

Universidad ORT Uruguay  
Facultad de Ingeniería

**Automatización de sistema de ventas y  
análisis de datos, integrado con  
Sistema de Información Geográfica**

Entregado como requisito para la obtención del título de  
Analista Programador

Giani Mauro Carlevaro de Sosa – 147376  
Tutor: Fernando Thul

**2018**

## **Declaración de autoría.**

Yo, Mauro Carlevaro, declaro que el trabajo que se presenta en esa obra es de nuestra propia mano. Puedo asegurar que:

- La obra fue producida en su totalidad mientras realizaba la entrega del obligatorio para la culminación de la carrera Analista Programador;
- Cuando he consultado el trabajo publicado por otros, lo he atribuido con claridad;
- Cuando he citado obras de otros, he indicado las fuentes. Con excepción de estas citas, la obra es enteramente mía;
- En la obra, he acusado recibo de las ayudas recibidas;
- Cuando la obra se basa en trabajo realizado conjuntamente con otros, he explicado claramente qué fue contribuido por otros, y qué fue contribuido por mi;
- Ninguna parte de este trabajo ha sido publicada previamente a su entrega, excepto donde se han realizado las aclaraciones correspondientes.

Mauro Carlevaro

26/09/2018

## **Agradecimientos.**

A Fernando Thul, tutor del proyecto, por la ayuda y el apoyo durante todo el transcurso del proyecto.

A los chicos de bedelía, quienes siempre contagiando buena energía asesoran y orientan ante cualquier consulta.

Muy especialmente a mi familia y amigos, gracias a ellos he podido realizar la carrera.

## **Abstract.**

El presente documento describe el proceso de elaboración de un sistema que se ocupa de resolver problemas en procesos de venta web, análisis de datos, administración y organización de una empresa mayorista que comercia productos de alimentación, mediante la automatización de tareas no incluidas en sus sistemas actuales.

El cliente, consultor contable, quien solicita la creación del sistema, necesita poder ofrecer soluciones a empresa mayorista de alimentos.

El principal problema a resolver es el proceso de venta web y la coordinación de envíos. Para esto se incluye un carrito de compras y se automatiza la adjudicación de pedidos a los repartos.

Para la construcción del sistema se consideró primordial que sea simple e intuitivo, de forma que mejore los tiempos de respuesta y que resuelva los principales problemas de la empresa como ser: mejorar el proceso de venta web, simplificar la coordinación de envíos, visualizar rápidamente la información en forma remota, contar con sistemas de avisos y alertas, valuación de stock por lotes y poder tener la opinión de los usuarios sobre el servicio brindado.

El sistema consta de una aplicación web y móvil, se desarrolló mayormente en Java y JavaScript con base de datos MySQL, está compuesto por una parte pública para el usuario (quien desea consumir los servicios de la empresa) y una parte privada, sólo para acceso de la empresa desde donde se realizan las tareas de gestión.

Se consideró importante integrar la exposición de resultados y reportes con un sistema de información geográfica además de representaciones de información por gráficas y resúmenes por tablas, con el objetivo de poder realizar en forma rápida la toma de decisiones mediante la simplificación en la exposición de los datos y la posibilidad de combinar distintas fuentes de información para visualizar distintos escenarios, permitiendo una mejor gestión.

Para la gestión del desarrollo del software se utilizó una adaptación de la metodología Scrum.

En conclusión, se logró una primera implementación de un sistema que permite optimizar el tiempo de trabajo mediante la automatización de tareas repetitivas, minimizar costos y mejorar la calidad de la información para la toma de decisiones.

## **Palabras clave.**

- Ingeniería
- Software
- Análisis
- Optimización
- Automatización
- Valoración
- Ahorro
- Tiempo
- Información geográfica
- Información espacial
- Reportes
- Gestión
- Scrum
- Ágil
- Metodologías ágiles
- Aplicación web

## **Nomenclatura y acrónimos.**

- SQA - estándares y testing.
- SCM – versionado y cambios.
- Usuario – quien desea consumir los servicios de la empresas
- Empresa – para quién se desarrolla el sistema

# Índice.

1.	Anteproyecto .....	14
1.1.	Introducción.....	14
1.2.	Presentación del cliente. ....	14
1.2.1.	Roles y contacto con el cliente .....	15
1.3.	Presentación del problema.....	15
1.4.	Descripción del entorno.....	16
1.5.	Análisis estratégico.....	16
1.6.	Lista de necesidades. ....	17
1.7.	Actores involucrados.....	19
1.8.	Objetivos.....	19
1.9.	Lista de requerimientos. ....	20
1.9.1.	Requisitos funcionales.....	20
1.9.2.	Requisitos no funcionales.....	24
1.10.	Alcance y limitaciones.....	24
1.11.	Estudio de alternativa .....	24
1.11.1.	Alternativa 1 (Adoptar una solución existente).....	25
1.11.2.	Alternativa 2 (Crear un sistema a medida).....	25
1.11.3.	Selección de alternativa .....	25
1.12.	Arquitectura.....	25
1.13.	Ánálisis de factibilidad. ....	27
1.13.1.	Operativa. ....	27
1.13.2.	Técnica. ....	27
1.13.3.	Legal .....	28
1.13.4.	Económica. ....	28
1.14.	Ánálisis de riesgo. ....	28

1.14.1.	Planes de contingencia y mitigación.....	30
1.15.	Plan de Proyecto.....	32
1.15.1.	Definición del proceso.....	32
1.15.1.1.	Metodología.....	32
1.15.1.2.	Ciclo de vida.....	32
1.15.1.3.	Funcionalidades a desarrollar (Pila del Producto o Product Backlog). ....	33
1.15.1.4.	Diagrama de Gantt.....	35
1.15.2.	Descripción de la estructura del proyecto.....	37
1.15.3.	Integrantes y roles.....	40
1.15.4.	Descripción y selección de herramientas.....	40
1.15.5.	Plan de Calidad (SQA). .....	43
1.15.5.1.	Plan de testing.....	44
1.15.5.1.1.	Test de caja negra. ....	44
1.15.5.1.2.	Test de caja blanca.....	45
1.15.6.	Estándares definidos y convenciones. ....	45
1.15.6.1.	Documentación.....	45
1.15.6.2.	Nomenclatura.....	46
1.15.7.	Plan SCM (Software Configuration Managment).....	47
1.15.7.1.	Gestión de configuración.....	47
1.15.7.2.	Control de versionado.....	47
1.15.7.3.	Plan de capacitación. ....	47
2.	Proyecto.....	48
2.1	Solución lograda. ....	48
2.1.1	Diagrama de clases. ....	48
2.1.2	Tecnologías utilizadas. ....	50
2.2	Requerimientos funcionales. ....	50

2.3	Diseño de la aplicación.....	56
2.3.1	Clases.....	56
2.3.2	Diagrama de clases.....	58
2.4	Ajuste de Sprints.....	61
2.4.1	Sprint 1. Análisis y Diseño.....	61
2.4.2	Sprint 2. Inicio de desarrollo de las funcionalidades.....	63
2.4.3	Sprint 3. Desarrollo de sistema de alertas, búsquedas y mantenimiento de órdenes de trabajo.....	64
2.4.4	Sprint 4. Sistema de fidelización, refactorización y mejora de funcionalidades.....	66
2.4.5	Sprint 5. Emisión de informes, mejora de interfaz de usuario y sistema de información geográfica.....	67
2.4.6	Sprint 6. Mejora del SIG, validaciones y app móvil. ....	69
2.4.7	Sprint 7. Cierre de funcionalidades y de documentación, optimización del diseño, refactorización y testeo.....	70
2.5	Estándares definidos y convenciones. ....	72
2.5.1	Documentación.....	72
2.5.2	Nomenclatura.....	72
2.6	Análisis e investigación de tecnologías. ....	73
2.6.1	Librerías Utilizadas. ....	75
2.7	Pruebas. ....	76
2.7.1	Introducción.....	76
2.7.2	Plan de testing.....	76
2.7.2.1	Test de caja negra. ....	77
2.7.2.2	Test de caja blanca.....	77
2.8	Plan de proyecto. ....	77
2.8.1	Introducción.....	77

2.8.2	Gestión de riesgos.....	77
2.9	Proceso de desarrollo.....	79
2.9.1	Introducción.....	79
2.9.2	Adaptación del proceso. ....	79
2.9.3	Seguridad. ....	79
2.10	Proceso de entrega.....	80
2.10.1	Manual de deploy. ....	80
2.11	Análisis de Esfuerzo. ....	81
3.	Conclusiones.....	85
3.1.	Lecciones aprendidas.....	86
4.	Referencias bibliográficas. ....	88
5.	Bibliografía.....	89
6.	Anexos .....	91
6.1.	Anexo 1. Manual de Usuario.....	91
6.1.1.	Funcionalidades. Sitio Cliente.....	91
6.1.2.	Funcionalidades. Sitio Empleado. ....	92
6.1.3.	Funcionalidades. Aplicación Móvil.....	102
6.2.	Anexo 2. Casos de uso.....	104
6.3.	Anexo 3. Diagramas de secuencia.....	111
6.4.	Anexo 4. Diagrama de componentes.....	113
6.5.	Anexo 5. Diagrama de despliegue.....	113
6.6.	Anexo 6. Metodología implementada en las entrevistas.....	114
6.7.	Anexo 7. Proceso de elaboración aplicado.....	114
6.8.	Anexo 8. Proceso de pruebas.....	115
6.9.	Anexo 9. Presentación.....	139
7.	Glosario. ....	143

## **Índice de tablas.**

Tabla 1. Rangos de probabilidades (Sommerville).....	28
Tabla 2. Valores de impacto (Sommerville).....	29
Tabla 3. Principales Riesgos.....	29
Tabla 4. Funcionalidades a desarrollar.....	34
Tabla 5. Sprint 1- Análisis y Diseño. ....	61
Tabla 6. Sprint 2. Inicio de desarrollo de las funcionalidades.....	63
Tabla 7. Sprint 3. Desarrollo de sistema de alertas, búsquedas y mantenimiento de órdenes de trabajo.....	65
Tabla 8. Sprint 4. Sistema de fidelización, refactorización y mejora de funcionalidades. ....	66
Tabla 9. Sprint 5. Emisión de informes, mejora de interfaz de usuario y sistema de información geográfica.....	67
Tabla 10. Sprint 6. Mejora del SIG, validaciones y app móvil. ....	69
Tabla 11. Sprint 7. Cierre de funcionalidades y de documentación, optimización del diseño, refactorización y testeo. ....	70
Tabla 12. Principales Riesgos.....	78
Tabla 13. Estimación de esfuerzo según funcionalidades. ....	83
Tabla 14. Caso de uso narrativo. Registro de Cliente. ....	105
Tabla 15. Caso de uso narrativo. Modificar datos Usuario .....	106
Tabla 16. Caso de uso narrativo. Realizar un pedido. ....	108
Tabla 17. Caso de uso narrativo. Calificación de pedidos.....	108
Tabla 18. Caso de uso narrativo. Repartos propios en curso.....	109
Tabla 19. Caso de uso narrativo. Dar de baja un empleado. ....	110

# Índice de ilustraciones

Ilustración 1. Diagrama de Gantt.....	35
Ilustración 2. Diagrama de Gantt.....	36
Ilustración 3. M.E.R. (Modelo Entidad Relación).....	37
Ilustración 4. Diagrama de clases.....	49
Ilustración 5. Diagrama de clases.....	58
Ilustración 6. Diagrama de clases.....	59
Ilustración 7. Diagrama de clases.....	60
Ilustración 8. Sprint 1- Análisis y Diseño. ....	62
Ilustración 9. Sprint 2. Inicio de desarrollo de las funcionalidades.....	63
Ilustración 10. Sprint 3. Desarrollo de sistema de alertas, búsquedas y mantenimiento de órdenes de trabajo.....	65
Ilustración 11. Sprint 4. Sistema de fidelización, refactorización y mejora de funcionalidades.....	66
Ilustración 12. Sprint 5. Emisión de informes, mejora de interfaz de usuario y sistema de información geográfica.....	67
Ilustración 13. Sprint 6. Mejora del SIG, validaciones y app móvil. ....	69
Ilustración 14. Sprint 7. Cierre de funcionalidades y de documentación, optimización del diseño, refactorización y testeo.....	70
Ilustración 15. Funcionalidades por sprint según cantidad de días. ....	83
Ilustración 16. Funcionalidades por sprint según cantidad de días. ....	84
Ilustración 17. Funcionalidades por sprint según porcentaje. ....	84
Ilustración 18. Login. ....	91
Ilustración 19. Menú administrador.....	93
Ilustración 20. Histórico repartos. ....	95
Ilustración 21. Alta Producto.....	96
Ilustración 22. Asignación Empleado a reparto.....	97
Ilustración 23. Montos de pedidos por mes. ....	98
Ilustración 24.Sistema de información geográfica. ....	100
Ilustración 25. App móvil. Login. ....	102
Ilustración 26. App móvil. Repartos Pendientes para el día.....	103
Ilustración 27. App móvil. Ubicación reparto. ....	103

Ilustración 28. Diagrama de caso de uso, ingresar pedido.	104
Ilustración 29. Diagrama de secuencia. Alta Cliente.	111
Ilustración 30. Diagrama de secuencia. Alta de Pedido.	112
Ilustración 31. Diagrama de componentes.	113
Ilustración 32. Diagrama de despliegue.	113
Ilustración 33. Pruebas. Login Incorrecto.	116
Ilustración 34. Pruebas. Alta Empleado datos correcto.	117
Ilustración 35. Pruebas. Alta Cliente, datos correctos.	118
Ilustración 36. Pruebas. Alta Producto, datos correctos.	119
Ilustración 37. Pruebas. Alta Producto.	120
Ilustración 38. Pruebas. Alta Reparto, datos correctos.	122
Ilustración 39. Pruebas. Alta Proveedor, sin ingresar dirección.	123
Ilustración 40. Pruebas. Calificación.	124
Ilustración 41. Pruebas. Alta Factura.	125
Ilustración 42. Pruebas. Consolidar compra.	125
Ilustración 43. Pruebas. Repartos pendientes para el día.	127
Ilustración 44. Pruebas. Repartos propios para el día en curso.	128
Ilustración 45. Pruebas. Repartos en Curso.	128
Ilustración 46. Pruebas. Estado Repartos.	129
Ilustración 47. Pruebas. Pedidos en reparto.	129
Ilustración 48. Pruebas. Factura.	130
Ilustración 49. Pruebas. Stock.	131
Ilustración 50. Pruebas. Productos.	131
Ilustración 51. Pruebas. Cantidad de pedidos por mes.	132
Ilustración 52. Pruebas. Monto contra cantidad de pedidos.	133
Ilustración 53. Pruebas. Cantidad de pedidos por Cliente y datos asociados.	134
Ilustración 54. Mapa. Road view.	135
Ilustración 55. Mapa. Satellite view.	136
Ilustración 56. Pruebas. Baja Empleado.	137
Ilustración 57. Pruebas. Mails.	138
Ilustración 58. Presentación.	142

## **1. Anteproyecto.**

### **1.1. Introducción.**

Con el anteproyecto se pretende establecer una base sólida, despejar dudas, aclarar ideas y anticipar problemas, que permita realizar un análisis preciso y conceptualizar eficazmente las tareas a desarrollar, estableciendo la base para poder realizar un proceso continuo y controlado como medio para cumplir los objetivos.

El anteproyecto es una herramienta para obtener el mayor provecho del trabajo a realizar y optimizar el esfuerzo. Además cumple la función de contrato con el cliente, por lo que deberá definirse claramente:

- el problema a solucionar. La correcta definición del problema repercutirá en la satisfacción del producto final.
- el alcance del proyecto y establecer metas.
- la solución, la propuesta de valor y se deberá especificar las funcionalidades críticas.

### **1.2. Presentación del cliente.**

Cliente: Sr. Fernando Lladó, contador especializado en tributaria. Solicita la confección del sistema para ofrecer a una empresa mayorista que se desenvuelve en el rubro alimenticio.

Consultor contable que se encarga de aportar soluciones a empresas, a quien le aportaría valor poder contar con un sistema que organice, automatice y simplifique procesos de la lógica de negocio de las empresas con las que trabaja con el fin de poder simplificar la tarea y poder reflejar de mejor forma la realidad de las mismas.

Dada su experiencia el cliente solicita la creación del sistema para resolver algunos de los principales problemas de una empresa mayorista como ser: el proceso de venta web, simplificar la coordinación de envíos, poder visualizar rápidamente la información de forma remota, contar con sistemas de avisos y alertas, valuación de stock por lotes y poder tener la opinión de los usuarios sobre el servicio brindado.

El sistema le representa al cliente un diferencial a la hora de ofrecer sus servicios.

### **1.2.1. Roles y contacto con el cliente.**

Cliente: es con quien se va a trabajar, es quien solicita la creación del sistema y quien conoce la lógica del negocio, por lo tanto definirá la prioridad de las funcionalidades a desarrollar.

Se tendrá, al menos, contacto con el cliente cada 20 días. Se le informarán los avances realizados, los problemas a solucionar y lo que se pretende lograr hasta la próxima reunión. En cada reunión se entrega una versión del sistema con los avances logrados, al utilizar las funcionalidades desarrolladas permite obtener una devolución de las mismas y así ir avanzando en conjunto con el cliente para lograr un producto que satisfaga las necesidades y evitar imprevistos. Con esta modalidad el cliente actúa como tester, por lo cual su implicancia en el desarrollo es mayor lo cual permite obtener un mejor producto.

Empresa: en base a los requisitos que tiene para solucionar la empresa se establecen los requerimientos del sistema, por tanto, la empresa es para quien se confecciona la solución.

Usuarios: son quienes usan el sistema, quienes solicitan los servicios de la Empresa.

### **1.3. Presentación del problema.**

La problemática a resolver se puede dividir en varios puntos, como ser:

- Necesidad de visualizar rápidamente la información para poder tomar decisiones a tiempo.
- Falta de claridad en la exposición de los productos y proceso de venta web ineficiente y desactualizado.
- Falta de coordinación en los envíos.
- Falta organización de la información, accesibilidad en forma remota y control de usuarios con acceso.
- Faltan controles de validación. La empresa estima que varios clientes repiten compras pero sería importante validar dicha hipótesis, al poder identificar comportamientos repetitivos y que el sistema simplifique la tarea al usuario.
- Se necesita poder realizar atención personalizada a fin de mejorar el servicio a los clientes y también poder tener la evaluación del usuario sobre el servicio brindado.
- Se necesita disponer de recordatorios para reponer stock, contar con la valuación del mismo y de las diferentes partidas de productos.

## **1.4. Descripción del entorno.**

Actualmente para concretar la venta-compra se envía una planilla electrónica de la empresa con los productos a quien desea comprar, quien va a comprar (usuario) elige los productos y los envía por mail. Si no hay stock de algún producto este se sustituye por similar o no se envía.

Los envíos se realizan a partir de las 48 hs. siguientes a la solicitud, en algunos casos se coordina por mail en donde el rango horario varía (de 8 a 17), a su vez, en la mayoría de los casos no se coordina y el reparto se realiza más de una vez para hacer efectiva la entrega de los productos.

Cuando se debe modificar algún producto se trabaja manualmente sobre la foto y los detalles del mismo, no accede de forma automática a los datos para que no sea necesario tocar el diseño cada vez que se genera un cambio.

Para la gestión de las tareas no se cuenta con un sistema que integre, procese la información y automatice tareas repetitivas.

## **1.5. Análisis estratégico.**

En el mercado existen sistemas que cuentan con las funcionalidades solicitadas pero se justifica el desarrollo de un sistema a medida, ya que, se vieron los siguientes beneficios en implementar un sistema personalizado:

- Permite tener solamente las funcionalidades deseadas, lo cual simplifica, redundando en la disminución de distracciones y por tanto se logrará un mejor aprovechamiento del tiempo.
- Se realizará con tecnologías de software libre que facilita la integración con otros sistemas y escalabilidad.
- A diferencia de la mayoría de los sistemas, se integrará la exposición de resultados con un sistema de información geográfica, con el objetivo de poder realizar en forma rápida la toma de decisiones mediante la simplificación en la exposición de los datos y la posibilidad de combinar distintas fuentes para visualizar escenarios alternos.

En cuanto al funcionamiento se pretende que se ingresen los productos a la base de datos y en forma automática se genere el carrito de compras, eliminando las tareas de mantenimiento. Al realizar la compra se debe elegir el rango horario de entrega y el día en que se efectuará.

Deberán quedar disponibles sólo los productos con stock, esto elimina el tiempo destinado a contestar consultas de stock y detalles de productos.

La integración con el sistema actual no sería problema ya que no se debe consumir ningún servicio, ni información existente y el nexo se daría con la página donde se realiza la compra la cual genera la información que va a procesar el sistema. Se tendría que acompañar el diseño de la página de compra al diseño ya existente en la web actual.

El proceso de venta web actual conlleva a la pérdida de clientes debido a la dificultosa forma de concretar la venta-compra.

Se pierde tiempo en contestar consultas por stock y precios, algunas quedan sin contestar o no se hace a tiempo, no existen controles para poder estimar resultados.

## **1.6. Lista de necesidades.**

La lista de necesidades surge a partir de la identificación de las deficiencias actuales, por lo que se listarán las funciones que hacen falta o que se puede mejorar su rendimiento.

A partir de la interacción con el cliente (quien solicita el sistema) y con el grupo objetivo, es que se adquiere el conocimiento necesario para desarrollar eficientemente la lógica del negocio y se identifican las funcionalidades a construir y optimizar.

Para lograr que sean productivas las reuniones con el cliente, las mismas se preparan con una guía de temas a tratar. También se le informa al cliente los temas que se pretenden solucionar para que las reuniones sean más productivas.

Los productos que se encuentren actualmente sin stock se deben poder ver pero no se deben poder ingresar en un pedido de compra.

Es necesario incluir un carrito de compras que minimice el tiempo necesario para realizar los pedidos.

Se precisa reemplazar el sistema de trabajo actual en base a planillas electrónicas, eliminar inconsistencias en la información y en los procesos. El sistema debe disminuir los tiempos necesarios para la revisión de documentación. Se debe tener rápido acceso a información, debe ser actual, fiable e integra. Informes de stocks, ventas, clientes, compras de mercadería y proveedores habituales servirán para obtener un análisis rápido y confiable. Con la información actualizada de la realidad en su contexto. Todo esto a fin de saber el estado de situación del negocio.

### **Listas de necesidades:**

- Incluir carrito de compras.
- Incluir agenda de envíos.
- Automatizar tareas, eliminar tareas repetitivas (reducir errores).

- Optimizar tiempos en las tareas, optimizando recursos y aumentando la eficiencia.
- Realizar un único sistema que unifique la información existente, eliminando inconsistencia de datos y ruidos en la comunicación.
- Incluir funcionalidad de stock mínimo con alertas cuando se está llegando a un número crítico de productos, el cual será definido por la empresa. También se debe poder prever cuándo y cuánto se va a necesitar más insumos. Se entiende stock mínimo como el valor de referencia que le sirve a la empresa para calcular el número de ventas y que se deben tener en cuenta los plazos y ciclos habituales de la demanda según el contexto.
- Incluir filtro por categoría, marca y posibilidad de realizar búsquedas de productos.
- El repartidor debe poder avisar cuando concreta la entrega y se debe visualizar en tiempo real desde la empresa para poder estimar mejor los tiempos de entrega de los clientes restantes y reportar problemas de ser necesario.
- Posibilidad de acceso remoto.
- Sistema de roles.
- En cada compra realizada quedará asociada la factura y los detalles de la compra del usuarios
- Posibilidad de generar reportes de clientes habituales, cuánto compran, compras en relación a distintos periodos, zonas de ubicación de los mismos y poder estimar compras futuras de los clientes.
- Historial de ventas a clientes y compras efectuadas a proveedores.
- Se debe contar con un sistema de información geográfica que permita realizar el seguimiento de los repartos.
- Se contará con gráficas para visualizar de mejor forma la información.
- Posibilidad de tener detallados los datos de los clientes.
- Posibilidad de envío de promociones, descuentos y mensajes en fechas especiales para el cliente, como ser: cumpleaños, fin de año o si se desea tener una atención con el cliente en base a la confianza depositada.
- Posibilidad de programar alertas de información de stock mínimo para realizar avisos cuando se deba reponer stock.

- Al ingresar la reposición de stock, el nuevo stock se debe ingresar como nuevo lote para poder diferenciarlo del anterior.
- Implementar sistema de calificación del servicio.
- Análisis de la información en forma automática respecto al conocimiento de los clientes (ubicación, preferencias de consumo, etc) y servicio brindado, lo cual es necesario para asegurar la calidad de los productos y del servicio.

Es importante poder visualizar los datos de forma rápida y tener la capacidad de evaluar cuánto representa la muestra analizada en función del entorno. Hacer distribuciones de frecuencia, presentar datos agrupados tabulados en forma cuantitativa y cualitativa. Toda funcionalidad debe de aportar valor en la realidad.

## **1.7. Actores involucrados.**

Los actores involucrados, accionistas o stockholders se pueden diferenciar en dos grupos: los usuarios del sistema y los empleados, quienes también pueden ser usuarios. Ambos grupos van a interactuar de distinta forma con el sistema.

Las acciones que ambos grupos deban poder realizar van a tener influencia directa o indirecta sobre los requerimientos del sistema.

### **Actores:**

- Usuarios: son quienes consumen los servicios ofrecidos por la Empresa. Usuarios del carrito de compras.
- Empresa: quien ofrece los productos y servicios. Administrador del sistema.
- Cliente: quien contrata la realización del sistema. Se encarga del análisis contable y estadístico de la información para la empresa.

## **1.8. Objetivos.**

El objetivo de desarrollo del sistema es: **optimizar tiempos y esfuerzos.**

Los objetivos planteados para ser resueltos por el sistema son que cumpla las reglas de negocio y dar solución a la lista de necesidades para que la mejora y simplificación en el desempeño de las tareas repercuta en un mejor ambiente laboral.

También se debe mejorar la toma de decisiones, mediante la creación de gráficas, reportes, permitir relacionar acciones con usuarios y posibilitar seguir el proceso para poder optimizar el mismo.

Se clasificarán las funcionalidades por importancia de forma de ir realizando primero las de mayor importancia y continuar en orden descendente para minimizar riesgos.

Cumplir con los objetivos implica solucionar las necesidades.

En caso de que se requiera o se necesite utilizar alguna información ya ingresada en el sistema, como fichas de datos de proveedores o clientes, la misma no se tendrá que volver a ingresar sino que el sistema la administrará de forma automática.

Para alcanzar los objetivos se divide el proceso de desarrollo en sprints. En los cuales se aclara el producto desarrollado en el sprint, los casos de prueba, documentación, el esfuerzo realizado (tiempo de trabajo), ajustes realizados o que se estiman necesarios, nuevos requerimientos si corresponden, trabajo adicional en caso que queden tareas de sprints anteriores.

## **1.9. Lista de requerimientos.**

A partir de las necesidades y los objetivos planteados se consideran los siguientes requerimientos.

Se dividirá en requerimientos funcionales y no funcionales. Para los primeros se declaran los servicios que proveerá el sistema, como reaccionará frente a entradas particulares y como se comportará ante situaciones especiales.

### **1.9.1. Requisitos funcionales.**

- I. **Login.** El sistema deberá restringir el acceso mediante login, desde donde se identifica el ingreso al sistema, si el usuario es un cliente se redirige a la página donde se encuentran los productos y si es un empleado se redirige al menú administrador. Los usuarios se loguearan con mail y contraseña.
- II. **El sistema sólo debe permitir comprar a usuarios registrados.** Se controlará el tipo de usuario, a fin de saber qué funcionalidades habilitar al mismo. Pueden ser usuarios clientes o empleados.
- III. **El sistema permitirá realizar mantenimientos** de usuarios, empleados, productos, pedidos, repartos, marcas, categorías y proveedores. Esto incluirá funcionalidades para realizar alta, baja y actualización de datos.

Los usuarios se dividirán en clientes y empleados.

Los usuarios se identificarán por una clave primaria, numérica y auto incremental. Los datos a registrar de los clientes son: nombre, apellido, clave de acceso al sistema (debe estar encriptada), email (será único para cada usuario), dirección (se compone por el departamento, calle, número de

puerta, latitud y longitud de la ubicación. Estos últimos dos datos los obtendrá el sistema de forma automática) y móvil de contacto.

Además de los datos anteriores los clientes podrán especificar si son personas físicas o empresas y a los empleados se les registrará la cédula, puesto y sueldo.

Los repartos contarán con un identificador numérico y auto incremental, se registrará la zona a la que corresponde el reparto, el vehículo y la cantidad máxima de pedidos que puede realizar.

Los productos se identifican por un id numérico auto incremental, se registrará el nombre, lote y marca de cada producto (estos tres ítems en su conjunto serán únicos, no se podrán existir dos productos con el mismo nombre, lote y marca), precio del proveedor, precio de venta y precio de oferta. Además se tendrá el stock actual y el stock que determinan la empresa como stock mínimo, a partir del cual se avisará para realizar la reposición del mismo. Se podrá identificar el producto como nuevo, recomendado y también agregar detalle y visibilidad del mismo. Contará foto, la fecha ingreso, la fecha de vencimiento, la categoría y al proveedor que corresponde.

Para las marcas se ingresará el nombre y la visibilidad dependiendo si cuenta o no con productos, deberán tener un identificador numérico auto incremental.

Las categorías contarán con un identificador igual que las marcas y se registrará su nombre, visibilidad y si es subcategoría de otra categoría.

Los proveedores tendrán un id, numérico y auto incremental, nombre, email, la dirección (compuesta por el departamento, calle, número de puerta, latitud y longitud de la ubicación. Estos últimos dos datos los obtendrá el sistema de forma automática), un teléfono y el ruc.

Para los pedidos se registrará el id del pedido (auto incremental), la fecha de solicitud, la dirección compuesta por departamento, calle y número de puerta, latitud y longitud de la ubicación. Estos últimos dos datos los obtendrá el sistema de forma automática. Además se cuenta con la posibilidad de ingresar una observación. También se debe tener registrado el cliente de la solicitud, el empleado a cargo, a qué reparto corresponde el pedido según la zona del mismo, la fecha y horario de entrega, el estado y la calificación dada por el cliente.

IV. **Ingreso de productos.** Al ingresar el producto si la marca y categoría no existe, en el formulario de ingreso de producto se deberá contar con la opción para ingresar la misma.

V. **El sistema no debe permitir** ingresar marcas, categorías y proveedores con el mismo nombre.

- VI. **Control de eliminación de información.** Antes de eliminar cualquier información se verifica que no integre ningún trabajo en curso.
- VII. **Baja empleado.** Al dar de baja un empleado no se debe borrar del sistema se deja registrado como cliente.
- VIII. **Identificadores.** Cada objeto o dato contará con identificador.
- IX. **Carrito de compras.** El sistema deberá permitir realizar pedidos.
- X. **Filtros.** Para que se pueda usar el carrito de compras el usuario debe estar logueado y los productos se podrán seleccionar por filtro de marca o categoría.
- XI. **Control de errores.** Cada formulario contará con su respectiva validación antes de enviar los datos al servidor.
- XII. **Visualización de factura de pedidos.** En cada compra realizada quedará asociada la factura y los detalles de la compra al usuario en cuestión. Será visible para el comprador y para los administradores del sistema.
- XIII. **Funcionalidad de Stock Mínimo.** La funcionalidad de stock mínimo contará con alertas cuando se está llegando a un número crítico de productos, el cual será definido por la empresa. Se visualizará la cantidad y la valuación del stock.
- XIV. **Reposición de stock.**
- XV. **Mantenimiento de órdenes de trabajo.** En caso de que no se pueda entregar el pedido en la fecha prevista. El empleado ingresará la observación y automáticamente el pedido se incluirá para el día siguiente. Para versiones posteriores se planea poder coordinar automáticamente con el usuario cuando realizar el reenvío del pedido.
- XVI. **Visualización de Pedidos** en el panel administrador. Se podrán ver los detalles de los pedidos, productos que lo integran y comprador del mismo.
- XVII. **Emisión de informes.** El sistema emitirá informes del accionar de los usuarios y estadísticas de uso.
- XVIII. **Servicios contratados.** Dado un cliente mostrar el total de todos los servicios contratados.
- XIX. **Ventas por fecha.** Dada una fecha obtener una lista de la cantidad de ventas realizadas.
- XX. **Listado de ventas.** Se contará con la posibilidad de obtener una lista de las ventas realizadas, el monto y los productos vendidos.

