



UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

EASY DIGITAL COMMERCE:
PLATAFORMA WEB PARA PROVEEDORES Y CONSUMIDORES
GEOLOCALIZADOS

MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL EN COMPUTACIÓN

FRANCO FELIPE MIGLIORELLI MARABOLÍ

*PROFESOR GUÍA:
JÉRÉMY BARBAY*

*PROFESOR CO-GUÍA:
ANDRÉS MUÑOZ ORDENES*

*MIEMBROS DE LA COMISIÓN:
ALEJANDRO HEVIA ANGULO
JOSÉ MIGUEL PIQUER GARDNER*

*SANTIAGO DE CHILE
2022*

RESUMEN DE LA MEMORIA
PARA OPTAR AL TÍTULO DE:
Ingeniero Civil en Computación
POR: Franco Felipe Migliorelli Marabolí
FECHA: 2022
PROFESOR GUÍA: Jérémy Barbay

En los últimos años el comercio en línea ha crecido constantemente, lo que le permite a proveedores y consumidores tener un nuevo y popular canal de ventas y compras respectivamente: internet. Sin embargo, al investigar en la web y hacer una encuesta, se observó que habían algunos problemas sin resolver. Los problemas que se estudian son dos: la excesiva tardanza y costo extra de los despachos a domicilio.

Debido al límite de tiempo, este proyecto no tiene el objetivo resolver aquellos problemas, en su lugar se basa en estudiar la capacidad del prototipo para resolver los problemas planteados. Esta plataforma web tiene, entre sus características un mapa con los diversos proveedores y un buscador personalizado de bienes y proveedores. En cuanto a la reacción, esta se estudia mediante un estudio de usabilidad y la realización de una encuesta que sirve para validar las hipótesis.

El resultado fue que se detectó varios problemas de usabilidad, se notificaron varias posibles mejoras, se validó la primera hipótesis y la segunda quedó inconclusa. Respecto a las fortalezas se destaca la utilidad de la aplicación y respecto a las debilidades está el escaso atractivo que tiene. En resumen, se deduce que el prototipo demuestra tener el potencial de resolver en numerosos casos los problemas mencionados, pero que carece de características que favorecen la experiencia de usuario, lo que fija mucho por mejorar.

Tabla de Contenido

1. Introducción	1
2. Estado del arte	3
2.1. Soluciones existentes	3
2.2. Diseño	3
2.3. Desarrollo del código	4
2.4. Gestor de base de datos y conexión	5
2.5. Puesta en producción	5
3. Problema	6
3.1. Descripción	6
3.2. Relevancia	7
3.3. Requisitos	8
4. Diseño	10
4.1. Desarrollo prevalidación	10
4.1.1. Diseño de base de datos	10
4.1.2. Diseño de la interfaz gráfica	11
4.1.3. Registro de usuario e inicio de sesión	13
4.1.4. Registro de bienes	15
4.1.5. Mapa con proveedores del inicio	16
4.1.6. Búsqueda personalizada	17

4.1.7.	Solicitud de bienes	18
4.1.8.	Notificaciones	19
4.1.9.	Evaluaciones de compras	20
4.2.	Desarrollo postvalidación	21
4.2.1.	Registro de usuario	21
4.2.2.	Búsqueda personalizada	21
4.2.3.	Registro de bien	22
4.2.4.	Evaluaciones de compras	22
5.	Validación	23
5.1.	Diseño de las pruebas de validación	23
5.2.	Perfil de participantes de la primera ronda de validaciones	26
5.3.	Resultados de la primera ronda de validaciones	26
5.4.	Preparación de la segunda ronda de validaciones	29
5.5.	Perfil de participantes de la segunda ronda de validaciones	29
5.6.	Resultados de la segunda ronda de validaciones	30
6.	Conclusión	34
	Anexos	40
A.	Anexo A	40
A.1.	Diseño de base de datos antiguo	40
A.2.	Diseño final de la base de datos	41
B.	Anexo B	41
B.1.	Página de inicio	41
B.2.	Inicio de sesión	42
B.3.	Registro de usuario	42
B.4.	Registro de bien	44
B.5.	Página de inicio de usuario registrado	44

B.6.	Búsqueda personalizada	45
B.7.	Resultados de la búsqueda personalizada	45
B.8.	Proceso de solicitud de bien	46
B.9.	Ajustes	47
B.10.	Notificaciones	49
C.	Anexo C	51
C.1.	Inicio de sesión	51
C.2.	Registro	52
D.	Anexo D	54
D.1.	Formulario pre-validación	54
D.2.	Formulario post-validación	54
E.	Anexo E	55
E.1.	Búsqueda personalizada	55
E.2.	Filtros de la búsqueda personalizada	55
E.3.	Resultados de la búsqueda personalizada	58
F.	Anexo F	59
F.1.	Primer paso del proceso de solicitud	59
F.2.	Segundo paso del proceso de solicitud	60
G.	Anexo G	61
G.1.	Notificaciones	61
G.2.	Ventas aceptadas	61
H.	Anexo H	62
H.1.	Evaluaciones	62
H.2.	Evaluar calidad	62
H.3.	Evaluar puntualidad	63
H.4.	Evaluar calidad y puntualidad	63
I.	Anexo I	64

I.1.	Registro de usuario post-validación	64
I.2.	Búsqueda personalizada post-validación	67
I.3.	Registro de bien post-validación	70
I.4.	Evaluaciones de compras post-validación	71
I.5.	Proceso de solicitud de bien post-validación	73
I.6.	Tienda post-validación	74

Capítulo 1

Introducción

Actualmente Chile comparte con Uruguay el primer lugar del ranking de países con mayor cobertura a internet en Latinoamérica [1], además de que en el país aumentó el consumo de computadores [2] y celulares [3]. Aquello ha sido acompañado por un crecimiento constante del comercio vía internet, lo que nos da a conocer la continua transformación del comercio en los últimos años. Gracias a los datos de "Expande Online" [4] podemos saber detalles del crecimiento del comercio en línea, por ejemplo, el crecimiento de 2019 respecto a 2018 fue de un 20 %, el crecimiento de 2020 respecto de 2019 fue de un 55 % y en cuanto al crecimiento de 2021 se situó en un 23 % según un "Reporte de Industria" de BlackSip[26].

Además, en este año se ha facilitado el comercio presencial en comparación con el año anterior gracias a la flexibilización de las restricciones sanitarias. Aquello es desfavorable para el comercio en línea, pero según los datos de "La Tercera" [5] el crecimiento continua. Esto se puede deber a muchos factores, pero se destaca la optimización de la cadena de logística y el aumento de la calidad de la infraestructura informática (hardware y software). Por lo tanto, se espera que el comercio en línea siga creciendo a pesar de la competencia presencial.

El aumento de ventas por internet, resulta positiva tanto para proveedores como para consumidores, sin embargo un éxito tan abrumador también puede desviar la atención de problemas, produciendo que no se resuelvan. Al investigar en diversos sitios de internet, se identificó que se enumeraban varios problemas en la modalidad de compra en línea. Por ejemplo en los sitios "Guía de compras Orlando" [6] , "América Retail" [7] y "Oberlo" [8] se menciona que el plazo de entrega de un producto adquirido por internet constituye un problema, ya que muchas veces este tiempo es mayor al tiempo que tarda en llegar a nuestras manos un producto adquirido en modalidad presencial. Otro problema que se repitió fue que el costo de despacho tiende a ser alto, en sitios como "UNADE" [9] e "Incentro" [10] se hace mención de este problema.

En este escenario, surge la motivación de solucionar los problemas, pero debido al tiempo del que se dispone, no se propone una solución definitiva, sino que se desarrolla un experimento. Este consiste en el desarrollo de un prototipo de plataforma web con el potencial de ser una solución y en la realización de un proceso de validación para determinar su utilidad. Este prototipo tiene el potencial de ser una solución a los problemas planteados gracias a un grupo de funcionalidades. Aquellas funcionalidades están estrechamente relacionadas con las

hipótesis planteadas , estas son:

1. La publicación en una plataforma de la geolocalización, la zona geográfica de operación y el horario de funcionamiento de los proveedores de bienes y servicios aumenta el conocimiento de los consumidores sobre las cercanías de su domicilio, por lo cual aumenta su interés por comprar bienes y/o servicios de proveedores cercanos geográficamente.
2. La publicación de datos espacio-temporales junto con un sistema de acuerdo de horario y día de entrega con la capacidad de los clientes de calificar la puntualidad del proveedor en la entregas previas produce un descenso del costo adicional asociado al despacho y del tiempo transcurrido entre el pedido y la entrega del bien.

Para desarrollar un prototipo con estas funcionalidades se utilizan los lenguajes Python, JavaScript, CSS y el lenguaje de marcado HTML; el motor PostgreSQL con su extensión PostGis; el framework Django; las bibliotecas Bootstrap, Leaflet y JQuery. Se escoge el framework Django porque tiene un sofisticado algoritmo de encriptación de contraseñas lo que implica que es buena opción para el desarrollo de una plataforma web con registro de usuarios, como consecuencia se ocupa Python como lenguaje de programación, dado que se desarrollan algunos formularios dinámicos y se ocupa la biblioteca Leaflet, se trabaja con el lenguaje JavaScript, como se deben guardar datos espaciales se ocupa la extensión PostGis de PostgreSQL y, por último, para otorgarle estilo a este prototipo, se usa principalmente las clases de Bootstrap y en menor medida el lenguaje CSS directamente. Esta conjunción de herramientas permiten implementar todas las funcionalidades propuestas previamente, consiguiendo así el desarrollo del prototipo para su validación.

La validación consiste en la simulación de actividades comerciales por medio de esta plataforma. Los consumidores determinan si las funcionalidades en diversos escenarios de esta plataforma derivaría en un cambio de sus preferencias que como consecuencia soluciona sus problemas o si continuarían consumiendo con los medios actuales. En cambio, para los proveedores, se registra si las funcionalidades derivarían en una variación o mantención en los costos cobrados y tiempo de despacho. Los resultados generados pueden validar las hipótesis, además de indicar los puntos fuertes y débiles del prototipo.

El objetivo principal de lo descrito es poner a prueba las hipótesis, las dos deben ser aprobadas para concluir que el prototipo resuelve el problema. En caso de validar las hipótesis, sería posible ofrecer una solución, al desarrollar una versión más completa del prototipo que esté disponible a largo plazo para todo público a través de internet. Las personas que prefieren comprar y/o vender cerca de su domicilio, resultarían en mayor medida beneficiadas por una solución, debido a la mayor facilidad de entrega del bien. Además, dado que se corresponde a un experimento, entonces se tiene la garantía de obtener, como mínimo, un aprendizaje a partir de lo desarrollado. [11].

Capítulo 2

Estado del arte

En este capítulo se explican las tecnologías involucradas, tanto en este trabajo como en las soluciones existentes. En mayor detalle se explica las tecnologías utilizadas que posibilitaron el desarrollo de las etapas del proyecto y en menor detalle se muestran las características de las soluciones existentes. Se estudiaron poco las soluciones existentes, debido a su poca similitud con las tecnologías consideradas.

2.1. Soluciones existentes

Actualmente existen diversos softwares en el mercado cuyo objetivo es ahorrar tiempo y dinero y, como resultado, aumentar las ganancias de las empresas que efectúan despachos. Estos softwares se enfocan en optimizar las rutas de despacho, es decir buscan la ruta óptima desde el punto de partida al de llegada, pero dejan de lado otras soluciones alternativas, como la solución que se propone en este trabajo de título. Además de lo anterior, estos softwares tienden a contar con un sistema de geolocalización de los conductores en tiempo real, lo que muestra que las soluciones existentes utilizan soluciones de internet de las cosas, lo cual no es considerado en este trabajo.

2.2. Diseño

Para diseñar la base de datos no se ocupó un medio físico, ya que en caso de reestructuraciones es necesario borrar y escribir de nuevo, por lo que se ocupó un medio digital especializado para tal objetivo. El medio utilizado fue "Lucidchart" [27], aplicación web cuyo propósito es facilitar la construcción de diagramas, entre los cuales están los modelos de bases de datos. Esta aplicación otorga la ventaja de crear, modificar y eliminar tablas y relaciones de forma dinámica.

Para diseñar los bocetos de interfaces de usuario tampoco se utilizó un medio físico, debido a al tiempo que habría implicado, en su lugar se ocupó otro recurso digital especializado. El

programa "Balsamiq Wireframes"[28] que sirve para el diseño de interfaces de usuario para aplicaciones web y de escritorio, fue el programa de escritorio escogido para desarrollar los bocetos. La aplicación otorga la ventaja de que mediante el uso de diversos componentes disponibles, como por ejemplo botones y cuadros de texto, se pueden crear diversos tipos de interfaces.

2.3. Desarrollo del código

Para desarrollar el código se utilizó un framework, ya que uno de los objetivos de los frameworks es optimizar el tiempo de desarrollo al proveer código de base[29]. El framework escogido fue Django debido a dos virtudes: su sistema de autenticación[31] y su patrón de diseño modelo-template-vista (MTV)[30]. El sistema de autenticación otorga la ventaja de que todas las contraseñas de usuarios quedan encriptadas, lo que aumenta la seguridad del sitio. Por su parte, el patrón MTV al delegar la responsabilidad de controlar la representación de vistas a una capa escrita en python, posibilita el despliegue de la información justa y necesaria. Esta combinación (optimización de desarrollo y características de seguridad) da la suficiente confianza en este framework para el desarrollo del prototipo.

El framework Django está escrito en lenguaje Python, lo que implica que este sea el lenguaje de desarrollo de la aplicación. Python tiene diversas características que lo hacen un lenguaje popular, entre ellas está que es sencillo de aprender por su similitud con el inglés, es multipropósito y tiene muchas librerías[32]. Sin embargo, también tiene sus desventajas, por ejemplo que ocupa mucha memoria y es lento[32], pero a pesar de esto, se usa de todos modos, ya que no corresponde a un obstáculo. Por lo tanto, las ventajas resultan ser mucho más relevantes en este contexto que las desventajas, debido a que de acuerdo a los requerimientos, las desventajas no perjudican la consecución de objetivos.

Para añadir a la página de inicio un mapa con las ubicaciones de proveedores se utilizó la librería Leaflet. Se consideró sobre otras opciones como OpenLayers porque este es más sencillo de ocupar, además de que es más relevante porque lo ocupan páginas conocidas como Facebook y Pinterest[38]. Por último, debido a que no se requería más de lo especificado en la documentación de Leaflet, se consideró la mejor opción[39].

Para optimizar el desarrollo del código se utilizó un IDE, ya que estos proveen de un entorno con diversas ayudas para los desarrolladores. Ayudas entre las que se encuentran la detección de errores y la indentación automática. El escogido fue Pycharm porque este IDE, además de proveer las funcionalidades mencionadas más otras como la búsqueda inteligente, es compatible con los lenguajes utilizados: Python, JavaScript, HTML y CSS[36]. Gracias a este IDE fue posible avanzar rápido, debido a que los errores de sintaxis se detectaron inmediatamente.

2.4. Gestor de base de datos y conexión

Para almacenar los datos se utilizó el gestor de base de datos PostgreSQL con su extensión PostGis. Se escogió este gestor de base de datos, principalmente porque es uno de las más populares y soporta tipos de datos que otros no los soportan, como lo son los arreglos. Este gestor, con ayuda de extensiones, puede ampliar su funcionalidad, por ejemplo puede almacenar ubicaciones geográficas con PostGis. Su desventaja es que tiene una velocidad de lectura más lenta al compararlo con otros gestores, pero dado que la velocidad no está entre los requisitos, se utiliza de todos modos[33].

Para conectarse a la base de datos desde el servidor se escogió utilizar el ORM (Object Relational Mapping) de Django, que se define como una API que abstrae la base de datos[37]. La ventaja que otorga es que reemplaza las consultas y sentencias SQL por métodos que funcionan en código Python, consiguiendo en este caso disminuir la cantidad de líneas de código. Se escogió este método en lugar de escribir consultas y sentencias SQL debido a su fácil uso y la simplificación de código que provee.

2.5. Puesta en producción

Para realizar las validaciones con usuarios potenciales del prototipo, este se subió a un servidor para que se pueda visitar mediante internet. Para conseguirlo, se escogió a Heroku, empresa que ofrece servicios de plataforma administrada (PaaS), ya que soporta el lenguaje python y es compatible con PostgreSQL con su extensión PostGis[35]. Además de lo anterior, cabe destacar que Heroku es uno de los PaaS más populares debido a la facilidad que presenta para subir y desplegar proyectos en sus servidores lo que permite el enfoque en el desarrollo del software y no en su puesta en marcha, dando así las razones para su elección[34].

Capítulo 3

Problema

En este capítulo se explica el origen del problema, su relevancia y las características con las que debe cumplir para que puede ser considerada una solución. En síntesis comienza con una simple investigación sobre los riesgos de comprar por internet, para un par de ellos se explica su relevancia mediante una encuesta y se determinan las características de la solución. Se puede destacar que el principal desafío de este proyecto fue incluir características novedosas a una plataforma web y presentarlas en un buen formato a los usuarios.

3.1. Descripción

Para resolver, al menos parcialmente, algún problema del comercio electrónico hay dos opciones: mejorar alguna plataforma existente o desarrollar una nueva. Como no está al alcance ninguna de las dos en los meses que comprende el trabajo de título, se decide hacer un prototipo para someterlo a un estudio y así estimar su utilidad. De esta forma es posible dar el primer paso hacia la segunda opción.

Al comenzar a hacer un prototipo de plataforma web de comercio en línea es necesario tener claro cuales son las fortalezas y debilidades de las plataformas actuales. Para mejorar la experiencia de los usuarios es necesario conservar las fortalezas y disminuir las debilidades. Para saber más sobre las debilidades se hicieron dos actividades: investigar en internet y realizar una encuesta.

Los resultados de la investigación en internet arrojó diversos riesgos de comprar en internet. Por ejemplo el problema de la desconfianza de pagar por internet y la incapacidad de ver físicamente y probar el producto los expuso "Incentro" [10], "Unade" [9], "guía de compras orlando" [6] y "América retail" [7]; además de estos hay otros problemas relevantes como son la tardanza de envío de productos que no necesariamente se ajusta a los estimado [6] y el sobrecoste por despacho [10]. Dada esta información se deduce que las causas de la decepciones de la experiencia de comprar en internet están bien definidas, sin embargo este trabajo se centra en estudiar los últimos dos.

¿Un problema de comprar en línea es el costo adicional por el despacho?

70 respuestas

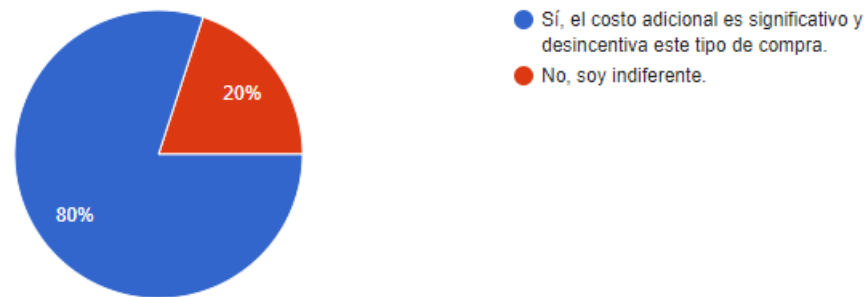


Figura 3.1: En esta imagen se muestra la distribución de las respuestas para la pregunta “¿Un problema de comprar en línea es el costo adicional por el despacho?” en un gráfico de torta

¿Un problema de comprar en línea es la tardanza de llegada de los productos?

70 respuestas

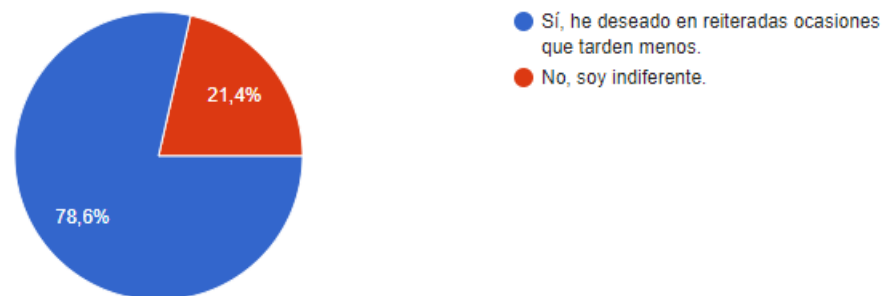


Figura 3.2: En esta imagen se muestra la distribución de las respuestas para la pregunta “¿Un problema de comprar en línea es la tardanza de llegada de los productos?” en un gráfico de torta

3.2. Relevancia

Para saber que tan relevantes son estos problemas, se hizo una encuesta mediante formularios de “Google” a 70 personas. Esta contenía cuatro preguntas, todas optativas, entre las cuales se encuentran: ¿Un problema de comprar en línea es la tardanza de llegada de los productos? y ¿Un problema de comprar en línea es el costo adicional por el despacho?. El resultado fue que el 78.6 % de los encuestados expresó su disconformidad con el tiempo de llegada a domicilio de los productos y un 80 % de disconformidad con el costo extra asociado a los despachos, por lo tanto se deduce que la tardanza y el gasto adicional de los despachos a domicilio corresponden a problemas que le afectan a una proporción importante de usuarios.

