

**INSTITUTO PROFESIONAL CIISA**

**INGENIERÍA EN INFORMÁTICA**

**DELYAPP**

**Un *software* de apoyo a la administración de local de comida rápida**

Seminario para optar al Título Profesional de Ingeniero en Informática

**Profesor Guía**

Patricio Sorich Huerta

**Profesor Revisor**

Carlos Duque

**Alumnos**

Nicolás González Araneda Gabriel Aravena Rivero

Ángel Bravo Enrique

**Santiago**

**Diciembre, 2020**

**DECLARACIÓN JURADA**

Yo, Nicolás Alejandro González Araneda, RUN 17.098.191-K, estudiante del 8º semestre de la carrera de Ingeniería en Informática, en mi calidad de Jefe del Proyecto Delyapp, que será desarrollado durante el segundo semestre del año 2020, junto con el señor Gabriel Aravena Rivero, RUT Nº 18.529.486-2 y el señor Ángel Alexis Bravo Enrique, RUT Nº 25.458.439-8 declaro bajo juramento, que el mencionado proyecto no ha sido ni se encuentra en desarrollo y tampoco ha sido implementado en organización alguna, por lo tanto, este será llevado a cabo y cumplirá todo el ciclo de procesos que fueron fijados por el Instituto Profesional Ciisa y que constituyen obligatoriedad para cumplir con el examen de titulación.

Nicolás Alejandro González Araneda

17.098.191-K

Ángel Alexis Bravo Enrique Gabriel Alejandro Aravena Rivero

25.458.439-8 18.529.486-2

**DEDICATORIA**

Quisiera dedicar esta tesis a mis cercanos, principalmente a mi polola Carolina Ovalle quién sin su apoyo en momentos complicados, no podría haber llegado hasta acá. Fue un gran pilar durante este grandioso periodo de aprendizaje.

Igualmente, dedico esta memoria a mi familia que indudablemente me brindó su apoyo en todo momento y me dio un ambiente grato y de confianza mientras estudiaba incluso en momentos de convivencia.

Finalmente debo agradecer a mis colegas y en especial a uno que me alentó en sacar esta interesante carrera y me mostró un mundo de la informática que quedé encantado.

Muchas gracias don Michael Silva por ser sumamente bondadoso por compartir conocimientos relevantes de la materia, un excelente mentor y gran sensei en los sistemas de información, Me apoyó incluso en momentos de mucho trabajo para él, a la hora que fuese y además aportó herramientas tecnológicas a disposición de este proyecto sin fines de lucro.

**Nicolás Alejandro González Araneda**

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

**Gabriel Aravena**

Esta tesis va con dedicación a mis familiares, en especial a mis abuelos que nunca se cansaron de alentarme para que siguiera con mis estudios profesionales, gracias a ellos que son mi fuente de inspiración, por ellos llegué hasta aquí.

También se lo dedico a mi madre, tía, tío y hermanos que me apoyaron en los momentos más difíciles para poder seguir pedaleando en este largo camino.

Se lo dedico también a mis amigos que me facilitaron las herramientas y los conocimientos cuando los necesitaba. También agradecido por que estuvieron en las buenas y en las malas, gracias por ello.

**Ángel Alexis Bravo Enrique**

**AGRADECIMIENTOS**

Al haber concluido con el proyecto, a nombre del grupo DELYAPP quisiéramos agradecer a todos aquellos que estuvieron junto a nosotros, nuestros docentes que fueron de vital importancia en momentos de pleno desarrollo especialmente los de estos dos últimos semestres en la construcción total de la documentación y del producto. Les agradecemos su confianza y buena disposición durante esta grandiosa ruta de aprendizaje.

Obviamente no podemos dejar de agradecer al personal administrativo de CIISA, cada uno de ellos cumplen una labor importante que nos pudieron haber ayudado en cualquier trámite o necesidad en su momento.

Finalmente agradecemos a nuestras familias que, definitivamente, son quienes nos motivaron de diferentes modos estos cuatro años de carrera.

**Equipo Delyapp**

**ÍNDICE**

[SIGLAS Y ABREVIATURAS 9](#_Toc59892818)

[RESUMEN 10](#_Toc59892819)

[INTRODUCCIÓN 11](#_Toc59892820)

[FASE DE INICIO 13](#_Toc59892821)

[2.1. Mercado Objetivo 14](#_Toc59892822)

[2.2. Necesidad 15](#_Toc59892823)

[2.3. Solución 15](#_Toc59892824)

[FASE DE ESTRUCTURACIÓN 17](#_Toc59892825)

[3.1. Objetivo General 18](#_Toc59892826)

[3.2. Justificación del proyecto 18](#_Toc59892827)

[3.3. Integrantes del equipo de proyecto 19](#_Toc59892828)

[3.4. Alcances 20](#_Toc59892829)

[3.4.1. Del proyecto 20](#_Toc59892830)

[3.4.1.1. Exclusiones 20](#_Toc59892831)

[3.4.2. Del producto 21](#_Toc59892832)

[3.5. Metodología 21](#_Toc59892833)

[**3.5.1 Herramientas en Scrum** 22](#_Toc59892834)

[**3.5.2 Ciclo de vida del *Software*** 23](#_Toc59892835)

[**3.5.3** **Roles** 25](#_Toc59892836)

[3.6. Investigación inicial 25](#_Toc59892837)

[**3.6.1.** **Aspecto legal** 25](#_Toc59892838)

[**3.6.2.** **Organizacional** 25](#_Toc59892839)

[**3.6.3.** **Técnica** 26](#_Toc59892840)

[**3.6.4.** **Metodología** 26](#_Toc59892841)

[**3.6.5.** **Finanzas** 26](#_Toc59892842)

[3.7. Estructura de Desglose de Trabajo 26](#_Toc59892843)

[3.8. Administración de Riesgos 27](#_Toc59892844)

[**3.8.1.** **Identificación de riesgos** 27](#_Toc59892845)

[**3.8.2.** **Categorización de riegos** 27](#_Toc59892846)

[**3.8.3.** **Medición y evaluación de riesgos** 28](#_Toc59892847)

[**3.8.4.** **Plan de actividades** 29](#_Toc59892848)

[**3.8.5.** **Seguimiento de riesgos** 31](#_Toc59892849)

[**3.9.** **Base Tecnológica** 31](#_Toc59892850)

[FASE DE CONSTRUCCIÓN 33](#_Toc59892851)

[4.1. Análisis 34](#_Toc59892852)

[4.2. Diseño y arquitectura 38](#_Toc59892853)

[4.3 Plan de pruebas 38](#_Toc59892854)

[4.4 Construcción 39](#_Toc59892855)

[4.5 Ejecución del Plan de Pruebas 40](#_Toc59892856)

[CONCLUSIONES 42](#_Toc59892857)

[GLOSARIO 44](#_Toc59892858)

[BIBLIOGRAFÍA 47](#_Toc59892859)

[ANEXOS 49](#_Toc59892860)

[DOCUMENTO DE INICIO 50](#_Toc59892861)

[DOCUMENTO DE MARCO METODOLOGICO 56](#_Toc59892862)

[DOCUMENTO DE INVESTIGACION INICIAL 71](#_Toc59892863)

[DOCUMENTO DE GOBIERNO DEL PROYECTO 83](#_Toc59892864)

[DOCUMENTO PLAN DE RIESGO 89](#_Toc59892865)

[DOCUMENTO DE DISEÑO Y ARQUITECTURA 111](#_Toc59892866)

[DOCUMENTO DE PLAN DE PRUEBAS 131](#_Toc59892867)

SIGLAS Y ABREVIATURAS

* **PC:** *Personal Computer* (Computador personal).
* **UML:** *Unified Modeling Language* (Lenguaje Unificado de Modelado).
* **UX:** *User Experience* (Experiencia de usuario).
* **DB:** *Data Base* (Base de Datos).
* **PMBOK:** *Project Management Body of Knowledge* (Cuerpo de Conocimientos de la Gestión de Proyectos).
* **PMP:** Precio medio ponderado.
* **HH:** Horas hombre.
* **WWW:** *World Wide Web*, (Red informática mundial), sistema lógico de acceso y búsqueda de la información disponible en Internet, cuyas unidades informativas son las páginas web.
* **Https:** *Hypertext Transfer Protocol Secure* “Protocolo seguro de transferencia de hipertexto”.
* **IEEE830:** Estándar para la especificación de los requerimientos del *Software*.
* **SW:** *Software*.
* **Pyme:** Pequeña y Mediana Empresa.
* **GUI:** Interfaz gráfica de usuario.
* **MSP:** Proveedor de servicios gestionados (MSP, por sus siglas en inglés).
* **IPv4:** El Protocolo de Internet versión 4 (en [inglés](https://es.wikipedia.org/wiki/Idioma_ingl%C3%A9s), *Internet Protocol version* 4, IPv4), es la cuarta versión del [Internet *Protocol*](https://es.wikipedia.org/wiki/Internet_Protocol) (IP), un protocolo de interconexión de redes basados en [Internet](https://es.wikipedia.org/wiki/Internet).
* **IPv6:** El IPv6 es una actualización al [protocolo](https://es.wikipedia.org/wiki/Familia_de_protocolos_de_internet) [IPv4](https://es.wikipedia.org/wiki/IPv4), diseñado para resolver el problema de [agotamiento de direcciones](https://es.wikipedia.org/wiki/Agotamiento_de_las_direcciones_IPv4).
* **SSD:** La unidad de estado sólido, disco de estado sólido o SSD (acrónimo inglés de *solid-state drive*) es un tipo de [dispositivo de almacenamiento de datos](https://es.wikipedia.org/wiki/Dispositivo_de_almacenamiento_de_datos) que utiliza [memoria no volátil](https://es.wikipedia.org/wiki/Memoria_no_vol%C3%A1til), como la [memoria flash](https://es.wikipedia.org/wiki/Memoria_flash), para almacenar datos, en lugar de los [platos o discos](https://es.wikipedia.org/wiki/Plato_(disco_duro)) [magnéticos](https://es.wikipedia.org/wiki/Disco_magn%C3%A9tico) de las [unidades de discos duros](https://es.wikipedia.org/wiki/Unidad_de_disco_duro) (HDD) convencionales.
* **UPS:** Sistemas de alimentación ininterrumpida (SAI), en inglés *uninterruptible power supply* (UPS), es un [dispositivo](https://es.wikipedia.org/wiki/M%C3%A1quina) que gracias a sus [baterías](https://es.wikipedia.org/wiki/Bater%C3%ADa_el%C3%A9ctrica) y otros elementos almacenadores de energía, durante un [apagón eléctrico](https://es.wikipedia.org/wiki/Apag%C3%B3n_el%C3%A9ctrico) puede proporcionar [energía eléctrica](https://es.wikipedia.org/wiki/Energ%C3%ADa_el%C3%A9ctrica) por un tiempo limitado a todos los dispositivos que tenga conectados. Otra función que se puede añadir a estos equipos es mejorar la calidad de la energía eléctrica que llega a las cargas, filtrando subidas y bajadas de tensión y eliminando armónicos de la red en caso de usar [corriente alterna](https://es.wikipedia.org/wiki/Corriente_alterna).
* **SSL :** Seguridad de la capa de transporte (en inglés: *Transport Layer Security* o TLS) y su antecesor *Secure* *Sockets* *Layer* (SSL; en español capa de puertos seguros) son [protocolos criptográficos](https://es.wikipedia.org/wiki/Protocolo_criptogr%C3%A1fico), que proporcionan comunicaciones [seguras](https://es.wikipedia.org/wiki/Seguridad) por una [red](https://es.wikipedia.org/wiki/Red_de_computadoras), comúnmente [Internet](https://es.wikipedia.org/wiki/Internet).
* **OSI:** El modelo de interconexión de sistemas abiertos (ISO/IEC 7498-1), más conocido como “modelo OSI”, (en inglés, *Open Systems Interconnection*) es un [modelo de referencia](https://es.wikipedia.org/w/index.php?title=Modelo_de_referencia&action=edit&redlink=1) para los [protocolos de la red](https://es.wikipedia.org/wiki/Protocolo_de_comunicaciones) (no es una [arquitectura de red](https://es.wikipedia.org/wiki/Arquitectura_de_red)).

RESUMEN

La elaboración del documento que se presenta es el proyecto Delyapp. El punto es concebir la percepción o la idea hasta definir la realización de esta. En el mismo contexto, se ajusta en el desarrollo sobre una estructura de trabajo para cumplir con un producto de *software* que cumpla con los estándares del cliente y el equipo.

En la actualidad muchas Pymes no conocen qué es sumarse a la era digital. La gestión de venta es un proceso que, generalmente, se lleva de forma análoga, con lápiz y papel o en una planilla. Este proceso tradicional no es el óptimo para llevar un control de inventario transparente y conciso, que se pueda consultar en cualquier momento para extraer información importante. Al no tener un control de entrada y salida de los productos que se usan y los gastos adicionales (luz, arriendo, sueldos, patente, etc.) que se hacen en una Pyme, pueden generar problemas futuros en el cálculo de margen de ganancias.

Particularmente en las pequeñas empresas dedicadas al rubro de la comida rápida, la situación no es muy distinta. En muchas ocasiones es posible observar que no existe un control sobre los gastos y las ganancias, y toda la gestión se lleva a cabo de forma prácticamente intuitiva, sin tener claridad de cómo evoluciona el negocio, de cuánto se debe cobrar por los productos que se ofrecen o de cómo se cubrirán los gastos asociados al mismo.

Por lo tanto, un apoyo tecnológico bajo esta mirada se hace una labor para nuestro equipo desarrollador con el fin de causar un impacto positivo en el emprendimiento de una Pyme.

**CAPITULO 1**

# INTRODUCCIÓN

El ejercicio consiste en plantear la importancia de tomar decisiones al momento de armar un producto en un local comercial de comida rápida y, hoy más que nunca, las aplicaciones de comida rápida como: UberEat, o Rappi, destacan por su popularidad en el servicio que entregan.

De este modo, cada día existen nuevos locales que se suman a las aplicaciones mencionadas, su propósito es vender de forma on-line, con pago automático y *delivery* (entrega a domicilio).

Estos locales de comida al momento de calcular sus productos recurren a planillas Excel o, derechamente, al lápiz y al papel, siempre con la idea de ofrecer un producto vendible, que sea rentable, para luego pagar los ingredientes de su inventario y gastos fijos.

Otro punto importante, a tener en cuenta, es considerar la merma que existe al armar productos, tales como pizza, completos y todo lo demás. La merma no es considerada o considerada en menor grado, y esto impacta también en el precio de venta.

Teniendo presente lo anterior, este proyecto busca concurrir desde una postura tecnológica, teniendo a disposición herramientas que le permitan y apoyen en el proceso del negocio, en este caso un local de comida rápida, con gran eje en la utilización de los alimentos que se arman con ingredientes que pueden ser calculados.

**CAPITULO 2**

# FASE DE INICIO

En este capítulo se contextualiza el proyecto y se conoce al cliente que cumple un papel de vital importancia a la hora de comenzar, ya que permite ahondar en las circunstancias en las que se desenvuelve su negocio, conocer sus problemas y, por lo tanto, como resultado entregar una propuesta de solución a dicha necesidad.

## 2.1. Mercado Objetivo

El *software* está orientado como servicio a las Pymes de comida rápida que no cuenten con servicios tecnológicos, que los ayuden a llevar a cabo la gestión de ventas e inventario, es decir, principalmente pequeños negocios que están iniciando en el rubro y no tienen claridad respecto a cómo controlar sus ingresos.

Es por esto que contamos con un cliente, que será nuestro prototipo para construir el *software* en base a sus necesidades generales, y de esta forma abordar las necesidades del rubro.

**La Chabelita:**

La señora Chabela es dueña de un local de comida rápida ubicado en la comuna de Pudahuel. Se dedica a la venta de alimentos como: Papas fritas, completos, churrascos, salchipapas, chorrillanas y bebidas.

Su dueña, la señora Chabela es la administradora del local, ella denomina su negocio como “emprendimiento familiar” y le ayudan algunos integrantes en su labor.

El local La Chabelita no posee redes sociales ni *software* de apoyo en su local.

.

## 2.2. Necesidad

Muchas Pymes actualmente no tienen un orden en la gestión de ventas, llevan los registros de sus ingresos y salidas de productos manualmente, o simplemente no los llevan. Además, un gran número de ellas tienen el problema de poder calcular el precio final de su producto elaborado, para tener un margen de ganancia que permita sustentar el negocio en el tiempo.

Es necesario que sea multiplataforma o acceder desde cualquier dispositivo ya sea: *Smartphone*, tablet, PC, etc. Con una interfaz (GUI) intuitiva y responsiva, es decir, que se ajuste a cualquier pantalla, de acuerdo a las peticiones según UX (experiencia de usuario).

## 2.3. Solución

Analizar y resolver el problema de manipulación de inventario con técnicas financieras que permita tener control y transparencia en cuanto a los ingresos y compras del local comercial en los ingredientes, mermas y gastos fijos. Además, la intención utilizada en estas técnicas, es reforzar a las Pymes con el cálculo del margen de ganancia y el precio final sugerido de cada producto.

Toda esta ayuda estará implementada dentro del *software* donde, también se contará con un sistema de venta y pago en línea (o presencial), incluyendo el despacho a domicilio (*delivery*) o retiro en local.

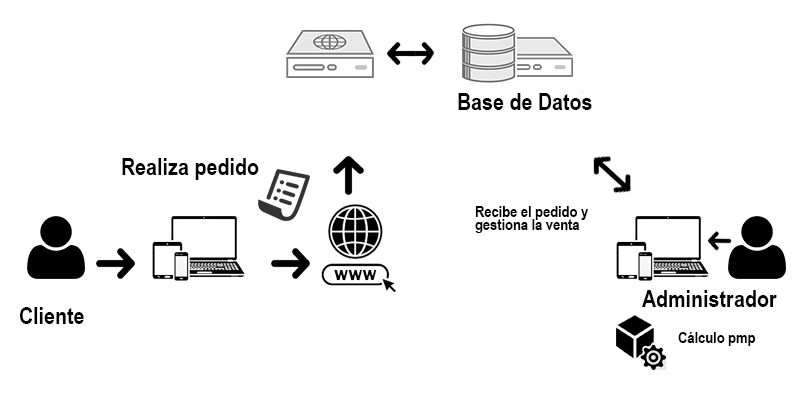


Diagrama Nº 01 – Solución

Finalmente, con el propósito de facilitar la creación de un producto, se permitirá registrar ingredientes que se utilizan frecuentemente para elaborar dichos productos que luego serán comprados por un cliente, se incluye la venta presencial con medios de pagos online y efectivo.

**CAPITULO 3**

# FASE DE ESTRUCTURACIÓN

El siguiente capítulo tiene la finalidad de sentar las bases para el desarrollo del proyecto, definiendo de forma clara cómo se aborda el problema. Definir junto al cliente cuáles son precisamente las funcionalidades que debe tener el *software* y cómo serán implementadas. Para más detalles consultar el anexo 8.1 “**DOCUMENTO DE INICIO”.**

## 3.1. Objetivo General

Diseñar, desarrollar e implementar un *software* con tecnología web que permita apoyar la gestión, control y seguimiento de las ventas e inventario de pequeños negocios dedicados a la venta de comida rápida. El *software* se presentará el 18/12/2020 con un costo no superior a 920 HH (horas hombre).

## 3.2. Justificación del proyecto

Con la construcción de este *software* se apoyará a la gestión de ventas de las Pymes, con un manejo transparente de los productos adquiridos por la misma.

Llevar los registros dentro del sistema serán de gran beneficio pues permitirá llevar una mejor gestión en la Pyme, con esto llevamos los registros manuales o las planillas al *software*.

El *SW* permitirá obtener balances y reportes solicitados. En base a esto se podrá hacer un estudio de mejora para la Pyme.

En síntesis, se espera mejorar el flujo de trabajo en las ventas y en las compras. Lo que significaría un beneficio en la forma en que se registran estas acciones, dejando atrás los registros manuales o en plantilla.

* Contribuirá a la venta automatizada de la Pyme.
* Mayor control sobre las compras, mermas e inventario.
* Proveerá un cálculo interno para las compras que ayudan a establecer precio en base a los gastos generales de un período de tiempo.
* Proveerá un cálculo sugerido aplicado a los ingredientes.
* Aportará información de despacho ya sea *delivery* o retiro de producto en local.

## 3.3. Integrantes del equipo de proyecto

El equipo “Delyapp”, está compuesto por los siguientes colaboradores:

|  |  |
| --- | --- |
| **Integrante** | **Cargo** |
| Chabela | *Product Owner* |
| Nicolás González | *Scrum Master* |
| Gabriel Aravena | *Scrum Team* |
| Ángel Bravo | *Scrum Team* |

Tabla Nº 01 – Integrantes del equipo

La descripción detallada de roles, junto a la asignación de las tareas de cada integrante del equipo, serán explicadas en el anexo 8.2 “**MARCO METODOLÓGICO”.**

## 3.4. Alcances

Determinar de forma clara y concreta los objetivos que se intentarán lograr a lo largo del desarrollo del proyecto, lo que conllevará a un cumplimiento exitoso.

Se definen los alcances del proyecto y producto de *software* comprometidos a continuación:

### 3.4.1. Del proyecto

Contemplar todos los objetivos y exclusiones que se llevarán a cabo durante el desarrollo del proyecto.

Los alcances son:

* + Documentación del proyecto.
  + *Software* funcional.
  + Planificación del proyecto.
  + Criterios de aceptación.
  + Validación de Funcionalidades.

#### 3.4.1.1. Exclusiones

Durante el análisis y planificación del proyecto se definieron las siguientes exclusiones:

* No se contempla la confección de manuales del producto.
* No se contempla el registro mediante cuentas de redes sociales.
* No se contempla la planificación y/o ejecución de actividades de soporte del producto.

### 3.4.2. Del producto

Los entregables del producto son los siguientes:

* Inicio de sesión con roles de administrador y cliente.
* Compra en línea en modo ambiente de integración.
* Pago presencial.
* Administración de Inventario.
* Cálculo automático del margen de ganancia para el precio final.
* Cálculo de merma de un producto.
* Publicar imágenes de los productos por parte del administrador.
* *Delivery* para el local que posea.

## 3.5. Metodología

En este punto se indicarán las metodologías utilizadas durante la ejecución del proyecto, tanto para la gestión de este, como para la construcción del producto de *software*.

El equipo optó por la metodología basada en los principios ágiles *SCRUM* para realizar actividades, tareas, procesos de forma rápida y adaptable.

No cubre el ciclo de vida del *software* completamente, tan sólo, cubre las cinco fases que son: Requisitos, diseño, codificación, pruebas y despliegue. Se mencionan los aspectos más relevantes que se realizan a través del *software* JIRA. Para más detalles consultar el anexo 8.2 “**MARCO METODOLÓGICO”.**

**3.5.1 Herramientas en Scrum**

Bajo la metodología Scrum se cubrió ciertas partes del ciclo de vida del *software* con herramientas propias en la metodología SCRUM y las que restan, fueron adicionadas al marco y se explican en el punto 3.2 en mezcla de metodologías en el anexo 8.2 “**DOCUMENTO DE MARCO METODOLOGICO**” al igual que las herramientas que se nombran:

* Captura de requerimientos (*History User*)
* Criterios de aceptación cliente
* *Backlog* del producto (*Sprint Backlog*)
* *Backlog* del *Sprint*
* Reunión diaria
* Revisión del *sprint*
* Retrospección del *sprint*

Se complementaron herramientas como:

* UML Lenguaje unificado de modelamiento
* Formulario estándar para pruebas
* Formulario estándar para pruebas ejecutadas
* Formulario Validación y cierre de producto

En el siguiente diagrama se muestra el concepto del trabajo en Scrum.

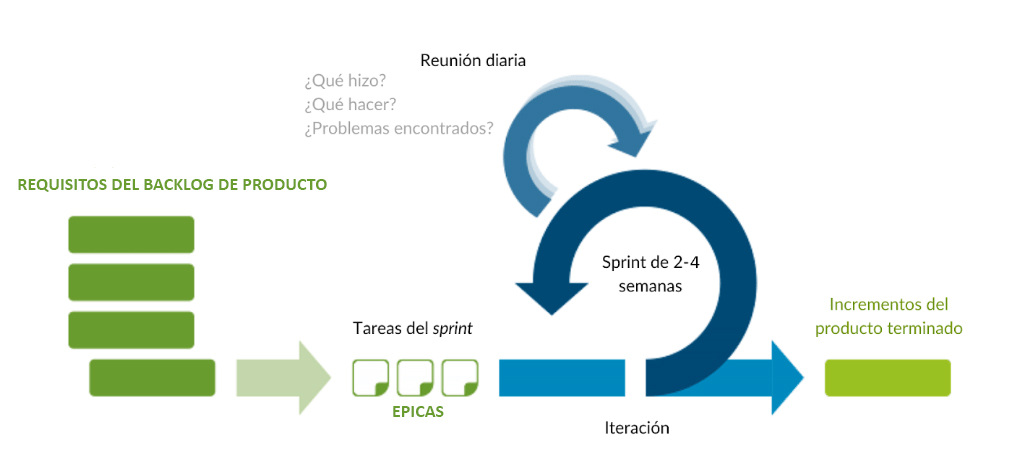


Diagrama Nº 02 – Diagrama de trabajo en Scrum

Respecto al cronograma general, se confeccionó una carta Gantt para la cubrir todo el proyecto y producto. Se usó el *software* Project de la suite de Office para este fin.

**3.5.2 Ciclo de vida del *Software***

Fue cubierto por las herramientas antes mencionadas, además, los aspectos de cada etapa fueron medidos por indicadores de criticidad y complejidad clasificados por niveles. Según las cinco fases se detalla cada una:

Requisitos:

Encuesta

* Historia de usuario
* Backlog del producto
* Backlog del *sprint*
* Casos de uso
* Especificación de casos de uso (Formulario IEEE830)
* Revisión del *sprint*

Diseño:

* Historia de usuario
* Épica
* Modelo relacional de base de datos
* Diagrama de clases
* Modelo de datos
* Diccionario de datos
* Diagramas de procesos
* Prototipos / Mockups

Codificación:

* Historia de usuario
* Modelo de procesos y especificación de historias de usuario
* Modelo de datos
* Diccionario de datos
* Reuniones diarias

Pruebas:

* Historias de usuario a casos de prueba
* Aplica formularios de plan y ejecución de pruebas
* Criterios de aceptación

Despliegue:

* Aplica formulario “Validación y cierre de producto”
* Aplicar versión de control del código fuente actualizado
* Instalación en modo producción

Mantención:

* Aplica formulario “Formulario de Aseguramiento de Calidad”
* Aplica formulario “Formulario de cambios y mantención”
  + 1. **Roles**

A continuación, se nombran los roles involucrados en el proyecto:

* Dueño de producto (*Product* *owner*): Chabela (La Chabelita).
* Equipo de desarrollo (*Scrum* *team*): Ángel Bravo y Gabriel Aravena.
* Facilitador de proyectos (*Scrum* *master*): Nicolás González.
* Interesados (*Stakeholders*): Patricio Sorich.

