



UNIVERSIDAD DEL BÍO-BÍO
FACULTAD DE CIENCIAS EMPRESARIALES
ESCUELA DE INGENIERIA CIVIL INFORMATICA

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

Carlos Pérez Fuentealba

Profesor Guía

Luis Gajardo Diaz

Memoria para optar a título de Ingeniero Civil en Informática

Chillán, junio de 2019

Agradecimientos

Primero que todo, agradecer a mi familia, mis padres Verónica y Jaime, a mi mamá Tere y a mi hermana Daniela que me han dado su apoyo y confianza a lo largo de todo mi proceso universitario.

Además, agradecer a mi polola Alexandra que ha sabido darme su apoyo incondicional para salir adelante en especial los dos últimos años de carrera que fueron más difíciles.

Agradecer también a mis amigos y Compañeros que me han dado ánimos en momentos de complicaciones.

A mi cuñado Luis que ha dedicado de su tiempo para apoyarme en este proceso.

Finalmente, a la universidad y sus docentes por formarme como un profesional competente.

Dedicatoria

Principalmente a mi familia que me ha formado como persona íntegra y me han dado la posibilidad de estudios.

A mi polola por su compañía permanente e incondicional.

Y a todos los que creyeron en mi meta de estudios.

Resumen

El presente informe se lleva a cabo para optar al título de Ingeniero Civil en Informática. El proyecto lleva por nombre “Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies”, en este informe detallará el cómo se identifica el problema, cómo se desarrolla la solución, la metodología de trabajo, la factibilidad de desarrollar la solución y toda la documentación pertinente del proyecto. Todo esto orientado a los comercios de grandes superficies, particularmente supermercados.

Se ha detectado que en gran parte de los supermercados se encuentran organizaciones de empaquetadores, los cuales se encargan de ayudar a los clientes al momento de su compra, ya sea empaquetando sus cosas o depositándolas en el carro. Normalmente los supermercados no se hacen responsables de estas organizaciones de una manera formal. Es por ello que los supermercados mantienen un contacto con uno o más personas que son los “encargados” de estas organizaciones. Básicamente el problema radica en la gestión de estas organizaciones, ya que carecen de herramientas tecnológicas que permiten un mejor manejo del grupo de empaquetadores. Generalmente los miembros de estas organizaciones deben gastar tiempo de al menos uno de sus días para tomar sus turnos (no todos realizan la toma de turnos presencial), tiempo que se podría invertir en otra actividad. La gestión de la información de estas organizaciones se lleva en papel o en planillas Excel, lo cual hace que la información no esté disponible a la brevedad para los encargados.

De acuerdo a lo antes mencionado, es que realizar una aplicación que abarque los aspectos de la problemática parece ser una óptima solución. Sería una herramienta muy valorada tanto para los encargados que van a ver como mejora su registro y acceso a la información de su organización, como también beneficia a los empaquetadores permitiendo que puedan realizar sus tomas de turnos de manera remota desde cualquier lugar a solo un par de toques en sus SmartPhone.

La aplicación es desarrollada siguiendo la metodología evolutiva incremental, ya que, durante el proceso de construcción de este software, se irán generando prototipos que conforme a las entregas tendrán incrementos significativos. Esta metodología se considera apropiada debido al tiempo que se posee para realizar el proyecto.

En pocas palabras, la realización del proyecto es factible de acuerdo a estudios realizados y además entrega valor a sus usuarios simplificando sus labores tanto de encargado como de empaquetador.

Abstract

This report is carried out to apply for the degree of Civil Engineer in Computer Science. The project is called “Mobile application for packager management in large-area stores”, in this report it will detail how the problem is identified, how the solution is developed, the work methodology, the feasibility of developing the solution and all the relevant documentation of the project. All this oriented to the stores of big surfaces, particularly supermarkets.

It has been detected that in most of the supermarkets there are organizations of packers, which are responsible for helping customers at the time of purchase, either packing their things or depositing them in the car. Normally supermarkets are not responsible for these organizations in a formal way. That is why supermarkets maintain contact with one or more people who are the "managers" of these organizations. Basically the problem lies in the management of these organizations, since they lack technological tools that allow a better handling of the group of packers. Generally, members of these organizations must spend time on at least one of their days to take their turns (not all of them take face-to-face shifts), time that could be invested in another activity. The information management of these organizations is carried out on paper or in Excel spreadsheets, which makes the information not available soon to those in charge.

According to the aforementioned, it is that making an application that covers aspects of the problem seems to be an optimal solution. It would be a highly valued tool for both those who are going to see how it improves their registration and access to the information of their organization, as well as benefits to the packers allowing them to take their turns taking remotely from anywhere to just a couple of touches on your SmartPhone.

The application is developed following the incremental evolutionary methodology, since, during the process of building this software, prototypes will be generated that will increase significantly according to the deliveries. This methodology is considered appropriate due to the time taken to complete the project.

In short, the realization of the project is feasible according to studies carried out and also delivers value to its users by simplifying their work as both manager and packager.

Tabla de Contenido

Introducción.....	11
Capítulo 1 Descripción del problema	13
1.1 Descripción de la situación actual	14
1.2 Descripción de la propuesta	15
Capítulo 2 Definición del proyecto.....	16
2.1 Objetivos del proyecto	17
2.1.1 Objetivo general	17
2.1.2 Objetivos específicos	17
2.2 Modelo de negocio	18
2.2.1 Modelo de negocio actual	18
2.2.2 Modelo de negocio propuesto	19
2.3 Ambiente de ingeniería de software	20
2.3.1 Metodología de desarrollo	20
2.3.2 Estándares	21
2.4 Modelos	23
2.4.1 Modelo entidad relación	23
2.4.2 Herramientas.....	25
2.5 Arquitectura de software	27
Capítulo 3 Especificación de requerimientos de software.....	28
3.1 Propósito.....	29
3.2 Alcances y limitaciones.....	30
3.3 Requerimientos específicos.....	31
3.3.1 Requisitos funcionales	31
3.3.2 Requisitos no funcionales	32
3.4 Encuesta.....	33
3.4.1 Definición de encuesta	33
3.4.2 Características de encuesta	33
3.4.3 Encuesta a Jumbo Chillán	34
Capítulo 4 Tecnologías Emergentes	35
4.1 Framework Angular	36
4.1.1 Arquitectura	36
4.1.2 Módulos de Angular	37

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

4.1.3 Componentes en Angular	38
4.1.4 Servicios en Angular	38
4.2 Framework Ionic	39
4.2.1 Metas de Ionic.....	39
4.2.2 Licencia.....	39
4.2.3 Compatibilidad de marco	40
4.3 Bases de datos no relacionales.....	41
4.3.1 ¿Qué son las bases de datos NoSQL?.....	41
4.3.2 ¿Cuándo usar base de datos no relacionales?	42
4.3.3 Ventajas de los sistemas NoSQL	42
4.3.4 Firebase.....	43
4.4 Firebase en el proyecto	45
Capítulo 5 Estudio de factibilidad.....	47
5.1 Factibilidad técnica.....	48
5.2 Factibilidad operativa	48
5.3 Factibilidad económica.....	49
5.4 Conclusión de factibilidad	51
Capítulo 6 Análisis.....	52
6.1 Casos de uso.....	53
6.1.1 Actores	53
6.1.2 Diagrama de casos de uso	54
6.1.3 Especificación de casos de uso	55
Capítulo 7 Diseño.....	63
7.1 Diseño Front-end	64
7.1.1 MockUp.....	64
7.1.2 Diseño de interfaz gráfica	66
7.1.3 Diseño de navegación.....	69
7.2 Diseño Back-end.....	71
7.2.1 Modelo físico de base de datos Firebase	71
7.2.2 Diseño de Servidor Web	74
Capítulo 8 Pruebas	75
8.1 Pruebas unitarias	76
8.1.1 escenarios de prueba	76

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

8.1.2 Metodología de pruebas	76
8.1.3 Pruebas funcionales	77
8.2 Pruebas de rendimiento	80
8.3 Pruebas de usuario.....	82
8.3.1 Usabilidad.....	82
8.3.2 Eficiencia	83
8.4 Conclusiones de pruebas.....	83
Conclusiones.....	84
Trabajos futuros	85
Bibliografía	86
Anexo 1: Encuesta a Jumbo Chillán.....	87
Encuesta aplicada a empaquetadores Jumbo Chillan	87
Encuesta aplicada a encargados de organización Jumbo Chillan	93
Anexo 2: MockUps	99
Anexo 3: Diseño de interfaz gráfica.....	103
Anexo 4: Tamaño en Bytes por usuario registrado	113
Anexo 5: Pruebas de usuario, Jumbo Chillán	114

Índice de Ilustraciones

Ilustración 1: Proceso de negocio, toma de turno	18
Ilustración 2: Proceso de negocio, toma de turno propuesta.	19
Ilustración 3: Modelo entidad relación.....	23
Ilustración 4: Arquitectura de Angular.....	37
Ilustración 5: Base de datos Firebase, parte 1	45
Ilustración 6: Base de datos Firebase, parte 2	46
Ilustración 7: Fórmula VAN.....	50
Ilustración 8: Diagrama de casos de uso.....	54
Ilustración 9: Vista de la aplicación, Formulario de registro.....	66
Ilustración 10: Vista de la aplicación, Ajuste de planilla	67
Ilustración 11: Vista de la aplicación, Toma de turnos para empaques.....	68
Ilustración 12: Diseño de navegación, usuario encargado	69
Ilustración 13: Diseño de navegación, usuario empaquetador	70
Ilustración 14: Modelo físico de Firebase, parte 1	72
Ilustración 15: Modelo físico de Firebase, parte 2	73
Ilustración 16: Arquitectura de módulos y componentes	74
Ilustración 17: Prueba de rendimiento con JMeter.....	82
Ilustración 18: Encuesta a empaquetador, pregunta 1	88
Ilustración 19: Encuesta a empaquetador, pregunta 2	89
Ilustración 20: Encuesta a empaquetador, pregunta 3	90
Ilustración 21: Encuesta a empaquetador, pregunta 4	91
Ilustración 22: Encuesta empaquetador, pregunta 5.....	92
Ilustración 23: Encuesta empaquetador, pregunta 6.....	93
Ilustración 24: Encuesta encargados, pregunta 1	94
Ilustración 25: Encuesta encargados, pregunta 2	95
Ilustración 26: Encuesta encargados, pregunta 3	96
Ilustración 27: Encuesta encargados, pregunta 4	97
Ilustración 28: Encuesta encargados, pregunta 5	98
Ilustración 29: MockUp Login (ambos usuarios).	99
Ilustración 30: MockUp, Listado de empaques.	99
Ilustración 31: MockUp, Formulario para agregar empaquetador.	100
Ilustración 32: MockUp, Scanner QR para control de asistencia.	100
Ilustración 33: MockUp, Toma de turnos (ambos usuarios).	101
Ilustración 34: MockUp, Generar informe.	101
Ilustración 35: MockUp, Acerca de.....	102
Ilustración 36: Interfaz gráfica, Login.....	103
Ilustración 37: Interfaz gráfica, Home.....	104
Ilustración 38: Interfaz gráfica, Generar listado.....	104
Ilustración 39: Interfaz gráfica, Ver listado	105
Ilustración 40: Interfaz gráfica, Lista de empaques.....	106
Ilustración 41: Interfaz gráfica, Actualizar información de empaquetador.	107
Ilustración 42: Interfaz gráfica, Menú encargado	108

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

Ilustración 43: Interfaz gráfica, lector código Qr.....	109
Ilustración 44: Interfaz gráfica, informe de asistencia.	110
Ilustración 45: Interfaz gráfica, toma de turno usuario encargado	111
Ilustración 46: Interfaz gráfica, Acerca de	112
Ilustración 47: Firebase, Campos (atributos) de usuarios	113

Índice de Tablas

Tabla 1: Requisitos funcionales.	31
Tabla 2: Requisitos no funcionales.	32
Tabla 3: Factibilidad económica, gastos.	49
Tabla 4: Factibilidad económica, beneficios.	50
Tabla 5: Factibilidad económica, índice de descuento y periodo de tiempo	50
Tabla 6: Inversión, gastos, beneficios y flujos por año	51
Tabla 7: Resultado de cálculo de VAN	51
Tabla 8: Caso de uso, Login	55
Tabla 9: Caso de uso, toma de turnos.....	56
Tabla 10: Caso de uso, informe mensual.	57
Tabla 11: Caso de uso, Gestionar empaque.....	58
Tabla 12: Caso de uso, generar listado	59
Tabla 13: Caso de uso, ingresar falta a turno	60
Tabla 14: Caso de uso, modificar planilla de turnos.....	61
Tabla 15: Caso de uso, escanear código QR.....	62
Tabla 16: Prueba unitaria, Login.....	77
Tabla 17: Prueba unitaria, registrar usuario (empaque)	77
Tabla 18: Prueba unitaria, tomar turno	78
Tabla 19: Prueba unitaria, configurar planilla.	78
Tabla 20: Prueba unitaria, escanear código QR.	79
Tabla 21: Pruebas de usuario, encargados.	83
Tabla 22: Pruebas de usuario, empaques.	83

Introducción

Es muy habitual, a la hora de ir a un supermercado, ver a jóvenes que prestan a los clientes el servicio de ayuda con su mercadería, ya sea dejando los productos en el carro o empaquetando en bolsas los artículos de los clientes.

Esto favorece en gran medida a los supermercados, por el hecho de que agilizan el proceso de pago, ya que el cliente sólo se centra en pagar y no de empaquetar sus productos. Sin embargo, los supermercados no se encargan de pagarles un sueldo ni de gestionar a este grupo de jóvenes (llamados “empaquetadores”), por lo tanto, deben existir “encargados” de dichas organizaciones.