- XXI. **Sistema de información geográfico.** Se contará con un sistema de información geográfico, el cual permitirá la visualización de la información espacial en su contexto, además da la posibilidad de combinar diferentes fuentes de datos y realizar análisis espacio temporal.
- XXII. **Servicio de mails.** Se contará con la posibilidad que cualquier usuario pueda contactarse con la empresa por medio de un formulario desde la web, se implementara como servicio de envío de mail.
- XXIII. **Agenda de envíos.** Al dar de alta la compra se elige rango horario y posibles días para realizar la entrega.
- XXIV. **Calificar.** Al finalizar un servicio el cliente contará con la posibilidad de calificar el servicio y agregar una observación.
- XXV. **Búsquedas.** El sistema deberá permitir realizar búsquedas de los productos y pedidos.
- XXVI. **Configuración ambiente de testing.**
- XXVII. **Asignación Pedidos a Reparto.** La asignación de un pedido a un reparto se hará de forma automática según la dirección de entrega y la zona que le comprende a cada reparto.
- XXVIII. **Control de reparto.** Se debe validar que el reparto no esté saturado, esto se hará en base a la fecha seleccionada para la entrega del pedido. En caso de que el reparto ya tenga disponibilidad esa fecha se bloqueara impidiendo que se seleccione como fecha posible. En cada reparto se registrará el empleado a cargo del mismo.
- XXIX. **Aviso de entrega realizada.** Al entregar un pedido el empleado a cargo del reparto registrará que se realizó, lo cual se reflejara en forma instantánea en el sistema. Permitiendo realizar un seguimiento desde la empresa en tiempo real. Se podrá realizar mediante conexión web por un explorador o utilizando la app móvil mediante conexión api rest.
- XXX. **Sistema de fidelización.** Posibilidad de envío de promociones, descuentos y mensajes en fechas especiales para el cliente, como ser: cumpleaños, fin de año o si se desea tener una atención con el cliente en base a la confianza depositada.

### **1.9.2. Requisitos no funcionales.**

Los requisitos no funcionales comprenden la capacidad que debe poseer el sistema para satisfacer las necesidades, objetivos y especificaciones funcionales.

La aplicación será web, de forma que se pueda implementar la modalidad de teletrabajo.

Se desarrollará con tecnologías open source.

Para asegurar una buena performance se realizaran pruebas de testeo, mayoritariamente esta tarea la realizará el grupo objetivo y la empresa a medida que se vayan entregando los sprints.

La aplicación deberá restringir el acceso a las funcionalidades mediante el uso de mail y contraseña.

La contraseña deberá estar encriptada.

El sistema debe ser multiplataformas, tener una interfaz gráfica simple y amigable, de forma que permita trabajar de forma intuitiva.

Se debe cumplir con las fechas de entrega establecidas por el cronograma académico definido por Universidad ORT Uruguay

**El sistema tiene que estar funcionando para el 26 de setiembre.**

### **1.10. Alcance y limitaciones.**

Se desarrollará la aplicación para simplificar y optimizar el desempeño de la empresa, el alcance será cubrir las necesidades y cumplir los objetivos planteados.

No se realizará sistema de factura electrónica, en caso de ser necesario se integrara con alguna solución ya existente.

### **1.11. Estudio de alternativa.**

De la investigación realizada se ve que a pesar de la existencia de sistemas que ya cuentan con las funcionalidades solicitadas, se justifica el desarrollo de un sistema a medida, ya que no tendrán que asumir costos de licencias y el desarrollo será escalable fácilmente ya que se realizará con tecnologías de software libre.

La investigación constó en ver distintas soluciones de e-commerce que se ofrecen en el mercado para evaluar si ameritaba realizar un desarrollo de cero o utilizar una solución ya existente.

### **1.11.1. Alternativa 1 (Adoptar una solución existente).**

Las dos principales desventajas vistas en caso de adoptar una solución existente son que a la hora de escalar se va a estar restringido por las soluciones que brinde la plataforma elegida y que las soluciones existentes que cuentan con la posibilidad de generar un sistema de información geográfico contenían muchas más funcionalidades de las que se necesitan y además subían mucho los costos. Por lo que si se quería optar por esta alternativa se iba a tener un sistema más complejo de lo que se pretendía y bastante más costoso.

### **1.11.2. Alternativa 2 (Crear un sistema a medida).**

La creación de un sistema a medida permite tener sólo las funcionalidades necesarias y al desarrollarse con tecnologías de software libre no se cuenta con limitaciones a la hora de escalar, además de reducir significativamente los costos.

### **1.11.3. Selección de alternativa.**

En base a lo analizado en las distintas alternativas se optó por crear un sistema a medida con tecnologías de software libre.

El sistema se desarrollara utilizando java, javascript y html.

Para la parte de la vista se utilizará la librería jquery y bootstrap ya que facilitan la parte de diseño para la interacción con el usuario. En cuanto la implementación del sistema de información geográfica se utilizarán las librerías de open layers, leaflet y se integrará con funcionalidades provistas por la api de Google maps.

Como gestor de base de datos se utilizará MySql ya que cumple con las prestaciones necesarias.

## **1.12. Arquitectura.**

La arquitectura describe el conjunto de patrones y abstracciones utilizados para procesar la información y resolver los conflictos.

Se utilizará arquitectura de tres niveles, en donde la carga se divide en 3 capas. Una para la presentación o interfaz de usuario, otra para el modelo de negocio y otra para la persistencia. Cada capa sólo tiene relación con la siguiente.

Podemos dividir los patrones de diseño en:

- Patrones Creacionales, con ellos se pretende separar la lógica de creación de objetos y encapsularlos. Se intenta trabajar con interfaces para lograr que la implementación concreta quede aislada.
- Patrones de Comportamiento, para definir como las clases y objetos interactúan. Gestionan algoritmos, relaciones y responsabilidades entre objetos. Se utilizará el patrón Iterator, permite moverse por los elementos de un conjunto secuencialmente sin tener que exponer la implementación, ejemplo: listas.
- Patrones Estructurales, para crear clases u objetos incluidos dentro de estructuras más complejas. Especifican la forma en que se relacionan las clases. Se implementará el patrón estructural Fachada, para tener una interfaz unificada para acceder de forma simple a otro grupo de interfaces o subsistemas.

Se utilizará el patrón DAO para la persistencia (data access object). Va a manejar el acceso a la información de la base de datos, permite separar la lógica de acceso a datos de la persistencia de objetos, lo cual aporta flexibilidad al momento de hacer un cambio en la lógica de negocio.

La clase DAO encapsulará todos los accesos a la fuente de datos, desacoplando la lógica de negocio de la de acceso a datos.

Se utilizará Herencia para aprovechar la reutilización y extensión del código implementado.

- Se usará el modelo vista controlador (MVC), ya que permite separar los datos de la lógica de negocio. El patrón se basa en la reutilización de código y en la separación de objetos. En donde:
  - La vista presenta al modelo, entonces le solicita a este la información que debe representar y la envía al usuario.
  - El controlador responde a eventos, acciones de los usuarios y realiza peticiones al modelo.
  - El modelo es la representación de los datos, por lo cual gestiona el acceso a la información, envía la información a la vista y las peticiones llegan a través del controlador.

Con la aplicación de estos patrones y técnicas se pretende reducir acoplamiento y lograr una alta cohesión para aumentar robustez, fiabilidad y reutilización del código.

## **1.13. Análisis de factibilidad.**

El análisis de factibilidad sirve para guiar la toma de decisiones en el transcurso del proyecto. Se intenta tener previstos todos los componentes necesarios para que se pueda realizar el proyecto, lo cual implica la consideración de riesgos.

### **1.13.1. Operativa.**

Al ser un desarrollo web se necesitan como recursos disponibles para el funcionamiento del sistema conexión a internet y un navegador web, preferentemente Google Chrome, Firefox u Opera.

El sistema será multiplataforma e independiente de los demás sistemas con que se cuenten.

Es imprescindible que el uso sea intuitivo y simple.

Se pretende que no haya necesidad de confeccionar un manual de usuario y que con la descripción de las funcionalidades sea suficiente para poder utilizar el sistema eficientemente.

### **1.13.2. Técnica.**

El sitio no requerirá grandes requerimientos de ancho de banda.

La aplicación web estará activa las 24hs. permitiendo trabajar a distancia sin importar la zona horaria en que se encuentre la persona.

Como sistema de gestión de bases de datos relacional se utilizara MySQL y en caso de requerir PostgreSQL con PostGIS, dependerá de la información espacial con que se deba trabajar. Por lo que el servidor en el cual se alojará la aplicación debe tener instalado MySQL Server 5.7 en adelante.

PostGIS convierte el sistema de bases de datos PostgreSQL en una base de datos espacial mediante la adición de 3 características: tipos de datos espaciales, índices espaciales y las funciones que operan sobre ellos.

Se utilizará apache tomcat, apache como servidor de aplicaciones web donde se va a alojar el sistema de información y tomcat será el contenedor de servlets.

Apache Tomcat funciona como un contenedor de servlets. Tomcat implementa las especificaciones de los servlets y de JavaServer Pages (JSP), entonces es un contenedor web con soporte de servlets y JSPs. Tomcat puede funcionar como servidor web por sí

mismo. Dado que Tomcat fue escrito en Java, funciona en cualquier sistema operativo que disponga de la máquina virtual Java.

La aplicación correrá correctamente en cualquier navegador con soporte HTML5, es conveniente que tenga un explorador actualizado.

Es necesario un visor de archivos pdf para poder visualizar correctamente los distintos documentos y un software que pueda trabajar con hoja de cálculo o planilla electrónica.

### **1.13.3. Legal.**

Los únicos requisitos que tiene el proyecto desde el punto de vista legal es cuidar la seguridad del sitio administrativo y la seguridad de los datos personales de los usuarios. Para esto las contraseñas serán encriptadas.

Las herramientas a utilizar serán de licencia libre.

### **1.13.4. Económica.**

Actualmente se cuenta con dominio, hosting y pcs por lo cual no habrá que realizar ninguna inversión económica.

## **1.14. Análisis de riesgo.**

Se considerara los riesgos en función de su probabilidad de ocurrencia y su gravedad. Para los de mayor gravedad y con mayor probabilidad de ocurrencia se realizarán planes de contingencia, a fin de mitigar su impacto en caso de suceder.

Rangos de probabilidades (Sommerville)	
Muy baja	<10%
Baja	10-25%
Moderada	25-50%
Alta	50-75%
Muy alta	>75%

Tabla 1. Rangos de probabilidades (Sommerville).

<b>Valores de impacto (Sommerville)</b>
Catastrófico
Serio
Tolerable
Insignificante

Tabla 2. Valores de impacto (Sommerville).

A continuación se detallan los principales riesgos detectados que pueden hacer que el proyecto no se desarrolle en el tiempo requerido y con la calidad deseada. Para estos problemas es necesario tener soluciones planificadas a modo que si aparecen se tenga un plan de contingencia. Se llegó a estos riesgos evaluando la exposición al riesgo (ER) para lo que se consideró la probabilidad de ocurrencia del riesgo y el impacto que tendría en el proyecto.

$$ER = \text{Probabilidad de ocurrencia} \times \text{Impacto (pérdida estimada)}$$

<b>Principales Riesgos (Riesgo, Probabilidad e Impacto)</b>			
<b>Id</b>	<b>Riesgo</b>	<b>Probabilidad</b>	<b>Impacto</b>
1	Cambios en los requerimientos.	Moderada	Serio
2	Mala comprensión de los requerimientos.	Moderada	Serio
3	Perdida de documentación.	Muy baja	Tolerable
4	Falta de acceso al cliente.	Baja	Serio
5	Falta de experiencia en proyectos reales.	Moderada	Tolerable
6	El tiempo requerido para el desarrollo del sistema es subestimado.	Alta	Catastrófico
7	Dificultad en el control de calidad.	Bajo	Tolerable
8	Herramientas seleccionadas para el proyecto incorrectas.	Muy Baja	Tolerable
9	Cambio de prioridades del cliente.	Baja	Serio

Tabla 3. Principales Riesgos.

### **1.14.1. Planes de contingencia y mitigación.**

#### **1. Cambios en los requerimientos.**

Como plan de contingencia se realizará un esquema de requerimientos y se verificarán en el desarrollo del proyecto en conjunto con el cliente.

En caso que igual ocurra el riesgo se cuenta con días extras previstos para mitigar el impacto y afrontar los cambios.

#### **2. Mala comprensión de los requerimientos.**

Al finalizar cada sprint se tendrá un entregable para los grupos de testeo (cliente y grupo objetivo) lo cual servirá para ir minimizando riesgo de errores en los requerimientos y por tanto en las funcionalidades. Además en las reuniones con el cliente se mostrarán los avances.

En caso que igual ocurra el riesgo se cuenta con días extras previstos para mitigar el impacto y corregir los errores.

#### **3. Perdida de documentación.**

Se implementarán respaldos obligatorios además de los respaldos regulares que se consideren necesarios de los avances del proyecto. Los respaldos se podrán hacer en drive en la nube y el código mediante sourcetree en un repositorio en bitbucket.

En caso que igual ocurra el riesgo se cuenta con la posibilidad de contactar con el cliente para poder reconstruir la documentación y mitigar el impacto.

#### **4. Falta de acceso al cliente.**

Se establecerán reuniones con el cliente, además de contar con otros medios de comunicación como ser: teléfono, mail y video conferencia.

En caso que ocurra se habló con el cliente que se continuará el desarrollo con los lineamientos y requerimientos acordados hasta el momento.

#### **5. Falta de experiencia en proyectos reales.**

Durante todo el proceso se contará con la asistencia del tutor, además en cada etapa del desarrollo se investigará y se contará con material de consulta especializado en el tema.

#### **6. El tiempo requerido para el desarrollo del sistema es subestimado.**

Se contará con un cronograma de actividades para administrar el tiempo. En caso de atrasarse alguna etapa se deberá aumentar la carga horaria, de darse un caso crítico se dejan los sábados y domingos con horas extras para dedicar al proyecto. Dependiendo de la etapa del proyecto que se este, también se contará con días previstos para contingencia.

#### **7. Dificultad en el control de calidad.**

Además de las pruebas estipuladas en el plan de testing, se harán pruebas de aceptación por parte del cliente y de los usuarios.

En caso que igual ocurra el riesgo se cuenta con la posibilidad de poder implementar algún sistema de automatización como Selenium y se cuenta la asesoría del tutor para poder mitigar el impacto.

#### **8. Herramientas seleccionadas para el proyecto incorrectas.**

La parte más delicada sería la integración con el sistema de información geográfica pero al contar con la guía del tutor y la investigación realizada es muy baja la posibilidad que las herramientas seleccionadas para el proyecto sean una dificultad.

En caso que suceda el riesgo se cuenta con días extras para mitigar el impacto.

#### **9. Cambio de prioridades del cliente.**

El cambio en las prioridades del cliente tendría un serio impacto en el proyecto. En caso de darse, según en las funcionalidades que repercutan, deberá evaluarse la posibilidad de realizar los cambios necesarios. A pesar de esto, al haber echo un análisis desde un punto de vista sistémico y en conjunto con el cliente se estima como bajo este riesgo.

## **1.15. Plan de Proyecto.**

El plan de proyecto comprende las acciones necesarias para llevar a cabo los objetivos de forma eficiente. Es lo que permitirá obtener los resultados deseados.

### **1.15.1. Definición del proceso.**

El proyecto se desarrollará tomando conceptos de la metodología SCRUM y como producto se obtendrá un desarrollo en ambiente web y una aplicación nativa para móviles.

#### **1.15.1.1. Metodología.**

Para el desarrollo del proyecto se utilizarán metodologías ágiles y se implementará una adaptación del marco de trabajo de la metodología SCRUM acorde a las características del proyecto.

#### **1.15.1.2. Ciclo de vida.**

El ciclo de vida se llevará a cabo con la adaptación del marco de trabajo SCRUM, por lo que comenzará con una lista de requerimientos del producto que se desea desarrollar, la cual se define en conjunto con el cliente quien indica las prioridades.

Se irán realizando procesos de entrega cortos (20 días aproximadamente) conocidos como “Sprints”. Luego de definidas las tareas se definirá como se van a realizar con su respectiva estimación de tiempo.

Al finalizar cada sprint se testea y se revisa el diseño.

El ciclo de vida de este proyecto terminaría con la entrega del producto.

Para planificar las tareas se asignó una duración temporal en base a la complejidad de la tarea a desarrollar y la experiencia del equipo de trabajo, por todo esto se tiene en cuenta un margen de tiempo disponible para investigación y por los posibles imprevistos que pudieran surgir.

Se detallará la planeación del proyecto mediante la lista de tareas con su duración medida en días y mediante el diagrama de Gantt.

En cada entrega al cliente se le expondrán las funcionalidades entregadas.

### 1.15.1.3. Funcionalidades a desarrollar (Pila del Producto o Product Backlog).

ID	Nombre	Fecha de inicio	Fecha de fin	Duración
1	Ideación del proyecto	10/03/18	19/03/18	10
2	Metodología de trabajo y estructura de Anteproyecto	20/03/18	5/04/18	17
3	1ra reunión con el Cliente	5/04/18	5/04/18	1
4	Análisis del problema	7/04/18	7/04/18	1
5	Investigación	7/04/18	10/04/18	4
6	Elaboración de solución	8/04/18	8/04/18	1
7	Presentación del problema	9/04/18	9/04/18	1
8	2da reunión con el Cliente	9/04/18	9/04/18	1
9	Correcciones 1ra y 2da reunión	10/04/18	12/04/18	3
10	Lista de necesidades	12/04/18	15/04/18	4
11	Análisis estratégico	14/04/18	16/04/18	3
12	Estudio de objetivos	14/04/18	16/04/18	3
13	Alcance y limitaciones	16/04/18	18/04/18	3
14	Lista de requerimientos	17/04/18	18/04/18	2
15	Análisis de factibilidad	19/04/18	22/04/18	4
16	Análisis de riesgo	20/04/18	20/04/18	1
17	Plan del proyecto	21/04/18	24/04/18	4
18	ABM- Productos	24/04/18	15/05/18	22
19	Carrito de compras	24/04/18	16/05/18	23
20	Configuración entorno de desarrollo	25/04/18	25/04/18	1
21	Configuración de ambiente de testing	2/05/18	4/05/18	3
22	Presentación Anteproyecto	10/05/18	10/05/18	1
23	Sprint 1- 3ra reunión con Cliente, entrega Anteproyecto	14/05/18	14/05/18	1
24	ABM- Usuarios	14/05/18	17/05/18	4
25	Login	14/05/18	17/05/18	4
26	Correcciones 3ra reunión	15/05/18	16/05/18	2
27	Documentación (se realiza durante todo el proyecto)	24/04/18	25/09/18	155
28	ABM- Empleados	18/05/18	20/05/18	3
29	ABM- Proveedores	18/05/18	20/05/18	3
30	Realizar pedidos	18/05/18	22/05/18	5
31	Agendar envíos	23/05/18	26/05/18	4
32	Visualización de pedidos	27/05/18	28/05/18	2
33	Visualización de factura	29/05/18	31/05/18	3
34	Stock mínimo	1/06/18	8/06/18	8
35	Sprint 2- 4ta reunión con Cliente, entrega avance 3ra reunión	4/06/18	4/06/18	1

36	Aviso entrega realizada	9/06/18	10/06/18	2
37	Mantenimiento ordenes de trabajo	11/06/18	15/06/18	5
38	Sistema de alertas	16/06/18	17/06/18	2
39	Búsquedas	18/06/18	22/06/18	5
40	5ta reunión con Cliente, entrega avance 4ta reunión	20/06/18	20/06/18	1
41	Visualización información en mapa	23/06/18	15/07/18	23
42	Sprint 3- 6ta reunión con Cliente, entrega avance 5ta reunión	16/07/18	16/07/18	1
43	Sistema fidelización	17/07/18	25/07/18	9
44	Calificar	26/07/18	29/07/18	4
45	Sprint 4- 7ma reunión con Cliente, entrega avance 6ta reunión	30/07/18	30/07/18	1
46	Mostrar servicios contratados	31/07/18	5/08/18	6
47	Sprint 5- 8va reunión con Cliente, entrega avance 7ma reunión	6/08/18	6/08/18	1
48	Emisión de informes	4/08/18	19/08/18	16
49	Sprint 6- 9na reunión con Cliente, entrega avance 8va reunión	20/08/18	20/08/18	1
50	Optimización diseño, interfaz de usuario	21/08/18	25/09/18	36
51	Presentación	18/09/18	24/09/18	7
52	Sprint 7- 10ma reunión con Cliente, entrega del proyecto	26/09/18	26/09/18	1

Tabla 4. Funcionalidades a desarrollar.

#### 1.15.1.4. Diagrama de Gantt.

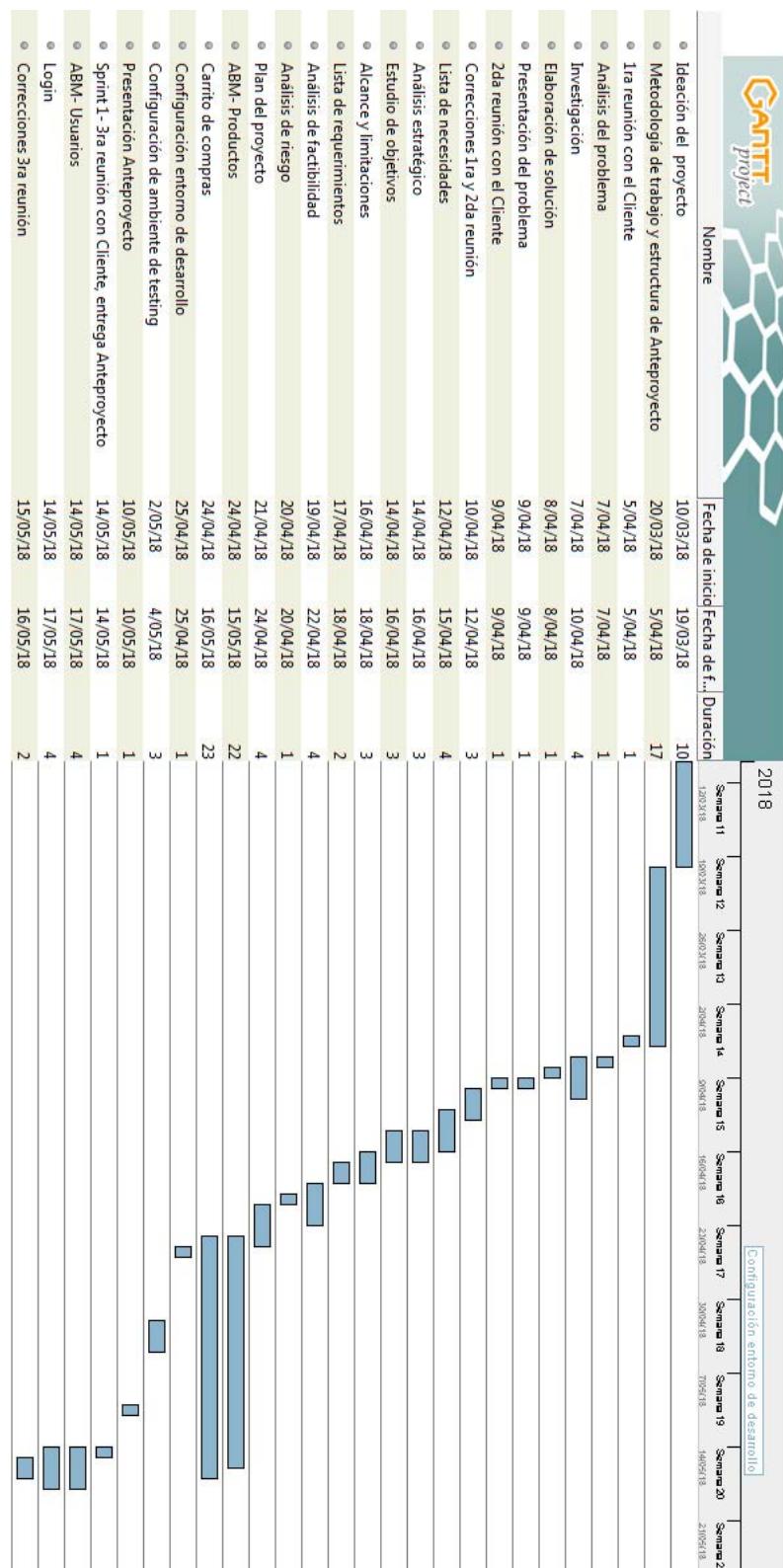


Ilustración 1. Diagrama de Gantt.

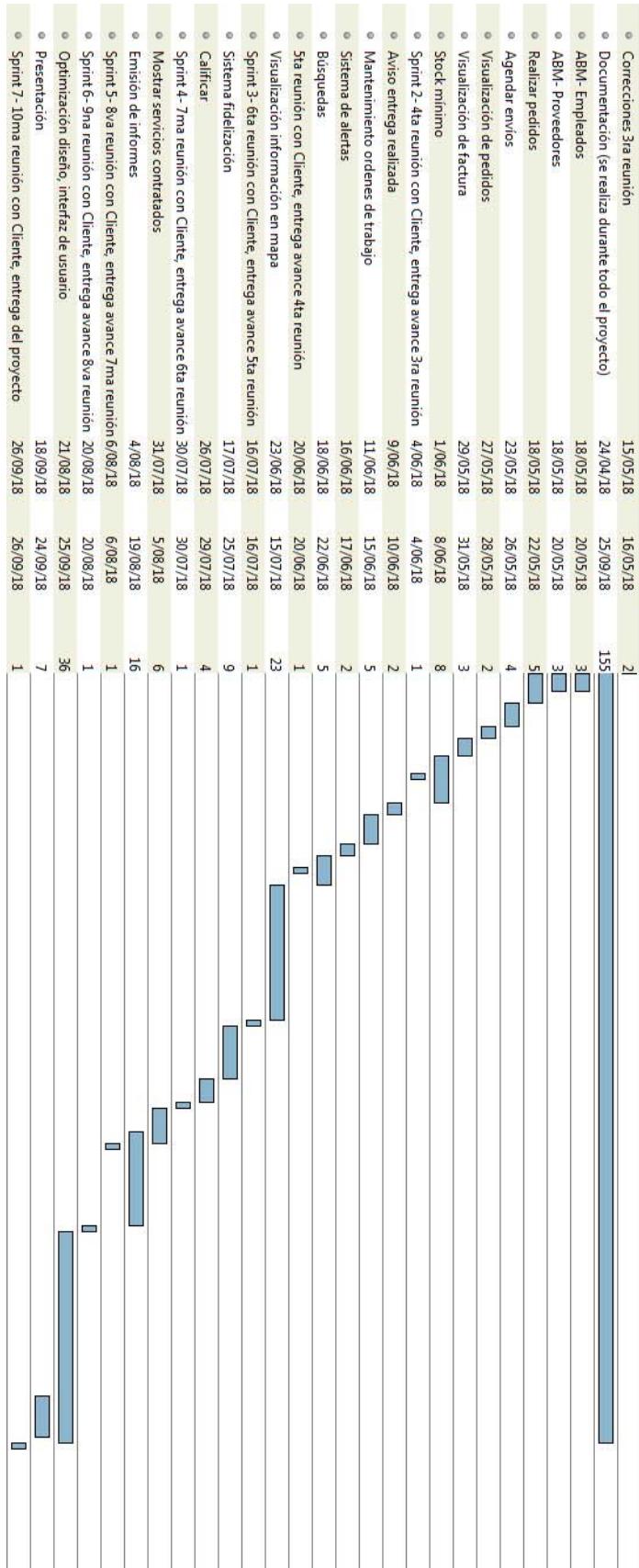


Ilustración 2. Diagrama de Gantt.

## 1.15.2. Descripción de la estructura del proyecto.

Modelo entidad relación y normalización.

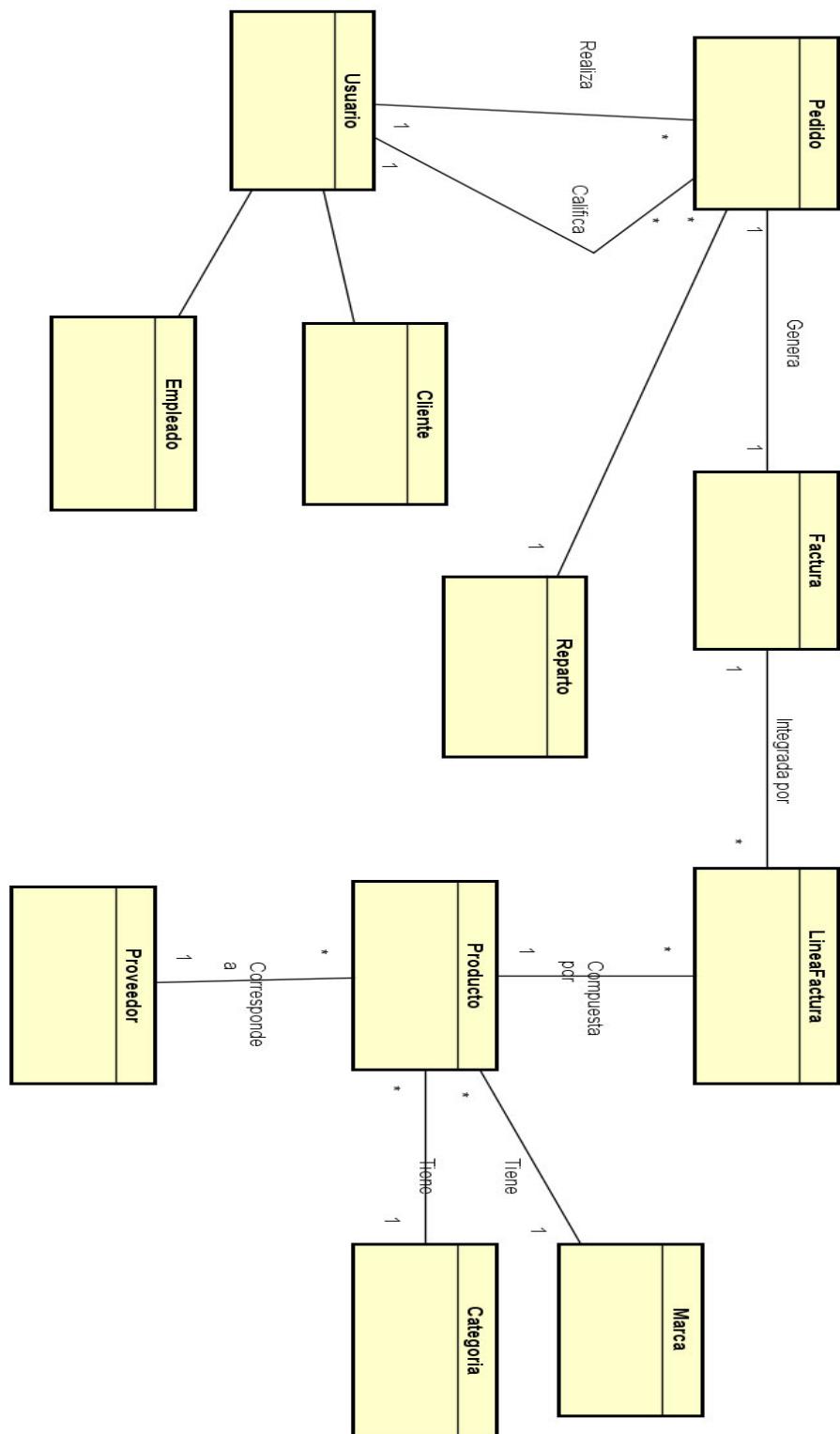


Ilustración 3. M.E.R. (Modelo Entidad Relación).

El UML presentado es una primera aproximación a la solución del problema el mismo se ajustará en el desarrollo del proyecto.