Para más información puede consultar el anexo 8.2 “**MARCO METODOLÓGICO”.**

## 3.6. Investigación inicial

Se han investigado varios aspectos que conciernen al proyecto, con la finalidad de obtener información y discernir cuál es la mejor opción a utilizar. A continuación, se mencionan los aspectos más relevantes. Para más detalles consultar el anexo 8.3 **“INVESTIGACIÓN INICIAL”**.

* + 1. **Aspecto legal**
* LEY N° 19.496 sobre la protección de los derechos de los consumidores.
* LEY N° 19.628 sobre protección de la vida privada.
* LEY N°21.096 consagra el derecho a protección de los datos personales.

Se ha determinado en base a la información, la necesidad de implementar medidas en la seguridad del *software* con la finalidad de que se adapten a las leyes mencionadas.

* + 1. **Organizacional**

Se ha detectado un impacto positivo en el ámbito económico y tecnológico de los negocios que utilicen nuestra aplicación.

Por otra parte, es posible detectar un posible impacto negativo en el ámbito estratégico de estos negocios, puesto que podrían tener que modificar sus métodos de trabajo para adaptarse al *software*.

* + 1. **Técnica**

Luego de la investigación se ha determinado utilizar el *framework* Laravel 7 para el desarrollo del *software*, el motor de base de datos MySql para el almacenamiento de datos.

* + 1. **Metodología**

Se ha decidido utilizar la metodología Scrum para el desarrollo del proyecto, basados en las características del equipo y las del *software*. Utilizando las herramientas que se consideran en la metodología y adicionando y adicionando otras para abordar completamente el ciclo de vida del *software*.

* + 1. **Finanzas**

Se decidió utilizar el método PMP (precio medio ponderado) para el cálculo del margen de ganancia, ya que considera los gastos fijos unitarios, los gastos variables del negocio y el margen de ganancia esperado.

## 3.7. Estructura de Desglose de Trabajo

Con el propósito de llevar un control sobre nuestro proyecto de utilizó una carta Gantt confeccionada con la herramienta Project de Office. Esto permitió tener visión del avance y *status* de las fases y actividades del proyecto.

Se presenta a continuación un resumen de la carta Gantt. Para más detalles consultar el anexo **“CARTA GANTT”.**

## 3.8. Administración de Riesgos

* + 1. **Identificación de riesgos**

Para identificar los riesgos se ha analizado toda la documentación del proyecto, principalmente el documento de inicio y el documento de investigación, identificando las vulnerabilidades y las posibles amenazas a las que está expuesto el proyecto. Luego del análisis se ha realizado una verificación incluyendo a todo el equipo y consultando a expertos, con la finalidad de reevaluar la veracidad de los riesgos y su posible impacto en el proyecto. Para más detalles consultar el anexo 8.5 **“PLAN DE GESTIÓN DE RIESGOS”.**

* + 1. **Categorización de riegos**

Con la finalidad de establecer con mayor claridad en qué ámbitos del proyecto impacta cada uno de los riesgos identificados, se han clasificado en los siguientes grupos:

* Riesgos del proyecto: amenazan la planificación del proyecto, en términos de tiempo, coste (esfuerzo) o calidad del proyecto. Tienen un impacto en el calendario definido y en los recursos del proyecto.
* Riesgos del producto: amenazan la calidad y la planificación temporal del *software* que hay que producir. Pueden ocasionarse porque el problema a resolver es más difícil de lo que se pensaba. Si un riesgo técnico se convierte en realidad, la implementación puede ser difícil o incluso imposible.

En la **Tabla No 2** se muestran los riesgos identificados y la categoría a la que pertenece.

|  |  |
| --- | --- |
| **Riesgo** | **Categoría** |
| 1. Daño en las dependencias de desarrollo (incendio, derrumbe, catástrofe natural). | Proyecto |
| 1. Cambio de forma inesperada en las normas legales. | Proyecto |
| 1. El alcance de aplicación está mal definido. | Proyecto |
| 1. Los cambios no son controlados y el alcance está en continuo crecimiento. | Proyecto |
| 1. Poco compromiso de parte del cliente. | Proyecto |
| 1. El producto no cumple con los requisitos. | Proyecto |
| 1. Bajo compromiso por parte del equipo de desarrollo. | Proyecto |
| 1. Falta de un miembro del equipo de desarrollo por motivos de fuerza mayor. | Proyecto |
| 1. Cambio en las condiciones del país por razones de fuerza mayor. | Proyecto |
| 1. Desconocimiento de las tecnologías a utilizar por parte del equipo de desarrollo. | Tecnológico |
| 1. Los servicios externos a contratar (ej: *hosting*) no cumplen con las características necesarias. | Tecnológico |

Tabla Nº 02 – Riesgos categorizados

* + 1. **Medición y evaluación de riesgos**

Para medir los riesgos de forma cuantitativa, se ha determinado evaluar dos aspectos, la probabilidad y el nivel de impacto. Para ambas mediciones se asignó un nivel de 1 a 5. De esta forma fue posible determinar un nivel de exposición de cada riesgo según muestra la **Tabla No 3**.

|  |  |  |  |
| --- | --- | --- | --- |
| **Nivel de exposición (Probabilidad + Impacto)** | | | |
| Extremo | Suma entre 9 y 10 | Alto | Suma entre 7 y 8 |
| Moderado | Suma entre 5 y 6 | Bajo | Suma entre 2 y 4 |

Tabla Nº 03 – Medición del nivel de exposición

* + 1. **Plan de actividades**

El plan de actividades tiene la finalidad de establecer procedimientos que permitan gestionar los riesgos, es decir, acciones que permitan evitar los riesgos o disminuir su impacto en los casos en que no se puedan evitar.

En la **Tabla No 4** se muestran los riesgos identificados, con su nivel de exposición y las actividades establecidas.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Riesgo | Exposición | Actividades de Evitación | Actividades de Mitigación | Actividades de Contingencia |
| 1. El alcance de aplicación está mal definido | Extremo | Realizar una reunión con el cliente y validar el alcance definido. | Realizar una reunión con el cliente y modificar el alcance. | Realizar una reunión con el cliente y negociar los alcances. |
| 1. El producto no cumple con los requisitos. | Alto | Realizar una correcta toma de requerimientos y aclarar con el cliente las funcionalidades a abordar. | Negociar con el cliente los requerimientos más críticos para ser implementados. | Realizar una nueva toma de requisitos considerando los errores cometidos. |
| 1. Poco compromiso de parte del cliente. | Alto | Mantener una constante comunicación, informando los avances y cambios del proyecto. | Exigir la participación activa del cliente indicando la relevancia que tiene en el proyecto. | Considerar los requerimientos fijados en el proyecto, traspasando el riesgo al cliente. |
| 1. Desconocimiento de las tecnologías a utilizar por parte del equipo de desarrollo. | Alto | Utilizar herramientas que el equipo conozca y haya utilizado con anterioridad. | Estudiar la tecnología e investigar sus actualizaciones. | Cambiar rápidamente de herramienta a una que sea de conocimiento del equipo. |
| 1. Daño en las dependencias de desarrollo (incendio, derrumbe, catástrofe natural). | Moderado | Realizar el proyecto en un lugar apto, manteniendo estándares mínimos de seguridad. | Realizar capacitación respecto a las normas de seguridad y acciones en caso de emergencia. | Migrar a un ambiente de desarrollo online. |
| 1. Bajo compromiso por parte del equipo de desarrollo. | Moderado | Contar con un equipo de confianza con una buena relación interpersonal. | Exigir el cumplimiento de las tareas asignadas. | Desvincular a la persona implicada, buscando un reemplazo en el menor tiempo posible. |
| 1. Los cambios no son controlados y el alcance está en continuo crecimiento. | Moderado | Establecer que las modificaciones de gran volumen deben considerarse para otro proyecto. | No aceptar cambios que modifiquen considerablemente nuestro alcance. | Replanificar el proyecto con las nuevas modificaciones y respetando a fecha final. |
| 1. Falta de un miembro del equipo de desarrollo por motivos de fuerza mayor. | Moderado | Contar con un equipo confiable, y con una alta motivación.  Mantener cuidados de salud. | Contar con un registro de posibles reemplazantes en caso de requerirlo. | Dividir las tareas entre los participantes mientras no se cuente con un reemplazo. |
| 1. Cambio en las condiciones del país por razones de fuerza mayor. | Moderado | Tener un estilo de trabajo de baja dependencia de la situación nacional. Fomentando el trabajo a distancia. | Cambiar a un estilo de trabajo completamente online. | Aceptar el riesgo, comunicando al cliente los posibles retrasos en el proyecto. |
| 1. Los servicios externos a contratar (ej: *hosting*) no cumplen con las características necesarias. | Moderado | Realizar una investigación sobre la calidad del servicio a contratar. | Cambiar el servicio contratado por uno que cumpla las expectativas. | Acordar con el cliente un servicio con menores prestaciones. |
| 1. Cambio de forma inesperada en las normas legales. | Bajo | Revisar constantemente los proyectos de ley y posibles cambios en la normativa legal que nos afecta. | Modificar los alcances del proyecto, redefiniendo los requerimientos afectados. | Acordar con el cliente las funcionalidades que se pueden prescindir. |

Tabla Nº 04 – Plan de actividades

* + 1. **Seguimiento de riesgos**

Para asegurar un correcto seguimiento de los riesgos, se ha definido un encargado, quien deberá realizar las siguientes acciones tras cada *sprint*:

* Chequear el estado de los riesgos presentes o residuales.
* Monitorear el comportamiento de los nuevos riesgos identificados.
* Implementar los planes de respuesta a los nuevos riesgos.
* Evaluar la efectividad del proceso de gestión de los riesgos y realizar los cambios pertinentes.
  1. **Base Tecnológica**

En la **Tabla No 5** se nombran las tecnologías que se utilizaron en el desarrollo del producto a lo largo del proyecto.

|  |  |
| --- | --- |
| **Sistema operativo** | **Ubuntu server** |
| Lenguaje programación | PHP |
| Editor de código fuente | Visual Studio Code |
| Control de versiones | GitHub |
| *Framework* | Laravel 7 |
| Motor de base de datos | MySql |
| Librería | jQuery librería de JavaScript |
| Servidor | Lemp |

Tabla Nº 05 – Base tecnológica

**CAPITULO 4**

# FASE DE CONSTRUCCIÓN

Abarcar las principales actividades relacionadas a los requisitos reunidos en cada entrevista u otra estrategia usada que pronto son candidatos a funcionalidades que de forma incremental completando el producto, el *software* en sí.

## 4.1. Análisis

Se abarcaron las principales actividades relacionadas a los requisitos y los candidatos a funcionalidades del *software* que son las épicas dentro de cada *sprint*.Las estrategias, actividades y herramientas definidas en el marco metodológico, se aplicaron en estas actividades.Es importante tener la visión general del avance a través de la planificación por medio de una carta Gantt asociados a esta etapa.Las historias de usuario expuestas fueron categorizadas para cada *sprint* que serán 6 *Sprint* de 3 semanas cada una, en el la **Tabla No 6**. Optamos como herramienta de toma de requisitos, una encuesta a través de un formulario que facilita la Suite de Google llamada Forms.Bajo el marco metodológico nos apoyamos en la herramienta de Jira para cubrir parte del ciclo de vida del producto. El proceso contempló historias de usuarios que se agrupan en épicas. En cada *sprint* 4 épicas que forman el producto. Estas épicas contienen historias de usuario que a su vez contemplan subtareas a desarrollar y entregar en cada *sprint*.Las actividades principales con las que contara el *software* y las que serán interactuadas por el usuario se resumen en el **Diagrama No 3.**

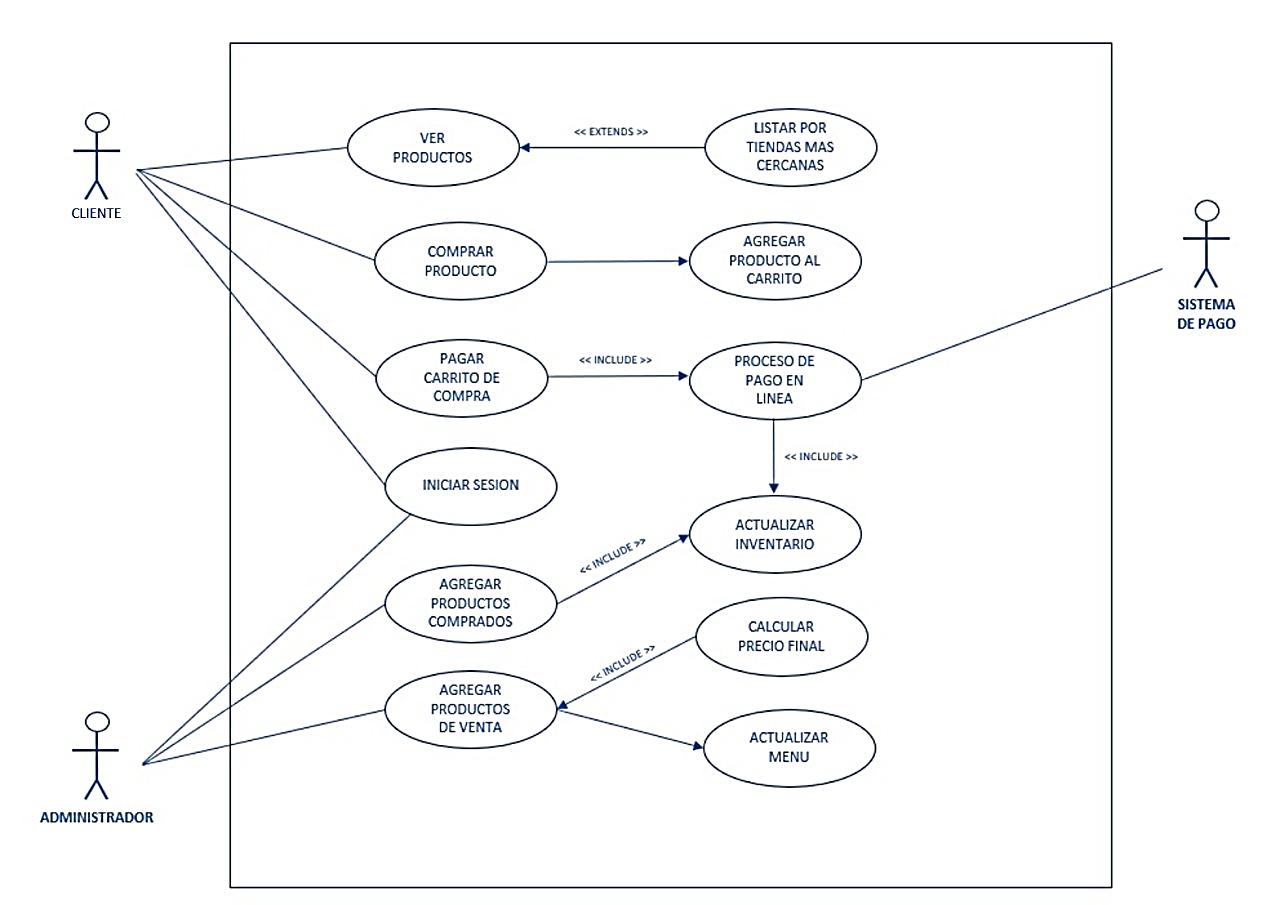
****

Diagrama Nº 03 – Casos de uso

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **EPICAS** | | | **HISTORIAS DE USUARIO** | | |
| **ID EPICA** | **TITULO** | **ID HISTORIA USUARIO** | **ROL** | **FUNCIONALIDAD** | **RAZON** |
| E1 | REALIZAR COMPRA COMO CLIENTE | H.U. 1 | Como Cliente | Quiero tener la opción de agregar productos a un carrito de compras | Para poder generar una compra. |
| H.U. 2 | Como Cliente | Quiero tener un buscador | Para poder tener un mejor filtro de lo que deseo encontrar. |
| H.U. 3 | Como Cliente | Quiero solicitar *delivery* | Para no tener que ir a buscar mi pedido al local. |
| H.U. 4 | Como Cliente | Quiero poder pagar a través de una tarjeta bancaria | Para hacerlo a través de internet, sin la necesidad de acercarme al local. |
| H.U. 5 | Como Cliente | Quiero recibir un correo de confirmación de mi pedido | Para asegurar que solicité la orden correctamente. |
| H.U. 6 | Como Cliente | Quiero poder visualizar el estado de mi pedido y el tiempo de despacho | Para poder organizarme con los tiempos de despacho o la demora que tomará el *delivery*. |
| E2 | GESTIONAR EL INVENTARIO DE LA TIENDA  (ADMINISTRADOR) | H.U. 7 | Como administrador | Quiero ingresar los ingredientes que ocupo en mi negocio, con su precio y las cantidades que tengo | Para tener un inventario |
| H.U. 8 | Como administrador | Quiero ingresar los productos que vendo, indicando la cantidad de ingredientes que ocupo en él | Para conocer la cantidad de ingredientes que gasto cada mes. |
| H.U. 9 | Como administrador | Quiero tener la opción de modificar el inventario | Para actualizar la cantidad de ingredientes cuando compre nuevamente. |
| H.U. 10 | Como administrador | Quiero tener la opción de auditar el inventario | Para saber cuáles han sido las pérdidas o mermas. |
| H.U. 11 | Como administrador | Quiero que al realizar una venta se actualice automáticamente el inventario | Para tener el inventario siempre actualizado. |
| H.U. 12 | Como administrador | Quiero poder visualizar la cantidad de ingredientes que tengo disponibles | Para saber cuándo es necesario comprar ingredientes nuevamente. |
| H.U. 13 | Como administrador | Quiero poder visualizar la cantidad de ingredientes que he gastado o perdido cada mes | Para saber cómo varían estos gastos. |
| E3 | ADMINISTRAR EL MENU DE LA TIENDA  (ADMINISTRADOR) | H.U. 14 | Como administrador | Quiero publicar los productos que tengo en venta | Para que los clientes puedan conocer mi menú y realizar compras. |
| H.U. 15 | Como administrador | Quiero que cuando publique un producto me sugieran un precio (no obligatorio) | Para poder tener un margen de ganancia que permita solventar mi negocio. |
| H.U. 16 | Como administrador | Quiero eliminar los productos del menú. | Para modificar el menú si es necesario. |
| H.U. 17 | Como administrador | Quiero modificar los productos que tengo en venta | Para poder cambiar los precios si es necesario. |
| E4 | GESTIONAR VENTAS (ADMINISTRADOR) | H.U. 18 | Como administrador | Quiero poder realizar una venta en mi local a través del *software* | Para que se registre automáticamente. |
| H.U. 19 | Como administrador | Quiero poder visualizar las ventas realizadas durante todos los meses | Para saber cómo evoluciona el negocio y tener claridad de cuánto se vende mensualmente. |
| H.U. 20 | Como administrador | Quiero visualizar cuánto dinero he gastado, cuánto ha ingresado y cuáles han sido las ganancias del negocio cada mes | Para saber si es necesario hacer modificaciones de precio y si las ganancias son las esperadas. |

Tabla Nº 06 – Historias de usuario

## 4.2. Diseño y arquitectura

Las bases de este proyecto fueron diseñadas a partir de los elementos definidos en la etapa de análisis. El concepto de diseño basadas en las historias de usuario, como también se describe el correspondiente diseño que tiene el *software* tanto en lo visual y técnicamente. Además, se realizó el diseño del diagrama de clases que muestra la interacción entre clases y métodos.

Por último, se detalla la arquitectura usada, contiene su parte física (*hardware*) e intangible (*software*) al respecto.

El detalle del documento diseño y arquitectura contemplado en el proyecto, se puede encontrar en el anexo 8.7 **“DOCUMENTO DE DISEÑO Y ARQUITECTURA”**.

## 4.3 Plan de pruebas

Se aplicará en cada *sprint* un plan de prueba que pueda involucrar una o más historias de usuario que componen los requisitos del *software*.

Se medirá con el caso de prueba que se definió con el resultado esperado para cada caso, esto permite descartar errores al momento de validar el *sprint*.

Al concluir con un *sprint*, debe cumplir con el 100% del objetivo en cada iteración, de lo contrario, se recurre a pasar lo pendiente con compromiso de arreglar lo pendiente más la funcionalidad del siguiente *sprint*. Para revisar el detalle de pruebas realizadas, se integra a este documento la ejecución de las mismas pruebas.

Se realizaron pruebas de concepto en la parte de pasarela de pago, por ejemplo, además la mayoría de las pruebas son de tipo caja negra se enfocan en una entrada 0 y salida 1 sin importar su funcionamiento interno.

En el anexo 8.8 “**DOCUMENTO DE PLAN DE PRUEBAS**” del presente documento, se detallarán los casos de prueba a realizar y los resultados que se esperan para que estos sean aprobados.

## 4.4 Construcción

Al haber terminado el análisis y la metodología, se acordó junto al equipo SCRUM realizar 6 *sprints* los cuales irán incrementando y sumando funcionalidades al producto.

Al comienzo de cada *sprint* se realizó un *sprint* *meeting* (reunión) de carácter organizativo en las tareas a desarrollar en cada etapa junto a los responsables.

Durante cada *sprint*, se realizaron reuniones diarias a nivel de equipo desarrollador para tener las actividades claras colaborando así en el avance de cada etapa.

Al final de cada iteración en un *sprint* se probó la pieza de *software* entre los integrantes el incremento de producto, también se registró cada caso de prueba que contiene una o muchas historias de usuario.

A continuación, se indican el objetivo de cada *sprint*:

* Primer *sprint*:
* Creación de mantenedores.
* Segundo *sprint*:
* Gestión de inventario.
* Tercer *sprint*:
* Administración del local.
* Cuarto *sprint*:
* Administración de ventas.
* Quinto *sprint*:
* Despliegue
* Sexto *sprint*:
* Producción

En el anexo 8.2 “**DOCUMENTO DE MARCO METODOLÓGICO**” se detallan las historias de usuario y los *sprints*.

## 4.5 Ejecución del Plan de Pruebas

Luego de que se definió el plan de pruebas, se desarrollaron los casos en el presente documento con el objetivo de completar el producto final con el 100% de las funcionalidades correctas detrás de cada caso de prueba.

Los criterios de aceptación en el conjunto de historias de usuario registradas por cada caso de prueba deben cumplirse según los criterios de calidad según el grupo Delyapp y del *Product-owner* (cliente). El detalle de la estrategia abordada y de los resultados de estas pruebas serán explicados en el anexo 8.9 “**DOCUMENTO DE EJECUCION DE PLAN DE PRUEBAS**”.

**CAPITULO 5**

# CONCLUSIONES

Como conclusión el equipo encargado del proyecto Delyapp ha cumplido con el objetivo de construir un *software* para apoyar en la gestión, administración y ventas de productos de pequeñas empresas (Pymes).

Esto ha sido posible gracias al conocimiento profesional de los integrantes sobre temas de análisis, solución, programación y bajo el compromiso de cada uno en largas investigaciones sobre temas contables con el fin de dar un valor agregado al producto.

**CAPITULO 6**

# GLOSARIO

En esta parte se definen las palabras y conceptos que pueden no ser conocidos por quienes revisen este documento, por lo que se hace una pequeña definición y su origen donde se obtuvo la misma.

***Scrum*:** Metodología ágil, aplicada de manera regular para trabajar colaborativamente en equipo.

**Alcance**: Es el total de los productos a desarrollar dentro del proyecto, representa usualmente la totalidad de trabajo necesario para terminar dicho proyecto.

***Software***: Se conoce como *software*​, *logicial* o soporte lógico al [sistema formal](https://es.wikipedia.org/wiki/Sistema_formal) de un [sistema informático](https://es.wikipedia.org/wiki/Sistema_inform%C3%A1tico), que comprende el conjunto de los componentes lógicos necesarios que hacen posible la realización de tareas específicas, en contraposición a los componentes físicos que son llamados [hardware](https://es.wikipedia.org/wiki/Hardware). La interacción entre el software y el hardware hace operativo un ordenador (u otro dispositivo), es decir, el software envía instrucciones que el hardware ejecuta, haciendo posible su funcionamiento.

***Hardware***: El *hardware*, equipo o soporte físico​ en informática se refiere a las partes físicas, tangibles, de un [sistema informático](https://es.wikipedia.org/wiki/Sistema_inform%C3%A1tico), sus componentes eléctricos, electrónicos, electromecánicos y mecánicos.​ Los cables, así como los gabinetes o cajas, los [periféricos](https://es.wikipedia.org/wiki/Perif%C3%A9rico_(inform%C3%A1tica)) de todo tipo, y cualquier otro elemento físico involucrado, componen el hardware o soporte físico; contrariamente, el soporte lógico e intangible es el llamado [software](https://es.wikipedia.org/wiki/Software).

**Base de Datos**: Conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso.

**Diagrama**: Dibujo geométrico que representa gráficamente una proposición a la resolución de un problema planteado.

***Framework***: Estructura conceptual y tecnológica de asistencia definida, normalmente, con artefactos o módulos concretos de *software*, que puede servir de base para la organización y desarrollo de *software*.

**Iteración**: Se refiere a repetir varias veces un proceso con la intención de alcanzar una meta deseada, objetivo o resultado.

**Lenguaje de programación**: Lenguaje formal que le proporciona a una persona, en este caso el programador, la capacidad de escribir una serie de instrucciones o secuencias de órdenes en forma de algoritmos con el fin de controlar el comportamiento físico y/o lógico de una computadora, de manera que se puedan obtener diversas clases de datos.

**Librería**: Es un conjunto de implementaciones funcionales, codificadas en un lenguaje de programación, que ofrece una interfaz bien definida para la funcionalidad que se invoca.

***Login***: Proceso mediante el cual se controla el acceso individual a un sistema informático mediante la identificación del usuario utilizando credenciales provistas por el usuario.

***Product Owner****:* Persona responsable de asegurar que el equipo aporte valor al negocio. Representa las partes interesadas internas y externas.

***Delivery*:** (reparto o entrega) es una actividad parte de la función [logística](https://es.wikipedia.org/wiki/Log%C3%ADstica) que tiene por finalidad colocar bienes, servicios, fondos o información directo en el lugar de consumo o uso (Al cliente final).