Una solución permitiría que el costo extra por despacho y su tardanza disminuyan respecto a la situación actual. La disminución de la tardanza y del sobrecoste puede ser producto de una optimización de la cadena logística o del cambio de las elecciones de los consumidores al preferir bienes y servicios más cercanos geográficamente. En este caso se estudia la

segunda posibilidad, ya que en caso contrario el desarrollo del proyecto debería enfocarse en el hardware, lo que está más relacionado con la ingeniería eléctrica, en lugar de que con la computación.

3.3. Requisitos

Dado que lo que se busca es aumentar el conocimiento de los consumidores sobre su entorno cercano, se deben desarrollar funcionalidades novedosas, ya que no es un problema que frecuentemente se desee resolver mediante el uso de softwares. Para aumentar el conocimiento de los consumidores sobre los datos espacio-temporales de los proveedores cercanos es necesario tener la capacidad de guardar este tipo de información en la base de datos. Por otro lado, dado que se debe presentar información espacio-temporal, lo cual es novedoso para un marketplace, se debe encontrar un formato que sea fácil de entender, a pesar de los escasos ejemplos. Por el tipo de datos a guardar y mostrar, se observa que tanto a nivel de backend como de frontend que hay un desafío importante.

Este proyecto de software tiene diversos requerimientos funcionales y no funcionales. El objetivo de los requerimientos funcionales, en este caso, es permitirle a los usuarios interactuar con un prototipo de marketplace que considera datos espacio temporales. En cambio, el objetivo de los requerimientos no funcionales es que esta experiencia no sea tediosa para el usuario.

Requerimientos funcionales:

1. Solicitar bien.
2. Calificar calidad del bien.
3. Crear cuenta.
4. Iniciar sesión.
5. Buscar por precio y/o calidad.
6. Calificar puntualidad del proveedor.
7. Buscar por datos espacio temporales.
8. Visualizar mapa con proveedores.
9. Visualizar notificaciones de solicitudes de compras y ventas.

Requerimientos no funcionales:

1. Tamaño de letras adecuado.
2. Contraste entre componentes adecuado.
3. Rapidez en la presentación de resultados de consultas.

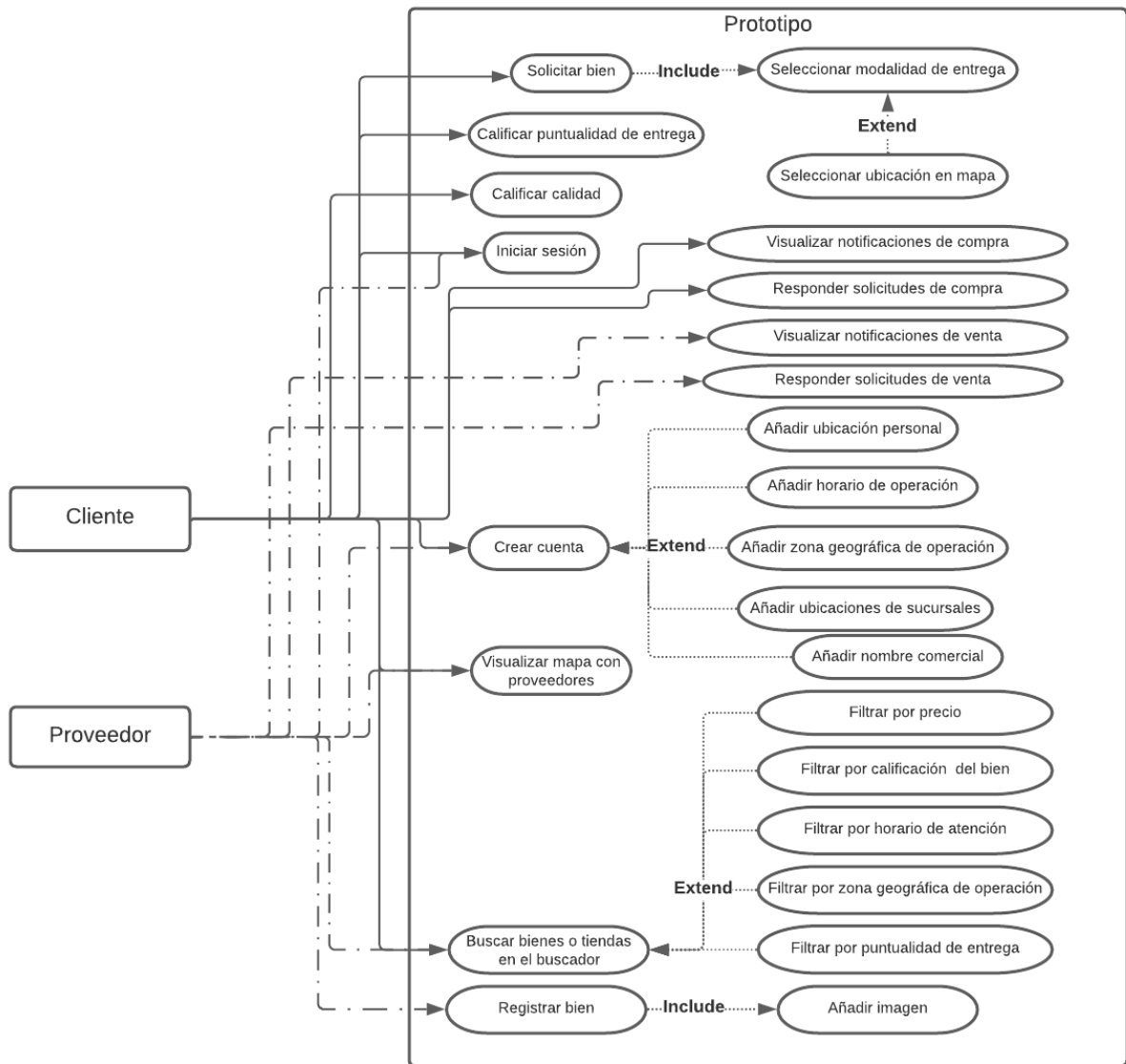


Figura 3.3: Diagrama de casos de uso

Capítulo 4

Diseño

El desarrollo se divide en dos partes: desarrollo prevalidación y postvalidación. El hito que marca la división es la validación, de la cual se anotaron muchos puntos a mejorar, de estos se tomaron en cuenta un reducido grupo, debido al límite de tiempo. No se toman todos los puntos en consideración debido al límite de tiempo fijado.

4.1. Desarrollo prevalidación

Este desarrollo se guía por el plan propuesto previamente. Comprende desde el diseño de la base de datos hasta la elaboración del sistema de puntuaciones de calidad y puntualidad de llegada de productos y servicios. Cabe destacar que la etapa del diseño de la base de datos, a pesar de ser transversal durante el desarrollo, se explica en una sola sección.

4.1.1. Diseño de base de datos

Para comenzar, se diseña una base de datos que se compone de tan sólo 5 tablas, pero luego de analizarse, se determina que no corresponde a un buen diseño (desde la página 40 se puede observar y descargar el diseño completo). Habría funcionado para cumplir con el objetivo, pero dado que se plantea que es deseable el desarrollo posterior, entonces se modifica para que sea más flexible ante los cambios. Las falencias se enumeran a continuación: Falencias:

1. Por cada usuario se considera el atributo "nombre comercial", mientras este puede ser o no ser proveedor, lo que le puede otorgar a esta entidad un atributo, la mayoría de las veces, inútil.
2. Por cada usuario, se considera la latitud y la longitud, lo cual le impide a los proveedores añadir más de un local comercial.
3. Por cada usuario se considera el alcance y sólo los proveedores publican dichos datos, por lo que por cada cliente se reserva espacio extra.

4. Por cada usuario se considera la cantidad de veces evaluado y la acumulación de estrellas, lo que sólo les sirve a los usuarios proveedores, por lo tanto es una reserva de memoria innecesaria, además de que impide conocer detalles de las evaluaciones, por ejemplo las ventas evaluadas.
5. Se hacen tablas separadas de "productos" y "servicios" con los mismos atributos, lo que implica que también la tabla "ventas" se deba dividir en ventas de productos y ventas de servicios.

Por las numerosas falencias expuestas, se modifica el modelo de base de datos a medida que se desarrolla el código, hasta llegar al modelo final, que consiste en 12 tablas. Este modelo no condensa en una entidad dos o más entidades, lo que permite que los cambios en la plataforma sean más fáciles (En la subsección "Diseño final de la base de datos" del Anexo A se puede observar y descargar el diseño completo). Por ejemplo, si se quiere añadir la funcionalidad "sumar nueva ubicación a la tienda" con el modelo antiguo, es necesario transformar los atributos latitud y longitud para que guarden arreglos de flotantes, en lugar de sólo un flotante, y luego copiar los datos existentes desde el viejo modelo al nuevo. En cambio, añadir un nuevo punto con el nuevo modelo, consiste en sumar un nuevo punto al objeto MultiPoint de la extensión PostGis, lo que consta de menos pasos. A continuación se enumeran las mejoras en el modelo de base de datos respecto al antiguo.

Mejoras:

1. Se crea la entidad "Shop" la cual agrupa los atributos exclusivos de las tiendas, por ejemplo su alcance geográfico, horario de atención y ubicaciones.
2. Se crea la tabla "Sales" cuyos atributos permiten identificar de forma correcta cada venta.
3. Se crea la tabla "Score" la cual guarda en cada registro la puntuación y el tipo de puntuación (calidad o puntualidad) que se asocia a una venta respectiva.
4. Se crean las tablas "TypeGood", "Subcategory" y "UnitMeasurement" las cuales permiten agrupar en diversos grupos los bienes que se suben a la plataforma y gracias a ello hacer búsquedas personalizadas mediante estas divisiones.
5. Se crea la tabla "DispatchHelp" la cual sirve para asociar a cada venta con un texto de ayuda para el despacho.
6. Se crea la tabla "Phase" la cual permite diferenciar a las ventas según su fase, las cuales pueden ser "venta aceptada", "venta rechazada", "venta con propuesta pendiente" y "solicitud de venta pendiente".

4.1.2. Diseño de la interfaz gráfica

Al tener el borrador de base de datos, se procede a realizar bocetos de la interfaz gráfica de usuario y a pedir la opinión de algunos potenciales usuarios (Los bocetos se pueden

observar en las diversas subsecciones del Anexo B). Previo al comienzo del proyecto también se realiza un procedimiento similar, la diferencia es que no contemplan todas las vistas y que no se solicita ninguna opinión. Al tener retroalimentación de potenciales usuarios es posible añadir, modificar y descartar características de las interfaces en una fase temprana. El beneficio es un menor tiempo invertido en la modificación de interfaces, ya que modificar bocetos es más rápido, al menos en este caso, que modificar el código fuente.

Se desarrollan 74 vistas de usuarios que representan diversos estados del prototipo, las cuales se dividen en los siguientes grupos: página de inicio, inicio de sesión, registro de usuario, pagina de inicio de usuario registrado, ventana de notificaciones, búsqueda personalizada, registro de nuevo bien, proceso de compra y ajustes. Se consulta a dos usuarios potenciales (uno vende mediante internet y el otro compra y vende mediante internet) y se detectan una serie de falencias que afectan tanto a la estética como a la usabilidad del prototipo. Aquellas fallas se pueden resumir en los siguientes puntos:

1. No es atractiva ni tiene un propósito claro.
2. La página no indica a simple vista como añadir un bien.
3. La funcionalidad "trazar polígonos" es innecesaria, ya que los despachos tienden a abarcar comunas enteras.

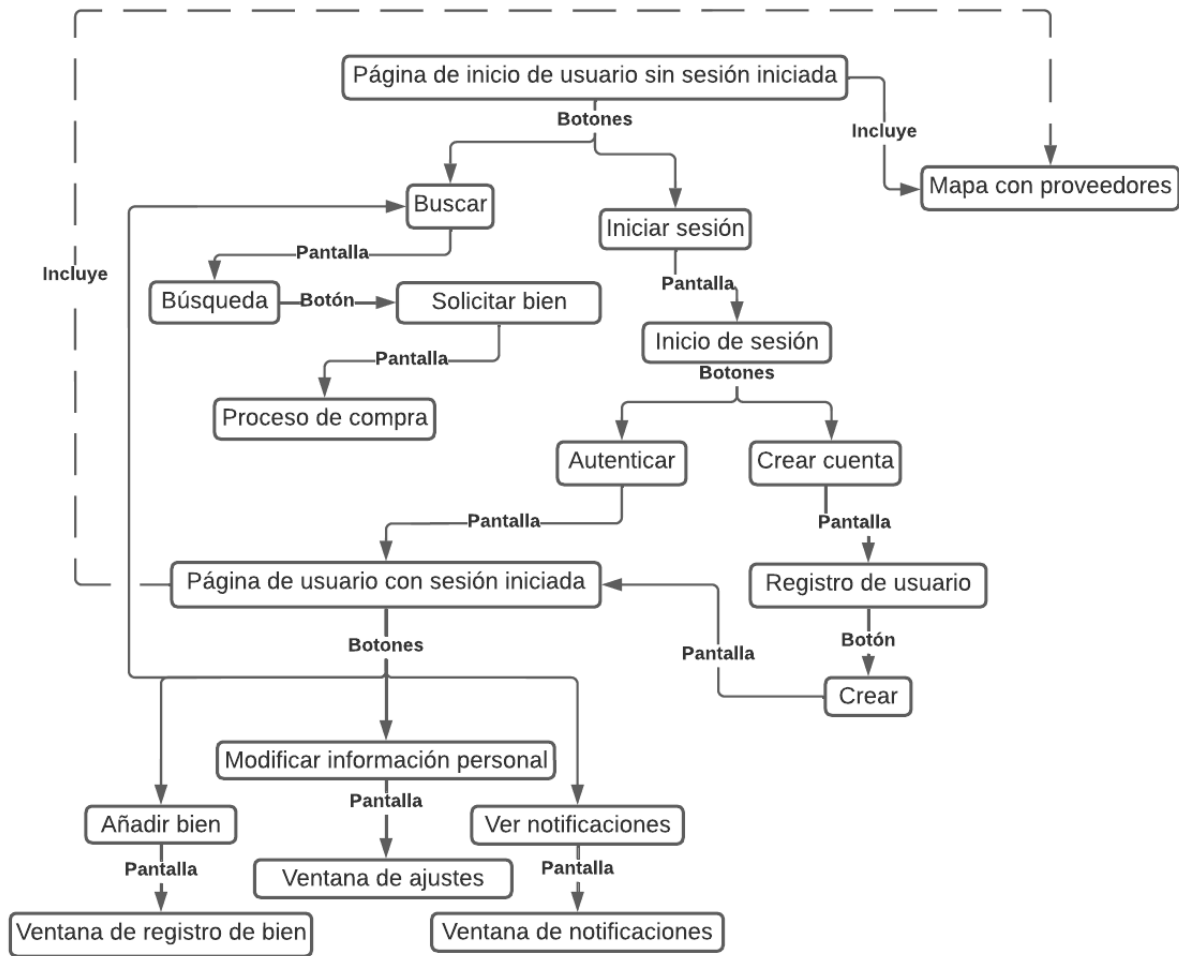


Figura 4.1: Arquitectura de la información de la interfaz evaluada

4.1.3. Registro de usuario e inicio de sesión

Al tener lista la idea sobre el aspecto que debe tener la interfaz de usuario, se procede a programar el código. Para poder crear a los usuarios del tipo cliente y proveedor, se crea una aplicación con las entidades necesarias de base de datos y se modifica la configuración original del proyecto. En la nueva aplicación se crean las clases "UserCustomer", "UserManager", "Shop" y "TypeUser", las cuales en conjunto permiten la creación de dos tipos de usuarios, administrador y externo. Su clave primaria es el correo electrónico y otros diversos atributos, por ejemplo el nombre y número de contacto, que son relevantes en el comercio por internet. La mayoría de los atributos se guardan en campos de texto y de enteros, y una minoría se guardan en campos menos comunes, como los son los arreglos para guardar los id's de los diversos tipos de usuario de cada usuario y los campos del tipo Point de la extensión PostGis en los que se guardan las ubicaciones tanto para usuarios como para tiendas. Para que el sistema de autenticación original de usuario funcione para un tipo de usuario personalizado, en el archivo "settings" se añade la variable "AUTH_USER_MODEL" que almacena la ruta hacia el nuevo modelo de usuario.

Para inicializar sesión se ocupan dos vistas distintas: la vista de inicio de sesión y la vista de registro de usuario, la cual al registrar un usuario, inicializa sesión automáticamente (se muestran en el Anexo C). En ambas vistas se ocupa la librería de Django "Crispy forms", ya que esta le otorga estilo de forma automática a los formularios de Django. La vista de inicio de sesión sólo tiene dos campos para escribir las credenciales correspondientes, en cambio la vista de registro tiene más campos para colocar la información personal y es interactiva. Es decir, tiene la capacidad de mostrar más preguntas como consecuencia de las respuestas recibidas, lo que tiene el objetivo de mostrarle a cada usuario las preguntas justas y necesarias. A continuación se detallan los campos del formulario de registro, desde el primero hasta el último, y sus respectivas características interactivas:

1. En el primer campo se puede colocar uno o más nombres.
2. En el segundo campo se puede colocar uno o ambos apellidos.
3. En el tercer campo se coloca el correo personal que sirve como clave primaria.
4. En el cuarto campo se coloca el número de contacto.
5. En el quinto campo se consulta si se desea colocar la ubicación personal. En caso de tener una respuesta afirmativa se muestra un mapa en pantalla para indicar con un marcador dicha ubicación.
6. En el sexto campo se consulta la condición del usuario, es decir, si quiere utilizar la plataforma para consumir y/o proveer. En caso de identificarse como proveedor, se presentan en pantalla cuatro nuevos campos, los cuales se describen a continuación:
 - (a) El primer campo consulta acerca de la existencia del nombre comercial. En caso de existir, presenta un campo para añadirlo.
 - (b) El segundo campo consulta acerca de la equivalencia entre ubicación personal y ubicación comercial. Al contestarse se consulta sobre si se desea añadir más u otras ubicaciones comerciales, dependiendo de la respuesta recibida.
 - (c) El tercer campo consulta acerca si desea añadir una zona de despachos, que consiste en un conjunto de comunas en las que se despacha. Al contestarse de forma afirmativa, se consulta sobre las comunas del país que comprenden dicha zona. Para añadir una comuna a la zona de alcance, se marca su casilla respectiva en una tabla. En caso de querer marcar todas las comunas que pertenecen a una región o provincia, se marca la casilla correspondiente a la región o provincia respectiva.
 - (d) El cuarto campo consulta sobre el horario de atención. En caso de contestarse afirmativamente, se consulta en que tipo de estructura se desea colocar el horario, las opciones son: tabla estática y tabla dinámica. La tabla estática consiste en una tabla con una columna con diversos intervalos horarios, una fila con diversos días y por cada intersección de sus proyecciones, hay una casilla, las cuales sirven para indicar la disponibilidad en cierto intervalo horario. La tabla dinámica consiste en dos campos en los que se indica manualmente los intervalos de inicio y fin de la jornada para un conjunto determinado de días, se le denomina dinámica porque con la ayuda de botones se pueden crear nuevos campos para indicar

nuevos intervalos e incluso añadir nuevas tablas dinámicas para añadir intervalos para otros conjuntos de días.

7. El penúltimo campo corresponde a la creación de contraseñas y el último a su confirmación.

Para posibilitar la creación de un formulario dinámico con la capacidad de desplegar diversas tablas y mapas, se utilizó JavaScript y Leaflet. Con el lenguaje JavaScript se le otorgó dinamismo al documento HTML, lo cual se logra escribiendo funciones que modifican la estructura del archivo HTML. Mediante la librería Leaflet fue posible colocar en el formulario mapas, los cuales se encapsularon en elementos HTML llamados Iframe, lo que permitió identificarlos con distintos id's.

4.1.4. Registro de bienes

Al tener a los usuarios registrados, se procede a desarrollar la funcionalidad de publicar bienes y servicios. En la propuesta se indica que esta etapa es posterior, pero se adelanta debido a que se decide por publicar en el mapa sólo a los usuarios de los tipos "proveedor" y "cliente-proveedor" con al menos un bien publicado. Esto debido a que no existe una razón para publicar a los usuarios del tipo "cliente", ya que no se contempla la posibilidad de contactar directamente a aquellos desde otra cuenta. Para desarrollar esta funcionalidad, principalmente se crea una nueva aplicación en el proyecto y se utiliza una nueva forma de representar formularios.

