en el resultado que ofrezca el código, una vez haya sido escaneado.
- XP, es una metodología ágil, pero puede tornarse difícil de manejar si no se sigue correctamente el proceso establecido, se recomienda no saltarse pasos.
- Es recomendable conocer el framework de desarrollo y sus componentes de manera que la programación sea fluida.
- Mantener reuniones periódicas, al inicio y durante el desarrollo del proyecto con el cliente para aclarar todas las dudas que se tengan, de tal forma que no haya ambigüedades entre el equipo.
- Comunicar al cliente algún problema en el proyecto, para encontrar de manera conjunta las posibles soluciones.

A la Facultad

- Se debería dictar cursos adicionales o talvez incluir dentro de alguna materia, Practicas con Linux y programación en Linux ya que es parte fundamental para el desarrollo web.
- Las clases con orientación administrativa están bien en la universidad, pero debería existir una rama adicional de emprendimiento para que los estudiantes puedan generar nuevas ideas de negocio, crear empleo, innovar el mercado, etc.

ANEXOS

1. ANEXO: MANUAL DE USUARIO

El presente documento está dirigido a los usuarios de la plataforma **FindUrGift**, el cual les permite la gestión de canjes de puntos por premios una vez que el usuario inicia sesión en el sistema.

1.1. Introducción

El sistema FindUrGift permite una fácil y rápida gestión de canje de sus puntos por cualquier artículo disponible en la plataforma. Es así que, les permite gestionar sus puntos, sus datos personales, datos de entrega y los productos a canjear.

Por otra parte, el acceso a la plataforma puede realizarse desde un dispositivo móvil, a través de un código QR que le haya sido entregado; o, a través de un código alfanumérico en el sitio web findUrGift.webcordova.com.

1.2. Conceptos importantes

FindUrGift es una aplicación web que puede ser accedido desde cualquier sistema operativo y cualquier navegador actualizado, así también desde un dispositivo móvil como un celular o una Tablet, el diseño se adapta a cualquier resolución.

Se puede acceder al sistema a través de un código QR que posea el usuario a través del dispositivo móvil, así como a través de un código alfanumérico en un computador.

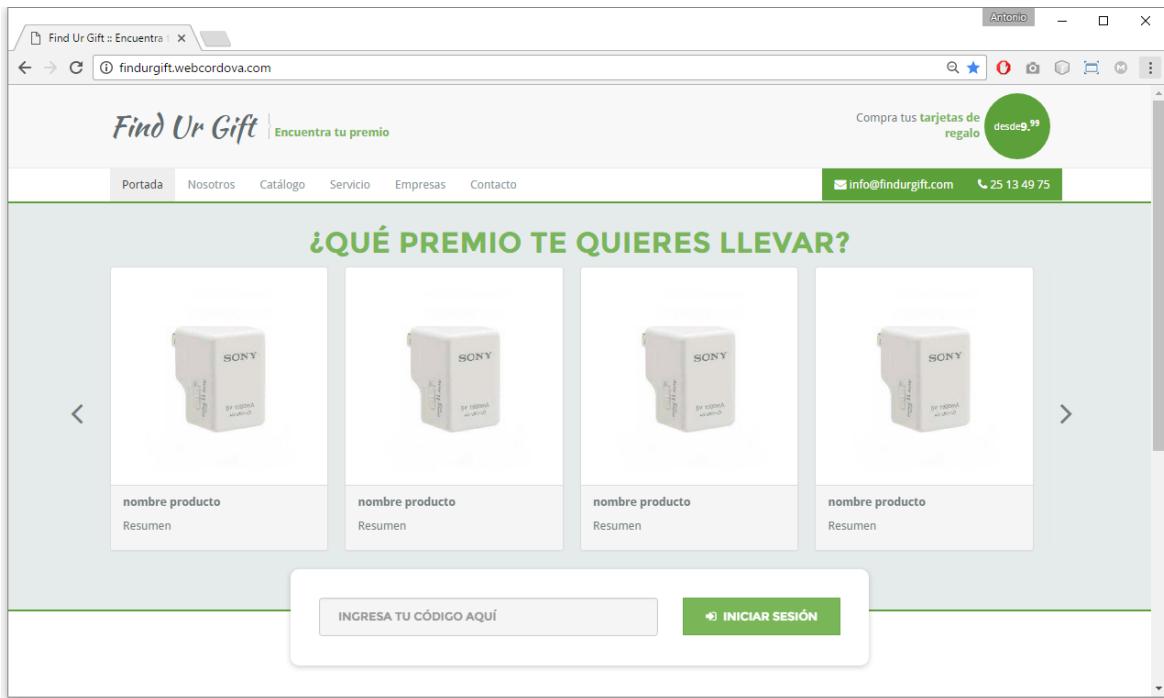


Gráfico 1-1 Ingreso al sistema – computador.
Fuente (Antonio Córdova, 2016)



Gráfico 1-2 Ingreso al sistema - iPhone5.
Fuente (Antonio Córdova, 2016)

En caso de utilizar un navegador, el usuario debe ingresar el código alfanumérico y a continuación presionar el botón iniciar sesión.

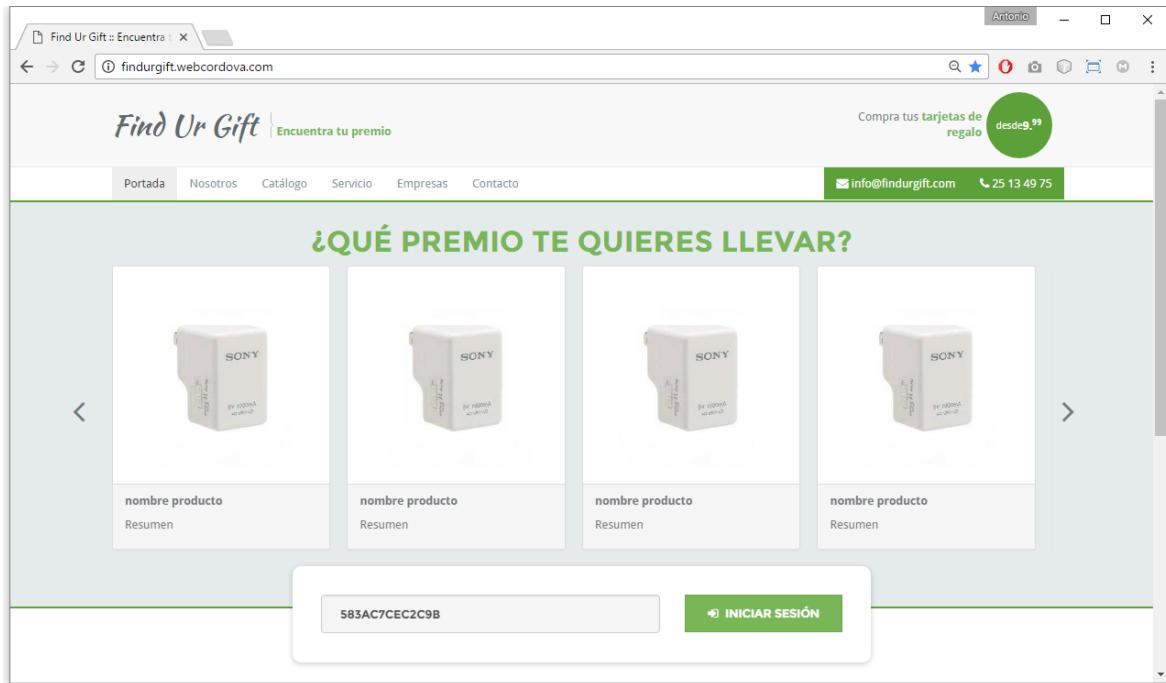


Gráfico 1-3 Ingreso de código.
Fuente (Antonio Córdova, 2016)

Si los datos ingresados son incorrectos se presentan dos escenarios posibles:

- Si no existe el código escrito: el mensaje será “código incorrecto”
- Si el código ya se utilizó y no le quedan más puntos: el código será “Ya no tiene más puntos para participar”

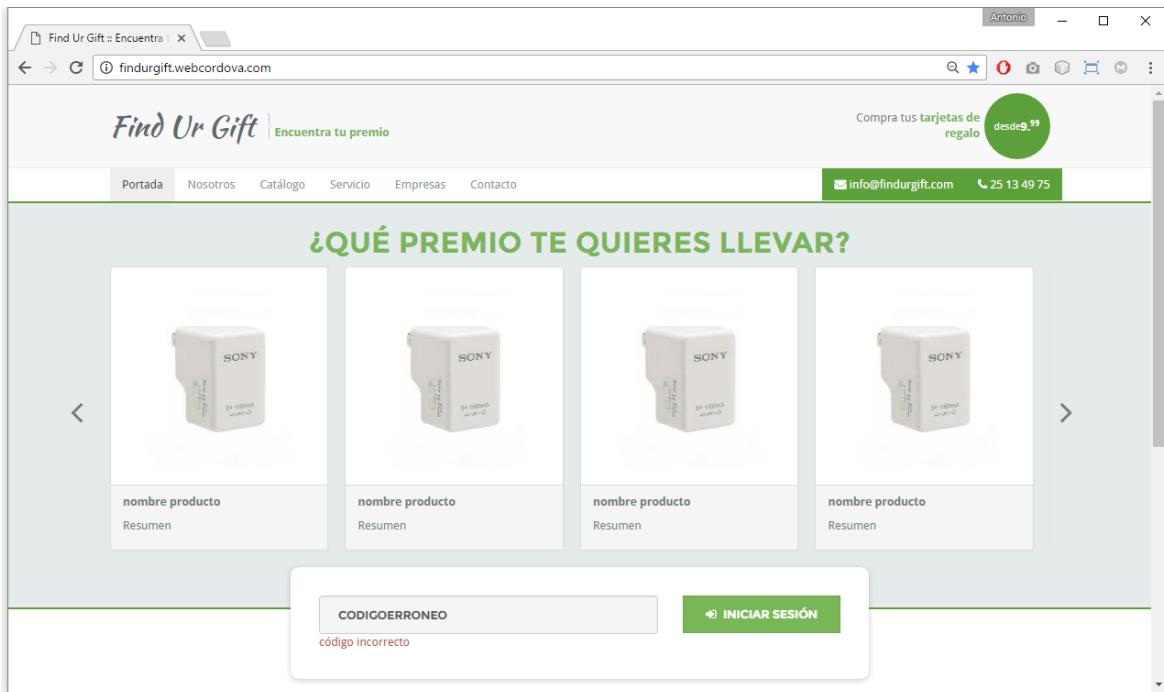


Gráfico 1-4 Inicio de sesión erróneo.
Fuente (Antonio Córdova, 2016)

Si el código ingresado es correcto, se visualizan los productos disponibles para el canje y otras opciones más.

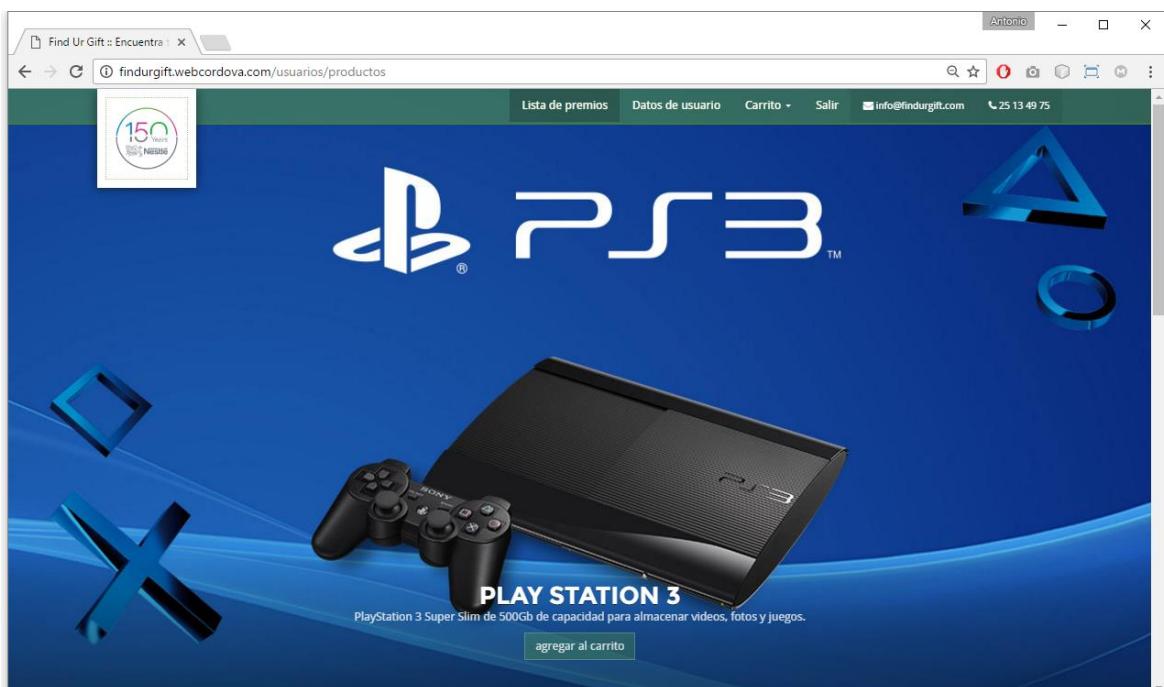


Gráfico 1-5 Portada principal usuarios.
Fuente (Antonio Córdova, 2016)

Se presentan varias categorías:

- Cabecera con menú principal

Los ítems son Lista de premios disponibles, sección de datos de usuario, carrito de canje donde van todos los productos seleccionados; y, un botón para salir. Adicionalmente datos de información de la empresa.