Gestionar una organización requiere de trabajo y responsabilidades que el encargado debe cubrir, entre otras funciones es responsable de registrar a los integrantes de la organización, distribuir los turnos, sancionar y dejar constancia de faltas o cambios de turnos entre los empaquetadores.

En Chile, se ha buscado regularizar el tema de los empaquetadores en los comercios de grandes superficies (Supermercados), con el fin de no tener trabajando a jóvenes sin alguna supervisión de dichas empresas.

Este tipo de trabajos es principalmente desarrollado por estudiantes universitarios que ven en este oficio de empaquetador un ingreso económico fácil, compatible y flexible con sus horarios de estudio y clases.

Es por ello que se propone una herramienta que ayude a realizar la labor del encargado de una manera más rápida y simple. Es ahí donde una aplicación móvil es ideal para suplir y amortiguar la carga del encargado.

El presente documento tiene como fin explicar el proyecto de título de una aplicación móvil tanto para encargados como para empaquetadores, detallando los beneficios para estos dos. Con respecto a los encargados, mejorar la gestión del personal y generar informes como de asistencia son funcionalidades que la aplicación hará por él. Mientras que los empaquetadores podrán realizar su “toma de turnos” de manera más práctica y ordenada desde sus Smart Phone.

Cada capítulo del presente informe detallará el desarrollo del proyecto de la aplicación, en el capítulo uno, pasando por el problema detectado y la propuesta de solución.

En el capítulo dos, el objetivo general y los específicos serán expuestos. Los modelos de procesos de negocio se demuestran con ilustraciones y explicaciones. La metodología de desarrollo, que para este caso se utilizó iterativo incremental. Para finalmente mencionar y explicar las herramientas utilizadas, los estándares en los que se rige el desarrollo de la aplicación y el modelo entidad relación.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

En el capítulo tres, la especificación de requerimientos del software, los alcances y limitaciones que tendrá la aplicación, los requerimientos específicos (funcionales y no funcionales) serán detallados más adelante.

En el capítulo cuatro, se exponen las tecnologías emergentes que se usan para el desarrollo de la aplicación. Es un ítem que se desea exponer en mayor profundidad debido a su relevancia e innovación a la hora de desarrollar.

En el capítulo cinco, se realiza el estudio de la factibilidad técnica, operativa y la económica, con su respectiva conclusión que no deben faltar en ningún proyecto para asegurar que sea rentable su realización.

En el capítulo seis, se hace un análisis que se demuestra con diagramas de caso de uso, y modelo de datos, y por último el diseño, tanto de la interfaz gráfica como del servidor web.

Finalmente, las pruebas y conclusiones cierran el desarrollo del informe.

Capítulo 1

Descripción del problema

1.1 Descripción de la situación actual

Los comercios de grandes superficies se clasifican en distintos tipos, ya sean de Retail, supermercados, etc. Para este caso, la aplicación móvil a desarrollar estará enfocada en supermercados, ya que en ellos participan los empaquetadores.

Según el INE (Instituto Nacional de Estadísticas), en la región del Biobío existen 173 establecimientos de supermercados. (A.G., 2019)

Estos establecimientos permiten que se conformen organizaciones de personas (generalmente estudiantes universitarios) que son gestionadas o representadas por uno o más individuos (encargados) de la misma organización.

Los establecimientos se desvinculan de estas organizaciones con respecto a remuneraciones, independiente de que ellas estén indirectamente prestando un servicio al supermercado.

Los supermercados son comercios ideales para estas organizaciones, ya que se les necesita mucho más, debido a que las compras son con mayor cantidad de artículos por persona (generalmente).

Un catastro de la cantidad de supermercados a nivel país Chile, especifica que existen 1371 salas de estos comercios. Se estima que, en promedio, el número de empaquetadores es de 30 por cada establecimiento. Los datos son proporcionados por el INE.

En vista de la estimación de la cantidad de empaquetadores por supermercado, y en base a experiencias, se puede detectar que la gestión de estos empaquetadores se lleva a cabo en la mayoría de los casos con documentos en papel, ya sea con currículum, libros de asistencia, hojas con planillas para tomar turnos e incluso los datos de cada empaquetador (nombre completo, Rut, establecimiento, id empaquetador, carrera, etc.) se encuentra usualmente en planillas Excel. Cabe mencionar que la toma de turnos consiste en reunir al grupo de empaquetadores para que, en orden, cada uno pueda hacer elección de los turnos a los cuales desea asistir. Esto es generalmente en forma presencial, lo cual significa que al menos una vez a la semana se debe invertir tiempo para asistir a esta reunión para la toma de turnos.

1.2 Descripción de la propuesta

Este proyecto plantea como solución al problema detectado, el desarrollo de una aplicación móvil que optimiza la forma en como los encargados (persona que gestiona la organización de empaquetadores) administran la organización. Dentro de las mejoras se puede observar, la inexistencia del uso del papel para el registro de los datos de empaquetadores, ya que la aplicación contiene toda la información que se necesita para gestionar al grupo de personas. Otro punto a favor es la modernización de la acción de toma de turnos, que permitirá hacer que ésta deje de ser presencial, es decir, los empaquetadores podrán pedir sus turnos desde sus Smart Phone. Quizás, la funcionalidad más atractiva y novedosa se encuentra en incorporar un lector de código QR, con el cual se controlará la asistencia a los turnos de cada empaquetador, para así poder generar informes que serán de utilidad para los encargados.

Básicamente, la propuesta busca suplir la forma en cómo se gestionan estas organizaciones, modernizando los procesos y la forma de tratar y registrar la información.

Las funcionalidades de la propuesta son:

- Agregar, editar y eliminar un miembro de la organización.
- Los empaquetadores podrán seleccionar sus turnos semanales desde sus SmartPhone desde cualquier lugar.
- Escáner QR para el control de asistencia.
- Logearse en el sistema, para diferenciar entre tipos de usuarios.
- Generar un listado con los nombres de los empaquetadores ordenados de forma aleatoria para que tomen sus turnos en ese determinado orden.
- Registrar mediante lectura QR las faltas a turnos.
- Generar listado con detalle de control de asistencia.
- Modificar o seleccionar una planilla semanal con los horarios que acomoden a las distintas organizaciones.

Capítulo 2

Definición del proyecto

2.1 Objetivos del proyecto

2.1.1 Objetivo general

Crear una aplicación móvil para gestionar y realizar toma de turnos para empaques en comercios de grandes superficies.

2.1.2 Objetivos específicos

- Lograr un registro de los datos personales de cada empaquetador con el fin de tener la información más a mano y concisa.
- Controlar la asistencia y retrasos de los empaquetadores a su lugar de trabajo.
- Optimizar el tiempo para acción de la toma de turno, al realizarla con un par de “click’s” (toques en el dispositivo móvil) desde los Smart Phone de cada empaquetador.

2.2 Modelo de negocio

2.2.1 Modelo de negocio actual

El principal proceso es la toma de turnos, consiste básicamente en hacer participar a empaquetadores y encargado. Primero que todo, se reúnen en un lugar previamente acordado, para que luego cada empaquetador deposite su credencial en una bolsa que tendrá el encargado, posterior a eso, se extraen las credenciales de los empaquetadores en forma consecutivas una tras otra y de manera al azar. Si la credencial es de un determinado empaque, este tiene la facultad para poder tomar sus turnos eligiendo los que desee y estén disponibles. Si la credencial extraída no es la de determinado empaque, este deberá esperar hasta que su credencial sea sacada por el encargado.

El principal proceso de negocio actual es representado por la Ilustración 1: Proceso de negocio, toma de turno.

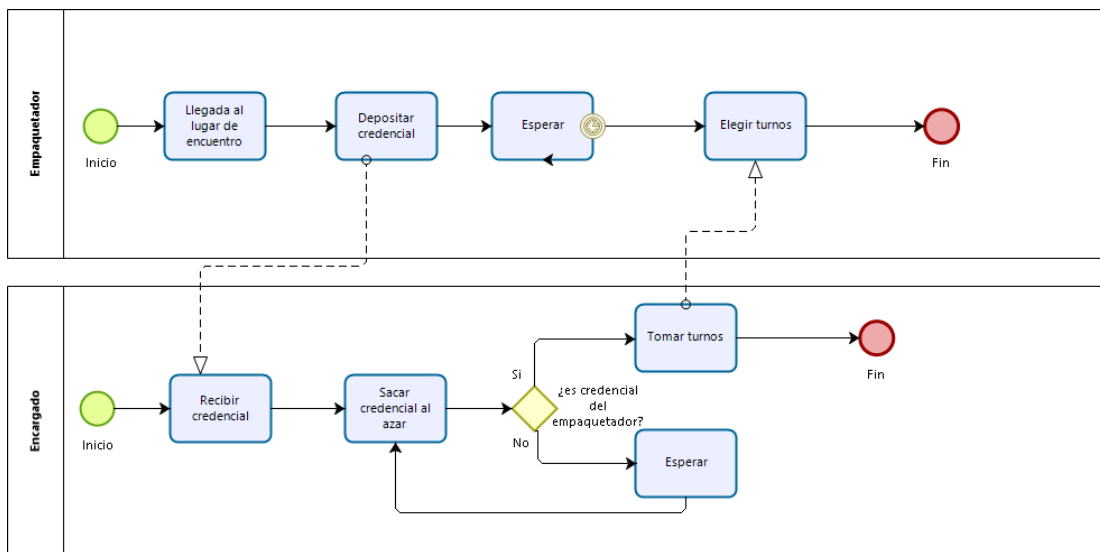


Ilustración 1: Proceso de negocio, toma de turno.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

2.2.2 Modelo de negocio propuesto

El proceso de toma de turnos propuesto consiste en realizar la acción mediante la aplicación desarrollada, la cual permite que esta sea llevada a cabo de manera no presencial o sin fijar un punto de encuentro como se hacía anteriormente¹.

El proceso es llevado a cabo por dos actores, el encargado y el empaquetador. La secuencia es la siguiente: El encargado publica un listado con el número de credencial (id) de los empaquetadores ordenados de forma aleatoria, para que un día y hora determinada los empaquetadores sepan en qué posición o momento les tocará seleccionar sus turnos para la semana. Luego el empaquetador revisa dicho listado y espera su momento para tomar los turnos hasta que le corresponde, para finalmente seleccionar los turnos que desea. Luego, el encargado debe publicar la plantilla semanal con los turnos tomados por los empaquetadores. El principal proceso de negocio de la propuesta es representado por la Ilustración 2: Proceso de negocio, toma de turno propuesta.

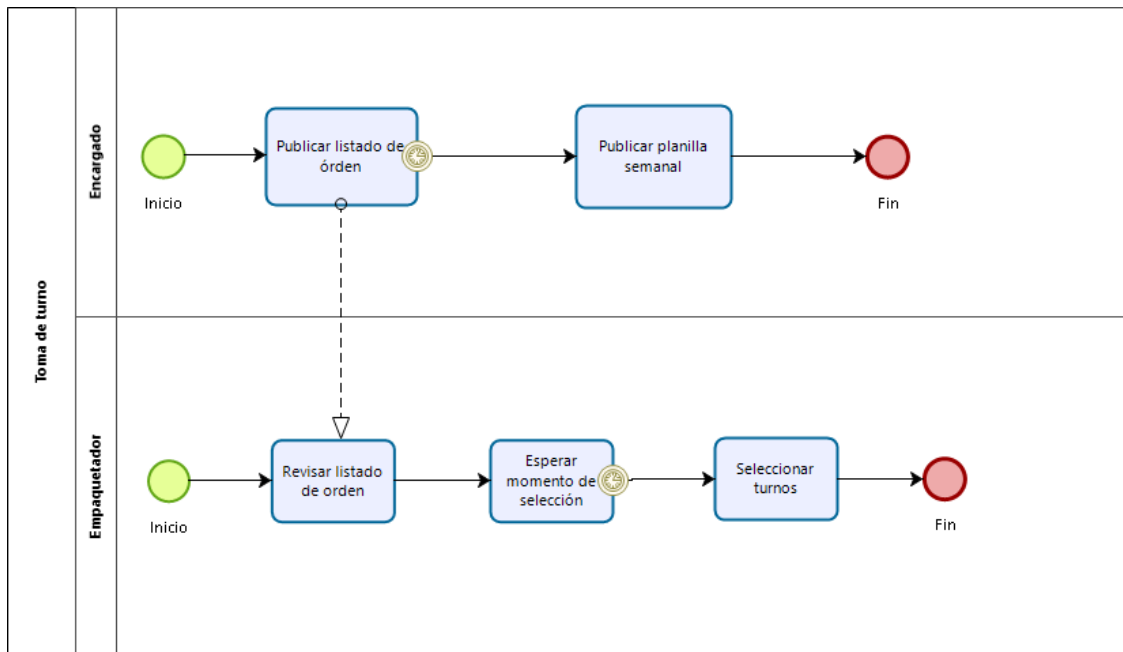


Ilustración 2: Proceso de negocio, toma de turno propuesta.

¹ Forma anterior de toma de turno, ver 2.2.1 Modelo de negocio actual, capítulo 2.

2.3 Ambiente de ingeniería de software

2.3.1 Metodología de desarrollo

Con respecto a la metodología de Desarrollo, se ha optado por utilizar un modelo iterativo incremental, ya que se implementarán versiones de la aplicación cada vez más completas llamadas “iteraciones” que para efectos del desarrollo del proyecto serán dos. En ambas iteraciones se repetirá un proceso de trabajo similar para proporcionar un resultado completo sobre el producto final.

“El modelo de proceso incremental se centra en que en cada incremento se entrega un producto que ya opera. Los primeros incrementos son versiones desnudas del producto final, pero proporcionan capacidad que sirve al usuario y también le dan una plataforma de evaluación”. (Pressman, 2010).