Al registro de bien se accede mediante un SVG (elemento HTML) que representa un signo de suma, el cual está en el header común a todas las vistas. Cabe destacar que a todas las funcionalidades se accede mediante botones en formato SVG a excepción del registro de usuario, que se accede mediante un enlace. En esta vista el formulario se presenta en una grilla, en la cual pueden coexistir más de un campo en una misma fila, este formato se extrae de Bootstrap. Para que los elementos HTML "select" cambien su contenido en función de las selecciones previas de otros "select" relacionados, se ocupa la librería de JQuery Chained. Además, se ocupa código JavaScript de la documentación de Bootstrap para hacer un tipo de validación personalizada (las vistas se muestran en el Anexo D).

En el formulario se consulta por nombre, precio, imagen, tipo de bien, categoría, subcategoría, disponibilidad, unidad de medida, su capacidad para ser dividido y su descripción. Todos los campos son obligatorios a excepción de la descripción. Para almacenar esta información en la base de datos se crean las entidades: "TypeGood" para guardar el tipo de bien del bien, "Category" para guardar la categoría del bien, "Subcategory" para guardar la subcategoría del bien, "UnitMeasurement" para guardar la unidad de medida del bien y "Good" que guarda la información específica del bien, como lo son el nombre, el precio, la subcategoría, la tienda a la que pertenece, la imagen, la unidad de medida, la descripción, su disponibilidad y su capacidad de ser dividido. Gracias a las relaciones presentes entre las diversas entidades es posible que en el formulario dinámico se puedan seleccionar sólo las categorías pertenecientes su tipo de bien respectivo, las subcategorías pertenecientes a su categoría respectiva y las unidades de medida disponibles para cada categoría.

4.1.5. Mapa con proveedores del inicio

La capacidad de guardar usuarios y bienes mediante la interfaz provee lo necesario para desarrollar el mapa que muestra las ubicaciones de los diversos proveedores en la página de inicio. En detalle, consiste en que se muestran las localizaciones que se entregan en el registro por los usuarios de tipo "proveedor" y "cliente-proveedor" que tienen al menos un bien disponible. Aquello se logra con la librería Leaflet, el lenguaje JavaScript y los iconos extraídos de Icons8.

En la página de inicio se despliegan dos tipos de informaciones: el mapa que contiene las ubicaciones de los proveedores y los últimos productos subidos a la plataforma. Sin embargo sólo se explica el primer tipo de información, ya que el segundo corresponde a un desarrollo no presente anticipadamente en la propuesta. Para localizar a los proveedores primero en el archivo "views" se consultan las ubicaciones, nombres comerciales y subcategorías de cada proveedor, con al menos un bien disponible, en la base de datos y se escriben en formato JSON con una estructura específica definida en la documentación de Leaflet, el cual se envía al template mediante un contexto de Django. En este caso se utiliza ese tipo de estructura para colocar marcadores, ya que nos permite presentarlos con iconos personalizados en el mapa. Dichos marcadores se enlazan con un "pop up" para que, al ser tocados, se presente un recuadro con el nombre de la tienda (en caso de que no se tenga, se muestran los nombres de los integrantes), las subcategorías de los productos y/o servicios que ofrece y, por último, un botón con la etiqueta "Visitar tienda" para conocer los productos y/o servicios que ofrece la tienda.

Al tener un mapa con los diversos proveedores distribuidos se puede generar una sobrexposición de información, ya que el usuario puede desear ver sólo una fracción de los proveedores. Para resolver lo anterior se desarrolla un filtro que esconde y muestra marcadores, lo que se logra con la documentación de Leaflet. Para conseguir esto, el template examina la información recibida desde el servidor con JavaScript para separar los proveedores en tres grupos, entre los que ofrecen productos, servicios y ambos. Luego se almacenan los marcadores de los tres grupos distintos en tres arreglos distintos, los cuales cada uno crean su "layerGroup", los que se añaden al mapa asociados con una etiqueta. El resultado es un mapa con el cual para ver algún tipo de proveedor, es necesario seleccionar ese tipo de proveedor en el filtro que se muestra en la esquina superior derecha del mapa.

Plataforma de geolocalización de clientes y vendedores



Búsqueda personalizada



Iniciar sesión

¡Explora el mapa para encontrar proveedores de diversos bienes!

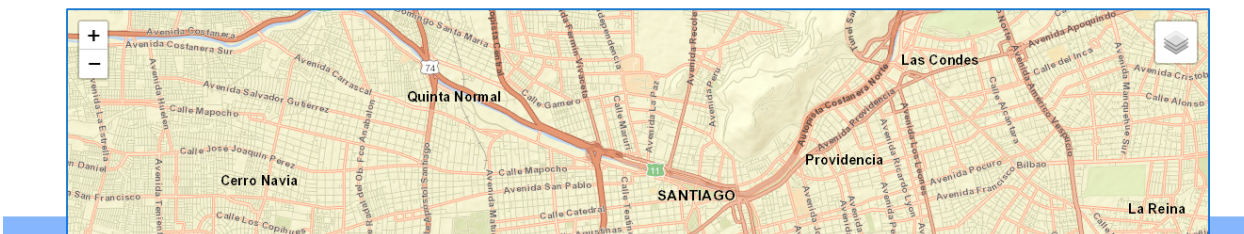


Figura 4.2: Página de inicio

4.1.6. Búsqueda personalizada

Dado que en este punto la única forma de ver los bienes subidos es mediante el mapa y el listado de últimos productos subidos, se procede a desarrollar la búsqueda personalizada. La búsqueda personalizada consiste en la presentación de bienes y tiendas que cumplan con los criterios establecidos en un formulario (las vistas están en el Anexo E). Se pueden presentar en dos formatos: un mapa con los proveedores que tienen al menos un bien ofertado o una lista de bienes que cumplen con los criterios respectivos. En caso de no aplicar los filtros, se muestra una lista de todos los bienes subidos al prototipo. Ambos formatos de presentación tienen sus ventajas y desventajas: el mapa otorga la ventaja de conocer inmediatamente la ubicación del proveedor y la desventaja de que no es posible solicitar bienes inmediatamente, ya que primero se debe visitar la tienda. Por otro lado la lista tiene la ventaja de que permite solicitar un bien inmediatamente y la desventaja de que no muestra información del proveedor.

Para utilizar los filtros se ocupan switch's que presentan y borran campos del formulario, por ejemplo al activar el filtro "tipo de bien" se consulta el tipo de bien a mostrar (producto o servicio) y se muestra un nuevo "switch" que al activarse permite escoger la categoría del tipo de bien a mostrar, por lo tanto se trata de un formulario dinámico, por lo que se hace uso de ajax para que al hacer una búsqueda, no se borren los filtros seleccionados. Luego de que el formulario envía una petición ajax, el servidor consulta el tipo de petición y al resultar ser del tipo "XMLHttpRequest", procede a generar un diccionario de Python, con una llave que contiene un arreglo de objetos JSON's, con la información solicitada en la búsqueda, el cual se envía en un JsonResponse importado desde "django.http". Esta información es recibida por un script del documento HTML, el cual la transforma en un objeto JSON para extraer fácilmente la información y escribirla en el documento HTML.

El servidor, para responder a la solicitud recibida, consulta a la base de datos sobre los

bienes y/o proveedores que cumplen con los requisitos solicitados, lo que se hace mediante una serie de pasos:

1. Se almacena en dos variables una lista de objetos cada una. Una almacena una lista de bienes y la otra la lista de tiendas presentes en la base de datos.
2. En caso de que se filtre por tipo de bien, categoría o subcategoría, de la variable de bienes se descartan aquellos bienes que no cumplen con lo solicitado.
3. En caso de que se filtre por precio, de la variable de bienes se descartan los bienes cuyo precio no esté en el rango especificado.
4. En caso de que se filtre por puntualidad del proveedor o calidad del bien, se guarda en dos variables una cantidad entera cada una, en una la mínima y en la otra la máxima cantidad de estrellas, lo que sirve para discriminar los objetos a usar para escribir los JSON's que se añaden al arreglo.
5. En caso de que se filtre por horario de atención, se descartan las tiendas junto con sus bienes que tengan un horario que no abarque a lo solicitado. Por ejemplo si se solicita filtrar las tiendas que atiendan los martes desde 10 a.m a 11 a.m, entonces se descartan todas las tiendas junto con sus bienes que tengan un horario que no permite asegurar la atención en el intervalo solicitado, como lo es desde 10 a.m hasta 10:30 a.m.
6. En el caso en el que se filtre por alcance geográfico, el usuario debe indicar el punto en el mapa en el que desea el despacho, ya que de esta manera se descartan las tiendas junto con sus bienes que no declaran tener despachos a la comuna a la cual pertenece dicho punto.

El punto 6 que explica el filtro por alcance geográfico es más complejo que el resto, por lo que se explica con mayor detalle a continuación. Para lograr determinar a que comuna corresponde cada par posible de latitud y longitud se siguen una serie de pasos. Primero se crea la entidad "CommunesBorders" y se transfieren a esta los datos del archivo con extensión "shp" que procede del mapa vectorial descargado de la biblioteca del congreso nacional de Chile. Lo anterior permite que en la base de datos se tenga por cada comuna una serie de vértices que conforman un polígono, logrando así delimitar cada comuna. Con esto es posible consultar a qué comuna pertenece cada par latitud, longitud del país, transformar el resultado y averiguar si se encuentra entre los alcances de la tienda almacenados. Es necesaria la transformación, ya que el atributo alcance de la tienda corresponde a un arreglo de enteros, en el que cada entero representa a una comuna.

4.1.7. Solicitud de bienes

Para que los usuarios tengan la capacidad de solicitar bienes a través del prototipo se crea una aplicación llamada "ventas". Esta aplicación crea las vistas del proceso de solicitud (las que están en el Anexo F), estas despliegan información detallada sobre el producto y el proveedor y presenta los campos para ordenar una compra. El proceso de compra consiste

en dos fases, la primera fase consiste en mostrar la información relevante concerniente al producto y al proveedor. En lo que respecta al producto, se indica su nombre, descripción, el precio por unidad o masa u otra unidad de medida, si se puede obtener una fracción y tanto el promedio como la cantidad de veces evaluada su calidad. Por otro lado, en la información del proveedor se muestra el nombre de la tienda (en caso de tener), sus integrantes, tanto el promedio como la cantidad de veces evaluada su puntualidad, su horario y su ubicación. La segunda fase consiste en indicar la cantidad que se desea del bien, la fecha y hora de entrega y el tipo de entrega, ya sea retiro en el local o con despacho, en el caso de despacho permite escribir una ayuda adicional. Al colocar dicha información y enviarla, esta se guarda en la base de datos y se le envía una notificación al proveedor.

Para crear ambas vistas se ocupa un sólo documento HTML, lo que es posible gracias a que con Django se puede escribir el código HTML pertinente en función del contexto suministrado. Es posible colocar información extraída desde la base de datos debido a que el template recibe un contexto, lo que es equivalente a un diccionario de Python, desde el archivo "views". Para que estas vistas puedan enviar información certera al proveedor se hace uso principalmente de JavaScript, ya que este supervisa que se entregue toda la información necesaria al proveedor. El último paso lo toma el servidor, al detectar la petición POST y escribir dicha información en la base de datos.

4.1.8. Notificaciones

Dado que en el prototipo la solicitud de bienes es mediante un acuerdo, es necesario diseñar un sistema de notificaciones de solicitudes de venta y respuestas a solicitudes (las vistas están en el Anexo G). La interfaz del sistema consiste en una ventana de notificaciones que muestra principalmente las solicitudes enviadas y recibidas. Las solicitudes enviadas se dividen en: sin respuesta, aceptadas, rechazadas y respondidas con condiciones. Por otra parte, las recibidas se dividen en: sin contestar, aceptadas, rechazadas y contestadas con condiciones. De esta manera se ordena la información, lo cual permite estar al tanto del estado de las solicitudes enviadas como de la existencia y cantidad de las solicitudes a bienes ofertados.

Para que cada usuario pueda seguir el estado de cada una de sus solicitudes, se ocupan tres atributos: "shop", "client" y "phase". El atributo "shop" sirve para identificar a la tienda, el atributo "client" sirve para identificar al cliente, el atributo "phase" sirve para identificar la fase de la solicitud, las cuales a su vez se dividen en: solicitud de compra, consulta aceptación de intervalo, solicitud aceptada y solicitud rechazada. Los atributos "shop" y "client" son constantes y permiten que las solicitudes respectivas se muestren a los usuarios objetivos, pero el atributo "phase" no, ya que se cambia manualmente por el usuario objetivo al aceptar, rechazar y condicionar las solicitudes.

Es posible programar el mecanismo que muestra la información en pantalla gracia a la organización de la información diseñada previamente. Este mecanismo consiste en consultar la base de datos para obtener los ocho tipos de solicitudes concernientes a un usuario, guardarlos en ocho arreglos distintos, insertarlos en un contexto de Django y mostrar dicha información dinámicamente en el documento HTML. Esta información al llegar el template no se muestra inmediatamente, para ver los diversos tipos de solicitudes se utilizan funciones escritas en

JavaScript, las cuales al ser ejecutadas borran la información previa y muestran la información solicitada. El mecanismo que se señala se basa en el patrón MTV, ya que se ocupan 3 capas para desplegar la información solicitada en pantalla.

4.1.9. Evaluaciones de compras

Para calificar las compras se crea un sistema que le permite a cada usuario calificar sus respectivas compras en las cuales ya haya transcurrido el tiempo de entrega (las vistas están en el Anexo H). Para acceder a dicha funcionalidad, en la interfaz de usuario de tipo "cliente" y "cliente-proveedor" se añade un icono que consiste en un conjunto de estrellas que redirigen a la nueva vista. Esta vista muestra, en el componente "accordion" de Bootstrap versión 5.2, tres opciones: "calificar calidad y puntualidad", "calificar calidad" y "calificar puntualidad". Al tocar en cualquiera de estas opciones se muestran las compras del usuario que cumplen con el criterio.

Para programarlo se crea una nueva aplicación con nombre "assessment", junto a lo cual se añade el código necesario para integrarla al proyecto, como la "url" y la renderización de la vista. Para guardar dicha información en la base de datos se crean en el archivo "models" las nuevas entidades de la base de datos: "Score" y "TypeScore". Estas entidades sirven para guardar en la base de datos las evaluaciones junto con toda su información pertinente: cantidad de estrellas, cliente, venta y tipo de evaluación (evaluación de puntualidad o calidad). Para consultar las ventas no evaluadas, se crean dos atributos booleanos en la entidad "Sales" de la aplicación "ventas": Puntuality evaluated y Quality evaluated con valor por defecto falso. Con esta base es posible consultar las ventas no evaluadas y escribir su evaluación respectiva en la base de datos.

Al tener lista la estructura, en la base de datos, para recibir calificaciones, se procede a crear el flujo de acciones para consultar y evaluar ventas. El servidor al recibir la petición GET procede a consultar a la base de datos sobre las ventas que cumplan con todas las siguientes condiciones:

1. El cliente de la venta corresponda al cliente que envía la solicitud.
2. La fecha y hora de entrega de la venta ya han transcurrido.
3. La venta tiene alguna evaluación pendiente.
4. La solicitud fue aceptada.

El resultado de la consulta lo recibe otro método para luego transformarlo en un arreglo de JSON's, el cual se envía mediante un contexto al template de Django. El template transforma la información recibida en un elemento script de HTML que sirve para leer dicha información desde otro script. Desde aquel script se ejecutan dos principales tareas: borrar y mostrar las cartas de evaluación, que correspondan a la opción seleccionada. Cabe destacar que cada carta de evaluación se rellena con la información del JSON correspondiente y la calificación se envía en una solicitud de tipo POST desde un formulario del mismo tipo.

4.2. Desarrollo postvalidación

El desarrollo anterior da como resultado un producto mínimo viable que permite poner a prueba las hipótesis. Dado que el resultado es que las hipótesis tienden a ser apoyadas y la mayoría tiende a comentar que las características complementarias, como el estilo del prototipo, son decisivas para su masificación, se decide hacer una serie de cambios en la interfaz de usuario. Las mejoras de interfaz se enfocan en el registro de usuario, la búsqueda personalizada, el registro de bien y la evaluación de compras debido a que se concentraron la mayoría de los errores en estas funcionalidades.

4.2.1. Registro de usuario

En el registro de usuario, en lugar de ocupar la librería Crispyforms, se utiliza código de la documentación de Bootstrap para estructurar un formulario con la particularidad de tener más de un campo por fila. Lo que además de otorgarle una apariencia más compacta, también permite que se vea la totalidad del formulario con menor esfuerzo y, como consecuencia, aumenta la velocidad de llenado. Además de lo mencionado, no se cambia nada más en lo que respecta a la apariencia (las vistas están en la subsección "Registro de usuario post-validación" del Anexo I).

Una falencia del registro de usuario es que su algoritmo de validación verifica la correctitud de sólo algunos datos en el servidor, lo que ocupa recursos del servidor innecesariamente. Por lo tanto, se programa un mecanismo de validación escrito en JavaScript que revisa los datos desde el lado del cliente y en caso de no encontrar inconsistencias, permite su envío al servidor y en caso de encontrarlas, evita el envío y señala las inconsistencias. Este nuevo mecanismo utiliza expresiones regulares para verificar que no se ingrese información errónea ni que falte información en los diversos campos. En caso de encontrar aquello, prepara y muestra una lista para indicar los campos con errores y destaca en rojo los campos erróneos. Lo que logra un uso eficiente de los recursos y aumenta la seguridad del sitio.

4.2.2. Búsqueda personalizada

La búsqueda personalizada añade un buscador de texto que funciona con "Full text search" de PostgreSQL y cambia su aspecto. Este descarta registros cuyo nombre no cumple con un mínimo nivel de similitud, y respecto al cambio de aspecto, este se centra en la presentación de los filtros y de los resultados (las vistas están en la subsección "Búsqueda personalizada post-validación" del Anexo I). Los filtros se guardan en items del componente "Accordion" de Bootstrap y los resultados de la búsqueda se dividen en dos componentes "Tab" de Bootstrap, los cuales se encapsulan en componentes "ScrollSpy" de Bootstrap. A continuación se detallan los cambios visuales:

1. Para filtrar por precio se presentan dos sliders en una misma barra. El primero fija el precio mínimo (el mínimo que se puede alcanzar corresponde al precio del bien más

barato) y el segundo el precio máximo (el máximo que se puede alcanzar corresponde al precio del bien más caro).

2. Para filtrar por calidad y puntualidad se ocupa el mismo mecanismo, se muestra una fila de cinco estrellas y se escogen de la misma fila la cantidad mínima y máxima de estrellas.
3. Para filtrar por horario de atención se escoge un día y una hora en inputs HTML de tipo "date" y "time" respectivamente.
4. Para filtrar por zona de alcance geográfico se muestra un mapa en un elemento "Off-canvas" de Bootstrap y se escoge un punto, el cual sirve para descartar a las tiendas cuya zona de alcance no llega hasta la comuna a la que pertenece dicho punto.
5. Para cambiar el formato de presentación de los resultados, se escoge la pestaña correspondiente, las cuales son "Lista" y "Mapa". Por defecto está seleccionada el modo "Lista".

4.2.3. Registro de bien

En el caso del formulario para registrar un bien, este ya cuenta con un método de validación de datos, pero carece de la verificación del límite del tamaño de aquellos. En caso de no verificar el tamaño de los datos, se puede intentar ingresar datos que pasen el límite. Como respuesta la base de datos los rechaza y se produce un error de código 500, lo que perjudica la fluidez de la navegación. Con la adición de esta validación es posible evitar las fallas del servidor debido al intento de ingresar registros con atributos muy pesados.

Al registro también se le añaden dos nuevas preguntas: "¿El bien se puede entregar a domicilio?" y "¿El bien se puede retirar personalmente en la tienda?". Esto se hace debido a que existen servicios que no se pueden ejecutar en el domicilio personal y otros que no se pueden llevar a cabo en una sucursal. Para lograr esto se añaden dos nuevos atributos a la entidad "Good" los cuales son "deliverable" (se puede entregar) y "takeable" (se puede tomar). Gracias a esto es posible desarrollar la interacción entre la base de datos y la interfaz que da como resultado un proceso de solicitud de bienes más realista. Se considera más realista, ya que paran de ocurrir situaciones como en las que se consulta si se desea que un servicio de limpieza domestica se ejecute en la tienda o en el domicilio personal.

4.2.4. Evaluaciones de compras

Respecto a las evaluaciones de las compras, se cambia el modo en que se procesan los envíos de las evaluaciones. Al evaluar una compra, en lugar de mandar una solicitud POST, se cambió por una solicitud xmlhttp, ganando así la ventaja de que no se recargue la página por cada evaluación. Al contestar cada solicitud, se reemplaza el botón para evaluar por un ticket que indica que la compra ya fue evaluada por alguno de los dos criterios (calidad o puntualidad). Esto da como resultado un mecanismo más veloz.