- Producto destacado

Esta zona está visible si existe un producto destacado, caso contrario no se presenta al usuario.

- Categorías

Se visualiza 3 zonas, para hombres, para mujeres y para niños

- Productos disponibles para el canje

Zona donde se despliega todos los productos disponibles para el canje, sin categorización

- Formulario de contacto

En caso que el usuario lo requiera, puede escribir en el formulario

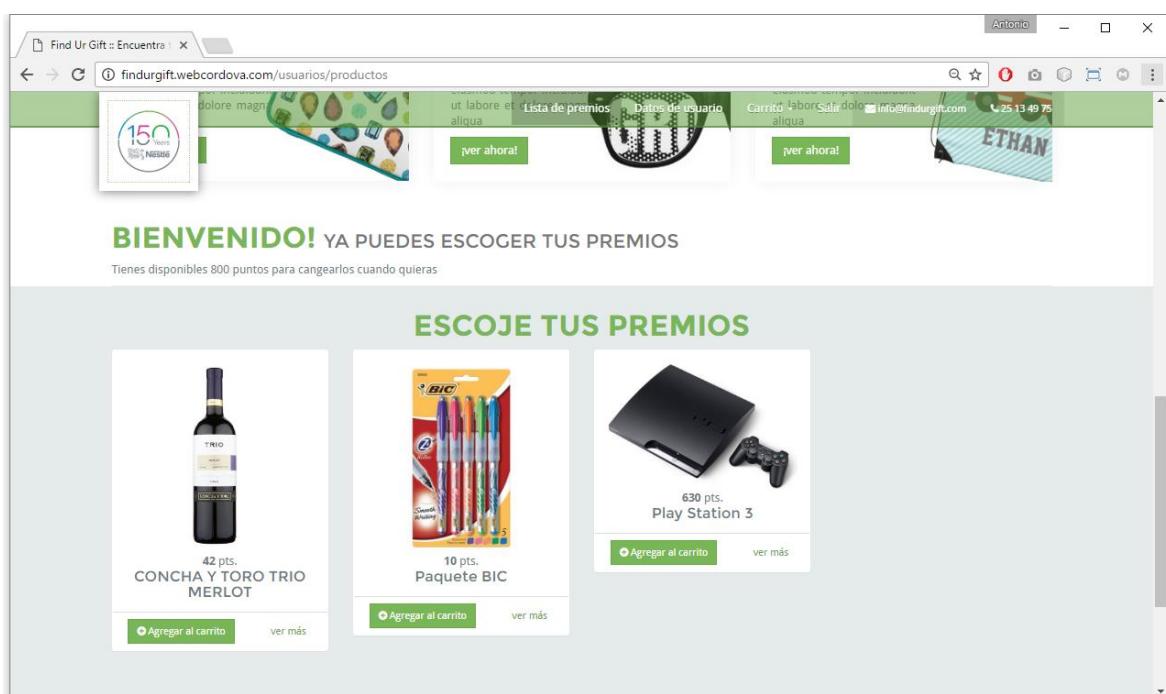


Gráfico 1-6 Productos disponibles para el canje.
Fuente (Antonio Córdova, 2016)

1.3. Funcionalidades

FindUrGift tiene las siguientes funcionalidades

- Gestión de productos: permite visualizar los productos disponibles para el canje
- Gestión de usuario: permite actualizar o agregar información relacionada con el cliente.
- Gestión de carrito de productos: permite agregar o quitar productos del carrito, así como iniciar el proceso de canje.

1.4. Proceso de uso

1.4.1. Selección de producto/s

Se puede seleccionar cualquier producto visible en el sistema para el usuario, se debe hacer clic en Agregar al carrito, en la parte inferior de cada producto. Una vez, se haya agregado el producto al carrito, el botón cambia de color de verde a rojo, también su ícono y el texto: Quitar, es decir, mediante el mismo botón se puede agregar al carrito o quitar del carrito. A continuación, se ilustra mediante un ejemplo.



Gráfico 1-7 Botón: Agregar al carrito
Fuente (Antonio Córdova, 2016)



Gráfico 1-8 Botón: Quitar
Fuente (Antonio Córdova, 2016)

Por otra parte, el producto agregado se visualiza en la parte superior cuando se hace clic en el menú principal, en el ítem carrito.



Gráfico 1-9 Visualización de productos agregados al carrito

Fuente (Antonio Córdova, 2016)

Una vez finalizado el proceso de selección de productos, se hace clic en finalizar canje ubicado en el menú principal dentro del ítem Carrito, con ello se inicia el proceso de canje.

1.4.2. Proceso de canje

El proceso de canje tiene 3 pasos básicos y necesarios

- 1. Confirmar Productos:** Se despliega todos los productos seleccionados para realizar el canje.

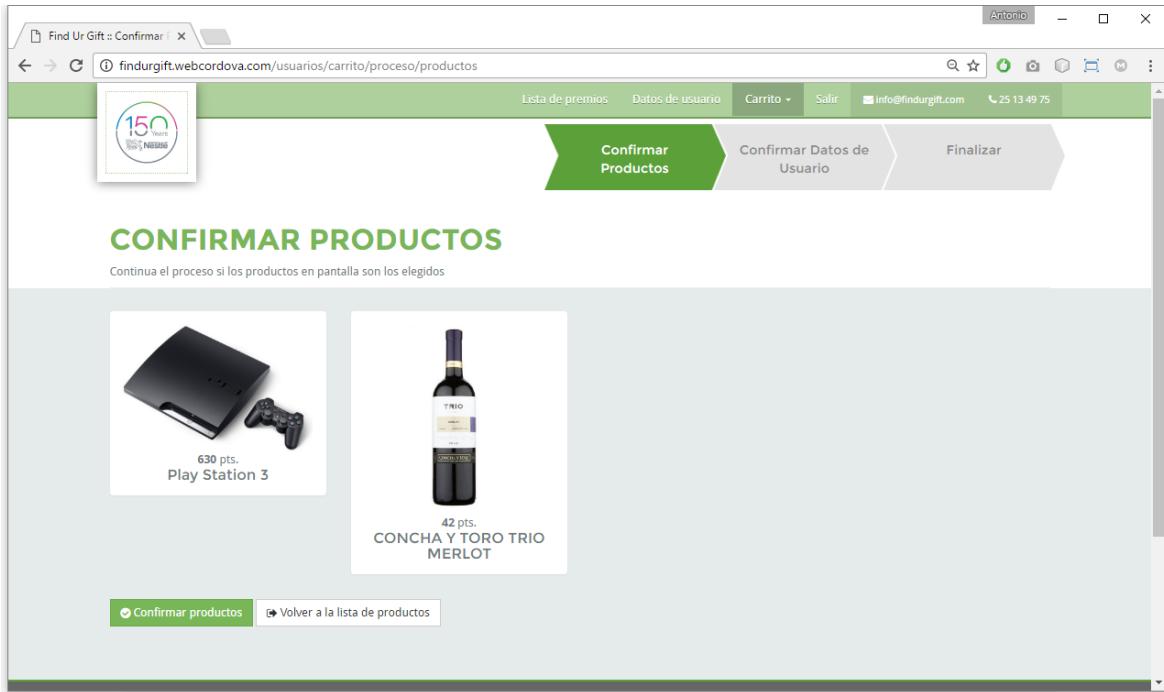


Gráfico 1-10 Confirmar productos.
Fuente (Antonio Córdova, 2016)

Es obligatorio hacer clic en confirmar productos para avanzar en el proceso, caso contrario se debe hacer clic en Volver a la lista de productos para quitar los productos o seleccionar otros.

- 2. Confirmar datos de usuario:** Si el usuario hizo clic en confirmar productos llega a esta sección en la cual se deben ingresar todos los datos de manera obligatoria. Finalmente se guarda la información haciendo clic en Guardar.

CONFIRMAR DATOS DE CLIENTE

Continua el proceso una vez hayas ingresado todos los datos siguientes

Nombres: Gustavo
Apellidos: Velasquez
Email: gvelasquez@sanjuanito.com
Teléfono Fijo: 2660763
Teléfono Móvil: 0994539773

DATOS DIRECCIÓN DE ENTREGA

Dirección de entrega: Shyris E15-102 y Portugal
Ciudad de entrega: Quito

Guardar **Regresar a productos**

Gráfico 1-11 Confirmar Datos.
Fuente (Antonio Córdova, 2016)

- Finalizar: el sistema procesa la información y realiza un envío al correo registrado la confirmación de los productos escogidos.

FELICIDADES

Se ha enviado un correo electrónico a tu cuenta con los datos de productos seleccionados y datos de contacto en caso requieras contactarnos.

Recuerda que tu premio será enviado dentro de los próximos 5 días laborables.

Ver más premios

CONTÁCTANOS

Sed ut perspiciatis unde omnis iste natus error sit voluptatem rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi explicabo. Donec ut voluptatum metus. Aliquam tortor lorem, Sed ut perspiciatis unde omnis iste natus error sit oluptatem

Nombre:
Apellido:
Correo Electrónico:

Mensaje:

Enviar

Gráfico 1-12 Fin Proceso de canje
Fuente (Antonio Córdova, 2016)

Finalmente, le aparece un mensaje de felicitación por el canje realizado, posteriormente el usuario recibirá un correo con el detalle de los productos canjeados.

2. ANEXO: MANUAL DE EMPRESA

El presente documento está dirigido a los usuarios empresariales de la plataforma ***FindUrGift***, es decir, quienes administran los paquetes adquiridos, la selección de los productos por paquete y la administración de sus correspondientes datos.

2.1. Introducción

Un usuario empresarial es quien adquiere paquetes con un cierto número de códigos, cuando la empresa adquiere un paquete, tiene acceso a la plataforma de manera que pueda controlar los productos disponibles en cada paquete adquirido.

El proceso de selección de productos ha sido creado pensando en el usuario final, de manera que, la selección se facilite a través de varios mecanismos tecnológicos como un filtro de categorías y selección rápida.

2.2. Conceptos importantes

FindUrGift puede ser administrado desde un dispositivo móvil ya que cuenta con diseño adaptable a cualquier dispositivo.

Para el acceso a la plataforma – zona empresarial – El personal administrativo le otorgará un usuario y contraseña, obligatoriamente el usuario es el correo electrónico proporcionado por el cliente, la contraseña inicial se la generará de manera aleatoria, por lo que es potestad del usuario cambiarla una vez haya ingresado al sistema.