### Producto.

(id\_producto, nombre, proveedor, lote, precio, precio\_oferta, stock, stockminimo, nuevo, id\_proveedor, recomendado, visible, detalle, id\_categoria, id\_marca, fechaIngreso, fotoProducto )

Clave primaria: id\_producto

Unique Key: nombre, proveedor, lote

Clave foránea: id\_categoria de la tabla Categoría, de la tabla Marca id\_marca y de la tabla Proveedor id\_proveedor

### Categoría.

( id\_categoria, nombre, visible, es\_subcategoria\_de )

Clave primaria: id\_categoria

### Marca.

( id\_marca, nombre, visible )

Clave primaria: id\_marca

### LíneaFactura.

(id\_lineaFactura, id\_producto, cantidad, id\_factura )

Clave primaria: id\_lineaFactura

Clave foránea: id\_factura de la tabla Factura y de la tabla Producto id\_producto

### Factura.

( id\_factura, observacion, id\_pedido )

Clave primaria: id\_factura

Clave foránea: id\_pedido de la tabla Pedido

Pedido.

(id\_pedido, fechaSolicitud, departamento, calle, numeroPuerta, fechaEntrega, horarioEntrega, observacion, id\_usuario, id\_reparto, estado, calificacion, latitud, longitud)

Clave primaria: id\_pedido

Clave foránea: id\_usuario de la tabla Usuario y de la tabla Reparto id\_reparto

Usuario.

(id\_usuario, nombre, apellido, password, email, departamento, calle, numeroPuerta, telefono, latitud, longitud, fechaIngreso, fechaNacimiento)

Clave primaria: id\_usuario

Unique key: email

Cliente.

(id\_usuario, personaFisica )

Clave primaria: id\_usuario

Clave foránea: id\_usuario hereda de Usuario

Empleado.

(id\_usuario, puesto, sueldo, cedula)

Clave primaria: id\_usuario

Clave foránea: id\_usuario hereda de Usuario

Reparto.

( id\_reparto, zona, vehiculo, max\_pedidos)

Clave primaria: id\_reparto

Proveedor.

(id\_proveedor, nombre, email, departamento, calle, numeroPuerta, latitud, longitud, telefono, ruc)

Clave primaria: id\_proveedor

### **1.15.3. Integrantes y roles.**

Tomando como base que el proyecto se desarrollará con el modelo SCRUM como referencia.

### **1.15.4. Descripción y selección de herramientas.**

Se desea implementar metodologías ágiles e implementar todo lo que se pueda del marco de trabajo Scrum.

Tecnologías y herramientas:

- Java
- Javascript
- html
- ajax
- json
- jdbc (mysql)
- servlets
- apache tomcat
- bootstrap
- jquery y jquery mobile
- ccs
- bitbucket
- geojson
- openlayers. Librería javascript
- Leaflet. Librería javascript

Plataformas, ides:

- Netbeans
- mysqlworkbench
- xampp
- sourcetree
- ganttpoint
- postman. Api development environment
- Advanced rest client para google chrome
- Qgis. Sistema de información geográfica de escritorio

El sistema se dividirá en dos grandes ramas, una parte de la aplicación sería de acceso público y otra para acceso de los funcionarios.

La solución se realizará con herramientas libres, lo cual posibilitará tener un completo acceso en la creación y actualización de la misma, permitiendo minimizar limitantes de recursos y dependencia de terceros. De cada librería y tecnología utilizada se incluye la licencia correspondiente.

Como lenguaje base se pretende usar Java y MySQL como base de datos.

### Lenguaje. Java

Se optó por Java ya que es un lenguaje multiplataforma debido a que es independiente de la plataforma en la cual se ejecute, flexible (se adapta a todo tipo de dispositivos móviles) permite integración con otras herramientas, es libre, posee soporte, amplia documentación y variada opciones de frameworks en caso de considerarse pertinente.

### Base de Datos. MySQL

MySQL es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual GPL/Licencia comercial por Oracle Corporation y está considerada como la base de datos open source más popular del mundo y una de las más populares junto a Oracle y Microsoft SQL Server, sobre todo para entornos de desarrollo web.

Tomando como referencia lo expresado en <https://es.wikipedia.org/wiki/MySQL>, MySQL cuenta con:

- Amplio subconjunto del lenguaje SQL.
- Disponibilidad en gran cantidad de plataformas y sistemas.

- Conectividad segura.
- Replicación.
- Búsqueda e indexación de campos de texto.
- Posibilidad de selección de mecanismos de almacenamiento que ofrecen diferentes velocidades de operación, soporte físico, capacidad, distribución geográfica, transacciones, entre otras.”

### Entorno de Desarrollo (IDE). Netbeans

Como entorno de desarrollo se usará Netbeans, es de licencia libre y es un entorno sencillo, amigable, potente y tiene amplia documentación.

Datos de la versión: Netbeans IDE. Version: 8.2.

### Documentación y planificación

Para la documentación se usará LibreOffice para trabajar de forma local y Google docs en forma remota, debido a que ofrece la posibilidad de tener la documentación en línea, fácil acceso, trabajar en forma conjunta con otras personas. Además posee varias herramientas como Google drive (repositorio) en donde se tiene acceso a un disco donde guardar toda la información necesaria para tener acceso por Internet.

También se cuenta con un calendario donde poder hacer la planificación en el cual se pueden ir siguiendo el desarrollo del proyecto.

### Diagramado

Se utilizará el software Umbrello UML Modeller, versión 2.24.0, como herramienta para el desarrollo UML y demás diagramas. La elección se debe a que el equipo ya cuenta con experiencia en el mismo y es de licencia libre.

### Diagramas de Gantt

Se utilizará el software: GanttPV 0.11b.

### Librerías:

- Para subir imágenes:
  - fileupload.jar
  - io.jar
- Para encriptar md5 password:
  - commons-codec-1.11.jar

- Para trabajo con mails:
  - Javax.mail.jar
- Para Base de datos:
  - MySQL JDBC Driver
  - PostgreSQL JDBC Driver
- Servicio Rest:
  - Java EE Web 7 API Library
  - Jersey 2.5.1
- Servidor:
  - Apache Tomcat or TomEE

### **1.15.5. Plan de Calidad (SQA).**

El SQA (Software Quality Assurance, Garantía de Calidad del Software) consiste en el diseño de un plan para monitorear el proceso de desarrollo asegurando la calidad del producto.

Para el plan de calidad del proyecto se decide dividir las pruebas teniendo por un lado las que realiza el grupo objetivo y el cliente y por otro las que corresponden al equipo de desarrollo. Entonces se delegan tareas de testing al grupo objetivo y al cliente ya que esto aumentará la calidad del producto, debido a que será testeado bajo distintos marcos de referencia y contextos, tantos y tan diversos como las personas que realicen las pruebas. Además representará un importante ahorro de tiempo, el cual se empleará para mitigar los problemas que vayan surgiendo.

En el proyecto se entiende calidad como: el conjunto de propiedades que permite evaluar su funcionamiento, siendo este de calidad si logra satisfacer las necesidades, tanto implícitas como explícitas y por lo tanto es una herramienta que aporta valor al proceso en el que se aplique. Logrando ahorrar tiempo, automatizar tareas repetitivas y simplificar el funcionamiento para la realización de las tareas, entendiendo que para esto el uso debe resultar intuitivo.

Durante el proyecto se adoptarán conceptos del modelo de desarrollo SCRUM ya que, es un modelo de referencia que determina un marco de desarrollo ágil. Se caracteriza por adoptar una estrategia de desarrollo incremental, basar la calidad del resultado más en el conocimiento tácito de las personas en equipos auto organizados, que en la calidad

de los procesos implementados, hay solapamiento de las distintas fases del desarrollo (síncronas) en lugar de realizarlas en forma secuencial o en cascada

Un principio clave de Scrum es reconocer que durante un proyecto los clientes pueden cambiar de idea sobre lo que quieren y necesitan, ya que es normal que no tengan claro lo que necesitan y que la realidad está en un cambio constante. Por lo tanto, Scrum adopta una aproximación pragmática, aceptando que el problema no puede ser completamente entendido o definido, y centrándose en maximizar la capacidad del equipo de entregar rápidamente y responder a requisitos emergentes.

### **1.15.5.1. Plan de testing.**

El objetivo del Plan de Testing es encontrar errores desde fases tempranas del desarrollo del ciclo de vida del proyecto, para esto con la culminación de cada sprint quedará culminada una versión pasable de testeo por el cliente y el grupo objetivo.

Se definirán casos de prueba que permitan verificar el cumplimiento de los requerimientos de cada prueba. Se realizarán los siguientes tipos de test:

- Test de funcionalidad.
- Test de sistema.
- Test de aceptación del cliente y usuarios con el fin de descartar problemas en la interpretación de los requerimientos o que estos sean inadecuados. Para esto el cliente y los usuarios validarán el funcionamiento del sistema.

Los tests se realizarán empezando sobre los algoritmos de mayor criticidad. En el plan de pruebas del proyecto se adjuntará la evidencia de ejecución de dichos planes.

#### **1.15.5.1.1. Test de caja negra.**

Lo realizará cliente y grupo objetivo.

Se denomina Caja Negra a aquel elemento que es estudiado desde el punto de vista de las entradas que recibe y las salidas o respuestas que produce, sin tener en cuenta su funcionamiento interno.

En otras palabras, de una caja negra nos interesarán su forma de interactuar con el medio que le rodea (en ocasiones, otros elementos que también podrían ser cajas negras) entendiendo qué es lo que hace, pero sin dar importancia a cómo lo hace.

Por tanto, de una caja negra deben estar muy bien definidas sus entradas y salidas, es decir, su interfaz; en cambio, no se precisa definir ni conocer los detalles internos de su funcionamiento.

### **1.15.5.1.2. Test de caja blanca.**

Realizado por desarrollador.

Su cometido es comprobar los flujos de ejecución dentro de cada unidad (función, clase, módulo, etc.) pero también pueden testear los flujos entre unidades durante la integración, e incluso entre subsistemas, durante las pruebas de sistema. El tester escoge distintos valores de entrada para examinar los posibles flujos de ejecución y cerciorarse que se devuelven los valores de salida correctos.

Al estar basadas en una implementación concreta, si ésta se modifica, por regla general las pruebas también deberán rediseñarse.

### **1.15.6. Estándares definidos y convenciones.**

Todas las funcionalidades implementadas deberán tener un nombre que actúe como descriptor de su funcionamiento y en algunos casos, de ser necesario, se agregarán comentarios explicativos.

Se deberá identar correctamente el código.

### **1.15.6.1. Documentación.**

Para la documentación se utilizaran los estándares definidos por la Universidad en los documentos:

- Documento 302 (Universidad ORT Uruguay): Normas específicas para la presentación de trabajos finales de carrera (TFDC) de la Facultad de Ingeniería.

Motivo: Es requisito cumplir con los estándares que describe para presentar trabajos académicos.

- Documento 303 (Universidad ORT Uruguay): Lista de Verificación del Formato.

Motivo: Sirve para corroborar que se haya cumplido con los estándares definidos en el documento 302.

- Documento 306 (Universidad ORT Uruguay): Guías para títulos, abstracts e informes de corrección.

Motivo: Es requisito cumplir con los estándares que describe para presentar trabajos académicos.

- Guía de proyectos Analista Programador.  
<https://aulas.ort.edu.uy/mod/folder/view.php?id=81654>

### **1.15.6.2. Nomenclatura.**

Los nombres a asignar a variables, constantes, funciones, entidades, etc., deben ser representativos y dar información de la tarea que realizan. Para los cuales no se utilizarán tildes ni la letra “ñ”, cuando una palabra esta compuesta por “ñ” en su lugar se usará “ni”.

Cuando el nombre esté formado por más de una palabra, cada palabra debe empezar con mayúscula, de ser necesario para mayor claridad se podrá separar por infra guión (“\_”).

Para:

- campos de texto se utilizará el prefijo “txt” más el nombre del dato que contendrá (Ej.: txtDireccion).
- botones se utilizará el prefijo “btn” más el nombre de la acción que realiza (Ej.: btnAgregar).
- formularios se utilizará el prefijo “form” más el nombre que relaciona al formulario con su función (Ej.: formClientes).
- etiquetas se utilizará el prefijo “label” seguido del nombre del dato que contendrá (Ej.: labelEdad).

#### En la base de datos:

Los nombres deben tener relación con los datos que guardan.

Los nombres de las tablas serán en singular. Un concepto que esté en varias tablas debe ser llamado en todas con el mismo nombre.

Si la tabla es resultado de la relación de dos o más tablas se utilizará el carácter \_ (guión bajo) para unir los nombres de éstas.

## **1.15.7. Plan SCM (Software Configuration Management).**

La gestión de configuración de software controla la forma de elaboración del código, el seguimiento del estado de las fases del desarrollo, la integración de los distintos módulos del producto y el plan de capacitación.

### **1.15.7.1. Gestión de configuración.**

La configuración del sistema se realizará en conjunto con el cliente. Con esto se logra minimizar los errores que puedan producirse por los cambios de ambiente y cumplir con las pruebas tempranas.

Para la configuración se instala un servidor Apache tomcat con base de datos MySQL. Se corre el script para la creación de la base de datos y se sube el sistema al servidor. Para el correcto funcionamiento también se controla que los exploradores web estén actualizados y que se cuente con el software apropiado para poder trabajar con archivos pdf y hojas de cálculo.

### **1.15.7.2. Control de versionado.**

Para el control, gestión de versionado y realizar de forma ordenada el desarrollo del código se utilizará la herramienta Sourcetree y un repositorio en Bitbucket. Ya que está pensando para la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones y se cuenta con repositorios privados gratuitos.

Cada día se debe subir al repositorio lo realizado, lo cual se realiza mediante la herramienta Sourcetree.

Para la documentación también se controlará el versionado mediante la utilización de GoogleDrive.

### **1.15.7.3. Plan de capacitación.**

Se pretende que el uso del sistema sea lo más intuitivo posible, minimizando la necesidad de capacitación tanto para el cliente como para el usuario.

De todas formas se trabajará en conjunto con el cliente a fin de poder abordar todas las inquietudes.

Al ser intuitivo y simple el sistema, la sección de descripción de funcionalidades (2.4) actuará como manual de usuario en conjunto con los casos de uso y el proceso de pruebas.

## **2. Proyecto.**

### **2.1 Solución lograda.**

En el transcurso del desarrollo del proyecto se vio que varias de las funcionalidades requeridas implicaron mayor dificultad de la estimada y también requirieron desarrollar otras funcionalidades no previstas al inicio del proyecto, por lo cual, la criticidad de las tareas debió ser modificada respecto a la original. Todo esto se especifica en el desarrollo de los Sprints. Se definió un plan de acción con el fin de realizar una óptima gestión de los recursos.

A pesar de todo lo realizado quedaron cosas a mejorar en futuras versiones como las funcionalidades en las gráficas y en el sistema de información. También es factible de mejora la presentación de informes, delimitación y adjudicación de zonas de repartos, optimizar rutas de reparto e incluir la posibilidad de análisis de rutas. También se puede mejorar en la refactorización de código y la presentación de los pdfs.

Para futuras versiones sería de interés permitir que al agregar un nuevo reparto se puedan delimitar nuevas zonas para los repartos en forma automática, aunque al momento no es algo necesario para el cliente.

También restaría implementar hacer un seguimiento de los mensajes que se reciben y envían para tener información que permita generar un sistema de respuestas automáticas por el sistema. Otra funcionalidad a seguir desarrollando sería en la optimización del geocodificador.

Como principal logro a destacar es que se cumplieron los objetivos y que el cliente se manifestó conforme y entusiasmado para seguir trabajando en la mejora del proyecto

Se pudo construir una primera versión de un sistema que se autogestiona, permitiendo automatizar tareas y hacer más eficiente la exposición de la información para la toma de decisiones. Debido a esto no se necesita de usuario especializado para su mantenimiento. Dicha aplicación sirve como punto de partida ya que a raíz de lo desarrollado surgieron nuevas ideas para seguir mejorando funcionalidades y la lógica del negocio.

En la sección de requerimientos funcionales se detalla las funcionalidades efectivamente desarrolladas en la solución lograda.

#### **2.1.1 Diagrama de clases.**

Se modificó el diagrama y la estructura de las tablas en la base de datos respecto a la versión original presentada en el anteproyecto.

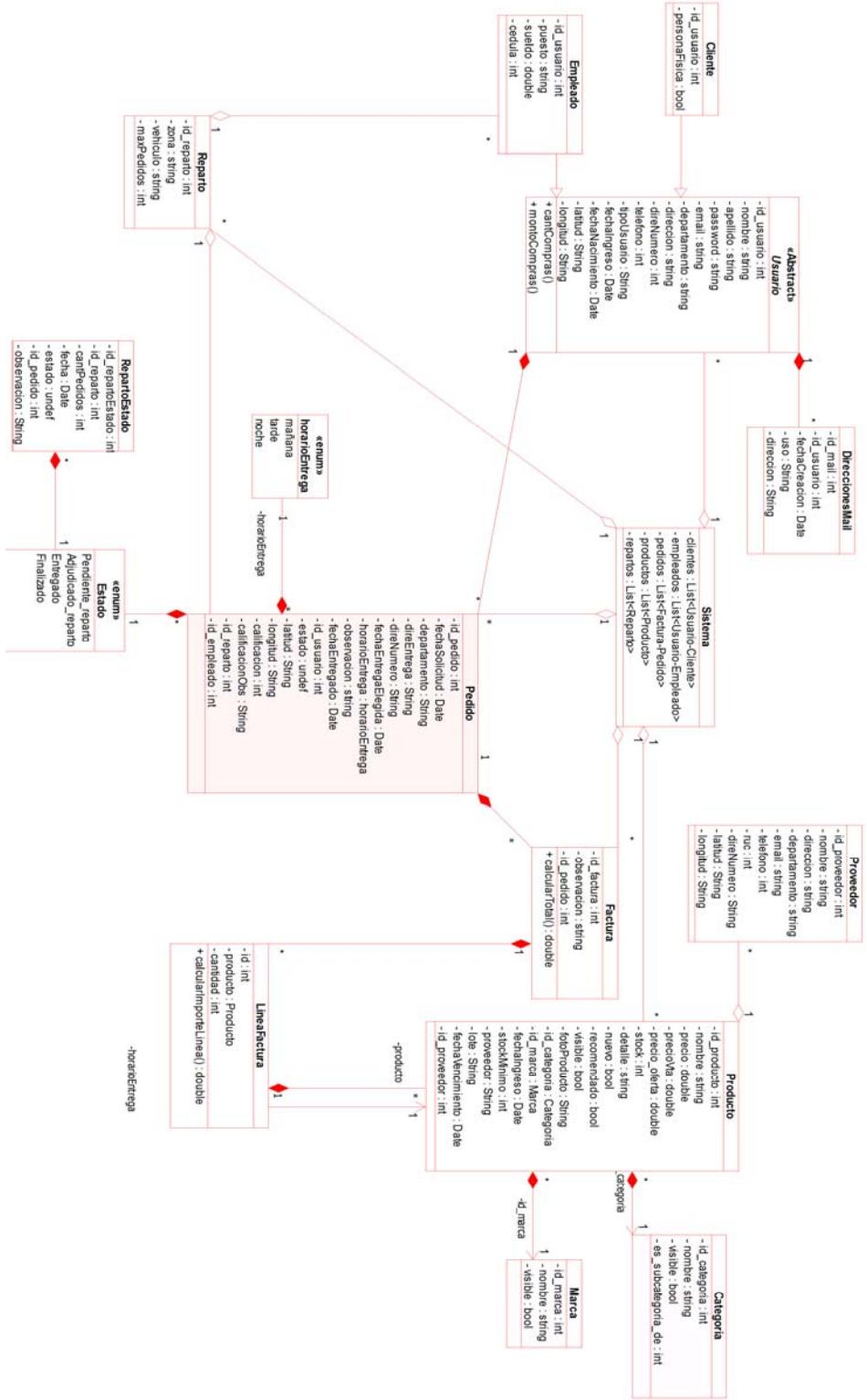


Ilustración 4. Diagrama de clases.

En la sección de “Requerimientos funcionales” se describen las funcionalidades que se desarrollaron en el proyecto, se mantuvieron los requerimientos no funcionales planteados en el anteproyecto.

### **2.1.2 Tecnologías utilizadas.**

Las principales tecnologías utilizadas para el desarrollo del proyecto fueron Java como lenguaje de programación para la lógica, JavaScript para el desarrollo del sistema de información geográfica, interfaces de usuario y validaciones, HTML para interfaces de usuario.

También se uso template AdminLTE y jquery, jquerymobile y leaflet como librerías para la interfaz de usuario.

Para la representación de la información espacial se uso GeoJSON para codificar las estructuras de datos geográficos

Como motor de base de datos se utilizó MySQL y Apache Tomcat como servidor.

## **2.2 Requerimientos funcionales.**

- I. **Login.** El sistema debe restringir el acceso mediante login, desde donde se identifica el ingreso al sistema, si el usuario es un cliente se redirige a la pagina donde se encuentran los productos y si es un empleado se redirige al menú administrador. Los usuarios se loguean con mail y contraseña.
- II. **El sistema sólo debe permitir comprar a usuarios registrados.** Se controla el tipo de usuario, a fin de saber que funcionalidades habilitar al mismo. Pueden ser usuarios clientes o empleados.

En la página principal el usuario cuenta con acceso al listado de productos y los filtros de búsqueda, también se cuenta con acceso a la página de contacto, la posibilidad para ver los pedidos vigentes y ver los pedidos finalizados.

Además se tiene acceso al carrito de compras y la posibilidad de modificar los datos del usuario

- III. **El sistema debe permitir realizar mantenimientos** de usuarios, empleados, productos, pedidos, repartos, marcas, categorías y proveedores. Esto incluye funcionalidades para realizar alta, baja y actualización de datos.
  - a. Los usuarios se dividen en clientes y empleados.

- b. Los usuarios se identificarán por una clave primaria, numérica y auto incremental. Los datos a registrar de los clientes son: nombre, apellido, clave de acceso al sistema (debe estar encriptada), email (será único para cada usuario), dirección (se compone por el departamento, calle, número de puerta, latitud y longitud de la ubicación. Estos últimos dos datos los obtendrá el sistema de forma automática) y móvil de contacto.
- c. Además de los datos anteriores los clientes deben poder especificar si son personas físicas o empresas y a los empleados se les registra la cédula, puesto y sueldo.
- d. Los repartos deben contar con un identificador numérico y auto incremental, se registra la zona a la que corresponde el reparto, el vehículo y la cantidad máxima de pedidos que puede realizar.
- e. Los productos se identifican por un id numérico auto incremental, se registra el nombre, lote y marca de cada producto (estos tres ítems en su conjunto serán únicos, no pueden existir dos productos con el mismo nombre, lote y marca), precio del proveedor, precio de venta y precio de oferta. Además se debe tener el stock actual y el stock que determinan la empresa como stock mínimo, a partir del cual se avisa para realizar la reposición del mismo. Se tiene que poder identificar el producto como nuevo, recomendado y también agregar detalle y visibilidad del mismo. El mismo debe contar con foto, la fecha ingreso, la fecha de vencimiento, la categoría y al proveedor que corresponde.
- f. Para las marcas se ingresa el nombre y la visibilidad dependiendo si cuenta o no con productos, debiendo tener un identificador numérico auto incremental.
- g. Las categorías tienen que contar con un identificador igual que las marcas y se debe registrar su nombre, visibilidad y si es subcategoría de otra categoría.
- h. Los proveedores cuentan con un id, numérico y auto incremental, nombre, email, la dirección (compuesta por el departamento, calle, número de puerta, latitud y longitud de la ubicación. Estos últimos dos datos los obtiene el sistema de forma automática), un teléfono y el ruc.
- i. Para los pedidos se registra el id del pedido (auto incremental), la fecha de solicitud, la dirección compuesta por departamento, calle y número de puerta, latitud y longitud de la ubicación. Estos últimos dos datos los obtiene el sistema de forma automática. Además se cuenta con la posibilidad de ingresar una observación. También se debe tener registrado el cliente de la solicitud, el empleado a cargo, a qué reparto corresponde el pedido según la zona del mismo, la fecha y horario de entrega, el estado y la calificación dada por el cliente.

Para ingresar un pedido el cliente sólo tiene disponible para elegir las fechas en que los repartos aún cuentan con lugar para agregar pedidos a entregar. Al ingresar una orden de trabajo (un pedido) se identifica el usuario y la ubicación espacial de la entrega, la misma se asocia al reparto que corresponda según la zona.

- IV. **Alta de clientes, empleados, proveedores.** En estas altas se debe incluir geolocalización para que el cliente pueda ver si fue correcta la ubicación geográfica asociada a la dirección dada.
- V. **Pedidos Vigentes.** Los clientes deben poder visualizar sus pedidos en curso.
- VI. **Pedidos Finalizados.** Los clientes tienen que poder visualizar un histórico de sus pedidos.
- VII. **Vista Facturas.** Los clientes tienen que poder ver la factura asociada a cada pedido.
- VIII. **Ingreso de productos.** Al ingresar el producto si la marca y categoría no existe, en el formulario de ingreso de producto se deberá contar con la opción para ingresar la misma.
- IX. **El sistema no debe permitir** ingresar marcas, categorías y proveedores con el mismo nombre.
- X. **Control de eliminación de información.** Antes de eliminar cualquier información se verifica que no integre ningún trabajo en curso.
- XI. **Baja empleado.** Al dar de baja un empleado no se debe borrar del sistema se deja registrado como cliente.
- XII. **Identificadores.** Cada objeto o dato tiene que incluir identificador.
- XIII. **Carrito de compras.** El sistema debe permitir realizar pedidos.
- XIV. **Filtros.** Para que se pueda usar el carrito de compras el usuario debe estar logueado y pudiéndose seleccionar los productos por filtro de marca o categoría.
- XV. **Control de errores.** Cada formulario tiene que contar con su respectiva validación antes de enviar los datos al servidor.
- XVI. **Visualización de factura de pedidos.** En cada compra realizada queda asociada la factura y los detalles de la compra al usuario en cuestión. La factura debe ser visible para el comprador y para los administradores del sistema.
- XVII. **Funcionalidad de Stock Mínimo.** La funcionalidad de stock mínimo tiene que incluir alertas para saber cuando se está llegando a un número crítico de

productos, el cual es definido por la empresa. Se debe visualizar la cantidad y la valuación del stock.

- XVIII. **Cantidad de pedidos por reparto.** Se deben visualizar todos los repartos y ver la cantidad de pedidos asociado a los mismos. En caso de no tener pedidos igual se tiene que ver el reparto.
- XIX. **Cantidad de pedidos por reparto por fecha.** Se visualizan sólo los repartos con pedidos y se encuentran ordenados por fecha de entrega.
- XX. **Compra empleado.** Los empleados tienen que poder actuar como clientes.
- XXI. **Alta empleado.** Cualquier persona debe poder ser dada de alta como empleado, sin importar si la misma ya fue empleada y actualmente esta como cliente.
- XXII. **Reposición de stock.** Al ingresar la reposición de stock se tienen que ver los datos de nombre del producto, precio del proveedor, precio de venta y stock del lote anterior. En la reposición de stock sólo se ven los productos que están por debajo del stock mínimo y no tienen nuevo lote del mismo producto.
- XXIII. **Producto sin stock.** Cuando un producto queda sin stock y no tiene lote nuevo con stock, debe poder ser visualizado el producto pero no se puede ingresar al carrito de compras.
- XXIV. **Control de stock, stock mínimo.** Cuando un lote de un producto se acaba pero hay un nuevo lote con stock se debe de dejar de visualizar el lote sin stock y se visualiza el que tiene stock. En la vista del usuario nunca se tiene que dejar de ver el producto y tampoco se ve sin unidades disponibles sólo se actualiza la información de un lote al otro.
- XXV. **Stock valuación.** Se deben poder ver todos los productos.
- XXVI. **Mantenimiento de órdenes de trabajo.** En caso de que no se pueda entregar el pedido en la fecha prevista. El empleado ingresa la observación y automáticamente el pedido se incluye para el día siguiente. Para versiones posteriores se planea poder coordinar automáticamente con el usuario cuando realizar el reenvío del pedido.
- XXVII. **Visualización de Pedidos** en el panel administrador. Se deben poder ver los detalles de los pedidos, productos que lo integran y comprador del mismo.
- XXVIII. **Emisión de informes.** El sistema tiene que poder emitir informes del accionar de los usuarios y estadísticas de uso.
- XXIX. **Servicios contratados.** Dado un cliente mostrar el total de todos los servicios contratados.

**XXX. Ventas por fecha.** Dada una fecha obtener una lista de la cantidad de ventas realizadas.

**XXXI. Listado de ventas.** Se debe incluir la posibilidad de obtener una lista de las ventas realizadas, el monto y los productos vendidos.

**XXXII. Sistema de información geográfico.** Se tiene que poder contar con un sistema de información geográfico, el cual permita la visualización de la información espacial en su contexto, además da la posibilidad de combinar diferentes fuentes de datos y realizar análisis espacio temporal.

**XXXIII. Servicio de mails.** Se debe contar con la posibilidad que cualquier usuario pueda contactarse con la empresa por medio de un formulario desde la web, se implementara como servicio de envío de mail.

**XXXIV. Agenda de envíos.** Al dar de alta la compra se elige rango horario y posibles días para realizar la entrega.

**XXXV. Calificar.** Al finalizar un servicio el cliente debe poder tener la posibilidad de calificar el servicio y agregar una observación. Cuando el pedido está finalizado es que se le habilita al cliente la posibilidad de calificar.

**XXXVI. Búsquedas.** El sistema debe permitir realizar búsquedas de los productos y pedidos mediante distintos filtros.

**XXXVII. Asignación Pedidos a Reparto.** La asignación de un pedido a un reparto se debe hacer de forma automática según la dirección de entrega y la zona que le comprende a cada reparto.

Cada día quedan armados los repartos con sus pedidos de forma automática y quedan disponibles para ser adjudicados a un empleado a cargo.

**XXXVIII. Reparto y Zona de Reparto.** Actualmente el cliente no ha tenido la necesidad de modificar las zonas de los repartos, son fijas, por lo cual al tenerlas predefinidas no le interesa poder modificarlas, de todas formas para futuras versiones se tiene pensado agregar la funcionalidad que permita desde el panel administrador modificar las zonas adjudicadas a los repartos. En la versión actual del programa se identificará la zona para adjudicar un pedido al reparto correspondiente según las coordenadas de los mismos.

**XXXIX. Control de reparto.** Se debe validar que el reparto no esté saturado, esto se hace en base a la fecha seleccionada para la entrega del pedido. En caso de que el reparto ya tenga disponibilidad esa fecha se bloqueara impidiendo que se seleccione como fecha posible. En cada reparto se tiene que registrar el empleado a cargo del mismo.

**XL. Aviso de entrega realizada.** Al entregar un pedido el empleado a cargo del reparto registra lo que se realizó, esto se reflejara en forma instantánea en el sistema. Permitiendo realizar un seguimiento desde la empresa en tiempo

real. Se debe poder realizar mediante conexión web por un explorador, utilizando la app móvil o conectándose al servicio rest. En caso de no poder entregar el pedido también se registra y el mismo debe quedar pendiente para el día siguiente.

Si no se puede entregar un pedido se marca como pendiente y pudiéndose agregar una observación, quedando pendiente para el día siguiente.

- XLI. **Repartos pendientes para el día.** Se deben ver los repartos pendientes de entregar para el día en curso.

Cuando se adjudican los pedidos a un empleado a cargo del reparto se envía mail recordatorio al cliente que se le enviará su pedido.

- XLII. **Repartos en curso.** Se deben ver los repartos que están siendo entregados en el día.

- XLIII. **Repartos propios en curso.** Cada usuario tiene que poder ver los repartos que tiene en curso a su cargo.

- XLIV. **Estado Repartos.** Se tienen que visualizar todos los repartos y el estado, permitiendo acceder a los pedidos y a todos sus datos.

A medida que se van entregando los pedidos se marcan como entregados lo cual se visualiza en tiempo real en el sistema, tanto documentalmente como en el sistema de información geográfica.