Capítulo 5

Validación

En este capítulo se abordan los detalles, tanto del desarrollo como de los resultados, de la validación. La validación consistió en una serie de pruebas con participantes, en cada una se le pidió a los usuarios que ejecutaran una serie de tareas y que contestaran una serie de preguntas. Los resultados permiten validar las hipótesis propuestas y obtener importantes conclusiones.

5.1. Diseño de las pruebas de validación

La validación consiste en una serie de sesiones que se dividen en dos partes cada una: la primera consiste en evaluar la usabilidad y la segunda en encuestar para validar las hipótesis. Para evaluar la usabilidad se le solicita a los participantes que realicen una serie de tareas en la plataforma. Al finalizar con lo anterior se considera que el usuario conoce lo suficientemente bien el prototipo para responder las preguntas de la encuesta. Con los resultados es posible informar sobre el nivel de éxito del trabajo, ya que un buen software debe tener la capacidad para solucionar un problema y debe ser fácil de usar.

Antes de hacer las validaciones se siguen una serie de pasos:

1. Se despliega el prototipo en un servidor web.
2. Se verifica que las funcionalidades estén funcionando correctamente en el servidor.
3. Se escriben en un archivo de tipo Python utilizando la ORM de Django diversos escenarios para cambiar los datos en la plataforma de una forma fácil y rápida. Para crear estos escenarios se varían los parámetros: número de servicios, de productos, de usuarios tipo proveedor, calidad de productos y servicios, calificaciones de puntualidad de los diversos proveedores, y por último, ubicaciones de usuarios en el mapa y alcance geográfico de despachos.

Las pruebas de usabilidad tienen el objetivo de conocer la experiencia de usuario, factor importante para resolver, al menos, parcialmente el problema. Para evaluar la usabilidad se

le pide al participante que realice una serie de tareas, dependiendo de su tipo de usuario con el cual se identifica. Si se identifica con dos tipos, se hace una fusión de tareas, es decir que se considera una serie que incluye ambos grupos. Se solicita a los usuarios de tipo cliente:

1. Registrarse en la aplicación.
2. Explorar el mapa y responder que tipo de información contiene.
3. Realizar dos búsquedas personalizadas: la primera por un producto impuesto y la segunda por un producto a elección personal.
4. Consultar calidad y puntualidad asociadas a los productos buscados.
5. Solicitar un producto o servicio.
6. Calificar calidad y puntualidad.

Para el usuario de tipo proveedor se le solicita:

1. Registrarse en la aplicación.
2. Explorar el mapa y responder que tipo de información contiene.
3. Ofertar un producto.
4. Aceptar una solicitud de compra.
5. Rechazar una solicitud de compra.
6. Proponer un intervalo de llegada a la dirección de despacho solicitada.

Se realizan mediciones en el momento en que los participantes ejecutan dichas tareas para validar la usabilidad, las que se dividen en tres grupos: la eficacia, eficiencia y satisfacción. Con estos datos se puede estimar la proporción de personas que ocuparían la plataforma sin problemas en un contexto real. Además de lo anterior, se puede conocer las fortalezas y debilidades del prototipo, es decir las funcionalidades que la mayoría entiende y las que son difíciles de entender.

Para medir la eficacia se tiene en cuenta lo siguiente:

1. Porcentaje de tareas realizadas con éxito en un primer intento.
2. Cantidad total y promedio de errores por tarea realizada.
3. Cantidad total y promedio de peticiones de ayuda por tarea realizada.

Para medir la eficiencia se tiene en cuenta lo siguiente:

1. Tiempo empleado para realizar por primera vez una tarea.

2. Tiempo empleado para realizar por segunda vez una tarea.

Para medir la satisfacción se tiene en cuenta lo siguiente:

1. Porcentaje de usuarios satisfechos.
2. Porcentaje de usuarios que recomendarían la plataforma.

La encuesta sirve para saber si los diversos tipos de usuario consideran que el prototipo es útil para el cumplir con el objetivo para el cual fue desarrollado. Los usuarios se pueden dividir en tres conjuntos: clientes, proveedores y clientes-proveedores según su actividad en la web. Si consumen se les considera "clientes", si ofertan se les considera "proveedores" y si efectúan ambas actividades, se les considera "clientes-proveedores". De acuerdo a su tipo de usuario, responden las preguntas de un conjunto en específico, ya que a todos los tipos les corresponde un conjunto. Existe un conjunto para usuarios tipo "cliente" y otro conjunto para usuarios tipo "proveedores", en el caso en que se consideren "clientes-proveedores" se consideran las preguntas de ambos conjuntos.

Conjunto de preguntas para usuarios tipo "cliente":

1. ¿Te registrarías en esta página para comprar por internet? ¿Por qué?
2. ¿Te ayudaría esta página a encontrar proveedores de bienes y/o servicios más cercanos geográficamente? ¿Por qué?
3. ¿Aumentaría tu interés en comprar productos y/o servicios cuyos proveedores se encuentren más cercanos geográficamente?

Conjunto de preguntas para usuarios tipo "proveedor":

1. ¿Te registrarías en esta página para ofertar tus productos y/o servicios por internet? ¿Por qué?
2. ¿El sistema de acuerdo entre ambas partes del horario de entrega y la evaluación de puntualidad disminuiría el tiempo de entrega?
3. ¿Las características de esta plataforma te ayudarían a ahorrar tiempo y dinero en despachos?

Si la respuesta anterior es afirmativa:

4. ¿Cómo consecuencias se verían beneficiados los clientes con un despachos más expedito y con un menor costo asociado a este?

Al tener los resultados de la evaluación de usabilidad, se utilizan para mejorar el prototipo en caso de que exista el margen de tiempo que permita dicho desarrollo. Por otra parte, al tener los resultados de la encuesta, el proceso a seguir es diferente, se considera

que se valida la primera hipótesis si más del 50% de los usuarios tipo "cliente" contestan afirmativamente todas las preguntas y se valida la segunda si más del 50% de los proveedores contestan afirmativamente a todas las preguntas. Se considera esta regla, ya que si se tiene una probabilidad mayor al 50% de aprobación de la plataforma, entonces se concluye que en un contexto real la plataforma tendería a popularizarse y como consecuencia, resolvería, al menos parcialmente el problema, en cambio si es menor o igual al 50%, entonces en un contexto real tendería a ignorarse. Esto se deduce a partir del cálculo de probabilidades.

5.2. Perfil de participantes de la primera ronda de validaciones

Se validó el prototipo con 9 participantes, quienes además de realizar las tareas y contestar las preguntas, hicieron comentarios sobre las características de esta, los que se anotaron para mejorar el prototipo. Aquellos se distribuyen en todos los tipos de usuario y además cumplen con el requisito mínimo de tener al menos 3 ventas y/o compras en los últimos 3 años. Sus perfiles se resumen en la siguiente tabla:

	Cliente	Proveedor	Cliente-Proveedor
Cantidad de participantes	5	1	3
Frecuencia mínima de venta	0	1 vez a la semana	2 veces al año
Frecuencia máxima de venta	0	1 vez a la semana	3 veces a la semana
Frecuencia promedio de venta	0	1 vez a la semana	1 vez a la semana
Frecuencia mínima de compra	1 vez al año	0	2 veces al año
Frecuencia máxima de compra	2 veces al mes	0	1 vez al mes
Frecuencia promedio de compra	1 vez al mes	0	1 vez cada dos meses

5.3. Resultados de la primera ronda de validaciones

A continuación se muestra una tabla en la que por cada tarea realizada se consideran de los errores y de las peticiones de ayuda, su cantidad total y promedio por participante. Cabe destacar que cada tarea se ejecuta dos veces, pero con una variación en uno de los parámetros. Por ejemplo el primer intento se diferencia del segundo, en la exploración del mapa, en la cantidad de proveedores mostrados. Además de lo anterior, también se considera el tiempo promedio invertido por cada tarea. Los resultados son los siguientes.

Tarea	Errores	Promedio de errores	Peticiones	Promedio de peticiones	Tiempo promedio [minutos]
Explorar mapa con proveedores en Maipú	4	0.4	7	0.7	1.5
Explorar mapa con proveedores en todo Santiago	0	0	0	0	0.5
Registro de usuarios con datos ficticios	9	1	7	0.8	2.7
Registro de usuario con datos reales	2	0.2	1	0.1	1.1
Buscar bien impuesto	4	0.5	4	0.5	2.25
Consultar calidad del bien impuesto	1	0.125	2	0.25	0.3
Consultar puntualidad del proveedor del bien impuesto	2	0.25	3	0.37	0.6
Solicitar bien impuesto	1	0.125	1	0.125	1.06
Calificar puntualidad de llegada de la compra del bien impuesto	3	0.375	1	0.125	0.44
Calificar calidad de la compra del bien impuesto	0	0	2	0.25	0.3
Ofertar bien impuesto	3	0.75	0	0	0.6
Buscar bien a elección	0	0	0	0	0.16
Consultar calidad del bien a elección	0	0	0	0	0.16
Consultar puntualidad del bien a elección	0	0	0	0	0.16
Solicitar bien a elección	0	0	0	0	0.16
Calificar puntualidad de llegada de la compra del bien a elección	0	0	0	0	0.16
Calificar calidad de la compra del bien a elección	0	0	0	0	0.16
Ofertar bien a elección	0	0	1	0.25	0.375
Aceptar o rechazar solicitud de venta con despacho cercano	1	0.25	0	0	0.47
Aceptar o rechazar solicitud de venta con despacho lejano	0	0	0	0	0.5
Proponer intervalo de llegada para venta con despacho cercano	2	0.5	0	0	0.65
Proponer intervalo de llegada para venta con despacho lejano	0	0	0	0	0.16

En la tabla no es posible colocar todos los resultados, dado que la proporción de usuarios satisfechos y de usuarios que recomendarían la plataforma no tienen un lugar en su estructura. Por otro lado, el porcentaje de tareas ejecutadas correctamente al primer intento tampoco tiene un lugar en la estructura de la tabla, pero se calcula con los datos presentados. En la tabla se observa que existen doce tareas, de las cuales sólo una no presentó complicaciones en ningún caso, por lo tanto el 8 % de las tareas se ejecutaron con éxito en el primer intento. Respecto a la satisfacción el 66 % de los participantes comunicó sentirse satisfecho después de ocupar el prototipo y el 100 % dijo que sí recomendaría el prototipo. Por lo tanto, se puede concluir de la prueba de usabilidad que la mayoría de los potenciales usuarios consideran útil esta plataforma, pero que debe mejorar en lo que concierne a la experiencia de usuario, ya que tan sólo el 8 % de las tareas se pudo ejecutar con éxito en un primer intento, además de los numerosos comentarios de potenciales mejoras.

Respecto a las preguntas para validar las hipótesis, las respuestas que dieron los usuarios para ciertas preguntas no fueron totalmente claras. Algunos respondieron que dependía de ciertas circunstancias la realización de cierta actividad, por ejemplo registrarse para solicitar bienes depende de la calidad de los productos de los proveedores cercanos. A pesar de lo anterior las respuestas de los usuarios se pueden dividir en tres grupos "sí", "no" y "depende de que las opciones que presente la plataforma sean buenas". A continuación se presenta en una tabla el detalle de las respuestas otorgadas.

Pregunta	Porcentaje de sí's	Porcentaje de no's	Porcentaje de depende's
¿Te registrarías en esta página para comprar por internet?	75 %	12.5 %	12.5 %
¿Te ayudaría esta página a encontrar proveedores de bienes y/o servicios más cercanos geográficamente?	100 %	0 %	0 %
¿Aumentaría tu interés en comprar productos y/o servicios cuyos proveedores se encuentren más cercanos geográficamente?	75 %	0 %	25 %
¿Te registrarías en esta página para ofertar tus productos y/o servicios por internet?	75 %	0 %	25 %
¿El sistema de acuerdo entre ambas partes del horario de entrega y la evaluación de puntualidad disminuiría el tiempo de entrega?	50 %	50 %	0 %
¿Las características de esta plataforma te ayudarían a ahorrar tiempo y dinero en despachos?	0 %	25 %	75 %
¿Cómo consecuencias se verían beneficiados los clientes con un despachos más expedito y con un menor costo asociado a este?	50 %	0 %	50 %

En la tabla se observa un resumen de las respuestas a las preguntas de la encuesta, pero no es posible validar alguna sin información detallada de lo que contestó cada uno. El 63 % de los participantes de tipo cliente tienen la certeza de que este prototipo les ayudaría a encontrar productos y/o servicios más cercanos geográficamente y como consecuencia disminuiría el costo asociado a la llegada del bien, lo que valida la primera hipótesis. Respecto a los proveedores, ninguno contestó afirmativamente a todas las preguntas, tendieron a contestar que dependía de las circunstancias, por lo tanto queda inconclusa la segunda hipótesis, pero dado que al menos el 75 % de ellos dijeron que dependía tanto de la cantidad como de la ubicación de los clientes, es muy probable que sí ocuparían la plataforma, porque entre los clientes hay un amplio consenso de que esta página les ayudaría a encontrar proveedores más cercanos. Por lo tanto, después de la primera validación, sólo se puede validar la primera hipótesis, debido a que sobre el 50 % de los clientes reconocen la utilidad para encontrar proveedores de dicha plataforma, pero queda inconclusa la segunda, ya que los proveedores tendieron a contestar que dependía de las circunstancias.

5.4. Preparación de la segunda ronda de validaciones

Al finalizar la primera ronda de validaciones, se obtuvo importante información con la cual se siguió desarrollando el prototipo y se modificó un poco el diseño de las validaciones. Dado que el prototipo mostró importantes debilidades en la usabilidad, se modificó el diseño de la interfaz para que esta fuera más fácil de entender. Además de lo anterior, se observó que al pedir la tarea de realizar una búsqueda personalizada, los usuarios tendían a ocupar pocas herramientas, por lo que aquella tarea se dividió en cinco:

1. Buscar bien por calidad.
2. Buscar bien por puntualidad.
3. Buscar tienda con bien genérico por disponibilidad.
4. Buscar tienda con bien genérico por zona de alcance.
5. Buscar tienda cercana con bien genérico.

5.5. Perfil de participantes de la segunda ronda de validaciones

En la segunda ronda de validaciones se contó con 11 participantes, de los cuales 7 participaron por segunda vez y 4 participaron por primera vez. Cabe destacar que en este caso se llegó a un número mayor de personas, sin embargo en diversos casos no se pudo solicitar las ejecuciones de todas las tareas. Se debe a que en esta ronda no todos los participantes contaban con suficiente tiempo y la cantidad de tareas aumentó, es decir aumentó el tiempo estimado y disminuyó el tiempo disponible. En la siguiente tabla se resume el perfil de los participantes:

	Cliente	Proveedor	Cliente-Proveedor
Cantidad de participantes	8	1	2
Cantidad de participantes con tiempo para todas las tareas	6	1	1
Frecuencia mínima de venta	0	1 vez a la semana	2 veces al año
Frecuencia máxima de venta	0	1 vez a la semana	3 veces al año
Frecuencia promedio de venta	0	1 vez a la semana	1 vez cada 2 meses
Frecuencia mínima de compra	2 veces al año	0	2 veces al año
Frecuencia máxima de compra	1 vez al mes	0	3 veces al año
Frecuencia promedio de compra	11 veces al año	0	1 vez cada 2 meses

5.6. Resultados de la segunda ronda de validaciones

Para la segunda ronda de validaciones se consideran los mismos indicadores para la evaluación de usabilidad, pero las tareas no son idénticas. Hay tres principales diferencias en lo que respecta a las tareas pedidas a los usuarios:

1. La primera se mencionó en la sección "Preparación de la segunda ronda de validaciones".
2. La segunda corresponde a que en esta oportunidad no se impuso la búsqueda de un bien en específico, ya que tanto en el primer como en el segundo intento se dio la libertad de buscar un bien a elección.
3. La tercera diferencia es que sólo se les pidió la ejecución de las tareas 2 veces a los cuatro nuevos participantes.

A las personas que participaron por segunda vez no se les pidió el segundo intento de ejecución para las tareas porque los cambios de la interfaz no fueron tantos para que se considerase una interfaz totalmente nueva. Los resultados de las evaluaciones de la ejecución de tareas se dividieron en dos tablas, en la primera tabla se colocan los resultados de la primera ejecución de las tareas y en la segunda los resultados de la segunda ejecución.

Tarea	Errores	Promedio de errores	Peticiones	Promedio de peticiones	Tiempo promedio [minutos]
Explorar mapa con proveedores en Maipú	1	0.09	2	0.18	1.1
Registro de usuario 1° intento	0	0	2	0.18	1.8
Buscar bien por calidad 1° intento	4	0.37	1	0.09	0.4
Buscar bien por puntualidad 1° intento	0	0	1	0.09	0.21
Consultar calidad asociada al bien 1° intento	2	0.18	0	0	0.08
Consultar puntualidad del bien 1° intento	4	0.37	3	0.27	0.33
Buscar tienda con bien genérico por disponibilidad horaria 1° intento	2	0.18	2	0.18	0.47
Buscar tienda con bien genérico por zona de alcance geográfico 1° intento	1	0.09	7	0.64	1.47
Buscar tienda cercana con bien genérico 1° intento	5	0.45	4	0.36	0.55
Solicitar bien 1° intento	4	0.4	2	0.2	0.92
Calificar puntualidad de llegada 1° intento	0	0	0	0	0.09
Calificar calidad del bien 1° intento	0	0	0	0	0.1
Ofertar bien	0	0	1	0.5	2.5
Aceptar solicitud retiro en tienda	0	0	0	0	0.33
Rechazar solicitud cercana	0	0	0	0	0.46
Proponer intervalo solicitud lejana	0	0	0	0	0.33

Tarea	Errores	Promedio de errores	Peticiones	Promedio de peticiones	Tiempo promedio [minutos]
Registro de usuario 2° intento	1	0.25	0	0	1.13
Buscar bien por calidad 2° intento	0	0	0	0	0.08
Buscar bien por puntualidad 2° intento	0	0	0	0	0.13
Consultar calidad asociada al bien 2° intento	0	0	0	0	0.06
Consultar puntualidad del bien 2° intento	0	0	0	0	0.06
Buscar tienda con bien genérico por disponibilidad horaria 2° intento	0	0	0	0	0.13
Buscar tienda con bien genérico por zona de alcance geográfico 2° intento	0	0	0	0	0.33
Buscar tienda cercana con bien genérico 2° intento	0	0	2	1	0.83
Solicitar bien 2° intento	0	0	0	0	0.33
Calificar puntualidad de llegada 2° intento	0	0	0	0	0.06
Calificar calidad del bien 2° intento	0	0	0	0	0.06

Al comparar los resultados de la evaluación de usabilidad de la primera ronda de validaciones con la segunda hay numerosas diferencias. En la mayoría de los casos disminuyó el promedio de errores, promedio de peticiones de ayuda y el tiempo empleado promedio por tarea. Para poder comparar los resultados de ambas rondas, a pesar de sus diferencias ya mencionadas, se aplicaron una serie de consideraciones, las cuales son:

1. Las nuevas tareas de búsqueda de la segunda ronda de validaciones se consideraron en una sola tarea de búsqueda.
2. Las tareas de aceptar y rechazar solicitudes se consideraron en una sola tarea.
3. Las tareas de calificar calidad y puntualidad, se consideraron como una sola.
4. Para obtener los promedios de las tareas que representan a más de una se calcula el promedio de los promedios de la cantidad de errores, cantidad de peticiones de ayuda y de los tiempos empleados de las tareas que las conforman a cada una.
5. Sólo se compararon los datos del primer intento de ejecución de cada una de las tareas.

Los detalles de la variación se indican a continuación:

Tarea	Variación promedio de errores	Variación promedio de peticiones	Variación tiempo promedio [minutos]
Registro de usuario	-77 %	-74 %	-27 %
Buscar bien	-56 %	-46 %	-72 %
Explorar mapa	-77 %	-74 %	-27 %
Consultar calidad asociada al bien	+44 %	-100 %	-73 %
Consultar puntualidad del bien	+48 %	-27 %	-45 %
Solicitar bien	+220 %	+60 %	-13 %
Calificar bien	-100 %	-100 %	-49 %
Ofertar bien	-100 %	(+)	+316 %
Aceptar o rechazar bien	-100 %	0 %	-16 %
Proponer intervalo solicitud lejana	-100 %	0 %	-49 %

El desarrollo posterior al feedback tendió a beneficiar la experiencia de usuario, pero en algunos casos se produjo el efecto contrario. Se puede destacar que se logró mejorar la experiencia de usuario del registro de usuario, la búsqueda de bienes y la calificación de las compras. Sin embargo hubo retrocesos en lo que respecta a la solicitud de bienes y las consultas de la calidad y la puntualidad asociadas al bien y al proveedor respectivamente. Se pueden explicar los logros debido a que se tomó en consideración los resultados y los comentarios obtenidos de la primera ronda de evaluaciones. Respecto a los retrocesos se puede explicar debido a que en aquellos casos se innovó en la interfaz sin basarse en los comentarios.