Para acceder al sistema se debe hacer clic en “*Empresas*” ubicado en el menú principal

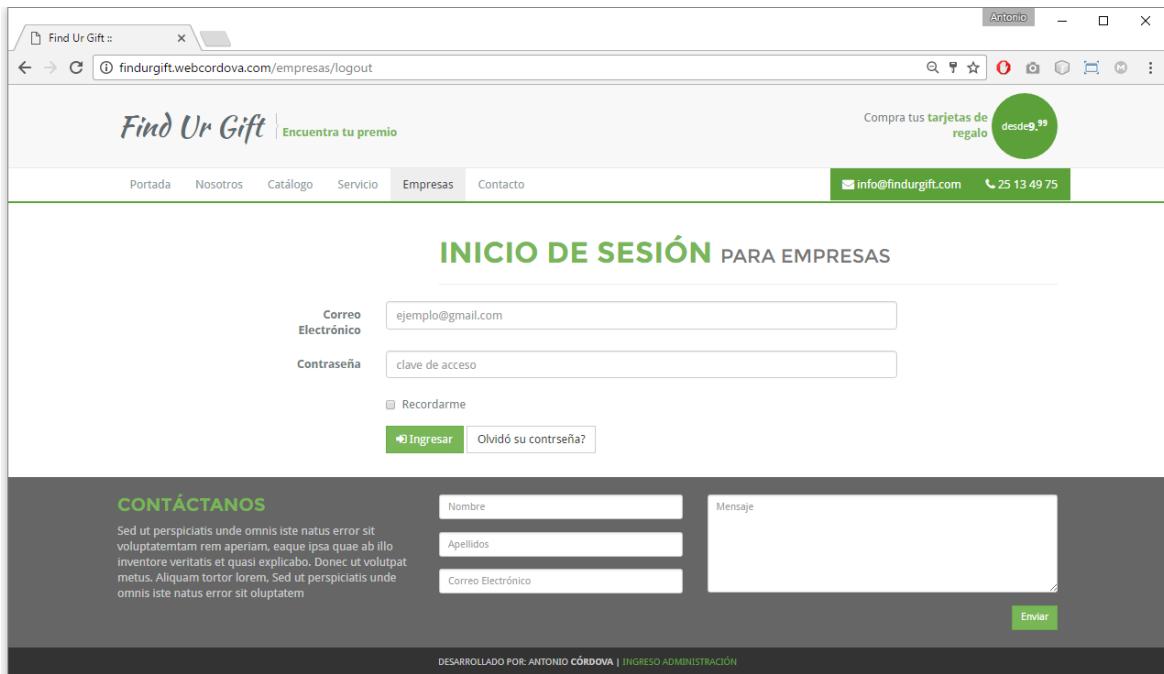


Gráfico 2-1 Inicio de sesión – Empresas.
Fuente (Antonio Córdova, 2016)

This is a mobile-optimized version of the login page. The layout is similar to the desktop version but adapted for a smaller screen. The 'INICIO DE SESIÓN PARA EMPRESAS' header is at the top. The 'Correo Electrónico' and 'Contraseña' fields are present, along with a 'Recordarme' checkbox and a green 'Ingresar' button. The 'CONTÁCTANOS' section below includes fields for 'Nombre', 'Apellidos', 'Correo Electrónico', and a 'Mensaje' area with a 'Enviar' button. The overall design is clean and modern.

Gráfico 2-2 Inicio de sesión – iPhone5.
Fuente (Antonio Córdova, 2016)

Cuando el usuario ingrese los datos puede ocurrir dos eventos, los datos ingresados son correctos o alguno de ellos puede estar equivocado, en este último caso, el mensaje que se le presenta al usuario es: **correo electrónico y/o contraseña incorrectos**

The screenshot shows a web browser window for 'Find Ur Gift' with the URL 'findurgift.webcordova.com/empresas/login'. The page title is 'INICIO DE SESIÓN PARA EMPRESAS'. A red error message box at the top contains the text 'correo electrónico y/o contraseña incorrectos'. Below it is a form with fields for 'Correo Electrónico' (example@gmail.com) and 'Contraseña' (clave de acceso). There is a 'Recordarme' checkbox, a green 'Ingresar' button, and a link 'Olvidó su contraseña?'. At the bottom, there is a 'CONTÁCTANOS' section with input fields for 'Nombre', 'Apellidos', and 'Correo Electrónico', and a 'Mensaje' text area with a green 'Enviar' button. The footer includes the text 'DESARROLLADO POR: ANTONIO CÓRDOVA | INGRESO ADMINISTRACIÓN'.

Gráfico 2-3 Inicio de sesión erróneo – empresas.
Fuente (Antonio Córdova, 2016)

Cuando se ingresó correctamente los datos, el sistema permite el ingreso y la administración de denominada **Zona Empresarial**. En ella, el usuario puede visualizar varias actividades:

- **Menú principal:** Navegación interna del sistema para usuarios empresariales, en el cual podrá acceder de manera directa a las opciones programadas.
- **Resumen de datos empresariales:** Información de la empresa
- **Paquetes adquiridos:** Visualización de los paquetes adquiridos con la opción de selección de productos en cada paquete.

A continuación, la pantalla inicial, una vez se ingresó los datos correctamente del usuario empresarial:

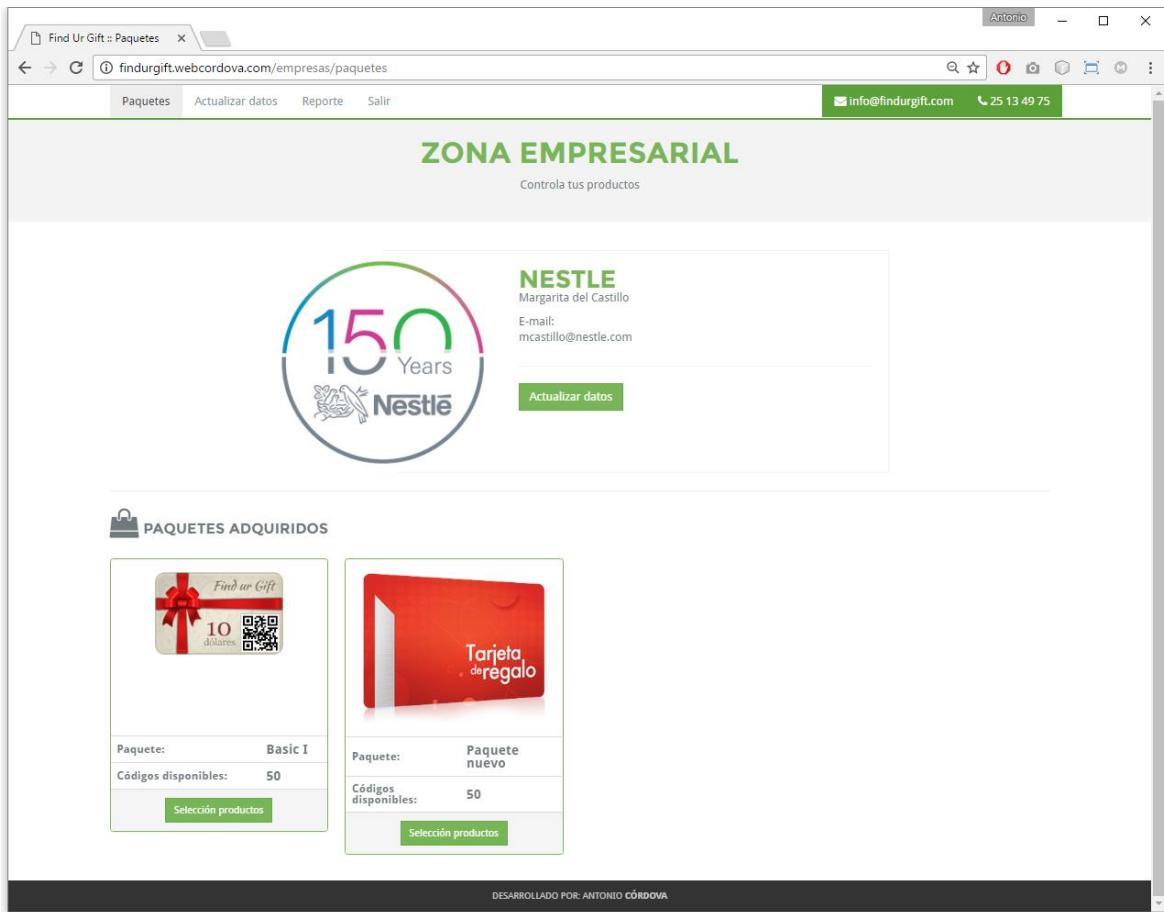


Gráfico 2-4 Zona empresarial.
Fuente (Antonio Córdova, 2016)

2.3. Proceso de uso

Un usuario empresarial es quien adquiere paquetes con un cierto número de códigos, los cuales serán entregados a sus clientes de manera que puedan canjear por productos disponibles en la plataforma, para ello, la empresa deberá seleccionar los productos que crea necesario.

2.3.1. Selección de productos

Para realizar este proceso se debe hacer clic en “**seleccionar productos**”. El sistema muestra una nueva pantalla donde se encuentran todos los productos activos, los cuales pueden ser filtrados.

En esta pantalla se muestran todos los productos activos que pueden ser seleccionados para mostrar a los clientes que tengan el código que, a su vez, pertenezca al paquete seleccionado.

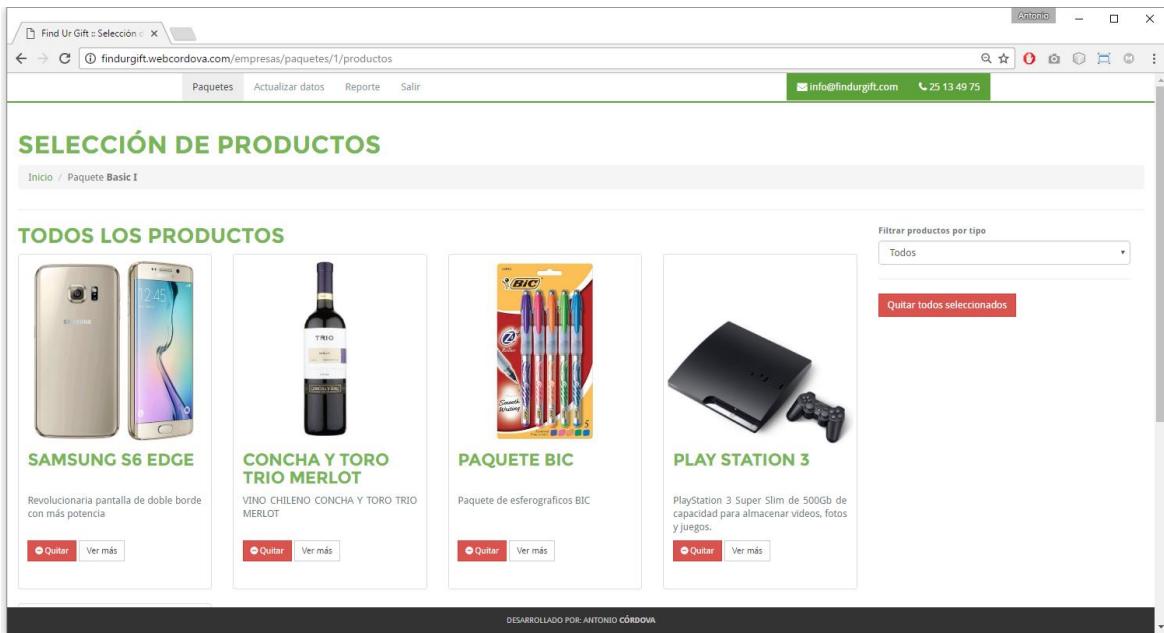


Gráfico 2-5 Selección de productos.
Fuente (Antonio Córdova, 2016)

Basta con hacer clic en seleccionar para que el producto sea activado; y, por el contrario, el botón quitar lo desactiva. Estos botones intercambian su funcionamiento dependiendo si están seleccionados o no, de forma que pueden



Gráfico 2-6 Opciones sobre producto.
Fuente (Antonio Córdova, 2016)

Para filtrar los productos por categorías disponibles se debe seleccionar la categoría de su preferencia en el combo de selección que está ubicado al costado derecho, el cual, estará en esta posición constante a pesar que se desplace en los productos, de manera que siempre este visible.