- XLV. **Envío mails.** El usuario debe poder enviar desde la web mails de consulta.

- XLVI. **Sistema de fidelización.** Posibilidad de envío de promociones, descuentos y mensajes en fechas especiales para el cliente, como ser: cumpleaños, fin de año o si se desea tener una atención con el cliente en base a la confianza depositada. Posibilidad de seleccionar desde qué cuentas se trabaja para el envío de los mensajes.

- XLVII. **Exportar datos a pdf y a planilla de cálculo.** Se requiere poder exportar datos de informes a pdf y planillas electrónicas.

## **2.3 Diseño de la aplicación.**

En base a los requerimientos y a la lógica de negocio desarrollada se destacan 2 sistemas uno público y otro interno como solución al problema. Estos surgen de la identificación de las 2 necesidades principales:

- Agilizar comercialización de los productos.
- Gestionar eficientemente las órdenes de trabajo.

### **2.3.1 Clases.**

Se describen las clases mapeadas en la base de datos.

#### Usuario.

La clase Usuario es abstracta, crea un comportamiento general que heredan la clase Cliente y Empleado permitiendo utilizar el polimorfismo y la reutilización de código.

#### Cliente.

Tiene toda la información de los clientes, hereda métodos y atributos de Usuario y agrega las propiedades específicas de la clase Cliente.

#### Empleado.

Tiene toda la información de los empleados, hereda métodos y atributos de Usuario y agrega las propiedades específicas de la clase Empleado.

#### Proveedor.

Cuenta con los datos de los proveedores, no se agregó a la herencia de usuario ya que no va a ser usuario del sistema

#### Reparto.

Cuenta con los datos del empleado a cargo, el vehículo, la cantidad máxima de pedidos que admite y la zona que le corresponde.

#### RepartoEstado.

Es la relación entre el reparto y los pedidos, aquí se tiene el seguimiento del reparto cuando se encuentra en curso, con pedidos asignados.

#### HorarioEntrega

Enumerado que tiene los posibles horarios de entrega.

### Producto.

En esta clase se almacenan los datos de los productos.

### Pedido.

En esta clase se guardan los datos del pedido.

### Marca.

Se almacenan las distintas marcas de los productos, ingresadas en el sistema. También sirve para permitir realizar filtros por marca y flexibiliza el sistema dando la posibilidad de actualicen las marcas en el sistema de acuerdo a la realidad en el momento que se necesite.

### Categoría.

Se almacenan las distintas categorías de los productos, ingresadas en el sistema. También sirve para permitir realizar filtros por categoría y flexibiliza el sistema dando la posibilidad de actualicen las categorías en el sistema de acuerdo a las necesidades.

### Factura.

Aquí se relacionan los Pedidos con los datos de los productos que lo integran, lo cual generá la factura.

### LíneaFactura.

Esta clase tiene como propósito ser la relación entre la Factura y los Productos que integrarán la misma. Aquí se almacena que productos integran la factura y la cantidad de cada producto.

### Estado.

Esta clase tiene los posibles estados del pedido y el reparto.

Pendiente\_reparto, en caso que el pedido este ingresado pero no tenga reparto adjudicado para su distribución.

Adjudicado\_reparto, cuando el pedido ya fue procesado y se le asignó un reparto en base a la zona de la dirección de entrega elegida.

Entregado, cuando se entregó el pedido en destino, se da por finalizado el ciclo del pedido.

### Mensaje.

Esta clase se encarga de guardar y posibilitar el seguimiento de los mails realizados en el sistema.

### 2.3.2 Diagrama de clases.

Se modificó el diagrama y la estructura de las tablas en la base de datos respecto a la versión original presentada en el anteproyecto.

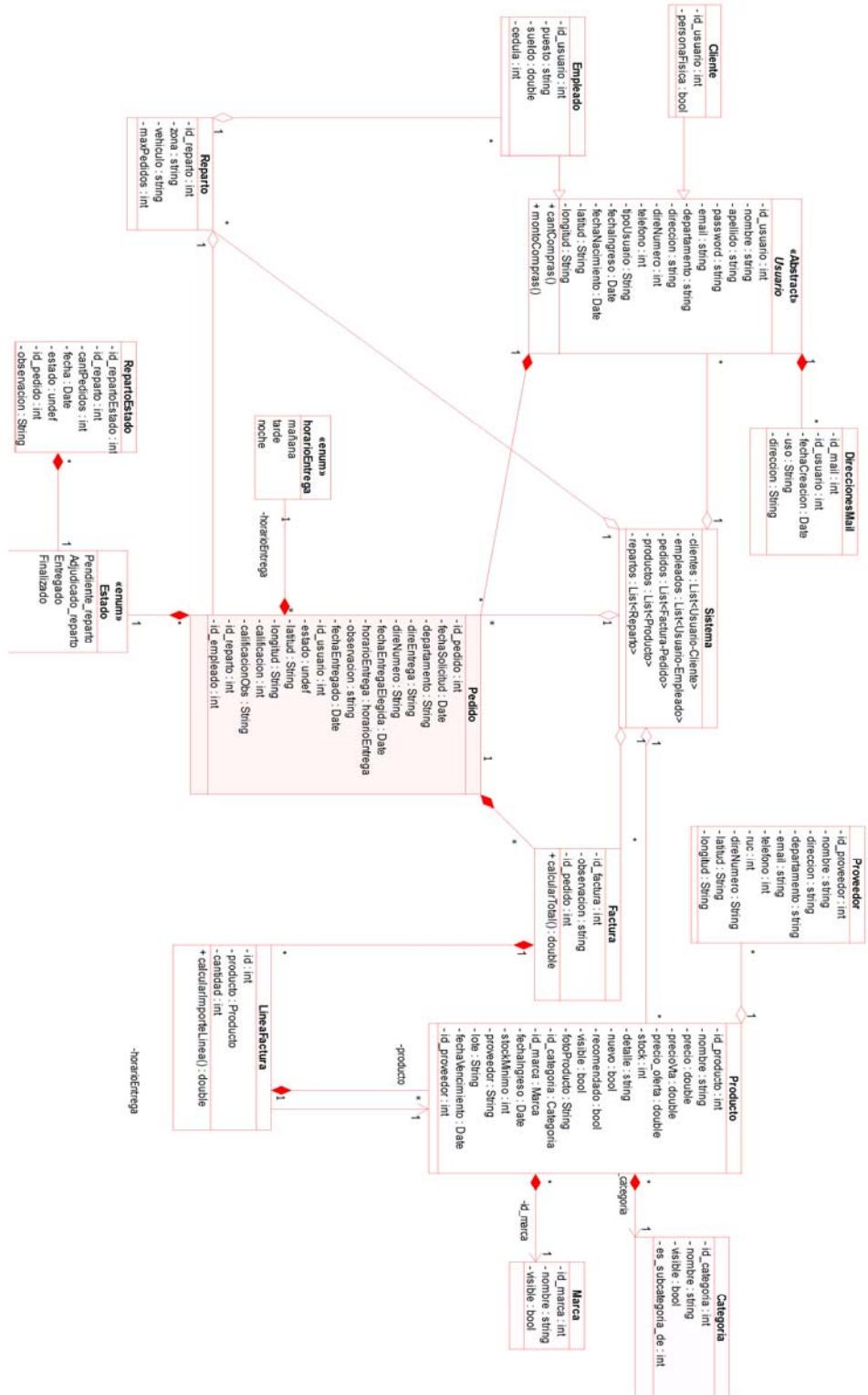


Ilustración 5. Diagrama de clases.

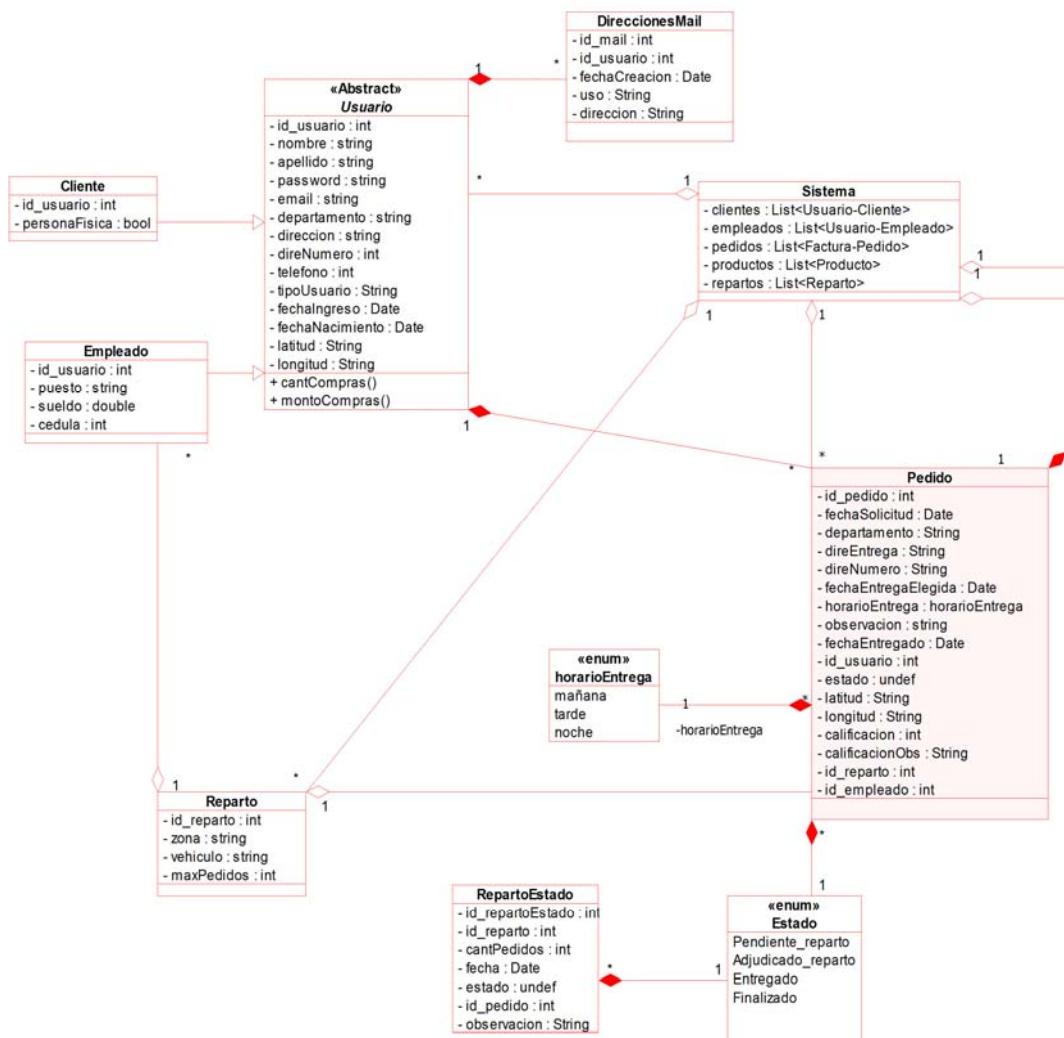


Ilustración 6. Diagrama de clases.

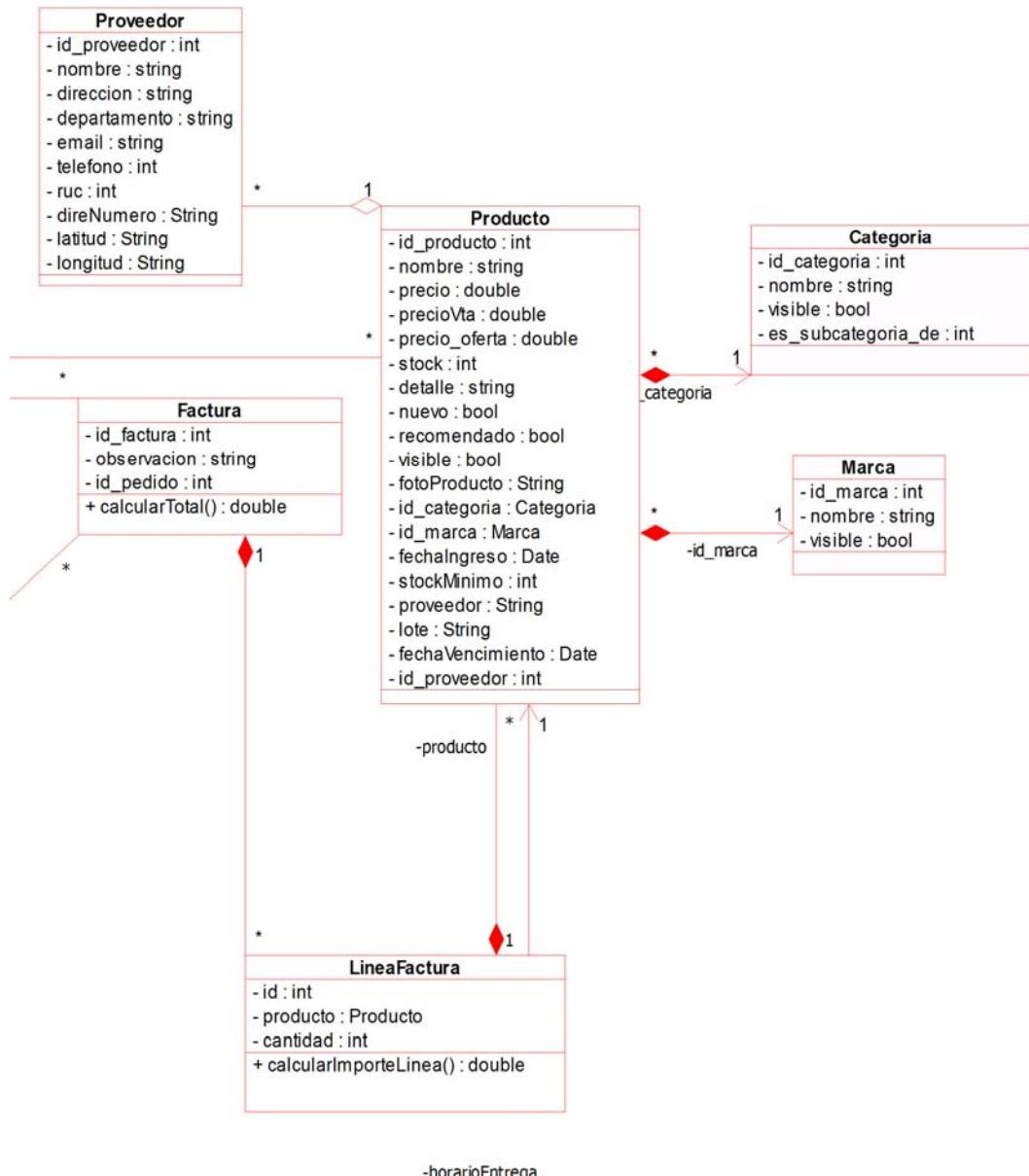


Ilustración 7. Diagrama de clases.

## 2.4 Ajuste de Sprints.

Se debieron ajustar los sprints ya que a medida que se fue avanzando en el proyecto se ha encontrado la necesidad de desarrollar funcionalidades no consideradas inicialmente.

### 2.4.1 Sprint 1. Análisis y Diseño.

Tarea	Fecha de inicio	Fecha de fin	Duración
Ideación del proyecto	10/03/18	19/03/18	10
Metodología de trabajo y estructura de Anteproyecto	20/03/18	5/04/18	17
Reunión con tutor	22/03/18	22/03/18	1
Reunión con tutor	29/03/18	29/03/18	1
1ra reunión con el Cliente	5/04/18	5/04/18	1
Reunión con tutor	5/04/18	5/04/18	1
Análisis del problema	7/04/18	7/04/18	1
Investigación	7/04/18	10/04/18	4
Elaboración de solución	8/04/18	8/04/18	1
Determinación de clases y diagrama UML	8/04/18	9/04/18	2
Modelo de proceso	9/04/18	9/04/18	1
Presentación del problema	9/04/18	9/04/18	1
2da reunión con el Cliente	9/04/18	9/04/18	1
Correcciones 1ra y 2da reunión	10/04/18	12/04/18	3
Lista de necesidades	12/04/18	15/04/18	4
Reunión con tutor	12/04/18	12/04/18	1
Análisis estratégico	14/04/18	16/04/18	3
Estudio de objetivos	14/04/18	16/04/18	3
Alcance y limitaciones	16/04/18	18/04/18	3
Lista de requerimientos	17/04/18	18/04/18	2
Ánálisis de factibilidad	19/04/18	22/04/18	4
Reunión con tutor	19/04/18	19/04/18	1
Ánálisis de riesgo	20/04/18	20/04/18	1
Plan del proyecto	21/04/18	24/04/18	4
ABM- Productos	24/04/18	15/05/18	22
Carrito de compras	24/04/18	16/05/18	23
Configuración entorno de desarrollo	25/04/18	25/04/18	1
Reunión con tutor	26/04/18	26/04/18	1
Configuración de ambiente de testing	2/05/18	4/05/18	3
Reunión con tutor	3/05/18	3/05/18	1
Construcción de pruebas	4/05/18	4/05/18	1
Testeo y refactorización	5/05/18	6/05/18	2
Reunión con tutor	8/05/18	8/05/18	1
Presentación Anteproyecto	10/05/18	10/05/18	1
Sprint 1- 3ra reunión con Cliente, entrega Anteproyecto	14/05/18	14/05/18	1

Tabla 5. Sprint 1- Análisis y Diseño.



Ilustración 8. Sprint 1- Análisis y Diseño.

### Lecciones aprendidas.

En el primer sprint se definió la metodología de trabajo y se elaboró el anteproyecto con el fin de tener un análisis sistemático de lo que va a implicar el desarrollo del proyecto. Todo esto con el fin de disminuir riesgos al tener el análisis y diseño de todas las funcionalidades a desarrollar para cubrir las necesidades del cliente.

En este sprint se vio la importancia de tener un resumen de los puntos más importantes para que el cliente pueda visualizarlo lo más fácil, rápido y con el menor esfuerzo posible el proyecto en su totalidad, además de contar con el documento del anteproyecto. También se pudo aprender en la realidad el valor de tener algún método gráfico que permita la rápida asimilación del proyecto.

Otro aprendizaje que se tuvo fue poder ir mejorando las presentaciones para realizarlas en forma breve y con economía de medios en el modo de expresar los conceptos con exactitud, ya que se dispone de poco tiempo para presentar las ideas y poder captar la atención del cliente.

Se tuvieron que postergar tareas como la configuración del ambiente de testing y la construcción de pruebas, lo cual repercutió en tener que aumentar las horas de trabajo en sprints siguientes.

Con el ambiente de testing se tuvo problemas para configurar el sitio de azure y generar la base de datos ya que por lo que pudimos ver en la clase de taller de proyecto requiere pago generar una base de datos MySQL.

Las reuniones con el tutor son una vez por semana pero se mantiene contacto permanente por mail, móvil y por video conferencia según se necesite.

En cuanto a la gestión de riesgos y su replanificación, para este primer sprint de los riesgos detallados en la gestión de riesgos los más factibles que podían darse era mala comprensión de los requerimientos, perdida de documentación, el tiempo requerido para el desarrollo del sistema es subestimado y falta de acceso al cliente. De estos riesgos sólo se presentó problemas en el tiempo requerido para el desarrollo del sistema debido fundamentalmente al inconveniente de no poder configurar el ambiente de testing para la construcción de pruebas en azure.

## 2.4.2 Sprint 2. Inicio de desarrollo de las funcionalidades.

Tarea	Fecha de inicio	Fecha de fin	Duración
ABM- Usuarios	14/05/18	17/05/18	4
Login	14/05/18	17/05/18	4
Correcciones 3ra reunión	15/05/18	16/05/18	2
Reunión con tutor	17/05/18	17/05/18	1
Reunión con tutor	24/05/18	24/05/18	1
Documentación (se realiza durante todo el proyecto)	24/04/18	25/09/18	155
ABM- Empleados	18/05/18	20/05/18	3
ABM- Proveedores	18/05/18	20/05/18	3
Realizar pedidos	18/05/18	22/05/18	5
Agendar envíos	23/05/18	26/05/18	4
Visualización de pedidos	27/05/18	28/05/18	2
Reunión con tutor	29/05/18	29/05/18	1
Visualización de factura	29/05/18	31/05/18	3
Stock mínimo	1/06/18	8/06/18	8
Sprint 2- 4ta reunión con Cliente, entrega avance 3ra reunión	4/06/18	4/06/18	1

Tabla 6. Sprint 2. Inicio de desarrollo de las funcionalidades.

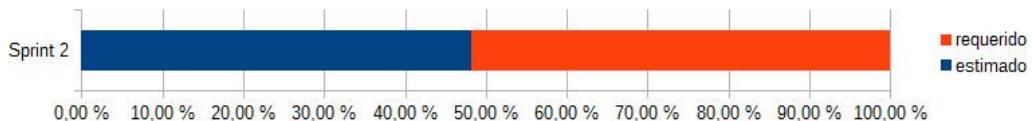


Ilustración 9. Sprint 2. Inicio de desarrollo de las funcionalidades.

### Lecciones aprendidas.

En el sprint no se pudo llegar a la planificación realizada en el anteproyecto en algunos puntos como:

Completar el desarrollo de ABM (alta, bajas y modificaciones) de proveedores ya que se tuvo que dedicar mayor tiempo a investigar para desarrollar eficientemente la funcionalidad de stock mínimo.

Tener un entregable para presentar ya que se avanzó en el desarrollo de la lógica pero faltó tiempo para la interfaz de usuario prefiriéndose, a modo de agilizar, seguir con el desarrollo de la lógica y luego realizar la interfaz de usuario.

Para el sprint 3 se pretende tener producto entregable para que el cliente pueda ir testeando la aplicación.

Implementar la lógica del Stock mínimo llevo más tiempo del estimado, ya que incluye más funcionalidades de las consideradas inicialmente.

Falto tiempo para correcciones, validaciones y testeo. Se tuvo investigando otras alternativas a azure pero todas requieren el ingreso de una tarjeta para poder implementar el sitio en Internet aunque sean gratuitas, por lo que hay que estar controlando las prestaciones que consume el sitio para que no se generen cargos, por lo que la implementación del sitio en la web se deja para seguir consultando en la próxima clase de taller de proyecto.

Varias funcionalidades implicaron mayor esfuerzo del estimado ya que habían más requisitos de los evaluados inicialmente, los cuales estaban empaquetados en las funcionalidades a desarrollar por lo que se requirió mayor tiempo de desarrollo.

Se sigue trabajando en la documentación.

Por todo esto se tuvieron que replanificar actividades y los riesgos. Ya que por falta de experiencia en el trabajo con azure no se pudo tener el sitio en Internet para que el grupo objetivo y el cliente pudiesen actuar de testers y probar el producto que se iba logrando. Las actividades demoradas se replanifican para los siguientes sprints.

#### **2.4.3 Sprint 3. Desarrollo de sistema de alertas, búsquedas y mantenimiento de órdenes de trabajo.**

Tarea	Fecha de inicio	Fecha de fin	Duración
Reunión con tutor	07/06/18	07/06/18	1
Aviso de entrega realizada	9/06/18	10/06/18	2
Mantenimiento de órdenes de trabajo	11/06/18	15/06/18	5
Reunión con tutor	14/06/18	14/06/18	1
Sistema de alertas	16/06/18	17/06/18	2
Búsquedas	18/06/18	22/06/18	5
5ta reunión con Cliente, entrega avance 4ta reunión	20/06/18	20/06/18	1
Reunión con tutor	21/06/18	21/06/18	1
Visualización información en mapa	23/06/18	15/07/18	23
Reunión con tutor	28/06/18	28/06/18	1

Construcción de pruebas	09/07/18	09/07/18	1
Testeo y refactorización	10/07/18	10/07/18	1
Reunión con tutor	12/07/18	12/07/18	1
Sprint 3- 6ta reunión con Cliente, entrega avance 5ta reunión	16/07/18	16/07/18	1

Tabla 7. Sprint 3. Desarrollo de sistema de alertas, búsquedas y mantenimiento de órdenes de trabajo.

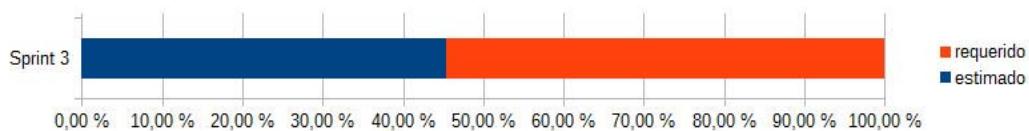


Ilustración 10. Sprint 3. Desarrollo de sistema de alertas, búsquedas y mantenimiento de órdenes de trabajo.

### Lecciones aprendidas.

Algunos puntos se subestimaron en cuanto a su complejidad como el sistema de stock, por lo que se necesitó más tiempo del planificado inicialmente. Se tuvo que considerar los distintos ingresos de mercadería al almacén como lotes para que cuando se repongan se vea si hay un lote anterior con stock del mismo producto, en tal caso se sigue mostrando el anterior para que no caduque. El lote nuevo se muestra cuando queda sin unidades disponibles uno anterior, si no hay ningún lote con stock se muestra el producto pero no se posibilita incluirlo en el carrito de compras y se despliega un anuncio: “Momentáneamente sin stock”.

Falto tiempo para correcciones y validaciones y testeo.

Para no recargar demasiado sprints siguientes hubo que dedicar más tiempo del estimado, resultó fundamental tener previsto un margen de días para contingencias.

Aviso de entrega realizada y las búsquedas se pasan para el siguiente sprint.

Lograr utilizar la geocodificación llevo más tiempo del estimado.

Se dedico tiempo a tareas que no estaban previstas para este sprint como la configuración del ambiente de testing y mejorar la interfaz de usuario, se logro configurar el servidor pero aún resta resolver problema de permisos para generar la base de datos. A medida que se avanza en el desarrollo del sistema se va documentando.

Se estuvo investigando para la implementación del aviso de entrega realizada desde una app móvil y por web services.

En cuanto a web services tal vez sea descartado porque algunos hosting no soportan estos servicios y se deja la posibilidad de hacerlo por la app móvil o por medio de la web del sitio.

En cuanto a los riesgos, ocurrió que el tiempo requerido para el desarrollo fue subestimado, un poco por falta de experiencia y otro porque a medida que se iba avanzando en el proyecto se mejoraba la comprensión, lo que permitió mejorar las funcionalidades pero requirió más tiempo. Debido a esto se necesito recurrir a los días reservados para imprevistos para poder hacer frente a la situación, con el fin de no seguir recargando en demasía los siguientes sprints.

#### **2.4.4 Sprint 4. Sistema de fidelización, refactorización y mejora de funcionalidades.**

Tarea	Fecha de inicio	Fecha de fin	Duración
Sistema de fidelización	17/07/18	25/07/18	9
Reunión con tutor	19/07/18	19/07/18	1
Configuración de búsquedas	20/07/18	23/07/18	4
Reunión con tutor	26/07/18	26/07/18	1
Calificar	26/07/18	29/07/18	4
Aviso de entrega realizada	16/07/18	29/07/18	14
Refactorización y mejora de funcionalidades	25/07/18	30/07/18	6
Sistema de información geográfica	22/07/18	30/07/18	9
Sprint 4- 7ma reunión con Cliente, entrega avance 6ta reunión	30/07/18	30/07/18	1

Tabla 8. Sprint 4. Sistema de fidelización, refactorización y mejora de funcionalidades.



Ilustración 11. Sprint 4. Sistema de fidelización, refactorización y mejora de funcionalidades.

#### Lecciones aprendidas.

Se sigue trabajando en el aviso de entrega realizada, resta resolver la conexión de la app móvil para enviar y recibir datos, se está investigando para realizar api rest.

A esta altura se tiene un diseño de vistas aceptable pero se sigue trabajando en mejorar el diseño, se investiga para ver la opción de implementar template porque para llegar al nivel deseado habría que dedicar más tiempo, lo cual achicaría mucho el margen de error en caso de que surja algún imprevisto, en este momento se considera mejor dejar

de lado el diseño para dedicarle más tiempo al desarrollo y mejorar las funcionalidades y la lógica del sistema.

A partir de la clase de ambiente de testing la solución se encontró es utilizar un crédito de 100USD que se da para azure y utilizarlo en generar la base de datos MySQL para poder correr la implementación pero como se cobra a medida que se usa el sitio con el profesor se cree conveniente no dedicarle demasiado tiempo más hasta terminar el proyecto, el tutor también cree conveniente seguir trabajando en el desarrollo de la aplicación y subir a azure cuando el proyecto este terminado.

Se empezó a trabajar en el sistema de información geográfica, se seguirá en el siguiente sprint.

En cuanto a los riesgos se vio un cambio de prioridad del cliente haciendo mayor énfasis en el desarrollo de la interfaz gráfica y en el desarrollo del sistema de información geográfica, por lo cual en los futuros sprints se dedicaran más días de los previstos para afrontar riesgos al desarrollo de estas actividades.

#### **2.4.5 Sprint 5. Emisión de informes, mejora de interfaz de usuario y sistema de información geográfica.**

Tarea	Fecha de inicio	Fecha de fin	Duración
Mostrar servicios contratados	31/07/18	5/08/18	6
Reunión con tutor	2/08/18	2/08/18	1
Emisión de informes	4/08/18	19/08/18	16
Documentación	29/08/18	06/08/18	9
Construcción de pruebas	5/08/18	5/08/18	1
Testeo y refactorización	5/08/18	6/08/18	2
Sistema de información geográfica	31/07/18	06/08/18	7
Investigación	20/07/18	06/08/18	18
Sistema de fidelización	31/07/18	06/08/18	7
Sprint 5- 8va reunión con Cliente, entrega avance 7ma reunión	6/08/18	6/08/18	1

Tabla 9. Sprint 5. Emisión de informes, mejora de interfaz de usuario y sistema de información geográfica.



Ilustración 12. Sprint 5. Emisión de informes, mejora de interfaz de usuario y sistema de información geográfica.

## Lecciones aprendidas.

Se continúa trabajando en la implementación y desarrollo del sistema de información geográfica, se investigan varias librerías javascript como ser openlayers y leaflet. Openlayers y leaflet no se pueden usar en forma conjunta.

El trabajo en lograr la geocodificación llevo más tiempo del planificado inicialmente ya que se tuvo que investigar como lograr la interacción entre java y javascript.

También resulto dificultosa la integración de varias funcionalidades para que trabajen en forma conjunta en el mapa. Como forma de acelerar el desarrollo se investiga la implementación de node.js.

Node.js es un entorno de ejecución para JavaScript, es un entorno en tiempo de ejecución para la capa del servidor, trabaja en forma asíncrona.

Se incluyeron en el mapa las funcionalidades de búsqueda por barrio, por dirección y por usuario, ya sea cliente o empleado. También se cuenta con dos tipos de capas base, vista de satélite y mapa de calle. Además se incluyó la posibilidad de poder diferenciar del total de pedidos los están en los repartos del día, también se puede hacer seguimiento al estado de los pedidos que están siendo entregados.

Otras funcionalidades con las que se cuenta son los controles de zoom, vista de pantalla completa, mantener una referencia en un mapa secundario a una escala mayor a la que se está trabajando en el mapa principal para mantener referencia de la ubicación. Se cuenta con la posibilidad de controles en la forma de visualizar la capa de barrios y la opción de agregar las referencias a los barrios de Montevideo para poder verlos diferenciados según su extensión, pudiéndose apreciar su área.

Se continúa trabajando en la documentación, en la aplicación móvil, en la implementación del sistema de fidelización y en la implementación del sistema de envío de mails.

También se está investigando como implementar el sistema de información geográfica sin incluir servidores web (mapserver, geoserver) y sin tener necesidad de contar con base de datos espacial, con el fin de reducir los costos de alojamiento web para la aplicación.

Además de investigar como realizar la integración entre java y javascript, también se tuvo que ver la potencialidad de usar ajax, json y geojson para no tener que usar ni servidor de datos espaciales, ni base de datos espacial y para realizar la comunicación con la aplicación móvil.

Para el desarrollo de la interfaz con el usuario administrador se eligió implementar el template AdminLTE. Por lo que se investigó como integrar con el actual desarrollo de la parte front-end.

Para el desarrollo de la aplicación móvil se aplica las librerías jquery y jquery Mobile para agilitar el desarrollo de la interfaz de usuario.