En las tablas mostradas y analizadas no se han considerado 3 indicadores: porcentaje de usuarios satisfechos, porcentaje de usuarios que recomendarían el prototipo y porcentaje de tareas correctamente ejecutadas al primer intento. El porcentaje de usuarios satisfechos correspondió al 73 %, el porcentaje de usuarios que la recomendarían se situó en un 91 % y el porcentaje de tareas que se ejecutaron correctamente en su primer intento se situó en un 25 % (se fusionaron las tareas de aceptar y rechazar solicitud). Se puede deducir de los datos presentados que el porcentaje de satisfacción subió un 10 %, el porcentaje de recomendación disminuyó un 9 % y el porcentaje de de tareas correctamente ejecutadas en un primer intento aumentó en un 17 %.

Respecto a las preguntas para validar las hipótesis, sólo se les realizaron a los nuevos participantes, ya que el resto ya las contestó. Las nuevas respuestas variaron la proporción de usuarios que se registrarían en la plataforma, pasando de un 75 % a un 91 %, pero respecto al resto de preguntas para usuarios tipo "cliente" no hubo cambios. No hubo cambios en las distribuciones restantes de las respuestas de los usuarios de tipo "cliente", debido a que la proporciones se mantuvieron. Por el otro lado, dado que no hubo nuevos participantes de tipo "proveedor", las distribuciones de las respuestas a las preguntas de aquel grupo no varió. Por lo tanto se mantienen sin modificaciones los resultados de las validaciones de las hipótesis: la primera se acepta y la segunda queda inconclusa.

Capítulo 6

Conclusión

El proceso consistió en varias y diversas etapas. La primera consistió en el diseño de la base de datos, el que se modificó continuamente debido a su rigidez ante los cambios. Posteriormente en la etapa de diseño de interfaces de usuario hubo una retroalimentación que confirmó que la estructura general era fácil de entender, pero con detalles que debían cambiar. Lo anterior dio inicio al desarrollo del código, lo que se hizo con un conjunto de novedades, por ejemplo la utilización de los mapas vectoriales para delimitar los bordes de las comunas y la utilización de PostGis que permite guardar datos espaciales y hacer consultas espaciales. Lo que posibilitó la realización de la etapa de validación, la que consta de una prueba de usabilidad y de una encuesta por participante. La que a su vez, gracias al feedback de la primera ronda, permitió dar como resultado un prototipo con una mejor experiencia de usuario en comparación con la primera versión.

Respecto al objetivo del trabajo de título, se puede resumir de la siguiente forma: la elaboración de un prototipo para validar con potenciales usuarios 2 hipótesis. El prototipo sí se pudo desarrollar como se planificó, ya que se utilizó de forma correcta el framework Django, el gestor PostgreSQL con su extensión PostGis y la librería de JavaScript Leaflet. En cuanto a las validaciones, estas se efectuaron gracias al correcto despliegue del prototipo en la web y la utilización de vídeo llamadas vía Zoom. En conjunto permitieron monitorear su utilización en tiempo real y posibilitó una comunicación fluida, que sirvió para anotar los indicadores de usabilidad y diversos comentarios. Si no se hubiera finalizado alguna etapa exitosamente, se habría prolongado la extensión del trabajo de título o se habría fallado en la consecución de los objetivos planteados en la propuesta.

Se pueden analizar, del trabajo realizado, los resultados obtenidos en las validaciones. De la evaluación de usabilidad se concluye que hubo una mejora en la experiencia de usuario al comparar los resultados de la primera ronda con la segunda. Sin embargo quedaron detalles que mejorar, por ejemplo la capacidad del buscador de texto que busca por similitud de palabras y se necesita que también busque por similitud semántica. De los resultados del proceso de validación de hipótesis se concluye que se acepta la primera, pero la segunda queda inconclusa debido a que los proveedores no estaban convencidos. Lo anterior se puede resumir en que se consiguió que los potenciales usuarios tendieran a aceptar el prototipo, pero que a pesar de eso, careció de varias funcionalidades. Se tendió a notar la ausencia de estas porque

tienen el potencial de favorecer su masificación, entre estas está un "chat" entre clientes y proveedores para solucionar dudas. Además aumenta la confianza de los proveedores, ya que aquellos no desean invertir tiempo en plataformas poco conocidas, debido a la escasa probabilidad de éxito en sus ventas.

La aplicación no tuvo buenos resultados de usabilidad en la primera ronda de validaciones debido a que falló principalmente su interfaz de usuario, la que no tuvo un aspecto convencional, por ejemplo el buscador no tenía la funcionalidad para buscar por texto y el estilo de letra era poco común, por lo que costaba entenderse. Por otra parte, en la segunda ronda de validaciones falló la escasez de instrucciones en algunas vistas, pero mejoró respecto a la primera versión. Respecto a las hipótesis, se aceptó la primera hipótesis debido a que los usuarios del tipo "cliente" en una proporción mayor al 50% declararon que el prototipo era útil para lo que se desarrolló. Por otro lado la segunda hipótesis quedó inconclusa porque los proveedores no confiaron en un prototipo con poco atractivo para atraer público, pero tampoco se negaron a validar la hipótesis en caso de tener la plataforma funcionando constantemente en la web. Por lo que se puede destacar que hubo una buena implementación de lo propuesto, pero que no considero al mismo nivel el desarrollo de una plataforma atractiva para los potenciales usuarios.

Dado que se pudo validar sólo una de las hipótesis y la restante quedó inconclusa, se puede criticar que el diseño del experimento tiene fallas. En este caso las tiene, debido a que se asumió que los proveedores tendrían la seguridad de contestar las preguntas que les correspondía, pero no coincidió con la realidad. Esta inseguridad tiene su origen en que no supieron estimar el beneficio que les otorgaría el prototipo debido a que no conocían una cifra aproximada de potenciales clientes cercanos geográficamente. Como los proveedores no tienen dicha seguridad, es necesario que el prototipo esté en funcionamiento constante en la red para que se pueda validar la segunda hipótesis.

El trabajo realizado es importante ya que deja avanzado bastante código para desarrollar una aplicación mas sofisticada que pueda contribuir de mejor manera a solucionar los problemas mencionados. Además muestra cuales son los desafíos a los que se podría enfrentar un desarrollador que desee construir una aplicación semejante. Por ejemplo se destaca lo fundamental que es tener una buena interfaz y diseño de base de datos. Por último, dado que se aborda un par de problemas relevantes, es adecuado inferir que los resultados son un aporte para la solución de la problemática que aqueja a muchas personas.

Dado que se desarrollaron funcionalidades para diversos tipos de usuario, implica que hay varias enseñanzas respecto al desarrollo de una plataforma web, sin embargo todas las enseñanzas no aparecen en el informe. Algunas de estas se enumeran a continuación:

1. Al desarrollar un prototipo, es fundamental identificar que es lo novedoso, averiguar como guardar dicha información de forma eficiente y presentar la información de una forma entendible por la mayoría. Con las características no novedosas es posible obtener numerosos ejemplos, lo que las hacen más fáciles de implementar.
2. Se debe considerar que el prototipo está sujeto a modificaciones, ya sea por errores en la planificación o por falta de especificaciones, y que no se debe planificar como una estructura rígida.

3. En el caso de hacer formularios dinámicos, es una buena práctica hacer una validación con JavaScript, ya que así no se borran las respuestas contestadas a preguntas cargadas de manera dinámica en el documento al recargar la página.
4. Es conveniente guardar información como el conjunto de comunas del país en una o varias tablas de la base de datos, ya que así se ahorran varias líneas de código.

Dado lo anterior, se puede deducir que también fue una excelente oportunidad para progresar en el área del desarrollo de aplicaciones web.

Para evitar las dificultades que se tuvieron con respecto a la usabilidad se pudo utilizar los 10 principios de Nielsen. Estos principios sirven para anticipar los posibles problemas que tenga un usuario al navegar por un sitio web. Esta información facilita que un sitio desde un inicio esté enfocado en otorgar una buena experiencia de usuario, lo cual habría logrado un desarrollo más eficiente del prototipo.

Debido a la información recopilada en las dos rondas de validaciones, se puede planificar trabajo futuro, cuyo objetivo sería mejorar el atractivo de la página. Por el lado del cliente se eliminarían algunas funcionalidades innecesarias, por ejemplo escoger un rango de estrellas de calidad en el buscador por escoger una estrella que represente el mínimo. Además se añadiría un buscador de texto que pueda discriminar por similitud semántica. Por el lado del proveedor se añadirían más funcionalidades que los pueda mantener informados de lo que pasa en la página mientras no están conectados. Por ejemplo un sistema de envío de correos que informa sobre sus bienes solicitados y un sistema de notificaciones que avisa de las solicitudes recibidas. Al desarrollar estas características se podría buscar otra vez participantes para obtener una retroalimentación y desarrollar iterativamente, hasta que se alcance un buen nivel de desarrollo. Con eso se puede poner en marcha la plataforma en la web con su dominio propio, lo que, como consecuencia, permitiría hacer un estudio para validar la segunda hipótesis.

Bibliografía

- [1] <https://es.statista.com/estadisticas/1136646/tasa-penetracion-mas-altas-internet-america-latina-caribe/>
Última visita: 23 - 08 - 2022
- [2] <https://www.latercera.com/pulso-pm/noticia/quiebre-de-stock-escasez-de-chips-y-alta-demanda-ponen-presion-a-la-industria-de-los-computadores/KFZCI262WJB6FPM64SNDSJA5RI/>
Última visita: 24 - 08 - 2022
- [3] <https://www.biobiochile.cl/noticias/ciencia-y-tecnologia/moviles-y-apps/2021/06/03/chile-rompe-record-de-ventas-de-moviles-en-el-primer-trimestre-de-2021-con-2-9-millones-de-unidades.shtml>
Última visita: 24 - 08 - 2022
- [4] <https://expandeonline.cl/blog/ecommerce-chile/>
Última visita: 24 - 08 - 2022
- [5] <https://www.latercera.com/piensa-digital/noticia/e-commerce-sigue-en-alza-y-empresas-apuestan-por-el-cumplimiento-con-sus-usuarios/7IDIMGJX4FBXFJSP2WLCCVSM4I/>
Última visita: 24 - 08 - 2022
- [6] <https://guiadecomprasorando.com/las-compras-online-ventajas-y-desventajas>
Última visita: 24 - 08 - 2022
- [7] <https://www.america-retail.com/trade-retail-management/catman-cuales-son-las-ventajas-y-desventajas-de-comprar-por-internet/>
Última visita: 24 - 08 - 2022
- [8] <https://cl.oberlo.com/blog/ventajas-y-desventajas-que-debes-conocer-acerca-del-ecommerce>
- [9] <https://unade.edu.mx/ventajas-y-desventajas-del-ecommerce/>
Última visita: 24 - 08 - 2022
- [10] <https://www.incentro.com/es-es/blog/stories/ventajas-desventajas-comercio-electronico/>
Última visita: 24 - 08 - 2022

- [11] <https://eresciencia.com/9-beneficios-metodo-cientifico/>
Última visita: 29 - 08 - 2022
- [12] <https://www.crehana.com/blog/desarrollo-web/lenguajes-de-programacion-mas-usados/>
Última visita: 31 - 08 - 2022
- [13] <https://www.akademus.es/blog/emprendedores/python-vs-java-cual-es-mejor-conoce-sus-diferencias/>
Última visita: 31 - 08 - 2022
- [14] <https://docs.aws.amazon.com>
Última visita: 31 - 08 - 2022
- [15] <https://www.universidadviu.com/co/actualidad/nuestros-expertos/lenguaje-sql-historia-y-conceptos-basicos>
Última visita: 31 - 08 - 2022
- [16] <https://desarrolloweb.com/home/bases-de-datos>
Última visita: 31 - 08 - 2022
- [17] <https://es.gadget-info.com/difference-between-mysql>
Última visita: 31 - 08 - 2022
- [18] <https://ayudaleyprotecciondatos.es/bases-de-datos/postgresql-vs-mysql/>
Última visita: 31 - 08 - 2022
- [19] <https://mape309site.wordpress.com/2017/11/15/ventajas-y-desventajas-de-mysql-oracle-visual-foxpro-y-access/>
Última visita: 31 - 08 - 2022
- [20] <https://dataaustral.com/2021/02/19/oracle-ventajas-y-caracteristicas/>
Última visita: 31 - 08 - 2022
- [21] <https://blog.glajumedia.com/top-frameworks-de-desarrollo-web-en-el-2020/>
Última visita: 31 - 08 - 2021
- [22] <https://openwebinars.net/blog/10-web-famosas-sabias-usaban-django/>
Última visita: 31 - 08 - 2022
- [23] <https://blog.nubecolectiva.com/5-populares-aplicaciones-que-usan-tecnologia-react/>
- [24] <https://www.mercadolibre.cl/>
Última visita: 31 - 08 - 2022
- [25] <https://www.amazon.com/>
Última visita: 31 - 08 - 2022

- [26] <https://www.elmostrador.cl/agenda-pais/2022/04/25/ecommerce-en-alza-63-de-chilenos-ya-compra-en-linea>
Última visita: 30 - 09 - 2022
- [27] <https://www.lucidchart.com>
Última visita: 19 - 10 - 2022
- [28] <https://balsamiq.com/>
Última visita: 19 - 10 - 2022
- [29] <https://www.epitech-it.es/que-es-un-framework/>
Última visita: 19 - 10 - 2022
- [30] <https://espifreelancer.com/mtv-django.html>
Última visita: 19 - 10 - 2022
- [31] <https://www.freecodecamp.org/espanol/news/para-que-se-utiliza-django-de-python-5-razones-claves-por-las-que-uso-el-framework-django-para-proyectos/>
Última visita: 19 - 10 - 2022
- [32] <https://keepcoding.io/blog/ventajas-y-desventajas-de-python/>
Última visita: 20 - 10 - 2022
- [33] <https://www.ionos.es/digitalguide/servidores/know-how/postgresql/>
Última visita: 20 - 10 - 2022
- [34] <https://openwebinars.net/blog/introduccion-heroku>
Última visita: 20 - 10 - 2022
- [35] <https://www.heroku.com/>
Última visita: 20 - 10 - 2022
- [36] <https://www.jetbrains.com/es-es/pycharm/features/>
Última visita: 16 - 11 - 2022
- [37] <https://docs.djangoproject.com/en/4.1/topics/db/queries/>
Última visita: 16 - 11 - 2022
- [38] <https://mappinggis.com/2016/11/openlayers-vs-leaflet-mejor/>
Última visita: 16 - 11 - 2022
- [39] <https://leafletjs.com/examples.html>
Última visita: 16 - 11 - 2022
- [40] <https://jungleworks.com/es/Los-5-mejores-software-de-optimizaci>
Última visita: 24 - 11 - 2022