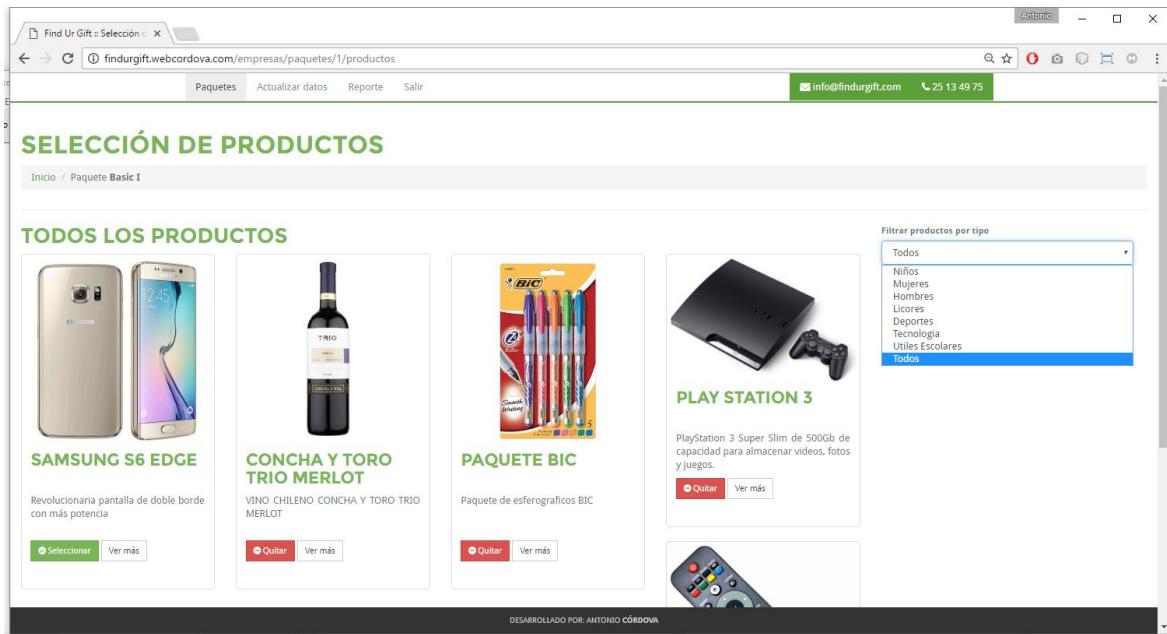


Gráfico 2-7 Combo filtro de productos por tipo.

Fuente (Antonio Córdova, 2016)

Finalmente, es molesto quitar los productos haciendo clic uno por uno, por lo que es de gran ayuda el botón ubicado en la parte inferior del filtro de productos por categorías llamado ***“Quitar todos los seleccionados”*** para hacerlo todo de un solo clic.

2.3.2. Actualización de datos

El usuario puede actualizar sus datos informativos haciendo clic en la pestaña superior *Actualizar Datos*, ubicado en el menú principal, aquí podrá cambiar los siguientes datos:

- Nombre de la empresa
- Nombre del representante
- Correo electrónico
- Contraseña
- Logotipo empresarial

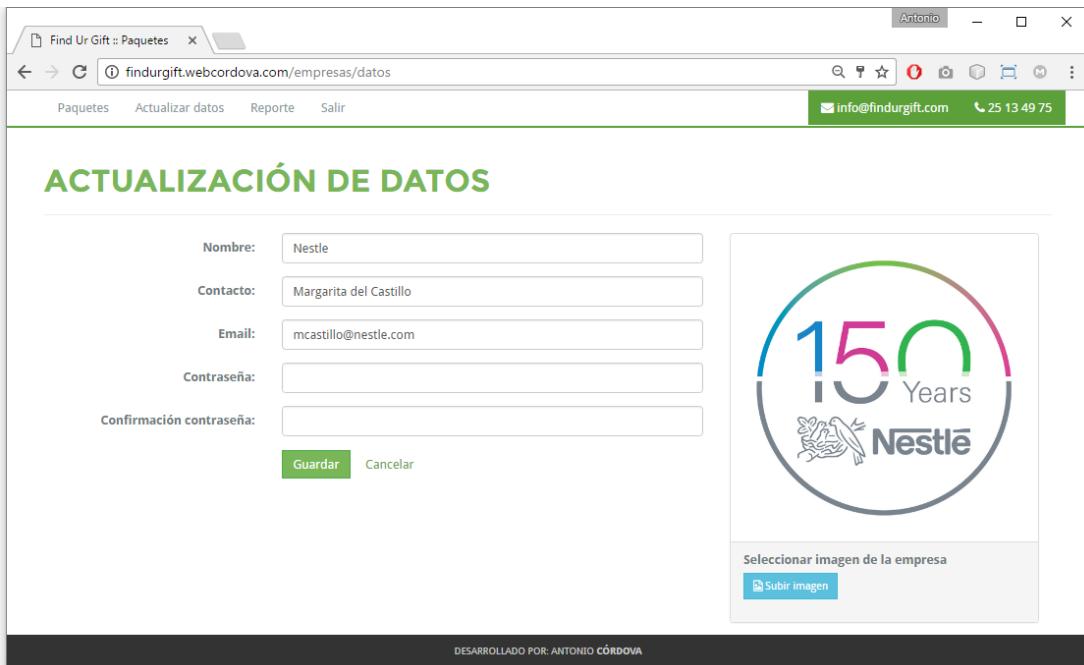


Gráfico 2-8 Actualización de datos – empresa.

Fuente (Antonio Córdova, 2016)

Es importante aclarar que el logotipo que se coloque en esta sección será la que aparece en la sección de usuarios, en otras palabras, los usuarios que sean portadores del código para realizar canjes, cuando ingresen al sistema verán este logotipo.

Todos los campos son obligatorios, a excepción de la contraseña, si existe un campo incompleto, aparece el mensaje **Campo obligatorio**

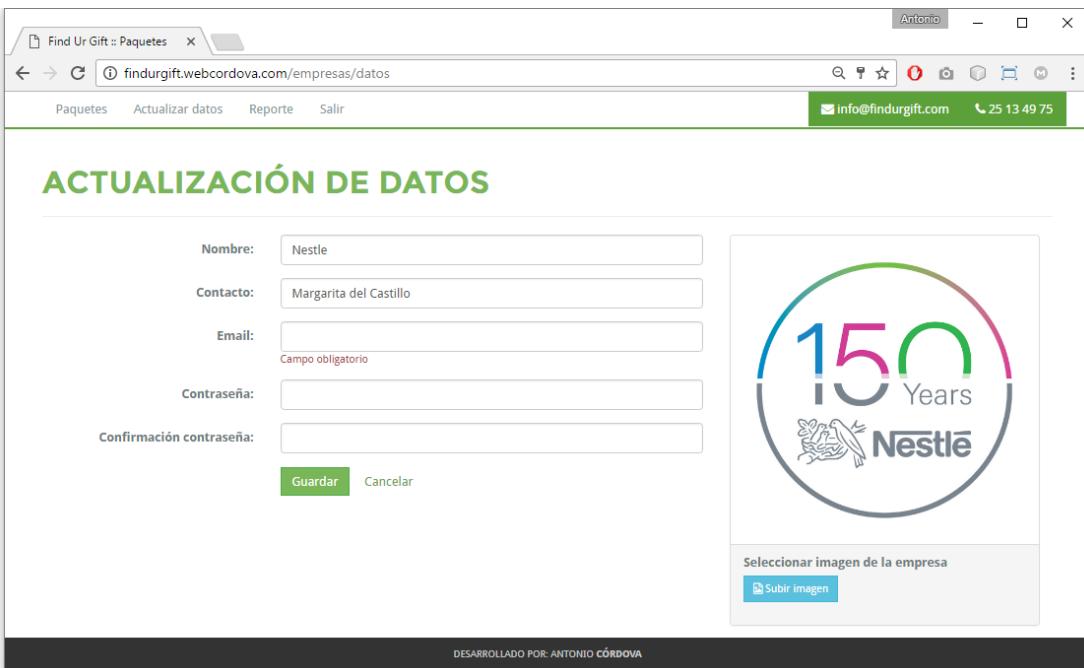


Gráfico 2-9 Error en actualización de datos.

Fuente (Antonio Córdova, 2016)

3. ANEXO: MANUAL DE ADMINISTRACIÓN

El presente documento está dirigido a los usuarios administradores de la plataforma **FindUrGift**, es decir, al director general quien está a cargo de absolutamente toda la administración de la plataforma por lo que es el usuario principal o súper administrador.

3.1. Introducción

El súper administrador tiene la responsabilidad de administrar:

- Empresas
- Paquetes
- Productos

Adicional a los anteriores, puede:

- Asignar paquetes a cualquier empresa
- Ver los usuarios de cada empresa

3.2. Conceptos importantes

El acceso a la administración se encuentra en la parte inferior del sitio **FindUrGift**, allí se encuentra un enlace que redirige al formulario de acceso, en el cual se solicita un correo electrónico y la respectiva contraseña.

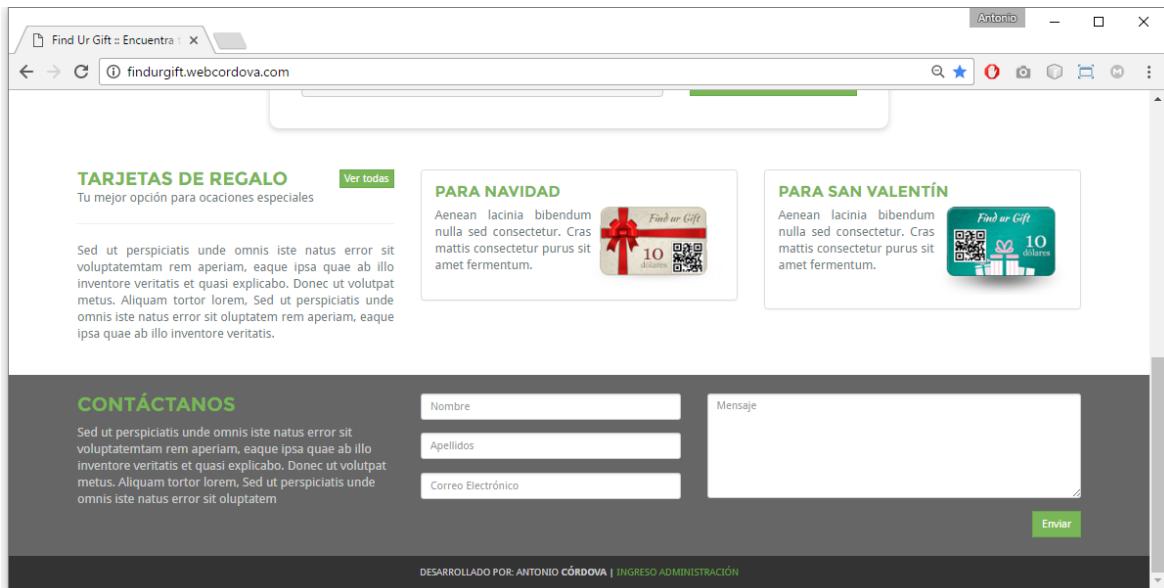


Gráfico 3-1 Link de acceso a la Administración.

Fuente (Antonio Córdova, 2016)

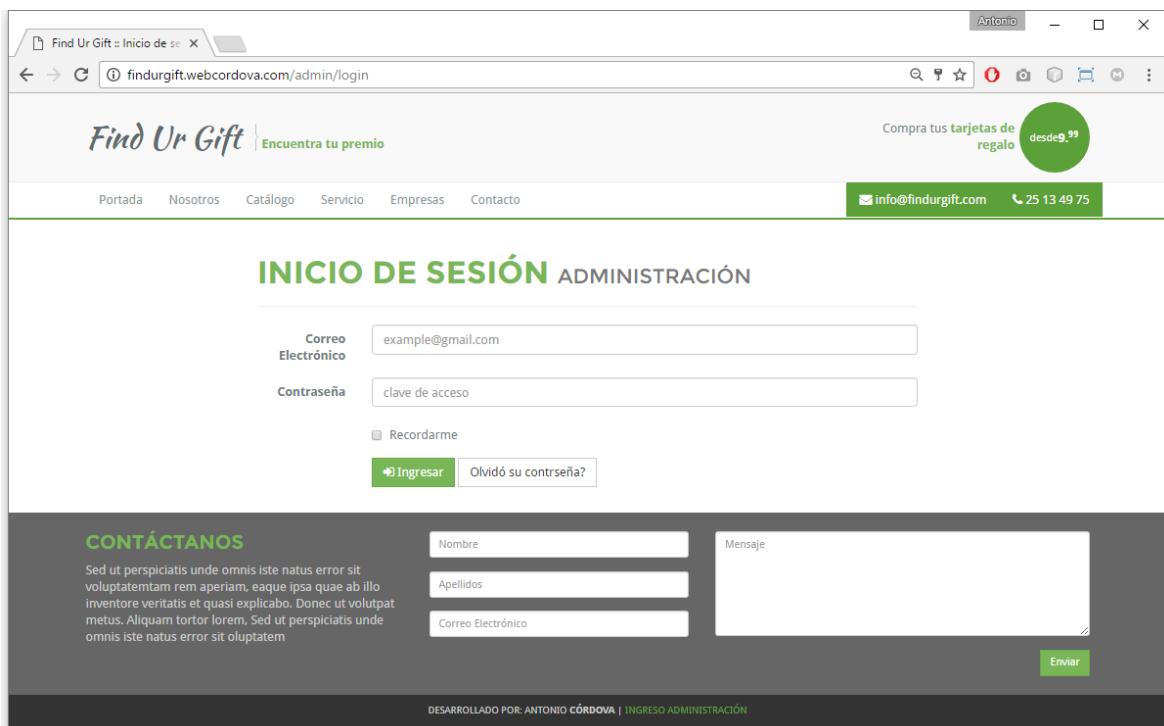


Gráfico 3-2 Inicio de sesión – Administración.