Para el primer incremento se implementan requisitos como, “agregar, eliminar y editar empaquetador”, hacer login al sistema, ingresar atrasos, ingresar cambios de turno y listar prioridad.

En el segundo incremento, y final, se implementan requerimientos como, tomar turno, escanear código QR y generar informes.

No obstante, en esta ocasión, la implementación de esta forma de trabajo no hará participar a clientes hasta el final del desarrollo, por lo cual, el análisis, verificación y comprobación de cada versión será hecha por el desarrollador.

2.3.2 Estándares

Las normas ISO, están orientadas a ordenar la gestión de una empresa en sus distintos ámbitos. La alta competencia internacional acentuada por los procesos globalizadores de la economía y el mercado, y el poder e importancia que ha ido tomando la figura y la opinión de los consumidores, ha propiciado que dichas normas, pese a su carácter voluntario, hayan ido ganando un gran reconocimiento y aceptación internacional.

Las normas ISO son establecidas por el Organismo Internacional de Estandarización (ISO), y se componen de estándares y guías relacionados con sistemas y herramientas específicas de gestión aplicable en cualquier tipo de organizaciones. Se crearon con la finalidad de ofrecer orientación, coordinación, simplificación y unificación de criterios a las empresas y organizaciones con el objeto de reducir costes y aumentar la efectividad, así como estandarizar las normas de productos y servicios para las organizaciones internacionales.

En lo particular la familia de normas ISO/IEC 25000, conocida como SQuaRE (System and Software Quality Requirements and Evaluation), tiene por objetivo la creación de un marco de trabajo común para evaluar la calidad del producto de software. Es el resultado de la evolución de otras normas anteriores, especialmente de las normas ISO 9126, que describe las particularidades de un modelo de calidad del producto de software. Esta familia de normas ISO/IEC 25000 se encuentra compuesta por cinco divisiones: (ISO25000, s.f.).

- ISO/IEC 2500: División para gestión de la calidad.
- ISO/IEC 2501n: División para modelo de calidad.
- ISO/IEC 2502n: División para la medición de calidad.
- ISO/IEC 2503n: División para los requisitos de calidad.
- ISO/IEC 2504n: División para la evaluación de calidad.

Con respecto a la norma ISO 25012, que hace referencia a la calidad de producto de datos, sobre los cuales se construye un sistema para la evaluación de un producto de datos. En un modelo de calidad de datos se establecen las características de calidad de datos que se deben tener en cuenta a la hora de evaluar las propiedades de un producto de datos determinados. La calidad del producto de datos se puede entender como el grado en que los datos satisfacen los requisitos definidos por la organización a la que pertenece el producto.

El modelo de calidad de producto de datos definido por el estándar ISO/IEC 25012 se encuentra compuesto por 15 características:

- Exactitud.
- Completitud.
- Consistencia.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

- Credibilidad.
- Actualidad.
- Accesibilidad.
- Conformidad.
- Confidencialidad.
- Eficiencia.
- Precisión.
- Trazabilidad.
- Comprensibilidad.
- Disponibilidad.
- Portabilidad.
- Recuperabilidad.

La aplicación móvil para la gestión de empaquetadores hace un manejo de datos correspondientes a personas vinculadas a la organización, por ende, el manejo y protección de los datos es un punto importante a cubrir. Entre las características que cubren la aplicación podemos mencionar:

- **Exactitud:** Es el grado en que los datos representan correctamente el verdadero valor del atributo deseado. Esto se refleja en la limitación de los datos que se pide para hacer registro de personas para vincular a la organización, es decir, se necesitan datos en concreto y valores exactos para ser registrados en la aplicación.
- **Consistencia:** grado en que los datos están libres de contradicción con otros datos en un contexto específico. La aplicación hace uso de datos específicos, por ende, no se genera incoherencia entre datos para el registro del personal.
- **Accesibilidad:** Grado en que los datos pueden ser accedidos en un contexto específico. La aplicación mantiene los datos siempre accesibles para su revisión.
- **Confidencialidad:** grado en que los datos aseguran que solo son accedidos e interpretados por usuarios autorizados. La aplicación tendrá como medida de seguridad un logeo distinto para cada usuario perteneciente a alguna organización, protegido por contraseña, por lo cual los datos de cada uno serán resguardados. Sin embargo, el encargado, y solo él/ella, entre sus registros tendrá la posibilidad de ver los datos del personal de la organización.
- **Portabilidad:** Grado en que a los datos se les permite ser reemplazados de un sistema a otro o eliminados. La aplicación permitirá eliminar a un empaquetador de los registros de la aplicación o bien, modificar su información y guardar estos cambios.

2.4 Modelos

2.4.1 Modelo entidad relación

El modelo se muestra en la Ilustración 3: Modelo entidad relación. y explica el cómo se relacionan las distintas entidades detectadas. Si bien, para el desarrollo de la aplicación se usó una base de datos no relacional, se muestra la siguiente ilustración con el objetivo de una mejor comprensión. En los siguientes capítulos se especifica la implementación del modelo de base de datos no relacional.

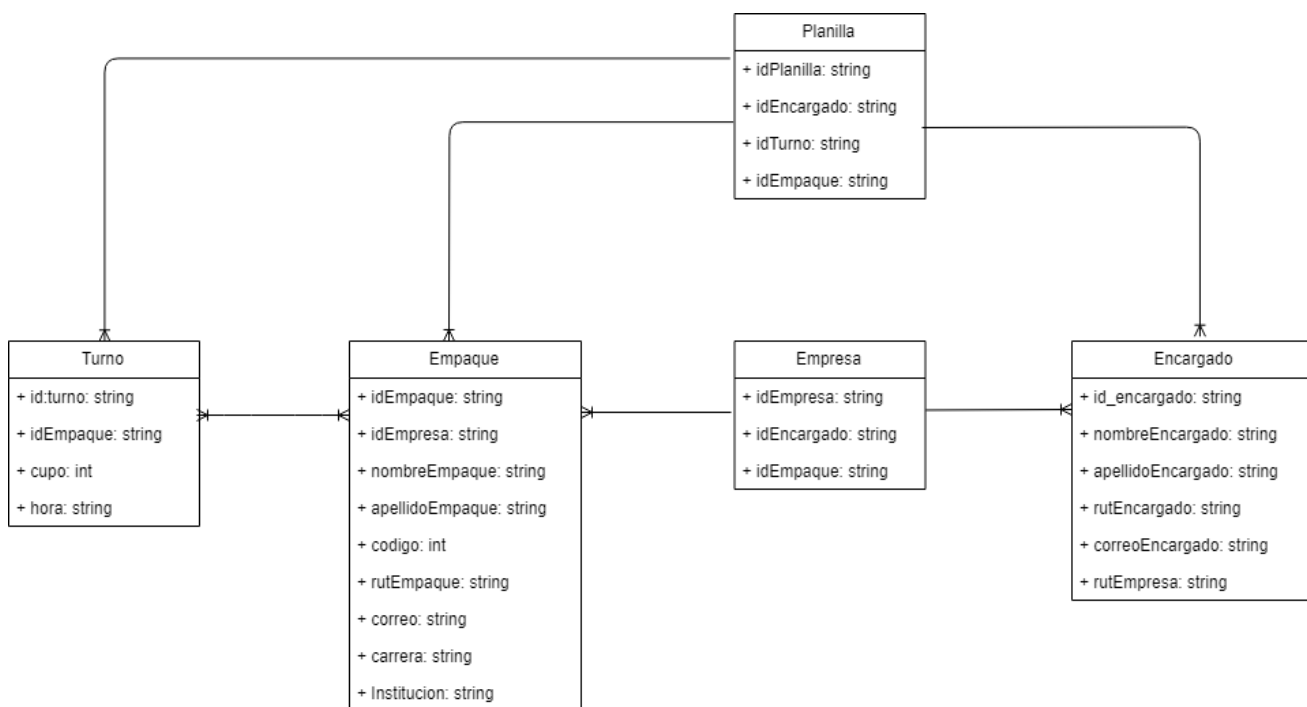


Ilustración 3: Modelo entidad relación.

La entidad Turnos consiste en la jornada laboral que deben realizar los empaquetadores en un determinado día y a una determinada hora.

La entidad Empaque hace referencia al empaquetador en sí, es básicamente la persona que trabaja en la organización y que debe cumplir con las normas establecidas por los encargados.

La entidad Encargado hace referencia a la persona que gestionará a los empaquetadores de la organización, además de establecer el formato de la planilla de toma de turnos.

La entidad Planilla hace referencia a la tabla en la cual se registrarán los turnos de cada empaquetador semanalmente.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

La entidad Empresa, es la institución o comercio en el cual se desenvuelve la organización de empaquetadores.

Con respecto a los atributos que tiene cada entidad.

Empaque:

- idEmpaque: Identificador o número de credencial de cada empaque.
- rutEmpaque: Rut del empaque.
- nombreEmpaque: Nombre del empaque.
- apellidoEmpaque: Apellido del empaque.
- correo: Email con el que se registra al empaquetador.
- Institución: Casa de estudio del empaque (en caso de que sea estudiante).
- Carrera: Carrera que está cursando.
- Codigo: Código opcional que identifica.
- rutEmpresa: Rut de la empresa a la cual pertenece.

Turno:

- idTurno: Identificador del turno.
- idEmpaque: Identificador del empaque que toma el turno.
- cupo: Cantidad de empaques que puede llegar a tener el turno.
- hora: Hora del turno.

Encargado:

- idEncargado: Identificador o número de credencial del encargado.
- nombreEncargado: Nombre del encargado.
- apellidoEncargado: Apellido del encargado.
- rutEncargado: Rut del encargado.
- rutEmpresa: Rut de la empresa a la cual pertenece.
- correoEncargado: Email del encargado, le permite iniciar sesión.

Planilla:

- idPlanilla: Identificador o número de la planilla.
- idEncargado: identificador del encargado que está en planilla.
- idEmpaque: Identificador del empaquetador que está en planilla.
- idTurno: Identificador del turno que está en planilla.

Empresa:

- idEmpresa: Identificador de la empresa.
- idEncargado: Identificador del encargado.
- idEmpaque: Identificador del empaquetador.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

2.4.2 Herramientas

Para el desarrollo de los proyectos, hay diversas herramientas que se utilizan a lo largo del ciclo, tanto para el desarrollo del sistema como para la documentación del mismo. Las herramientas generan algo llamado artefactos, que es un producto tangible resultante del proceso de desarrollo de software.

2.4.2.1 Framework y lenguajes

Un Framework, o marco de trabajo, es un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia, además, representa una arquitectura de software que modela las relaciones generales de las entidades del dominio. Mientras que un lenguaje de programación es un lenguaje formal diseñado para realizar procesos que puedan ser llevados a cabo por una máquina, por ejemplo, un computador. Un lenguaje de programación proporciona los elementos necesarios para traducir los pasos de un pseudocódigo en formato comprensible para la máquina.

Los utilizados en el proyecto son:

- Angular, framework de desarrollo para JavaScript que sirve para facilitar la implementación de la aplicación móvil. (Angular oficial, s.f.).
- HTML, es la sigla que representa HyperText Markup Language. Básicamente es un conjunto de etiquetas que sirven para definir texto y otros elementos que compondrán el diseño de una página web. (w3schools, s.f.)
- Ionic, es un SDK de código abierto para el desarrollo de aplicaciones móviles, construido sobre AngularJs y usando tecnologías web como HTML y CSS busca generar aplicaciones híbridas para dispositivos móviles. (Ionic Framework, 2019)
- JavaScript, es el lenguaje que nos permite crear funcionalidades dinámicas a nuestro diseño, permitiendo mejorar la interfaz de usuario del lado del cliente. (Eich, 2016)
- CSS, es la sigla de Cascading Style Sheet, lenguaje de hojas de estilo que nos permite crear páginas web de una manera más exacta controlando el aspecto de los archivos o documentos que definen el HTML.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

2.4.2.2 Software de gestión del proyecto

Durante el desarrollo del proyecto se hace uso de diversos programas o softwares que ayudan con la gestión del proyecto, ya sea para documentar, modelar, respaldar información con estudios y estadísticas, etc. A continuación, se da a conocer las herramientas de software para la gestión.

- Ofimática, para el desarrollo del informe de la aplicación. (Microsoft, 2016).
- Bootstrap, herramienta para el diseño de la aplicación. (Twitter, 2018)
- Bizagi, herramienta que permite crear modelos de negocio. (Limited, 2019).
- Dia: herramienta que permite diseñar modelos de caso de uso. (Larsson, 2011).
- Draw.io: Pagina web que permite diseñar modelos entidad relación. (Alder, 2000).
- Survio: Es una herramienta excelente para la investigación cualitativa y cuantitativa en las ciencias sociales. Su fácil manejo, las posibilidades graficas que ofrece, su capacidad de almacenar y tratar los datos, hacen de su elección un éxito. Se utilizó para la realización de encuestas en Jumbo Chillán. (Ricart, 2018).
- Kata Kuntur: herramienta para el modelamiento de bases de datos, es multiplataforma y soporta los gestores de bases de datos más conocidos. Se Utilizó para la confección del modelo de datos. (Mazuelos, 2017)

2.5 Arquitectura de software

La arquitectura del software es el nivel de diseño, que hace foco en aspectos más allá de los algoritmos y estructuras de datos de la computación; el diseño y especificación de la estructura global del sistema es un nuevo tipo de problema.

La aplicación desarrollada no posee el patrón Modelo-Vista-Controlador (MVC) clásico, ya que es basado en componentes, esto debido a que se utiliza Angular.