Anexos

A. Anexo A

A.1. Diseño de base de datos antiguo

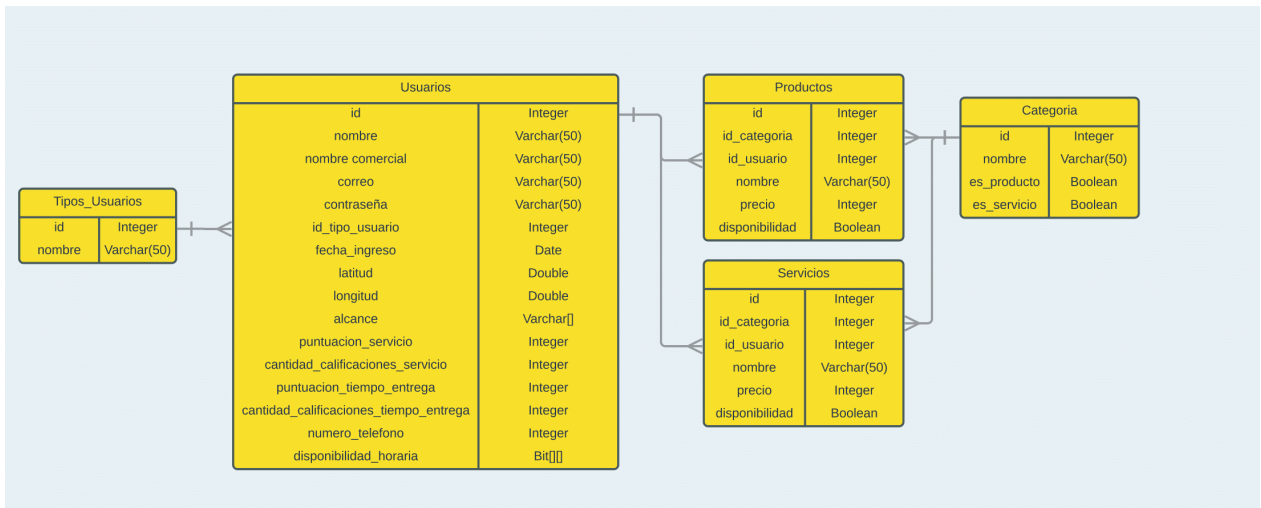


Figura 1: Primera versión del modelo de la base de datos

Link de descarga: https://1drv.ms/b/s!AgpnNcCQEh0dg_dfemDsyIFN4KCZBg?e=GLtcLc

A.2. Diseño final de la base de datos

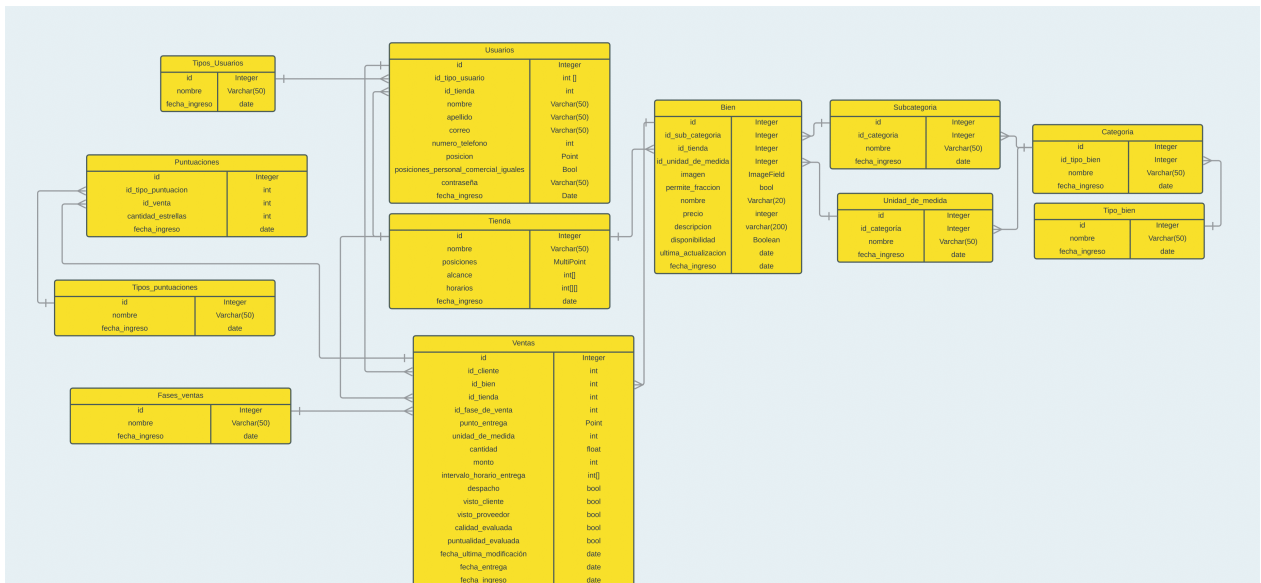


Figura 2: Versión final del modelo de la base de datos

Link de descarga: https://1drv.ms/b/s!AgpnNcCQEh0dg_dgoaA5wMoxn1ukzQ?e=YGmcoY

B. Anexo B

B.1. Página de inicio

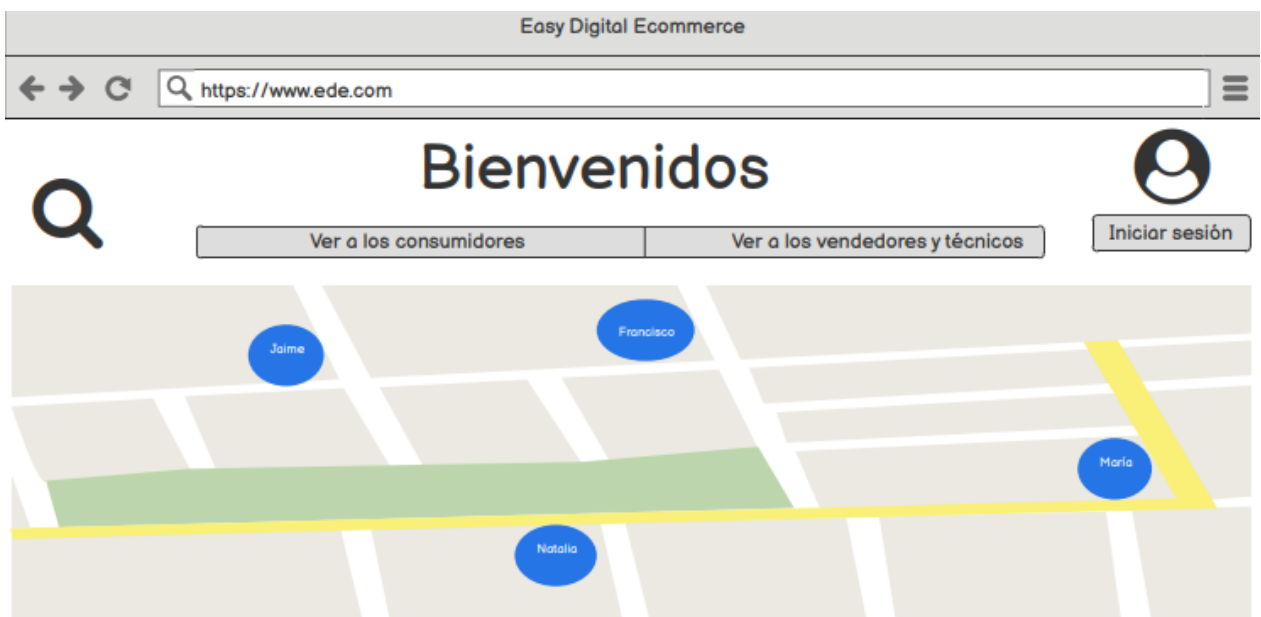


Figura 3: Vista de inicio de usuario sin cuenta o sin sesión iniciada

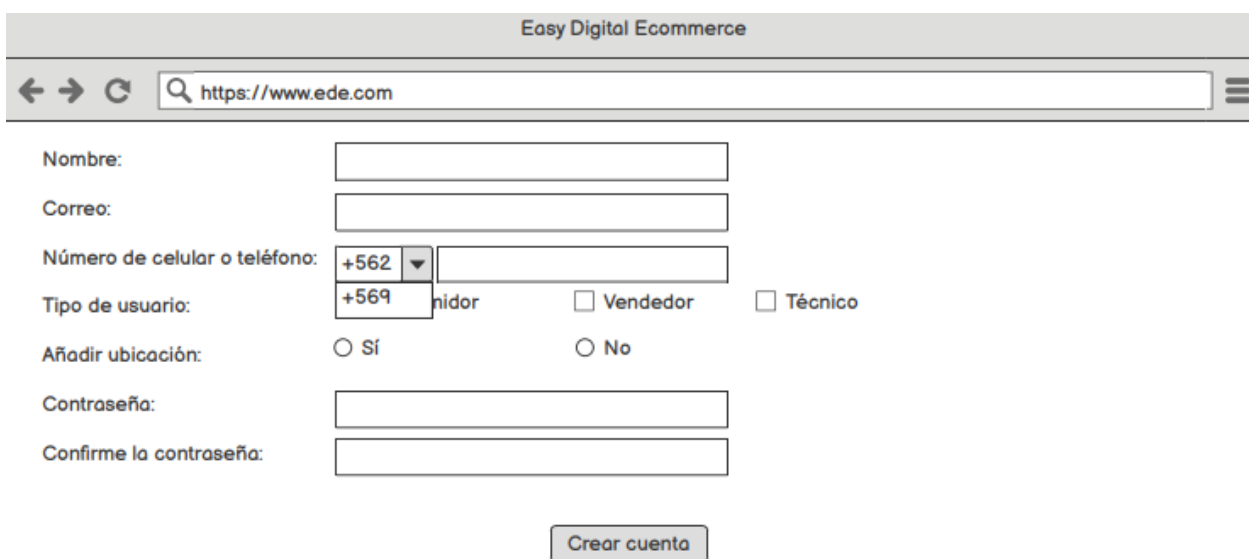
B.2. Inicio de sesión



The screenshot shows a web browser window with the title "Easy Digital Ecommerce" and the URL "https://www.ede.com". The main heading is "Inicio de sesión". Below the heading are two input fields labeled "correo" and "contraseña". There are two links: "Se me olvidó mi contraseña" and "No tengo una cuenta". At the bottom is a "Confirmar" button.

Figura 4: Vista de inicio de sesión

B.3. Registro de usuario



The screenshot shows a web browser window with the title "Easy Digital Ecommerce" and the URL "https://www.ede.com". The registration form includes the following fields and options:

- Nombre: [input field]
- Correo: [input field]
- Número de celular o teléfono: +562 [input field]
- Tipo de usuario: +569 [input field] Vendedor Técnico
- Añadir ubicación: Sí No
- Contraseña: [input field]
- Confirme la contraseña: [input field]

At the bottom is a "Crear cuenta" button.

Figura 5: Vista de registro de usuario

Easy Digital Ecommerce

← → ↻ 🔍 https://www.ede.com ☰

Nombre:

Correo:

Número de celular o teléfono: +569

Tipo de usuario: Consumidor Vendedor Técnico

Añadir ubicación: Sí No

Añadir horario de operación: Sí No

Añadir zona de despachos: Sí No

Contraseña:

Confirme la contraseña:

Figura 6: Vista de registro de usuario tipo vendedor. En este caso se observan más campos a completar.

Easy Digital Ecommerce

← → ↻ 🔍 https://www.ede.com ☰

Nombre:

Correo:

Número de celular o teléfono: +569

Tipo de usuario: Consumidor Vendedor Técnico

Añadir ubicación: Sí No

Añadir horario de operación: Sí No

Añadir zona de despachos: Sí No

Contraseña:

Confirme la contraseña:

Figura 7: Vista de registro de usuario tipo técnico. En este caso también se observan campos extras a completar.

B.4. Registro de bien

The screenshot shows a web browser window with the URL `https://www.ede.com`. The page title is "Añadir oferta". On the left, there is a home icon. The form contains the following fields and controls:

- Nombre:** A text input field.
- Tipo de oferta:** A dropdown menu with "Producto" selected.
- Descripción:** A text area with "Servicio" entered.
- Disponibilidad:** A toggle switch that is turned on.
- Con despacho:** A toggle switch that is turned on.
- Publicar:** A button at the bottom of the form.

Figura 8: Vista de registro de bien

B.5. Página de inicio de usuario registrado



Figura 9: Vista de inicio de usuario con sesión iniciada

B.6. Búsqueda personalizada



Figura 10: Vista de búsqueda personalizada

B.7. Resultados de la búsqueda personalizada



Figura 11: Resultados de la búsqueda en formato de lista

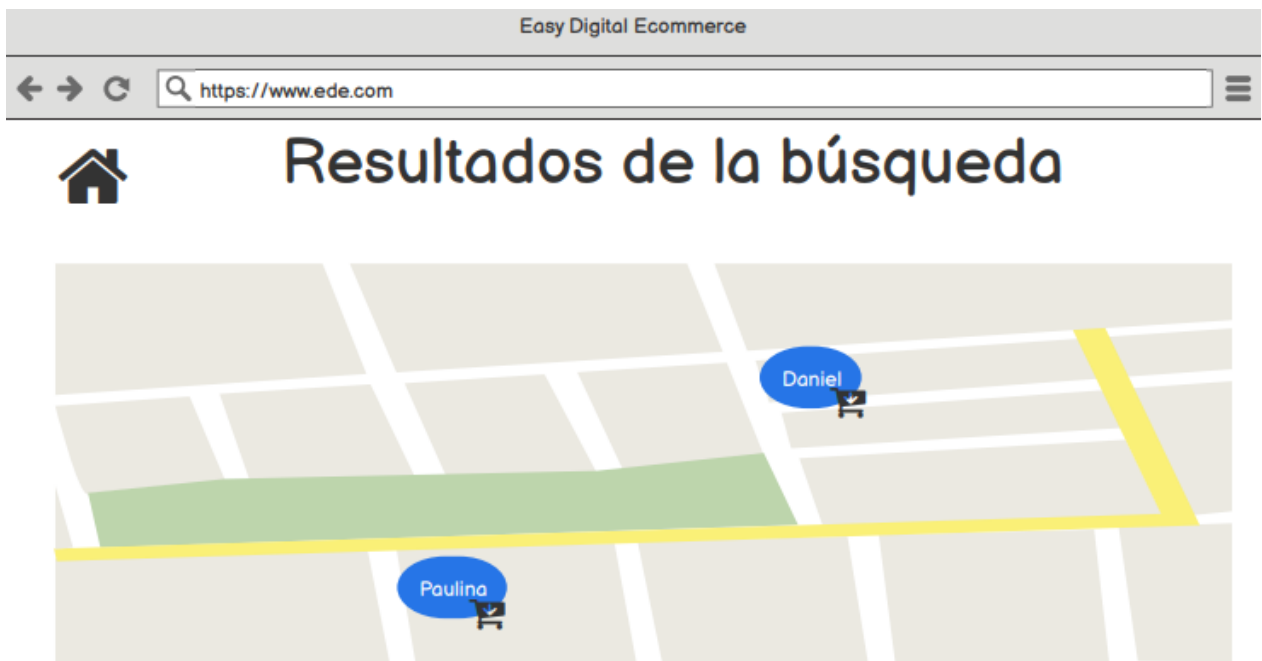


Figura 12: Resultados de la búsqueda en formato de mapa

B.8. Proceso de solicitud de bien



Figura 13: Vista del primer paso del proceso de solicitud



Figura 14: Vista del segundo paso del proceso de solicitud

B.9. Ajustes



Figura 15: Vistas de ajustes de la información ingresada por el usuario al registrarse



Figura 16: Vista de ajustes de los datos de contacto



Figura 17: Vista de ajustes de la ubicación

B.10. Notificaciones

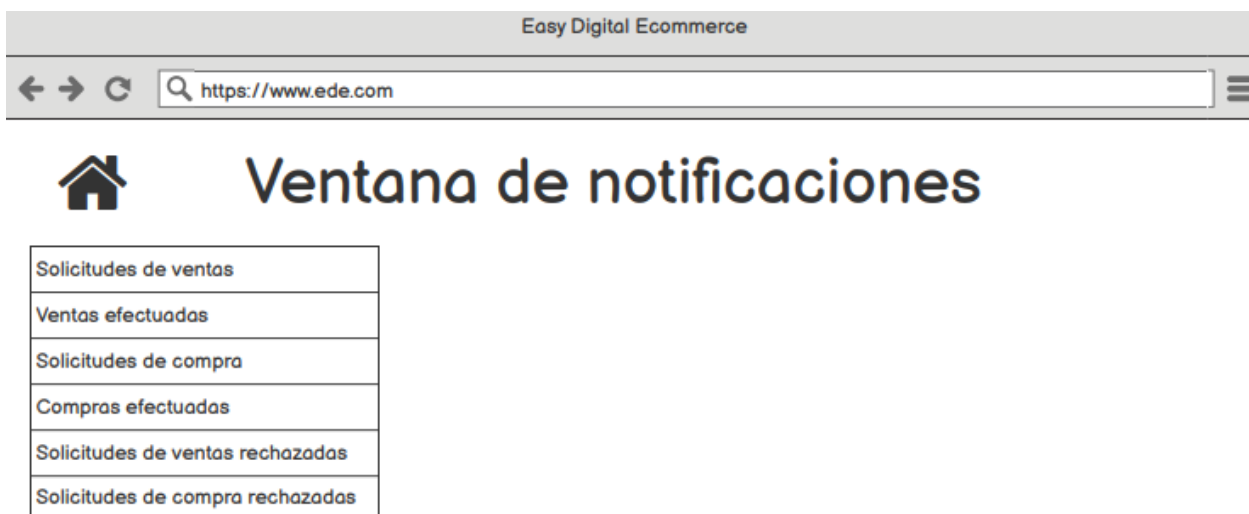


Figura 18: Ventana de notificaciones

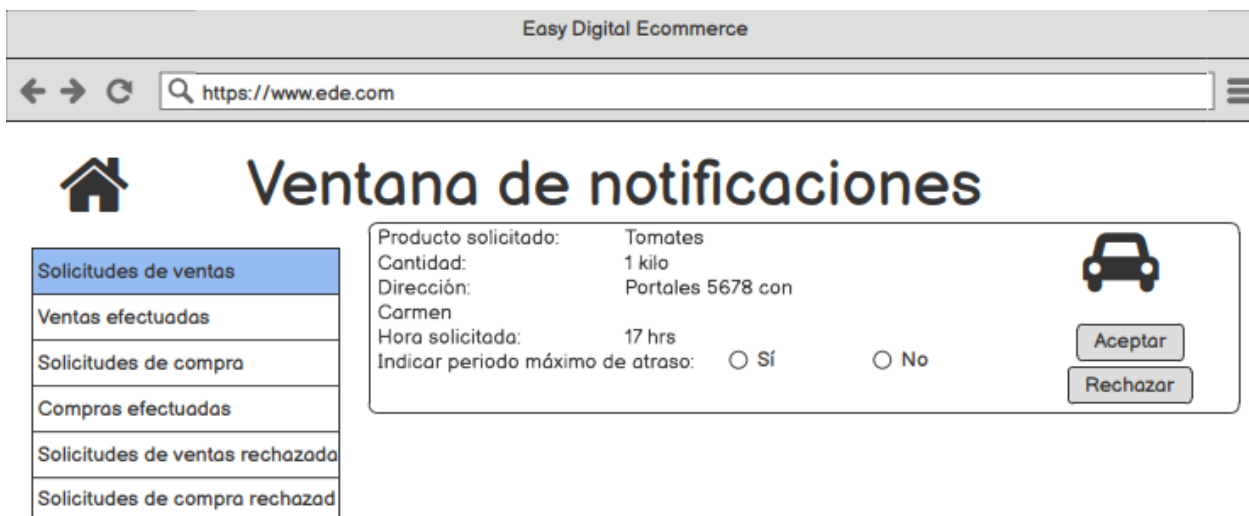


Figura 19: Notificaciones de ventas recibidas. Se observa que es una solicitud debido a la flexibilidad de respuesta, ya que se puede aceptar, rechazar e indicar un periodo de atraso. Lo último actúa como una solicitud para el cliente, ya que aquel también puede rechazar el intervalo horario de entrega.

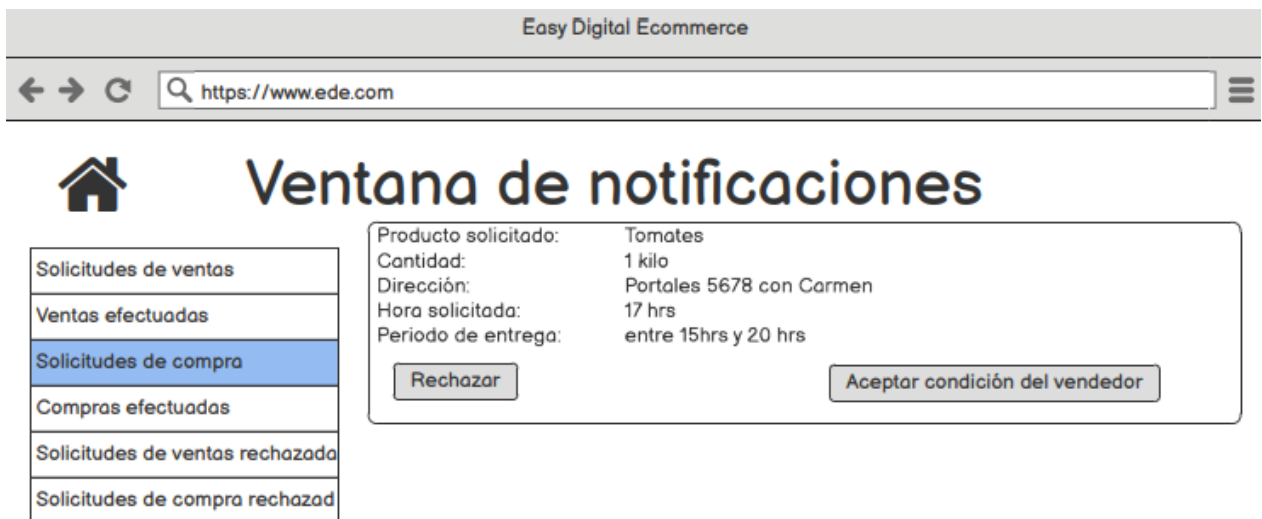


Figura 20: Notificación de compra. Se observa que el vendedor propone un horario de llegada, lo cual puede ser aceptado o rechazado por el cliente.



Figura 21: Vista de las compras efectuadas por el usuario. En este caso la compra se puede evaluar porque aún no ha sido evaluada y ya transcurrió el plazo de entrega.

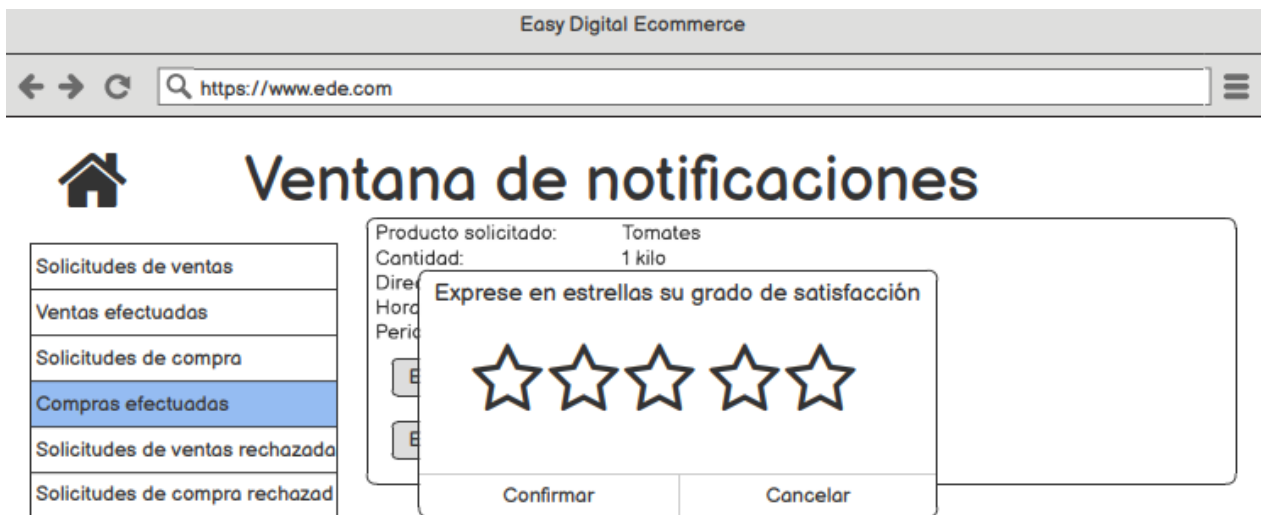


Figura 22: Presentación de ventana para evaluar compra

C. Anexo C

C.1. Inicio de sesión



Figura 23: Vista de inicio de sesión de usuario

C.2. Registro

Registro de usuario

Los campos con "*" son obligatorios

Nombre/s:*

Puede colocar más de un nombre si lo desea

Apellido/s:*

Puede colocar su primer apellido o segundo apellido o ambos

Correo electrónico:*

Número de contacto*

Indique su código

¿Desea agregar su ubicación personal?*

Sí

No

Indique su tipo de usuario por favor* © Easy Digital Commerce 2022

Figura 24: Vista de registro de usuario por defecto

¿Desea indicar la zona en la que efectúa despachos o realiza sus servicios? *

Sí

No

¿En qué área efectúa despachos? *

En todas las comunas de Chile

En un conjunto de comunas que no abarca todas las comunas de Chile

r) Seleccione la/s región/es en la/s que despacha*

XV	I	II	III	IV	V	VI	VII
Región de Arica y Parinacota	Región de Tarapacá	Región de Antofagasta	Región de Atacama	Región de Coquimbo	Región de Valparaíso	Región del Libertador General Bernardo O'Higgins	Región del Maule
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
XVI	VIII	IX	XIV	X	XI	XIII	RM
Región de...	Región del...	Región de...	Región de...	Región de...	Región de...	Región de...	Región...

© Easy Digital Commerce 2022

Figura 25: Campo de cobertura geográfica de despacho en el registro de usuario tipo proveedor

¿Desea agregar sus horarios en los que está activo/a?*

- Sí
 No

¿Desea indicar sus horarios en una tabla estática o dinámica?*

- Tabla estática
 Tabla dinámica

	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado	Domingo
00:00 - 01:00	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
01:00 - 02:00	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
02:00 - 03:00	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
03:00 - 04:00	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
04:00 - 05:00	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
05:00 -	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

© Easy Digital Commerce 2022

Figura 26: Campo de horario de operación en formato de tabla estática en el registro de usuario tipo proveedor

¿Desea agregar sus horarios en los que está activo/a?*

- Sí
 No

¿Desea indicar sus horarios en una tabla estática o dinámica?*

- Tabla estática
 Tabla dinámica

Inicio del intervalo de funcionamiento		Fin del intervalo de funcionamiento		Acciones disponibles				
Horas	Minutos	Horas	Minutos	Añadir fila				
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="button" value="⊕"/>				
Valido para:	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado	Domingo	Añadir tabla dinámica
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="button" value="⊕"/>

Contraseña:*

- Su contraseña no puede asemejarse tanto a su otra información personal.
- Su contraseña debe contener al menos 8 caracteres.
- Su contraseña no puede ser una clave utilizada comúnmente.