Fuente (Antonio Córdova, 2016)

Cuando el usuario concluya el proceso de inicio de sesión haciendo clic en Ingresar puede ocurrir dos eventos, los datos ingresados son correctos o alguno de ellos puede estar equivocado, en este último caso, el mensaje que se le presenta al usuario es: **correo electrónico y/o contraseña incorrectos**

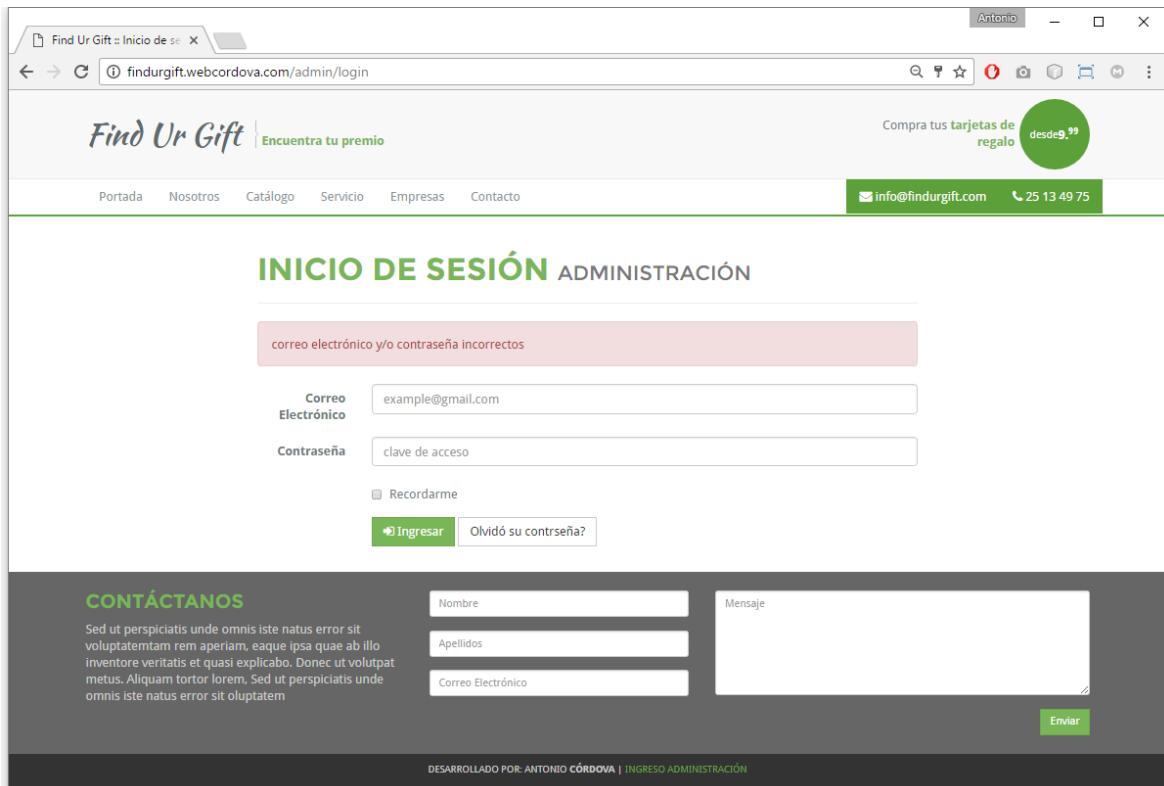


Gráfico 3-3 *Inicio de sesión erróneo – Administración.*
Fuente (Antonio Córdova, 2016)

3.3. Proceso de uso

La administración tiene varios procesos administrativos,

- Productos
- Empresas
- Paquetes

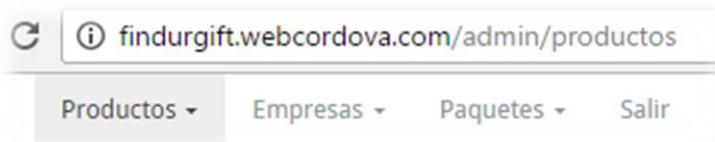


Gráfico 3-4 *Menú principal*

3.3.1. Gestión de Productos

En esta sección se administra los productos para las empresas, es decir, se puede crear, editar o listar todos los productos. Cuando el usuario haga clic en listar productos o una vez que haya iniciado sesión, la primera pantalla que se muestra es el listado de todos los productos.

Nombre	Imagen	Tipo	Zona	Compra Venta	Ganancia Radio	Fecha caducidad	Destacado Portada	Stock	Opciones
Samsung S6 Edge		Tecnología	nacional	850.00 980.00	130.00 1.15	no aplica		5	editar eliminar
CONCHA Y TORO TRIO MERLOT		Licores	nacional	17.75 30.00	12.25 1.69	no aplica		3	editar eliminar
Paquete BIC		Utiles Escolares	nacional	4.25 6.50	2.25 1.53	no aplica		14	editar eliminar
Play Station 3		Tecnología	Quito	380.00 450.00	70.00 1.18	no aplica		1	editar eliminar
Control Remoto		Tecnología	nacional	5.00 10.00	5.00 2.00	no aplica		10	editar eliminar

DESARROLLADO POR: ANTONIO CÓRDOVA

Gráfico 3-5 Lista de Productos.
Fuente (Antonio Córdova, 2016)

Como se puede ver, se encuentra toda la información relevante de los productos de tal forma que se pueda evaluar el estado, stock, ganancia y radio de cada uno de ellos.

El siguiente proceso será crear un producto.

3.3.1.1. Crear Producto

Para poder crear un producto se realiza los siguientes pasos:

1. Hacer clic en Productos (menú principal), luego,
2. Clic en Agregar nuevo
3. Llenar todos los campos obligatorios
4. Si un tipo de producto no existe en la lista, se puede agregar un tipo nuevo haciendo clic en Agregar Tipo

Nuevo tipo: ¿disponible?

Agregar Tipo

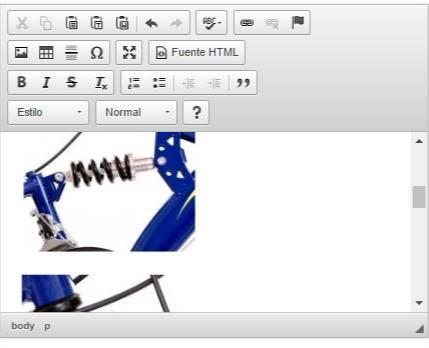
Gráfico 3-6 Agregar Tipo.
Fuente (Antonio Córdova, 2016)

5. Si una zona geográfica no existe en la lista, se puede agregar una zona nueva haciendo clic en Agregar zona
6. Clic en Guardar

Productos ▾ Empresas ▾ Paquetes ▾ Salir Find Ur Gift

AGREGAR NUEVO PRODUCTO

*campos obligatorios

Nombre*:	Bicicleta GTI	
Tipo*:	Diversión, deporte y accesorios	<input type="button" value="Agregar Tipo"/>
Zona geográfica*:	nacional	<input type="button" value="Agregar zona"/>
Resumen*:	Bicicleta Montañera 26 Gti Doble Suspensión 21 Velocidades, apta para montaña	
Descripción*:		
Precio de compra*:	124	
Precio de venta*:	175	
Fecha de caducidad:		
Stock:	5	
<input type="checkbox"/> ¿Producto en portada? <input type="checkbox"/>		
<input style="background-color: #4CAF50; color: white; padding: 5px 10px; border: none; border-radius: 5px; font-weight: bold; margin-right: 10px;" type="button" value="Guardar"/> <input style="border: 1px solid #4CAF50; color: #4CAF50; padding: 5px 10px; border-radius: 5px; font-weight: bold;" type="button" value="Cancelar"/>		

DESARROLLADO POR: ANTONIO CÓRDOVA

*Gráfico 3-7 Agregar nuevo Producto – ejemplo.
Fuente (Antonio Córdova, 2016)*

Concluido el proceso de creación, aparece el siguiente mensaje **Se agregó el producto Bicicleta GTI correctamente**, caso contrario el sistema presentará los errores en cada campo erróneo. Los errores controlados son los siguientes:

- Campo obligatorio: Significa que el usuario no ingreso ningún dato en este campo
- Campo debe ser una fecha válida: AAAA-MM-DD: significa que no se escribió la fecha en el formato especificado
- Campo debe ser un decimal con “.”: en los campos para ingresar datos numéricos, se ingresó un valor con coma en lugar de punto.
- Campo debe ser un número: Se presenta este error cuando se ingresa datos como texto en lugar de datos numéricos.

La imagen se sube independientemente del formulario, pero es indispensable realizarlo antes de guardar la información.

3.3.1.2. Opciones de Producto

Existen dos opciones básicas sobre un producto, editarlo o eliminarlo. Para llevar a cabo estas opciones se debe listar los productos haciendo clic en productos > lista

Destacado Portada	Stock	Opciones	
	5	editar	eliminar
	3	editar	eliminar

Gráfico 3-8 Opciones de producto.
Fuente (Antonio Córdova, 2016)

Editar Producto

Cuando se haya hecho clic en editar, nos aparece nuevamente el formulario para ingresar datos del producto, pero en esta ocasión, aparecen los campos llenos en donde ya se hayan ingresado datos.

The form includes the following fields:

- Nombre***: Bicicleta GTI
- Tipo***: selecciona uno
- Zona geográfica***: nacional
- Resumen***: Bicicleta Montañera 26 Gti Doble Suspensión 21 Velocidades, apta para montaña
- Descripción***: A rich text editor with various styling options like bold, italic, underline, and image insertion.
- Características**: A list of features: • Aro de aluminio 26". • 21 velocidades. • Frenos integrales.
- IMAGEN GENÉRICA***: Displays a blue mountain bike image.
- Selección de imagen de producto**: Buttons for "Subir imagen" (Upload image) and "Borrar" (Delete).
- IMAGEN DESTACADA**: Placeholder for the featured image.

Gráfico 3-9 Edición de producto.
Fuente (Antonio Córdova, 2016)

Finalmente, concluido los cambios en la edición del producto, se hace clic en guardar.

Una vez más los campos se validan de manera similar cuando se creó el producto:

- Campo obligatorio
- Campo debe ser una fecha válida
- Campo debe ser un decimal con “.”

- Campo debe ser un número

Eliminar Producto

Cuando se haya hecho clic en eliminar, aparece un pop-up o una ventana emergente con el título Eliminar producto y con la pregunta emergente si está de acuerdo con eliminar el producto.