**Uber Eat:** Es una plataforma en línea de pedido de comida a domicilio, que fue creada por [Uber](https://es.wikipedia.org/wiki/Uber) en 2014 en [San Francisco](https://es.wikipedia.org/wiki/San_Francisco_(California)), [California](https://es.wikipedia.org/wiki/California).

**Rappi:** Las características definitorias de Rappi es la amplia gama de productos y servicios disponibles para la entrega mediante una app móvil.

**Interfaz:** Una interfaz (interfaces, en plural) se utiliza en [informática](https://es.wikipedia.org/wiki/Inform%C3%A1tica) para nombrar a la conexión funcional entre dos sistemas, programas, dispositivos o componentes de cualquier tipo, que proporciona una comunicación de distintos niveles, permitiendo el intercambio de información. Esto es un ejemplo de la realidad virtual.

**Responsiva:** El diseño web adaptable (también diseño web adaptativo o responsivo; este último [calco](https://es.wikipedia.org/wiki/Calco_sem%C3%A1ntico) del inglés responsive web design), es una filosofía de diseño y desarrollo cuyo objetivo es adaptar la apariencia de las [páginas web](https://es.wikipedia.org/wiki/P%C3%A1ginas_web) al dispositivo que se esté utilizando para visitarlas.

***Tablet:*** Una tableta , comúnmente abreviada como tableta , es un [dispositivo móvil](https://en.wikipedia.org/wiki/Mobile_device) , generalmente con un [sistema operativo móvil](https://en.wikipedia.org/wiki/Mobile_operating_system) y [circuitos de](https://en.wikipedia.org/wiki/Circuitry) procesamiento de pantalla [táctil](https://en.wikipedia.org/wiki/Touchscreen) , y una [batería recargable](https://en.wikipedia.org/wiki/Rechargeable_battery) en un paquete único, delgado y plano.

***Smartphone:*** El teléfono inteligente (del inglés smartphone) es un [dispositivo móvil](https://es.wikipedia.org/wiki/Dispositivo_m%C3%B3vil) que combina las funciones de un [teléfono celular](https://es.wikipedia.org/wiki/Tel%C3%A9fono_celular) y de una [computadora u ordenador de bolsillo](https://es.wikipedia.org/wiki/PDA). Estos dispositivos funcionan sobre una [plataforma informática móvil](https://es.wikipedia.org/wiki/Plataforma_(inform%C3%A1tica)), con mayor capacidad de almacenar y capaz de realizar tareas simultáneamente, tareas que realiza una [computadora](https://es.wikipedia.org/wiki/Computadora), y con una mayor conectividad que un teléfono convencional.

**Servidor:** Un servidor es un conjunto de computadoras capaz de atender las peticiones de un [cliente](https://es.wikipedia.org/wiki/Cliente_(inform%C3%A1tica)) y devolverle una respuesta en concordancia. Los servidores se pueden ejecutar en cualquier tipo de computadora, incluso en computadoras dedicadas a las cuales se les conoce individualmente como «el servidor». En la mayoría de los casos una misma computadora puede proveer múltiples servicios y tener varios servidores en funcionamiento. La ventaja de montar un servidor en computadoras dedicadas es la seguridad. Por esta razón la mayoría de los servidores son procesos diseñados de forma que puedan funcionar en computadoras de propósito específico.

***Switch*:** Conmutador (*switch o switches*) es el dispositivo digital lógico de interconexión de equipos que opera en la [capa de enlace de datos](https://es.wikipedia.org/wiki/Capa_de_enlace_de_datos) del [modelo OSI](https://es.wikipedia.org/wiki/Modelo_OSI). Su función es interconectar dos o más host de manera similar a los [puentes de red](https://es.wikipedia.org/wiki/Puente_de_red), pasando datos de un segmento a otro de acuerdo con la [dirección MAC](https://es.wikipedia.org/wiki/Direcci%C3%B3n_MAC) de destino de las [tramas](https://es.wikipedia.org/wiki/Trama_de_red) en la red y eliminan

***Rack:*** Es una interfaz modular entre [servidores](https://en.wikipedia.org/wiki/Web_server)[web](https://en.wikipedia.org/wiki/Web_application) y [aplicaciones web](https://en.wikipedia.org/wiki/Web_application) desarrolladas en el [lenguaje de programación Ruby](https://en.wikipedia.org/wiki/Ruby_programming_language) . Con Rack, [las interfaces de programación de aplicaciones](https://en.wikipedia.org/wiki/Application_programming_interface) (API) para *[frameworks](https://en.wikipedia.org/wiki/Web_framework" \o "Marco web)* [web](https://en.wikipedia.org/wiki/Web_framework" \o "Marco web) y [middleware](https://en.wikipedia.org/wiki/Middleware) están [envueltas](https://en.wikipedia.org/wiki/Wrapper_function) en una única [llamada de método que](https://en.wikipedia.org/wiki/Method_(computer_programming)) maneja [solicitudes](https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol#Request_message) y [respuestas](https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol#Response_message)[HTTP](https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol#Request_message) . do la conexión una vez finalizada esta.

***Firewall:*** En [informática](https://es.wikipedia.org/wiki/Inform%C3%A1tica), un cortafuegos (del término original en [inglés](https://es.wikipedia.org/wiki/Idioma_ingl%C3%A9s) *firewall*) es la parte de un [sistema informático](https://es.wikipedia.org/wiki/Sistema_inform%C3%A1tico) o una [red informática](https://es.wikipedia.org/wiki/Red_inform%C3%A1tica) que está diseñada para bloquear el acceso no autorizado, permitiendo al mismo tiempo comunicaciones autorizadas.[1](https://es.wikipedia.org/wiki/Cortafuegos_(inform%C3%A1tica)#cite_note-1)​

***Let’s Encrypt: Let’s Encrypt:*** Es una autoridad de certificación (AC, o CA por sus siglas en inglés) gratuita, automatizada, y abierta, que existe para el beneficio del público. Es un servicio provisto por el [*Internet Security* *Research Group* (ISRG)](https://abetterinternet.org/).

***Open source:*** El código abierto es un modelo de desarrollo de [software](https://es.wikipedia.org/wiki/Software_de_c%C3%B3digo_abierto) basado en la colaboración abierta.[1](https://es.wikipedia.org/wiki/C%C3%B3digo_abierto#cite_note-1)​ Se enfoca más en los beneficios prácticos (acceso al código fuente) que en cuestiones éticas o de libertad que tanto se destacan en el [software libre](https://es.wikipedia.org/wiki/Software_libre).[2](https://es.wikipedia.org/wiki/C%C3%B3digo_abierto#cite_note-2)​ Para muchos el término «libre» hace referencia al hecho de adquirir un software de manera gratuita.

***Framework:*** En el [desarrollo de software](https://es.wikipedia.org/wiki/Desarrollo_de_software), un entorno de trabajo es una estructura conceptual y tecnológica de asistencia definida, normalmente, con artefactos o módulos concretos de software, que puede servir de base para la organización y desarrollo de [*software*](https://es.wikipedia.org/wiki/Software). Típicamente, puede incluir soporte de [programas](https://es.wikipedia.org/wiki/Programa_(computaci%C3%B3n)), [bibliotecas](https://es.wikipedia.org/wiki/Biblioteca_(programaci%C3%B3n)), y un [lenguaje interpretado](https://es.wikipedia.org/wiki/Lenguaje_interpretado), entre otras herramientas, para así ayudar a desarrollar y unir los diferentes componentes de un proyecto.

Representa una [arquitectura de software](https://es.wikipedia.org/wiki/Arquitectura_de_software) que modela las relaciones generales de las entidades del dominio, y provee una estructura y una especial metodología de trabajo, la cual extiende o utiliza las aplicaciones del dominio.

**CAPITULO 7**

# BIBLIOGRAFÍA

1. Siglas y abreviaturas – WikiPedia (diciembre de 2020)

<https://es.wikipedia.org/>

1. *Framework* – Documentación de Laravel 7

<https://laravel.com/docs/7.x>

1. Transbank – Documentación de inicio

<https://www.transbankdevelopers.cl/documentacion/como_empezar>

1. Pago en línea – Integración de laravel con Transbank

<https://github.com/freshworkstudio/transbank-web-services>

1. Inventario – Precio medio ponderado (PMP)

<https://economipedia.com/definiciones/precio-medio-ponderado.html#:~:text=El%20precio%20medio%20ponderado%20(PMP,valoraci%C3%B3n%20de%20existencias%20muy%20com%C3%BAn>.

**CAPITULO 8**

# ANEXOS



**INSTITUTO PROFESIONAL CIISA**

**INGENIERÍA EN INFORMÁTICA**

## DOCUMENTO DE INICIO

**DELYAPP**

**Un *software* de apoyo a la administración de local de comida rápida**

**Profesor Revisor:**

Patricio Sorich Huerta

**Alumnos:**

Nicolás González Araneda

Gabriel Aravena Rivero

Ángel Bravo Enrique

**Santiago**

**Diciembre 2020**

INDICE

[1 Historial del Documento 52](#_Toc59230286)

[2 Introducción 53](#_Toc59230287)

[3 Cliente 53](#_Toc59230288)

[4 Solución Propuesta 53](#_Toc59230289)

[5 Objetivo General 54](#_Toc59230290)

[6 Justificación del Proyecto. 54](#_Toc59230291)

**Historial del Documento**

|  |  |  |  |
| --- | --- | --- | --- |
| Fecha | Versión Documento | Descripción | Responsable |
| 16/05/2020 | 1.0 | Inicio documento, Información proyecto, Introducción, Cliente. | Ángel Bravo |
| 19/05/2020 | 1.1 | Solución propuesta, Objetivo general, Justificación del Proyecto, modificación Cliente. | Nicolás González |
| 19/06/2020 | 1.2 | Eliminación Información proyecto. | Gabriel Aravena |
| 28/06/2020 | 1.3 | Cambió los párrafos: Cliente | Nicolás González /  Ángel  Bravo |
| 21/07/2020 | 1.4 | Mejora de solución propuesta | Ángel Bravo |

**Introducción**

La gestión de ventas es un proceso que generalmente se lleva de forma análoga, con lápiz y papel o una planilla, este proceso tradicional no es el más óptimo para llevar un historial de compra claro y conciso, esto se puede ver afectado en el cálculo del margen de ganancia al no tener controlado los gastos que se hacen en un negocio, en este contexto el *software* brindará una pequeña ayuda para todas estas tareas ya mencionadas.

Por lo general los locales de comida rápida no pueden pagar el arriendo de *software* por lo que prefieren llevar su actividad monetaria manualmente, pues este *software* estará al alcance monetario por el enfoque al contexto Pyme.

**Cliente**

El proyecto está orientado a un servicio. Se realizaron encuestas para enviar a 2 locales de este rubro para conocer el contexto del desarrollo del proyecto:

* Pinche Pollo: Pollo crispy con la mejor receta en su especialidad, papas rústicas, distintos tipos de salsas, se puede pedir o retirar en el local.
* La Chabelita ofrece churrascos, completos, italianos, papas fritas, salchipapas, chorrillanas, pollo al coñac.
* Los lunáticos por el bajón, local de comida rápida ubicado en Pudahuel, su dueño Juan Marín es administrador y ofrece papas fritas, bebidas, pollos asados, completos y churrascos.

**Solución Propuesta**

Se propone construir un *software* donde el cliente (usuario) se va a conectar mediante un dispositivo con conexión a internet, donde ingresará a la página web https://delyapp.cl y podrá visualizar los productos disponibles de los locales que incluye buscador por comidas o localización. El cliente podrá seleccionar el o los productos a pedir, también puede pagar lo solicitado en línea o cuando le llegue el producto además tiene la opción de solicitar en producto en modo *delivery* o puede recoger en el local.

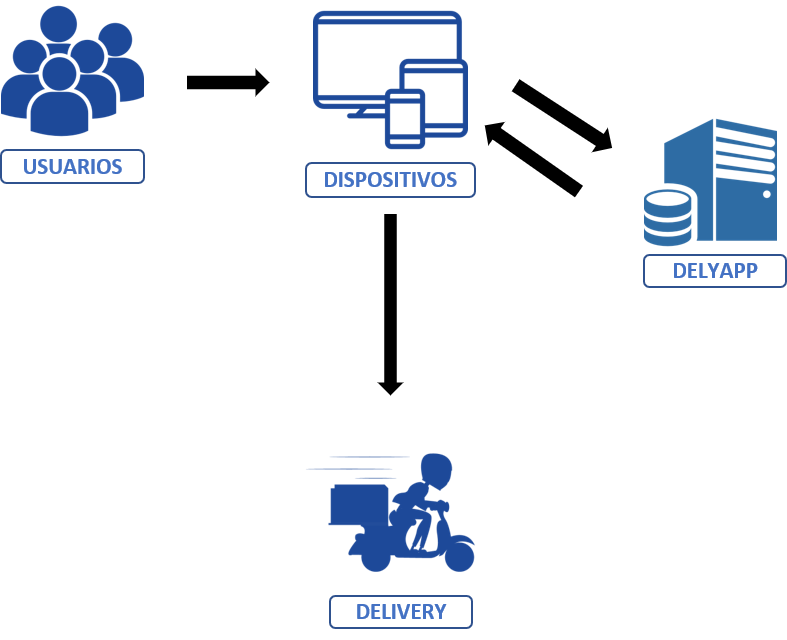


Diagrama N°4 – Solución propuesta

**Objetivo General**

Diseñar, desarrollar e implementar un *software* con tecnología web para apoyar la gestión de ventas, realizar reservas y servicio *delivery* (entrega a domicilio en un rango especifico), poniendo a disposición un servicio alojado bajo el concepto “*cloud computing*” o en la nube, que llegue el registro de la cada Pyme y las solicitudes de pedido para el administrador/a de cada comida rápida.

**Justificación del Proyecto**

Se espera mejorar el flujo de trabajo en las ventas y en las compras. ​Lo que significaría un gran beneficio en la forma en que se registran estas acciones, pues solucionaría la forma actual que se llevan los registros manualmente o en plantilla.

* Contribuye a la venta automatizada de la Pyme.
* Mayor control sobre las compras y proveedores (inventario).
* Proveerá un cálculo interno para las compras que ayudan a establecer precio en base a los gastos generales de un periodo de tiempo.



**INSTITUTO PROFESIONAL CIISA**

**INGENIERÍA EN INFORMÁTICA**

## DOCUMENTO DE MARCO METODOLOGICO

**DELYAPP**

**Un *software* de apoyo a la administración de local de comida rápida**

**Profesor Revisor:**

Patricio Sorich Huerta

**Alumnos:**

Nicolás González Araneda

Gabriel Aravena Rivero

Ángel Bravo Enrique

**Santiago**

**Diciembre 2020**

INDICE

[1 Historial del Documento](#_Toc59233062)

[2 Resumen del Proyecto](#_Toc59233063)

[3 Definición Metodológica](#_Toc59233064)

[3.1 ¿Por qué se optó por Scrum?](#_Toc59233065)

[3.2 Herramientas utilizadas](#_Toc59233066)

[3.3 Mezcla de metodologías](#_Toc59233067)

[4 Ciclo de vida del *software*](#_Toc59233068)

[5 Roles](#_Toc59233069)

[6 *Sprint* definidos](#_Toc59233070)

**Historial del Documento**

|  |  |  |  |
| --- | --- | --- | --- |
| Fecha | Versión Documento | Descripción | Responsable |
| 20-06-2020 | 1.0 | Inicio Plan de Gestión de Riesgos | Gabriel Aravena |
| 28-06-2020 | 1.1 | Modificación Plan de Actividades | Ángel Bravo |
| 21-07-2020 | 1.2 | Modificación todo el documento | Nicolás González |

Tabla N° 1 Historial de documento

**Resumen del Proyecto**

|  |
| --- |
| **Información del proyecto** |
| Nombre: Delyapp |
| Cliente: Chabela |
| Interesado: Patricio Sorich |
| Responsable proyecto: Nicolás González |
| Fecha de inicio: 31-marzo-2020 |
| Fecha fin: 18-diciembre-2020 |

Tabla N° 2 – Resumen de proyecto

DELYAPP es que es un proyecto enfocado al apoyo comercial y de inventario a las Pymes que abarca el sector de la comida rápida. Nació bajo la necesidad de clientes como la señora Chabela “La Chabelita” y el señor Juan Pérez “Pinche Pollo” de llevar control sobre sus compras y ventas, el servicio *delivery* también se incluyó.

Se propuso construir una aplicación web empleando elementos de scrum para dar solución al cliente según su problema o necesidad.

**Definición Metodológica**

De acuerdo a los objetivos del proyecto, se decidió utilizar Scrum como marco metodológico utilizando algunas herramientas del *software* Jira con el fin de cubrir el ciclo del *software* en las que se aplicaran 6 fases seleccionadas:

Requisitos, Diseño y codificación, pruebas, despliegue y mantención.

**¿Por qué se optó por Scrum?**

Porque permitió al equipo por medio de la metodología basada en los principios ágiles realizar actividades, tareas, procesos de forma rápida y adaptable.

El equipo de desarrollo se compuso de 3 integrantes, una cifra ideal para abarcar esta metodología. Además, el equipo contaba con experiencia previa en *Scrum*.

Para conceptos de colaboración y transparencia permitió al equipo estar al día en sus tareas. Este ambiente fue bien aceptado entre el equipo y se cumplen actividades como reuniones diarias de 15 minutos y planeación de inicio en cada *sprint*.

Respecto a la duración de cada *sprint* es de 3 semanas, definimos 6 *sprints* con 6 épicas que son las tareas más importantes del proyecto. Se estableció periodos cortos de tiempos para llevar un control preciso en el desarrollo del producto debido a que es un equipo pequeño.

**Herramientas utilizadas**

Respecto al cronograma general, se confeccionó una carta Gantt para la cubrir todo el proyecto y producto. Se usó el *software* *Project* de la suite de Office para este fin.

Bajo la metodología Scrum se cubrió ciertas partes del ciclo de vida del *software* con herramientas propias en la metodología SCRUM y las que restan, fueron adicionadas al marco y se explican el punto 3.2 en mezcla de metodologías.

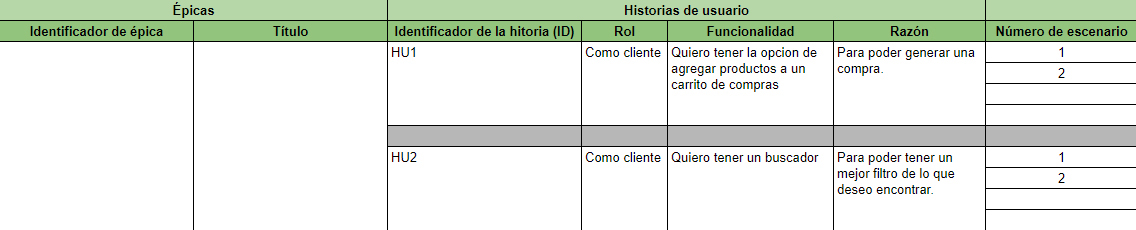
- Captura de requerimientos (*History User*): Este elemento contempla actividades que abarcan los requerimientos que fueron las funcionalidades del sistema y reglas del negocio, se llevó a cabo encuestas al cliente que darán respuesta a las tareas que se irán apilando, e irán a dar a la pila de tareas (*Sprint* Backlog).

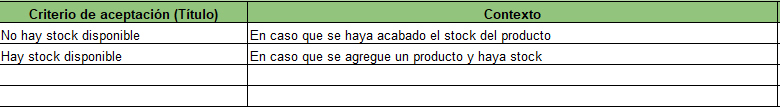
Las encuestas se realizaron por formularos de Google Forms, una herramienta de la suite de Google y en medida de lo posible, se entrevistó a personas dueños/as del local usando en adición video conferencia por Google Meet. Por lo general las preguntas son: Qué quiere, Como lo quiere y Para qué lo quiere.

- Épica: Grupo de tareas o actividades que forman parte de la confección de un elemento o componente a ser evaluado.

- Backlog del producto (*Sprint Backlog*): Es el conjunto de historias de usuario, básicamente es la mayor fuente de información para posteriormente realizar la planificación de los *sprints*.

Los indicadores del backlog contienen épicas que están compuestas por historias de usuario, identificador de la épica y título. Las historias de usuario contienen un identificador, rol, funcionalidad y una razón. Los criterios de aceptación están categorizadores por escenarios, contexto, evento y resultado.





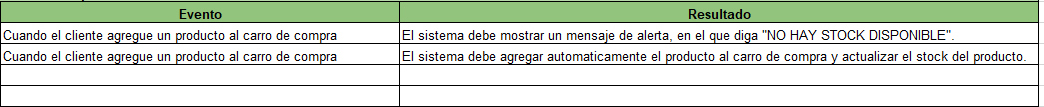
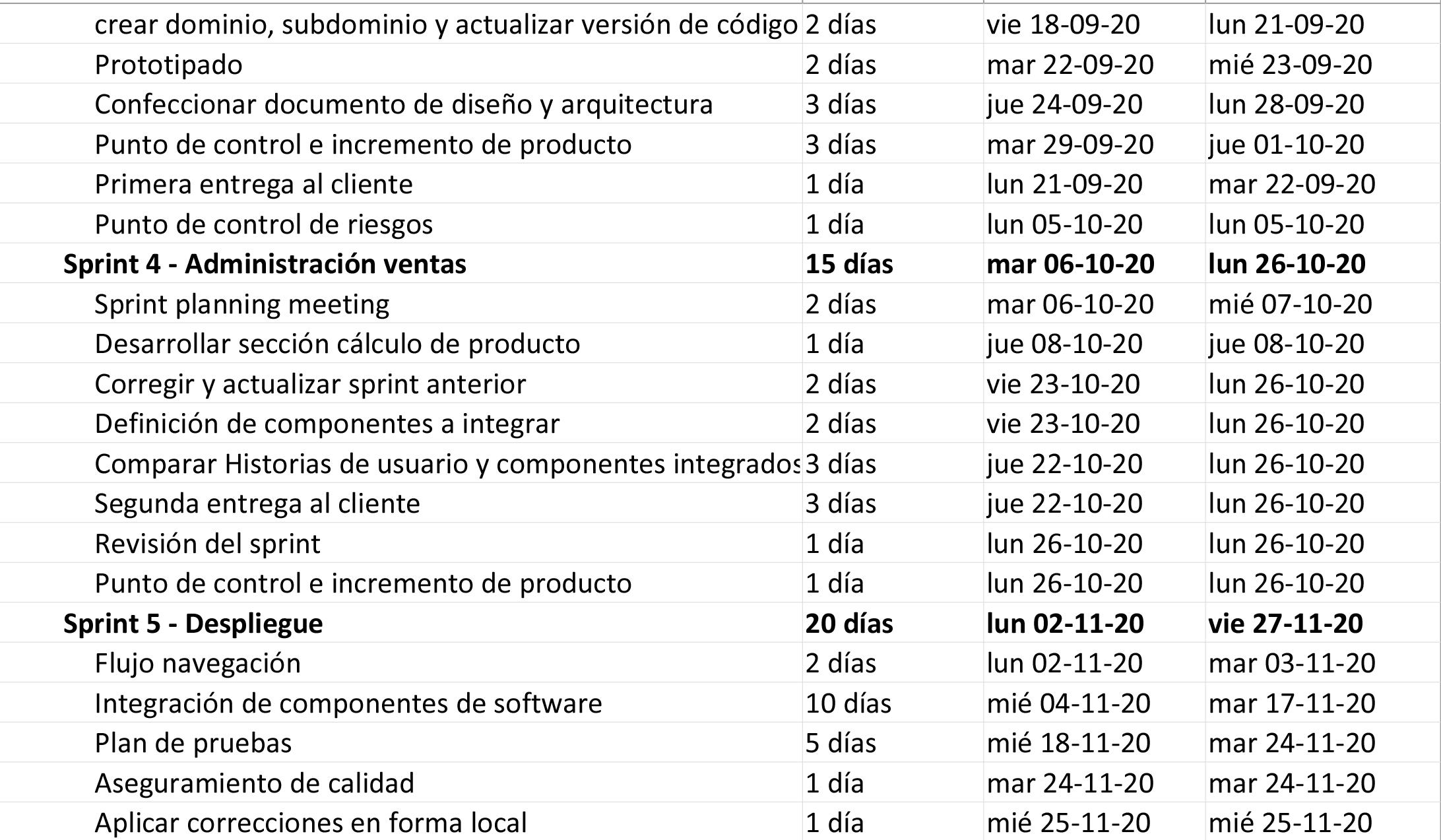
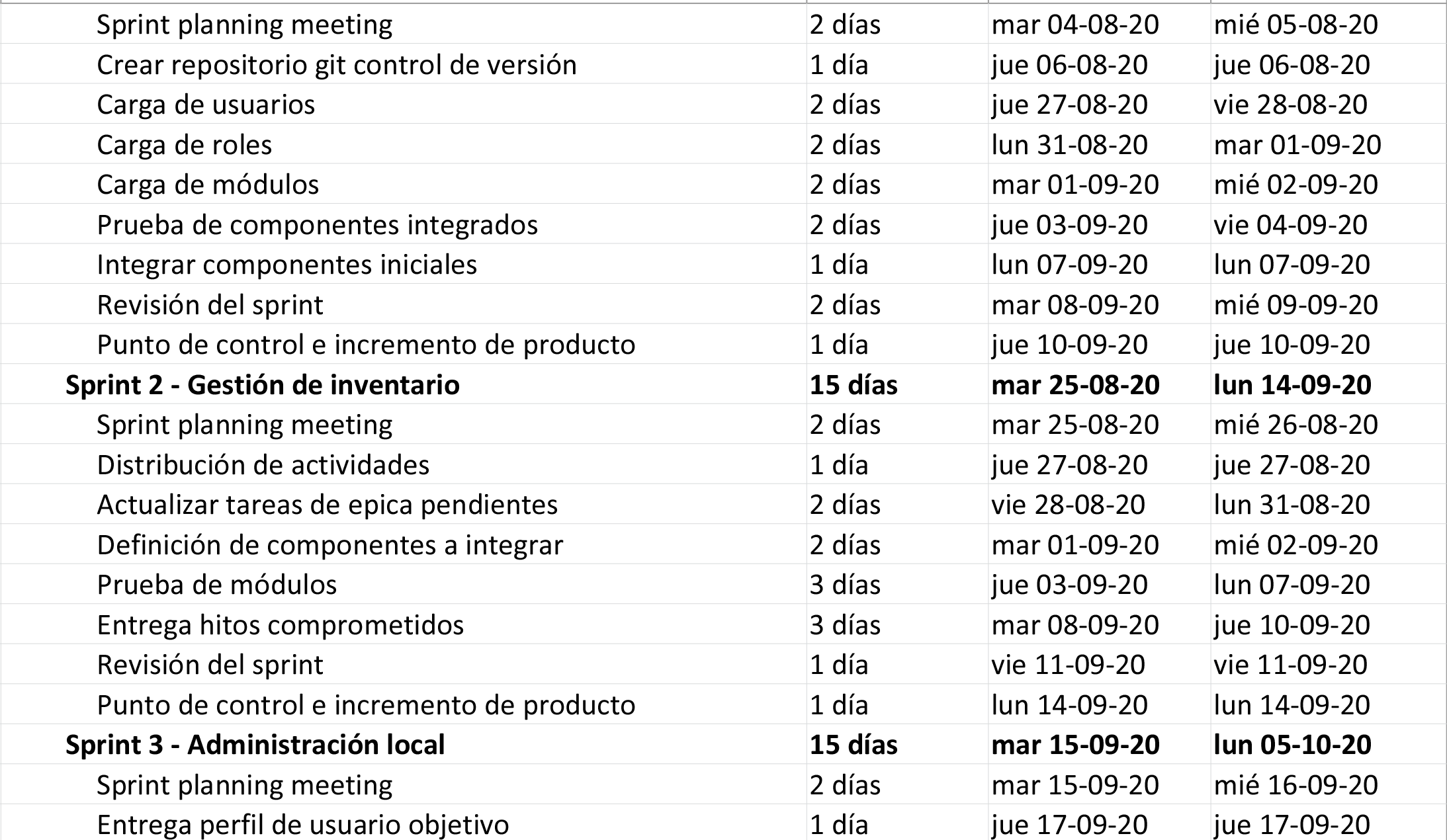
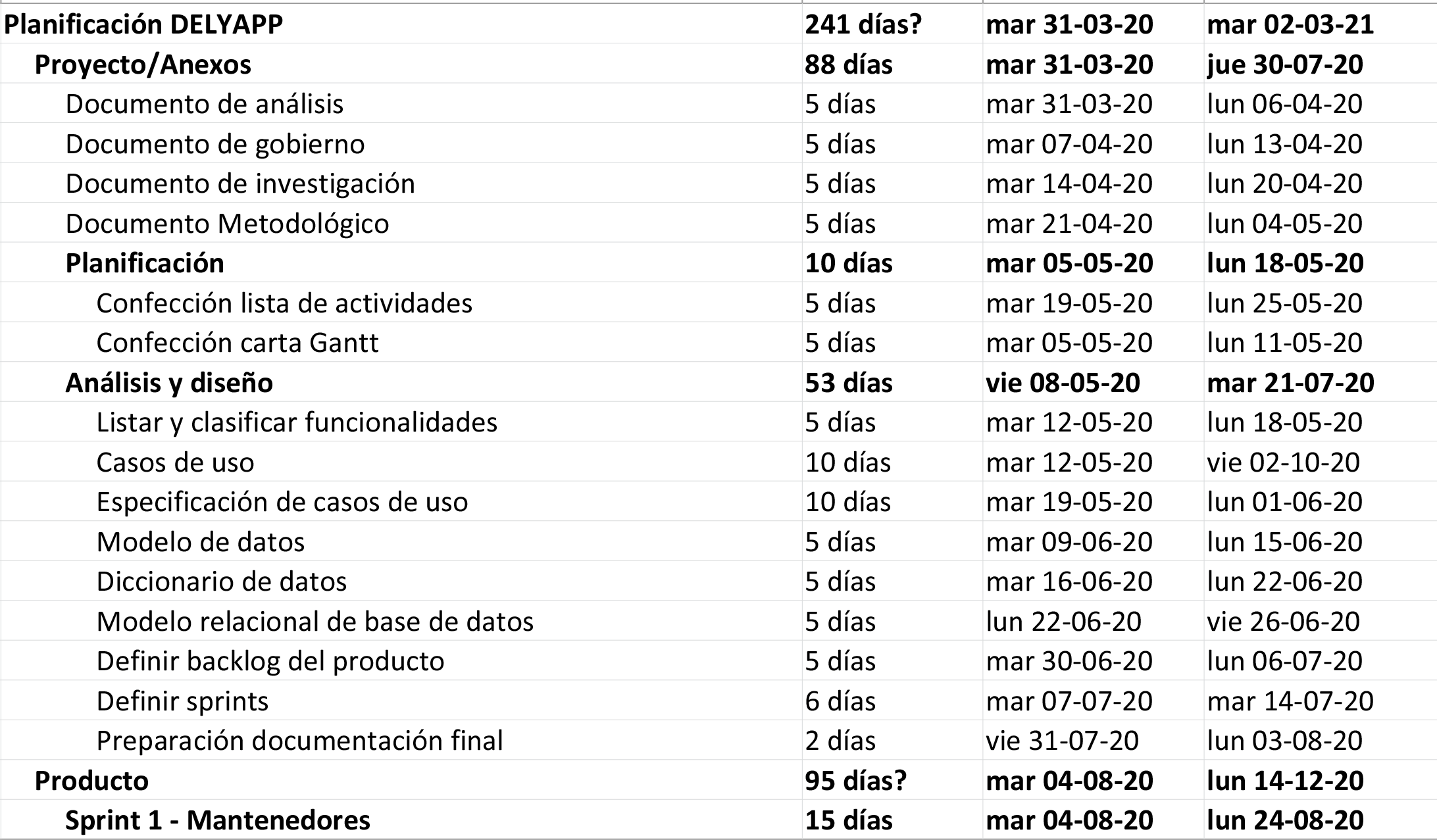