Angular, más bien utiliza un patrón Modelo-Vista-Vista-Modelo (MVVM) ya que el modelo tiene mucha relación con la vista. Esto es así por el concepto base de angular de “two-way data binding”, ya que la forma de sincronizar los datos entre la vista y el modelo-vista es totalmente dependiente, es decir, en la vista podemos modificar el modelo y en el modelo podemos modificar la vista. Esto hace que la independencia que se produce en el modelo-vista-controlador clásico aquí no se produzca, por lo tanto, tiende a llamarse modelo-vista-vista-modelo. (Shaw, 1994)

Capítulo 3

Especificación de requerimientos de software

3.1 Propósito

La aplicación móvil busca cubrir la mayor parte de necesidades que puedan tener los encargados de las organizaciones de empaquetadores en comercios de grandes superficies. Las principales funcionalidades que debe tener un sistema como este es la acción de tomar turnos para los empaques. Igual de importante es la gestión para con los empaques, específicamente la información correspondiente a cada uno de ellos.

Los principales requisitos del software son:

- **Tomar turno:** Es la acción de realizar la toma de turnos de forma remota desde los dispositivos móviles de los empaquetadores, seleccionando los turnos que se deseen y estén disponibles. Esta elección de turnos será posterior a la publicación del listado que expone el orden con el cual se deben tomar los turnos. Cabe mencionar que este orden será al azar semanalmente.
- **Gestionar empaquetador:** Es básicamente generar un CRUD para así lograr registrar la información pertinente a cada uno de los empaquetadores miembros de la organización. Con esto se busca tener los datos del personal más a mano y ordenado.
- **Controlar asistencia:** Registro que se necesita para dejar constancia de que un empaquetador llega tarde a su turno o que no asiste. Esta funcionalidad es para los encargados, que en muchas ocasiones penalizan estos atrasos y el registro les podría ser de gran ayuda.
- **Generar detalle de asistencia:** Detalle de datos importantes del control de asistencia, por ejemplo, el día, hora y nombre de empaquetador al cual se escanea el QR.
- **Escanear código QR:** El escanear un código QR existente en las credenciales de cada empaquetador al inicio de cada turno, permite a los encargados tener un control de asistencia a los turnos. La finalidad es que se tenga un documento con los empaques que asistieron durante el día. Esta funcionalidad será opcional para los encargados, ya que, si no se aplica, no impide el funcionamiento de alguna otra característica de la aplicación.

3.2 Alcances y limitaciones

La aplicación móvil tiene como principal característica la optimización de procesos y acciones. Entre sus acciones está la toma de turnos, la cual permite realizarla de forma online, con esto se reduce el tiempo que tienen que invierten los empaquetadores para poder tomar sus turnos semanales.

Con respecto a los encargados, estos poseen la facultad de agregar, editar y eliminar empaquetadores de la organización, además de controlar la asistencia mediante un lector de código QR que cuyos datos serán de utilidad para lograr generar informes semanales y mensuales, los cuales entregan información como la cantidad de turnos semanales o mensuales de cada encargado. Existe la opción de registrar las faltas (atrasos o ausencia a turnos) y registrar los cambios de turno.

El usuario tendrá la opción para visualizar una agenda semanal con los turnos disponibles para que pueda seleccionar los que desee.

Como existen dos tipos de usuario en la aplicación, esta contará con un login que discriminará el tipo de perfil de los usuarios.

Las dos principales limitaciones de la aplicación será el impedimento de asignación de cajas en los turnos, esto debido a que es imposible saber el número de cajas, ni las cajas disponibles y la otra limitación será el control de cualquier tipo de cobro monetario que los encargados les hagan a los empaquetadores.

3.3 Requerimientos específicos

3.3.1 Requisitos funcionales

Los requisitos funcionales de un sistema, son aquellos que describen cualquier actividad que este deba realizar, en otras palabras, el comportamiento o función particular de un sistema o software cuando se cumplen ciertas condiciones.

Los requisitos funcionales identificados son expuestos en la Tabla 1: Requisitos funcionales.

ID	Nombre	Descripción
RF-01	Agregar empaquetador	El encargado posee la facultad de agregar empaquetadores a la organización mediante la aplicación móvil.
RF-02	Eliminar empaquetador	El encargado posee la facultad de eliminar empaquetadores de la organización mediante la aplicación móvil.
RF-03	Editar empaquetador	El encargado posee la facultad de editarlos datos de empaquetadores de la organización mediante la aplicación móvil.
RF-04	Escanear QR	El encargado podrá escanear el código QR de los empaquetadores para así capturar los datos y generar un posterior informe semanal o mensual. Además, este registro de asistencia, permite dejar evidencia de los atrasos de los empaquetadores, registrando su hora de llegada al turno.
RF-05	Detalle de control de asistencia	El encargado podrá obtener un informe (listado) con detalle del control de asistencia, especificando datos como la hora, el día y el nombre del empaquetador.
RF-06	Tomar turnos	Los empaquetadores podrán elegir sus turnos desde una planilla semanal.
RF-07	Listado de empaquetadores	El encargado tendrá acceso al listado de todos los empaquetadores de la organización, con todos sus datos ingresados al momento de registrar al empaquetador en la aplicación.
RF-08	Lista aleatoria	La aplicación móvil genera un listado aleatorio con el nombre de los empaquetadores. Esto establece la prioridad para elegir los turnos.
RF-09	Definir cantidad de turnos	El encargado tiene la facultad de definir la cantidad máxima de turnos que pueden seleccionar los empaquetadores a la hora de tomar turnos.

Tabla 1: Requisitos funcionales.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

3.3.2 Requisitos no funcionales

Los requisitos no funcionales representan características generales y restricciones de la aplicación o sistema que se esté desarrollando. Pueden clasificarse en requerimientos de usabilidad, eficiencia y seguridad.

Los requisitos no funcionales son expuestos en la Tabla 2: Requisitos no funcionales.

ID	Nombre	Descripción
RNF-01	Operatividad	La aplicación debe tener las funcionalidades disponibles en todo momento para los usuarios.
RNF- 02	Diferenciar organización	La aplicación debe ser capaz de diferenciar a que organización pertenece cada empaquetador.
RNF-03	Interfaz agradable	La aplicación, visualmente va orientada al ambiente de los supermercados, por ende, será amistosa para los usuarios objetivo.
RNF-04	Usabilidad	La aplicación es simple e intuitiva a la hora de usar sus funcionalidades, con conceptos conocidos y más aún por personas que estén envueltas en el ambiente de organizaciones de empaquetadores.
RNF-05	Compatibilidad	La aplicación es disponible para múltiples plataformas, como Android, iOS, Windows Phone.

Tabla 2: Requisitos no funcionales.

3.4 Encuesta

3.4.1 Definición de encuesta

Una encuesta es un procedimiento dentro de los diseños de una investigación descriptiva en el que el investigador recopila datos mediante un cuestionario previamente diseñado, sin modificar el entorno ni el fenómeno donde se recoge la información ya sea para entregarlo en forma de tríptico, gráfica o tabla. Los datos se obtienen realizando un conjunto de preguntas normalizadas dirigidas a una muestra respectiva.

3.4.2 Características de encuesta

Tipos de encuestas

- Encuesta descriptiva: Reflejan las actitudes o condiciones presentes. Significa que intentan describir en qué situación se encuentra una determinada población.
- Encuesta analítica: Buscan, además de describir, explicar el porqué de una determinada población.

Tipos de preguntas

- De respuesta abierta: En la encuesta se le pide al interrogado que responda él mismo a la pregunta formulada.
- De respuesta cerrada: En estas, los encuestados deben elegir una de las opciones para responder.

Objetivos de una encuesta

- Medir las relaciones entre variables económicas, sociales, etc.
- Evaluar las estadísticas demográficas como errores, omisiones e inexactitudes.
- Conocer patrones de las variables y sus factores.
- Investigar las características de la población.
- Dar una respuesta de los cuales los entrevistados darán una opinión congruente.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

3.4.3 Encuesta a Jumbo Chillán

Para llevar a cabo la encuesta, se ha solicitado a los encargados de la organización de empaquetadores de Jumbo Chillán, la colaboración para responder dos encuestas. Una de las encuestas es orientada a los empaquetadores, mientras que la segunda es orientada solo para los encargados.

Las características de estas encuestas son, básicamente, de tipo analítica ya que se busca recopilar información que permita formular inferencias explicativas. Con respecto al tipo de pregunta, es de respuestas cerradas, ya que los encuestados tienen opciones concretas para seleccionar sus respuestas.

Se realizaron dos encuestas por el hecho de que se necesitaba saber la perspectiva tanto de los empaquetadores como también la de los encargados, pero orientados a distintos enfoques. La encuesta para los empaquetadores es principalmente para saber la disponibilidad y usabilidad con dispositivos móviles, mientras que la encuesta para encargados busca saber si los encargados están interesados en aplicar tecnología a su gestión en la organización.

Como catastro, se obtuvo un total de 40 personas que respondieron a la encuesta de empaquetadores, mientras la encuesta de encargados, tuvo un total de 8 personas (total de encargados en Jumbo Chillan) que la respondieron.

Los resultados de las preguntas se pueden ver en detalle en el Anexo 1: Encuesta a Jumbo Chillán

Capítulo 4

Tecnologías Emergentes

4.1 Framework Angular

Comúnmente llamado Angular 7, es un framework para aplicaciones web desarrollado en TypeScript, de código abierto, mantenido por Google, que se utiliza para crear y mantener aplicaciones web orientadas al Frontend. Su objetivo es aumentar las aplicaciones basadas en navegadores con capacidad de Modelo-Vista-Controlador (MVC), en un esfuerzo para hacer que el desarrollo y las pruebas sean más fáciles.

La biblioteca lee el HTML que contiene atributos de las etiquetas personalizadas adicionales, entonces obedece a las directivas de los atributos personalizados, y une las piezas de entrada o salida de la página a un modelo representado por las variables estándar de JavaScript.

Angular es la evolución de AngularJS aunque incompatible con su predecesor. Es una plataforma que facilita la creación de aplicaciones con la web. Combina plantillas declarativas, inyección de dependencias, herramientas de extremo a extremo y mejores prácticas integradas para resolver los desafíos de desarrollo. Permite a los desarrolladores crear aplicaciones que viven en la web, el móvil o el escritorio. (Angular oficial, s.f.).

4.1.1 Arquitectura

Angular es una plataforma y un marco para la creación de aplicaciones en HTML y TypeScript. Implementa la funcionalidad central y opcional como un conjunto de bibliotecas que importa a sus aplicaciones.

Los bloques de construcción básicos de una aplicación angular son NgModules, que proporcionan un contexto de compilaciones para los componentes. NgModules recopila códigos relacionados en conjuntos funcionales; una aplicación angular se define por un conjunto de NgModules. Una aplicación siempre tiene al menos un módulo raíz que permite el arranque y por lo general, tiene muchos más módulos de funcionales. Los componentes definen vistas y utilizan servicios.

Tanto los componentes como los servicios son simples clases, con decoradores que marcan su tipo y proporcionan metadatos que le indican a angular como usarlos. Los metadatos de una clase de componentes lo asocian con una plantilla que define una vista. Una plantilla combina HTML ordinario con directivas angular y un marcador de enlace que permite a angular modificar el HTML antes de representarlo. Los metadatos para una clase proporcionan la información que angular necesita para que esté disponible para los componentes a través de la inyección de dependencias.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

La Ilustración 4: Arquitectura de Angular. muestra de forma esquemática un diseño de esta arquitectura:

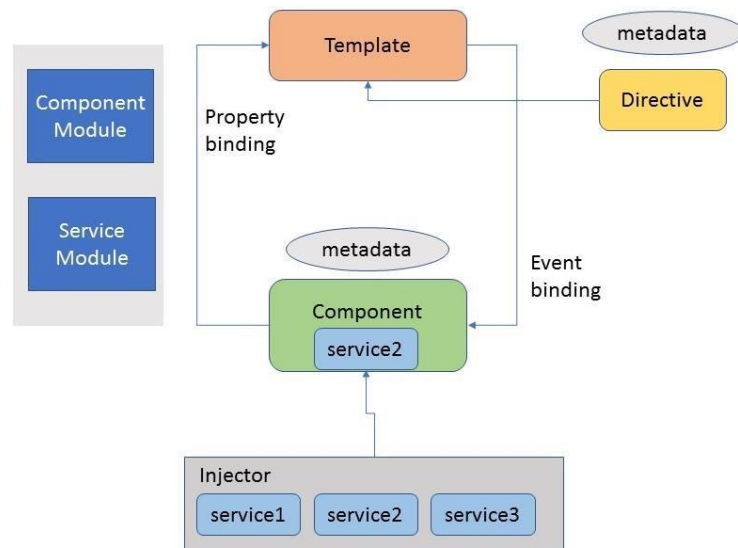


Ilustración 4: Arquitectura de Angular.

4.1.2 Módulos de Angular

Las aplicaciones Angular son modulares y tienen su propio sistema de modularidad llamado NgModules. Son contenedores para un bloque cohesivo de código. Los NgModules pueden contener componentes, servicios y otros archivos de código cuyo alcance está definido por el NgModule que lo contiene.

Cada aplicación Angular tiene al menos una clase NgModule, que se denomina “AppModule” y reside en un archivo llamado “app.module.ts”.

Si bien una aplicación pequeña puede tener solo un NgModule, la mayoría de las aplicaciones tienen muchos más módulos de funciones.

NgModules proporciona un contexto de compilación para sus componentes. Siempre tienen un componente raíz que se crea durante el arranque, pero cualquier NgModule puede incluir cualquier número de componentes adicionales que se pueden cargar a través del enrutador. Los componentes que pertenecen a un NgModule comparten un contexto de compilación.

La declaración de importación de NgModule es la siguiente:

```
“import { NgModule } from '@angular/core’;”
```

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

4.1.3 Componentes en Angular

Un componente controla un parche de pantalla, llamado vista. Los componentes individuales definen y controlan cada una de las vistas. Define la lógica de la aplicación de un componente dentro de la clase. La clase interactúa con la vista a través de una API de propiedad y métodos.