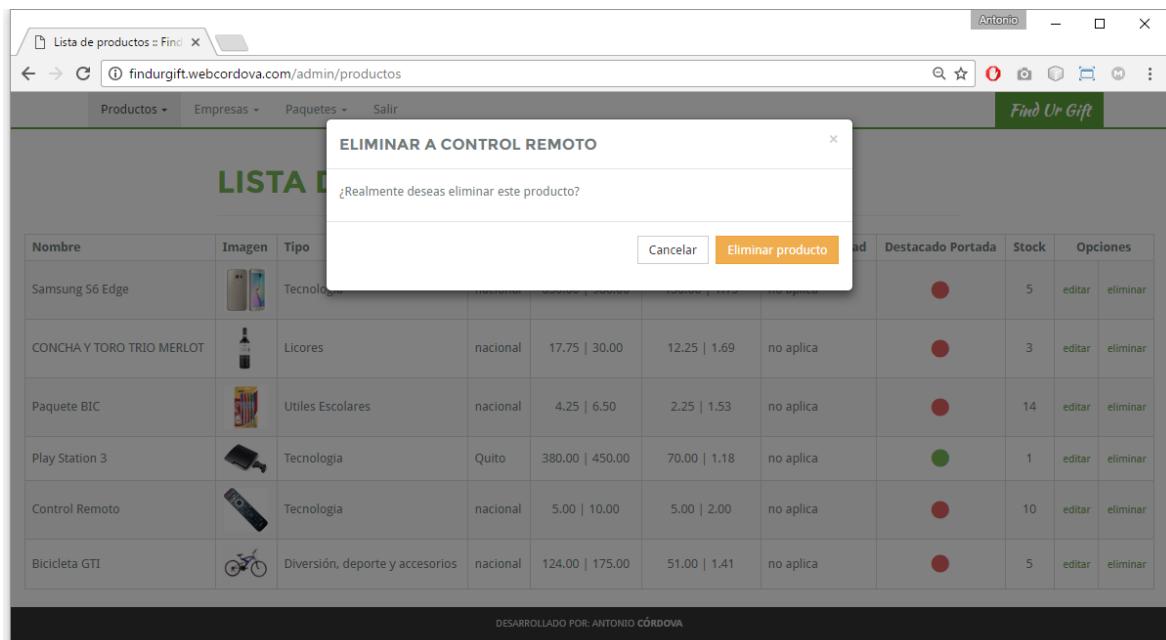


Gráfico 3-10 Pop-up eliminar producto.
Fuente (Antonio Córdova, 2016)

Al hacer clic en “*Eliminar producto*” el sistema borra el producto y finalmente envía un mensaje: Se ha eliminado al producto Control Remoto

Nombre	Imagen	Tipo	Zona	Compra Venta	Ganancia Radio	Fecha caducidad	Destacado Portada	Stock	Opciones
Samsung S6 Edge		Tecnología	nacional	850.00 980.00	130.00 1.15	no aplica		5	editar eliminar
CONCHA Y TORO TRIO MERLOT		Licores	nacional	17.75 30.00	12.25 1.69	no aplica		3	editar eliminar
Paquete BIC		Utiles Escolares	nacional	4.25 6.50	2.25 1.53	no aplica		14	editar eliminar
Play Station 3		Tecnología	Quito	380.00 450.00	70.00 1.18	no aplica		1	editar eliminar
Bicicleta GTI		Diversión, deporte y accesorios	nacional	124.00 175.00	51.00 1.41	no aplica		5	editar eliminar

DESARROLLADO POR: ANTONIO CÓRDOVA

Gráfico 3-11 Mensaje final de eliminación.

Fuente (Antonio Córdova, 2016)

3.3.2. Gestión de Paquetes

Un paquete es un conjunto de códigos, los cuales son utilizados para ingresar al sistema y realizar los canjes respectivos, los paquetes son una parte importante del sistema ya que, administrativamente, se puede generar paquetes de distinto costo y distinta cantidad de códigos, es decir, el administrador puede crear un paquete indicando el valor por código y cantidad de códigos requeridos.

La primera gestión es la visualización de los paquetes creados a través de una tabla que muestra la información relevante de los mismos.

NOMBRE	IMAGEN	CANTIDAD CÓDIGOS	COSTO POR CÓDIGO	PRECIO TOTAL	TASA CONVERSIÓN	ESTADO	Opciones
Basic I		50	10.00	500.00	1.2		editar eliminar
Paquete nuevo		4	800.00	3200.00	1.0		editar eliminar
Mi paquete		4	20.00	80.00	1.1		editar eliminar

DESARROLLADO POR: ANTONIO CÓRDOVA

Gráfico 3-12 Lista de paquetes

Fuente (Antonio Córdova, 2016)

3.3.2.1. Crear Paquete

Para crear un paquete nuevo el usuario tiene la posibilidad de hacerlo a través de los siguientes pasos:

1. Desde el botón azul Agregar Producto, o, desde el menú principal haciendo clic en Paquetes > Agregar nuevo.
2. El sistema despliega el formulario para completar la información necesaria.
3. Completar los datos requeridos
4. El precio total del paquete es calculado automáticamente cuando los dos campos anteriores (Cantidad de códigos y costo por código), han sido completados
5. Incluir una imagen del paquete
6. Clic en guardar para ingresar la información

Una vez realizado este proceso, el sistema valida la información escrita por el cliente, de encontrarse datos errores el sistema vuelve a mostrar el formulario con los comentarios respectivos en cada componente del formulario, los posibles errores a mostrarse son:

- **Campo obligatorio**, aquellos campos que tienen en el título (*), son obligatorios ser llenados.
- **Campo debe ser entero**, no se puede escribir texto, únicamente números sin coma.
- **Campo debe ser un decimal**, campo debe ser escrito como decimal, utilizando el punto para separar la parte entera del decimal.

The screenshot shows a web-based application for managing packages. At the top, there's a navigation bar with links for 'Productos', 'Empresas', 'Paquetes', and 'Salir'. The main title is 'AGREGAR NUEVO PAQUETE'. The form fields include:

- Nombre de paquete: DEMO I
- Cantidad de códigos: 5
- Costo por código: 500
- Precio Total: 2500
- Tasa de conversión: 1
- Opciones: ¿Paquete Disponible? ¿Destacado?
- Buttons: Guardar (Save) and Cancelar (Cancel)

On the right side, there are two separate sections for selecting package images:

- A preview area titled 'Seleccionar imagen del paquete' showing a blue gift box icon with a white ribbon.
- An upload area titled 'Seleccionar imagen del paquete' with a 'Subir imagen' (Upload image) button.

At the bottom of the page, a footer note reads: DESARROLLADO POR: ANTONIO CÓRDOVA

Gráfico 3-13 Agregar nuevo paquete.
Fuente (Antonio Córdova, 2016)

Cuando los datos están correctos, el sistema re-direcciona al usuario administrador al listado completo de paquetes, en el cual se muestra el nuevo paquete creado con el mensaje: **Se agregó el paquete DEMO I correctamente**

NOMBRE	IMAGEN	CANTIDAD CÓDIGOS	COSTO POR CÓDIGO	PRECIO TOTAL	TASA CONVERSIÓN	ESTADO	OPCIONES
Basic I		50	10.00	500.00	1.2		editar eliminar
Paquete nuevo		4	800.00	3200.00	1.0		editar eliminar
Mi paquete		4	20.00	80.00	1.1		editar eliminar
DEMO I		5	500.00	2500.00	1.0		editar eliminar

DESARROLLADO POR: ANTONIO CÓRDOVA

Gráfico 3-14 Paquete agregado correctamente.
Fuente (Antonio Córdova, 2016)

3.3.2.2. Opciones de Paquete

Existen dos opciones básicas sobre un paquete, editarlo o eliminarlo. Para llevar a cabo estas opciones se debe listar los productos haciendo clic en paquetes > lista

ESTADO	OPCIONES	
	editar	eliminar
	editar	eliminar

Gráfico 3-15 Opciones de paquete.
Fuente (Antonio Córdova, 2016)

Editar Paquete

Para editar un paquete, se debe hacer clic en editar ubicado en el extremo derecho de cada registro en la lista de paquetes. Una vez se haya hecho clic en editar, nos aparece nuevamente el formulario para ingresar datos del paquete, pero en esta ocasión, aparecen los campos llenos en donde ya se hayan ingresado datos.

EDICIÓN DE PAQUETE

The screenshot displays the 'Edit Package' form. It includes fields for 'Nombre de paquete' (DEMO I), 'Cantidad de códigos' (5), 'Costo por código' (500.00), 'Precio Total' (2500.00), and 'Tasa de conversión' (1.0). Under 'Opciones', there are checkboxes for '¿Paquete Disponible?' (checked) and '¿Destacado?' (checked). Below the form are two sections: 'Seleccionar imagen del paquete' showing a blue gift box icon, and 'Seleccionar imagen del paquete' with a 'Subir imagen' button.

Gráfico 3-16 Editar Paquete.
Fuente (Antonio Córdova, 2016)

Finalmente, concluido los cambios en la edición del paquete, se hace clic en guardar. Una vez más los campos se validan de manera similar cuando se creó el producto:

- Campo obligatorio
- Campo debe ser entero
- Campo debe ser un decimal

Eliminar Paquete

Para eliminar una empresa, se debe hacer clic en editar ubicado en el extremo derecho de cada registro en la lista de paquetes. Cuando se haya hecho clic en eliminar, aparece un pop-up o una ventana emergente con el título Eliminar paquete y con la pregunta emergente si está de acuerdo con eliminar el producto.

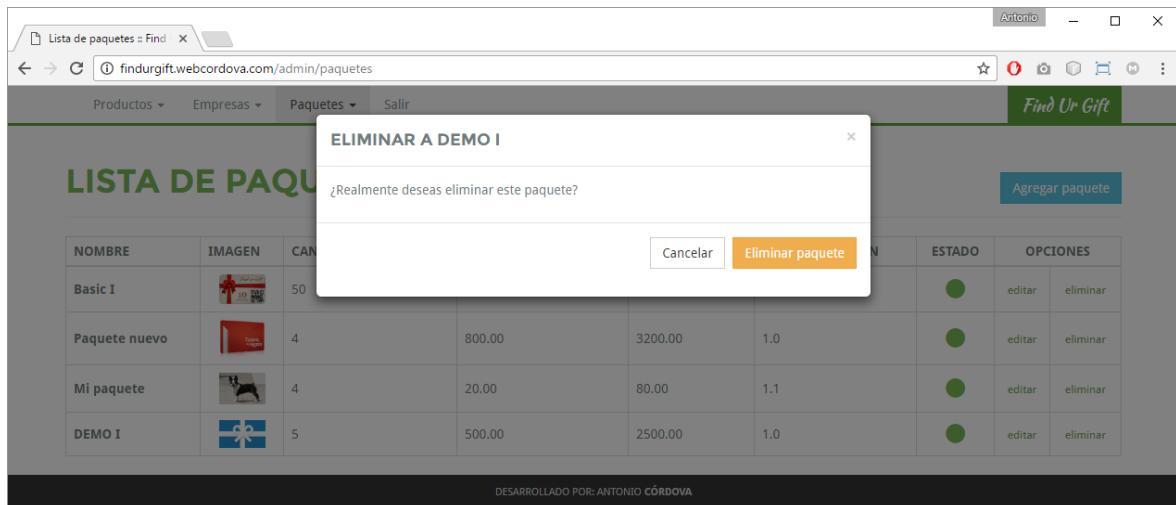


Gráfico 3-17 Eliminar paquete.
Fuente (Antonio Córdova, 2016)

Al hacer clic en “*Eliminar paquete*” el sistema borra el paquete y finalmente envía un mensaje: **Se ha eliminado al paquete DEMO I**

Nota: El sistema muestra un mensaje de alerta en caso de que alguna empresa ya haya adquirido el paquete, por lo que no se podrá borrar el paquete

3.3.3. Gestión de empresas

Las empresas cumplen un rol importante ya que a ellas se les asigna paquetes, que, a través de ellos, sus clientes pueden acceder a los productos para el respectivo canje. Es por ello que las empresas requieren información de usuario como correo electrónico y clave de acceso, primordiales para iniciar sesión en la plataforma. El usuario administrador tiene la obligación de crear estos datos iniciales para el usuario empresarial

Se puede visualizar todas las empresas creadas a través de una lista, para ello se debe hacer clic en Empresas > Lista, desde el menú principal.

	Nombre Empresa	Contacto	E-mail	Tasa costo	Paquetes	Usuarios	Opciones
	Pronaca	Jaime Herrera	jaime@pronaca.com	1.00			editar
	Nestle	Margarita del Castillo	mcastillo@nestle.com	1.40			editar

DESARROLLADO POR: ANTONIO CÓRDOVA

Gráfico 3-18 Lista de empresas.
Fuente (Antonio Córdova, 2016)

Esta tabla muestra, además de los datos de cada empresa, otras opciones importantes sobre las empresas como la asociación de paquetes y los usuarios que ya han realizado o están realizando canjes. Más adelante se profundizará estas opciones.