© Easy Digital Commerce 2022

Figura 27: Campo de horario de operación en formato de tabla dinámica en el registro de usuario tipo proveedor

D. Anexo D

D.1. Formulario pre-validación

The screenshot shows a pre-validation form with a blue header bar containing navigation icons and labels: 'Búsqueda personalizada', 'Añadir producto o servicio', 'Calificar compras', 'Notificaciones', and 'Cerrar sesión'. The form fields include: 'Nombre' (text input), 'Precio' (text input), 'Imagen' (file selection button 'Seleccionar archivo' and 'Sin...lec.'), 'Tipo de bien que oferta' (dropdown menu), 'Categoría del bien que oferta' (dropdown menu), 'Subcategoría del bien que oferta' (dropdown menu), 'Unidad de medida' (dropdown menu), '¿El bien ya está disponible?' (radio button 'Si'), '¿Podría vender una fracción del bien ofertado?' (radio button 'Si'), and 'Descripción' (text area). A footer bar contains the copyright notice '© Easy Digital Commerce 2022'.

Figura 28: Formulario de registro de bien

D.2. Formulario post-validación

The screenshot shows the post-validation results of the form. Red error messages are displayed above the 'Nombre' and 'Precio' fields: 'Nombre inválido' and 'Precio inválido'. The 'Categoría del bien que oferta' and 'Subcategoría del bien que oferta' dropdown menus are also highlighted in red with error messages: 'Seleccione una categoría válida' and 'Seleccione una subcategoría válida'. Green checkmarks are visible next to the 'Tipo de bien que oferta', 'Unidad de medida', '¿El bien ya está disponible?', '¿Podría vender una fracción del bien ofertado?', and 'Descripción' fields. A blue button labeled 'Añadir producto' is located at the bottom left. The footer bar contains the copyright notice '© Easy Digital Commerce 2022'.

Figura 29: Resultados de la validación

E. Anexo E

E.1. Búsqueda personalizada

Búsqueda personalizada

Añadir producto o servicio

Calificar compras

Notificaciones

Cerrar sesión

Búsqueda personalizada

En caso de no aplicar filtros se muestran las últimas publicaciones subidas a la plataforma.

Filtrar por:

Tipo de bien (producto o servicio)

Precio

Calidad

Puntualidad

Horario de atención

Cobertura geográfica de despachos

Mostrar los resultados en:

un mapa

una lista

Buscar

© Easy Digital Commerce 2022

Figura 30: Formulario por defecto de la búsqueda personalizada

E.2. Filtros de la búsqueda personalizada

Búsqueda personalizada

Añadir producto o servicio

Calificar compras

Notificaciones

Cerrar sesión

Búsqueda personalizada

En caso de no aplicar filtros se muestran las últimas publicaciones subidas a la plataforma.

Filtrar por:

Tipo de bien (producto o servicio)

Precio

Calidad

Puntualidad

Horario de atención

Cobertura geográfica de despachos

Seleccione, si lo desea, la cantidad mínima de estrellas de los bienes a mostrar en pantalla ★★★★★

Seleccione, si lo desea, la cantidad máxima de estrellas de los bienes a mostrar en pantalla ★★★★★

Mostrar los resultados en:

un mapa

una lista

© Easy Digital Commerce 2022

Figura 31: Formulario al activar el filtro de calidad

En caso de no aplicar filtros se muestran las últimas publicaciones subidas a la plataforma.

Filtrar por:

Tipo de bien (producto o servicio) Precio Calidad Puntualidad Horario de atención Cobertura geográfica de despachos

Sólo mostrar a los proveedores cuya zona de operación llegue a mi ubicación personal Sólo mostrar a los proveedores cuya zona de operación llegue a la ubicación a elegir en el mapa

Indique la ubicación por favor



Figura 32: Formulario al activar el filtro de cobertura geográfica


Búsqueda personalizada

En caso de no aplicar filtros se muestran las últimas publicaciones subidas a la plataforma.

Filtrar por:

Tipo de bien (producto o servicio) Precio Calidad Puntualidad Horario de atención Cobertura geográfica de despachos

Rango horario de atención
Hora de inicio -> 11:23 | Hora de término -> 15:19



Día/s de atención

Lunes Martes Miércoles Jueves Viernes Sábado Domingo

Mostrar los resultados en: un mapa una lista

© Easy Digital Commerce 2022

Figura 33: Formulario al activar el filtro de horario de atención

Búsqueda personalizada

En caso de no aplicar filtros se muestran las últimas publicaciones subidas a la plataforma.

Filtrar por:

Tipo de bien (producto o servicio) Precio Calidad Puntualidad Horario de atención Cobertura geográfica de despachos

Precio mínimo: Precio máximo:

Mostrar los resultados en: un mapa una lista

© Easy Digital Commerce 2022

Figura 34: Formulario al activar el filtro de precio

Búsqueda personalizada

En caso de no aplicar filtros se muestran las últimas publicaciones subidas a la plataforma.

Filtrar por:

Tipo de bien (producto o servicio) Precio Calidad Puntualidad Horario de atención Cobertura geográfica de despachos

Selecione, si lo desea, la cantidad mínima de estrellas de los bienes, según la puntualidad de entrega, a mostrar en pantalla ★★★★★

Selecione, si lo desea, la cantidad máxima de estrellas de los bienes, según la puntualidad de entrega, a mostrar en pantalla ★★★★★

Mostrar los resultados en: un mapa una lista

© Easy Digital Commerce 2022

Figura 35: Formulario al activar el filtro de puntualidad

Búsqueda personalizada

En caso de no aplicar filtros se muestran las últimas publicaciones subidas a la plataforma.

Filtrar por:

Tipo de bien (producto o servicio)
 Precio
 Calidad
 Puntualidad
 Horario de atención
 Cobertura geográfica de despachos

Escoja el tipo de bien a mostrar:
 Filtrar por categoría:

Escoja el tipo la categoría a mostrar:
 Filtrar por subcategoría:

Escoja la subcategoría a mostrar:

Mostrar los resultados en: un mapa
 una lista

Figura 36: Formulario al activar el filtro de tipo de bien

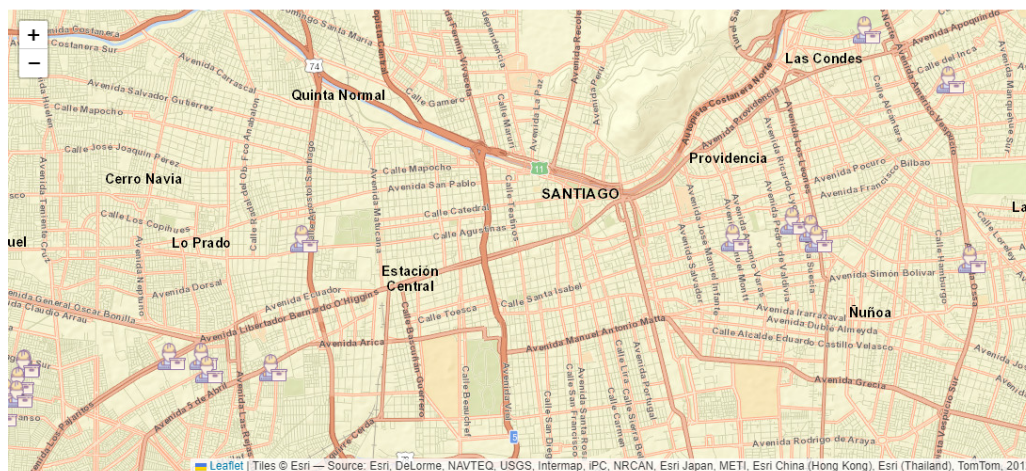
E.3. Resultados de la búsqueda personalizada

Mostrar los resultados en: un mapa una lista

[Buscar](#)

	<p>Nombre: Gasfiteria</p> <p>Precio: \$0 el Indeterminado</p> <p>Descripción:</p> <p>2022-10-18 00:35</p>	 Comprar Calificación del producto 4
	<p>Nombre: Sueter cafe</p> <p>Precio: \$30000 el Unidad</p> <p>Descripción:</p>	 Comprar

Figura 37: Resultados de la búsqueda personalizada en formato de lista



© Easy Digital Commerce 2022

Figura 38: Resultados de la búsqueda personalizada en formato de mapa

F. Anexo F

F.1. Primer paso del proceso de solicitud

1° Paso: Leer la información detallada sobre el bien y el proveedor

33% completado

Tienda de FRANCISCA

Integrantes:

- FRANCISCA Olivares

Horario de atención:

- Lunes: 0:0 a 23:59
- Martes: 0:0 a 23:59
- Miércoles: 0:0 a 23:59
- Jueves: 0:0 a 23:59
- Viernes: 0:0 a 23:59
- Sábado: 0:0 a 23:59
- Domingo: 0:0 a 23:59

Puntualidad de despacho:

- ★★★★★
- 4,0
- Cantidad de veces evaluado: 3

Producto : Sueter cafe

© Easy Digital Commerce 2022

Figura 39: Al solicitar un bien se presentan detalles del bien y del proveedor. En este caso el integrante de la tienda, el horario de atención y la calificación de puntualidad con la que ha sido evaluado.

- 4,0
- Cantidad de veces evaluado: 3

Producto : Sueter cafe

Calidad del Producto:

- ★★★★★
- 3,0
- Cantidad de veces evaluado: 1

Descripción:

Precio: \$30000 por Unidad

No se puede vender una fracción del bien

Ubicación:



Figura 40: Esta imagen corresponde a una continuación de la imagen de la vista anterior. Aquí se muestran la calificación de calidad con la que ha sido evaluado el bien, su precio y la ubicación de la tienda. Para avanzar con el proceso se da clic en el botón siguiente.

F.2. Segundo paso del proceso de solicitud

2° Paso: Determinar cantidad, lugar, fecha y hora

66% completado

Seleccione la cantidad de Unidades:

Seleccione el día de entrega del bien:

Seleccione la hora de la entrega del bien:

Seleccione un tipo de entrega:

Retirar en el local del proveedor Solicitar despacho a domicilio

Figura 41: En esta vista se pide la información necesaria para solicitar el bien. Al solicitar, se pide una confirmación, la que si se da, permite el envío de la solicitud.

G. Anexo G

G.1. Notificaciones



Figura 42: Ventana de notificaciones

G.2. Ventas aceptadas

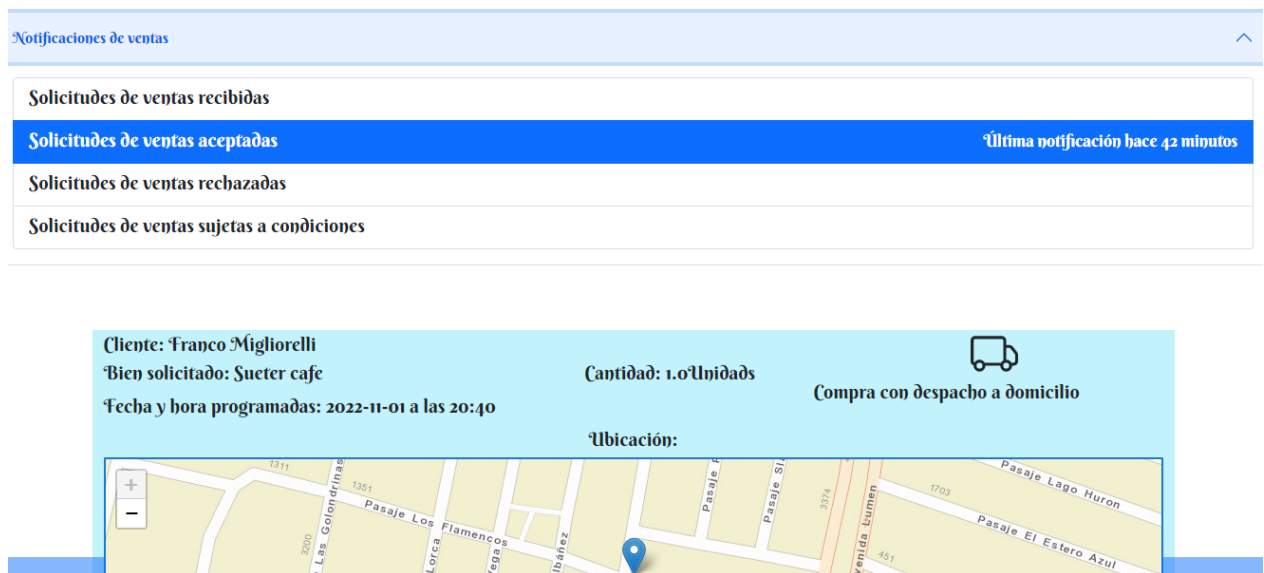


Figura 43: En esta vista se pueden ver las ventas del usuario aceptadas. Incluyen las ventas que aceptó directamente y en las que propuso un horario de llegada y fueron aceptadas por el cliente.

H. Anexo H

H.1. Evaluaciones



Figura 44: Vista por defecto de la ventana de evaluaciones

H.2. Evaluar calidad

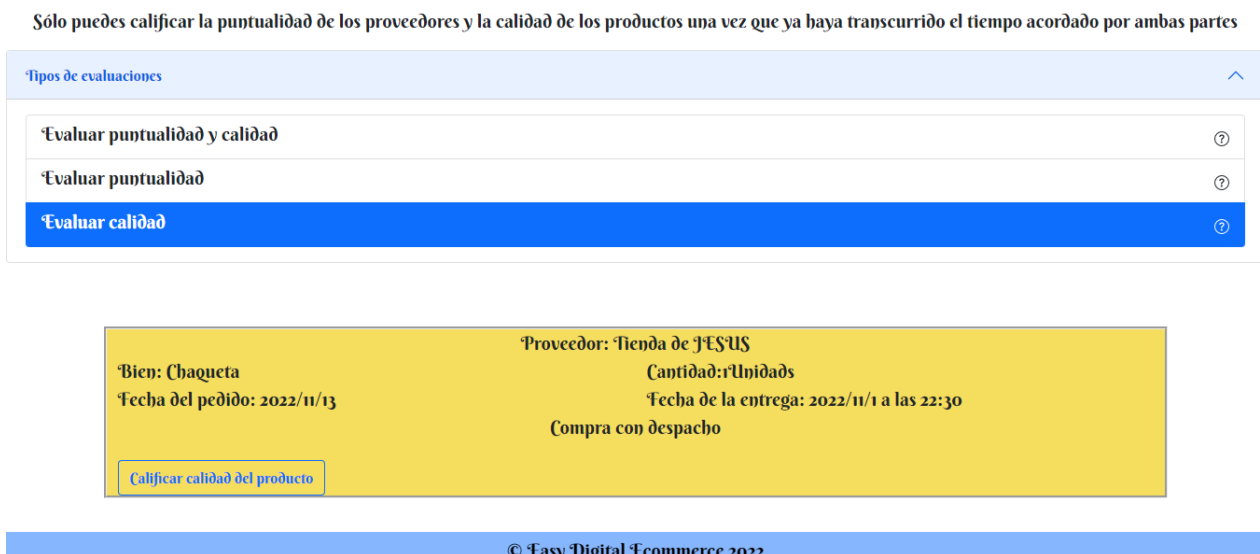
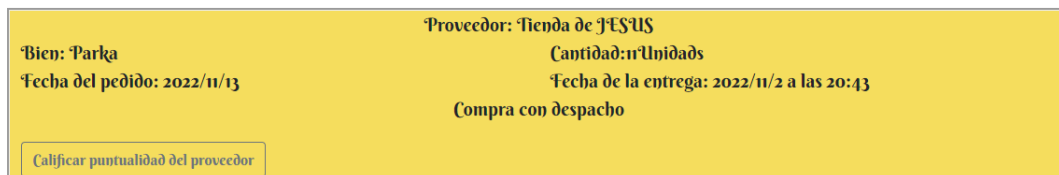
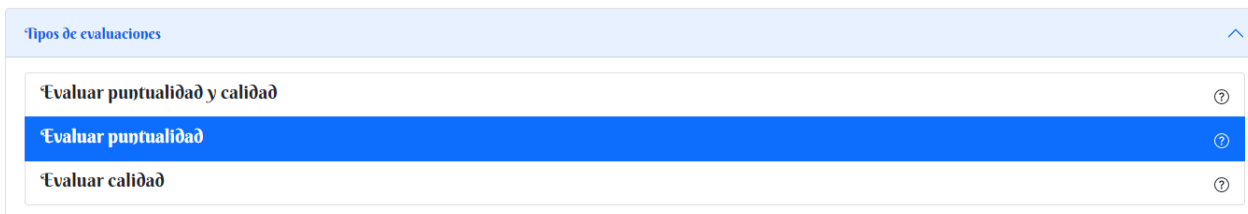


Figura 45: Al seleccionar "Evaluar calidad" se despliega aquel elemento

H.3. Evaluar puntualidad

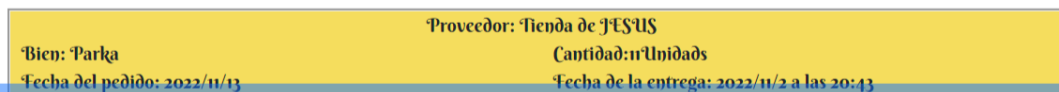
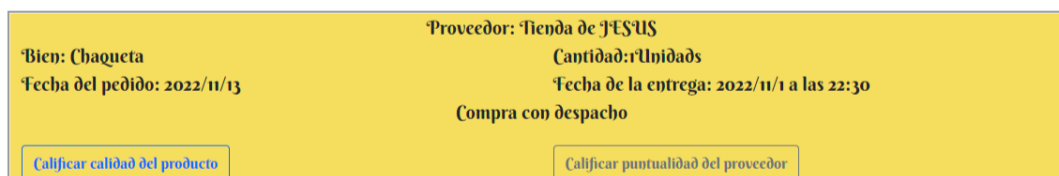
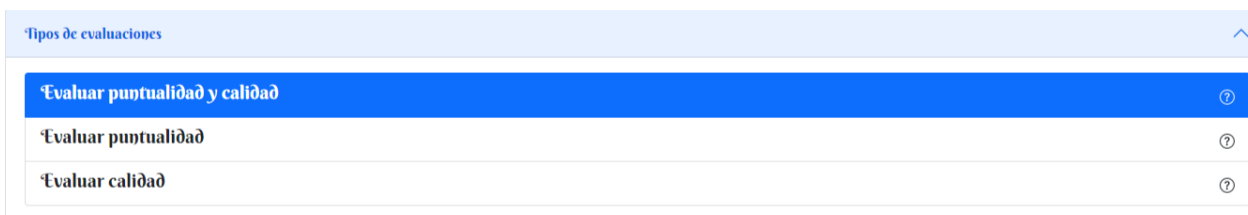
Sólo puedes calificar la puntualidad de los proveedores y la calidad de los productos una vez que ya haya transcurrido el tiempo acordado por ambas partes



© Easy Digital Commerce 2022

Figura 46: Vista que se despliega al seleccionar "Evaluar puntualidad"

H.4. Evaluar calidad y puntualidad



© Easy Digital Commerce 2022

Figura 47: Al seleccionar "Evaluar puntualidad y calidad" se despliegan aquellos elementos

I. Anexo I

I.1. Registro de usuario post-validación

Búsqueda personalizada Iniciar sesión Crear cuenta

Registro de usuario

Los campos con "*" son obligatorios

Nombre/s:* <input type="text"/> <small>Puede colocar más de un nombre si lo desea</small>	Apellido/s:* <input type="text"/> <small>Puede colocar su primer apellido o segundo o ambos apellidos</small>	
Correo electrónico:* <input type="text" value="ejemplo@correo.com"/>	Código <input type="text" value="+562"/>	Número de contacto <input type="text"/>

¿Desea agregar su ubicación personal?*

Sí
 No

Indique su/s tipo/s de usuario por favor* (?)

Cliente
 Proveedor de bienes y/o servicios

Contraseña:*

- Su contraseña no puede asemejarse tanto a su otra información personal.