Generalmente se tendrá un árbol de componentes que forman la aplicación y cada persona lo puede organizar como estime conveniente. Siempre existirá un componente padre y a partir de ahí podrán colgar todas las ramas que sean necesarias para crear la aplicación.

4.1.4 Servicios en Angular

Los servicios son una categoría amplia que abarca cualquier valor, función o característica que necesita una aplicación. Un servicio es típicamente una clase con un propósito bien definido, debe hacer algo específico y hacerlo bien.

Angular distingue los componentes de los servicios para aumentar la modularidad y la reutilización.

Los servicios son una parte fundamental en el desarrollo de aplicaciones Angular, básicamente es un proveedor de datos que mantiene la lógica de acceso a ellos y operativa relacionada con el negocio y las cosas que se hacen con los datos dentro de la aplicación. Los servicios serán consumidos por los componentes, que delegarán en ellos la responsabilidad de acceder a la información y la realización de operaciones con los datos. (Alvarez, 2017).

Para crear un servicio en Angular se utiliza el comando “generate service”, indicando a continuación el nombre del servicio que queremos generar. Se hace de la siguiente manera al usar Angular CLI:

“ng generate service nombreServicio”.

Para poder usar los servicios, se deben agregar a un módulo, inmediatamente lo podrás usar en cualquiera de los componentes a los que pertenecen a ese módulo.

4.2 Framework Ionic

Ionic es una herramienta de UI de código abierto para crear aplicaciones móviles y de escritorio de alta calidad y rendimiento, usando tecnologías web como HTML, CSS y JavaScript.

Ionic se centra en la experiencia de usuario de la interfaz o la interacción de la interfaz de usuario de una aplicación (controles, interacciones, gestos, animaciones). Es fácil de aprender, se integra muy bien con otras bibliotecas o marcos como Angular, o se puede usar de forma independiente sin un marco de frontend usando un simple script incluido.

Actualmente, Ionic tiene integración oficial con Angular, pero el soporte para Vue y React está en desarrollo. (Ionic Framework, 2019).

4.2.1 Metas de Ionic

- **Multiplataforma:** crear y desplegar aplicaciones en multiplataformas, como iOS nativo, Android, escritorio y web, todo con una base de código. Escribir una vez, correr en cualquier lugar.
- **Basado en estándares web:** Esta construido encima de confiable tecnología web estandarizada, como HTML, CSS y JavaScript, utilizando API web moderna como elementos personalizados y DOM de sombras.
- **Bonito diseño:** Limpio sencillo y funcional. Está diseñado para funcionar y mostrarse maravillosamente listo para usar en todas las plataformas.
- **Sencillez:** Se ha creado pensando en la simplicidad, de modo que la creación de aplicaciones sea agradable, fácil de entender y accesible para casi cualquier persona con habilidades de desarrollador web.

4.2.2 Licencia

Es un proyecto gratuito y de código abierto, publicado bajo la licencia MIT permisible. Esto significa que puede ser utilizado en proyectos personales o comerciales de forma gratuita. MIT es la misma licencia utilizada por proyectos tan populares como jQuery y Ruby on Rails.

4.2.3 Compatibilidad de marco

Mientras que las versiones anteriores de Ionic (Ionic V3) estaban estrechamente relacionadas con Angular, V4 del marco se modificó para funcionar como una biblioteca de componentes web independiente, con integraciones para los marcos de JavaScript más recientes, como Angular. Ionic se puede usar en la mayoría de los framework frontend con éxito, incluyendo React y Vue, aunque algunos framework necesitan soporte completo de componentes web.

- JavaScript: Uno de los objetivos principales de Ionic 4 es eliminar cualquier requisito duro en un solo marco para alojar los componentes. Esto significa que los componentes principales pueden funcionar de forma independiente con solo una etiqueta de script en una página web. Si bien trabajar con marcos puede ser excelente para equipos grandes y aplicaciones más grandes, ahora es posible usar Ionic como una biblioteca independiente en una sola página, incluso, en un contexto como WordPress.
- Angular: Angular siempre ha estado en el centro de lo que hace grande a Ionic. Si bien los componentes principales se han escrito para que funcionen como una biblioteca de componentes web independientes, el paquete “@ionic/angular” facilita la integración con el ecosistema angular. “@ionic/angular” incluye toda la funcionalidad que los desarrolladores Angular esperaban de Ionic 2/3, y se integra con las bibliotecas de Angular, como el enrutador Angular.

4.3 Bases de datos no relacionales

Cuando hablamos de sistemas gestores de base de datos, el más usado es MySQL. A veces puede parecer que no es necesario buscar alternativas, pues se trata de una estupenda opción, pero lo crítico es que hay todo un mundo más allá de MySQL. De hecho, en función de nuestras necesidades, MySQL no es siempre la mejor alternativa.

En este sentido, las bases de datos NoSQL (o bases de datos no relacionales) están empezando a ocupar una importante parcela dentro de este espectro de soluciones ya que resuelven necesidades habituales para aplicaciones web, apps móviles o en el internet de las cosas (Iot).

4.3.1 ¿Qué son las bases de datos NoSQL?

Se puede decir que la aparición del término NoSQL aparece con la llegada de la web 2.0 ya que hasta ese momento solo subían contenido a la red aquellas empresas que tenían un portal, pero con la llegada de aplicaciones como Facebook, Twitter o Youtube, cualquier usuario podía subir contenido, provocando así un incremento exponencial de los datos.

Es en este momento cuando empiezan a aparecer los primeros problemas de la gestión de toda esa información almacenada en bases de datos relacionales. En un principio, para solucionar estos problemas de accesibilidad, las empresas optaron por utilizar un mayor número de máquinas, pero pronto se dieron cuenta de que esto no solucionaba el problema, además de ser una solución muy cara. La otra solución era la creación de sistemas pensados para un uso específico que con el paso del tiempo han dado lugar a soluciones robustas, apareciendo así el movimiento NoSQL.

Hablar de bases de datos NoSQL es hablar de estructuras que nos permiten almacenar información en aquellas situaciones en las que las bases de datos relacionales generan ciertos problemas debido principalmente a problemas de escalabilidad y rendimiento de las bases de datos relacionales donde se dan cita miles de usuarios concurrentes y con millones de consultas diarias.

Además, las bases de datos NoSQL son sistemas de almacenamiento de información que no cumplen con el esquema entidad-relación. Tampoco usan estructura de datos en forma de tabla donde se almacenando los datos, sino que para el almacenamiento hacen uso de otros formatos como clave-valor, mapeo de columnas o grafos.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

4.3.2 ¿Cuándo usar base de datos no relacionales?

En general, podemos encontrar de gran utilidad una base de datos NoSQL cuando nuestras necesidades se centran en:

- **Velocidad:** Si una aplicación necesita almacenar o acceder a mucha información en poco tiempo, se necesita una base de datos que aporte gran velocidad. Las bases de datos documentales son capaces de ser mucho más rápidas que las relacionales, ya que pueden atender proyectos que exigen muchas operaciones por segundo.
- **Tamaño:** En cuanto al tamaño de la base de datos, si tenemos una cantidad de información enorme, entonces tenemos unas necesidades importantes de volumen. Las bases de datos relacionales tienen tendencia a funcionar más lentamente cuando en una tabla se encuentran cantidades muy grandes de registros (del orden de un millón para arriba). Situaciones así obligan a los administradores a buscar soluciones, como dividir las tablas en diversos segmentos, incrementando los tiempos y recursos que destinamos a la gestión de los datos. Este no es un problema en las bases de datos NoSQL, que son capaces de administrar volúmenes gigantescos de datos.

4.3.3 Ventajas de los sistemas NoSQL

Esta forma de almacenar información ofrece ciertas ventajas sobre los modelos relacionales. Entre las ventajas más significativas se pueden destacar:

- **Se ejecutan en máquinas con pocos recursos:** Estos sistemas, a diferencia de los sistemas basados en SQL, no requieren de apenas computación, por lo que se pueden montar en máquinas de un coste más reducido.
- **Escalabilidad horizontal:** Para mejorar el rendimiento de estos sistemas simplemente se consigue añadiendo más nodos, con la única operación de indicar al sistema cuales son los nodos que están disponibles.
- **Manejo de gran cantidad de datos:** Esto debido a que utiliza una estructura distribuida, en muchos casos mediante tablas Hash.
- **No genera cuellos de botella:** El principal problema de los sistemas SQL es que necesitan transcribir cada sentencia para poder ser ejecutada, y cada sentencia compleja requiere además de un nivel de ejecución aún más complejo, lo que construye un punto de entrada en común, que ante muchas peticiones puede ralentizar el sistema.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

4.3.4 Firebase

Es una plataforma de backend como servicio, dispone de una serie de herramientas para el desarrollo de aplicaciones, el almacenamiento y sincronización de datos en la nube, medición del comportamiento del usuario y soluciones para monetizar productos. Es una base de datos NoSQL (base de datos no relacional). (Google, s.f.).

Firestore nos ofrece distintos tipos de servicios, entre los más destacados se encuentran: (Ruiz, 2017).

4.3.4.1 Base de datos Realtime

Con la base en tiempo real de Firestore, podrás guardar todos los datos que requiera tu aplicación. Se lleva muy bien con React y su patrón reactivo que permite actualizar los datos en los componentes automáticamente. Los datos se almacenan en formato JSON y se pueden agregar reglas para permitir requests con token o solo desde una URL por ejemplo.

4.3.4.2 Autenticación

Es un servicio que nos simplifica el inicio de sesión y la gestión de la misma en nuestra aplicación. Si la usamos en aplicaciones web es extremadamente fácil de configurar, sobre todo si usamos el proveedor de Google, aun así, si usamos otros de los disponibles por Firestore, ya sea correo/contraseña, teléfono, Facebook, Twitter, GitHub, anónimo, también es muy fácil, solo es un paso más en el caso de las redes.

4.3.4.3 Almacenamiento

Este servicio es muy bueno para aplicaciones que requieran guardar archivos del usuario. También nos sirve si queremos subir estáticos ya que existe un botón desde la interfaz.

4.3.4.4 Cloud Functions

Es un producto bastante reciente en Firestore, tiene como objetivo la transformación de nuestro código del backend en pequeñas piezas del mismo (funciones). Estas funciones son creadas con NodeJS y se suben a la vez que hacemos deploy. Al crearse generan una URL a la que podemos llamar desde AJAX para que se ejecute el código pertinente. Firestore nos provee triggers para comunicar con ciertas funciones de sus productos, así como un trigger de HTTP.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

Cabe destacar que existen otros tipos de servicios como:

- TestLab: Que nos permite probar la aplicación antes de subirla a algún tipo de dispositivo.
- Monitoreo de rendimiento: Que mide el tiempo que los usuarios pasan durante distintos procesos.
- Notificaciones: Permite gestionar el envío de notificaciones a usuarios.
- Configuración remota: Permite modificar el comportamiento y la apariencia de la aplicación sin que los usuarios tengan que tocar algo.
- Enlaces dinámicos: Podemos asegurar que a pesar de las redirecciones que supone el proceso de instalar una aplicación, desde un anuncio lleguemos a la vista deseada, ganando así una experiencia que el usuario agradece.
- Admob: Permite colocar publicidad en distintos lugares de la aplicación.

4.3.4.5 Planes de Firebase

En Firebase se dispone de varios planes; uno fijo y uno de pago según los servicios consumidos. Es recomendable empezar con un plan Spark que se denomina gratuito, y una vez ya se necesiten más funciones, pasar al plan Flame.

- El primero es el plan básico y gratuito.
- El segundo es el plan Flame, a 25 dólares al mes.
- El tercero es el plan Blaze, que se paga según servicios.

En la siguiente **¡Error! No se encuentra el origen de la referencia.** se pueden apreciar los planes y las características de cada uno.

4.4 Firebase en el proyecto

Como ya se mencionó, Firebase es un gestor de bases de datos no relacionales, por ende, generar un modelo de datos es algo que aún no está definido ni documentado para su diseño.

Para efectos del actual proyecto, se explica por qué usar Firebase y ejemplificar mediante un esquema el comportamiento de las “collections” y los “documents” utilizados en la base de datos.

Se ha decidido usar Firebase por varias razones, la primera razón es por la innovación de utilizar nuevas tecnologías, Firebase es relativamente nuevo y tiene una gran proyección con respecto de su potencial y su uso. Otro motivo es por la necesidad de construir web services, api rest y configuración de servidor. Firebase tiene una librería multiplataforma, de esta manera podemos utilizar la misma lógica del código para cada lenguaje de programación. Firebase permite implementar Push Notification, que básicamente es una librería bastante demandada en las aplicaciones móviles, por lo tanto, una gran ventaja.

Es por las características anteriores que se ha decidido hacer uso de Firebase Google como gestor de base de datos.

A continuación, en la Ilustración 5: Base de datos Firebase, parte 1. se muestra la base de datos generada por las collections y los documents. La collection “empresa” es la principal que abarca la mayor parte de los registros, luego el documento “111111” es el Rut de una determinada empresa, a su vez, ese documento contiene la collection “empaques” que dentro de ella están los documentos de registro de cada empaquetador de dicha empresa.