3.3.3.1. Crear Empresa

Para crear una empresa nueva el usuario administrativo tiene la posibilidad de hacerlo a través de los siguientes pasos:

1. Desde el botón azul Agregar empresa (cuando se visualice las empresas), o, desde el menú principal haciendo clic en Empresas > Crear.
2. El sistema despliega el formulario para completar la información necesaria.
3. Completar los datos requeridos
4. Incluir una imagen o logotipo de la empresa
5. Clic en guardar para ingresar la información

Una vez realizado este proceso, el sistema valida la información escrita por el cliente administrador, si existen errores, el sistema vuelve a mostrar el formulario con los comentarios respectivos en cada componente del formulario, los posibles errores a mostrarse son:

- **Campo requerido:** significa que es un campo obligatorio
- **Campo numérico:** El campo debe ser un decimal
- **Campo de email:** El campo debe ser correo electrónico válido
- **Contraseña muy corta:** La contraseña debe tener al menos 6 caracteres

AGREGAR NUEVA EMPRESA

*campos obligatorios

Nombre*:	Moderna Alimentos
Contacto*:	Jorge Luis Terán
Email*:	jteran@moderna.com
Contraseña*:
Tasa de puntos*:	1.2

Guardar Cancelar

Seleccionar imagen de la empresa

MODERNA ALIMENTOS
GANACLUB
El Club de Ganadores

Seleccionar imagen de la empresa

Subir imagen

DESARROLLADO POR: ANTONIO CÓRDOVA

Gráfico 3-19 Agregar nueva empresa.
Fuente (Antonio Córdova, 2016)

Cuando los datos están correctos, el sistema re-direcciona al usuario administrador al listado completo de empresas, en el cual se muestra la nueva empresa creada con el mensaje: **Se agregó la empresa Moderna Alimentos correctamente**

LISTA DE EMPRESAS

Agregar empresa

Se agregó la empresa Moderna Alimentos correctamente

	Nombre Empresa	Contacto	E-mail	Tasa costo	Paquetes	Usuarios	Opciones
	Pronaca	Jaime Herrera	jaime@pronaca.com	1.00			
	Nestle	Margarita del Castillo	mcastillo@nestle.com	1.40			
	Moderna Alimentos	Jorge Luis Terán	jteran@moderna.com	1.20			

DESARROLLADO POR: ANTONIO CÓRDOVA

Gráfico 3-20 Empresa creada exitosamente.
Fuente (Antonio Córdova, 2016)

3.3.3.2. Opciones de Empresa

Las empresas tienen varias opciones de configuración, la opción más básica es la edición de la empresa para actualizar sus datos; se puede también asociar paquetes a las empresas, es decir si una empresa adquiere un paquete, el usuario administrador tiene la facultad para asociar el paquete con la empresa que compro dicho paquete.

Paquetes		Usuarios	Opciones
			editar

Gráfico 3-21 Opciones de empresa.

Fuente (Antonio Córdova, 2016)

Editar Empresa

Para editar una empresa, se debe hacer clic en editar ubicado en el extremo derecho de cada registro en la lista de empresas. Una vez se haya hecho clic en editar, nos aparece nuevamente el formulario para ingresar datos de la empresa, pero en esta ocasión, aparecen los campos llenos en donde ya se hayan ingresado datos.

EDITAR EMPRESA

*campos obligatorios

Nombre*:

Contacto*:

Email*:

Contraseña*:

Confirmación contraseña:

Tasa de puntos*:

Seleccionar imagen de la empresa

El Club de Ganadores

Seleccionar imagen de la empresa

Gráfico 3-22 Editar empresa

Fuente (Antonio Córdova, 2016)

Finalmente, concluido los cambios en la edición de la empresa, se hace clic en guardar. Una vez más los campos se validan de manera similar cuando se creó el producto:

- Campo obligatorio
- Campo debe ser un decimal
- Confirmar contraseña (Opcional) en caso que se decida cambiar la clave.

Nota: No existe la opción eliminar empresa, debido a que cumple varias gestiones sobre paquetes y adicionalmente ver sus clientes (usuarios). La administración desea mantener

el histórico de empresas creadas de tal forma que a futuro decidan adquirir nuevos paquetes.

3.3.3.3. Opciones sobre paquetes

Existen dos acciones base sobre paquetes asociados a cada empresa, y estas son adquirir un nuevo paquete y ver los paquetes adquiridos.

Adquirir un nuevo paquete

Para asociar un paquete a una empresa se debe hacer clic en el icono más (+) que tiene cada registro en la lista de las empresas.

Paquetes	Usuarios	Opciones

Gráfico 3-23 Asociar paquete con empresa.
Fuente (Antonio Córdova, 2016)

Al hacer clic en este ícono, se muestra un pop-up con información sobre los paquetes disponibles con varias opciones como se muestra a continuación:

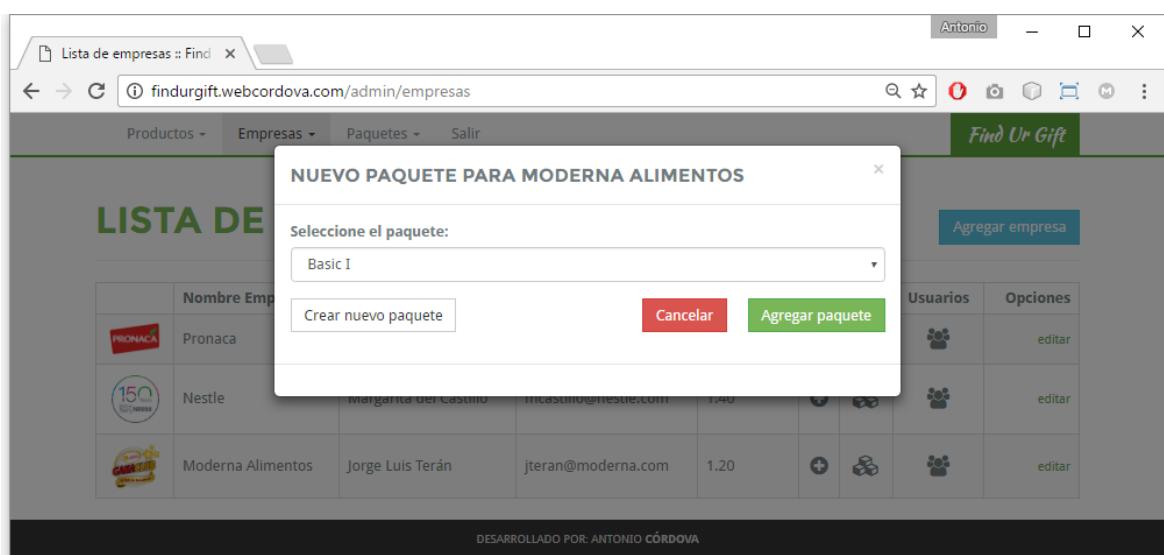


Gráfico 3-24 Nuevo paquete para empresa.
Fuente (Antonio Córdova, 2016)

Esta ventana emergente muestra una lista de paquetes activos para que el usuario administrador pueda seleccionar uno de tal forma que se pueda asociar el paquete, si no existe el paquete buscado, tiene la opción de crear un nuevo paquete (ver gestión de paquetes para crear uno nuevo).

Si el usuario hizo clic en agregar paquete, después de seleccionarlo, el sistema asocia el paquete con la empresa, así también genera los códigos respectivos para la empresa seleccionada y el paquete asociado.

Finalmente, el sistema muestra el mensaje siguiente: **Se agregó el paquete Mi paquete y sus códigos para la empresa Moderna Alimentos**

The screenshot shows a web browser window titled "Lista de empresas :: Find". The URL is "findurgift.webcordova.com/admin/empresas". The page has a green header bar with the text "Find Ur Gift" and a blue button "Agregar empresa". Below the header is a green banner with the message "Se agregó el paquete Mi paquete y sus códigos para la empresa Moderna Alimentos". The main content area is titled "LISTA DE EMPRESAS" and contains a table with three rows of company data:

	Nombre Empresa	Contacto	E-mail	Tasa costo	Paquetes	Usuarios	Opciones
	Pronaca	Jaime Herrera	jaime@pronaca.com	1.00			editar
	Nestle	Margarita del Castillo	mcastillo@nestle.com	1.40			editar
	Moderna Alimentos	Jorge Luis Terán	jteran@moderna.com	1.20			editar

At the bottom of the page, there is a dark footer bar with the text "DESARROLLADO POR: ANTONIO CÓRDOVA".

Gráfico 3-25 Asociación de paquete exitoso.
Fuente (Antonio Córdova, 2016)

Paquetes asociados

Para visualizar una lista de paquetes asociados se debe hacer clic en el ícono de los cubos que tiene cada registro en la lista de las empresas.

Paquetes	Usuarios	Opciones
(circulado en rojo)		

Gráfico 3-26 Opciones sobre paquetes - ver paquetes.

Fuente (Antonio Córdova, 2016)

A continuación, se visualiza la lista de los paquetes adquiridos, adicionalmente se cuenta con una nueva opción para ver los códigos generados a partir de la compra del paquete. En otras palabras, Cuando una empresa adquiere un paquete, el sistema automáticamente genera la cantidad de códigos indicados en el paquete para que la empresa pueda disponer de ellos.

Paquetes disponibles :: F

Antonio

Paquetes disponibles :: F

findurgift.webcordova.com/admin/empresas/3/paquetes

Productos Empresas Paquetes Salir

Find Ur Gift

PAQUETES DISPONIBLES PARA MODERNA ALIMENTOS

Empresas / Paquetes

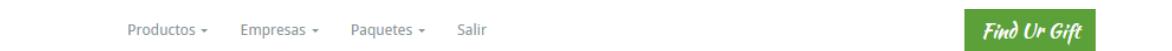
Paquete	Cantidad de Códigos	Códigos Disponibles	Estado	Códigos
Mi paquete	4	4		

DESARROLLADO POR: ANTONIO CÓRDOVA

Gráfico 3-27 Lista de paquetes asociados a empresas.

Fuente (Antonio Córdova, 2016)

Finalmente, podemos ver los códigos QR generados haciendo clic en el ícono ubicado al extremo derecho de cada registro de paquetes. El usuario puede exportar estos códigos a formato PDF para ser entregados al cliente empresarial.



CÓDIGOS GENERADOS

Empresas / Paquetes / Códigos

Código QR	Código HEX	Estado
	58839c973f117	●
	58839c9740b22	●
	58839c974182e	●
	58839c974252c	●

DESARROLLADO POR: ANTONIO CÓRDOVA

Gráfico 3-28 Códigos generados.
Fuente (Antonio Córdova, 2016)

3.3.3.4. Opciones sobre usuarios

Esta opción fue desarrollada para mostrar la lista de usuarios quienes ya han accedido a la plataforma a través del código QR y están canjeando productos. Para ver estos usuarios se debe hacer clic en el ícono de usuario en la lista de las empresas.

Paquetes	Usuarios	Opciones
		editar
		editar
		editar

Gráfico 3-29 Opciones sobre usuarios.
Fuente (Antonio Córdova, 2016)

Al hacer clic en este ícono, se muestra la sección de los clientes con su respectiva información.

The screenshot shows a web browser window titled "Usuarios registrados :: Find Ur Gift". The URL in the address bar is "findurgift.webcordova.com/admin/empresas/1/usuarios". The page header includes links for "Productos", "Empresas", "Paquetes", and "Salir", along with a "Find Ur Gift" logo. The main content area is titled "USUARIOS REGISTRADOS DE PRONACA" and displays a table of registered users:

Nombre	Apellidos	E-mail	Tlf. Fijo	Tlf. Movil	Ptos. Disponibles	Estado
ana	urgiles	amurgiles@puce.edu.ec	3461234	098456789	70	
					100	

At the bottom of the page, a dark bar contains the text "DESARROLLADO POR: ANTONIO CÓDOVA".

Gráfico 3-30 Usuarios registrados.
Fuente (Antonio Córdova, 2016)

4. ANEXO: MANUAL TÉCNICO

4.1. Introducción

El presente documento está dirigido al personal técnico de sistemas de **FindUrGift**, ya que contiene información pormenorizada sobre aspectos internos del sistema, configuración inicial, entre otros.