Tabla N° 3 – Formato de historias de usuario





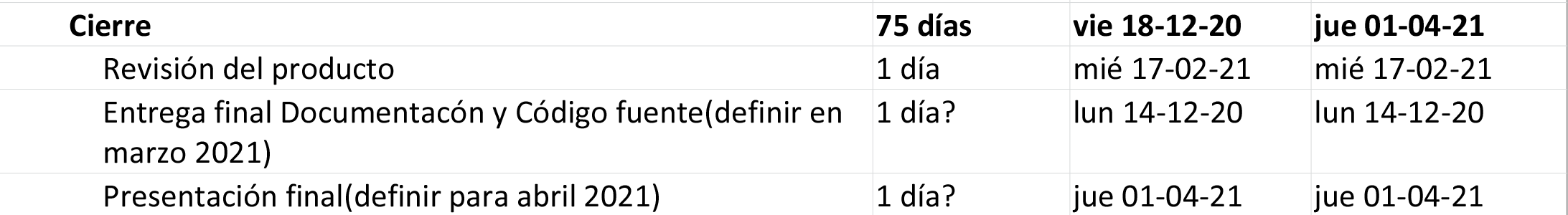


Tabla N° 4 - Carta Gantt

- *Backlog del Sprint*: Conjunto de elementos que son tareas técnicas que se trabajaron en el *sprint* que van a dar la pieza de *software* que se suma a otras en y se fue realizando el producto incremental, Tiene fecha de inicio y límite. Si la épica no se termina dentro del *sprint* planificado, se pasará al siguiente. Cada *sprint* tiene un rango de duración que fluctúa entre dos a cuatro semanas.

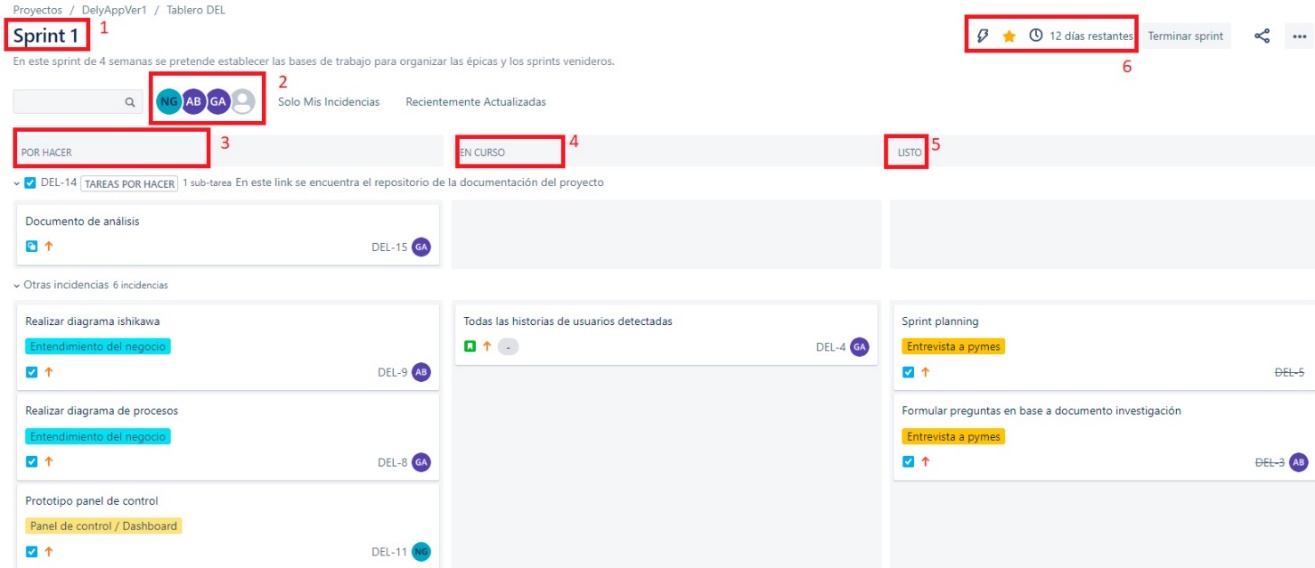


Diagrama N° 2 - *Sprint Backlog*

1. Indica el *sprint* actual
2. Son los responsables e informador de las actividades
3. Por hacer
4. En curso
5. Listo
6. Tiempo en que finaliza el *sprint*

- Reunión diaria: Se reúne el equipo para conversar de las actividades que se han realizado en una video conferencia por no más de 30 minutos. Además de las actividades se revisan aspectos de calidad con el fin de controlar y monitorear los avances y superar los obstáculos mientras se desarrolla en medio de la plática.

Revisión del *sprint*: El equipo desarrollador presenta al cliente los requisitos completados en la iteración, en forma de incremento de producto preparado para ser entregado. El cliente revisa el entregable y se adaptan las mejoras necesarias.

A continuación, se detalla la agenda:

**Agenda de revisión del *Sprint***

* Historias de usuario asociadas a la épica del *sprint*
* Demostración de pieza de *software* desarrollado
* Épicas que quedan pendientes (si es que las hay)

**Agenda de retrospección del *Sprint***

* ¿Qué aprendimos en este *sprint*?
* ¿Qué hicimos bien?
* ¿Qué pudimos mejorar? En torno a Personas, herramientas, procesos, relaciones
* Tomar acciones a realizar

- UML Lenguaje unificado de modelamiento: Es un lenguaje gráfico para visualizar,

graficar, especificar y construir documento de diseño del sistema. Se utilizaron

herramientas para los casos de uso y su especificación bajo el formato (Formulario

IEEE830)

-Formulario estándar para pruebas: Se expone formato con indicadores para la

realización del plan de pruebas. Figura del encabezado:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| N° de prueba | ID prueba | Funcionalidad | Dato de entrada | Salida esperada | Resultado | Estado de prueba |

Tabla N° 1 – Formato Plan de Pruebas

- Chequeo de aseguramiento de calidad: En este formato es donde se encuentran los criterios de aceptación definidos para comprometer la calidad del producto.

Este debe ser firmado por el *product owner* y el *stake holder* para validar la pieza

de código para el paso a producción. Se presenta el formato a continuación:

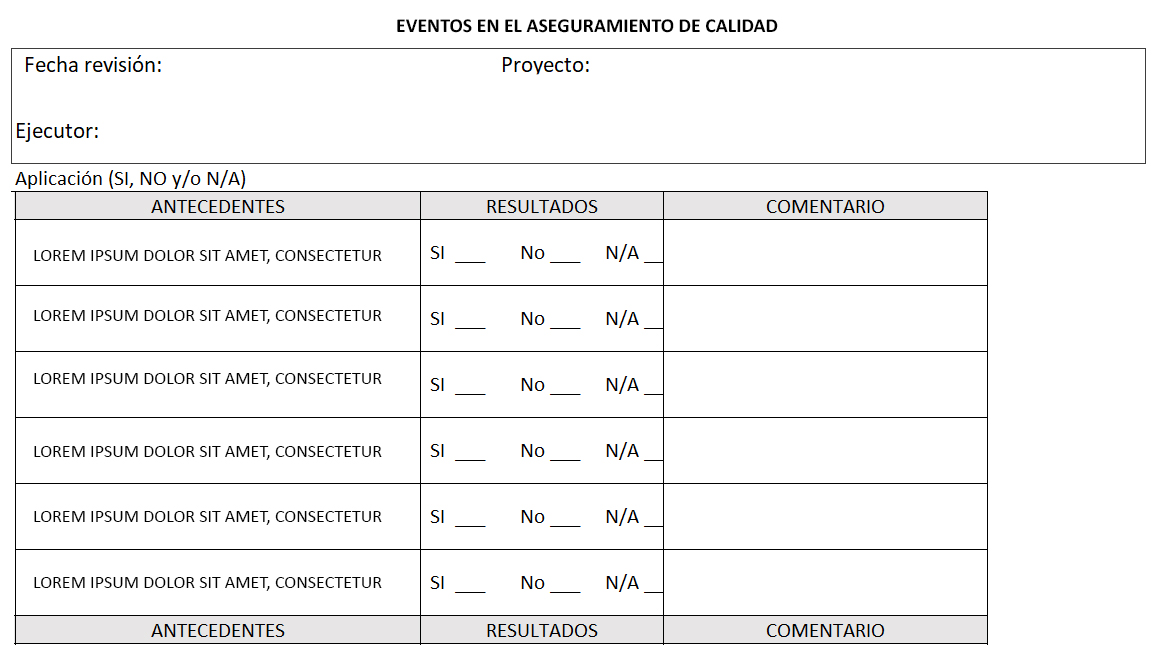


Diagrama N° 3 – Formulario de Aseguramiento de Calidad

* Formulario control de cambios y mantención: Este documento se dispone al

momento en que el cliente quiere un cambio sobre la marcha o para mantención

realizada al sistema.



Tabla N° 5 – Formulario de cambios y mantención

**Mezcla de metodologías**

Para el apoyo de metodología ágil se utilizaron herramientas del sistema de trabajo llamado *KANBAN* que van a sustentar o reforzar las tareas en *SCRUM*. Particularmente Jira apoya la organización de tareas y viene incluido el tablero *KANBAN*. Fue ideal para la coordinación en nuestro equipo de trabajo porque propone un sistema de uso colaborativo.

**Ciclo de vida del *software***

Fue cubierto por las herramientas antes mencionadas, además, los aspectos de cada etapa serán medidos por indicadores de criticidad y complejidad además fueron clasificados por niveles. Según las seis fases se detalla cada una:

• Requisitos:

- Encuesta

- Historia de usuario

- *Backlog* del producto

- *Backlog* del *sprint*

- Casos de uso

- Especificación de casos de uso (Formulario IEEE830)

-Revisión del *sprint*

• Diseño:

-Historia de usuario

-Épica

-Modelo relacional de base de datos

-Modelo de datos

-Diccionario de datos

-Diagramas de procesos

-Prototipos / *Mockups*

• Codificación:

-Historia de usuario

-Modelo de procesos y especificación de historias de usuario

-Modelo de datos

-Diccionario de datos

-Reuniones diarias

• Pruebas:

-Aplica formulario plan de pruebas

-Historias de usuario

• Despliegue:

- Aplica formulario aseguramiento calidad

-Documento diseño

-Instalación en local

• Mantención:

- Aplica formulario aseguramiento calidad

-Aplica formulario de cambio y mantención

**Roles**

A continuación, se describen los roles involucrados en el proyecto:

Dueño de producto (*Product owner*): Es una sola persona, quien determina las prioridades del proyecto. Representa a la empresa o a los usuarios.

Equipo de desarrollo (*Scrum team*): es el grupo de trabajo que llevará a la realidad el producto que necesita el propietario.

Facilitador de proyectos (*Scrum master*): gestiona las dinámicas del equipo de trabajo y le ayuda a llegar a la consecución de su objetivo.

Interesados (*Stakeholders*): son quienes tienen algún interés en el producto y observan su desarrollo, ya sea como clientes, patrocinadores, directivos de la compañía u otros actores externos.***Sprint* definidos**

|  |
| --- |
|  |
| ***Sprint* 1 - Mantenedores** |
| *Sprint planning meeting* |
| Crear repositorio git control de versión |
| Carga de usuarios |
| Carga de roles |
| Carga de módulos |
| Plan de pruebas |
| Entrega hitos comprometidos |
| Revisión del *sprint* |
| Punto de control e incremento de producto |
| ***Sprint* 2 - Gestión de inventario** |
| *Sprint* *planning* meeting |
| Distribución de actividades |
| Actualizar tareas de épica pendientes |
| Definición de componentes a integrar |
| Plan de pruebas |
| Entrega hitos comprometidos |
| Revisión del *sprint* |
| Punto de control e incremento de producto |
| ***Sprint* 3 - Administración local** |
| *Sprint planning meeting* |
| Correcciones *sprint* anterior |
| Actualizar tareas de épica pendientes |
| Definición de componentes a integrar |
| Plan de pruebas |
| Entrega hitos comprometidos |
| Revisión del *sprint* |
| Punto de control e incremento de producto |
| ***Sprint* 4 - Administración ventas** |
| *Sprint planning meeting* |
| Correcciones *sprint* anterior |
| Actualizar tareas de épica pendientes |
| Definición de componentes a integrar |
| Plan de pruebas |
| Entrega hitos comprometidos |
| Revisión del *sprint* |
| Punto de control e incremento de producto |
| ***Sprint* 5 - Despliegue** |
| Aseguramiento de calidad |
| Aplicar correcciones en forma local |
| Coordinación con cliente para probar en forma local |
| ***Sprint* 6 - producción** |
| Cargar última versión código fuente |
| Crear y configurar BD MySql |
| Levantar sitio en modo producción |
| Cierre |

Tabla N° 6 – *Sprint* Definidos



**INSTITUTO PROFESIONAL CIISA**

**INGENIERÍA EN INFORMÁTICA**

## DOCUMENTO DE INVESTIGACION INICIAL

**DELYAPP**

**Un *software* de apoyo a la administración de local de comida rápida**

**Profesor Revisor:**

Patricio Sorich Huerta

**Alumnos:**

Nicolás González Araneda

Gabriel Aravena Rivero

Ángel Bravo Enrique

**Santiago**

**Diciembre, 2020**

[1 Historial del Documento](#_Toc60001516)

[2 Resumen](#_Toc60001518)

[3 Investigación](#_Toc60001519)

[3.1 Legal](#_Toc60001520)

[3.1.1LEY N° 19.496 sobre la protección de los derechos de los consumidores.](#_Toc60001521)

[3.1.2LEY N° 19.628 sobre protección de la vida privada](#_Toc60001522)

[3.1.3LEY N°21.096 consagra el derecho a protección de los datos personales](#_Toc60001523)

[3.2 Organizacional](#_Toc60001524)

[3.3 Procesos](#_Toc60001525)

[3.4 Técnica](#_Toc60001526)

[3.4.1 *Frameworks*](#_Toc60001527)

[3.4.1.1 Laravel 7:](#_Toc60001528)

[3.4.1.2 Symfony 4:](#_Toc60001529)

[3.4.2Motores de base de datos](#_Toc60001530)

[3.4.2.1 PostgreSQL](#_Toc60001531)

[3.4.2.2 MySQL](#_Toc60001532)

[3.4.2.3 MongoDB](#_Toc60001533)

[3.5 Metodológica](#_Toc60001534)

[3.6 Estudio Financiero:](#_Toc60001535)

[3.6.1 control de inventario:](#_Toc60001536)

[3.6.1.1 Clasificación de Inventarios ABC:](#_Toc60001537)

[3.6.1.2 Precio Medio Ponderado (P.M.P.):](#_Toc60001538)

[3.6.2 Margen de Ganancia:](#_Toc60001539)

[4 Conclusión:](#_Toc60001540)

**Historial del Documento**

|  |  |  |  |
| --- | --- | --- | --- |
| Fecha | Versión Documento | Descripción | Responsable |
| 21-05-2020 | 1.0 | Información del proyecto, introducción | Ángel Bravo |
| 25-05-2020 | 1.1 | Investigación de *frameworks*, aspecto legal | Nicolás González |
| 29-05-2020 | 1.2 | Investigación de aspecto legal, motores de base de datos, organizacional y metodología. | Gabriel Aravena |
| 20-06-2020 | 1.3 | Agregar resumen, eliminar Información del proyecto e Introducción. | Gabriel Aravena |
| 25-06-2020 | 1.4 | Agregar Estudio Financiero | Ángel Bravo |
| 28/06/2020 | 1.5 | Cambio en punto metodología, *framework* y revisión final. | Ángel Bravo  Nicolás González |
| 21/07/2020 | 1.6 | Agregar Conclusión | Ángel Bravo |

**Resumen**

DELYAPP es un *software* de inventarios y *delivery* que surge de los problemas de gestión que tienen algunas Pymes dedicadas al negocio de la comida rápida. Este *software* busca apoyar a cada Pyme en la gestión de ventas, además de ayudar en la gestión de inventarios bajo método financieros, esto permitirá que estos pequeños negocios sean capaces de llevar una gestión limpia, ordenada y automatizada. Delyapp es un *software* web que se desarrollará con las últimas tendencias web. Cuenta con tres responsables, Nicolás Gonzales, Ángel Bravo y Gabriel Aravena, teniendo como fecha de término del proyecto el 18 de diciembre de 2020.

**Investigación**

* *Framework* de desarrollo:
  + - Se investigó las distintas opciones de *frameworks*, considerando la naturaleza de nuestro *software*, su estructura y finalidad. Se pretende utilizar la mejor herramienta disponible para el desarrollo del *software*.
* Motores de bases de datos:
  + - Se investigó las posibles opciones de gestores de bases de datos, que se ajusten a la arquitectura del *software*, y nuestro presupuesto.
* Aspectos legales:
  + - Se investigó las leyes vigentes que conciernen a nuestro proyecto, en particular las leyes sobre protección de la vida privada y la ley de protección del derecho del consumidor. La investigación de la legislación es de suma importancia, ya que permite desarrollar el proyecto dentro del espectro legal, cuidando no tener problemas en este ámbito luego de la puesta en producción.
  + Aspectos organizacionales:
    - Se investigó el impacto que pueden generar los distintos elementos organizacionales existentes en nuestros potenciales clientes, con la finalidad de establecer cuáles podrían afectar el normal desarrollo de nuestro proyecto.

**Legal**

**LEY N° 19.496 sobre la protección de los derechos de los consumidores**

Este cuerpo legal establece derechos y obligaciones para consumidores y empresas y entre los temas principales que regula están:

* La regulación de la publicidad engañosa y la eliminación de la llamada "letra chica" en los contratos.
* El derecho a contar con información veraz y oportuna sobre los bienes y servicios ofrecidos.
* La no discriminación arbitraria por parte de empresas proveedoras de bienes o servicios.

**LEY N° 19.628 sobre protección de la vida privada**

Esta ley regula el trato de los datos de carácter personal, en registros o bancos de datos, por organismos públicos o privados, y es uno de los estatutos normativos más relevantes sobre la materia.

**LEY N°21.096 consagra el derecho a protección de los datos personales**

Modifica el artículo 19 de la Constitución Política de la República que consagra las garantías constitucionales de las personas, para agregar en su numeral 4º el Derecho a la Protección de los Datos Personales. De este modo en la Carta Fundamental, se reconoce la protección de los datos personales, y además señala que el tratamiento y protección de estos datos, se efectuará en la forma y condiciones que determine la ley.

**Organizacional**

La implementación de un sistema de información es una actividad que requiere análisis y planificación acorde con el impacto que esta actividad pueda implicar para la organización; en este sentido, se ha detectado que el mayor impacto positivo organizacional se generaría en la dimensión económica, puesto que nuestro sistema permite medir y controlar los gastos y ganancias. Otra dimensión afectada positivamente es la tecnológica, ya que implementar el sistema conlleva profesionalizar procesos que se llevaban a cabo de forma analógica. Finalmente, es posible detectar un posible impacto negativo en el ámbito estratégico, esto porque sumar nuestro sistema a sus procesos podría exigir un cambio estructural, principalmente a la forma en cómo se desarrolla en negocio.

**Procesos**

Como no se cuenta con un cliente, no es posible definir si se modificarán algunos procesos del cliente.

**Técnica**

***Frameworks***

**Laravel 7:**

Es el *framework* más popular y es uno de los primeros de la lista de los más usados de hoy en día, cuenta con muchas características positivas para el proyecto, una de ellas es que cuenta con Blade (sistemas de plantillas para las vistas), *Eloquent* (ORM que incluye este *framework* para el manejo de BD), controlar las rutas de nuestro sistema y además también cuenta con una comunidad muy grande y una documentación en su página oficial.

**Symfony 4:**

Este *framework* no es tan popular, pero si es uno de los numero uno en la lista de los más usados en la actualidad, también tiene características importantes que puede ser de aporte para el desarrollo del *software* tales como: Seguridad, ya que cuenta con un componente llamado SECURITY con el cual se puede controlar el acceso a nuestra aplicación, también cuenta con la integración de PHPUnit que permite llevar a cabo test unitarios, test funcionales o de integración. Archivos ENV que nos permiten configurar distintos aspectos de la aplicación en su etapa de desarrollo que posteriormente llevada a la etapa de producción bastara con definir esas mismas variables en el servidor, trabajar en distintos entornos.

**Motores de base de datos**

**PostgreSQL**

Es un sistema de gestión de bases de datos relacionales orientados a objetos y de código abierto, publicado bajo la licencia PostgreSQL.

Mediante un sistema denominado MVCC (Acceso concurrente multi versión, por sus siglas en inglés) PostgreSQL permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Cada usuario obtiene una visión consistente.

Algunas ventajas:

* Seguridad en términos generales
* Integridad en BD: restricciones en el dominio
* Integridad referencial
* Afirmaciones (*Assertions*)
* Disparadores (*Triggers*)
* Autorizaciones
* Conexión a sistema de gestión de bases de datos
* Transacciones y respaldos

**MySQL**

Es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual: Licencia pública general/Licencia comercial por *Oracle* *Corporation* y está considerada como la base de datos de código abierto más popular del mundo.

MySQL es una base de datos muy rápida en la lectura cuando utiliza el motor no transaccional MyISAM, pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. En aplicaciones web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones. Sea cual sea el entorno en el que va a utilizar MySQL, es importante monitorizar de antemano el rendimiento para detectar y corregir errores tanto de SQL como de programación.