Figura 48: Vista por defecto del registro de usuario

¿Desea agregar su ubicación personal?*

- Sí
 No

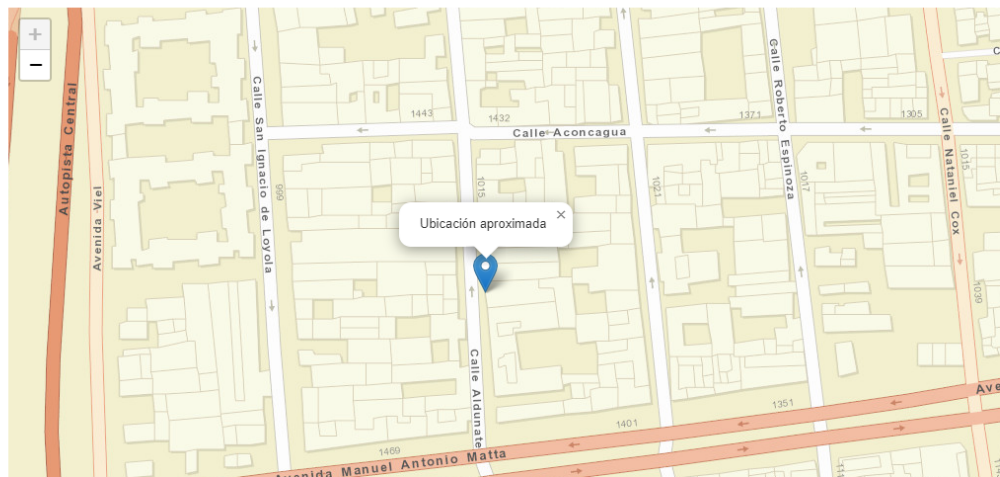



Figura 49: Mapa que se despliega al añadir la ubicación personal

¿Desea agregar su ubicación personal?*

Sí

No

Indique su/s tipo/s de usuario por favor* 

Cliente

Proveedor de bienes y/o servicios

¿Desea añadir un nombre comercial?*

Sí

No

¿Su ubicación personal corresponde a una ubicación de sucursal?*

Sí

No

¿Desea indicar la zona en la que efectúa despachos o realiza sus servicios? *

Sí

No

¿Desea agregar sus horarios en los que está activo/a? *

Sí

No

Contraseña:*

- Su contraseña no puede asemejarse tanto a su otra información personal.
- Su contraseña debe contener al menos 8 caracteres.
- Su contraseña no puede ser una clave utilizada comúnmente.
- Su contraseña no puede ser completamente numérica.


Confirmación de contraseña:*

Figura 50: Campos que se muestran al indicar que el usuario es de tipo proveedor

¿Desea agregar su ubicación personal?*

Sí

No

Indique su/s tipo/s de usuario por favor* 

Cliente

Proveedor de bienes y/o servicios

¿Desea añadir un nombre comercial?*

Sí

No

Nombre comercial:*

¿Su ubicación personal corresponde a una ubicación de sucursal?*

Sí

No

¿Desea añadir más ubicaciones de sucursales? *

Sí

No

Mapa de sucursales




Figura 51: Campos que se despliegan para añadir el nombre comercial y la/s ubicación/es de sucursal/es

¿Desea agregar sus horarios en los que está activo/a?
*

Sí
 No

Inicio del intervalo de funcionamiento		Fin del intervalo de funcionamiento		+
Horas	Minutos	Horas	Minutos	
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	

Valido para: Lunes Martes Miercoles Jueves Viernes Sabado Domingo


Contraseña:*

- Su contraseña no puede asemejarse tanto a su otra información personal.
- Su contraseña debe contener al menos 8 caracteres.

Figura 52: Campo que se despliega para indicar el horario de atención de la tienda. En este caso sólo se muestra la tabla dinámica, ya que el único participante que indicó su horario, sólo utilizó esta.

- Su contraseña no puede asemejarse tanto a su otra información personal.
- Su contraseña debe contener al menos 8 caracteres.
- Su contraseña no puede ser una clave utilizada comúnmente.
- Su contraseña no puede ser completamente numérica.

Confirmación de contraseña:*



Errores en los campos:

- Nombre
- Apellido
- Contraseña
- Nombre comercial
- Ubicación comercial
- Correo electrónico

Figura 53: Este tipo de vista se ofrece al intentar mandar un formulario con errores.

I.2. Búsqueda personalizada post-validación

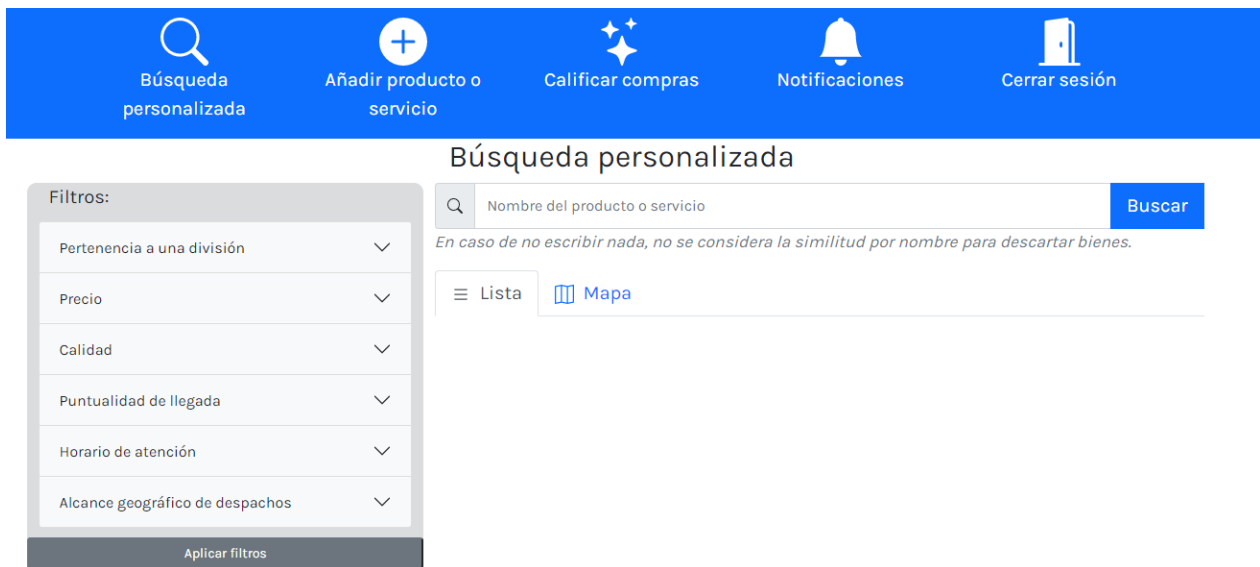


Figura 54: Vista que se presenta por defecto al visitar la ventana de búsqueda personalizada

Filtros:

Pertenencia a una división ⤴

Tipo de bien: ⤴ ⤵ 🗑️ ?

Precio ⤴

Mínimo: \$1000 - Máximo: \$120.000

Calidad ⤴

🗑️ ★ ★ ★ ★ ★ ?

Puntualidad de llegada ⤴

🗑️ ★ ★ ★ ★ ★ ?

Horario de atención ⤴

Seleccione una fecha para descartar las tiendas que no comercian en dicha fecha.

Día: 📅 🗑️

Alcance geográfico de despachos ⤴

Determine una ubicación geográfica para descartar a los proveedores que no llegan a aquella ubicación.

📍 **Coordenadas: -,-**

Seleccionar ubicación en el mapa desplegable

Seleccionar ubicación personal

Figura 55: Filtros de la búsqueda personalizada desplegados

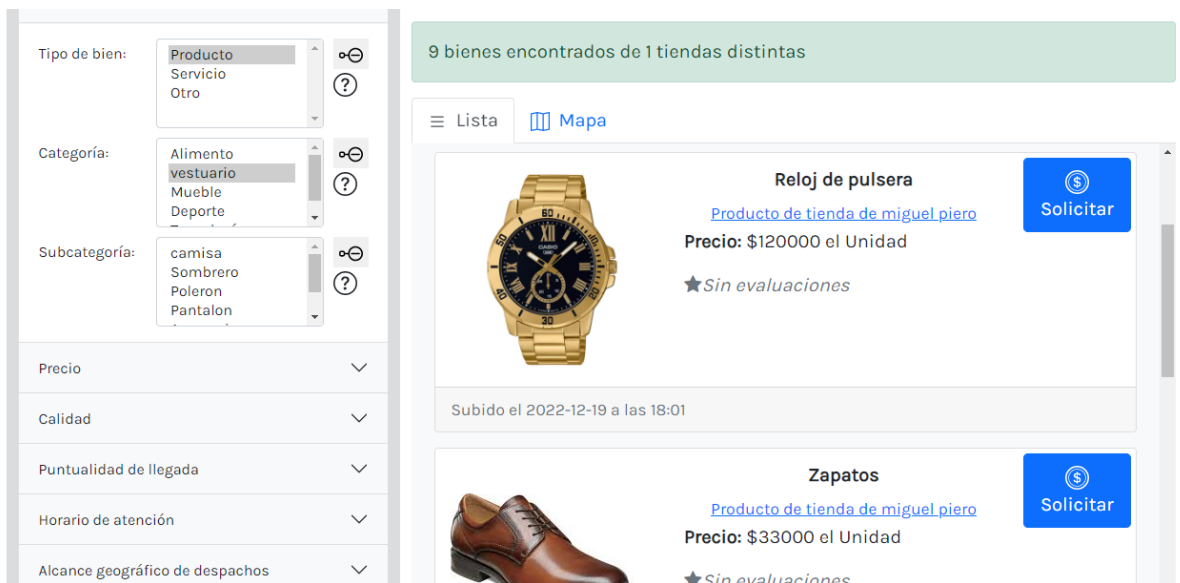


Figura 56: Lista de resultados al realizar una búsqueda aplicando el filtro que permite mostrar sólo los bienes de la categoría vestuario

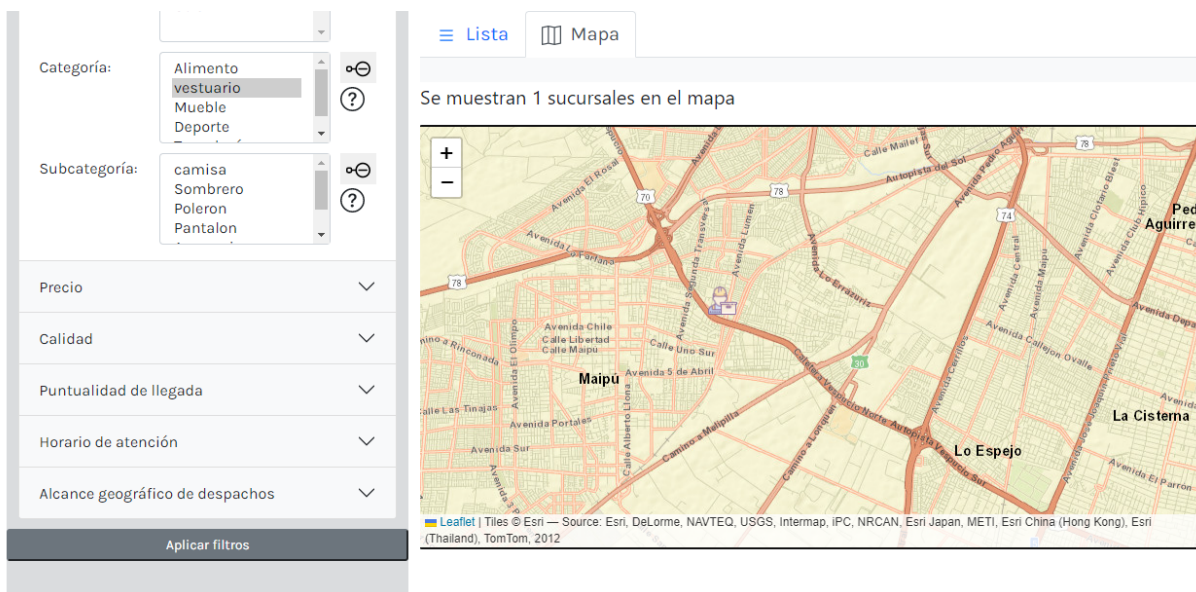


Figura 57: El filtro es el mismo al aplicado en la consulta anterior, con la diferencia de que se muestra la ubicación de la tienda que ofrece al menos un bien que cumple con el filtro

I.3. Registro de bien post-validación

Plataforma de geolocalización de clientes y vendedores

Búsqueda personalizada Añadir producto o servicio Calificar compras Notificaciones Cerrar sesión

Nombre del bien Precio Imagen
[input] [input] [Seleccionar archivo] [Ni..elec.]

¿El bien se puede entregar a domicilio? ¿El bien se puede retirar personalmente en la tienda?
[SÍ] [SÍ]

Tipo de bien que oferta Categoría del bien que oferta Subcategoría del bien que oferta
[Seleccione un tipo] [input] [input]

Unidad de medida ¿El bien ya está disponible? ¿Podría vender una fracción del bien ofertado?
[input] [SÍ] [SÍ]


Descripción
[input]

Figura 58: Vista de registro de bien por defecto. Se observa que hay dos nuevas preguntas en el formulario.

Tipo de bien que oferta Categoría del bien que oferta Subcategoría del bien que oferta
[Seleccione un tipo] ✓ [input] ⓘ [input] ⓘ
¡Bien hecho! La categoría es obligatoria La subcategoría es obligatoria

Unidad de medida ¿El bien ya está disponible? ¿Podría vender una fracción del bien ofertado?
[input] ⓘ [SÍ] ✓ [SÍ] ✓
La unidad de medida es obligatoria ¡Bien hecho! ¡Bien hecho!

Descripción
[input] ✓



- El nombre es obligatorio
- El precio es obligatorio
- La imagen es obligatoria
- La categoría es obligatoria
- La subcategoría es obligatoria
- La unidad de medida es obligatoria

Figura 59: La validación incluye un cuadro de advertencia que indica los campos con problemas.

I.4. Evaluaciones de compras post-validación



Figura 60: Vista por defecto de la ventana de evaluaciones

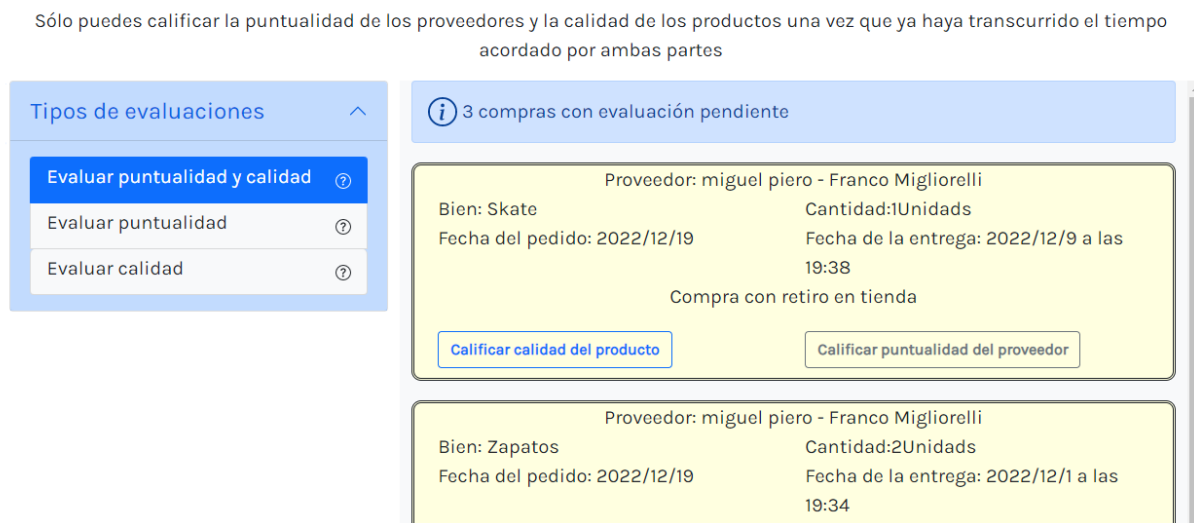


Figura 61: En esta sección se presenta una vista para evaluar la calidad y la puntualidad de las compras

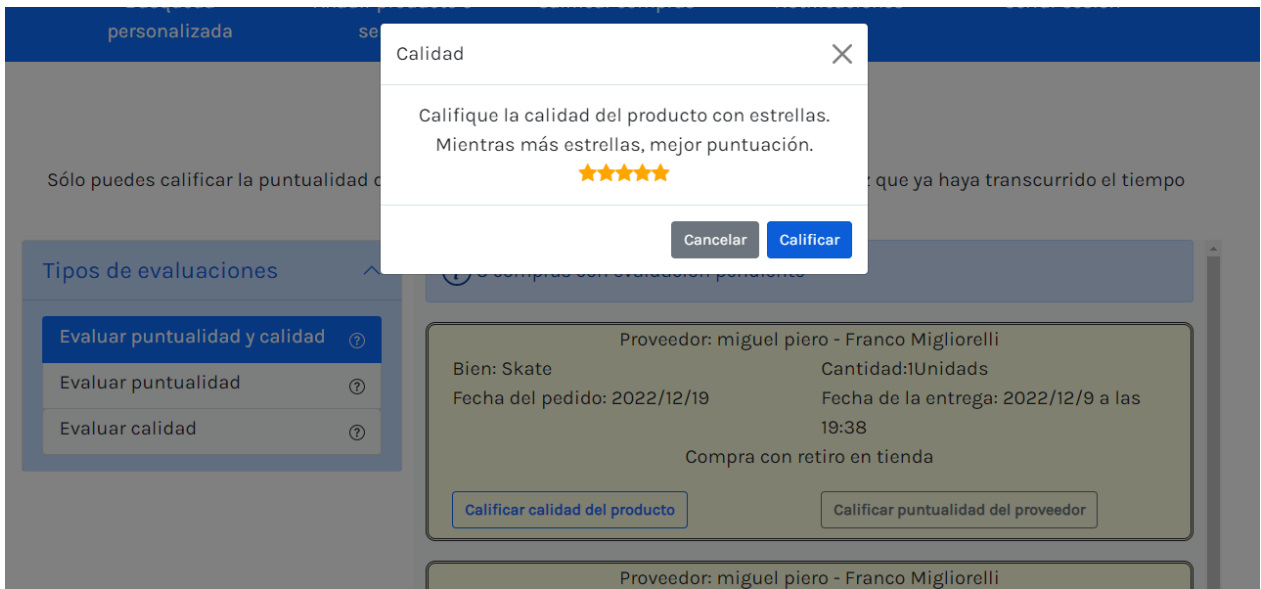


Figura 62: Al calificar se despliega una pequeña ventana emergente con ese único objetivo.

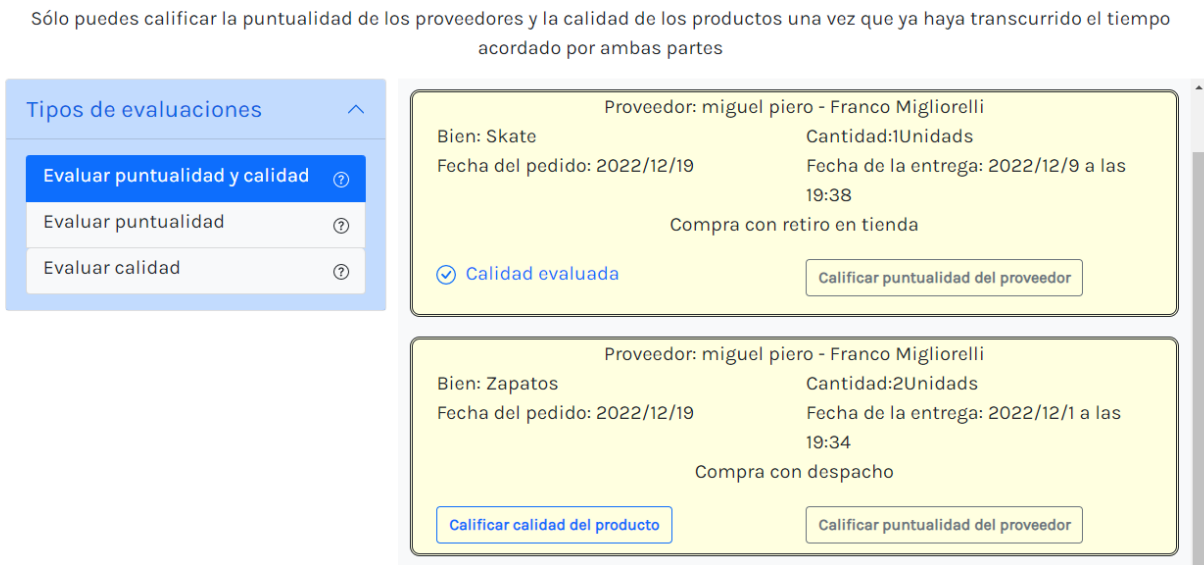


Figura 63: Se observa que después de evaluar por calidad una compra, un ticket reemplaza al botón. Así se logra que no se evalúe dos veces y que el usuario sepa que se envió su evaluación.

I.5. Proceso de solicitud de bien post-validación



Figura 64: Se modificó el proceso de solicitud de bien. En la interfaz sólo se muestra información concerniente al producto y en caso de saber sobre el proveedor, se presenta un link hacía su tienda desde la cual se puede saber más.



Figura 65: En esta vista no se aplicaron modificaciones importantes.

I.6. Tienda post-validación



Figura 66: Al visitar una tienda se presenta su información y sus bienes.



Figura 67: Su información desplegada luce así