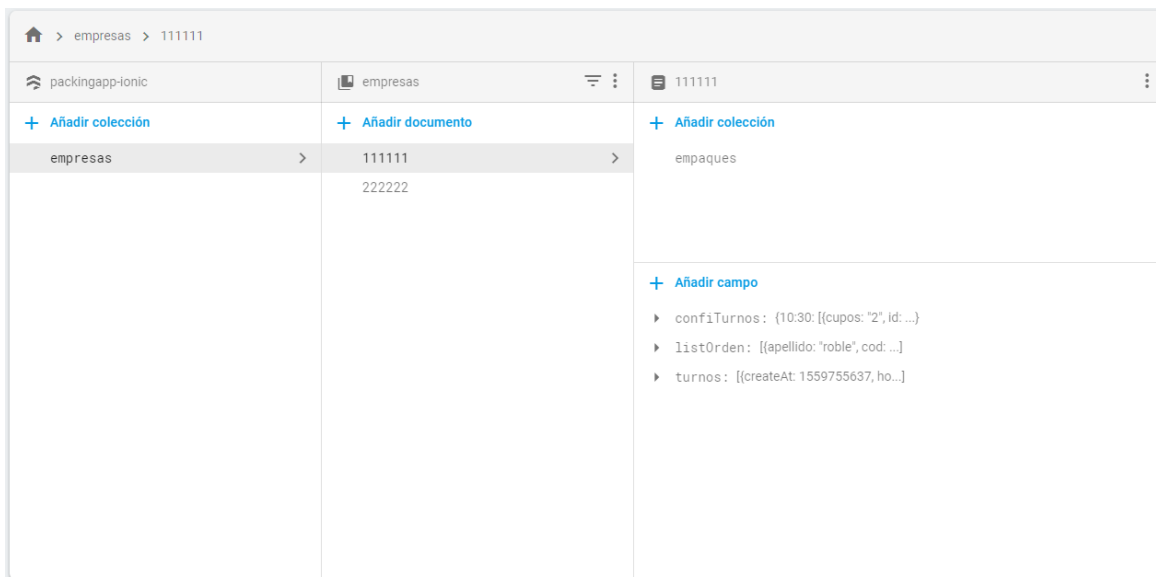


Ilustración 5: Base de datos Firebase, parte 1.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

Además de la collection “empaques” existen los campos “confiTornos” que guarda la configuración de cupos de la planilla, “listOrden” que guarda el arreglo del listado de empaquetadores generado de manera aleatoria para luego ser publicado, y “turnos” que guarda los turnos tomados en la toma de turno semanal.

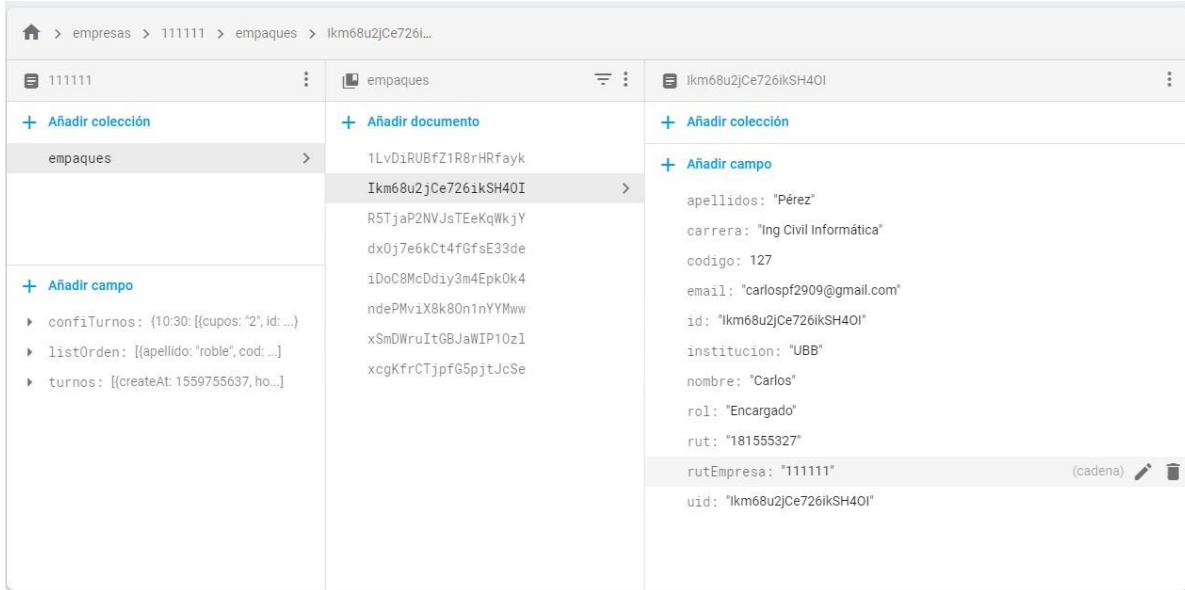


Ilustración 6: Base de datos Firebase, parte 2.

Capítulo 5

Estudio de factibilidad

5.1 Factibilidad técnica

En la actualidad hay múltiples lenguajes para el desarrollo de aplicaciones móviles, es por ello que es factible realizar un proyecto para implementar una App que permita gestionar empaques en comercios de grandes superficies. Existen entornos de desarrollo, editores de código, entre otras herramientas de software, que permiten realizar el proyecto.

Además, el desarrollador posee las capacidades técnicas en lenguaje de programación y conocimientos sobre herramientas necesarias para lograr un trabajo de calidad. Otro punto clave, es que el desarrollador tiene conocimientos del cómo funciona en ambiente en el cual se implementara la aplicación móvil, ya que se tiene experiencia en terreno del cómo funciona todo el proceso de gestión de empaques.

Por otro lado, la aplicación es factible con respecto a la portabilidad para los usuarios, ya que hoy en día los smartphones son una herramienta que cada persona tiene, con lo cual cada empaquetador estaría en condiciones de tener la App en su dispositivo.

5.2 Factibilidad operativa

La aplicación implementará funcionalidades con nombres que son intuitivas para los empaquetadores, ya que se usarán términos que generalmente se usan en la mayoría de las organizaciones de empaquetadores. Dicho esto, se estima que no se necesitará capacitar a los usuarios, que generalmente son personas jóvenes que están familiarizados con el manejo de apps móviles.

El uso de la aplicación podrá ser llevada a cabo por todos, ya que solo se necesita tener un smartphone con acceso a Internet, y hoy en día la mayoría cumple con esos dos requerimientos.

5.3 Factibilidad económica

Se trata de un producto con el cual se obtendrán beneficios por la prestación de servicios que ofrece. La aplicación se arrendará mensualmente a las distintas organizaciones de empaquetadores que se encuentren en los supermercados, como objetivo a 5 años, una cuota del mercado esperada que arriende el servicio es de apenas el 8% (14 organizaciones), esto correspondiente solo en la región del Biobío. No se está cerrado a expandir la aplicación para su uso en otras regiones, por lo cual puede ser presentado en cualquier parte del país, no obstante, para asuntos de este proyecto, se centrará en su comercialización solo en la región.

Inversión:

- Trabajo de ing. civil informática, \$5.000 la hora (250 hrs). \$1.250.000
- Ordenador para el desarrollador. \$300.000
- Total \$1.550.000

Gastos:

Tipo	Detalle	Mes	Año
Servidor Firebase	Servidor web para almacenar base de datos.	\$17.500	\$210.000
Cuenta PlayStore	Crear cuenta de desarrollador Android. El pago es solo una vez y es de 25 dólares.		\$17.500
Cuenta AppStore	Crear cuenta de desarrollador iOS. El pago es anual,		\$68.000
Total			\$295.500

Tabla 3: Factibilidad económica, gastos.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

Beneficios:

Tipo	Detalle	Mes	Año
Arriendo de app	Se busca arrendar la aplicación móvil a encargados de organización de empaques de distintas empresas de grandes superficies. El arriendo mensual será de \$10.000. Se espera comenzar con 5 organizaciones asociadas en el primer año y posteriormente ir aumentando el número de estos.	\$50.000	\$600.000
Total			\$600.000

Tabla 4: Factibilidad económica, beneficios.

El cálculo del VAN se realiza usando la siguiente formula expuesta en la Ilustración 7: Fórmula VAN.

$$VAN = -I_0 + \sum_{j=1}^n \frac{FN_j}{(1+i)^j}$$

Ilustración 7: Fórmula VAN.

Esta fórmula contempla factores como I_0 que representa la inversión inicial, n que indica el periodo a calcular en años, FN_j es el flujo de caja correspondiente a cada periodo e “ i ” es la tasa de descuento.

Tasa de descuento	0,1
años	5

Tabla 5: Factibilidad económica, índice de descuento y periodo de tiempo.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

	0	1	2	3	4	5
Inversión	\$1.550.000					
Gastos		\$295.500	\$278.000	\$278.000	\$278.000	\$278.000
Beneficios		\$600.000	\$1.080.000	\$1.440.000	\$1.560.000	\$1.680.000
Flujo		\$304.500	\$802.000	\$1.162.000	\$1.282.000	\$1.402.000

Tabla 6: Inversión, gastos, beneficios y flujos por año.

VAN	\$2.008.810
-----	-------------

Tabla 7: Resultado de cálculo de VAN.

5.4 Conclusión de factibilidad

Se puede apreciar que el resultado del cálculo del VAN arroja un resultado bastante atractivo y por mucho superior a 0, por ende, se puede concluir que la factibilidad es totalmente rentable.

Si bien los factores usados para este cálculo fueron bastante optimistas, cabe destacar que se realizó con una cuota de mercado muy pequeña a nivel regional, se estudió en base al 8% (14 instituciones) de organizaciones asociadas.

Capítulo 6

Análisis

6.1 Casos de uso

Un caso de uso es una descripción de las acciones de un sistema desde el punto de vista del usuario. Es una herramienta valiosa dado que es una técnica de aciertos y errores para obtener los requerimientos del sistema, justamente desde el punto de vista del usuario.

Los diagramas de caso de uso modelan la funcionalidad del sistema usando actores y casos de uso. Los casos de uso son servicios o funciones provistas por el sistema para sus usuarios.

“Un requerimiento es simplemente una declaración abstracta de alto nivel de un servicio que proporciona el sistema o una restricción de este”. (Sommerville, 2005).

6.1.1 Actores

- Encargado: Es aquella persona que puede ver y usar casi el 100% de las funcionalidades de la aplicación. Es el que se encarga de gestionar la organización y el que hará uso de funcionalidades como el control de asistencia, el ingresar personas a la organización, ver informes y publicar el listado con el orden para la toma de turnos.
- Empaquetador: Es la persona que pertenece a la organización y realizará la principal acción, la toma de turnos. El empaquetador podrá ver el listado aleatorio publicado por el encargado y saber en qué momento le toca seleccionar sus turnos para la semana. Este actor posee un código QR, el cual será escaneado por el encargado para confirmar su asistencia.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

6.1.2 Diagrama de casos de uso

La Ilustración 8: Diagrama de casos de uso. muestra a los actores, casos de uso y el sistema de la aplicación.

Se puede apreciar que el principal actor es “Encargado”, siendo participe en casi todos los casos de uso. Mientras que el actor “Empaque” (llamado Empaquetador) participa en login y toma turnos.

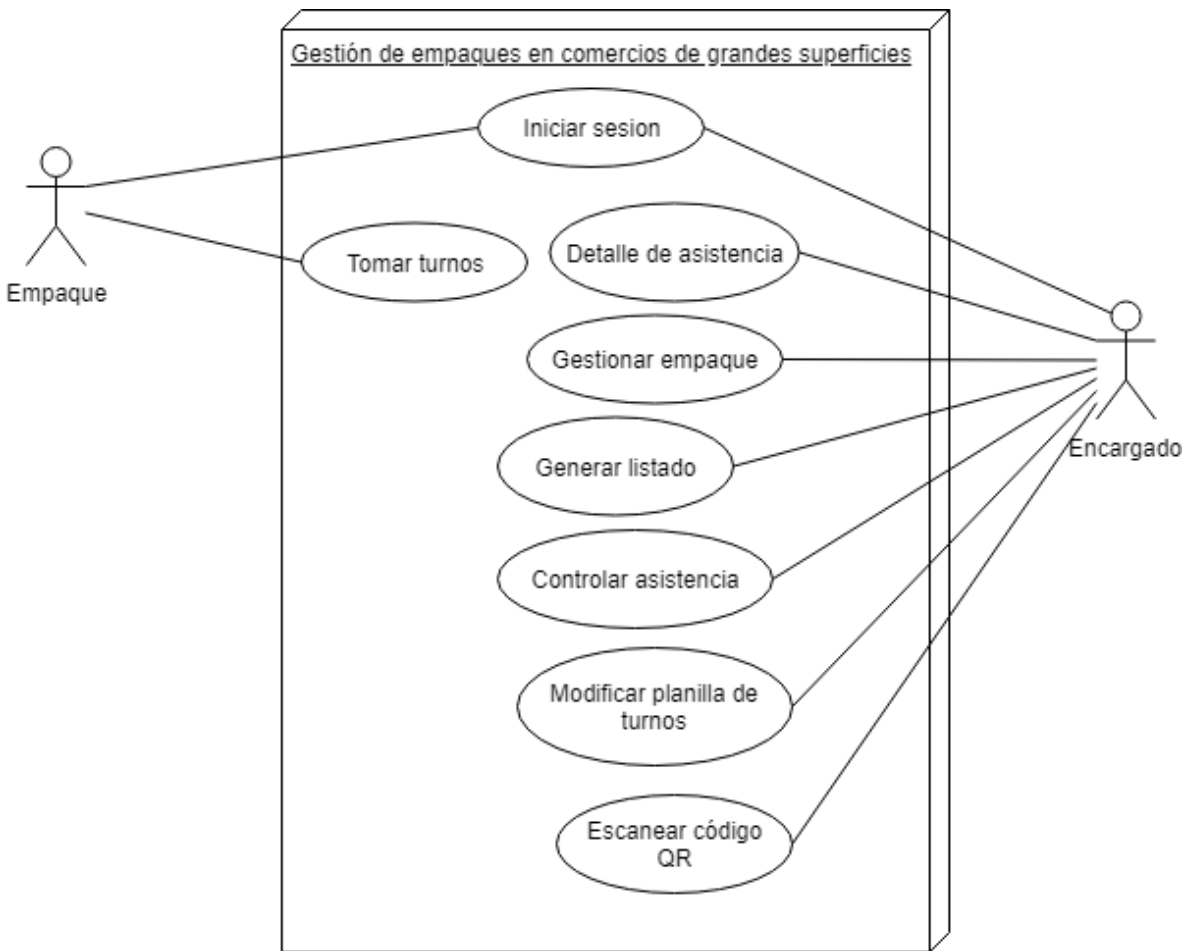


Ilustración 8: Diagrama de casos de uso.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

6.1.3 Especificación de casos de uso

A continuación, se especifica el funcionamiento de cada caso de uso.