Características adicionales:

* Usa GNU Automake, Autoconf, y Libtool para portabilidad
* Uso de multihilos mediante hilos del kernel.
* Usa tablas en disco b-tree para búsquedas rápidas con compresión de índice
* Tablas hash en memoria temporales
* El código MySQL se prueba con Purify (un detector de memoria perdida comercial) así como con Valgrind, una herramienta GPL.
* Completo soporte para operadores y funciones en cláusulas select y where.
* Completo soporte para cláusulas group by y order by, soporte de funciones de agrupación
* Seguridad: ofrece un sistema de contraseñas y privilegios seguro mediante verificación basada en el host y el tráfico de contraseñas está cifrado al conectarse a un servidor.
* Soporta gran cantidad de datos. MySQL Server tiene bases de datos de hasta 50 millones de registros.
* Se permiten hasta 64 índices por tabla (32 antes de MySQL 4.1.2). Cada índice puede consistir desde 1 hasta 16 columnas o partes de columnas. El máximo ancho de límite son 1000 bytes (500 antes de MySQL 4.1.2).
* Los clientes se conectan al servidor MySQL usando sockets TCP/IP en cualquier plataforma. En sistemas Windows se pueden conectar usando named pipes y en sistemas Unix usando ficheros socket Unix.
* En MySQL 5.0, los clientes y servidores Windows se pueden conectar usando memoria compartida.
* MySQL contiene su propio paquete de pruebas de rendimiento proporcionado con el código fuente de la distribución de MySQL.

**MongoDB**

Es un sistema de base de datos NoSQL, orientado a documentos y de código abierto. En lugar de guardar los datos en tablas, tal y como se hace en las bases de datos relacionales, MongoDB guarda estructuras de datos BSON (una especificación similar a JSON) con un esquema dinámico, haciendo que la integración de los datos en ciertas aplicaciones sea más fácil y rápida.

La base de datos MongoDB es adecuada para los siguientes usos:

* Almacenamiento y registro de eventos
* Sistemas de manejo de documentos y contenido
* Comercio electrónico
* Juegos
* Sistemas con alto volumen de lecturas
* Aplicaciones móviles
* Almacén de datos operacional de sitios web
  + Almacenamiento de comentarios
  + Votaciones
  + Registro de usuarios
  + Perfiles de usuarios
  + Sesiones de datos
* Proyectos que utilizan metodologías de desarrollo iterativo o ágiles
* Manejo de estadísticas en tiempo real

**Metodológica**

Se investigó la metodología Scrum debido a los cambios que van surgiendo con frecuencia. Esta metodología se adapta a estos cambios trabajando colaborativamente entre los involucrados en el proyecto. Dentro del anexo marco metodológico se detalla este punto.

**Estudio Financiero**

Se investigó sobre los aspectos financieros que va a manejar el cliente en su inventario y que se van a reflejar en el *software* como precio final que se quiera mostrar.

**control de inventario**

El control de inventario son los costos asociados a los productos disponibles en bodega o almacén para su venta futura.

Se investigó los indicadores financieros para llevar un correcto control de inventario.

**Clasificación de Inventarios ABC**

La clasificación ABC consiste en aplicar la regla de 80/20 o el principio de Pareto para segmentar entidades, tales como productos, clientes, proveedores, etc. Es muy utilizado en el ámbito del almacén para clasificar el inventario según su importancia.

Lo bueno del ABC es que se puede adaptar a diferentes criterios, según sea el almacén o tipo de mercadería que se maneja. Uno de los criterios mas usados con la clasificación ABC es el valor del inventario de cada referencia, calculado como su demanda anual multiplicada por su coste unitario. Después, se ordena de mayor a menor y se agrupan según el porcentaje que representan del total, los porcentajes resultantes serias:

* Categoría A: En torno al 20% de las referencias representan aproximadamente el 80% del valor del inventario, cumpliéndose la regla 80/20.
* Categoría B: En torno al 30% de las referencias representan aproximadamente el 15% del valor del inventario.
* Categoría C: En torno al 50% de las referencias representan solo el 5% del valor del inventario.

**Precio Medio Ponderado (P.M.P.)**

El PMP es un método de valorización en contabilidad que se obtiene al realizar un cálculo del valor medio de existencia (productos) que había al inicio y las cantidades ponderas según sus cantidades.

PMP relaciona los costes totales de adquirir existencias y sus cantidades respectivas, de este modo se obtiene una valorización de existencias homogéneas; es muy usado cuando no se tiene un criterio especifico a la hora de dar salida a las existencias.

Para calcular el PMP es necesario conocer una sencilla formula:

**PMP = ( ∑Pi \* Qi ) / ( ∑ Qi )**

**∑Pi=** Suma del precio de cada bien

**Qi=** Cantidad de bienes comprados

**∑Qi=** Cantidad total de bienes

**Margen de Ganancia**

El margen de ganancia es el porcentual que se añade al coste total del producto, lo que define el precio del producto por el que será comercializado al cliente. El margen de ganancia se calcula a partir de los gastos fijos unitarios, de los gastos variables y margen de beneficio esperado.

Para calcular el margen de ganancia se utilizará el método **Mark Up**, utilizaremos los siguientes índices de un negocio:

* **Porcentaje de Gastos Variables (GV):** Gastos variables que inciden directamente sobre la venta, como comisión, flete, etc. Se debe expresar en términos de porcentaje
* **Porcentaje de Gastos Fijos Unitarios (GF):** Gastos fijos del negocio que una unidad del producto debe “pagar”, tales como gastos administrativos, luz, agua, salarios, internet, etc. También expresados en porcentaje.
* **Margen de Ganancia Deseada (MG):** Porcentaje que el dueño o encargado desea tener como margen de ganancia en su negocio.
  + La fórmula para calcular el margen de ganancia con **Mark Up** es:
  + **100 / [100 - (GV + GF + MG)]**

**Conclusión:**

Luego de la investigación llevada a cabo en el punto de *framework*, se ha llegado al consenso de utilizar el *Framework* **Laravel 7** para construcción del *software*, principalmente por la robustez que ofrece y la experiencia de los miembros en su utilización.

Por otro lado, con respecto a los motores de base de datos, se ha resuelto utilizar **MySql** como el motor de base de datos para la aplicación. Se ha llegado a esta conclusión puesto que esta opción nos ofrece la creación de bases de datos relacionales, que consideramos indispensables. Además, cuenta con una vasta documentación, su popularidad hace que tenga una comunidad muy activa. Cuenta también con su propio paquete de pruebas, y la seguridad suficiente para el fin que se busca.

Y con respecto a punto de marco financiero, se aplicará todo los sub puntos mencionados en ella. Ya que cada punto explica lo que se implementara en el *software*: inventario, margen de ganancia y el cálculo de precio final.



**INSTITUTO PROFESIONAL CIISA**

**INGENIERÍA EN INFORMÁTICA**

## DOCUMENTO DE GOBIERNO DEL PROYECTO

**DELYAPP**

**Un *software* de apoyo a la administración de local de comida rápida**

**Profesor Revisor:**

Patricio Sorich Huerta

**Alumnos:**

Nicolás González Araneda

Gabriel Aravena Rivero

Ángel Bravo Enrique

**Santiago**

**Diciembre 2020**

INDICE

[1 Historial del Documento 85](#_Toc59231181)

[2 Resumen del proyecto 86](#_Toc59231182)

[3 Gobierno del Proyecto 86](#_Toc59231183)

[3.1 Equipo Desarrollador 86](#_Toc59231184)

[3.2 Clientes 86](#_Toc59231185)

[4 Roles del Proyecto. 86](#_Toc59231186)

[4.1 Product Owner. 86](#_Toc59231187)

[4.2 StakeHolder 87](#_Toc59231188)

[4.3 Scrum Master 87](#_Toc59231189)

[4.4 Scrum Team. 87](#_Toc59231190)

[5 Matriz RACI 87](#_Toc59231191)

**Historial del Documento**

|  |  |  |  |
| --- | --- | --- | --- |
| Fecha | Versión Documento | Descripción | Responsable |
| 20/06/2020 | 1.0 | Confección de documento | Ángel Bravo |
| 28/06/2020 | 1.1 | Agregar matriz RACI | Nicolás González |

**Resumen del proyecto**

El enfoque de este documento es en las personas y recursos involucrados por tanto se describen en los siguientes puntos.

**Gobierno del Proyecto**

La gobernabilidad del proyecto es de suma importancia ya que es una función de supervisión, proporciona al gerente del proyecto y al equipo una estructura, estándares y herramientas para dirigir el proyecto y también apoya y controla para lograr una entrega exitosa.

**Equipo Desarrollador**

El equipo de desarrollo está compuesto por 3 alumnos del Instituto Profesional de Ciisa, los cuales son:

* Nicolas González
* Gabriel Aravena
* Ángel Bravo

**Clientes**

El cliente en nuestro caso es La señora Chabela. Se denomina como “Emprendimiento familiar”. La señora Chabelita es dueña del producto y el profesor guía es Patricio Sorich quien revisa la documentación y el avance del producto de *software*.

**Roles del Proyecto**

Los roles están definidos por la tarea y responsabilidad que tienen en el proyecto. El detalle se encuentra en el anexo del marco metodológico.

**Product Owner**

Chabela, la dueña del producto, es encargada de describir los procesos del negocio, definir las necesidades de su Pyme, también de entregar las funcionalidades y una pequeña reseña de lo que como él se imagina el sistema.

**StakeHolder**

Patricio Sorich quien está dispuesto a entregar su opinión y dar *feedback* dentro de los procesos del proyecto. Es uno de los interesados en el proyecto y producto.

**Scrum Master**

Nicolas González, encargado de planificar, desarrollar y supervisar el correcto flujo que debe tener el proyecto para que sea exitosa.

**Scrum Team**

Compuesto por Ángel Bravo y Gabriel Aravena, encargados de correcto desarrollo de requerimientos funcionales y no funcionales cumpliendo con los estándares y respetando la gobernabilidad del cliente.

**Matriz RACI**

Una vez establecido las responsabilidades dentro del marco de trabajo en el desarrollo e identificar las tareas asignadas, se dibujará esta matriz RACI para mejor comprensión en las responsabilidades de cada miembro en el ciclo de vida del *software*.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ROLES  TAREAS | PRODUCT OWNER | STAKEHOLDERS | SCRUM MASTER | SCRUM TEAM |
| Identificación de necesidades | A | R | I | I |
| Estudio de análisis | A | C | R | I |
| Estudio de factibilidad | A | C | R | I |
| Planificación del Proyecto | A | C | R | I |
| Desarrollo del Proyecto | A | C | R | I |
| Desarrollo de producto | A | I | R | R |
| Validación de Entregables | A | C | R | R |
| Modificar necesidades | R | I | A | C |

Tabla N° 1 – Matriz RACI



**INSTITUTO PROFESIONAL CIISA**

**INGENIERÍA EN INFORMÁTICA**

## DOCUMENTO PLAN DE RIESGO

**DELYAPP**

**Un *software* de apoyo a la administración de local de comida rápida**

**Profesor Revisor:**

Patricio Sorich Huerta

**Alumnos:**

Nicolás González Araneda

Gabriel Aravena Rivero

Ángel Bravo Enrique

**Santiago**

**Diciembre 2020**

INDICE

[1 Historial del Documento 91](#_Toc59236817)

[2 Alcance del Documento 92](#_Toc59236818)

[3 Resumen del proyecto 92](#_Toc59236819)

[4 Identificación de Riesgos 92](#_Toc59236820)

[5 Categorías de Riesgo 92](#_Toc59236821)

[5.1 De Proyecto 93](#_Toc59236822)

[5.2 Tecnológico 93](#_Toc59236823)

[5.3 Riesgos Categorizados 93](#_Toc59236824)

[6 Medición y Evaluación de Riesgos 94](#_Toc59236825)

[6.1 Probabilidad de Ocurrencia 94](#_Toc59236826)

[6.2 Impacto 94](#_Toc59236827)

[6.3 Nivel de Exposición 95](#_Toc59236828)

[6.4 Tabla de Medición de Riesgos 96](#_Toc59236829)

[7 Plan de Actividades 97](#_Toc59236830)

[7.1 Actividades de Evitación 97](#_Toc59236831)

[7.2 Actividades de Mitigación 97](#_Toc59236832)

[7.3 Actividades de Contingencia 97](#_Toc59236833)

[7.4 Riesgos y sus actividades 98](#_Toc59236834)

[8 Seguimiento de Riesgos 100](#_Toc59236835)

**Historial del Documento**

|  |  |  |  |
| --- | --- | --- | --- |
| Fecha | Versión Documento | Descripción | Responsable |
| 20-06-2020 | 1.0 | Inicio Plan de Gestión de Riesgos | Gabriel Aravena |
| 28-06-2020 | 1.1 | Modificación Plan de Actividades | Ángel Bravo |
| 20-07-2020 | 2.0 | Re identificación de riesgos y plan de actividades | Gabriel Aravena |

**Alcance del Documento**

El plan de gestión de riesgos busca identificar los principales peligros a los que se encuentra expuesto el proyecto, en sus diversos aspectos. A través del desarrollo de un plan es posible solucionar o buscar alternativas a estos peligros.

El presente documento presenta la estrategia utilizada para identificar los riesgos, su clasificación, la forma en que se midieron y las estrategias para evitarlos o mitigarlos. Además, se presenta una estrategia de seguimiento, con la intención de monitorear su comportamiento durante todo el proceso de desarrollo del *software*, identificar si existen nuevos riesgos o si hay riesgos que pierden su impacto.

**Resumen del proyecto**

DELYAPP es un *software* de inventarios y *delivery* que surge de los problemas de gestión que tienen algunas Pymes dedicadas al negocio de la comida rápida. Este *software* busca apoyar a cada Pyme en la gestión de ventas, además de ayudar en la gestión de inventarios bajo métodos financieros, esto permitirá que estos pequeños negocios sean capaces de llevar una gestión limpia, ordenada y automatizada. DelyApp es un *software* web que se desarrollará con las últimas tendencias web. Cuenta con tres responsables, Nicolás Gonzales, Ángel Bravo y Gabriel Aravena, teniendo como fecha de término del proyecto el 18 de diciembre de 2020.

**Identificación de Riesgos**

Para identificar los riesgos se ha analizado toda la documentación del proyecto, principalmente el documento de inicio y el documento de investigación, identificando las vulnerabilidades y las posibles amenazas a las que está expuesto el proyecto. Luego del análisis se ha realizado una verificación incluyendo a todo el equipo y consultando a expertos, con la finalidad de reevaluar la veracidad de los riesgos y su posible impacto en el proyecto.

**Categorías de Riesgo**

Se categorizaron los riesgos identificados en dos grupos, por una parte, los riesgos de proyecto, y por otra los riesgos tecnológicos. Es importante hacer esta categorización, ya que de esta forma es posible identificar cómo impactan y cuáles son sus consecuencias.

**De Proyecto**

Son riesgos que amenazan la planificación del proyecto, en términos de tiempo, coste (esfuerzo) o calidad del proyecto. Tienen un impacto en el calendario definido y en los recursos del proyecto.

**Tecnológico**

Amenazan la calidad y la planificación temporal del *software* que hay que producir. Pueden producirse porque el problema a resolver es más difícil de lo que se pensaba. Si un riesgo técnico se convierte en realidad, la implementación puede ser difícil o incluso imposible.

**Riesgos Categorizados**

A continuación, se muestran los riesgos identificados según la categoría a la cual corresponden.

|  |  |
| --- | --- |
| Riesgo | Categoría |
| Daño en las dependencias de desarrollo (incendio, derrumbe, catástrofe natural). | Proyecto |
| Cambio de forma inesperada en las normas legales. | Proyecto |
| El alcance de aplicación está mal definido. | Proyecto |
| Los cambios no son controlados y el alcance está en continuo crecimiento. | Proyecto |
| Poco compromiso de parte del cliente. | Proyecto |
| El producto no cumple con los requisitos. | Proyecto |
| Bajo compromiso por parte del equipo de desarrollo. | Proyecto |
| Falta de un miembro del equipo de desarrollo por motivos de fuerza mayor. | Proyecto |
| Cambio en las condiciones del país por razones de fuerza mayor. | Proyecto |
| Desconocimiento de las tecnologías a utilizar por parte del equipo de desarrollo. | Tecnológico |
| Los servicios externos a contratar (ej: hosting) no cumplen con las características necesarias. | Tecnológico |

Tabla N° 1 – Riesgos por categoría

**Medición y Evaluación de Riesgos**

Los riesgos se midieron de forma cualitativa, considerando su probabilidad y el impacto que puedan generar al proyecto. Con estas características identificadas se determinó un nivel de exposición, que permite identificar los riesgos más críticos para su posterior tratamiento y seguimiento.

Es de suma importancia medir y evaluar los riesgos identificados, ya que esto permite priorizar ciertos riesgos que puedan tener una mayor influencia en el desarrollo del proyecto.

**Probabilidad de Ocurrencia**

En el caso de la probabilidad de ocurrencia del riesgo, se definen 5 niveles (raro, improbable, moderado, probable y casi certeza), a cada uno de estos niveles se le asignará un valor, del 1 al 5 respectivamente, que permite asociar la probabilidad con el impacto para determinar un nivel de exposición.

**Impacto**

En el caso del impacto, se definen 5 niveles (insignificante, menor, moderado, mayor y catastrófico), a cada uno de estos niveles se le asignará un valor, del 1 al 5 respectivamente, que permite, como se mencionó, asociar el impacto con la probabilidad de ocurrencia del riesgo y determinar el nivel de exposición.

**Nivel de Exposición**

El nivel de exposición relaciona el impacto con la probabilidad de ocurrencia. Permite calcular qué tan crítico es el riesgo y de esta forma establecer una prioridad de acción para gestionarlos.

Para determinar el nivel de exposición de cada riesgo, se tomará el valor asociado a la probabilidad y se sumará con el valor asociado al impacto, y dependiendo del valor obtenido se asignará un nivel de exposición de entre los 4 niveles definidos (bajo, moderado, alto o extremo), como muestra la siguiente tabla:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Nivel de Exposición | | Impacto | | | | |
| Insignificante  (1) | Menor (2) | Moderado (3) | Mayor  (4) | Catastrófico  (5) |
| Probabilidad de Ocurrencia | Casi certeza (5) |  |  |  |  |  |
| Mayor  (4) |  |  |  |  |  |
| Moderado (3) |  |  |  |  |  |
| Improbable (2) |  |  |  |  |  |
| Raro  (1) |  |  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Nivel de exposición (Probabilidad + Impacto)** | | | |
| Extremo | Suma entre 9 y 10 | Alto | Suma entre 7 y 8 |
| Moderado | Suma entre 5 y 6 | Bajo | Suma entre 2 y 4 |

**Tabla de Medición de Riesgos**

En la siguiente tabla se muestran los riesgos junto a su nivel de probabilidad, su nivel de impacto y nivel de exposición.

|  |  |  |  |
| --- | --- | --- | --- |
| Riesgo | Probabilidad de Ocurrencia | Impacto | Nivel de Exposición |
| 1. Daño en las dependencias de desarrollo (incendio, derrumbe, catástrofe natural). | 2 | 4 | Moderado |
| 1. Cambio de forma inesperada en las normas legales. | 2 | 2 | Bajo |
| 1. El alcance de aplicación está mal definido | 4 | 5 | Extremo |
| 1. Los cambios no son controlados y el alcance está en continuo crecimiento. | 3 | 3 | Moderado |
| 1. Poco compromiso de parte del cliente. | 4 | 4 | Alto |
| 1. El producto no cumple con los requisitos. | 4 | 4 | Alto |
| 1. Bajo compromiso por parte del equipo de desarrollo. | 2 | 4 | Moderado |
| 1. Falta de un miembro del equipo de desarrollo por motivos de fuerza mayor. | 3 | 3 | Moderado |
| 1. Cambio en las condiciones del país por razones de fuerza mayor. | 4 | 3 | Moderado |
| 1. Desconocimiento de las tecnologías a utilizar por parte del equipo de desarrollo. | 3 | 4 | Alto |
| 1. Los servicios externos a contratar (ej: hosting) no cumplen con las características necesarias. | 3 | 3 | Moderado |

Tabla N° 2 – Medición de Riesgos

**Plan de Actividades**

El plan de actividades tiene la finalidad de establecer procedimientos que permitan gestionar los riesgos, es decir, acciones que permitan evitar los riesgos o disminuir su impacto en los casos en que no se puedan evitar.

**Actividades de Evitación**

Las actividades de evitación son acciones que permiten reducir la probabilidad de ocurrencia de un riesgo, anticipándose a este.

**Actividades de Mitigación**

Las actividades de mitigación son acciones que se llevan a cabo cuando no se ha podido evitar el riesgo, y tienen la finalidad de reducir el impacto que este tiene en el proyecto.

**Actividades de Contingencia**

Son acciones que se llevan a cabo cuando no se ha podido evitar y tampoco mitigar el riesgo, tienen la finalidad de sanear los impactos generados por la ocurrencia del riesgo y permitir que el proyecto siga en pie.

**Riesgos y sus actividades**

En la siguiente tabla de muestran los riesgos ordenados por su nivel de exposición, junto a las actividades de evitación, mitigación y contingencia.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Riesgo | Exposición | Actividades de Evitación | Actividades de Mitigación | Actividades de Contingencia |
| 1. El alcance de aplicación está mal definido | Extremo | Realizar una reunión con el cliente y validar el alcance definido. | Realizar una reunión con el cliente y modificar el alcance. | Realizar una reunión con el cliente y negociar los alcances. |
| 1. El producto no cumple con los requisitos. | Alto | Realizar una correcta toma de requerimientos y aclarar con el cliente las funcionalidades a abordar. | Negociar con el cliente los requerimientos más críticos para ser implementados. | Realizar una nueva toma de requisitos considerando los errores cometidos. |
| 1. Poco compromiso de parte del cliente. | Alto | Mantener una constante comunicación, informando los avances y cambios del proyecto. | Exigir la participación activa del cliente indicando la relevancia que tiene en el proyecto. | Considerar los requerimientos fijados en el proyecto, traspasando el riesgo al cliente. |
| 1. Desconocimiento de las tecnologías a utilizar por parte del equipo de desarrollo. | Alto | Utilizar herramientas que el equipo conozca y haya utilizado con anterioridad. | Estudiar la tecnología e investigar sus actualizaciones. | Cambiar rápidamente de herramienta a una que sea de conocimiento del equipo. |
| 1. Daño en las dependencias de desarrollo (incendio, derrumbe, catástrofe natural). | Moderado | Realizar el proyecto en un lugar apto, manteniendo estándares mínimos de seguridad. | Realizar capacitación respecto a las normas de seguridad y acciones en caso de emergencia. | Migrar a un ambiente de desarrollo online. |
| 1. Bajo compromiso por parte del equipo de desarrollo. | Moderado | Contar con un equipo de confianza con una buena relación interpersonal. | Exigir el cumplimiento de las tareas asignadas. | Desvincular a la persona implicada, buscando un reemplazo en el menor tiempo posible. |
| 1. Los cambios no son controlados y el alcance está en continuo crecimiento. | Moderado | Establecer que las modificaciones de gran volumen deben considerarse para otro proyecto. | No aceptar cambios que modifiquen considerablemente nuestro alcance. | Re planificar el proyecto con las nuevas modificaciones y respetando a fecha final. |
| 1. Falta de un miembro del equipo de desarrollo por motivos de fuerza mayor. | Moderado | Contar con un equipo confiable, y con una alta motivación.  Mantener cuidados de salud. | Contar con un registro de posibles reemplazantes en caso de requerirlo. | Dividir las tareas entre los participantes mientras no se cuente con un reemplazo. |
| 1. Cambio en las condiciones del país por razones de fuerza mayor. | Moderado | Tener un estilo de trabajo de baja dependencia de la situación nacional. Fomentando el trabajo a distancia. | Cambiar a un estilo de trabajo completamente online. | Aceptar el riesgo, comunicando al cliente los posibles retrasos en el proyecto. |
| 1. Los servicios externos a contratar (ej: hosting) no cumplen con las características necesarias. | Moderado | Realizar una investigación sobre la calidad del servicio a contratar. | Cambiar el servicio contratado por uno que cumpla las expectativas. | Acordar con el cliente un servicio con menores prestaciones. |
| 1. Cambio de forma inesperada en las normas legales. | Bajo | Revisar constantemente los proyectos de ley y posibles cambios en la normativa legal que nos afecta. | Modificar los alcances del proyecto, redefiniendo los requerimientos afectados. | Acordar con el cliente las funcionalidades que se pueden prescindir. |

Tabla N° 3 – Riesgos y sus actividades

**Seguimiento de Riesgos**

Para el seguimiento de los riesgos, se definió un encargado de riesgos, quien tiene la tarea de monitorear y evaluar la evolución de los riesgos identificados, además de identificar nuevos riesgos que puedan amenazar el normal funcionamiento del proceso.

Cabe destacar el uso de la metodología Scrum ayuda a reducir los riesgos, ya que los ciclos iterativos cortos minimizan cualquier impacto imprevisto en el desarrollo del producto. Sin embargo, no es suficiente para controlarlos de manera exitosa.

Para esto, el encargado deberá ejecutar 4 acciones en cada uno del *sprint*:

* Chequear el estado de los riesgos presentes o residuales.
* Monitorear el comportamiento de los nuevos riesgos identificados.
* Implementar los planes de respuesta a los nuevos riesgos.
* Evaluar la efectividad del proceso de gestión de los riesgos y realizar los cambios pertinentes.

La intención de estas acciones es profundizar en el conocimiento de los riesgos, tener información actualizada de ellos, y tomar medidas concretas a tiempo, considerando el progreso que tienen a lo largo del proyecto.