4.2. Requerimientos técnicos

Aunque el rendimiento del sistema depende del número de transacciones que los usuarios realicen simultáneamente, se puede establecer los siguientes requerimientos mínimos para el correcto funcionamiento del sistema en ambiente de producción:

Requerimientos mínimos de hardware

El servidor en el cual se vaya a instalar el sistema debe entregar las características de hardware que posee. Si el servidor adquirido tiene especificaciones de hardware inferiores a las descritas en esta sección, el sistema podría presentar un malfuncionamiento.

Estos son los requerimientos mínimos apropiados para que el sistema pueda levantarse:

- Procesador: 2-core
- Memoria RAM: 2GB
- Disco Duro: 5GB

Recomendación mínima de hardware

En ambiente de producción, la recomendación para hardware esta descrita de manera general, de hecho, el sistema puede requerir un mayor rendimiento debido a varios

factores, puede ser el incremento del número de usuarios, cantidad de productos ingresados, conexiones simultáneas, etc.

- Procesador: 64-bits
- CPU: 8-core, 2.0 GHz o mas
- RAM: 32 GB
- Espacio libre en disco: 50GB

Requerimientos mínimos de software

De la misma manera, el servidor en el cual se vaya a alojar el sistema, debe proporcionar la información del software que posee, o, si el servidor es nuevo, se requiere de Linux 4.4 o similar SO basado en Linux estable y actualizado

4.3. Herramientas utilizadas para el desarrollo

- **Servidor:** Apache 2.4
- **Lenguaje de Programación:** PHP 5.5.9 y sus librerías correspondientes
 - OpenSSL PHP Extension
 - PDO PHP Extension
 - Mbstring PHP Extension
 - Tokenizer PHP Extension
 - XML PHP Extensión
- **Motor de base de datos:** MySQL en versión libre, o MariaDB 10.0.20
- **Administrador de BD:** phpMyAdmin 4.3.7

4.4. Configuración del sistema

Inicialmente se debe preparar el ambiente de producción, es decir el sistema operativo debe estar instalado y adicionalmente debe ser instalado el servidor, base de datos y PHP para que el sistema pueda funcionar.

Algunos servidores o sitios de alojamiento ya ofrecen el servicio disponible al instante, es decir, no se requiere de la instalación de sistema operativo ni configuraciones adicionales. Generalmente cuando se adquiere un servidor dedicado o un servidor virtualizado se debe realizar esta primera instalación o configuración, más, sin embargo, sitios de alojamiento (hosting) cuentan ya con esta configuración inicial, incluso con un sistema de administración del sitio como cPanel.

Estos administradores de sitio tienen una sección para FTP, es decir un acceso a los directorios para facilitar la carga de datos. Utilizar FTP para el manejo de archivos o si es un usuario más experto, utilizar acceso SSH.

Pasos para configurar los archivos en el servidor

1. Se debe ubicar el directorio raíz principal y acceso público al internet para alojar el sistema base.

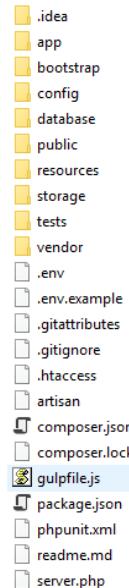


Gráfico 4-1 Estructura de archivos.
Fuente (Antonio Córdova, 2016)

2. Copiar los archivos del completo al directorio público

Nombre	Tipo	Tamaño	Última Modificación
.idea	folder	-	01-11-16 00:53:38
app	folder	-	16-08-16 00:23:58
bootstrap	folder	-	26-11-16 16:28:48
config	folder	-	06-03-16 13:52:34
database	folder	-	06-03-16 13:52:34
public	folder	-	01-11-16 00:51:54
resources	folder	-	03-06-16 02:05:28
storage	folder	-	06-03-16 13:52:34
tests	folder	-	06-03-16 13:52:34
vendor	folder	-	25-06-16 01:25:36
.env	file	427 Bytes	01-11-16 00:46:23
.env.example	file	365 Bytes	06-03-16 13:52:34
.gitattributes	file	61 Bytes	06-03-16 13:52:34
.gitignore	file	73 Bytes	06-03-16 13:52:34
.htaccess	file	92 Bytes	01-11-16 00:53:42
artisan	file	1 KB	06-03-16 13:52:34

Gráfico 4-2 Archivos en Servidor.
Fuente (Antonio Córdova, 2016)

3. Editar el archivo ENV para configurar el acceso a la base de datos

```
DB_HOST=localhost
DB_DATABASE=NOMBRE_DE_BASE_DATOS
```

DB_USERNAME=NOMBRE_DE_USUARIO
 DB_PASSWORD=CONTRASEÑA

Pasos para configurar la base de datos

Se requiere un respaldo completo de la base de datos con la cual se desea trabajar, puede ser la base de datos en entorno de desarrollo o de alguna migración.

1. Desde el servidor se debe crear una base de datos, un usuario nuevo y una contraseña segura.
2. Utilizar phpMyAdmin para realizar la carga de la base, o se puede realizar a través de conexión SSH.

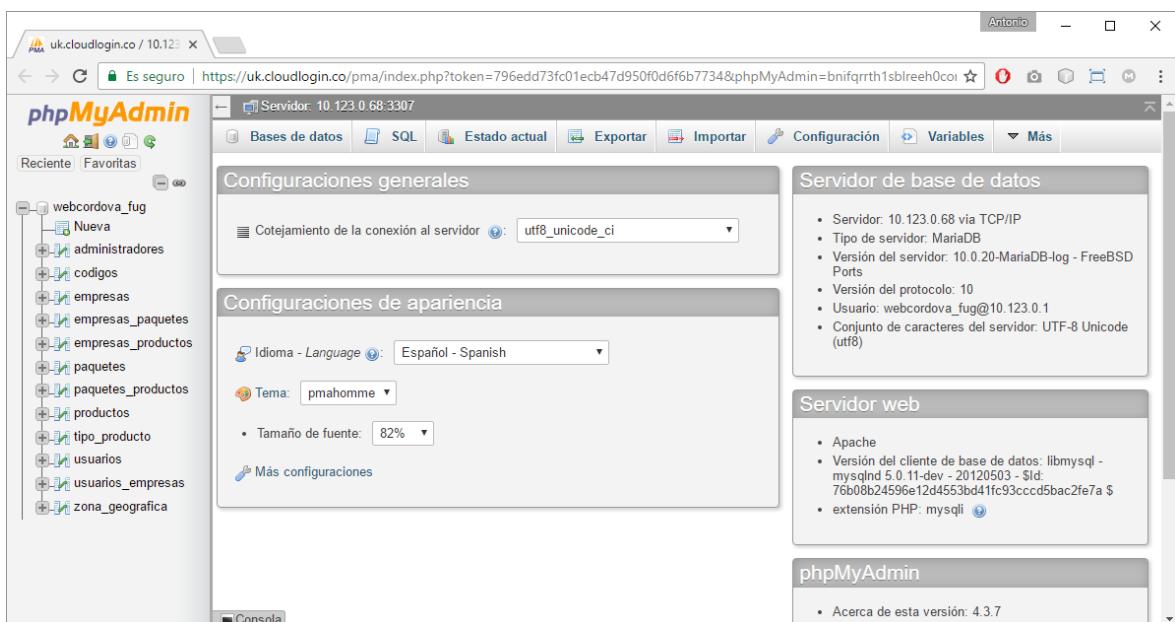


Gráfico 4-3 phpMyAdmin carga de BDD.

Fuente (Antonio Córdova, 2016)

5. Bibliografía

- Adermann, N., & Boggiano, J. (2015). *Dependency Manager for PHP*. Obtenido de Dependency Manager for PHP: <https://getcomposer.org/doc/00-intro.md>
- Agile Alliance. (19 de 02 de 2016). *Agile 101*. Obtenido de Agile Alliance: sitio wwb: <https://www.agilealliance.org>
- Alter Way. (21 de 02 de 2016). *WampServer: a Windows web development environment*. Obtenido de Pagina web de WampServer: <http://www.wampserver.com/en/>
- De la Cruz, F. (2014). Diagramas de Casos de Uso. *Casos de Uso: Generalidades*, (pág. 2). Quito.
- De la Cuz, F. (2014). Proceso de Desarrollo de Software. *Ingeniería de Software I*, (pág. 11). Quito.
- DENSO WAVE INCORPORATED. (2014, 05 01). *qrcode*. Retrieved from <http://www.qrcode.com/en/history/>
- El ciclo de vida de los sistemas informáticos. (2006). En V. Fernández Alarcón, *Desarrollo de sistemas de información: una metodología basada en el modelado* (págs. 40-42). Barcelona: Edicions UPC.
- Estudiantes de la Universidad Carlos III de Madrid. (22 de Diciembre de 2015). *EjemplosTIW*. Obtenido de Patrón de arquitectura Modelo Vista Controlador (MVC): <http://www.lab.inf.uc3m.es/~a0080802/RAI/mvc.html>
- Fundación para la Universitat Oberta de Catalunya. (2012). Introducción a las metodologías ágiles. *Otras formas de analizar y desarrollar*, 19-20.

- Gogala, M., MacIntyre, P., & MacDonald, A. (2011). *Pro PHP Programming*. New York: Apress.
- Google Inc. (20 de 02 de 2016). *AngularJS — Superheroic JavaScript MVW Framework*. Obtenido de Pagina Web de Angular JS: <https://docs.angularjs.org/misc/faq>
- Grupo ISSI. (12 de Noviembre de 2003). Ingeniería del Software y Sistemas de Información. *Metodologías Ágiles en el Desarrollo de Software*, 1-2. Alicante, Alicante, España: Patricio Letelier Torres.
- Hadamitzky, W., & Spahn, M. (1997). Kanji & Kana. En W. Hadamitzky, & M. Spahn, *Kanji & Kana* (pág. 9). North Clarendon: Tuttle Publishing.
- ISO/IEC 18004:2015. (16 de 02 de 2015). Obtenido de ISO - International Organization for Standardization: http://www.iso.org/iso/catalogue_detail.htm?csnumber=62021
- Modelo–vista–controlador*. (20 de 02 de 2016). Obtenido de Modelo–vista–controlador: <https://es.wikipedia.org/wiki/Modelo%20vista%20controlador>
- Palacios, D. (21 de 08 de 2015). *Style.net*. Obtenido de Por qué Laravel NO es un framework MVC y tú deberías olvidarte de MVC: <https://style.net/porque-laravel-no-es-mvc-y-tu-deberias-olvidarte-de-mvc/>
- ProjectLibre. (2015). *About ProjectLibre*. Obtenido de ProjectLibre: <http://www.projectlibre.org/>
- Ramos Martín, A., & Ramos Martín, M. J. (2014). Concepto de la Web 3.0. En A. Ramos Martín, & M. Ramos Martín, *Aplicaciones Web* (págs. 6-7). Madrid: Ediciones Parainfo.
- Rochkind, M. (2013). *Expert PHP and MySQL: Application Design and Development*. Boulder, Colorado.
- Sánchez Morales, M. (2012). Symfony es el Framework de desarrollo. En M. Sánchez Morales, *Manual de Desarrollo Web basado en ejercicios y supuestos prácticos*. (pág. 11). Malaga: CreateSpace .
- Sommerville, I. (2006). Ingeniería del software. En I. Sommerville, *Ingeniería del software* (M. I. Alfonso Galipienso, Trad., págs. 63-64). Madrid: Pearson Educación S.A.
- Tesco PLC. (2016). *Nuestro negocio: Our businesses*. Obtenido de Tesco web site: <http://www.escoplc.com/index.asp?pageid=276>

- The jQuery Team. (20 de 02 de 2016). Obtenido de jQuery Foundation - jquery.org:
<https://jquery.com/>
- The PHP Group. (20 de 02 de 2016). *PHP: Hypertext Preprocessor*. Obtenido de Sitio web oficial de PHP: www.php.net
- Toys R U. (2016). *Sobre la empresa: about us*. Obtenido de Toys R U web site:
<http://www.toysrusinc.com/about-us/>
- W3C. (20 de 02 de 2016). *HTML - CSS - W3C*. Obtenido de The World Wide Web Consortium: <https://www.w3.org/standards/webdesign/htmlcss>
- Wells, D. (2009). *XP: Extreme Programming*. Obtenido de ExtremeProgramming.org home : www.extremeprogramming.org
- Whatsapp, home. (2016). Obtenido de WhatsApp Web site:
<https://www.whatsapp.com/?l=es>