Iniciar sesión: Es la funcionalidad que permitirá la protección y el acceso a los datos mediante cuentas de usuario y clave. Para poder hacer Login a la aplicación se debe tener una cuenta que es proporcionada por el encargado de la organización de empaquetadores. En este caso de uso participan los dos actores del sistema.

Nombre: Iniciar sesión.
ID: C01
Descripción: Permite a los usuarios acceder a la aplicación y hacer uso de sus funcionalidades.
Actores: Empaquetador y encargado.
Precondiciones: Mantener conexión a internet para interactuar con la aplicación.
<p>Flujo normal:</p> <ol style="list-style-type: none"> 1. El caso de uso comienza cuando el usuario accede a la aplicación. 2. El usuario ingresa sus datos, “usuario” y “contraseña”. 3. La aplicación verifica y autentica. 4. La aplicación da acceso al usuario.
<p>Flujo alternativo:</p> <p>3A: La aplicación no logra autenticar al usuario, por ende, niega el acceso.</p>
Postcondiciones: No aplica.

Tabla 8: Caso de uso, Login.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

Tomar turnos: Es el principal caso de uso del sistema. Es uno de los fines que busca la aplicación, mediante la cual los empaquetadores pertenecientes a la organización pueden realizar la selección de turnos para la determinada semana. En esta funcionalidad participa el actor “empaque” que es el empaquetador registrado en el sistema.

Nombre: Tomar turnos
ID: C02
Descripción: Permite a los empaquetadores realizar la acción de toma de turnos.
Actores: Empaquetador.
Precondiciones: Estar autenticado y con acceso a las funcionalidades.
<p>Flujo normal:</p> <ol style="list-style-type: none"> 1. El caso de uso comienza con el acceso del empaquetador a la aplicación. 2. Luego el empaquetador espera a que sea su turno para la selección de turnos. 3. En su momento, selecciona los turnos que desea para la semana. 4. Finalmente presiona “enviar turnos” para confirmar sus elecciones.
<p>Flujo alternativo:</p> <p>3A: Al momento de la elección, al empaquetador no le sirve ningún turno disponible por lo cual solo presiona “enviar Turnos”.</p>
Postcondiciones: No aplica.

Tabla 9: Caso de uso, toma de turnos

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

Generar detalle de asistencia: Es una funcionalidad que permite al encargado generar informes en periodos de tiempo, que por determinada será de un mes, no obstante, la aplicación permitirá variar las fechas entre las cuales se quiera generar el informe. La información que se podrá obtener es la cantidad de turnos de los empaquetadores y las faltas que tenga en el determinado tiempo.

Nombre: Generar detalle de asistencia.
ID: C03
Descripción: Mediante la captura de código QR, se genera un registro con el control de asistencia de los empaquetadores a sus turnos, con el objetivo de que el encargado pueda ver un listado con la información de la asistencia.
Actores: Encargado.
Precondiciones: Estar autenticado y con acceso a las funcionalidades.
<p>Flujo normal:</p> <ol style="list-style-type: none"> 1. El caso de uso comienza con la captura de código QR del empaquetador. 2. Luego se genera el detalle con los datos. 3. El encargado puede visualizar dicho detalle del control.
<p>Flujo alternativo:</p> <p>1B: La lectura de QR es errónea, es decir, se intenta leer un código no registrado.</p>
Postcondiciones: no aplica.

Tabla 10: Caso de uso, informe mensual.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

Gestionar empaque: Es un caso de uso esencial para el funcionamiento de la aplicación móvil, consiste en que el encargado podrá agregar, eliminar, editar o cambiar el estado (activo, inactivo) de los empaquetadores. Esta funcionalidad permite la captura y registro de los datos para que la aplicación haga uso de ellos, y a su vez sirve de respaldo para los encargados a momento de que se necesite de dicha información de los miembros de la organización.

Nombre: Gestionar empaque
ID: C04
Descripción: El encargado agrega, edita, elimina y cambia de estado a los empaques de la organización.
Actores: Encargado.
Precondiciones: Estar autenticado y con acceso a las funcionalidades.
<p>Flujo normal:</p> <ol style="list-style-type: none"> 1. El caso de uso comienza cuando el encargado selecciona entre las opciones de agregar, editar, eliminar y estado. 2. Para agregar, se ingresan los datos del empaquetador y se guarda. 3. Para editar, seleccionar la opción editar, luego modificar los campos necesarios y finalmente guardar. 4. Para eliminar, seleccionar la opción eliminar de un determinado empaquetador, confirmar la eliminación y aceptar. 5. Para cambiar estado, solo marcar o desmarcar la opción “estado”.
<p>Flujo alternativo:</p> <p>2B: Ingresar mal un dato, pasar al punto 3.</p> <p>5B: Para volver al estado anterior, volver a marcar.</p>
Postcondiciones: no aplica.

Tabla 11: Caso de uso, Gestionar empaque

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

Generar listado: Funcionalidad que complementa la toma de turnos, ya que es una acción previa que entrega un listado de los miembros de la organización (que estén activos) en un orden aleatorio, para así, hacer uso de este listado y definir el orden en la cual seleccionaran los turnos cada empaquetador.

Nombre: Generar listado
ID: C05
Descripción: Listar todos los empaquetadores activos de forma aleatoria, para que de acuerdo a esta lista se establece el orden para la selección de turnos.
Actores: Encargado.
Precondiciones: Estar autenticado y con acceso a las funcionalidades.
Flujo normal: <ol style="list-style-type: none"> 1. El caso de uso comienza cuando el encargado genera el listado random 2. Luego el encargado publica el listado para conocimiento de los empaquetadores.
Flujo alternativo: no aplica.
Postcondiciones: no aplica.

Tabla 12: Caso de uso, generar listado.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

Controlar asistencia: El encargado podrá ingresar el o los empaquetadores que falten a sus respectivos turnos con el fin de tomar medidas al respecto.

Nombre: Controlar asistencia
ID: C06
Descripción: El encargado visualiza desde el registro que entrega el scanner QR un empaque haya llegado tarde o no asiste.
Actores: Encargado.
Precondiciones: Estar autenticado y con acceso a las funcionalidades.
<p>Flujo normal:</p> <ol style="list-style-type: none"> 1. El caso de uso comienza cuando el encargado registra alguna falta para con el encargado. 2. El encargado puede visualizar las faltas de los empaquetadores.
Flujo alternativo: no aplica.
Postcondiciones: no aplica.

Tabla 13: Caso de uso, ingresar falta a turno.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

Modificar planilla de turnos: La aplicación será un sistema de línea de producto, es decir, será un sistema genérico que puede ser adoptado por distintas instituciones de supermercados. En vista de eso, las organizaciones distribuyen los horarios de sus turnos de formas diferentes, por ende, la aplicación permite que el encargado modifique los horarios de los turnos y también la cantidad de empaquetadores que pueden asistir a los turnos. Estas consideraciones hacen que el caso de uso sea bastante amplio en su desarrollo.

Nombre: Modificar planilla de turnos
ID: C07
Descripción: Adaptar la planilla de toma de turnos.
Actores: Encargado.
Precondiciones: Estar autenticado y con acceso a las funcionalidades.
<p>Flujo normal:</p> <ol style="list-style-type: none"> 1. El encargado define los horarios de la planilla. 2. El encargado define la cantidad de cupos de cada turno. 3. Se guarda la configuración.
<p>Flujo alternativo:</p> <p>1B: El encargado puede cambiar posteriormente los horarios de la planilla.</p> <p>2B: El encargado puede ingresar por error la cantidad de cupos.</p>
Postcondiciones: no aplica.

Tabla 14: Caso de uso, modificar planilla de turnos.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

Escanear código QR: Es quizás uno de las funcionalidades más llamativas, ya que es una tecnología que agiliza los procesos por su rápido uso. Se busca que el encargado pueda controlar la asistencia mediante un lector de códigos QR, código QR que cada empaquetador tendrá, para así obtener la información y posteriormente generar un informe que estará a disposición del encargado.

Nombre: Escanear código QR
ID: C08
Descripción: Controlar asistencia mediante el escaneo de código QR de cada empaquetador.
Actores: Encargado.
Precondiciones: Estar autenticado y con acceso a las funcionalidades.
<p>Flujo normal:</p> <ol style="list-style-type: none"> 1. El caso de uso comienza cuando el encargado solicita la visualización del código QR del empaquetador. 2. El encargado hace una lectura con el escaner QR de su celular. 3. El lector captura la información del código QR del empaquetador. 4. La aplicación guarda la información de la asistencia.
<p>Flujo alternativo:</p> <p>3B: El lector QR no logra capturar los datos del código del empaque por posible inactividad del empaquetador.</p> <p>4B: la aplicación no guarda la información de la asistencia por la imposibilidad de leer el código QR del empaquetador.</p>
Postcondiciones: no aplica.

Tabla 15: Caso de uso, escanear código QR.

Capítulo 7

Diseño

7.1 Diseño Front-end

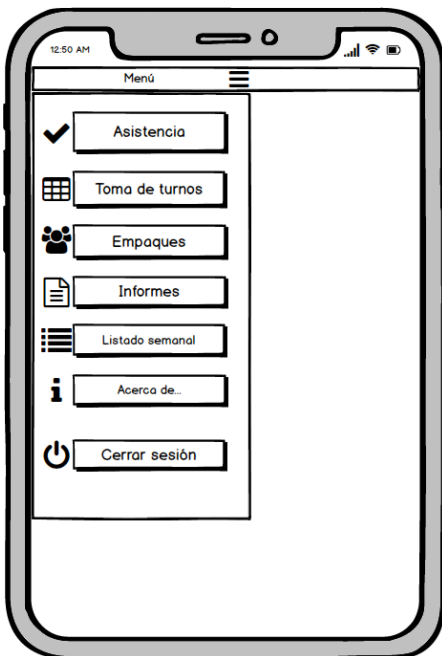
Front-end es la parte de una aplicación que interactúa con los usuarios, es por esto que se puede decir que está del lado del cliente. Es la parte de un programa o dispositivo a la que un usuario puede acceder directamente. Son todas las tecnologías de diseño que corren en el navegador web o cualquiera sea el dispositivo que se encarga de la interactividad con los usuarios.

HTML, CSS y JavaScript son los lenguajes principales del front-end de los cuales se desprenden una cantidad de frameworks y librerías que expanden sus capacidades para crear interfaz de usuario.

7.1.1 MockUp

Una herramienta muy útil a la hora de representar la idea de diseño de una aplicación es utilizar MockUps. Estos representan el cómo se verá la interfaz gráfica que esperamos tener una vez implementada la aplicación. A continuación, se presentan los principales MockUps para ambos usuarios.

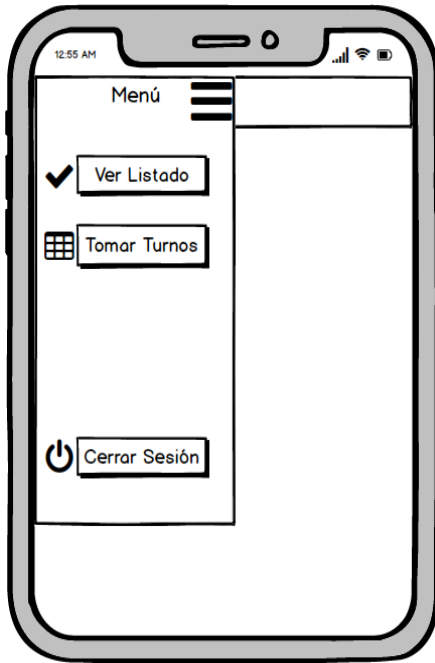
Para el usuario “Encargado” encontramos las siguientes vistas.



La barra de menú para el encargado posee opciones para la asistencia, que básicamente se usa para acceder al scanner QR y controlar la asistencia. La opción de “toma de turnos” permite tomar sus turnos para la respectiva semana. La opción “empaques” es para registrar un nuevo empaque. La opción “informes” entrega resultados de la asistencia de los empaquetadores en un determinado tiempo. “Listado semanal” es para generar un listado random el cual define el orden con el cual los empaques harán su toma de turnos. La opción “acerca de...” muestra información pertinente del desarrollador de la aplicación y el motivo por la cual se realizó la construcción de la aplicación móvil. Finalmente “cerrar sesión” cierra la sesión del usuario.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

Para el caso del “Empaquetador” encontramos las siguientes vistas.



La barra de menú para el empaquetador consta de tres posibles opciones. La primera es “ver listado” el cual sirve para que los empaquetadores puedan ver el listado que publica el encargado, el cual contiene el orden con el cual se debe tomar los turnos. Otra opción es “tomar turno” la cual es la principal función del empaquetado, ya que es aquí donde el empaquetador selecciona sus turnos para la semana. Finalmente, “cerrar sesión” la cual cierra la sesión del usuario empaquetador.

Las demás vistas representadas con MockUp se pueden encontrar en el Anexo 2: MockUps.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

7.1.2 Diseño de interfaz gráfica

Para efectos del proyecto, la aplicación móvil desarrollada busca considerar un diseño que sea representativo para con el grupo de personas que le darán uso a la aplicación. Esto conlleva a utilizar terminologías de índole conocido para todos y que logré expresar de la mejor manera posible las acciones o actividades que llevan a cabo las organizaciones de empaquetadores en supermercados.