Al intentar subir la web al sitio de azure presenta problemas con la integración de los servlets.

Para afrontar los problemas acontecidos fue crucial tener la planificación estudiada en el análisis de riesgos que permitió con los días de respaldo afrontar la situación exitosamente. Con estos días se pudo dedicar más esfuerzo para llegar a cubrir los requerimientos y minimizar la sobrecarga a los siguientes sprints.

#### 2.4.6 Sprint 6. Mejora del SIG, validaciones y app móvil.

Tarea	Fecha de inicio	Fecha de fin	Duración
Investigación	6/08/18	10/08/18	5
Refactorización y testeo	9/08/18	14/08/18	6
Reunión con tutor	9/08/18	9/08/18	1
Mejora del SIG	8/08/18	20/08/18	13
App móvil	10/08/18	18/08/18	9
Corrección y control de errores	12/08/18	19/08/18	8
Validaciones	12/08/18	19/08/18	8
Reunión con tutor	16/08/18	16/08/18	1
Sprint 6- 9na reunión con Cliente, entrega avance 8va reunión	20/08/18	20/08/18	1

Tabla 10. Sprint 6. Mejora del SIG, validaciones y app móvil.



Ilustración 13. Sprint 6. Mejora del SIG, validaciones y app móvil.

#### Lecciones aprendidas.

Se integró el template AdminLTE a la parte front-end del sistema.

Se van concluyendo funcionalidades, se finalizó la implementación de la aplicación móvil con el servicio rest.

Para el sistema de fidelización se logró implementar el envío de mail, a modo de recordatorio, para avisar que se va a realizar la entrega de los pedidos y se inició el

desarrollo para enviar mail saludando fechas festivas. Resta implementar el seguimiento de los mensajes.

Fue necesario dedicar tiempo extra para investigación y refactorización.

Se mejoró el seguimiento en la transición de la asignación de pedidos a los repartos. Se tienen los pedidos pendientes para el día, a los cuales se adjudica empleado a cargo y quedan en reparto en curso, los empleados pueden ver todos los repartos en curso, los que tienen asignados para entregar y también se cuentan con un histórico del estado de los repartos.

En cuanto a la replanificación de riesgos en caso de que se quiera realizar un cambio en los requerimientos a esta altura tendría que dejarse para luego de la entrega académica.

Se tuvo que dedicar más tiempo del estimado en el control de calidad ya que no se contó con todos los días extras estimados porque se tuvieron que ir usando para hacer frente a los problemas surgidos en sprints anteriores, fundamentalmente por haberse reducido significativamente el control de calidad que se esperaba realizara el grupo objetivo y el cliente, lo cual no fue posible por no haberse podido subir el sistema a la web. De todas formas se pudo hacer frente a este problema para llegar a un nivel aceptable, dejando para el siguiente sprint realizar los testos finales para el control de calidad.

#### 2.4.7 Sprint 7. Cierre de funcionalidades y de documentación, optimización del diseño, refactorización y testeo.

Tarea	Fecha de inicio	Fecha de fin	Duración
Optimización diseño, interfaz de usuario	21/08/18	25/09/18	36
Construcción e implementación de pruebas	21/08/18	22/08/18	2
Testeo y refactorización	22/08/18	25/08/18	4
Plan de liberación, deploy	23/08/18	23/08/18	1
Presentación	20/09/18	26/09/18	7
Sprint 7- 10ma reunión con Cliente, entrega del proyecto	26/09/18	26/09/18	1

Tabla 11. Sprint 7. Cierre de funcionalidades y de documentación, optimización del diseño, refactorización y testeo.



Ilustración 14. Sprint 7. Cierre de funcionalidades y de documentación, optimización del diseño, refactorización y testeo.

### Lecciones aprendidas.

Se termino de mejorar la interfaz de usuario tanto para la parte administrador como para la del cliente.

Se realizó el testeo de las funcionalidades implementadas.

Con azure no se pudo publicar el sitio debido a los problemas ya detallados en los sprints anteriores, se investigaron otras soluciones como Heroku pero como para usar de forma comercial requiere pago y por falta de tiempo no se llegó a implementar.

Se completo la documentación y se optimizó en cuanto a su extensión para que fuese lo menos extensa posible a fin de mejorar en claridad y utilidad.

Se resolvieron problemas en la implementación de la api rest, permitiendo que se pueda realizar la aplicación móvil nativa en android.

Se mejoró la funcionalidad del carrito de compras, el mantenimiento de pedidos y se armó script con paquete de datos de prueba. También se mejoró la interacción con las gráficas en la presentación de informes y se incluyó la posibilidad de exportar los datos a pdf o a hoja de cálculo.

A pesar de todo lo realizado quedaron cosas a mejorar en futuras versiones como las funcionalidades en las gráficas y en el sistema de información. También es factible de mejora la presentación de informes, delimitación y adjudicación de zonas de repartos, optimizar rutas de reparto e incluir la posibilidad de análisis de rutas. También se puede mejorar en la refactorización de código y la presentación de los pdfs.

Este primer prototipo permitió visualizar nuevas funcionalidades a desarrollar.

Para futuras versiones sería de interés permitir que al agregar un nuevo reparto se puedan delimitar nuevas zonas para los repartos en forma automática, aunque al momento no es algo necesario para el cliente.

También restaría implementar hacer un seguimiento de los mensajes que se reciben y envían para tener información que permita generar un sistema de respuestas automáticas por el sistema. Otra funcionalidad a seguir desarrollando sería en la optimización del geocodificador.

## **2.5 Estándares definidos y convenciones.**

Todas las funcionalidades implementadas tienen un nombre que actúa como descriptor de su funcionamiento. El código se deberá identar correctamente.

A continuación se detalla lo realizado en la implementación de estándares y convenciones.

### **2.5.1 Documentación.**

Para la documentación se utilizaran los estándares definidos por la Universidad en los documentos:

- Documento 302 (Universidad ORT Uruguay): Normas específicas para la presentación de trabajos finales de carrera (TFDC) de la Facultad de Ingeniería.

Motivo: Es requisito cumplir con los estándares que describe para presentar trabajos académicos.

- Documento 303 (Universidad ORT Uruguay): Lista de Verificación del Formato.

Motivo: Sirve para corroborar que se haya cumplido con los estándares definidos en el documento 302.

- Documento 306 (Universidad ORT Uruguay): Guías para títulos, abstracts e informes de corrección.

Motivo: Es requisito cumplir con los estándares que describe para presentar trabajos académicos.

- Guía de proyectos Analista Programador.  
<https://aulas.ort.edu.uy/mod/folder/view.php?id=81654>

### **2.5.2 Nomenclatura.**

Los nombres a asignar a variables, constantes, funciones, entidades, etc., deben ser representativos y dar información de la tarea que realizan. Para los cuales no se utilizarán tildes ni la letra “ñ”, cuando una palabra esta compuesta por “ñ” en su lugar se usará “ni”.

Cuando el nombre esté formado por más de una palabra, cada palabra debe empezar con mayúscula, de ser necesario para mayor claridad se podrá separar por infra guion (“\_”).

Para:

- campos de texto se utilizará el prefijo “txt” más el nombre del dato que contendrá (Ej.: txtDireccion).
- botones se utilizará el prefijo “btn” más el nombre de la acción que realiza (Ej.: btnAgregar).
- formularios se utilizará el prefijo “form” más el nombre que relaciona al formulario con su función (Ej.: formClientes).
- etiquetas se utilizará el prefijo “label” seguido del nombre del dato que contendrá (Ej.: labelEdad).

En la base de datos:

Los nombres deben tener relación con los datos que guardan.

Los nombres de las tablas serán en singular. Un concepto que esté en varias tablas debe ser llamado en todas con el mismo nombre.

Si la tabla es resultado de la relación de dos o más tablas se utilizará el carácter \_ (guión bajo) para unir los nombres de éstas.

## **2.6 Análisis e investigación de tecnologías.**

El análisis e investigación ha consumido gran cantidad de tiempo del proyecto ya que se tuvo que ahondar en java, HTML y JavaScript, además de las librerías utilizadas para la parte de la implementación de las vistas, debido a que se contaba con experiencia en cuanto a la implementación lógica pero no tanto en el desarrollo front-end.

Para el desarrollo front-end también fue necesario ver como integrar tecnologías como bootstrap, css, inclusión de template AdminLTE, jquery, jquerymobile, GeoJSON y ajax.

Se tuvo que indagar en distintas fuentes, blogs, tutoriales para ver como se realiza la comunicación entre java y javascript, ya que la implementación del sistema de información geográfica (SIG) es en javascript y necesita poder comunicarse con la lógica de negocio en java.

Para la implementación del SIG en cuanto a la parte teórica se tuvo que ver cómo realizar una aplicación web mapping con software open source para la visualización de la cartografía web, también se investigó sobre cartografía temática y en cuanto a software se vieron las librerías de openlayers y leaflet. También se investigó software SIG de escritorio, se vio qGIS y gvSIG pero por ser más rápida la curva de aprendizaje y ya contar experiencia se eligió trabajar con Qgis.

También se investigó la api de google maps y la plataforma de servicios en la nube, cuentan con dos cursos gratuitos en udacity, uno para desarrollo web front-end y otro para desarrollo móvil android, son enormes las potencialidades y la cantidad funcionalidades que ofrece pero para desarrollo comercial las mayorías de las opciones requieren pago, por lo cual en este momento no sería una opción.

Se estudio como se integran distintos tipos de capas base de OpenStreetMap, Bing, Google y Mapbox (proveedor de mapas on line), con el fin de realizar la mejor solución.

Para profundizar en el desarrollo web mapping, en base a lo que se pudo aprender, es fundamental ahondar en el aprendizaje de desarrollo de scripts con python.

Al pensar en crear la aplicación web que integrará la parte de información espacial, un tema a solucionar era como crear la arquitectura y las herramientas necesarias para ello, siempre se tuvo claro que se quería realizar mediante herramientas de software libre pero hubo que investigar sobre servidores y base de datos espaciales para ver las necesidades de desarrollo y hosting.

En cuanto a servidores que soporten componente espacial para la parte de información geográfica se vio geoserver y mapserver, la elección se fundamento a que son los 2 principales servidores de referencia para desarrollo open source debido a su fiabilidad y cuentan con amplia documentación.

Geosever está escrito en Java, por lo que es multiplataforma lo que lo hace más portable que MapServer. Soporta archivos WFS-transaccional lo cual permite la edición de información geográfica a través de servicios web.

También se destaca que cuenta con una interfaz gráfica web amigable a diferencia de MapServer en el que se debe editar un archivo mapfile para trabajar con él, un sólo error en el archivo .map y no cargará nada, por tanto al empezar no parece ser tan amigable pero logra mejor rendimiento en cuanto al uso de procesador y memoria del hardware. Esto se debe a las diferentes arquitecturas, MapServer está escrito en C lo cual administra mejor los recursos que un programa que corre sobre una máquina virtual.

Las principales ventajas de Mapserver son la performance y el filtrado de WMS. Las principales ventajas de Geoserver son la facilidad de uso, portabilidad y soporte de WFS-T.

En cuanto a base de datos se comparó entre MySQL, PostGRES + postgis, luego de haber ahondado y probado las distintas tecnologías se vio la posibilidad de trabajar con GeoJSON y la potencialidad que ofrece, el desarrollo se realizó con el fin de requerir menores prestaciones de servidor ya que se puede evitar el uso de un servidor geográfico mediante el empleo GeoJSON. Gestión de la cache de imágenes de mapa geowebcache

También se vio como trabaja la plataforma para publicar mapas en la nube Carto Builder, permite con conocimientos básicos desarrollar un interesante mapa temático con acceso en la web pero son bajas las posibilidades de personalización, en comparación a lo que se puede realizar con las tecnologías anteriormente mencionadas.

Para trabajar con la base de datos se exploró entre 2 soluciones pgAdmin4 y MySQL Workbench, se eligió la 2da debido a que el consumo de recursos de pgAdmin4 se vio que es mucho mayor.

Como ide base para realizar el proyecto se comparo Eclipse, IntelliJ y Netbeans. Se optó por Netbeans ya que para acceder a todas la funcionalidades necesarias con IntelliJ requería adquirir la opción paga por lo cual fue la 1ra en ser descartada. Entre Eclipse y Netbeans fue más difícil la decisión ya que se tiene experiencia en trabajar con ambas ide y presentan varias facilidades uno mejor que otra dependiendo el desarrollo a realizar pero se terminó decidiendo por Netbeans debido a que tiene mejor resuelta, o resultado más sencilla para el desarrollo de las vistas.

Se instaló advanced REST Client de google chrome y el software postman para el testeo de la conexión entre el servicio y la aplicación móvil. En netbeans se incluyó el plugin visual Mobile disegner.

### **2.6.1 Librerías Utilizadas.**

Además de las librerías ya especificadas en el anteproyecto se incluyeron las siguientes:

Para interfaz gráfica:

- Jquery
- Jquery Mobile
- jqPlot

Para exportar datos a pdf:

- Itexpdf

Para exportar datos a hoja de cálculo:

- Poi-3.11

Para sistema de información geográfica:

- Leaflet-MiniMap
- Leaflet-fullscreen
- Leaflet-esasyPrint
- Leaflet-panel-layers

- Leaflet-control.layerTree
- Leaflet-HtmlLegend
- Leaflet.markercluster
- Leaflet-search

Para gráficas:

- Char.js

## **2.7 Pruebas.**

Con las pruebas se pretende establecer procesos que permitan establecer la calidad del producto y comprobar que cosas se deben ajustar.

### **2.7.1 Introducción.**

Se sigue el plan de pruebas planteado en el anteproyecto.

### **2.7.2 Plan de testing.**

Se continúa con el plan de testing establecido en el anteproyecto, con la intención de encontrar errores desde fases tempranas del desarrollo del ciclo de vida del proyecto, la tarea se dificultó debido a no poder subir desde el inicio el sitio a la web por las limitaciones actuales en el servicio de Microsoft Azure, ya que no se pudo contar con la magnitud del testeo pensado para que pudiese desarrollar el cliente y el grupo objetivo.

Se realizaron tests específicos para probar las funcionalidades y otros para el sistema integrado.

- Test de funcionalidad.
- Test de sistema. En el plan de pruebas
- Para realizar test de aceptación del cliente y usuarios con el fin de descartar problemas en la interpretación, se fueron realizando entregas, como se detalla en los sprints, para que se pudiese realizar el test de aceptación tanto para el cliente como para el grupo objetivo.

La evidencia de ejecución de los distintos tests se adjunta en el plan de pruebas.

### **2.7.2.1 Test de caja negra.**

Estaba pensado para que el cliente y el grupo objetivo lo fuesen realizando durante todo el proceso de desarrollo del sistema pero se tuvo que realizar a medida que si iban entregando los sprints y fue necesario implementar de forma local la posibilidad para poder correr el sistema ya que no se pudo subir al sitio de azure. Esto dificultó ya que se tuvo que configurar las versiones finalizadas para poder trabajar en forma local en lugar de tenerlo en la web, lo cual permitía ir testeando de forma mucho más dinámica.

### **2.7.2.2 Test de caja blanca.**

Fue realizada por el desarrollador y al ir terminando los distintos flujos de ejecución se iban realizando las pruebas. Se elegían distintos valores de entrada para examinar los flujos de ejecución y ver que los valores que se obtienen sean correctos.

## **2.8 Plan de proyecto.**

El plan de proyecto comprende la información relativa a la gestión del proyecto.

### **2.8.1 Introducción.**

Fue necesario implementar algunas modificaciones, sobre todo en la asignación de tiempo de las tareas, en cuanto al plan presentado en el anteproyecto.

Se intentó respetar el plan de entregas pero el mismo sufrió modificaciones debido a los problemas ya especificados para publicar con el servicio de Microsoft Azure.

A continuación se analiza como fue el transcurso del desarrollo y como se fueron dando los riesgos a medida que avanzaba el tiempo.

### **2.8.2 Gestión de riesgos.**

Se siguió la gestión de riesgos planteada en el anteproyecto, fue fundamental dejar varios días para contención de riesgos ya que el testeo y comprobación de las funcionalidades llevo más tiempo del esperado, esto debido a que no se pudo publicar la aplicación lo cual hubiese significado una importante fuente de información y ahorro de tiempo.

Inicialmente se pretendió tener publicado el sitio desde el inicio del proyecto para que el cliente y el grupo objetivo actuaran de testers y tener retroalimentación constante.

Para poder sortear el problema se necesito más tiempo para hacer el testeo de la aplicación y debido a esto las reuniones con el cliente también fueron más extensas de lo planificado inicialmente.

Se detalla la ocurrencia de los riesgos considerados en el anteproyecto.

Principales Riesgos (Riesgo, Probabilidad e Impacto)			
Id	Riesgo	Probabilidad	Impacto
1	Cambios en los requerimientos.	Moderada	Serio
2	Mala comprensión de los requerimientos.	Moderada	Serio
3	Perdida de documentación.	Muy baja	Tolerable
4	Falta de acceso al cliente.	Baja	Serio
5	Falta de experiencia en proyectos reales.	Moderada	Tolerable
6	El tiempo requerido para el desarrollo del sistema es subestimado.	Alta	Catastrófico
7	Dificultad en el control de calidad.	Bajo	Tolerable
8	Herramientas seleccionadas para el proyecto incorrectas.	Muy Baja	Tolerable
9	Cambio de prioridades del cliente.	Baja	Serio

Tabla 12. Principales Riesgos.

La ocurrencia de los riesgos se detalló en el desarrollo de los sprints, en la sección “Ajuste de Sprints”.

## **2.9 Proceso de desarrollo.**

### **2.9.1 Introducción.**

Al haber trabajado con la metodología Scrum permite que el producto obtenido no sea una sorpresa para el cliente y el producto sea el resultante de un proceso conjunto. Ya que el cliente forma parte de todo el proceso de desarrollo, siendo este un factor primordial para el control de calidad y la disminución de riesgos.

En el desarrollo del proceso se cumplió con todas las instancias estipuladas por la Universidad. También se pudieron realizar todas las entregas planificadas con el cliente.

El detalle del proceso de desarrollo se puede ver en la pila del producto, en lo expuesto en cada sprint y en el análisis de esfuerzo que está a continuación.

### **2.9.2 Adaptación del proceso.**

La metodología Scrum es una forma de trabajo pensada para un equipo de al menos de 4 o 5 personas. Como para este desarrollo el equipo de trabajo consta de un desarrollador (estudiante) y el tutor, se tuvo que hacer una adaptación para poder aplicarlo al proyecto.

### **2.9.3 Seguridad.**

Como medidas de seguridad se pretendió mantener la aislación de capas entre sí, que las vistas no accedan al modelo y control de usuario específico para el sitio para clientes de la empresa y otro para el sitio administrador.

## **2.10 Proceso de entrega.**

### **2.10.1 Manual de deploy.**

A continuación se detalla en el manual de “deploy” los pasos de como instalar la aplicación ya que no se pudo publicar en el sitio de Azure.

Para poder desarrollar la aplicación se trabajo con NetBeans como IDE de desarrollo, XAMPP como panel de control y MySQL como motor de base de datos, las especificaciones de las versiones usadas son:

- NetBeans. <https://netbeans.org/downloads/>  
Version: NetBeans IDE 8.2.
- Java: 1.8.0. <https://www.java.com/es/download/>
- XAMPP. <https://www.apachefriends.org/es/download.html>  
Control panel v3.2.2
- MySQL. <https://www.mysql.com/products/workbench/>  
MySQL Workbench version 6.3.

Se recomienda trabajar con estas versiones o posteriores. Luego de tener instalados el ide, el panel de control y motor de base de datos, desde Workbench se debe correr el script de creación de la base de datos y desde NetBeans se abre el proyecto, desde donde ya queda todo pronto para correr desde el index.jsp el sistema.

Para desplegar la aplicación móvil se incluye el instalador, el cual se debe copiar al dispositivo que se quiera instalar y ejecutarlo pero al no estar subido el sitio a la web no va a funcionar la conexión de la api rest por lo cual para simular el comportamiento en forma local, desde el proyecto abierto en NetBeans se debe correr el archivo movil.jsp.

Usuario predeterminado para ingresar al sistema como administrador:

- Usuario: adm@admproyecto.com
- Password: admproyecto

## 2.11 Análisis de Esfuerzo.

Se realiza el análisis de esfuerzo comparando las tareas estimadas en el anteproyecto y las que se tuvieron que realizar. En la siguiente tabla las tareas correspondientes al anteproyecto están con fondo blanco y las tareas correspondientes a las que se desarrollaron en el proyecto con fondo verde.

<b>Id</b>	<b>Tarea</b>	<b>Fecha de inicio</b>	<b>Fecha de fin</b>	<b>Duración</b>
1	Ideación del proyecto	10/03/18	19/03/18	10
2	Metodología de trabajo y estructura de Anteproyecto	20/03/18	5/04/18	17
3	Reunión con tutor	22/03/18	22/03/18	1
4	Reunión con tutor	29/03/18	29/03/18	1
5	1ra reunión con el Cliente	5/04/18	5/04/18	1
6	Reunión con tutor	5/04/18	5/04/18	1
7	Análisis del problema	7/04/18	7/04/18	1
8	Investigación	7/04/18	10/04/18	4
9	Elaboración de solución	8/04/18	8/04/18	1
10	Determinación de clases y diagrama UML	8/04/18	9/04/18	2
11	Modelo de proceso	9/04/18	9/04/18	1
12	Presentación del problema	9/04/18	9/04/18	1
13	2da reunión con el Cliente	9/04/18	9/04/18	1
14	Correcciones 1ra y 2da reunión	10/04/18	12/04/18	3
15	Lista de necesidades	12/04/18	15/04/18	4
16	Reunión con tutor	12/04/18	12/04/18	1
17	Análisis estratégico	14/04/18	16/04/18	3
18	Estudio de objetivos	14/04/18	16/04/18	3
19	Alcance y limitaciones	16/04/18	18/04/18	3
20	Lista de requerimientos	17/04/18	18/04/18	2
21	Análisis de factibilidad	19/04/18	22/04/18	4
22	Reunión con tutor	19/04/18	19/04/18	1
23	Análisis de riesgo	20/04/18	20/04/18	1
24	Plan del proyecto	21/04/18	24/04/18	4
25	ABM- Productos	24/04/18	15/05/18	22
26	Carrito de compras	24/04/18	16/05/18	23
27	Configuración entorno de desarrollo	25/04/18	25/04/18	1
28	Reunión con tutor	26/04/18	26/04/18	1
29	Configuración de ambiente de testing	2/05/18	4/05/18	3
30	Reunión con tutor	3/05/18	3/05/18	1
31	Construcción de pruebas	4/05/18	4/05/18	1
32	Testeo y refactorización	5/05/18	6/05/18	2
33	Reunión con tutor	8/05/18	8/05/18	1
34	Presentación Anteproyecto	10/05/18	10/05/18	1
35	Sprint 1- 3ra reunión con Cliente, entrega Anteproyecto	14/05/18	14/05/18	1
36	ABM- Usuarios	14/05/18	17/05/18	4
37	Login	14/05/18	17/05/18	4

38	Correcciones 3ra reunión	15/05/18	16/05/18	2
39	Reunión con tutor	17/05/18	17/05/18	1
40	Reunión con tutor	24/05/18	24/05/18	1
41	Documentación (se realiza durante todo el proyecto)	24/04/18	25/09/18	155
42	ABM- Empleados	18/05/18	20/05/18	3
43	ABM- Proveedores	18/05/18	20/05/18	3
44	Realizar pedidos	18/05/18	22/05/18	5
45	Agendar envíos	23/05/18	26/05/18	4
46	Visualización de pedidos	27/05/18	28/05/18	2
47	Reunión con tutor	29/05/18	29/05/18	1
48	Visualización de factura	29/05/18	31/05/18	3
49	Stock mínimo	1/06/18	8/06/18	8
50	Sprint 2- 4ta reunión con Cliente, entrega avance 3ra reunión	4/06/18	4/06/18	1
51	Reunión con tutor	07/06/18	07/06/18	1
52	Aviso de entrega realizada	9/06/18	10/06/18	2
53	Mantenimiento de ordenes de trabajo	11/06/18	15/06/18	5
54	Reunión con tutor	14/06/18	14/06/18	1
55	Sistema de alertas	16/06/18	17/06/18	2
56	Búsquedas	18/06/18	22/06/18	5
57	5ta reunión con Cliente, entrega avance 4ta reunión	20/06/18	20/06/18	1
58	Reunión con tutor	21/06/18	21/06/18	1
59	Visualización información en mapa	23/06/18	15/07/18	23
60	Reunión con tutor	28/06/18	28/06/18	1
61	Reunión con tutor	5/07/18	5/07/18	1
62	Construcción de pruebas	09/07/18	09/07/18	1
63	Testeo y refactorización	10/07/18	10/07/18	1
64	Reunión con tutor	12/07/18	12/07/18	1
65	Sprint 3- 6ta reunión con Cliente, entrega avance 5ta reunión	16/07/18	16/07/18	1
66	Sistema de fidelización	17/07/18	25/07/18	9
67	Reunión con tutor	19/07/18	19/07/18	1
68	Configuración de búsquedas	20/07/18	23/07/18	4
69	Reunión con tutor	26/07/18	26/07/18	1
70	Calificar	26/07/18	29/07/18	4
71	Aviso de entrega realizada	16/07/18	29/07/18	14
72	Refactorización y mejora de funcionalidades	25/07/18	30/07/18	6
73	Sistema de información geográfica	22/07/18	30/07/18	9
74	Sprint 4- 7ma reunión con Cliente, entrega avance 6ta reunión	30/07/18	30/07/18	1
75	Mostrar servicios contratados	31/07/18	5/08/18	6
76	Reunión con tutor	2/08/18	2/08/18	1
77	Validaciones	12/08/18	19/08/18	8
78	Documentación	29/08/18	06/08/18	9
79	Construcción de pruebas	5/08/18	5/08/18	1
80	Testeo y refactorización	5/08/18	6/08/18	2
81	Sistema de información geográfica	31/07/18	06/08/18	7
82	Investigación	20/07/18	06/08/18	18
83	Sistema de fidelización	31/07/18	06/08/18	7
84	Sprint 5- 8va reunión con Cliente, entrega avance 7ma reunión	6/08/18	6/08/18	1

85	Emisión de informes y validaciones	04/08/18	19/08/18	16
86	Investigación	6/08/18	10/08/18	5
87	Refactorización y testeo	9/08/18	14/08/18	6
88	Reunión con tutor	9/08/18	9/08/18	1
89	Mejora del SIG	8/08/18	20/08/18	13
90	App móvil	10/08/18	18/08/18	9
91	Corrección y control de errores	12/08/18	19/08/18	8
92	Reunión con tutor	16/08/18	16/08/18	1
93	Sprint 6- 9na reunión con Cliente, entrega avance 8va reunión	20/08/18	20/08/18	1
94	Optimización diseño, interfaz de usuario	21/08/18	25/09/18	36
95	Construcción e implementación de pruebas	21/08/18	22/08/18	2
96	Testeo y refactorización	22/08/18	25/08/18	4
97	Plan de liberación, deploy	23/08/18	23/08/18	1
98	Presentación	20/09/18	26/09/18	7
99	Sprint 7- 10ma reunión con Cliente, entrega del proyecto	26/09/18	26/09/18	1

Tabla 13. Estimación de esfuerzo según funcionalidades.

En las siguientes gráficas se compara el tiempo estimado para las funcionalidades en el anteproyecto contra el tiempo que realmente llevó el desarrollo del proyecto.

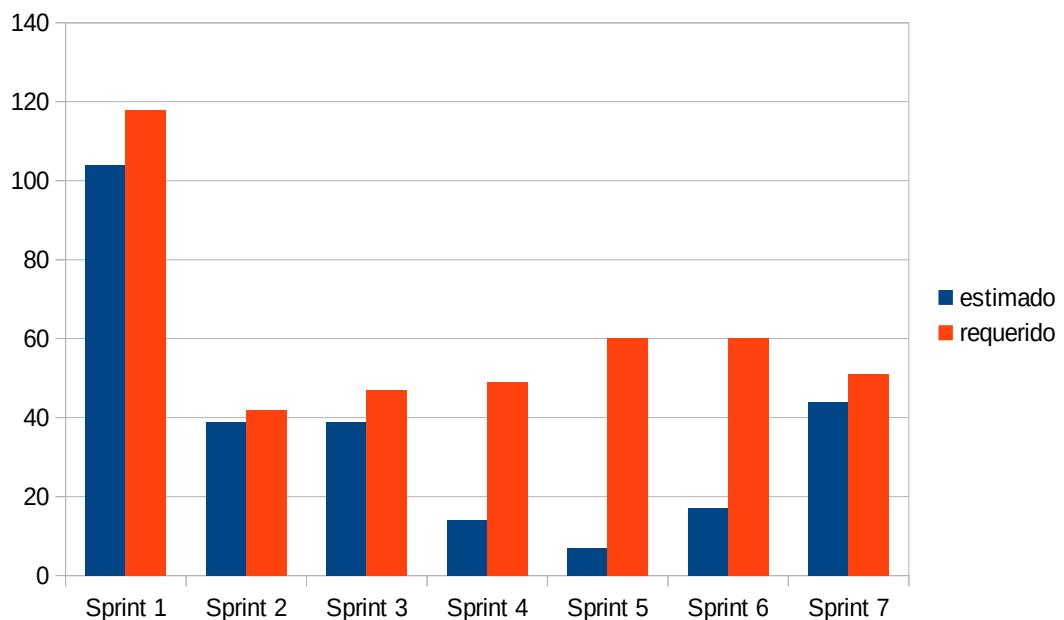


Ilustración 15. Funcionalidades por sprint según cantidad de días.

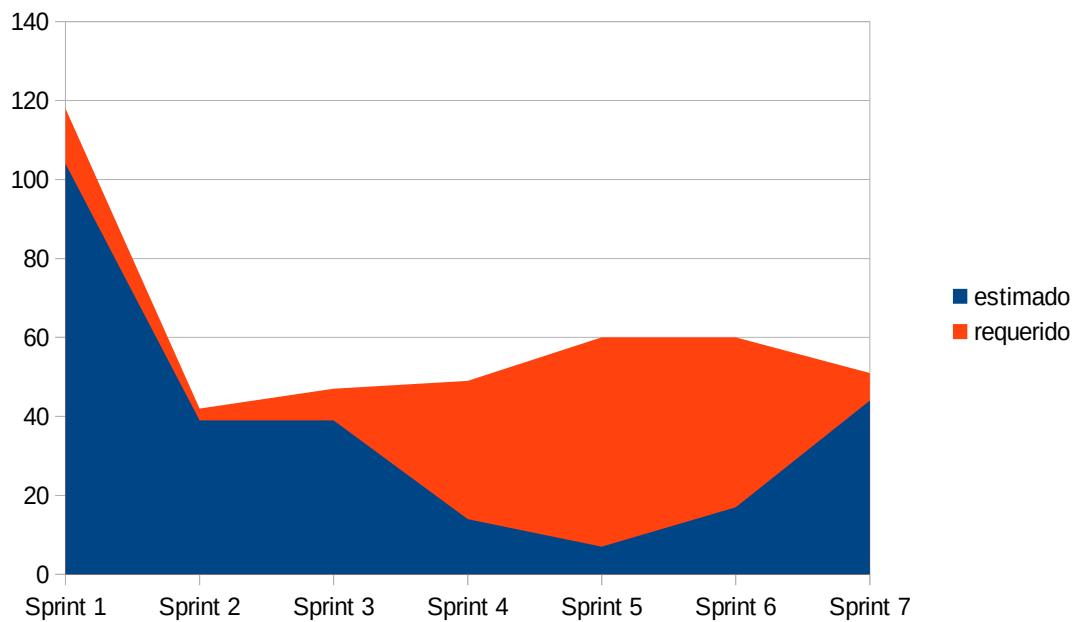


Ilustración 16. Funcionalidades por sprint según cantidad de días.