**INSTITUTO PROFESIONAL CIISA**

**INGENIERÍA EN INFORMÁTICA**

## DOCUMENTO DE ANÁLISIS

**DELYAPP**

**Un *software* de apoyo a la administración de local de comida rápida**

**Profesor Revisor:**

Patricio Sorich Huerta

**Alumnos:**

Nicolás González Araneda

Gabriel Aravena Rivero

Ángel Bravo Enrique

**Santiago**

**Diciembre 2020**

INDICE

[1 Historial del Documento 103](#_Toc59894638)

[2 Resumen del Proyecto 103](#_Toc59894639)

[3 Análisis 104](#_Toc59894640)

[3.1 Diagrama de caso de uso 105](#_Toc59894641)

[3.2 Definición de Requerimientos 106](#_Toc59894642)

[3.3 Definición funcional 109](#_Toc59894643)

[3.4 Modelo relacional de base de datos 110](#_Toc59894644)

**Historial del Documento**

|  |  |  |  |
| --- | --- | --- | --- |
| Fecha | Versión Documento | Descripción | Responsable |
| 17/07/2020 | 1.0 | Confección inicial de documento de análisis | Ángel Bravo, Nicolás González, Gabriel Aravena |
| 22/07/2020 | 1.1 | Casos de uso | Nicolás González |

Tabla No 01 - Historial de documento

**Resumen del Proyecto**

DELYAPP es un *software* de inventarios y *delivery* que surge de los problemas de gestión que tienen algunas Pymes dedicadas al negocio de la comida rápida. Este *software* busca apoyar a cada Pyme en la gestión de ventas, además de ayudar en la gestión de inventarios bajo método financieros, esto permitirá que estos pequeños negocios sean capaces de llevar una gestión limpia, ordenada y automatizada. DelyApp es un *software* web que se desarrollará con las últimas tendencias web. Cuenta con tres responsables, Nicolás Gonzales, Ángel Bravo y Gabriel Aravena, teniendo como fecha de término del proyecto el 18 de diciembre de 2020.

**Análisis**

Este documento de análisis abarcó las principales actividades relacionadas a los requisitos y los candidatos a funcionalidades del *software* que son las épicas dentro de cada *sprint*.

Las subtareas de cada épica corresponden a las historias de usuario asociadas a la épica que pueden ser más de 1 y se definen en cada inicio de *sprint* o *sprint planning*.

En la semana 15 comenzamos a planificar las actividades asociadas a la construcción. Se han realizado las primeras actividades de la fase de análisis los requisitos.

Las estrategias, actividades y herramientas definidas en el marco metodológico, se aplicarán en estas primeras actividades y se usarán como base de trabajo, los alcances de producto ya definidos, así como la solución.

En estas actividades cobra importancia monitorear el avance a través de la planificación, como el seguimiento de los riesgos definidos que aparecen en la Carta Gantt como puntos de control, especialmente los asociados a esta etapa.

Se expusieron las historias de usuarios, las cuales, fueron categorizadas para cada *sprint* que serán 6 *sprints* de 3 semanas cada una. Se reunieron 20 historias de usuarios. Optamos como herramienta de toma de requisitos, una encuesta a través de un formulario que facilita la Suite de Google llamada Forms. Esta encuesta fue genérica para los dos locales comerciales entrevistados.

Al recibir respuestas, se traspasó la información a una plantilla Excel los requisitos capturados. Para obtener detalles de la encuesta se encuentra en la carpeta anexos como “encuesta.pdf”.

Bajo el marco metodológico nos apoyamos en la herramienta de Jira para cubrir el ciclo de vida del producto. El proceso contempló historias de usuarios que se agrupan en épicas. En cada *sprint* hay épicas que son 4 que completa el producto. Estas épicas contienen historias de usuario que a su vez contemplan subtareas que desarrollar y entregar en cada *sprint*.

**Diagrama de caso de uso**

Se complementa con un diagrama de caso de uso.

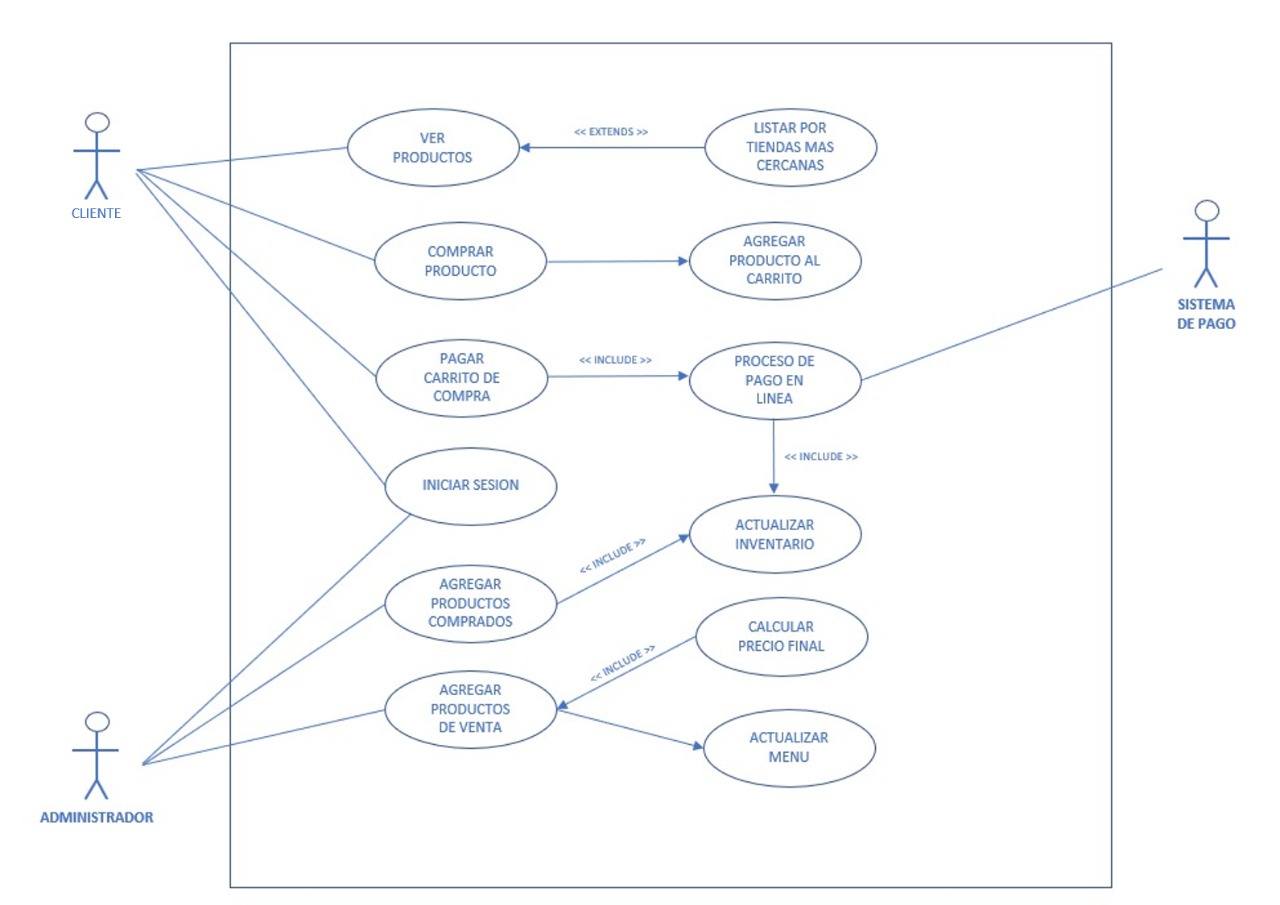


Diagrama No 01 - Diagrama de caso de uso

**Definición de Requerimientos**

Se definieron las historias de usuario con su título, épica que corresponde, *sprint* que corresponde, ID (identificación) y descripción.

La herramienta utilizada es Jira, un *software* bajo el marco de scrum que permitió a nuestro equipo gestionar el proyecto que sustentó una cultura de colaboración. El tablero de Jira Scrum es la herramienta que unió al equipo en torno a objetivos únicos y promueve una entrega iterativa e incremental.

Se dejó la tabla en su formato original en la carpeta de anexos llamada “Historias de usuario.xlsx”

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **EPICAS** | | | **HISTORIAS DE USUARIO** | | |
| **ID EPICA** | **TITULO** | **ID HISTORIA USUARIO** | **ROL** | **FUNCIONALIDAD** | **RAZON** |
| E1 | REALIZAR COMPRA COMO CLIENTE | H.U. 1 | Como Cliente | Quiero tener la opción de agregar productos a un carrito de compras | Para poder generar una compra. |
| H.U. 2 | Como Cliente | Quiero tener un buscador | Para poder tener un mejor filtro de lo que deseo encontrar. |
| H.U. 3 | Como Cliente | Quiero solicitar *delivery* | Para no tener que ir a buscar mi pedido al local. |
| H.U. 4 | Como Cliente | Quiero poder pagar a través de una tarjeta bancaria | Para hacerlo a través de internet, sin la necesidad de acercarme al local. |
| H.U. 5 | Como Cliente | Quiero recibir un correo de confirmación de mi pedido | Para asegurar que solicité la orden correctamente. |
| H.U. 6 | Como Cliente | Quiero poder visualizar el estado de mi pedido y el tiempo de despacho | Para poder organizarme con los tiempos de despacho o la demora que tomará el *delivery*. |
| E2 | GESTIONAR EL INVENTARIO DE LA TIENDA  (ADMINISTRADOR) | H.U. 7 | Como administrador | Quiero ingresar los ingredientes que ocupo en mi negocio, con su precio y las cantidades que tengo | Para tener un inventario |
| H.U. 8 | Como administrador | Quiero ingresar los productos que vendo, indicando la cantidad de ingredientes que ocupo en él | Para conocer la cantidad de ingredientes que gasto cada mes. |
| H.U. 9 | Como administrador | Quiero tener la opción de modificar el inventario | Para actualizar la cantidad de ingredientes cuando compre nuevamente. |
| H.U. 10 | Como administrador | Quiero tener la opción de auditar el inventario | Para saber cuáles han sido las pérdidas o mermas. |
| H.U. 11 | Como administrador | Quiero que al realizar una venta se actualice automáticamente el inventario | Para tener el inventario siempre actualizado. |
| H.U. 12 | Como administrador | Quiero poder visualizar la cantidad de ingredientes que tengo disponibles | Para saber cuándo es necesario comprar ingredientes nuevamente. |
| H.U. 13 | Como administrador | Quiero poder visualizar la cantidad de ingredientes que he gastado o perdido cada mes | Para saber cómo varían estos gastos. |
| E3 | ADMINISTRAR EL MENU DE LA TIENDA  (ADMINISTRADOR) | H.U. 14 | Como administrador | Quiero publicar los productos que tengo en venta | Para que los clientes puedan conocer mi menú y realizar compras. |
| H.U. 15 | Como administrador | Quiero que cuando publique un producto me sugieran un precio (no obligatorio) | Para poder tener un margen de ganancia que permita solventar mi negocio. |
| H.U. 16 | Como administrador | Quiero eliminar los productos del menú. | Para modificar el menú si es necesario. |
| H.U. 17 | Como administrador | Quiero modificar los productos que tengo en venta | Para poder cambiar los precios si es necesario. |
| E4 | GESTIONAR VENTAS (ADMINISTRADOR) | H.U. 18 | Como administrador | Quiero poder realizar una venta en mi local a través del *software* | Para que se registre automáticamente. |
| H.U. 19 | Como administrador | Quiero poder visualizar las ventas realizadas durante todos los meses | Para saber cómo evoluciona el negocio y tener claridad de cuánto se vende mensualmente. |
| H.U. 20 | Como administrador | Quiero visualizar cuánto dinero he gastado, cuánto ha ingresado y cuáles han sido las ganancias del negocio cada mes | Para saber si es necesario hacer modificaciones de precio y si las ganancias son las esperadas. |

Tabla No 02 - Historias de usuario

**Definición funcional**

Las funcionalidades se pueden agrupar en dos ámbitos. El primero se describen las funcionalidades que puede interactuar el administrador o dueño de local y luego las del usuario/cliente web:

Administrador:

* Interfaz del *admin*
* Contacto en modo cliente
* Configuración de la navegación
* Configuración del *home page*
* Módulo de ventas
* *Dashboard* y gráfico de estadísticas (Ganancias mensuales y semanales).
* Gestión de usuarios.
* Ofertas de productos.
* Personalización de la interfaz de *admin*

Cliente:

* Interfaz del cliente
* Carrito de compras
* Pasarela de pago Transbank
* Módulo para pedidos online
* Módulo de menú

**Modelo relacional de base de datos**

Se creo un modelo relacional de base de datos a la solución:

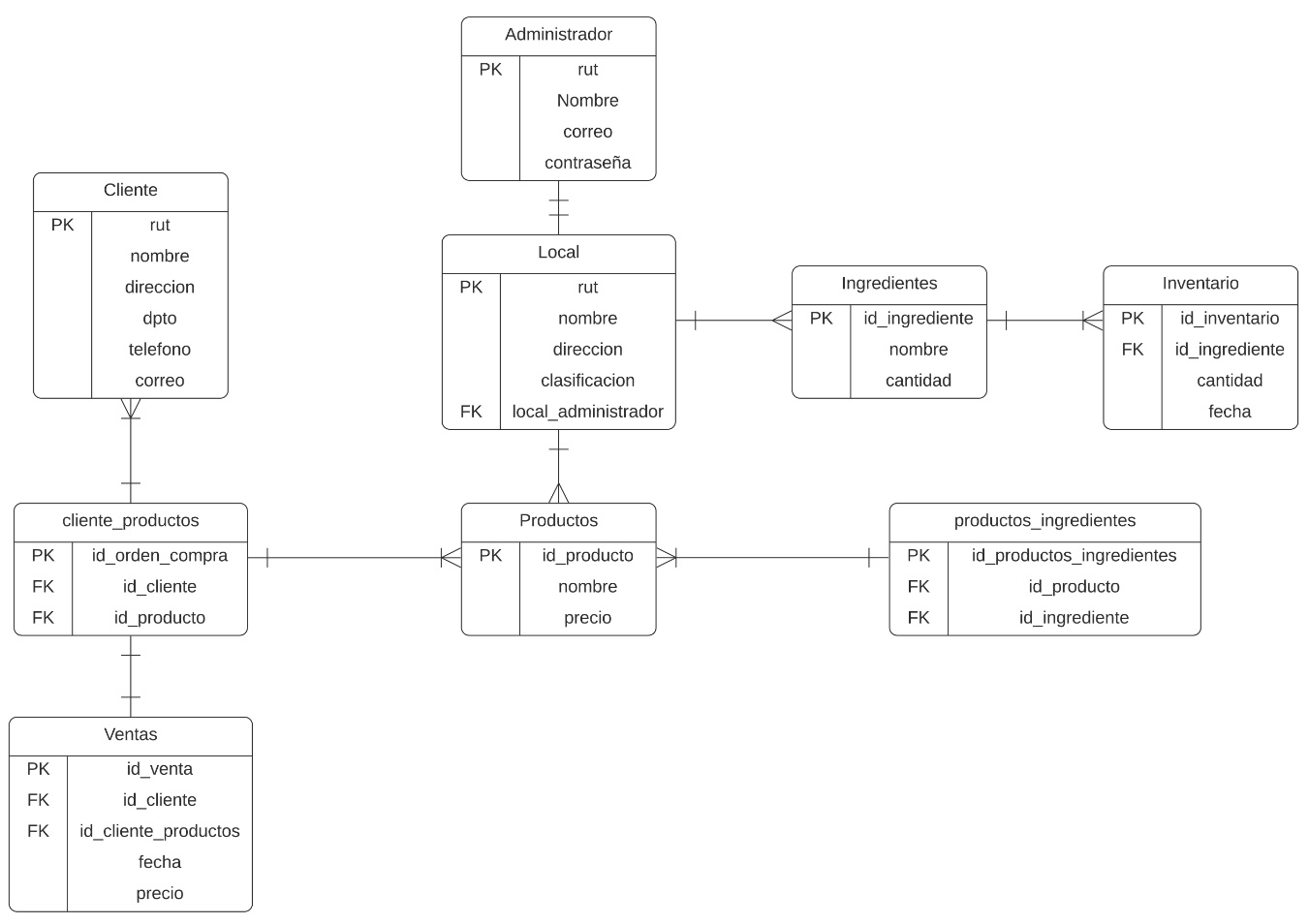


Diagrama No 02 - Modelo relacional de base de datos



**INSTITUTO PROFESIONAL CIISA**

**INGENIERÍA EN INFORMÁTICA**

## DOCUMENTO DE DISEÑO Y ARQUITECTURA

**DELYAPP**

**Un *software* de apoyo a la administración de local de comida rápida**

**Profesor Revisor:**

Patricio Sorich Huerta

**Alumnos:**

Nicolás González Araneda

Gabriel Aravena Rivero

Ángel Bravo Enrique

**Santiago**

**Diciembre 2020**

INDICE

[1 Historial del Documento 113](#_Toc58799831)

[2 Alcance del Documento 114](#_Toc58799832)

[3 Definiciones, Siglas y Abreviaciones 114](#_Toc58799833)

[4 Estándares del Proyecto 114](#_Toc58799834)

[4.1 Estándares de Diseño 115](#_Toc58799835)

[4.2 Estándares de Programación 115](#_Toc58799836)

[4.3 Convenciones 115](#_Toc58799837)

[5 Diseño del Sistema 116](#_Toc58799838)

[5.1 Diagrama de clases 116](#_Toc58799839)

[5.2 Modelo de Datos 118](#_Toc58799840)

[5.3 Diccionario de Datos 119](#_Toc58799841)

[5.4 Diagrama de Proceso. 122](#_Toc58799842)

[6 Arquitectura 124](#_Toc58799843)

[6.1 Solución Inicial 124](#_Toc58799844)

[6.2 Arquitectura de Hardware 125](#_Toc58799845)

[6.3 Arquitectura de *Software* 125](#_Toc58799846)

[6.4 Componentes de *Software* 129](#_Toc58799847)

[6.5 Diagrama de Despliegue 129](#_Toc58799848)

**Historial del Documento**

|  |  |  |  |
| --- | --- | --- | --- |
| Fecha | Versión Documento | Descripción | Responsable |
| 05-10-2020 | 1.0 | Estructuración de documento | Ángel Bravo |
| 06-10-2020 | 2.0 | Modificación de contenido | Ángel Bravo |
| 10-06-2020 | 3.0 | Actualización de contenido | Ángel Bravo |
| 01/12/2020 | 4.0 | Unificación documento de arquitectura con el de diseño | Nicolás González |
| 11/12/2020 | 5.0 | Creación diagrama de despliegue | Nicolás González |
| 14/12/2020 | 6.0 | Modificación de diagrama de clases | Nicolás González |

Tabla No 01 – Historial de versiones

**Alcance del Documento**

En el presente informe se especificaron los estándares de diseño que se implementó bajo el criterio de experiencias de usuario. Bajo el concepto de diseño y arquitectura se aplicó de forma tal que se cumplan los estándares que se trataron a lo largo de este documento. Además, se realizó el diseño del diagrama de clases que muestra la interacción entre clases y métodos y el diagrama de despliegue que ayuda de forma visual el comportamiento de la distribución física (estática) de los componentes *software* en los distintos nodos físicos de la red.

Por último, se detalla la arquitectura usada, contiene su parte física (hardware) e intangible (*software*) al respecto.

**Definiciones, Siglas y Abreviaciones**

* W3C: Consorcio mundial de la red
* Hardware: Es el conjunto de componentes físicos de los que está hecho el equipo o dispositivo
* *Software*:  Es el conjunto de programas o aplicaciones, instrucciones y reglas informáticas que hacen posible el funcionamiento del equipo.
* PHP: Lenguaje de código abierto muy popular, adecuado para desarrollo web y que puede ser incrustado en HTML
* MVC: El MVC o Modelo-Vista-Controlador es un patrón de arquitectura de *software* que, utilizando 3 componentes (Vistas, Modelos y Controladores) separa la lógica de la aplicación de la lógica de la vista en una aplicación
* HTML: Es un lenguaje de marcado que se utiliza para el desarrollo de páginas de Internet. Se trata de la sigla que corresponden a HyperText Markup Language, es decir, Lenguaje de Marcas de Hipertexto.
* MSP: Proveedor de servicios gestionados (MSP, por sus siglas en inglés)

**Estándares del Proyecto**

En esta sección se definirán los siguientes estándares de diseño que ha tomado en cuenta y se aplicarán para la construcción del *software.*

**Estándares de Diseño**

Para su mejor uso, manejo y accesibilidad a la información publicada en el *software* se manejará el diseño bajo los estándares de la W3C para que el usuario tenga una mejor experiencia al momento de navegar.

**Estándares de Programación**

Para la correcta codificación del código fuente y el desarrollo del *software* se utilizará los siguientes estándares de programación:

Una buena y correcta declaración de las variables y nombres para poder recordarlas con facilidad, y así que otra persona que pueda ver el código lo pueda entender sin tantas complicaciones y pueda modificarlas.

Declaración de variables:

Para la declaración de una variable la primera palabra debe iniciar con minúscula, si la variable contiene dos palabras debe estar separadas por un guion bajo. No llevar puntos, guion y otros caracteres no mencionados.

Asignaciones:

Para asignar un valor a un variable debe haber un espacio obligatoriamente entre el operador.

Condicionales y bucles:

Las condicionales y bucles en su declaración deben presentar un espacio entre el validador o condicional.

Comentario:

Para comentario pequeños de una línea se deben de usar dos slash “//”.

Para comentarios de más de unas líneas de debe utilizar como inicial “/\*” y final “\*/”, como también sirve el “<!—“ y “-->”.

**Convenciones**

Convención de identacion:

Esta convención habla de mover una línea de texto al lado derecho insertando espacios o tabuladores, para así generar una margen con el lado izquierdo y poder distinguirlo de los textos siguientes mejorando la estructura y legibilidad del código fuente escrito.

**Diseño del Sistema**

Para el diseño del sistema se utilizó modelo de datos, diagrama de clases y diagrama de flujo.

**Diagrama de clases**

El diagrama de clases representado, contiene una estructura estática del *software* a desarrollar. Permite guiarse por sus clases y métodos principales. El diagrama No 01 muestra un modelado estructural, debido a que describe la estructura del *software*.

Dentro de esta descripción vamos a encontrar información sobre los tipos de datos, sus atributos, relaciones y sus funciones.

Las clases con el método store() en el caso de Laravel, son propios del sistema.

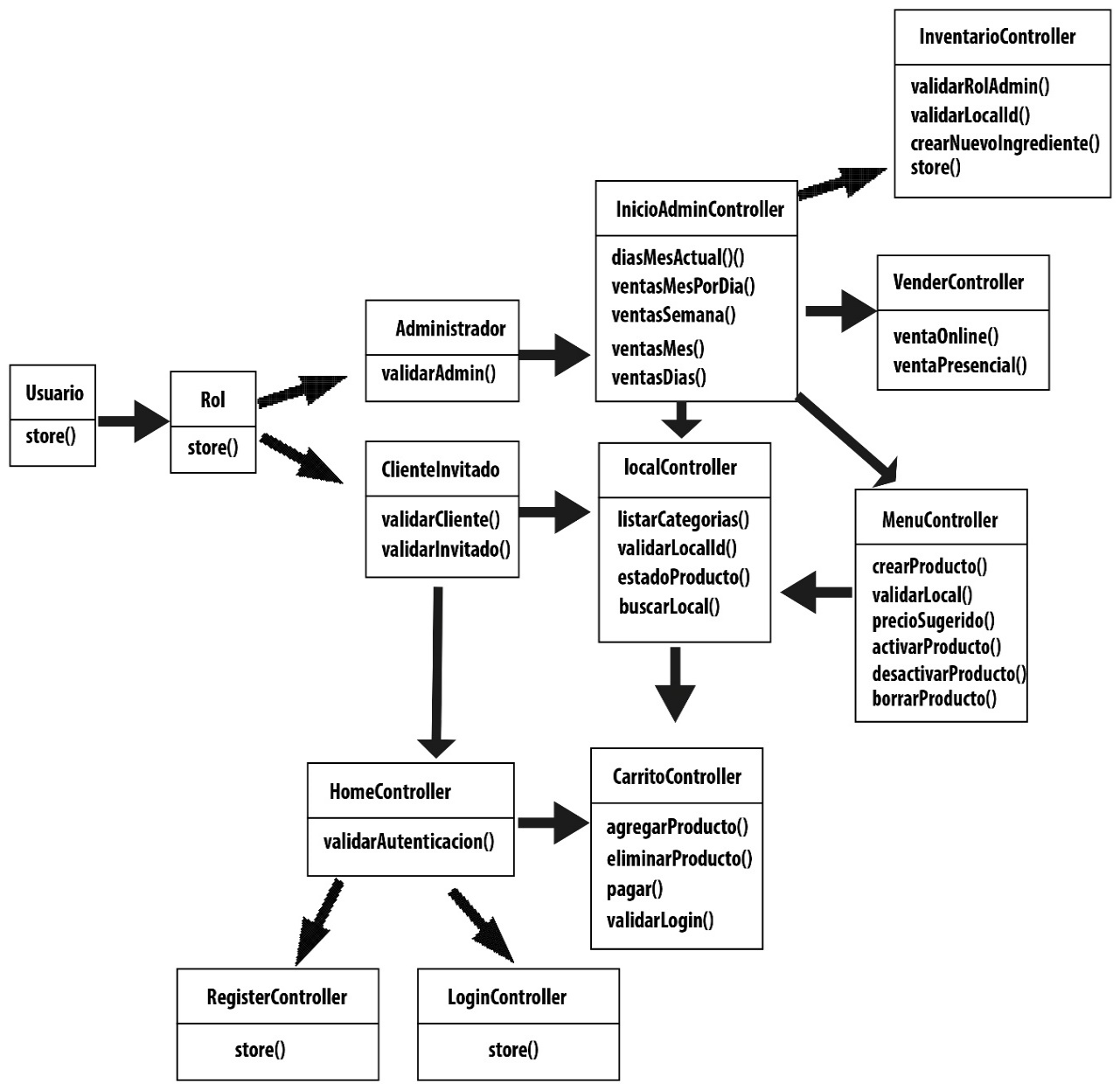


Diagrama No 01 - Diagrama de clases

**Modelo de Datos**

El modelo de datos nos define como va estar estructurado y conectada la base de datos que guardara la información del sistema.