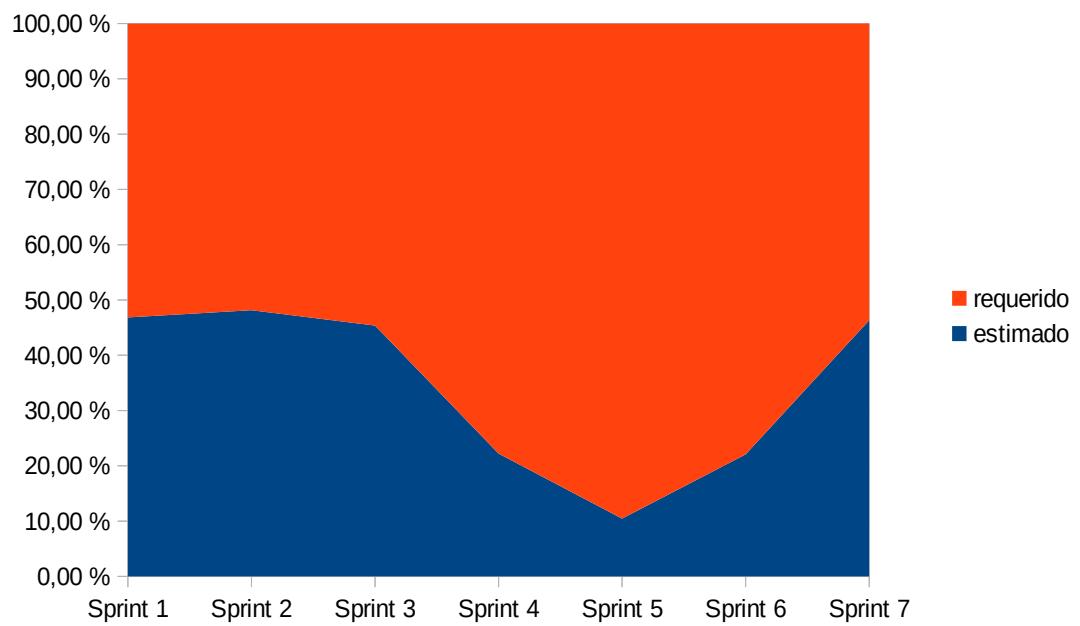


Ilustración 17. Funcionalidades por sprint según porcentaje.

### **3. Conclusiones.**

Es importante destacar que el cliente quedó sumamente satisfecho con lo logrado ya que se cumplieron los objetivos planteados, gracias a esto permitió extender nuestro vínculo más haya del proyecto de fin de carrera para seguir trabajando en el sistema.

Se logró una primera versión de un sistema que se autogestiona, permitiendo automatizar tareas y hacer más eficiente la exposición de la información para la toma de decisiones. Debido a esto no se necesita de usuario especializado para su mantenimiento. Dicha aplicación sirve como punto de partida ya que a raíz de lo desarrollado surgieron nuevas ideas para seguir mejorando funcionalidades y la lógica del negocio.

Algunos puntos a destacar son:

- La importancia del entendimiento de la lógica de negocio desde el inicio del proyecto y de sus requerimientos es crucial para un buen desarrollo, simple y ágil del proyecto, ya redefinir procesos lleva mucho tiempo y un alto costo. Para esto es fundamental contar con el cliente durante todo el proceso.
- El tiempo total necesario para el desarrollo fue sustancialmente mayor al planteado inicialmente. Por lo cual, para poder mitigar imprevistos es crucial tener márgenes con los cuales poder moverse para contención de riesgos.
- Debido a no poder subir el sitio a la web se tuvo que hacer ajustes metodológicos que implicaron mayor tiempos, ya que se tuvo que aumentar el tiempo de reuniones con el cliente y el tiempo para testear la aplicación ya que no se pudo contar con la retroalimentación constante que se esperaba durante todo el proyecto por parte del cliente y el grupo objetivo.

Poder contar con el cliente y un grupo objetivo como testers es importante ya que se traduce en un considerable ahorro de tiempo y aumento de la calidad del producto final.

- La integración de tecnologías, web, integración de arquitecturas, aplicación móvil y sobre todo para la parte del desarrollo del sistema de información geográfica era un reto que se presentaba al inicio del proyecto. En base a dedicar buena cantidad de horas del proyecto en investigación se pudo lograr.
- Para llegar a cumplir con los objetivos fue efectiva la forma de abordar el proyecto. Se planteo desde el comienzo abordar con la mayor exigencia, planteando tener todos los requerimientos al menos cuatro días antes de las fechas estipuladas, comenzando por las tareas de mayor criticidad en la medida que fuese posible.

- En cuanto al diseño se intento que prevalezca la simplicidad en la comunicación, posibilitando un fácil acceso a la información y que por esto no deje de ser un diseño elegante. Se desarrollaron funcionalidades con menús poco profundos y de rápido acceso.
- Sobre la relación entre componentes se trabajo para mantener un mayor grado de cohesión y un bajo grado de acoplamiento. Con esto se logra tener mejor dependencia a la hora de realizar cambios, logrando tener la menor repercusión posible en el resto de las clases y potenciando la reutilización.
- Lo anteriormente referido, repercute en lograr eficiencia en la usabilidad, tomando como criterio básico lo intuitivo y simple.
- Como concepto de calidad se preponderó la ausencia de errores en el desarrollo de la implementación, en lugar de, desarrollar mayor cantidad de funcionalidades pero sin control en los distintos componentes que influyen en la calidad del producto.
- En base a todo lo realizado fueron surgiendo nuevas funcionalidades y cosas a mejorar en futuras versiones, como ser, las funcionalidades en las gráficas y en el sistema de información. También es factible de mejora la presentación de informes, delimitación y adjudicación de zonas de repartos, optimizar rutas de reparto e incluir la posibilidad de análisis de rutas. Además se puede seguir optimizando la refactorización de código y la presentación de los pdfs.

Otra mejora prevista es implementar hacer el seguimiento de los mensajes que se reciben y envían para tener información que permita generar un sistema de respuestas automática. Otra funcionalidad a seguir desarrollando es en la optimización del geocodificador.

Todo lo que redunde en ahorro de tiempo y aumento de calidad es de gran importancia ya que reduce la presión por cumplir con los plazos estipulados y disminuye la posibilidad de errores. Permitiendo la construcción de un producto final que cubra las necesidades del cliente.

### **3.1. Lecciones aprendidas.**

El principal aprendizaje fue enfrentarse al desarrollo de un producto para una aplicación real de sus funcionalidades, en donde en primera instancia el principal problema a resolver no fue enfrentarse a la parte técnica sino resolver de forma adecuada de interacción entre los distintos actores en el proyecto. Fue crucial minimizar las dependencias de terceros, ya sea para obtener información o prestación de servicios y en los momentos que no se pudo minimizar este riesgo, se abordó lo antes posible la temática para tener un panorama claro y reducir riesgos.

También se aprendió a mejorar la estimación de tiempo y la importancia de dejar siempre un margen para contingencias.

Se pudo probar la agilidad que presenta la implementación de la metodología Scrum y es fundamental el contacto constante con el cliente, que este actúe como tester soluciona y simplifica posibles problemas además permite ver al cliente su responsabilidad en el desarrollo del proyecto y la importancia de su participación.

Fue un error dar por echo que se podía subir el sistema al sitio de azure con el plan disponible, por lo que se aprendió a evaluar las ventajas y desventajas de distintos sitios de alojamiento web según las tecnologías que se implementen, permitiendo evitar para futuros proyectos los problemas que se tuvieron al intentar subir el sistema a la web.

Para afrontar los problemas acontecidos fue crucial tener días de respaldo que permitieron dedicar más esfuerzo para llegar a cubrir los requerimientos y minimizar la sobrecarga a los siguientes sprints.

## **4. Referencias bibliográficas.**

Todos los sitios fueron visitados en el periodo de Marzo-Septiembre de 2018.

- [1] Fundación Wikipedia. *Wikipedia, la enciclopedia libre. MySQL*. [online]. Disponible: <https://es.wikipedia.org/wiki/MySQL>.
- [2] Fundación Wikipedia. *Wikipedia, la enciclopedia libre. Caja negra (sistemas)*. [online]. Disponible: [https://es.wikipedia.org/wiki/Caja\\_negra\\_\(sistemas\)](https://es.wikipedia.org/wiki/Caja_negra_(sistemas)).
- [3] Fundación Wikipedia. *Wikipedia, la enciclopedia libre. Pruebas de caja blanca*. [online]. Disponible: [https://es.wikipedia.org/wiki/Pruebas\\_de\\_caja\\_blanca](https://es.wikipedia.org/wiki/Pruebas_de_caja_blanca).
- [4] Universitas Miguel Hernández. *GeoServer vs Mapserver*. 2013. [online]. Disponible: <http://mastervcs.edu.umh.es/2013/09/03/geoserver-vs-mapserver/>.

## 5. Bibliografía.

Todos los sitios fueron visitados en el periodo de Marzo a Septiembre de 2018.

- AdminLTE Control Panel Template. [online]. Disponible: <https://adminlte.io/>
- Android app Builder. [online]. Disponible: <https://websitetoapk.com/>
- Fundación Wikipedia. *Wikipedia, la enciclopedia libre*. [online]. Disponible: <http://es.wikipedia.org/>
- Gamma Erich, Helm Richard, Johnson Ralph, Vlissides John. *Design patterns*. Massachusetts: Addison-Wesley, 1995.
- González, Raúl. *Material del curso de Ingeniería de Software, carrera Analista en Tecnologías de la Información Universidad ORT*, 2016.
- Gutiérrez, Juan. *Creación de servicio REST*. 2017. [online]. Disponible: <https://www.youtube.com/watch?v=kZQ60EW6gLg>
- JasperReports Library. 2018. [online]. Disponible: <https://community.jaspersoft.com/project/jasperreports-library>
- Mysql. *Reference Manual*. 2018. [online]. Disponible: <https://dev.mysql.com/doc/refman/8.0/en/>
- Nielsen, Jakob. *Designing Web Usability*. Indianapolis: New Riders, 2000.
- Nielsen, Jakob. *Usabilidad: Diseño de sitios web*. Madrid: Prentice Hall, 2000.
- Oracle, JavaMail. 2018. [online]. Disponible: <http://www.oracle.com/technetwork/java/javamail/index.html>
- PostGIS, Spatial and Geographic objects for PostgreSQL. 2018. [online]. Disponible: <https://postgis.net>.
- PostgreSQL. [online]. Disponible: <https://www.postgresql.org>
- Pressman, Roger S. *Ingeniería del Software: Un enfoque práctico*. México D. F.: Mc Graw-Hill, 2010.
- Qgis. 2018. [online]. Disponible: <https://www.qgis.org/es/site/>
- Ramsey, Paul. *Manual PostGIS*. 2018. [online]. Disponible: <http://postgis.refractions.net/documentation/postgis-spanish.pdf>

- Sherman, Gary. *The PyQGIS programmer's guide: extending QGIS 3.x with Python 3*. Chugiak: Locate Press, 2018.
- Sommerville. *Ingeniería del software*. Madrid: Pearson Educación, 2005.
- Stack Overflow comunidad de desarrolladores para encontrar soluciones a problemas de programación. [online]. Disponible: <https://stackoverflow.com/>
- Udacity, plataforma de aprendizaje en línea. [online]. Disponible: <https://www.udacity.com/>
- Udemy, plataforma de aprendizaje en línea. [online]. Disponible: <https://www.udemy.com/>
- W3schools, Sitio de consulta para desarrollo web. [online]. Disponible: <https://www.w3schools.com>

## **6. Anexos**

### **6.1. Anexo 1. Manual de Usuario.**

Una de las intenciones al generar un sistema intuitivo es reducir la necesidad de leer manuales de usuario para poder utilizar el sistema eficientemente, por lo cual la interfaz se desarrolla para que con saber las funcionalidades disponibles y lo que hacen, ya alcanza para poder ser productivo.

Se describe el funcionamiento en base a las funcionalidades

A continuación se detallan las funcionalidades desglosadas por sitio de acceso al cliente, sitio de acceso para empleados y aplicación móvil.

#### **6.1.1. Funcionalidades. Sitio Cliente.**

1. Login. Permite el acceso a realizar compras para usuarios ya registrados.

 info@info.com

[Inicio](#)   [Nosotros](#)   [Productos](#)   [Novedades](#)   [Recetas](#)   [Trabajar con Nosotros](#)   [Contacto](#)

Para poder comprar debe iniciar sesión, si no tiene una cuenta debe [registrarse](#)



The image shows a login form with two input fields: 'Email' and 'Password'. The 'Email' field has a small envelope icon to its right. The 'Password' field has a small lock icon to its right. Below the fields is a green 'Login' button.

Ilustración 18. Login.

2. Formulario de registro. Inscripción para nuevos usuarios que deseen poder operar en el sitio. Luego del registro se redirige al login.
3. Listado de productos, aquí se ven todos los productos.
4. Carrito de compras. Funcionalidad para poder comprar en el sitio.
5. Modificar datos. Permite modificar los datos de contacto ingresados por el usuario.
6. Visualización de pedidos vigentes. Permite hacer seguimientos de los pedidos.
7. Visualización de pedidos finalizados.
8. Visualización de la factura correspondiente a cada pedido.
9. Posibilidad de calificar servicio y agregar observación.
10. Agendar envío. Se registran los datos de envío para poder consolidar la compra.

### **6.1.2. Funcionalidades. Sitio Empleado.**

1. Cuenta con todas las funcionalidades del cliente.
2. Pantalla de inicio, con alertas de pedidos sin entregar y productos con stock a reponer, gráfica de cantidad de pedidos según mes de entrega de los mismos, resúmenes y accesos rápidos a la información.



Ilustración 19. Menú administrador.

3. Repartos.

- i. Pendientes para el día, se muestran los repartos pendientes a repartir para el día cursos, los cuales están disponibles para ser asignados a un empleado.
- ii. Acceso a los repartos propios en curso para el día, aquí a cada usuario le aparecerán sólo los repartos que tiene adjudicados para repartir.
- iii. Visualización de todos los repartos en curso para el día, aquí se podrán visualizar tanto los repartos propios como los de otros empleados.
- iv. Estado histórico de los repartos.

Estado Repartos							
Repertos							
Mostrar 10 ▾ registros							Buscar:
Id Reparto	Fecha de Entrega	Id Pedido	Pedidos Actuales	Zona	Vehículo	Máximo de Pedidos	Empleado a cargo
1	2018-09-16	7	1	a	DMA123	2	0
2	2018-09-23	6	1	b	ABC456	2	0
2	2018-08-23	1	2	b	ABC456	2	0
2	2018-09-02	5	1	b	ABC456	2	0
2	2018-09-16	2	2	b	ABC456	2	0

Mostrando 1 a 5 de 5 registros

Anterior	1	Siguiente
----------	---	-----------

Ilustración 20. Histórico repartos.

4. Altas.
  - i. Empleado
  - ii. Producto
  - iii. Categoría
  - iv. Marca

v. Reparto

vi. Proveedor

The screenshot shows a software application window titled "Productos". On the left, there is a form for "Alta Producto" (New Product) with fields for "Nombre" (Name), "Precio del proveedor" (Supplier Price), "Precio de Venta" (Sales Price), and "Stock" (Stock). Below this form is a table titled "Marcas Existentes" (Existing Brands) displaying two entries:

Id marca	Nombre
1	Los Niettos
2	Doña Celinda

At the bottom of the screen, there is a navigation bar with icons for back, forward, and search, and the text "Usuario: gianmauro@adinet.com.uy".

Ilustración 21. Alta Producto.

5. Ver ficha de producto.
6. Control de inventario.
  - i. Control de stock.
    - a) Entrada y salida del almacén.
    - b) Adjudicación y estado de los productos en los pedidos.
  - ii. Reponer stock, stock mínimo. Permite ver sólo los productos que llegaron a su stock mínimo y reponerlos, si hay varios lotes se repone a partir del último.
  - iii. Valuación de stock y reposición. Aquí se ven todos los productos en el sistema, su valuación y permite realizar la reposición de cada producto, permite reponer sin importar el lote.
  - iv. Posibilidad de dar de baja al stock existente.
7. Asignación de empleado a reparto.

Pedidos del día para adjudicar a reparto					
Reparto 2					
Zona	Vehículo	Máximo de Pedidos	Pedidos Actuales	Empleado a cargo	
b	ABC456	2	6		<button>Repartir</button>
Pedidos en reparto 2					
Id de Pedido	Fecha Entrega	Dirección	Horario	Cliente	Estado
2	2018-09-22	felipe sanguineti 2477	mañana	empé	Pendiente reparto

Ilustración 22. Asignación Empleado a reparto.

8. Relevamiento de entrega realizada.
9. Informes.
  - i. Listado de los productos
  - ii. Cantidad de pedidos por mes
  - iii. Cantidad de pedidos por cliente.
  - iv. Cantidad de ventas por fecha.
  - v. Monto de los pedidos por mes
  - vi. Exportar los datos a pdf o la posibilidad de exportarlo para trabajar desde una hoja de cálculo.

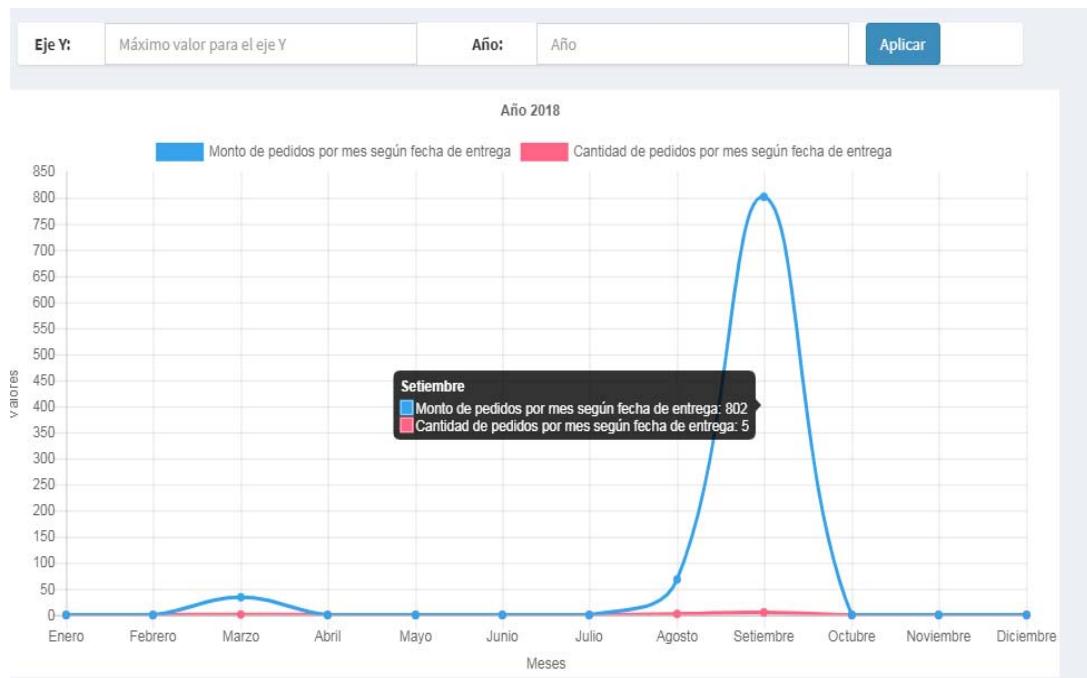


Ilustración 23. Montos de pedidos por mes.

10. Pedidos.

- i. Histórico de pedidos.
- ii. Cantidad de pedidos por reparto, se muestran todos los repartos.
- iii. Cantidad de pedidos por reparto por fecha, se muestran sólo los repartos con pedidos.

11. Selección de visibilidad de los productos.

12. Baja de empleado.

13. Sistema de información geográfica. Cuenta con control de escala, de zoom, posibilidad de maximizar pantalla y opción para imprimir. En cuanto búsquedas, se incluyó las opciones de búsqueda por dirección, por nombre de usuario o proveedor, por id de pedido y por barrio. Para esta última si se busca con la capa de barrios encendida, además de dirigirnos al barrio, se prenderá en verde el área del barrio en cuestión, para la capa de barrios también se tiene la posibilidad de agregar las referencias de representación y al ir pasando sobre los barrios con el mouse se puede ver el área que comprende. También se cuenta con la posibilidad de poder ver los repartos en curso para el día y los pedidos asociados.

Las capas se dividen en las capas base, vista de calle y vista de satélite, en cuanto a capas temáticas se tiene la de usuarios diferenciada en clientes y empleados con sus respectivos datos que se pueden consultar, así como también se puede hacer con la capa de proveedores, pedidos, barrios de Montevideo y zonas de repartos.

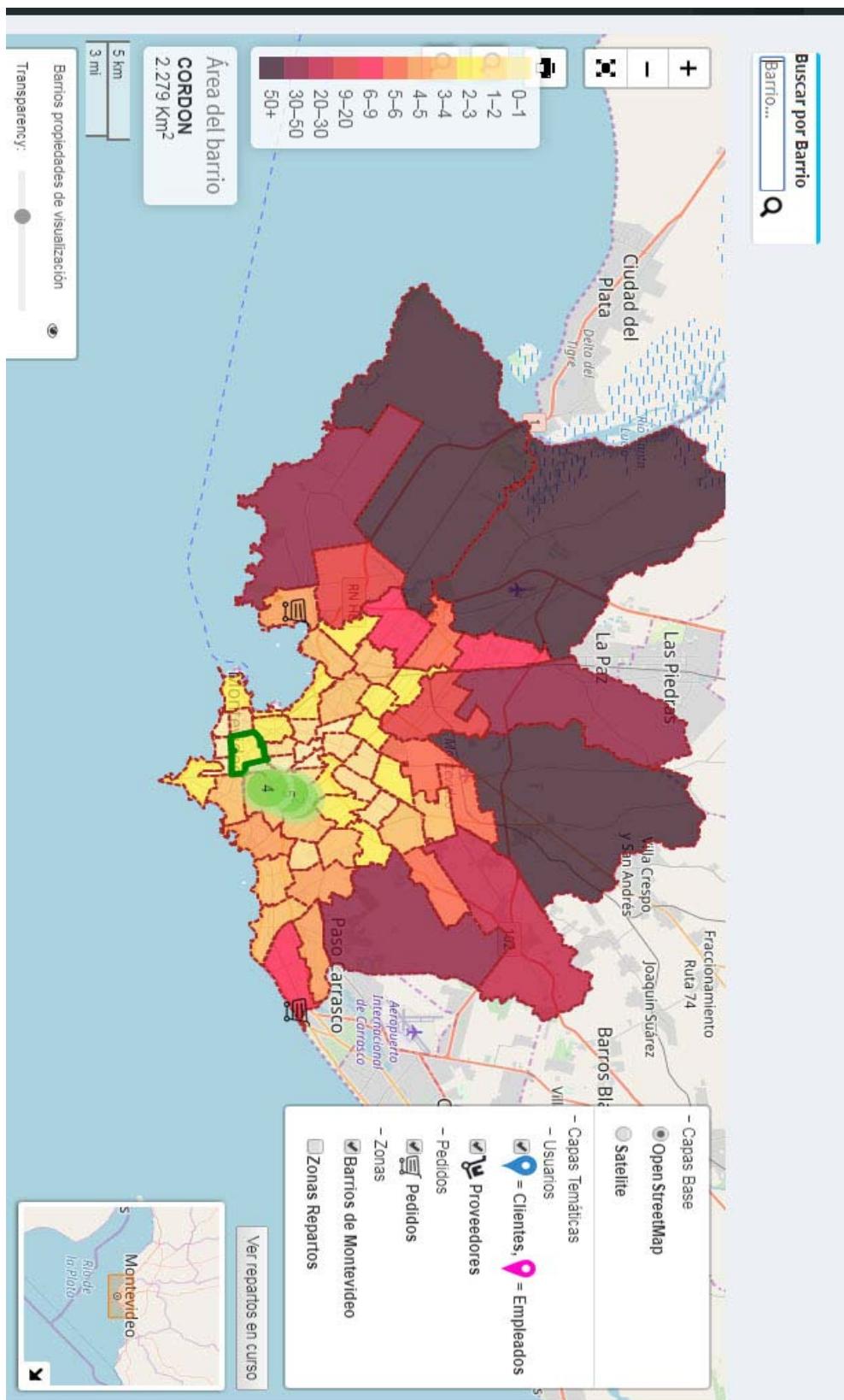


Ilustración 24. Sistema de información geográfica.

14. Modificación de datos de los productos.
15. Visualización de las facturas de los pedidos.
16. Sistema de alertas.
17. Asignar pedido a reparto.
18. Mail.
  - i. Mail de envío de Productos, esta funcionalidad permite elegir la dirección de email desde la cual se desea enviar las promociones.
  - ii. Mail de recepción de consultas, desde aquí se puede elegir la dirección de email en la que se quiere recibir las consultas.
19. Listados.
  - i. Listado de Productos.
  - ii. Listado de Stock Mínimo.
  - iii. Listado y valuación del todo el Stock.
  - iv. Listado de Proveedores.

### **6.1.3. Funcionalidades. Aplicación Móvil.**

1. Login. Acreditación del empleado.

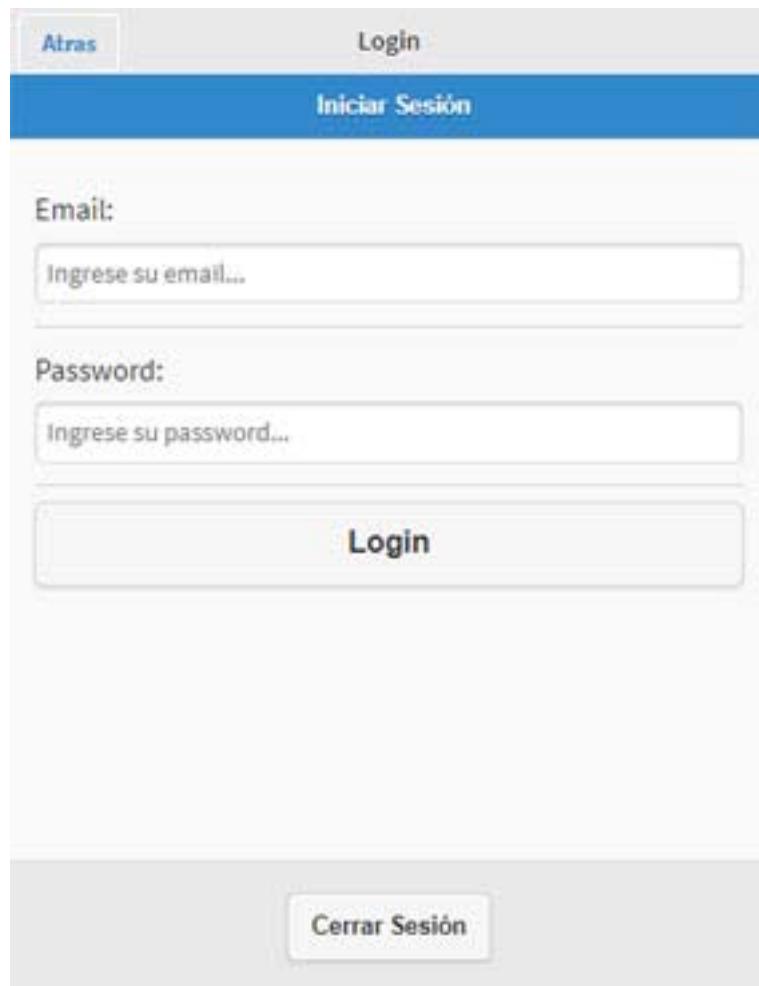


Ilustración 25. App móvil. Login.

2. Vista de repartos asignados al empleado para entregar en el día. Se ven los repartos que tiene pendiente el empleado y cuenta con las opciones para marcar si pudo entregar el pedido o si no se pudo. El estado de los repartos se visualiza en tiempo real desde la aplicación web tanto en los listados como en el seguimiento que se puede realizar desde el sistema de información geográfica.

The image consists of two side-by-side screenshots of a mobile application interface. Both screens have a header bar with 'Atras' (Back) on the left and the title 'Repartos propios para d...' (Own deliveries for the day...) in the center.

**Left Screen (Delivery Details):**

- Header:** 'Repartos propios para el día 22/9/2018' (Own deliveries for the day 22/9/2018)
- Title:** 'Reparto 2'
- Table:** A table with columns 'Observación', 'Dirección', 'Horario', and 'Cliente'. It lists three observations with their respective addresses, times, and clients.
- Buttons:** 'Cerrar Sesión' (Close Session) at the bottom.

**Right Screen (Delivery Status):**

- Header:** 'Repartos propios para d...' (Own deliveries for the day...)
- Title:** 'Reparto 2'
- Table:** A table with columns 'a cargo' (Assigned to) and 'Estado' (Status). It shows three entries, each with a status button labeled 'Entregado' (Delivered) or 'Sin Entregar' (Not Delivered).
- Buttons:** 'Cerrar Sesión' (Close Session) at the bottom.

Ilustración 26. App móvil. Repartos Pendientes para el día.

### 3. Vista ubicación. Se muestra la ubicación actual del reparto.



Ilustración 27. App móvil. Ubicación reparto.

## 6.2. Anexo 2. Casos de uso.

Los casos de uso describen los pasos a realizar para llevar a cabo algún proceso. Representa una interacción entre un usuario y un sistema, describen la interacción con el sistema, por lo tanto, facilitan la identificación de los principales requisitos funcionales.

Para la realización del diagrama de Caso de Uso se elige la funcionalidad ingresar pedido por ser una de las más importantes.

CASO DE USO INGRESAR PEDIDO, DIAGRAMA.

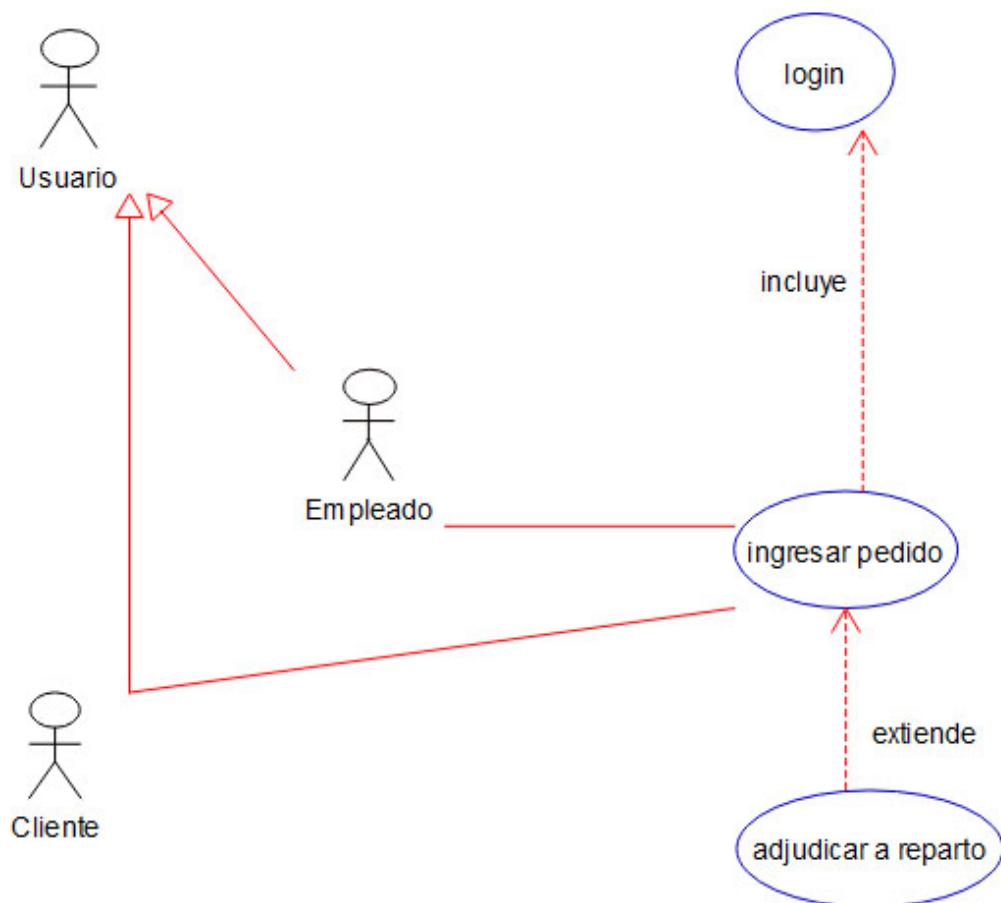


Ilustración 28. Diagrama de caso de uso, ingresar pedido.

## Casos de uso narrativos.

<b>Registro de Cliente</b>	
<b>Actor/es:</b>	Usuario: Cliente.
<b>Descripción:</b> Comportamiento del sistema cuando un usuario se registra como cliente.	
<b>Precondiciones:</b> <ol style="list-style-type: none"><li>El usuario no debe haberse autenticado en el sistema.</li></ol>	
<b>Curso Básico:</b> <ul style="list-style-type: none"><li>El actor ingresa los datos de su ubicación, departamento, calle y n.<sup>o</sup>.</li><li>Se ubica en el mapa y se despliega el resto del formulario.</li><li>Ingresa email, password, fecha de nacimiento, nombre, apellido y teléfono, todos los datos anteriores son sujetos a validación. Tiene opción para indicar si es empresa.</li><li>El sistema verifica los datos ingresados por el actor y verifica que el cliente no esté registrado en la base de datos</li><li>El sistema emite un mensaje de operación exitosa.</li></ul>	
<b>Cursos Alternativos:</b> <ol style="list-style-type: none"><li>El actor cancela la operación. El cliente no se registra.</li><li>Uno o más datos no son correctos. Se marca cada dato en el formulario y se permite su reingreso.</li><li>Ya existe un usuario registrado con ese e-mail. El sistema emite un mensaje informando que el cliente ya está registrado. El cliente no se registra.</li></ol>	
<b>Cursos Excepcionales:</b> <ul style="list-style-type: none"><li>Se pierde la conexión a internet. El cliente no será registrado.</li><li>Hay un error al acceder a la base de datos. El cliente no será registrado.</li></ul>	
<b>Pos condiciones:</b> <ul style="list-style-type: none"><li>El cliente queda ingresado en la base de datos del sistema.</li></ul>	

Tabla 14. Caso de uso narrativo. Registro de Cliente.

<b>Modificar datos Usuario</b>	
<b>Actor/es:</b>	Cliente registrado o Empleado
<b>Descripción:</b> Comportamiento del sistema cuando un usuario: Cliente o Empleado modifica sus datos.	
<b>Precondiciones:</b> <ol style="list-style-type: none"> <li>El usuario debe haberse autenticado en el sistema.</li> </ol>	
<b>Curso Básico:</b> <ul style="list-style-type: none"> <li>El sistema muestra los datos del cliente registrado y permite editarlos.</li> <li>El actor modifica sus datos. Los que se pueden modificar son su departamento, calle, número de puerta, nombre, apellido y teléfono.</li> <li>El sistema verifica los datos ingresados.</li> <li>El sistema emite mensaje de operación exitosa.</li> </ul>	
<b>Cursos Alternativos:</b> <ol style="list-style-type: none"> <li>El actor cancela la operación. No se modifican los datos del usuario y se emite aviso.</li> <li>Uno o más datos no son correctos. Se marca cada dato en el formulario y se permite su reingreso (paso 1), se suspende la modificación del usuario.</li> </ol>	
<b>Cursos Excepcionales:</b> <ul style="list-style-type: none"> <li>Se pierde conexión con el sistema. El usuario no se modifica.</li> <li>Se pierde conexión con la base de datos. El usuario no se modifica.</li> </ul>	
<b>Pos condiciones:</b> <ul style="list-style-type: none"> <li>Se modifican los datos del usuario en el sistema.</li> </ul>	

Tabla 15. Caso de uso narrativo. Modificar datos Usuario

<b>Realizar un pedido</b>	
<b>Actor/es:</b>	Cliente registrado
<b>Descripción:</b> Comportamiento del sistema cuando se compran productos.	
<b>Precondiciones:</b>	
1. El actor se encuentra registrado en el sistema.	
<b>Curso Básico:</b>	
<ul style="list-style-type: none"> <li>• El actor selecciona un producto y lo agrega al carrito.</li> <li>• El sistema actualiza el carrito de compras, con el monto resultante de la suma de los precios de cada uno de los productos y sus respectivas cantidades, y despliega los totales por ítem y total general.</li> <li>• El actor confirma la compra del pedido.</li> <li>• El sistema verifica que el actor esté identificado, que los ítems estén correctamente ingresados, que el pedido tenga al menos un ítem.</li> <li>• El actor ingresa la dirección de entrega y una observación.</li> <li>• El sistema geocodifica la dirección y valida los datos.</li> <li>• El actor elige el horario y día de entrega del pedido.</li> <li>• Se valida el pedido y se emite mensaje de que la operación fue exitosa.</li> </ul>	
<b>Cursos Alternativos:</b>	
<p>1. El actor elimina un ítem del carrito de compras o modifica la cantidad. El sistema actualiza los precios.</p> <p>2. El actor cancela el pedido. No se guarda la información, y el carrito de compras es vaciado.</p> <p>3. El actor no se ha identificado correctamente como cliente. El sistema le solicita que se logue o se registre para poder comprar. El sistema solicita al usuario que ingrese su email y contraseña de cliente. Si lo ingresa continua (paso 1), de lo contrario emite aviso a redirigir al login.</p> <p>4. El monto del pedido no supera el monto mínimo para realizar un pedido. Emite aviso y se cancela la operación.</p> <p>5. El actor cancela. No se confirma el pedido. Vuelve al carrito de compras.</p>	
<b>Cursos Excepcionales:</b>	
<ul style="list-style-type: none"> <li>• Se pierde conexión con el sistema. No se agrega el producto al carrito de compras.</li> <li>• Se pierde conexión con la base de datos. No se registra el pedido.</li> </ul>	

**Pos condiciones:**

- Se registra el pedido y se genera una orden de trabajo.

Tabla 16. Caso de uso narrativo. Realizar un pedido.

<b>Calificación de pedidos</b>	
<b>Actor/es:</b>	Usuario registrado del tipo cliente.
<b>Descripción:</b> Comportamiento del sistema cuando se califican los pedidos	
<b>Precondiciones:</b>	
<ol style="list-style-type: none"><li>1. El usuario debe haberse autenticado en el sistema.</li><li>2. El usuario tiene al menos un pedido ya recibido, entregado.</li></ol>	
<b>Curso Básico:</b>	
<ul style="list-style-type: none"><li>• El actor tiene que seleccionar en su menú “pedidos finalizados” y seleccionar calificar.</li><li>• El sistema verifica que el pedido esté en condiciones de ser calificado.</li><li>• El actor ingresa los comentarios, selecciona una puntuación de 1 a 5 y confirma mediante la acción calificar.</li><li>• El sistema verifica los datos ingresados.</li><li>• El sistema registra la calificación para el pedido.</li><li>• Se despliega en pantalla un mensaje confirmando la calificación.</li></ul>	
<b>Cursos Alternativos:</b>	
1. El actor cancela la acción. No se califica el pedido.	
<b>Cursos Excepcionales:</b>	
<ul style="list-style-type: none"><li>• Se pierde conexión con el sistema. No se califica el pedido.</li><li>• Se pierde conexión con la base de datos. No se califica el pedido.</li></ul>	
<b>Pos condiciones:</b>	
<ul style="list-style-type: none"><li>• Se califica el pedido deseado por el actor.</li></ul>	

Tabla 17. Caso de uso narrativo. Calificación de pedidos.

<b>Repartos propios en curso</b>	
<b>Actor/es:</b>	Usuario: Empleado.
<p><b>Descripción:</b> Comportamiento del sistema cuando un empleado quiere ver sólo los pedidos en repartos que tiene asignados para realizar él.</p>	
<p><b>Precondiciones:</b></p> <ol style="list-style-type: none"> <li>1. El usuario empleado debe haberse autenticado en el sistema.</li> </ol>	
<p><b>Curso Básico:</b></p> <ul style="list-style-type: none"> <li>• El sistema mediante una interfaz gráfica muestra los pedidos asignados a repartos para el empleado logueado que aún no fueron entregados.</li> <li>• El actor puede ver los pedidos que tiene pendientes para entregar en el día.</li> <li>• A medida que va llegando a destino va indicando si pudo entregar o no el pedido, lo cual se visualiza en el sistema en tiempo real. Puede hacer por la app web o por la versión móvil.</li> <li>• A medida que se marca si se pudo entregar o no el pedido se refleja en el estado del mismo.</li> <li>• Cuando se finalizan todos los pedidos se deja de ver en repartos en curso y se puede ver en “Estados Repartos” que se ven todos los repartos y filtrar la visualización según las necesidades.</li> <li>• Si algún pedido no se pudo entregar queda con estado pendiente de reparto y queda planificado para el día siguiente.</li> </ul>	
<p><b>Cursos Alternativos:</b></p> <ol style="list-style-type: none"> <li>1. No hay pedidos pendientes de envío.</li> </ol>	
<p><b>Cursos Excepcionales:</b></p> <ul style="list-style-type: none"> <li>• Se pierde la conexión a internet. No se muestran los pedidos en repartos asignados al empleado.</li> <li>• Hay un error al acceder a la BD. No se muestran los pedidos en repartos asignados al empleado.</li> </ul>	
<p><b>Pos condiciones:</b></p> <ul style="list-style-type: none"> <li>• Se muestran los pedidos asignados a repartos para el empleado logueado que consulta.</li> </ul>	

Tabla 18. Caso de uso narrativo. Repartos propios en curso.

<b>Dar de Baja un empleado.</b>	
<b>Actor/es:</b>	Usuario: Empleado.
<b>Descripción:</b> Comportamiento del sistema cuando se quiere dar de baja a un usuario empleado.	
<b>Precondiciones:</b>	
1. El usuario que va a dar la baja debe estar logueado.	
<b>Curso Básico:</b>	
<ul style="list-style-type: none"> <li>• El sistema mediante una interfaz gráfica muestra todos los usuarios empleados y clientes, cada empleado tiene la posibilidad de dar de baja.</li> <li>• Al darle la baja a un empleado instantáneamente queda como cliente, lo cual se visualiza en forma inmediata en el panel de clientes.</li> </ul>	
<b>Cursos Alternativos:</b>	
<ol style="list-style-type: none"> <li>1. El actor cancela la operación. No se da la baja al empleado.</li> <li>2. Existe un único empleado entonces no se permite la baja.</li> <li>3. El sistema encuentra que el empleado tiene pedidos pendientes. Notifica al actor y no se da la baja.</li> </ol>	
<b>Cursos Excepcionales:</b>	
<ul style="list-style-type: none"> <li>• Se pierde la conexión a internet. El empleado no será dado de baja.</li> <li>• Hay un error al acceder a la BD. El empleado no será dado de baja.</li> </ul>	
<b>Pos condiciones:</b>	
<ul style="list-style-type: none"> <li>• Se da de baja al usuario como empleado y queda como cliente. En ese momento pasa de visualizarse en el panel de empleados y se ve en el de clientes.</li> </ul>	

Tabla 19. Caso de uso narrativo. Dar de baja un empleado.

### 6.3. Anexo 3. Diagramas de secuencia.

#### Diagramas de secuencia. Alta Cliente.

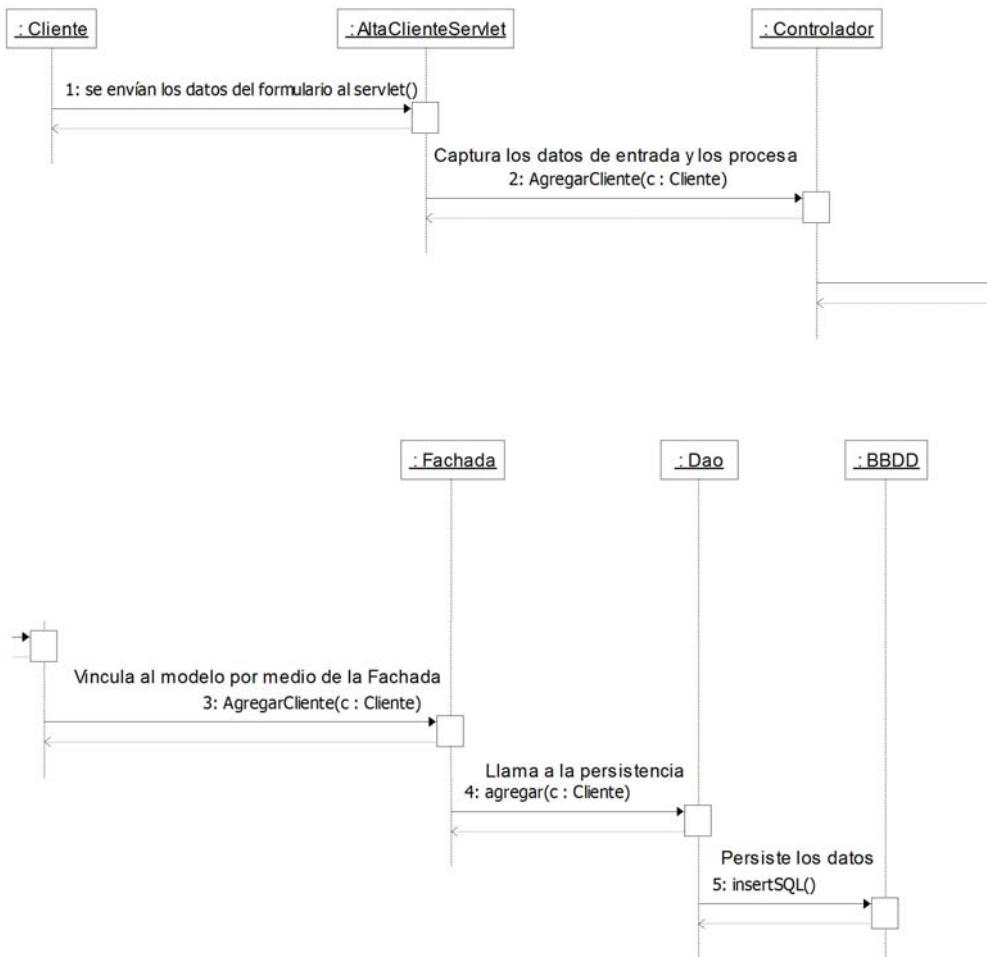


Ilustración 29. Diagrama de secuencia. Alta Cliente.

## Diagramas de secuencia. Alta de Pedido.

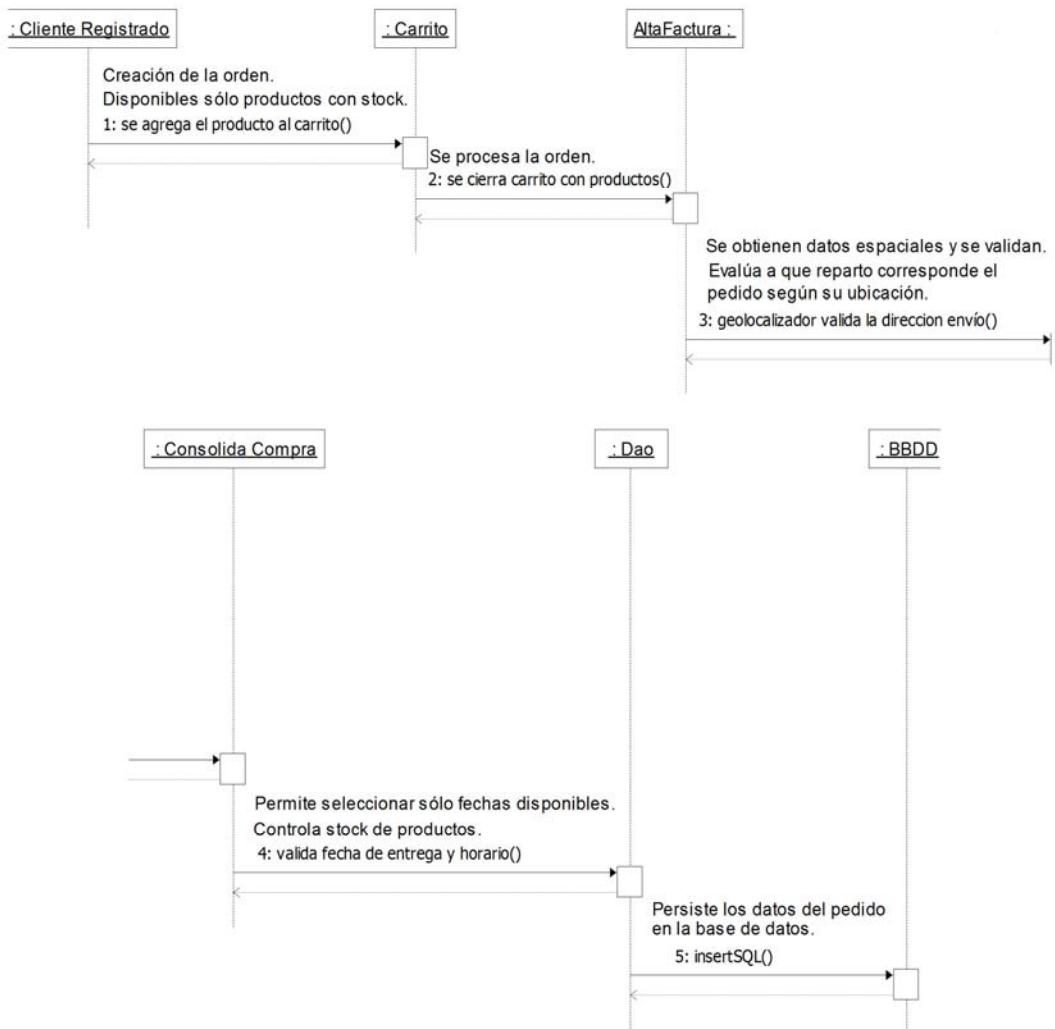


Ilustración 30. Diagrama de secuencia. Alta de Pedido.

#### **6.4. Anexo 4. Diagrama de componentes.**



Ilustración 31. Diagrama de componentes.

#### **6.5. Anexo 5. Diagrama de despliegue.**

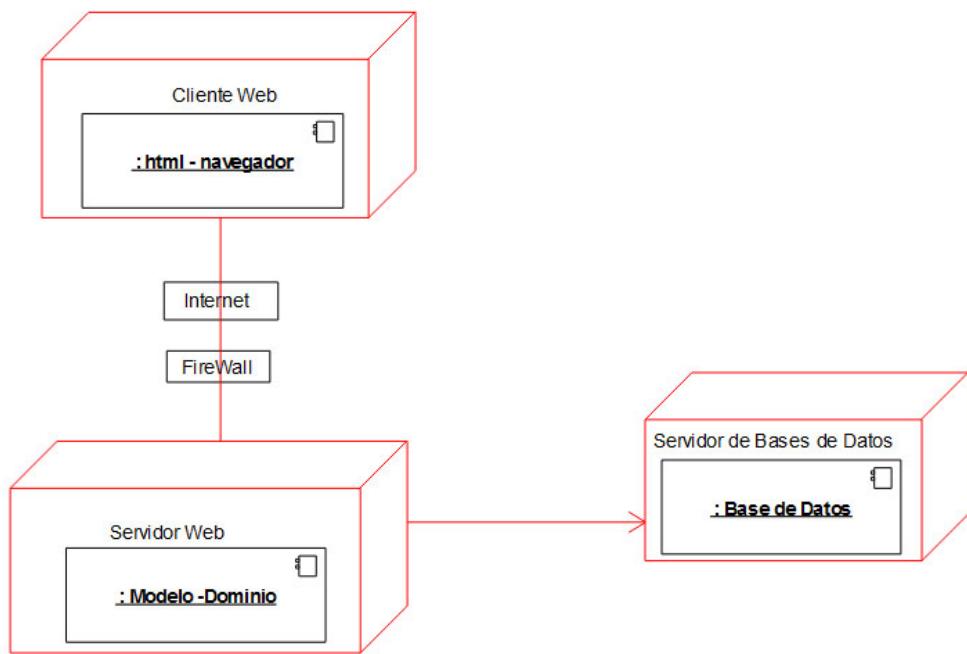


Ilustración 32. Diagrama de despliegue.

## **6.6. Anexo 6. Metodología implementada en las entrevistas.**

En base al material del curso de Ingeniería de Software 2016, profesor: Raúl González y de la bibliografía leída se elaboró un protocolo a seguir en las entrevistas para no olvidar ningún punto a tratar en las entrevistas y poder aprovechar al máximo el tiempo invertido. Por tanto se siguieron los siguientes puntos:

1. Respetar la hora planificada para comienzo y fin de las reuniones.
2. Enviar plan de la reunión, con los temas sugeridos a tratar y la fundamentación
3. Antes de prometer cualquier desarrollo a realizar se debe analizar la complejidad del trabajo, lo cual permita tener claramente definido los alcances, recursos y tiempos necesarios para completar las tareas necesarias.
4. Intentar realizar reuniones no muy extensas.
5. Mostrar avances, problemas a solucionar y lo que se piensa hacer.
6. Luego de la reunión generar un informe de contenido y la resolución de los distintos puntos a fin de desarrollar las funcionalidades correctamente y tener claras las ideas para presentar en la siguiente reunión.
7. Documentar.

## **6.7. Anexo 7. Proceso de elaboración aplicado.**

Para minimizar complicaciones se intentó mantener la siguiente metodología:

1. Planteamiento del problema.
2. Representación de los datos.
3. Diseñar un algoritmo que de solución al problema.
4. Optimizar algoritmos.
5. Codificar.
6. Depurar y refactorizar.
7. Documentar.

## **6.8. Anexo 8. Proceso de pruebas.**

Se detallan las pruebas realizadas.

- Login
  - datos correctos, pasos seguidos:
    1. Se va a pantalla de login.
    2. Se completa email y password.
    3. Se selecciona la opción “login”.
    4. Resultado: Se loguea exitosamente el usuario, lo que le permite poder comenzar a operar en el sistema, se redirige a la página de productos.
  - email inexistente, pasos seguidos:
    1. Se va a pantalla de login.
    2. Se completa email (el email ingresado no existe en el sistema) y password.
    3. Se selecciona la opción “login”.
    4. Resultado: El usuario no puede comenzar a operar en el sistema, se envía mensaje de “login incorrecto”.
  - password inexistente , pasos seguidos:
    1. Se va a pantalla de login.
    2. Se completa email y password (el password ingresado no existe en el sistema).
    3. Se selecciona la opción “login”.
    4. Resultado: El usuario no puede comenzar a operar en el sistema, se envía mensaje de “login incorrecto”.
  - email vacío, pasos seguidos:
    1. Se va a pantalla de login.
    2. Se completa password.
    3. Se selecciona la opción “login”.

4. Resultado: El usuario no puede comenzar a operar en el sistema, se envía mensaje pidiendo que se complete el email en formato correcto.
- password vacío, pasos seguidos:
    1. Se va a pantalla de login.
    2. Se completa email.
    3. Se selecciona la opción “login”.

4. Resultado: El usuario no puede comenzar a operar en el sistema, se envía mensaje pidiendo que se ingrese el password.

Para poder comprar debe iniciar sesión, si no tiene una cuenta debe [registrarse](#)

[Novedades](#) [Recetas](#) [Trabajar con Nosotros](#) [Contacto](#)

The diagram illustrates two failed login attempts. In the first attempt, both the email and password fields are filled with placeholder text ('.....'). A red arrow points from this attempt to a 'Login incorrecto' message below a second attempt. In the second attempt, the email field is correctly filled with 'datosincorrectos@prueba.com', but the password field contains placeholder text ('.....'). Both attempts point to a green 'Login' button.

Ilustración 33. Pruebas. Login Incorrecto.

- Alta Empleado
  - datos correctos usuario sin registro en el sistema, pasos seguidos:
    1. Se va a la pantalla de alta.
    2. Primero se ingresan los datos de ubicación, se selecciona “ubicar”.
    3. El sistema geocodifica los datos.
    4. Se completa los datos para registrar al empleado, se selecciona “registrarse”.

5. Resultado: Se registra al empleado para que pueda comenzar a operar en el sistema.
  - o datos correctos usuario registrado en el sistema como cliente
    1. Se siguen los pasos del 1 al 4 del item anterior.
    2. Resultado: Se actualizan los datos y se cambia el tipo de usuario de cliente a empleado.

Ilustración 34. Pruebas. Alta Empleado datos correcto.

- o cada uno de los datos obligatorios se probó dejarlo vacío para ver si funciona la validación, la cual funcionó exitosamente.
- Alta Cliente
  - o datos correctos usuario sin registro en el sistema
    1. Se va a la pantalla de alta.
    2. Primero se ingresan los datos de ubicación, se selecciona “ubicar”.
    3. El sistema geocodifica los datos.
    4. Se completa los datos para registrar al cliente, se selecciona “registrarse”.
    5. Resultado: Se registra al cliente para que pueda comenzar a operar en el sistema.

Para poder comprar debe iniciar sesión, si no tiene una cuenta debe [registrarse](#)

**Se registró el usuario**

The image shows a login interface. At the top, a green bar displays the message "Se registró el usuario". Below this, there are two input fields: "Email" with an envelope icon and "Password" with a lock icon. At the bottom is a green "Login" button.

Ilustración 35. Pruebas. Alta Cliente, datos correctos.

- cada uno de los datos obligatorios se probó dejarlo vacío para ver si funciona la validación, la cual resultó efectiva.
- email cliente repetido, pasos seguidos:
  1. Se va a la pantalla de alta.
  2. Primero se ingresan los datos de ubicación, se selecciona “ubicar”.
  3. El sistema geocodifica los datos.
  4. Se completa los datos para registrar al cliente, el email ingresado ya se encuentra en el sistema, se selecciona “registrarse”.
  5. Resultado: Se redirige a la página de login y se envía mensaje que no se registró al usuario.

- Alta Producto
  - datos correctos, pasos seguidos:
    1. Se va a la pantalla de alta.
    2. Se ingresan los datos del producto, si la marca o la categoría no existen se deben ingresar, desde el alta de producto se cuenta con la opción para realizarlo.
    3. Se selecciona ingresar.
    4. Resultado: Se despliega mensaje “se registró el producto”.

ID marca	Nombre
1	Los Nietitos
2	Doña Celinda

Ilustración 36. Pruebas. Alta Producto, datos correctos.

- nombre de producto y marca ya ingresado en el sistema, pasos seguidos:
  1. Se va a la pantalla de alta.
  2. Se ingresan los datos del producto, se ingresa un nombre y marca ya existentes.
  3. Se selecciona ingresar.
  4. Resultado: Se despliega mensaje que no se pudo realizar el registro del producto, que no se puede duplicar el nombre del

producto para una misma marca y se inicializa la página para poder realizar nuevamente el ingreso.

The screenshot shows a web-based application interface for managing products. On the left, there's a form titled 'Alta Producto' (Add Product) with fields for Name, Price from Supplier, Price to Client, Stock, Detail, Category, and Brand. To the right, there are two lists: 'Marcas Existentes' (Existing Brands) which shows two entries ('Los Nietitos' and 'Doña Celinda') with their IDs (1 and 2), and a 'Productos' (Products) list table with columns: Nombre, id Marca, Lote, Precio proveedor, Precio venta, Stock, id Categoría, Stock mínimo, and I. The table contains two rows for 'Aceitunas' and 'Aceitunas negras'.

Nombre	id Marca	Lote	Precio proveedor	Precio venta	Stock	id Categoría	Stock mínimo	I
Aceitunas	1	1	28.0	34.0	10	1	4	
Aceitunas negras	1	1	30.0	38.0	9	4	2	

Ilustración 37. Pruebas. Alta Producto.

- cada uno de los datos obligatorios se probó dejarlo vacío para ver si funciona la validación, la cual funciona correctamente.
  
  
- Alta Categoría
  - datos correctos, pasos seguidos:
    1. Se va a la pantalla de alta.
    2. Se ingresan los datos de la categoría y en caso de que sea subcategoría de otra categoría se indica de cual.
    3. Se selecciona ingresar.
    4. Resultado: Se ingresa la categoría al sistema.
  - nombre de categoría ya ingresada en el sistema, pasos seguidos:
    1. Se va a la pantalla de alta.
    2. Se ingresan los datos de la categoría, se ingresa un nombre de categoría ya existente en el sistema y en caso de que sea subcategoría de otra categoría se indica de cual.

- 3. Se selecciona ingresar.
  4. Resultado: La categoría no se puede ingresar y se despliega el mensaje indicando que la categoría ya existe.
  - cada uno de los datos obligatorios se prueba dejarlo vacío para ver si funciona la validación, la cual funciona correctamente.
  
- Alta Marca
  - datos correctos, pasos seguidos:
    1. Se va a la pantalla de alta.
    2. Se ingresan los datos de la marca.
    3. Se selecciona ingresar.
    4. Resultado: Se ingresa la marca al sistema.
  - nombre de marca ya ingresada en el sistema, pasos seguidos:
    1. Se va a la pantalla de alta.
    2. Se ingresan los datos de la marca, se ingresa un nombre de marca ya existente en el sistema.
    3. Se selecciona ingresar.
    4. Resultado: La marca no se puede ingresar y se despliega el mensaje que la marca ya existe.
  - cada uno de los datos obligatorios se prueba dejarlo vacío para ver si funciona la validación, la cual funciona correctamente.
  
- Alta Reparto, el alta de repartos está inhabilitada ya que al momento no se pretende ingresar nuevos repartos a la empresa. Al habilitar el alta de repartos se debe poder seleccionar la zona del mismo para lo cual se necesita definir por parte de la empresa las nuevas zonas o si se realiza sobre las reparticiones ya actuales y se tiene más de un reparto en la misma área.
  - datos correctos, pasos seguidos:
    5. Se va a la pantalla de alta.
    6. Se ingresan los datos del reparto.

7. Se selecciona ingresar.
8. Resultado: Se ingresa el reparto al sistema.

Ilustración 38. Pruebas. Alta Reparto, datos correctos.

- cada uno de los datos obligatorios se probó dejarlo vacío para ver si funciona la validación, la cual trabaja correctamente.
- Alta Proveedor
  - datos correctos, pasos seguidos:
    1. Se va a la pantalla de alta.
    2. Primero se ingresan los datos de ubicación, se selecciona “ubicar”.
    3. El sistema geocodifica los datos.
    4. Se completa los datos para registrar al proveedor, se selecciona “registrar”.
    5. Resultado: Se registra el proveedor.
  - cada uno de los datos obligatorios se probó dejarlo vacío para ver si funciona la validación, la cual funciona.

- también se probó con el siguiente paquete de datos:

- Datos correctos
- Sin dirección

localhost:8081 dice  
Geocode no tuvo éxito por la siguiente razón: ZERO\_RESULTS. Debe ingresar correctamente la dirección, si persiste el problema puede comunicarse con nosotros desde la sección contacto.

Aceptar

<b>Departamento:</b>	<b>Calle:</b>	<b>Nº:</b>
departamento	dirección	número de puerta
Latitud	Longitud	
<b>Email:</b>	<b>R.U.C:</b>	
email	RUC	
<b>Nombre:</b>	<b>Teléfono:</b>	
Nombre	teléfono	
<b>Registrar</b>		

Ilustración 39. Pruebas. Alta Proveedor, sin ingresar dirección.

- Con calle, con departamento y sin número
- Con número, con departamento y sin calle
- Con número, con calle y sin departamento
- Sitio Cliente. Mi cuenta.
  - Filtros de búsqueda, se testeó que los filtros de búsqueda por categoría, marca y datos de los productos funcionaran correctamente y luego al restaurar el menú de productos volviera al estado inicial sin los parámetros de la búsqueda.

- Mis Datos
  - Datos correctos
  - Sin datos
  - Datos numéricos en los de texto y viceversa
- Pedidos Vigentes, se comprobó que la visualización de los pedidos fuese correcta, al igual que la factura asociada al pedido.
- Pedidos Finalizados, se comprobó que la visualización de los pedidos fuese correcta, al igual que la factura asociada al pedido y también se testeó la funcionalidad para calificar, la cual tiene controlado el ingreso de datos correctos mediante despliegue de opciones de calificación.



Ilustración 40. Pruebas. Calificación.