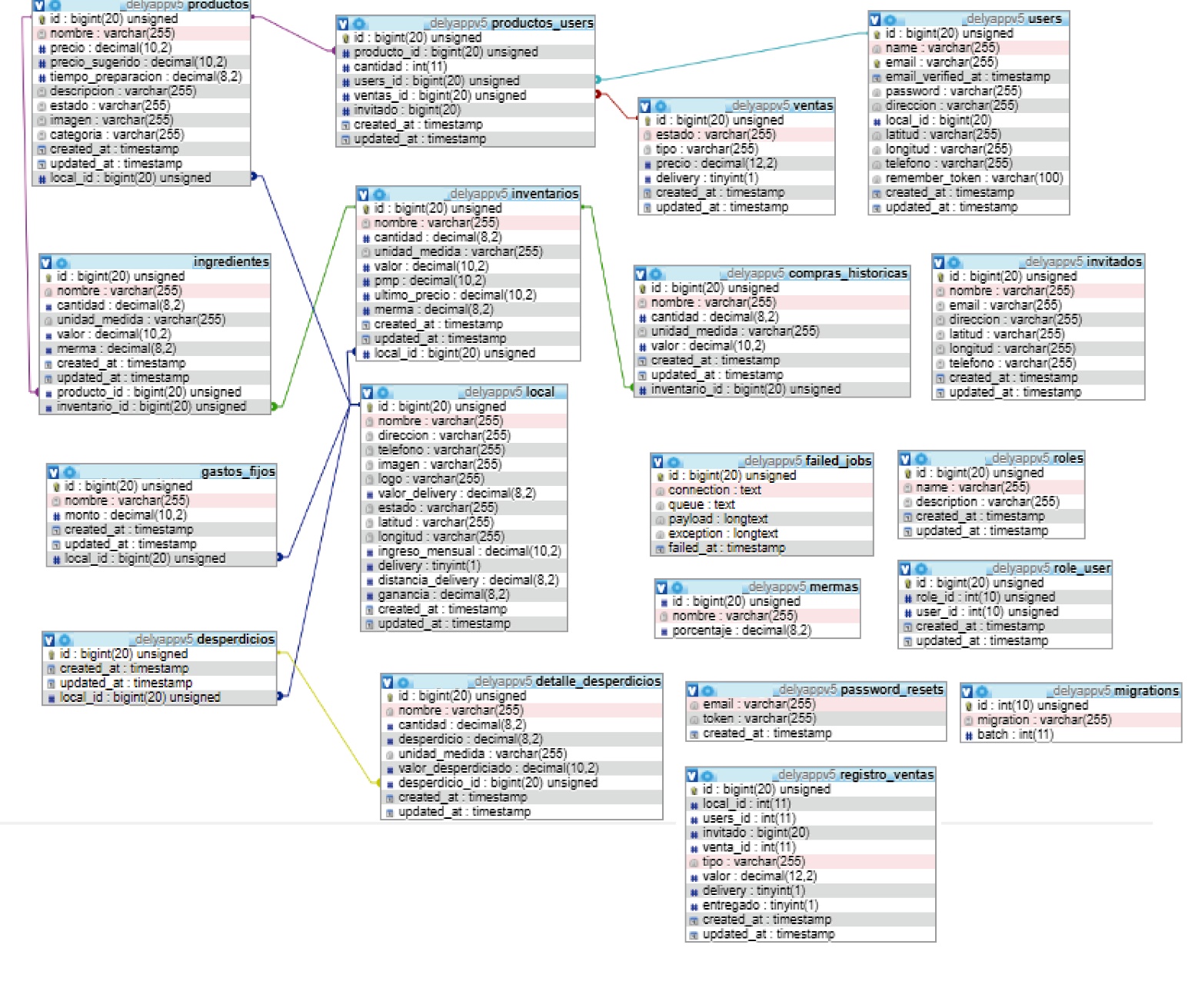


Diagrama No 02 - Modelo de datos

**Diccionario de Datos**

Dar precisión de los datos que se manejan en la base de datos del sistema y evitar malas interpretaciones o ambigüedad.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nombre: | Ingredientes |  |  |  |
| Descripción: | Guardar los ingredientes a usar para elaboración de un producto. | | | |
| Campo | Tipo | Tamaño | Descripción | |
| idingredientes | Integer | 11 | Identificador y clave de la tabla. | |
| nombre | Varchar | 45 | Nombre del ingrediente. | |
| unidad\_medida | Varchar | 45 | Tipo de medida que será ingresado al sistema | |
| cantidad | Varchar | 45 | Cantidad del ingrediente en unidad de medida señalada | |
| precio | Varchar | 45 | Precio de ingrediente ingresado | |
| productos\_idProducto | Integer | 11 |  | |
| Productos\_local\_idLocal | Integer | 11 | Identificador de local a donde pertenece el ingrediente | |
|  | | | | |
| Relaciones |  |  | Campos Claves | |
| Productos\_local\_idLocal con el campo local\_idLocal | | | idingedientes, productos\_local\_idLocal | |
|  | | |  | |
|  | | |  | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nombre: | Productos |  |  |  |
| Descripción: | Guarda los productos que se publicaran. | | | |
| Campo | Tipo | Tamaño | Descripción | |
| idProducto | Integer | 11 | Identificador y clave de la tabla. | |
| nombre | Varchar | 45 | Nombre del producto a publicar. | |
| precio | Varchar | 45 | Precio de productos a vender | |
| estado | Varchar | 45 | Estado en que se encuentra el producto | |
| local\_idLocal | Integer | 11 | Identificador de local a donde pertenece el producto | |
|  | | | | |
| Relaciones |  |  | Campos Claves | |
| local\_idLocal con el campo productos\_local\_idLocal | | | idProducto, local\_idLocal | |
|  | | |  | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nombre: | productos\_clientes | | |  |
| Descripción: | Guarda los productos que el cliente va seleccionado para su compra | | | |
| Campo | Tipo | Tamaño | Descripción | |
| idProducto\_clientes | Integer | 11 | Identificador y clave de la tabla. | |
| productos\_idProducto | Integer | 11 | Identificado del producto seleccionado | |
| Productos\_local\_idLocal | Integer | 11 |  | |
| clientes\_idClientes | Integer | 11 | Identificador del cliente. | |
| ventas\_idVentas | Integer | 11 |  | |
|  | | | | |
| Relaciones |  |  | Campos Claves | |
| clientes\_idClientes con el campo idClientes | | | productos\_idProducto, clientes\_idClientes | |
|  | | |  | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nombre: | clientes | | |  |
| Descripción: | Datos personales de los clientes | | | |
| Campo | Tipo | Tamaño | Descripción | |
| idClientes | Integer | 11 | Identificador y clave de la tabla. | |
| nombres | Varchar | 45 | Nombre del cliente | |
| correo | Varchar | 45 | Correo electrónico | |
| dirección | Varchar | 45 | Dirección del cliente | |
| Teléfono | Varchar | 45 | Teléfono de contacto del cliente | |
|  | | | | |
| Relaciones |  |  | Campos Claves | |
| idClientes con el campo clientes\_idClientes | | | idClientes | |
|  | | |  | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nombre: | ventas | | |  |
| Descripción: | Registro de las ventas realizadas por el sistema | | | |
| Campo | Tipo | Tamaño | Descripción | |
| idVentas | Integer | 11 | Identificador y clave de la tabla. | |
| estado | Varchar | 45 | Estado en que esta la venta | |
| Fecha | Date |  | Fecha que se realiza la venta | |
| Tipo | Varchar | 45 | Tipo de operación de venta | |
| Precio | Integer | 11 | El total a pagar por parte del cliente | |
|  | | | | |
| Relaciones |  |  | Campos Claves | |
| idVentas con el campo ventas\_idVentas | | | idVentas | |
|  | | |  | |
| Nombre: | gastos\_fijos | | |  |
| Descripción: | Guardara los utensilios que generan gastos fijos mensuales | | | |
| Campo | Tipo | Tamaño | Descripción | |
| Idgastos\_fijos | Integer | 11 | Identificador y clave de la tabla. | |
| nombres | Varchar | 45 | Nombre del utensilio | |
| valor | Integer | 11 | Fecha que se realiza la venta | |
| loc | Varchar | 45 | Tipo de operación de venta | |
| Precio | Integer | 11 | El total a pagar por parte del cliente | |
|  | | | | |
| Relaciones |  |  | Campos Claves | |
| idVentas con el campo ventas\_idVentas | | | idVentas | |
|  | | |  | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nombre: | Local | | |  |
| Descripción: | Guardar los datos básicos de los locales registrados | | | |
| Campo | Tipo | Tamaño | Descripción | |
| idLocal | Integer | 11 | Identificador y clave de la tabla. | |
| nombres | Varchar | 45 | Nombre del local | |
| Administrador | Varchar | 45 | Nombre del usuario quien administrara | |
| Dirección | Varchar | 45 | Dirección del local | |
| Teléfono | Varchar | 45 | Nuero telefónico del local | |
| *Delivery* | Integer | 4 | Valor numérico si ofrece *delivery* | |
|  | | | | |
| Relaciones |  |  | Campos Claves | |
| idLocal con el campo local\_idLocal | | | idLocal | |
|  | | |  | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nombre: | Inventario | | |  |
| Descripción: | Guardara los datos del inventario del local | | | |
| Campo | Tipo | Tamaño | Descripción | |
| idInventario | Integer | 11 | Identificador y clave de la tabla. | |
| nombres | Varchar | 45 | Nombre del inventario | |
| cantidad | Varchar | 45 | Cantidad del inventario | |
| unidad\_medida | Varchar | 45 | Unidad de medida que se ingresa la cantidad | |
| Valor | Decimal | 10.0 | Valor que tendrá el inventario | |
| local\_idLocal | Integer | 11 | Identificador del local | |
|  | | | | |
| Relaciones |  |  | Campos Claves | |
| local\_idLocal con el campo idLocal | | | idInventario, local\_idLocal | |
|  | | |  | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nombre: | Compras\_historicas | | |  |
| Descripción: | Guardara las compras realizadas | | | |
| Campo | Tipo | Tamaño | Descripción | |
| Idinventario\_historico | Integer | 11 | Identificador y clave de la tabla. | |
| nombres | Varchar | 45 | Nombre del dato | |
| Cantidad | Decimal | 10.0 | Cantidad de compra | |
| Unidad\_medida | Varchar | 45 | Unidad en que se mide la cantidad | |
| Valor | Decimal | 10.0 | Valor total de la compra | |
| Fecha | Date |  | Fecha donde se ejecutó la compra | |
| Inventario\_idinventario | Integer | 11 | Identificador del inventario | |
| Inventario\_local\_idLocal | Integer | 11 | Identificador del local | |
|  | | | | |
| Relaciones |  |  | Campos Claves | |
| Inventario\_idinventario con el campo idInventario  Inventario\_local\_idLocal con el campo local\_idLocal | | | Idinventario\_historico, Inventario\_idinventario, Inventario\_local\_idLocal | |
|  | | |  | |

Tabla No 01 - Diccionario de datos

**Diagrama de Proceso.**

**Proceso de Registrar Producto:**

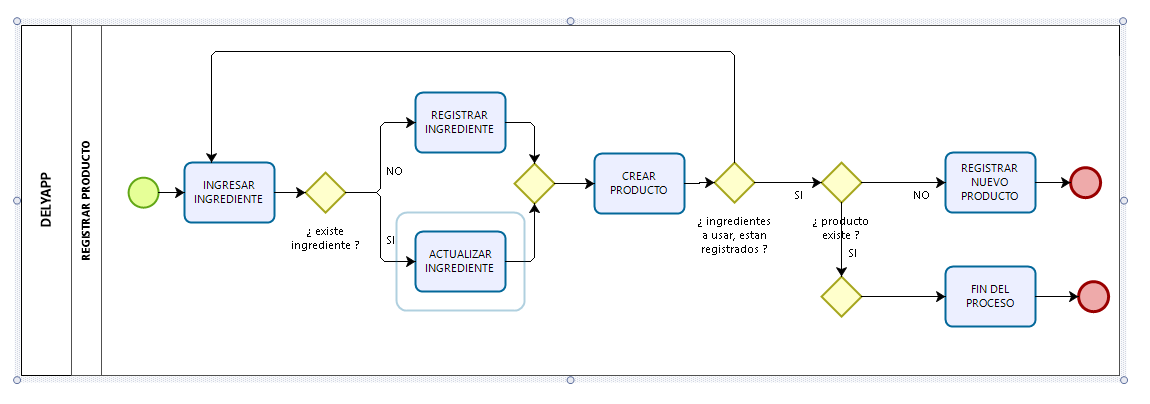


Diagrama No 03 – Diagrama de Registrar Producto

**Proceso de realizar una compra.**

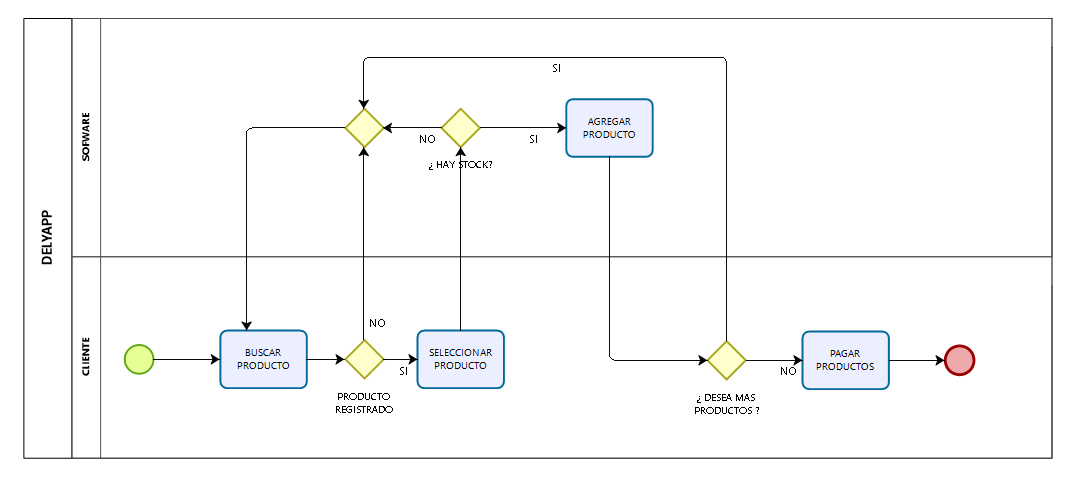


Diagrama No 04 - Diagrama de Realizar compra

**Arquitectura**

En la estrategia de arquitectura, se decidió implementar un método de gestión de recursos de TI que reemplaza las máquinas locales. La infraestructura en la nube fue la opción que se utilizó en cuestión.

La dinámica utilizada entre los desarrolladores, es que accedieron a los recursos o máquinas virtuales, con sus respectivas credenciales de acceso al CPanel para actualizar o realizar cambios al SW.

El dominio delyapp.cl se adquirió por un año en nic.cl, expira el 15/09/2021. En base a este dominio se realizó el despliegue del servicio web que implicó configuraciones DNS, virtualización, persistencia de datos entre otros aspectos que se detallan a continuación.

**Solución Inicial**

La arquitectura de hardware se definió con una infraestructura física en una empresa de servicios informáticos y data center llamada Grupo ZGH SpA en donde se almacenó y se procesó información.

Las características de hardware utilizados se mencionan en el punto 3.2.

La publicación del SW fue por medio de la interfaz de CPanel que permite realizar multitud de cosas para administrar sitios.

En este panel fue posible administrar el motor de base de datos MYSQL y los sitios web incluyendo los subdominios.

Los DNS del dominio <https://delyapp.cl/> se trabajaron en <https://nic.cl/> que en la parte de configuración técnica, en tipo de servicio servidores DNS permite alojamiento la configuración que entregó el servicio de Hosting

**Arquitectura de Hardware**

* El *Datacenter* del Grupo ZGH cuenta con 4 Enlaces de 10 Gbps por rutas físicamente distintas a los múltiples proveedores de tránsito IPv4 e IPv6 que asegura por sobre 99.85% de conectividad de primer nivel con los mejores tiempos de respuesta nacional e internacional.
* El servicio cuenta con hardware de red de calidad con *switches* independientes en cada *rack*, cableado categoría 6.
* Firewall físico configurados en alta disponibilidad capaces de procesar más de 10.000.000 de peticiones por segundo. También tiene distintas conexiones nacionales e internacionales, CDN´s y puntos de intercambio.
* El plan contratado de Web Hosting contempla *full* SSD Raid 10 lo que hace más rápido el procesamiento de información.
* UPS en cada rack.
* Respaldo energético en N+1 que significa entre otras cosas que cualquier componente sujeto a una falla es derivado en forma transparente e inmediata a otros componentes, evitando así caídas o bajas de tensión.

**Arquitectura de *Software***

En la solución inicial se expuso las herramientas de administración de dominio y DNS, en esta parte se profundizó en los conceptos de dominio y subdominios escritos bajo la virtualización aplicada en el entorno de Laravel en su versión 7 en el cual accedimos con un enlace de a través de una interfaz GUI que permitió la administración de los recursos informáticos.

El protocolo seguro de trasferencia de hipertexto (HTTP) cuenta con certificado SSL *open source* (libre uso) gracias a la colaboración de [*Let’s Encrypt*](https://letsencrypt.org/)destinado a la transferencia segura de datos de [hipertexto](https://es.wikipedia.org/wiki/Hipertexto), es decir, es la versión segura de [HTTP](https://es.wikipedia.org/wiki/HTTP).

En aquellas máquinas basadas en Linux, alojan los ambientes y las configuraciones adecuadas para el *framework* Laravel del *hosting* con las siguientes características:

* 4 Gb Espacio en Disco.
* Tráfico Mensual Ilimitado.
* Cuentas FTP Ilimitadas.
* 40 Subdominios.
* 40 Cuentas de Correo.
* 15 Base de Datos.
* 5 Dominio Adicional.
* 1 IPv6.

***Framework***

**Arquitectura de Laravel**

El marco de trabajo utilizado fue Laravel en su versión 7.x se puede observar en el diagrama n°1 la arquitectura de Laravel y el patrón arquitectónico usado con su respectivo flujo en una petición de usuario.

El lenguaje de programación basado del *framework* es PHP, dentro del árbol de directorios en el proyecto, se puede observar en el diagrama N° 2 que la mayoría de las clases u otros componentes están relacionados con este lenguaje.

Laravel tiene la ventaja de separar la aplicación en 3 capas, por lo que es un patrón de arquitectura de *software* que separa la lógica de la interfaz del usuario en tres componentes distintos:

**MVC**(Modelo-Vista-Controlador)**:**

“M” es por modelo, aquí se ocupan los campos de tablas en la base de datos como modelos para ocuparlos en los otros componentes.

“V” es por las vistas, donde la vista es la página HTML y el código que provee de datos dinámicos a la página, el modelo es la Base de Datos y la Lógica del proyecto y el controlador es el responsable de recibir los eventos de entrada desde la vista.

“C” es por controlador, donde se encuentra la mayoría de la lógica de programación e interactúa con los modelos

En el diagrama No 01 se observa el árbol de directorios que componen la aplicación:

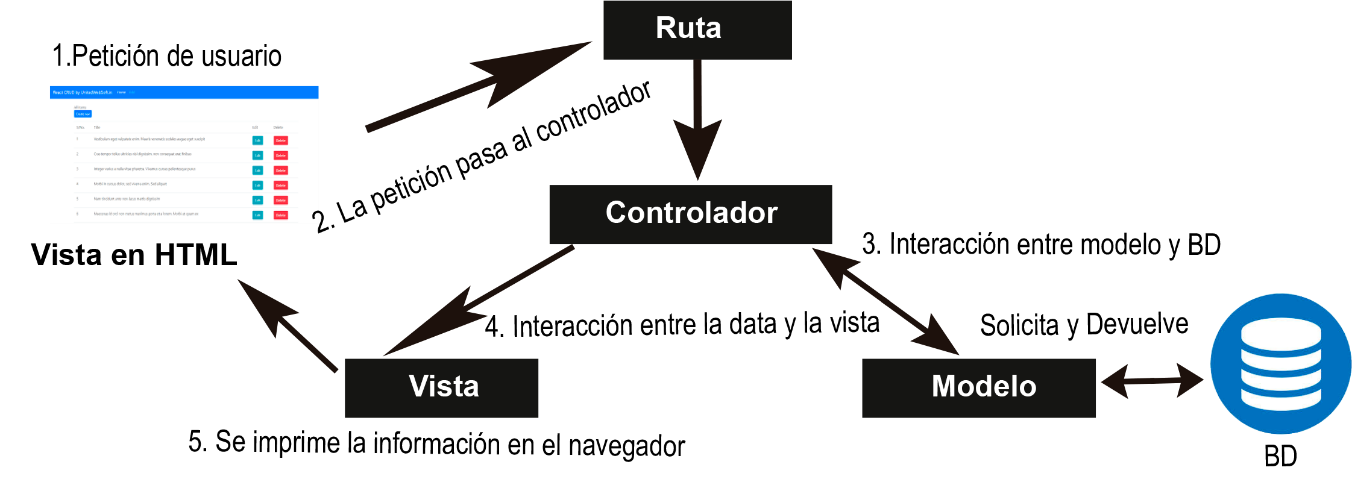


Diagrama No 05 – Diagrama de Modelo Vista Controlador

En el diagrama No 03 se observa el árbol de directorios que componen la aplicación:

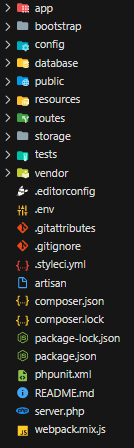


Figura No 01 Estructura aplicación Delyapp

Explicación de carperas o archivos más importantes que fueron utilizados:

* App: Aquí se encuentra gran parte de la lógica de programación aplicadas a los controladores, modelos, excepciones, proveedores de servicio, *kernel* que trabajan por parte del servidor.
* *Database*: En esta carpeta se encuentra toda la configuración relacionada a la base de datos con Eloquent que se incluye en Laravel, que ayuda a manejar procesos complejos tales como funciones que hacen consultas y peticiones en SQL de forma sencilla
* *Resources*: En este directorio se jayan las vistas de la aplicación principalmente. Estas vistas interactúan con el modelo y los controladores junto a la base de datos.
* Routes: Dentro del fichero web.php contiene las rutas de toda la aplicación, las cuales son del tipo GET o POST según el contexto aplicado a la petición del cliente.
* Vendor: Acá están todas las librerías del proyecto, muchas veces se instala con el manejador de dependencias Composer que se declaran el archivo de configuración en la raíz de la carta de proyecto que se llama package.json.
* .env: Este fichero de configuración se encuentra en la raíz del proyecto y es para colocar los parámetros tales como el nombre de la base de datos y donde apunta dicha instancia de conexión.

**Componentes de *Software***

* Sistema operativo: Ubuntu 18.04.5 LTS (GNU / Linux 4.15.0-115-generic x86\_64)
* Motor de base de datos: MySql Versión 8.0
* Manejador de dependencias PHP Composer 2.0
* *Framework* Laravel 7

**Diagrama de Despliegue**

Este diagrama explica la forma en que la aplicación web va a tomar el flujo que inicia desde una petición de usuario, hasta lo que obtiene como respuesta. En ese lapso se explica en el diagrama N° 06 cuando un usuario cliente interactúa con la interfaz, esta petición pasa del *front-end* al *back-end* (Laravel) que trabaja del lado del servidor que consulta a la base de datos y devuelve un resultado a las vistas correspondientes.

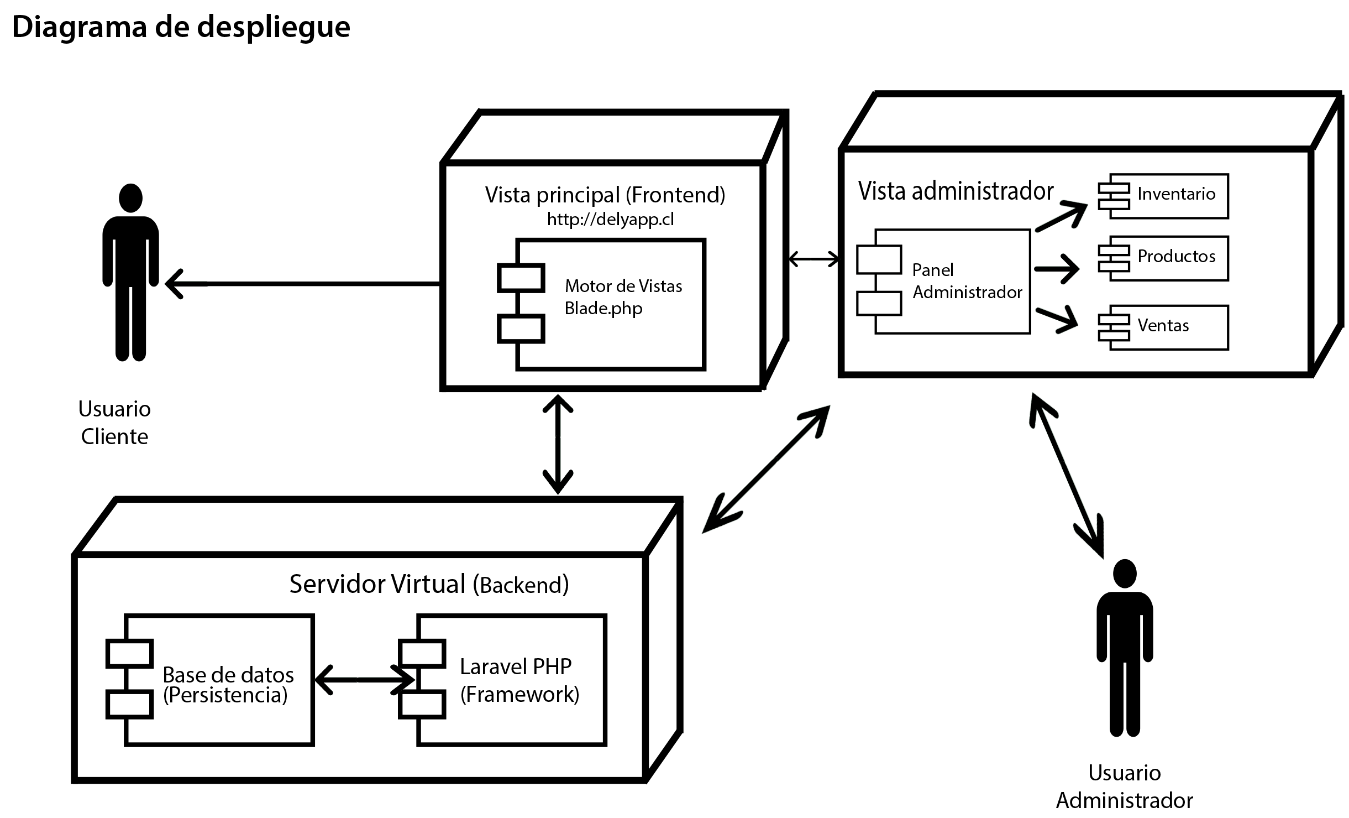


Diagrama No 06 - Diagrama de despliegue



**INSTITUTO PROFESIONAL CIISA**

**INGENIERÍA EN INFORMÁTICA**

## DOCUMENTO DE PLAN DE PRUEBAS

**DELYAPP**

**Un *software* de apoyo a la administración de local de comida rápida**

**Profesor Revisor:**

Patricio Sorich Huerta

**Alumnos:**

Nicolás González Araneda

Gabriel Aravena Rivero

Ángel Bravo Enrique

**Santiago**

**Diciembre 2020**

INDICE

[1 Historial del Versiones](#_Toc59831265)

[2 Alcance del Documento](#_Toc59831266)

[3 Información del Proyecto](#_Toc59831267)

[4 Objetivo de plan de pruebas](#_Toc59831268)

[5 Estrategia del plan de pruebas](#_Toc59831269)

[6 Ambiente de pruebas](#_Toc59831270)

[7 Casos de Prueba](#_Toc59831271)

[8 Procedimiento de resolución de fallas](#_Toc59831272)

**Historial del Versiones**

|  |  |  |  |
| --- | --- | --- | --- |
| Fecha | Versión Documento | Descripción | Responsable |
| 11-12-2020 | 1.0 | Crear documento de plan de prueba | Nicolás González |
|  |  |  |  |
|  |  |  |  |

Tabla No 01 – Historial de versiones

**Alcance del Documento**

En este documento se abarca la planificación de las pruebas que se realizaron con la finalidad de cumplir con los criterios de aceptación de un conjunto de historias de usuario que se explican en el punto 4 de los objetivos.