- Carrito de compras
  - Agregar productos al carrito, se testeó poder agregar productos al carro de compras, que no se puedan agregar productos sin stock, que funcione correctamente la reposición por lotes y que se visualice el lote más antiguo, salvo que se indique lo contrario.
  - Alta Factura, a este menú se ingresa luego de seleccionar “finalizar compra” en el carrito de compras, aquí se selecciona la ubicación de la entrega y se puede ingresar una observación. Paquete de datos testeados:
    - Datos correctos
    - Sin dirección

- Con calle, con departamento y sin número
- Con número, con departamento y sin calle
- Con número, con calle y sin departamento

## Alta Factura

**Departamento:**

Montevideo

-34.879267400

**Calle:**

felipe sanguinetti

-56.144749200

**Nº:**

2472

**Observación:**

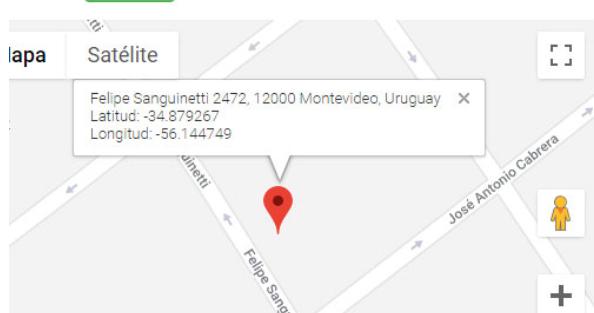
**Ingresar** 

Ilustración 41. Pruebas. Alta Factura.

- Consolidar Compra, las fechas pasibles de ser seleccionadas son del día actual en adelante menos en las que no haya reparto disponible para la entrega de pedidos. Se valida que se ingrese la fecha.

Usuario: gianimauro@adinet.com.uy  
Sesión: iniciada
Menu Administrador 

[Inicio](#) [Nosotros](#) [Productos](#) [Novedades](#) [Recetas](#) [Trabajar con Nosotros](#) [Contacto](#)

**Consolidar compra**

September 2018						
Su	Mo	Tu	We	Th	Fr	Sa
26	27	28	29	30	31	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	1	2	3	4	5	6

Horario de Entrega:  Mañana (de 06 a 12)  Noche (de 17 a 22)

Día de entrega:  

**Ingresar**

Ilustración 42. Pruebas. Consolidar compra.

- Contacto, tienen control de validación en todos los campos.
- Repartos
  - Pendientes para el día, se testeó que funcione la asignación de empleado al reparto.

**Menú de gestión**

Web Pública

FUNCIONALIDADES ADMINISTRADOR:

- Inicio
- Repartos
- Altas
- Stock

Pedidos del día para adjudicar a reparto

Reparto 2

Zona	Vehículo	Máximo de Pedidos	Pedidos Actuales	Empleado a cargo
b	ABC456	2	6	<b>Repartir</b>

Pedidos en reparto 2

ID de Pedido	Fecha Entrega	Dirección	Horario	Cliente	Estado
2	2018-09-22	felipe sanguineti 2472	mañana	emp@emp1	Pendiente_reparto

- Informes
- Geolocalización
- Pedidos
- Productos Visibilidad
- Baja de Empleado
- Mail de envíos

Usuario: gianinmauro

Ilustración 43. Pruebas. Repartos pendientes para el día.

- Propios en curso, se testeó que se visualicen correctamente los repartos y pedidos y se probó que funcione correctamente el indicador de estado del reparto.

Zona	Vehículo	Máximo de Pedidos	Pedidos Actuales	Empleado a cargo
b	ABC456	2	6	0

Situación del Pedido	Observación	Id de Pedido	Fecha Entrega	Dirección	Horario	Cliente	estado
Entregado	Observación	2	2018-09-22	felipe sanguinetti 2472	manana	emp@emp1	Adjudicado_reparto

Ilustración 44. Pruebas. Repartos propios para el día en curso.

- En curso, se muestran todos los repartos en curso. Se verifico que los datos desplegados sean correctos.

Zona	Vehículo	Máximo de Pedidos	Pedidos Actuales	Empleado a cargo
b	ABC456	2	6	0

Id de Pedido	Fecha Entrega	Dirección	Horario	Id usuario	estado
2	2018-09-22	felipe sanguinetti 2472	manana	emp@emp1	Adjudicado_reparto

Ilustración 45. Pruebas. Repartos en Curso.

- Histórico estado de repartos, se verifico que los datos desplegados sean correctos.

Estado Repartos								
Mostrar 10 ▾ registros		Buscar: <input type="text"/>						
Id Reparto	Fecha de Entrega	Id Pedido	Pedidos Actuales	Zona	Vehículo	Máximo de Pedidos	Empleado a cargo	Ver pedidos
1	2018-09-16	7	1	a	DMA123	2	0	<button>Ver Pedidos</button>
2	2018-03-23	6	1	b	ABC456	2	0	<button>Ver Pedidos</button>
2	2018-08-22	3	1	b	ABC456	2	0	<button>Ver Pedidos</button>
2	2018-08-23	1	1	b	ABC456	2	0	<button>Ver Pedidos</button>
2	2018-09-02	5	1	b	ABC456	2	0	<button>Ver Pedidos</button>
2	2018-09-16	4	1	b	ABC456	2	0	<button>Ver Pedidos</button>
2	2018-09-22	2	1	b	ABC456	2	0	<button>Ver Pedidos</button>

Mostrando 1 a 7 de 7 registros

Anterior 1 Siguiente

Ilustración 46. Pruebas. Estado Repartos.

Pedidos								
Pedidos en reparto								
Mostrar 10 ▾ registros		Buscar: <input type="text"/>						
Id	Dirección destino	Fecha Entrega elegida	Horario entrega	Observación	Fecha que se realizó la entrega	Usuario	Estado del pedido	Factura
6	felipe sanguinetti	2018-03-23	mañana	-	2018-03-25	gianimauro@adinet.com.uy	Entregado	<button>Ver Factura</button>

Mostrando 1 a 1 de 1 registros

Anterior 1 Siguiente

Ilustración 47. Pruebas. Pedidos en reparto.

Factura						
Factura						
Mostrar	10	▼	registros	Buscar:		
Foto	Producto	Precio	Cantidad	Total		
	Aceitunas	34.0	1	34.0		

Mostrando 1 a 1 de 1 registros

Anterior 1 Siguiente

- Sub Total 34.0
- Tax/IVA(6%) 6.12
- Total 40,12

Ilustración 48. Pruebas. Factura.

Las funcionalidades en “Repartos” se probaron con distintos paquetes de datos, los siguientes casos fueron testeados con y sin datos anteriores y posteriores al día en curso. Casos de prueba:

- Sin pedidos para el día en curso
- Con pedidos para el día en curso
- Con pedidos para el día en curso para un reparto
- Con pedidos para el día en curso para los dos repartos
  
- Stock
  - Baja de Stock, se testó que los datos desplegados sean correcto, que se pueda dar de baja el stock de los productos y que cuando el producto no tiene lotes disponibles se muestra como “Momentáneamente sin stock”, si hay otro lote con stock se muestra el siguiente lote y se deja de ver el anterior por defecto, en caso que se desee se puede modificar esta opción desde el panel administrador en el control de visualización de productos.
  - Control de Stock (stock mínimo), se controla sólo los productos que estén e el límite de mínimo de productos establecidos por el administrador y que no tengan ningún otro lote con disponibilidad. Al llegar al límite se muestran los productos que necesitan reposición. Se controla que funcione el control para que la fecha de vencimiento no pueda ser anterior a la fecha actual.

- Reposición y valuación de Stock, se testeó que los datos desplegados sean correctos, que el monto de la valuación sea correcto y que se pueda reponer el stock disponible de los distintos lotes.

Stock										
Valuación de Stock										
Mostrar 10 registros Buscar:										
Id	Nombre	Precio	Categoría	Marca	Proveedor	Lote	Stock	Valuación		
1	Aceitunas	28.0	Aceitunas	Los Nietitos	null	1	10	280.0	Reponer Stock	
2	Aceitunas negras	30.0	Aceitunas negras	Los Nietitos	null	1	9	270.0	Reponer Stock	
3	Cerezas	34.0	Cerezas	Doña Celinda	null	1	4	136.0	Reponer Stock	
4	Lentejas	10.0	Legumbres	Los Nietitos	null	1	5	50.0	Reponer Stock	
5	Aceitunas	30.0	Aceitunas	Los Nietitos	null	2	10	300.0	Reponer Stock	

Mostrando 1 a 5 de 5 registros Anterior 1 Siguiente

• Valuación total de todos los productos con stock = 1036,00

Ilustración 49. Pruebas. Stock.

- Informes

- Productos, se controla que el listado de productos sea correcto. También se probaron las funcionalidades para exportar la información a formato pdf y xls para trabajar con hojas de cálculo.

Productos										
Crear PDF Crear Hoja de Cálculo										
Productos										
Nombre	Id	Marca	Lote	Precio proveedor	Precio venta	Stock	id Categoría	Stock mínimo	Proveedor	Visible
Aceitunas	1	1	1	28.0	34.0	10	1	4	null	true
Aceitunas	1	2	2	30.0	38.0	10	1	4	null	false
Aceitunas negras	1	1	1	30.0	38.0	9	4	2	null	true
Cerezas	2	1	1	34.0	40.0	4	3	1	null	true
Lentejas	1	1	1	10.0	20.0	5	2	2	null	true

Mostrando 1 a 5 de 5 registros Anterior 1 Siguiente

Ilustración 50. Pruebas. Productos.

- Cantidad de pedidos por mes, se controla la correcta visualización de la gráfica y que el ingreso de valores para el eje Y.

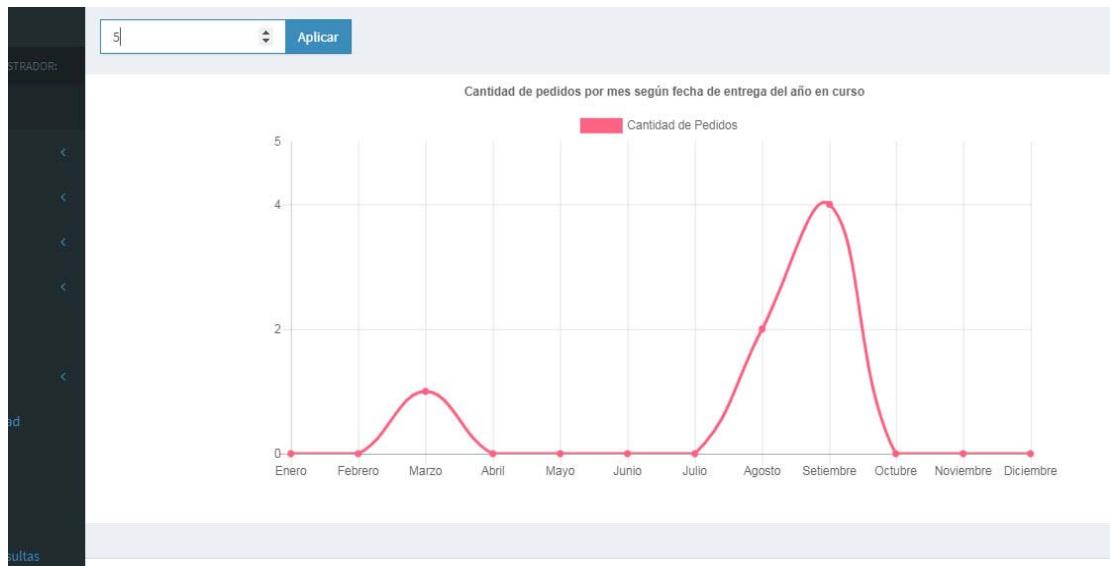


Ilustración 51. Pruebas. Cantidad de pedidos por mes.

- Cantidad de pedidos por fecha, se despliega los datos de todos los pedidos organizados por fecha, se controló contra los datos de la base de datos que fuese correcto el despliegue en la vista.
- Montos de pedidos por mes, se controla la correcta visualización y se testeó para distintos valores del eje Y y de año que no se caiga el sistema, en caso que no existan valores o los datos ingresados sean inconsistentes se muestra toda la gráfica con valor cero.

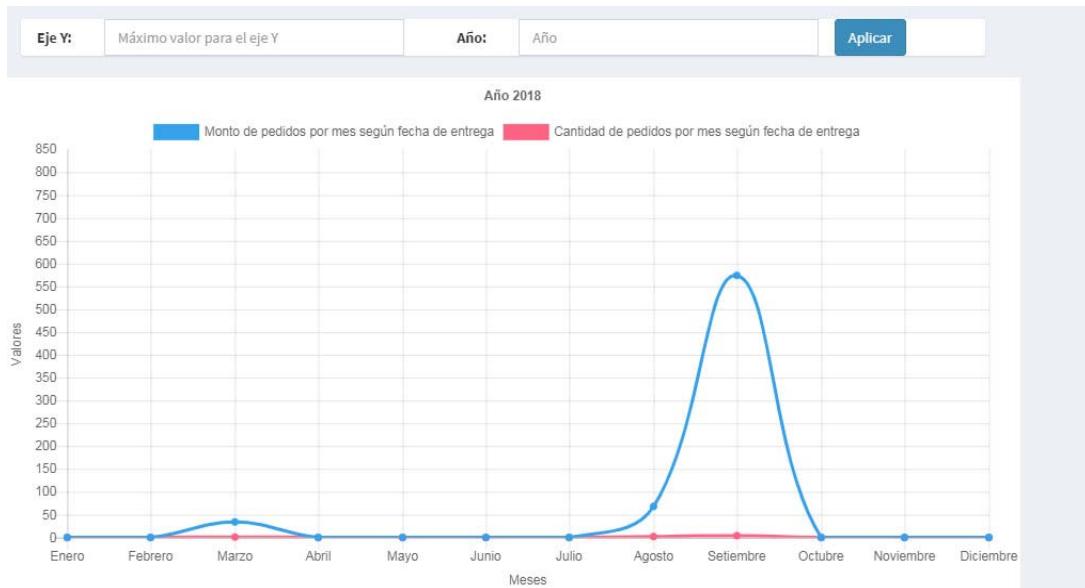


Ilustración 52. Pruebas. Monto contra cantidad de pedidos.

- Cantidad de pedidos por Cliente, se controla que la visualización de los clientes, sus respectivos pedidos y las facturas de estos sea correcta.

**Pedidos**

Cantidad de Pedidos por Cliente

Mostrar	10	registros	Buscar:					
<b>Id</b>	<b>Nombre</b>	<b>Apellido</b>	<b>Email</b>	<b>Departamento</b>	<b>Dirección</b>	<b>Teléfono</b>	<b>Cantidad de Pedidos</b>	<b>Pedidos</b>
1	Mauro	Carlevaro	gianimauro@adinet.com.uy	montevideo	felipe sanguinetti	123456	5	<b>Ver Pedidos</b>
2	Emp	Leado	emp@emp1	montevideo	pablo de maria	874568	1	<b>Ver Pedidos</b>
3	Ana	Rodriguez	cliente@cliente1	Montevideo	av italia	457812	1	<b>Ver Pedidos</b>

**Ventas**

Pedidos

Mostrar	10	registros	Buscar:				
<b>Id</b>	<b>Dirección destino</b>	<b>Fecha Entrega elegida</b>	<b>Horario entrega</b>	<b>Observación</b>	<b>Fecha que se realizó la entrega</b>	<b>Usuario</b>	<b>Factura</b>
2	felipe sanguinetti	2018-09-22	mañana	-	null	emp@emp1	<b>Ver Factura</b>

Mostrando 1 a 1 de 1 registros

**Factura**

Factura

Mostrar	10	registros	Buscar:	
Foto 2	<b>Producto</b>	<b>Precio</b>	<b>Cantidad</b>	<b>Total</b>
	Aceitunas	34.0	1	34.0

Mostrando 1 a 1 de 1 registros

• Sub Total 34.0

Ilustración 53. Pruebas. Cantidad de pedidos por Cliente y datos asociados.

- Geolocalización, en cuanto al sistema de información geográfica se testea que los datos desplegados de clientes, empleados, proveedores y pedidos sea correcto. También se prueban los controles de visualización de capas y referencias, el despliegue de información asociada a cada capa y que el seguimiento del estado de los repartos del día se pueda hacer correctamente.

En cuanto a las búsquedas se ingresaron variados datos para probar que funcionen.

También se testeó el correcto funcionamiento del control de impresión.

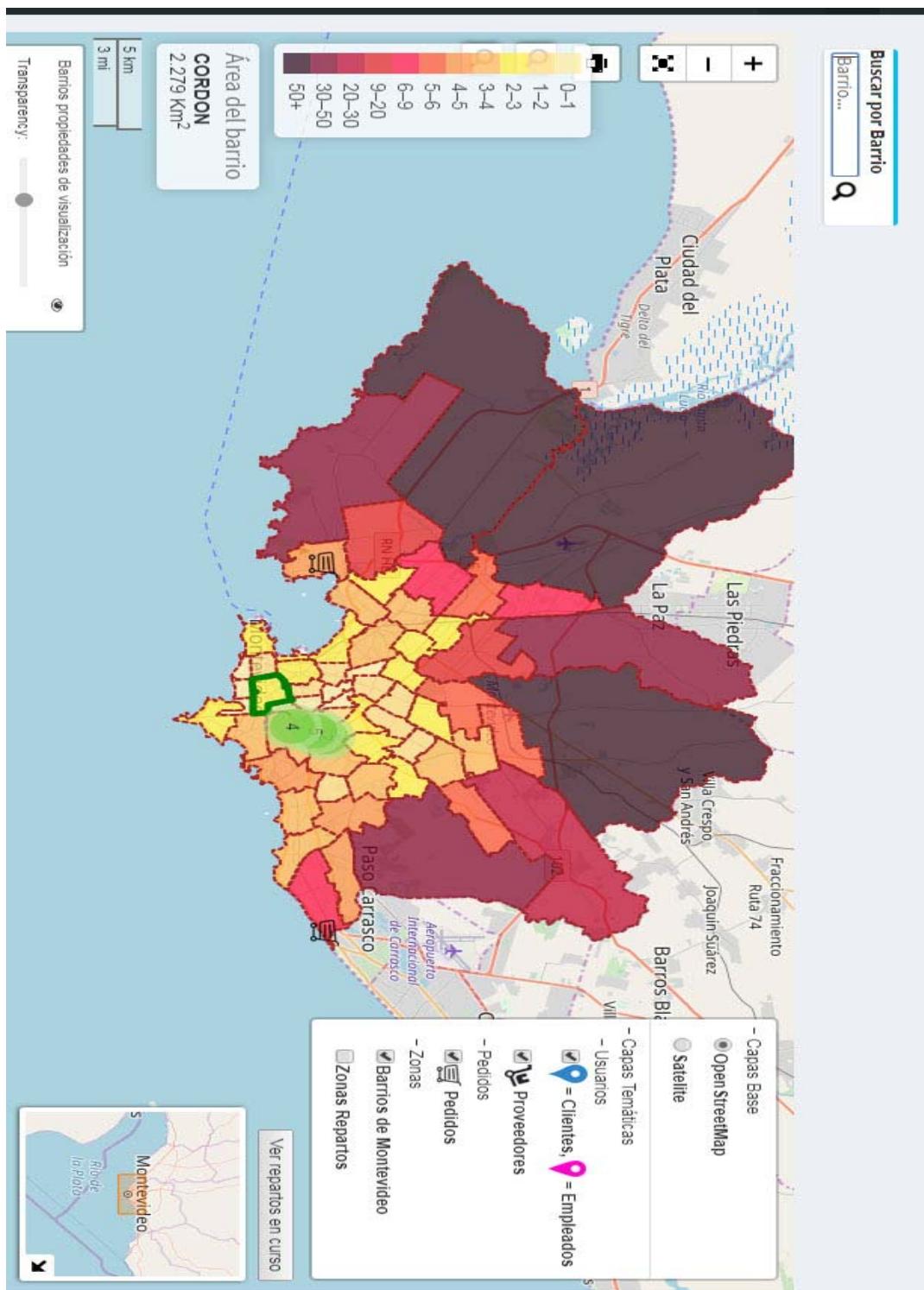


Ilustración 54. Mapa. Road view

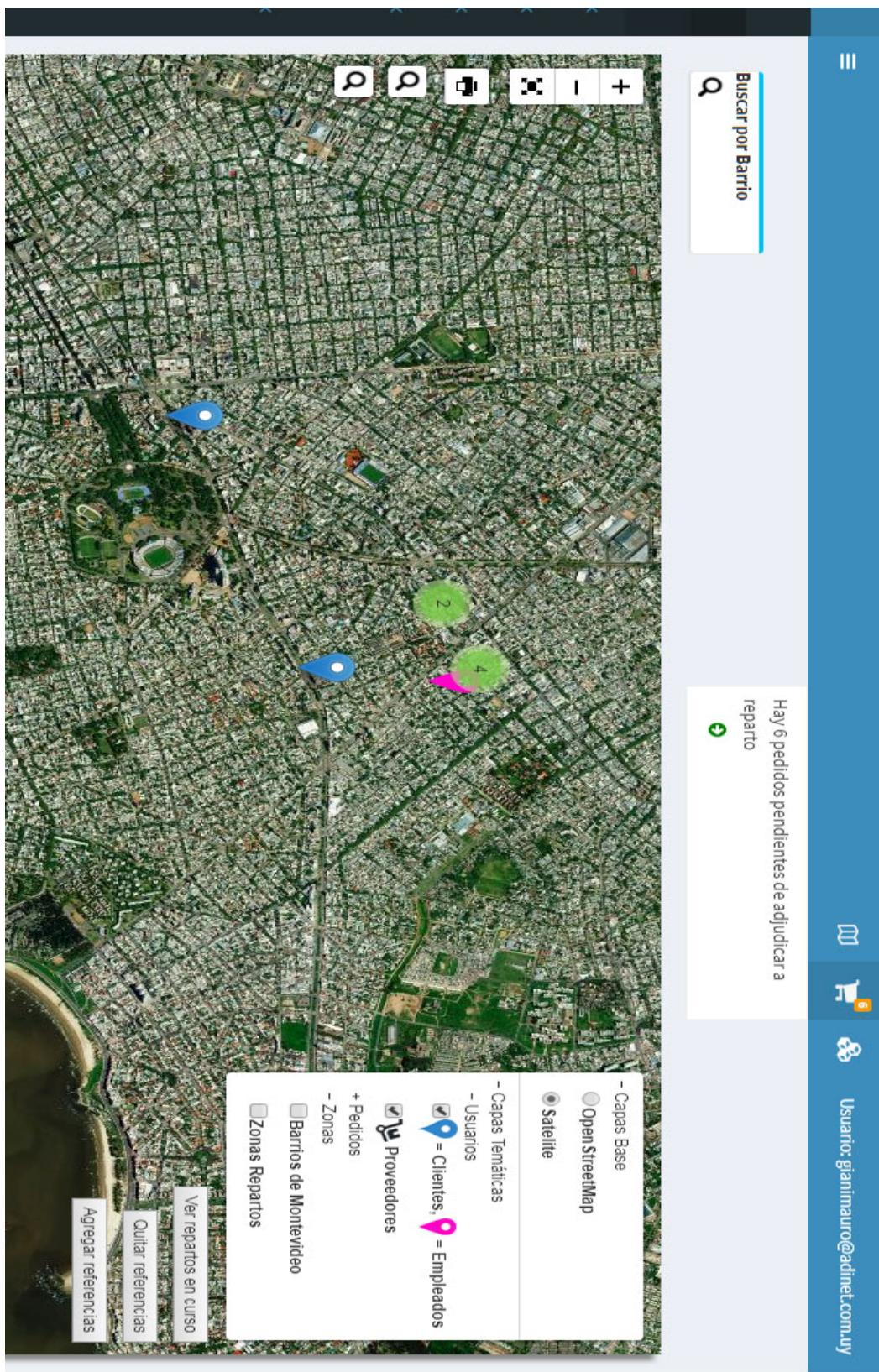


Ilustración 55. Mapa. Satellite view.

- Pedidos
  - Histórico de Pedidos, se controla que se despliegue correctamente la información y que se puedan modificar los datos de los pedidos en caso de ser necesario.
  - Pedidos por Reparto, se controla que se muestre correctamente la información asociada a los repartos.
  - Pedidos por Reparto por Fecha, se controla que se muestre sólo los repartos que han tenido pedido que se puedan ver los pedidos adjudicados y sus facturas correctamente.
- Productos Visibilidad, en este panel se testea que funcione correctamente la posibilidad de cambiar la visibilidad de los productos.
- Baja Empleado, en este panel se controla que se despliegue la información de los empleados y los clientes, al dar de baja un empleado se ve que quede como cliente y así no se pierdan sus datos, la baja de empleados se podrá realizar sólo si no tiene repartos a cargo.

**Baja de Empleado**

Baja de Empleado						
Mostrar:	10	▼	registros	Buscar:		
Id	Nombre	Apellido	Email	Departamento	Dirección	Teléfono
1	Mauro	Carlevaro	gianimauro@adinet.com.uy	montevideo	felipe sanguinetti	123456
2	Emp	Leado	emp@emp1	montevideo	pablo de maria	874568
5	Administrador	Proyecto	adm@admproyecto.com	montevideo	sarmiento	2

Mostrando 1 a 3 de 3 registros

Anterior 1 Siguiente

**Clientes**

Id	Nombre	Apellido	Email	Departamento	Dirección	Teléfono
3	Ana	Rodriguez	cliente@cliente1	Montevideo	av italia	457812
4	Jose	Carbajal	jose@jose	Montevideo	av italia	123458

Ilustración 56. Pruebas. Baja Empleado.

- Mail de Envíos, se probó que trabaje correctamente la funcionalidad.
- Mail de Recepción de Consultas, se testeó exitosamente la funcionalidad.

Dirección desde la que se envían los avisos Por el momento sólo se aceptan emails de gmail

Alta dirección envíos		Direcciones Existentes																					
Dirección de mail:	<input type="text" value="dirección"/>	Mostrar:	10	Buscar:	<input type="text"/>																		
Clave:	<input type="text" value="clave"/>																						
	<input type="button" value="Registrar"/>																						
<table border="1"> <thead> <tr> <th>Id dirección</th> <th>Usuario creador</th> <th>Fecha</th> <th>Uso</th> <th colspan="2">Dirección</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>1</td> <td></td> <td>Envío</td> <td colspan="2">cuentaproyectoati@gmail.co</td> </tr> <tr> <td>2</td> <td>1</td> <td></td> <td>Recepción</td> <td colspan="2">cuentaproyectoati@gmail.co</td> </tr> </tbody> </table>						Id dirección	Usuario creador	Fecha	Uso	Dirección		1	1		Envío	cuentaproyectoati@gmail.co		2	1		Recepción	cuentaproyectoati@gmail.co	
Id dirección	Usuario creador	Fecha	Uso	Dirección																			
1	1		Envío	cuentaproyectoati@gmail.co																			
2	1		Recepción	cuentaproyectoati@gmail.co																			
Mostrando 1 a 2 de 2 registros      Anterior <input type="button" value="1"/> Siguiente <input type="button" value="2"/>																							

Ilustración 57. Pruebas. Mails

## 6.9. Anexo 9. Presentación.

**ORT**  
UNIVERSIDAD ORT  
Uruguay

FACULTAD DE INGENIERÍA  
ING. BERNARD WAND-POLAK

### Automatización de sistema de ventas y análisis de datos, integrado con Sistema de Información Geográfico

Entregado como requisito para la obtención del título de Analista Programador

Giani Mauro Carlevaro de Sosa – 147376  
Tutor: Fernando Thul  
2018

**ORT**  
UNIVERSIDAD ORT  
Uruguay

FACULTAD DE INGENIERÍA  
ING. BERNARD WAND-POLAK

**Cliente.**

- Consultor contable, especializado en tributaria. Solicita creación de sistema para ofrecer solución a empresa mayorista de alimentos.

**Necesita:**

- Poder visualizar rápidamente la información.
- Mejorar el proceso de venta web.
- Simplificar coordinación de envíos.
- Optimizar acceso a la información, acceso remoto.
- Contar con la opinión de los usuarios.
- Sistema de alertas (pedidos, stock).
- Valuación de stock, sistema por lotes.

## Premisas para la construcción del sistema:

- Debe ser **simple e intuitivo**.
- Desarrollo con **tecnologías de software libre**.
- Marco de trabajo: adaptación de la **metodología Scrum**.
- Se ocupará de los procesos de venta web, análisis y organización mediante la **automatización de tareas**.
- Constará de **una aplicación web y una app móvil**.
- Se integrará con **sistema de información geográfica**.

Resumiendo, se pretende **optimizar el tiempo de trabajo** mediante la automatización de tareas, minimizar costos y **mejorar la calidad de la información** para la toma de decisiones.

## Principales tecnologías empleadas en el desarrollo del proyecto:

- Java 
- JavaScript 
- HTML 
- GeoJSON 

**ORT** | FACULTAD DE INGENIERÍA  
ING. BERNARD WAND-POLAK

## Estado actual del proyecto.

- Se cumplieron los objetivos planteados.
- Se logró una primera versión de un sistema que automatiza tareas y optimiza la exposición de información. Integrado por:
  - Web de acceso público.
  - Web acceso restringido a empleados.
  - App móvil para hacer seguimiento del estado de los pedidos.

**ORT** | FACULTAD DE INGENIERÍA  
ING. BERNARD WAND-POLAK

## Proceso de elaboración aplicado.

- Planteamiento del problema.
- Representación de los datos.
- Diseñar un algoritmo que de solución al problema.
- Optimizar algoritmos.
- Codificar.
- Depurar y refactorizar.
- Documentar.

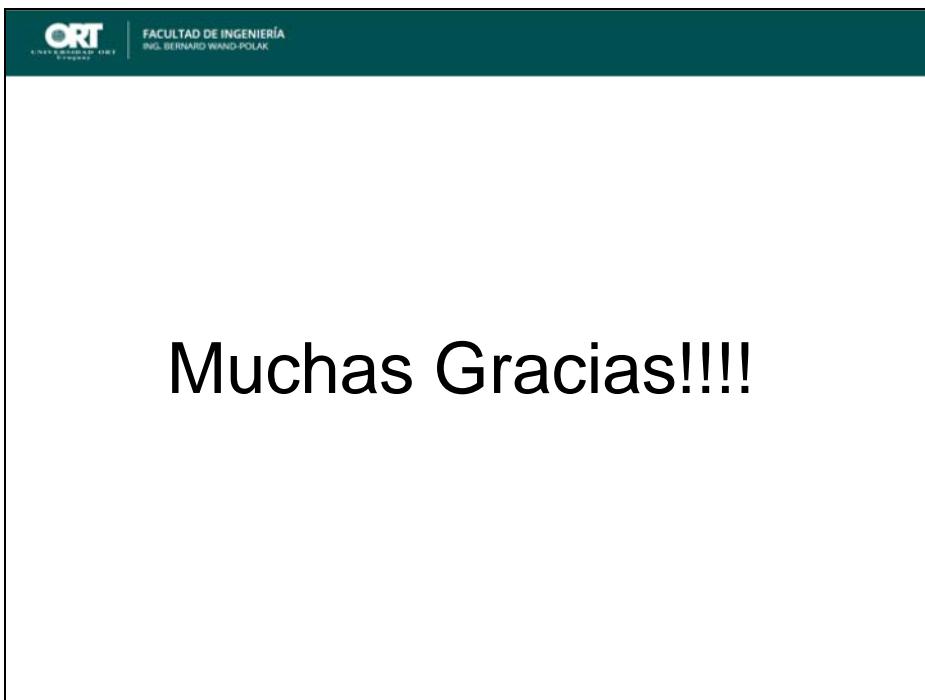


Ilustración 58. Presentación.

## **7. Glosario.**

- Ajax, técnica para crear aplicaciones interactivas para desarrollo web, permite ejecución asíncrona en JavaScript.
- Gson, biblioteca de código abierto en Java que permite la serialización entre objetos y la información se representa en formato JSON.
- JSON, formato de texto ligero para intercambio de datos.
- GeoJSON, formato abierto para representar elementos geográficos sencillos, codifica estructuras de datos geográficos.
- Back-end, capa en la cual corren las tecnologías del lado del servidor.
- Front-end, capa de presentación en la cual corren tecnologías del lado del cliente, del lado del navegador web.