**Información del Proyecto**

El proyecto Delyapp, lleva por objetivo apoyar al administrador de un local de comida rápida en la gestión de su local, ayudándolo a agilizar el cálculo de ingredientes que componen un producto tales como: pollo, completos, etc.

Por otro lado, se complementa con el cálculo de margen de ganancia según indicador de PMP, está técnica aplicada, sugiere poner un precio al producto sin perder ganancias, se incluye la merma o deshecho de cada producto y precio sugerido.

**Objetivo de plan de pruebas**

Validar las historias de usuario acordadas con el cliente en cada *sprint* de modo que cumpla con el enfoque adecuado para el funcionamiento del SW teniendo como objetivo entregar el producto comprometido, tomando en cuenta que son 6 *sprints*, serán 6 casos de prueba relevantes para el funcionamiento óptimo de la aplicación web, midiendo los resultados según el criterio de aceptación en las historias de usuario registradas, corroborando así que los componentes y, por lo tanto, el producto, cumpla con los criterios de calidad propuestos por el dueño del producto.

**Estrategia del plan de pruebas**

Se aplicará en cada *sprint* un plan de prueba que pueda involucrar una o más historias de usuario que componen los requisitos del *software*.

Se medirá con el caso de prueba que se definió con el resultado esperado para cada caso, esto permite descartar errores al momento de validar el *sprint*.

Al concluir con un *sprint*, debe cumplir con el 100% del objetivo, de lo contrario, se recurre a pasar lo pendiente con compromiso de arreglar lo pendiente más la funcionalidad del siguiente *sprint*. Para revisar el detalle de pruebas realizadas, se integra a este documento la ejecución de las mismas pruebas.

**Ambiente de pruebas**

Con el fin de conseguir el funcionamiento correcto de cada caso de prueba, se caracterizan a continuación los dispositivos usados por el equipo Delyapp:

Dispositivos usados por Gabriel Aravena:

Nombre equipo :

HP ENVY 13 Laptop 13-ah0xxx

Procesador :

Intel® Core™ i7-8550U 1.80-GHz quad core processor (2400-MHz FSB, 8.0-MB L3 cache, 15.0-W)

Memoria Ram:

8GB

Disco Duro :

SSD 256GB

Sistema Operativo : Microsoft Windows 10 Home

Nombre equipo :

Samsung Galaxy Note 9.

Procesador : Snapdragon 845 de ocho núcleos (cuatro de 2.8GHz y cuatro de 1.7GHz).

Memoria Ram:

6 GB.

GPU : Mali-G72 MP18.

Almacenamiento:

128 GB.

Sistema Operativo : Android 10.

Dispositivos usados por Nicolás González:

Nombre equipo : Motorola lite 8

Procesador : Helio P35 (4x2.3GHz; 4x1.8GHz)

Memoria Ram : 4 GB

GPU : PowerVR GE8320 a 680 MHz

Almacenamiento : 64 GB

Sistema Operativo : Android 9 Pie

Nombre equipo : Notebook HP 14-cf0005la

Procesador : Intel Core i5-8250u

Memoria Ram : 8 GB

Disco Duro : HDD 1TB, SSD 265GB

Almacenamiento : 32 GB

Sistema Operativo : Microsoft Windows 10 Pro

Dispositivos usados por Ángel Bravo:

Nombre equipo: Samsung Galaxy S8

Procesador : Snapdragon 835 de ocho núcleos (cuatro de 2.35GHz y cuatro de 1.9Ghz)

Memoria Ram: 4 GB

GPU: Mali-G71 a 650 MHz

Almacenamiento: 64 GB

Sistema Operativo : Android 9 Pie

Nombre equipo: Lenovo ThinkCenter M900z

Procesador: Intel Core i5-8500u

Memoria Ram: 8 GB

Disco Duro: HDD 500GB y SSD 265GB

Sistema Operativo : Microsoft Windows 10 Pro

**Casos de Prueba**

Se seleccionaron las historias de usuario que involucra cada prueba que se ven en la tabla N°2 de las historias de usuario con su respectivo escenario y criterios de aceptación.

Los casos de pruebas definidos se encuentran en la tabla N°3 de casos de pruebas, en él se encuentran las HU (historias de usuario) que componen el caso de prueba.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Identificador de la hitoria (ID)** | **Rol** | **Funcionalidad** | **Razón** | **Número de escenario** | **Criterio de aceptación (Título)** |
| HU1 | Como cliente | Quiero tener la opción de agregar productos a un carrito de compras y pagar online y presencialmente | Para poder generar una compra | 1 | No hay stock disponible |
| 2 | Hay stock disponible |
|  |  |
|  |  |
|  |  |  |  |  |  |
| HU2 | Como cliente | Quiero tener un buscador | Para poder tener un mejor filtro de lo que deseo encontrar. | 1 | Hay coincidencias |
| 2 | No hay coincidencias |
|  |  |
|  |  |
|  |  |  |  |  |  |
| HU3 | Como cliente | Quiero solicitar *delivery* | Para no tener que ir a buscar mi pedido al local. | 1 | Solicitar *delivery* |
|  |  |
|  |  |
|  |  |
|  |  |  |  |  |  |
| HU4 | Como cliente | Quiero poder pagar a través de una tarjeta bancaria | Para hacerlo a través de internet, sin la necesidad de acercarme al local. | 1 | Cliente cuenta con dinero en su tarjeta |
| 2 | Cliente no cuenta con dinero en su tarjeta |
|  |  |
|  |  |
|  |  |  |  |  |  |
| HU5 | Como cliente | Quiero recibir un correo de confirmación de mi pedido | Para asegurar que solicité la orden correctamente. | 1 | Compra exitosa |
| 2 | Compra fallida |
|  |  |
|  |  |
|  |  |  |  |  |  |
| HU6 | Como cliente | Quiero poder visualizar el estado de mi pedido y el tiempo de despacho | Para poder organizarme con los tiempos de despacho o la demora que tomará el *delivery*. | 1 | Compra exitosa |
|  |  |
|  |  |
|  |  |
|  |  |  |  |  |  |
| HU7 | Como administrador | Quiero ingresar los ingredientes que ocupo en mi negocio, con su precio y las cantidades que tengo | Para tener un inventario | 1 | Inventario no creado |
| 2 | Inventario creado |
|  |  |
|  |  |
|  |  |  |  |  |  |
| HU8 | Como administrador | Quiero ingresar los productos que vendo, indicando la cantidad de ingredientes que ocupo en él | Para conocer la cantidad de ingredientes que gasto cada mes. | 1 | Ingresar producto |
| 2 | Eliminar producto |
| 3 | Modificar producto |
|  |  |
|  |  |  |  |  |  |
| HU9 | Como administrador | Quiero tener la opción de modificar el inventario | Para actualizar la cantidad de ingredientes cuando compre nuevamente. | 1 | Modificar cantidades |
| 2 | Crear ingrediente |
| 3 | Eliminar ingrediente |
|  |  |
|  |  |  |  |  |  |
| HU10 | Como administrador | Quiero tener la opción de auditar el inventario | Para saber cuáles han sido las pérdidas o mermas. | 1 | Inventario creado |
| 2 | Inventario no creado |
|  |  |
|  |  |
|  |  |  |  |  |  |
| HU11 | Como administrador | Quiero que al realizar una venta se actualice automáticamente el inventario | Para tener el inventario siempre actualizado. | 1 | Venta exitosa |
|  |  |
|  |  |
|  |  |
|  |  |  |  |  |  |
| HU12 | Como administrador | Quiero poder visualizar la cantidad de ingredientes que tengo disponibles | Para saber cuándo es necesario comprar ingredientes nuevamente. | 1 | Inventario creado |
| 2 | Inventario no creado |
|  |  |
|  |  |
|  |  |  |  |  |  |
| HU13 | Como administrador | Quiero poder visualizar la cantidad de ingredientes que he gastado o perdido cada mes | Para saber cómo varían estos gastos. | 1 | Auditoría realizada |
| 2 | Inventario creado sin auditoría |
| 3 | Inventario no creado |
|  |  |
|  |  |  |  |  |  |
| HU14 | Como administrador | Quiero publicar los productos que tengo en venta | Para que los clientes puedan conocer mi menú y realizar compras. | 1 | Productos ingresados |
| 2 | Productos no ingresados |
|  |  |
|  |  |
|  |  |  |  |  |  |
| HU15 | Como administrador | Quiero que cuando publique un producto me sugieran un precio (no obligatorio) | Para poder tener un margen de ganancia que permita solventar mi negocio. | 1 | Agregar producto al menú |
| 2 | Precio mínimo |
|  |  |
|  |  |
|  |  |  |  |  |  |
| HU16 | Como administrador | Quiero eliminar los productos del menú. | Para modificar el menú si es necesario. | 1 | Menú creado |
|  |  |
|  |  |
|  |  |
|  |  |  |  |  |  |
| HU17 | Como administrador | Quiero modificar los productos que tengo en venta | Para poder cambiar los precios si es necesario. | 1 | Productos agregados |
|  |  |
|  |  |
|  |  |
|  |  |  |  |  |  |
| HU18 | Como administrador | Quiero poder realizar una venta en mi local a través del *software* | Para que se registre automáticamente. | 1 | Realizar venta |
|  |  |
|  |  |
|  |  |
|  |  |  |  |  |  |
| HU19 | Como administrador | Quiero poder visualizar las ventas realizadas durante todos los meses | Para saber cómo evoluciona el negocio y tener claridad de cuánto se vende mensualmente. | 1 | Ventas realizadas |
|  |  |
|  |  |
|  |  |
|  |  |  |  |  |  |
| HU20 | Como administrador | Quiero visualizar cuánto dinero he gastado, cuánto ha ingresado y cuáles han sido las ganancias del negocio cada mes | Para saber si es necesario hacer modificaciones de precio y si las ganancias son las esperadas. | 1 | Reporte general |
|  |  |
|  |  |
|  |  |
|  |  |  |  |  |  |

Tabla No 02 – Historias de usuario

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Id Caso de Prueba | Nombre Caso de Prueba | Prueba a realizar | Resultado esperado | Historia de usuario |
| CP-01 | Pasarela de pago | Integrar Transbank como demo de pago en ambiente de integración | Al seleccionar la opción crédito que pueda pagar en forma de prueba  Con datos ficticios | HU1, HU4 |
| CP-02 | Armar producto con ingredientes disponibles en el inventario que se sumen en el precio y se descuenten del inventario | Al registrar un producto que se despliegue las opciones de ingredientes que hay disponibles para armar un producto y al dar al botón registrar que se descuenten estos ingredientes del inventario | Que, al registrar el Producto, quede en estado disponible para la venta y además que en la vista de inventario se vea que se descontó lo utilizado | HU2, HU7, HU8, HU10, HU12, HU13 |
| CP-03 | Precio sugerido en un producto | Al armar un producto, que el sistema permita sugerir precio de venta de producto | Cuando sugiera un precio de la opción de tomar la sugerencia o poner manualmente el precio final | HU15 |
| CP-04 | Crear y contar ingredientes con precio y unidad de medidas en gramos, kilos y litros | Desde el inventario se puedan crear ingredientes con su respectiva cantidad, unidad de medida y precio | Que cuando se cree se pueda usar al registrar un producto | HU8, HU10, HU12 |
| CP-05 | Implementar gráfico de ventas según mes | Que se puedan ver el gráfico de las ventas semanales y por mes | Al iniciar como administrador debe verse el gráfico en inicio | HU19, HU20 |
| CP-06 | Habilitar venta presencial | Que en el panel administrativo haya un botón que permita registrar una venta presencial por medio de la opción de pago en efectivo. | Al momento de atender cliente presencial que se desplieguen 3 opciones de medio de pago: efectivo, débito y crédito | HU1 |

Tabla No 03 – Casos de prueba

**Procedimiento de resolución de fallas**

El siguiente diagrama pretende plasmar la forma en que se resuelven ciertas fallas que puedan ir surgiendo durante el desarrollo.

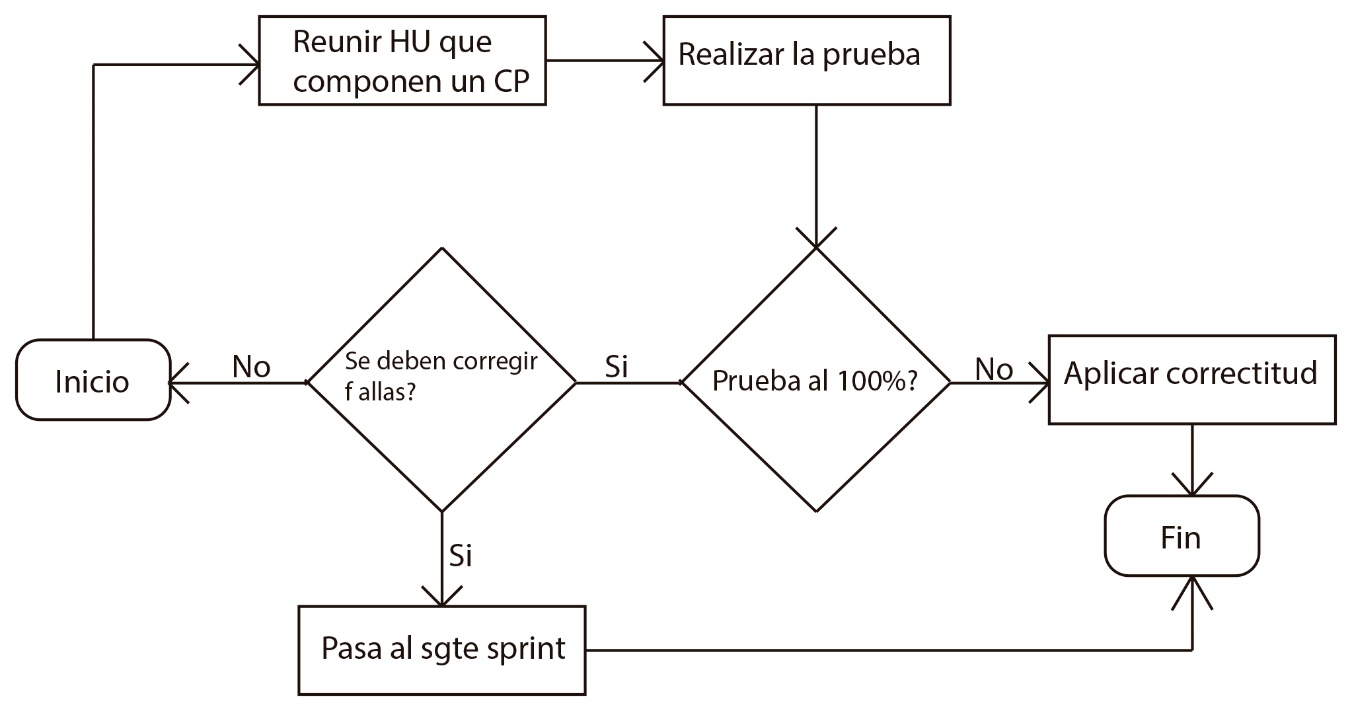


Tabla No 03 – Casos de prueba



**INSTITUTO PROFESIONAL CIISA**

**INGENIERÍA EN INFORMÁTICA**

## DOCUMENTO DE PLAN DE PRUEBAS EJECUTADO

**DELYAPP**

**Un *software* de apoyo a la administración de local de comida rápida**

**Profesor Revisor:**

Patricio Sorich Huerta

**Alumnos:**

Nicolás González Araneda

Gabriel Aravena Rivero

Ángel Bravo Enrique

**ÍNDICE**

[DOCUMENTO DE PLAN DE PRUEBAS EJECUTADO 1](#_Toc64211632)

[1. HISTORIAL DEL DOCUMENTO 3](#_Toc64211633)

[2. ALCANCE DEL DOCUMENTO 4](#_Toc64211634)

[3. INFORMACIÓN DEL PROYECTO 5](#_Toc64211635)

[4. OBJETIVO DE LA EJECUCIÓN DEL PLAN DE PRUEBAS 6](#_Toc64211636)

[5. ESTRATEGIA DE REALIZACIÓN DE PRUEBAS 6](#_Toc64211637)

[6. MATRIZ DE PRUEBAS REALIZADAS 7](#_Toc64211638)

1. **HISTORIAL DEL DOCUMENTO**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión Documento** | **Descripción** | **Responsable** |
| 12/02/2019 | 1.0 | Creación de documento de plan de pruebas ejecutado. | Nicolás González Araneda |

Tabla No 01 – Historial de versiones

1. **ALCANCE DEL DOCUMENTO**

Los resultados de casos de pruebas definidos en el documento “Plan de pruebas”, es lo que se detalla a continuación, los responsables en la ejecución de estos, son los integrantes del grupo desarrollador Delyapp que establecieron 3 iteraciones en el que cada integrante probó el caso de prueba asociado que en la tabla n°2 - Casos de prueba ejecutado, se muestra su porcentaje de aceptación.

1. **INFORMACIÓN DEL PROYECTO**

El proyecto Delyapp, basado en tecnología web, tiene como propósito apoyar al usuario administrador en el proceso de compra y venta de un producto, en el caso, comida rápida.

Ayuda en la manipulación de inventario con técnicas financieras que permita tener control y transparencia en cuanto a los ingresos y compras del local comercial en los ingredientes, mermas y gastos fijos. Además, la intención utilizada en estas técnicas, es reforzar a las Pymes con el cálculo del margen de ganancia y el precio final sugerido de cada producto.

Toda esta ayuda estará implementada dentro del *software* donde, también se contará con un sistema de venta y pago en línea (o presencial), incluyendo el despacho a domicilio (*delivery*) o retiro en local.

1. **OBJETIVO DE LA EJECUCIÓN DEL PLAN DE PRUEBAS**

El objetivo de la ejecución de plan de pruebas es completar los casos establecidos en el “Documento de plan de pruebas” con un 100% en el funcionamiento del producto en producción.

Las validaciones dependen del criterio de aceptación que viene desde las historias de usuario que se agruparon en 6 casos de pruebas ejecutado.

1. **ESTRATEGIA DE REALIZACIÓN DE PRUEBAS**

La estrategia de la realización de pruebas estuvo durante todo el proceso en que el *software* en desarrollo, se iteraron 2 veces en modo desarrollo y 1 vez en modo producción, el grupo desarrollador formado por 3 integrantes se dividieron las 3 iteraciones de la siguiente forma:

* Primera iteración: Gabriel Aravena Rivero.
* Segunda iteración: Nicolás González Araneda.
* Tercera iteración: Ángel Enrique Bravo.

La dinámica utilizada en tiempo es que duraba 1 día de iteración por caso de prueba y funcionalidad relacionada.

En pocas palabras la idea de esta estrategia es detectar fallas en cada prueba y corregirlas de forma inmediata con el fin de cumplir con el producto funcional en la fecha definida en la carta GANTT.

1. **MATRIZ DE PRUEBAS REALIZADAS**

La tabla No 02, de más abajo, representa los resultados de las pruebas realizadas por los integrantes del equipo Delyapp definidos en los puntos anteriores, se aplica un porcentaje de cumplimiento de la tarea. En el caso de no tener 100% en la iteración del otro integrante debería corregir.

Se fueron realizando pruebas a las de forma local y producción, las tres iteraciones dieron resultados satisfactorios en el momento en que se realizaron las pruebas al respecto.

Iteración 1 por Gabriel Aravena Rivero:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Id Caso de Prueba | Nombre Caso de Prueba | Prueba a realizar | Resultado esperado | Iteración N°1 | Resultado obtenido | Historia de usuario |
| CP-01 | Pasarela de pago | Integrar Transbank como demo de pago en ambiente de integración | Al seleccionar la opción crédito que pueda pagar en forma de prueba  Con datos ficticios | 50% | Se obtiene el resultado esperado | HU1, HU4 |
| CP-02 | Armar producto con ingredientes disponibles en el inventario que se sumen en el precio y se descuenten del inventario | Al registrar un producto que se despliegue las opciones de ingredientes que hay disponibles para armar un producto y al dar al botón registrar que se descuenten estos ingredientes del inventario | Que, al registrar el Producto, quede en estado disponible para la venta y además que en la vista de inventario se vea que se descontó lo utilizado | 100% | Se obtiene el resultado esperado | HU2, HU7, HU8, HU10, HU12, HU13 |
| CP-03 | Precio sugerido en un producto | Al armar un producto, que el sistema permita sugerir precio de venta de producto | Cuando sugiera un precio de la opción de tomar la sugerencia o poner manualmente el precio final | 100% | Se obtiene el resultado esperado | HU15 |
| CP-04 | Crear y contar ingredientes con precio y unidad de medidas en gramos, kilos y litros | Desde el inventario se puedan crear ingredientes con su respectiva cantidad, unidad de medida y precio | Que cuando se cree se pueda usar al registrar un producto | 100% | Se obtiene el resultado esperado | HU8, HU10, HU12 |
| CP-05 | Implementar gráfico de ventas según mes | Que se puedan ver el gráfico de las ventas semanales y por mes | Al iniciar como administrador debe verse el gráfico en inicio | 100% | Se obtiene el resultado esperado | HU19, HU20 |
| CP-06 | Habilitar venta presencial | Que en el panel administrativo haya un botón que permita registrar una venta presencial por medio de la opción de pago en efectivo. | Al momento de atender cliente presencial que se desplieguen 3 opciones de medio de pago: efectivo, débito y crédito | 100% | Se obtiene el resultado esperado | HU1 |

Iteración 2 por Nicolás González Araneda:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Id Caso de Prueba | Nombre Caso de Prueba | Prueba a realizar | Resultado esperado | Iteración N°1 | Resultado obtenido | Historia de usuario |
| CP-01 | Pasarela de pago | Integrar Transbank como demo de pago en ambiente de integración | Al seleccionar la opción crédito que pueda pagar en forma de prueba  Con datos ficticios | 100% | Se obtiene el resultado esperado | HU1, HU4 |
| CP-02 | Armar producto con ingredientes disponibles en el inventario que se sumen en el precio y se descuenten del inventario | Al registrar un producto que se despliegue las opciones de ingredientes que hay disponibles para armar un producto y al dar al botón registrar que se descuenten estos ingredientes del inventario | Que, al registrar el Producto, quede en estado disponible para la venta y además que en la vista de inventario se vea que se descontó lo utilizado | 100% | Se obtiene el resultado esperado | HU2, HU7, HU8, HU10, HU12, HU13 |
| CP-03 | Precio sugerido en un producto | Al armar un producto, que el sistema permita sugerir precio de venta de producto | Cuando sugiera un precio de la opción de tomar la sugerencia o poner manualmente el precio final | 100% | Se obtiene el resultado esperado | HU15 |
| CP-04 | Crear y contar ingredientes con precio y unidad de medidas en gramos, kilos y litros | Desde el inventario se puedan crear ingredientes con su respectiva cantidad, unidad de medida y precio | Que cuando se cree se pueda usar al registrar un producto | 100% | Se obtiene el resultado esperado | HU8, HU10, HU12 |
| CP-05 | Implementar gráfico de ventas según mes | Que se puedan ver el gráfico de las ventas semanales y por mes | Al iniciar como administrador debe verse el gráfico en inicio | 100% | Se obtiene el resultado esperado | HU19, HU20 |
| CP-06 | Habilitar venta presencial | Que en el panel administrativo haya un botón que permita registrar una venta presencial por medio de la opción de pago en efectivo. | Al momento de atender cliente presencial que se desplieguen 3 opciones de medio de pago: efectivo, débito y crédito | 100% | Se obtiene el resultado esperado | HU1 |

Iteración 3 por Ángel Bravo Enríquez:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Id Caso de Prueba | Nombre Caso de Prueba | Prueba a realizar | Resultado esperado | Iteración N°1 | Resultado obtenido | Historia de usuario |
| CP-01 | Pasarela de pago | Integrar Transbank como demo de pago en ambiente de integración | Al seleccionar la opción crédito que pueda pagar en forma de prueba  Con datos ficticios | 100% | Se obtiene el resultado esperado | HU1, HU4 |
| CP-02 | Armar producto con ingredientes disponibles en el inventario que se sumen en el precio y se descuenten del inventario | Al registrar un producto que se despliegue las opciones de ingredientes que hay disponibles para armar un producto y al dar al botón registrar que se descuenten estos ingredientes del inventario | Que, al registrar el Producto, quede en estado disponible para la venta y además que en la vista de inventario se vea que se descontó lo utilizado | 100% | Se obtiene el resultado esperado | HU2, HU7, HU8, HU10, HU12, HU13 |
| CP-03 | Precio sugerido en un producto | Al armar un producto, que el sistema permita sugerir precio de venta de producto | Cuando sugiera un precio de la opción de tomar la sugerencia o poner manualmente el precio final | 100% | Se obtiene el resultado esperado | HU15 |
| CP-04 | Crear y contar ingredientes con precio y unidad de medidas en gramos, kilos y litros | Desde el inventario se puedan crear ingredientes con su respectiva cantidad, unidad de medida y precio | Que cuando se cree se pueda usar al registrar un producto | 100% | Se obtiene el resultado esperado | HU8, HU10, HU12 |
| CP-05 | Implementar gráfico de ventas según mes | Que se puedan ver el gráfico de las ventas semanales y por mes | Al iniciar como administrador debe verse el gráfico en inicio | 100% | Se obtiene el resultado esperado | HU19, HU20 |
| CP-06 | Habilitar venta presencial | Que en el panel administrativo haya un botón que permita registrar una venta presencial por medio de la opción de pago en efectivo. | Al momento de atender cliente presencial que se desplieguen 3 opciones de medio de pago: efectivo, débito y crédito | 100% | Se obtiene el resultado esperado | HU1 |

Tabla No 02 – Casos de prueba ejecutado.