El diseño considera títulos descriptivos, botones intuitivos e iconos representativos. Las siguientes ilustraciones muestran el diseño de la aplicación ya funcional.

Para el usuario “encargado” una de sus principales vistas es el formulario para registrar nuevos y configurar la planilla semanal. A continuación, la muestran dichas vistas.

Con respecto al formulario, se ingresan los datos del nuevo empaquetado, su nombre, apellido, Email (posteriormente será su usuario para logear), Rut, código, carrera, institución y el rol (encargado o empaquetador). Básicamente al presionar “guardar” se envían los campos a la base de datos y se completa el registro.

The image shows a mobile application form titled "Formulario empaque". It consists of several input fields, each with an icon and a placeholder text:

- Nombres: Ingresar nombres (with a person icon)
- Apellidos: Ingresar apellidos (with a person icon)
- Email: Ingresar Email (with an envelope icon)
- Rut: Ingresar RUT (with a document icon)
- Código: Ingresar número de credencial (with a document icon)
- Carrera: Ingresar carrera de estudio (with a house icon)
- Institución: Ingresar casa de estudio (with a house icon)
- Rol: A dropdown menu.

 At the bottom of the form, there are two buttons: a blue button labeled "GUARDAR" and a red button labeled "CANCELAR".

Ilustración 9: Vista de la aplicación, Formulario de registro.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

Con respecto a la configuración de la planilla de toma de turnos, el encargado puede definir la cantidad de cupos que tendrá cada turno, a continuación, en la Ilustración 10: Vista de la aplicación, Ajuste de planilla. se muestra la vista donde se observa que el día Lunes a las 8:30, hay un cuadrado con el número dos, esto indica que para ese turno se definió un cupo máximo para dos empaquetadores. Los turnos que tengan un rectángulo con “N° CUPOS” es porque aún no se define la cantidad de cupos. Una vez se haya definido el cupo para todos los turnos, se presiona el botón guardar y la configuración es enviada a Firebase, donde se aloja dicha configuración.

← Ajustes De Planilla			
GUARDAR			
Hora	Lunes	Martes	
8:30	2	2	
9:30	1	2	
10:30	2	N° CUPOS	
12:00	4	N° CUPOS	
13:30	N° CUPOS	N° CUPOS	
15:30	N° CUPOS	N° CUPOS	
17:00	N° CUPOS	8	

Ilustración 10: Vista de la aplicación, Ajuste de planilla.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

La vista principal para el empaquetador es la toma de turnos, la cual consiste en que el empaquetador puede seleccionar los turnos que desee para la semana. Se puede apreciar que en la planilla se ve la cantidad de cupos que tiene un turno y también cuantas personas han tomado los turnos. Con esto, el empaquetador puede saber si hay cupos disponibles para los turnos, por ejemplo, el día lunes a las 8:30 no tiene cupos disponibles ya que tiene especificado “2/2”, para el caso del día lunes a las 12:00 existen 4 cupos, de los cuales están todos disponibles. Cabe destacar que los empaques solo deben tocar el turno que quieren y luego confirmar la toma de sus turnos.

A continuación, la Ilustración 11: Vista de la aplicación, Toma de turnos para empaques muestra la vista de la toma de turnos para empaquetadores.

← Tomar Turnos				
Hora	Lunes	Martes	Miércoles	Juev
8:30	0/ 1	0/ 1	0/ 1	0/ 2
9:30	0/ 1	2/ 2 100 100	0/	0/
10:30	0/ 2	1/ 10	0/ 5	0/
12:00	0/ 4	0/ 4	0/	0/
13:30	0/ 2	0/	0/	0/
15:30	0/ 5	0/	0/	0/
17:00	0/ 5	1/ 8 100	0/	0/
19:00	0/ 7	0/	0/	0/

Ilustración 11: Vista de la aplicación, Toma de turnos para empaques

Las demás vistas de la aplicación se pueden encontrar en el Anexo 3: Diseño de interfaz gráfica

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

7.1.3 Diseño de navegación

En los dos tipos de usuarios que interactúan con la aplicación, es el usuario “encargado” el que posee la mayor parte de las vistas, mientras que el usuario “empaque” solo ve las que necesita.

La Ilustración 12: Diseño de navegación, usuario encargado muestra la navegación del usuario “encargado”.

Primero que todo se debe hacer login con su email y contraseña para ingresar a Home. Luego, tiene las opciones de asistencia donde puede ingresar a scanner QR e informes. Desde Home se puede ingresar a “Generar listado” donde se publica el listado random para establecer el orden de toma de turnos. También se puede ingresar a la toma de turno donde el encargado selecciona sus turnos. En la vista “empaques”, se ve el listado de los empaques registrados, además se puede ingresar al formulario para agregar más usuarios. La vista “acerca de” entrega información del desarrollador de la aplicación. Finalmente se puede cerrar sesión.

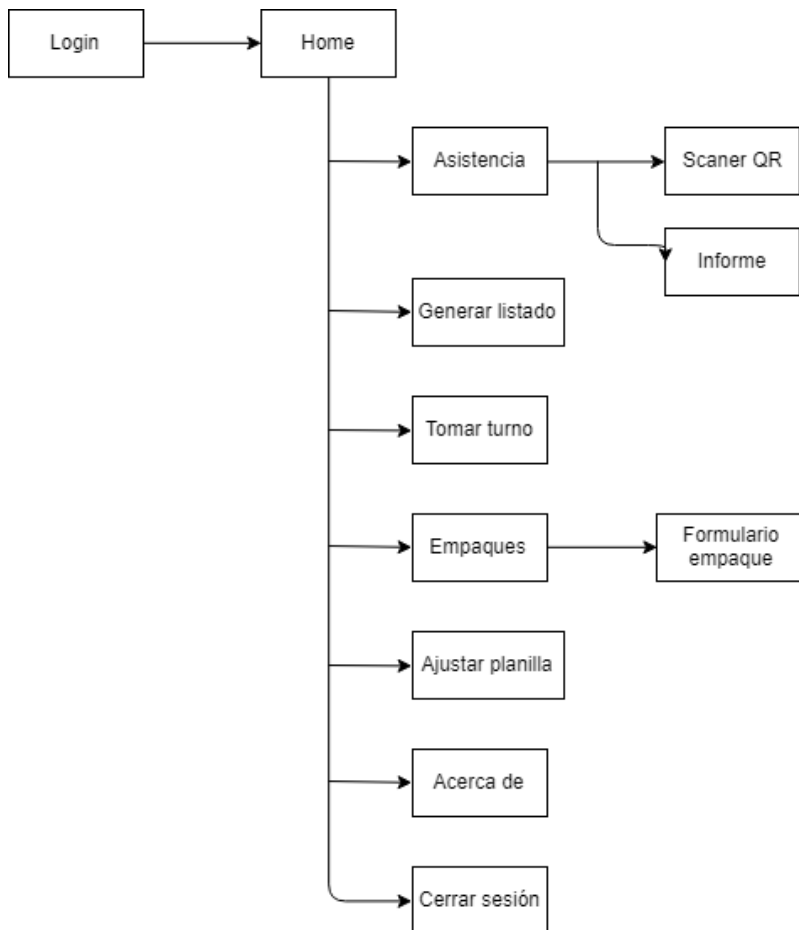


Ilustración 12: Diseño de navegación, usuario encargado

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

La Ilustración 13: Diseño de navegación, usuario empaquetador muestra la navegación del usuario Empaquetador. Esta comienza en la página de login, donde el usuario debe ingresar su email y contraseña para así autenticarse e ingresar a la vista principal, el Home. Una vez dentro de Home, el usuario puede ingresar a “ver listado”, donde puede observar el listado generado de manera aleatoria que establece el orden con que se tomaran los turnos. La vista “tomar turno” es donde el usuario empaquetador selecciona los turnos que desea para la semana. La vista “acerca de” entrega información sobre el desarrollador de la aplicación. Finalmente, el usuario puede cerrar sesión, lo cual lleva a la página de login.

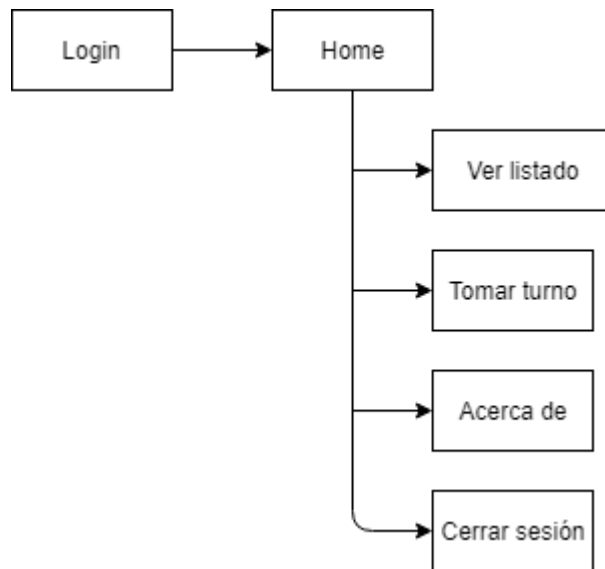


Ilustración 13: Diseño de navegación, usuario empaquetador

7.2 Diseño Back-end

El back-end es la parte del desarrollo que se encarga de que toda la lógica de una aplicación funcione. Se trata del conjunto de acciones que pasan en una web pero que no vemos como, por ejemplo, la comunicación con el servidor. En los siguientes puntos se muestra el diseño físico de Firebase, y el diseño del servidor web.

7.2.1 Modelo físico de base de datos Firebase

A continuación, se presenta el modelo físico de la base de datos de Firebase, el cual cuenta con una consola que permite observar el comportamiento de esta entre otras cosas. La consola de Firebase permite observar el esquema de la base de datos, observar el comportamiento de las cuentas de usuarios, el Storage del servidor, las reglas que se pueden configurar para la base de datos, entre otras cosas.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

La Ilustración 14: Modelo físico de Firebase, parte 1 muestra como está distribuida la gestión de las “Colecciones” y los “documentos” de la base de datos. La primera colección que existe es “empresas” la cual irá abarcando las empresas ingresadas a la base de datos, esto en forma de documentos. Luego, el documento “111111” es el Rut de una de las dos empresas que tiene la base de datos, a su vez, estos documentos abarcan una nueva colección llamada “empaques” y tres campos llamados “confiTornos”, “turnos”, “listOrden”. La colección “empaques” abarca a cada empaque con sus respectivos atributos (se puede apreciar en la ilustración 14). Los campos “confiTornos” almacena como arreglo de objetos la configuración de turnos que define el encargado desde la aplicación móvil, mientras que el campo “turnos” almacena como arreglo los turnos tomados por los empaquetadores, finalmente, el campo “listOrden” almacena el arreglo con el orden de toma de turnos para empaquetadores.

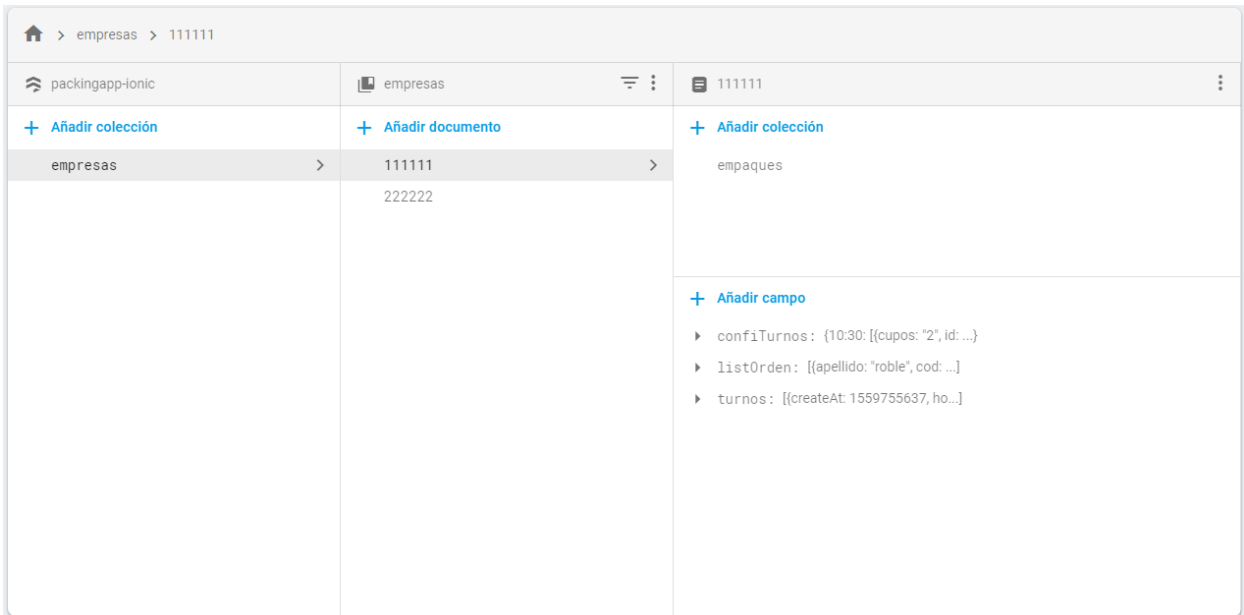


Ilustración 14: Modelo físico de Firebase, parte 1

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

En la Ilustración 15: Modelo físico de Firebase, parte 2 se aprecia los documentos contenidos en la colección “empaques”, los cuales son los registros de los empaquetadores, se puede apreciar que cada empaque (o documento) tiene campos que reflejan los atributos de cada empaquetador almacenado en la base de datos.

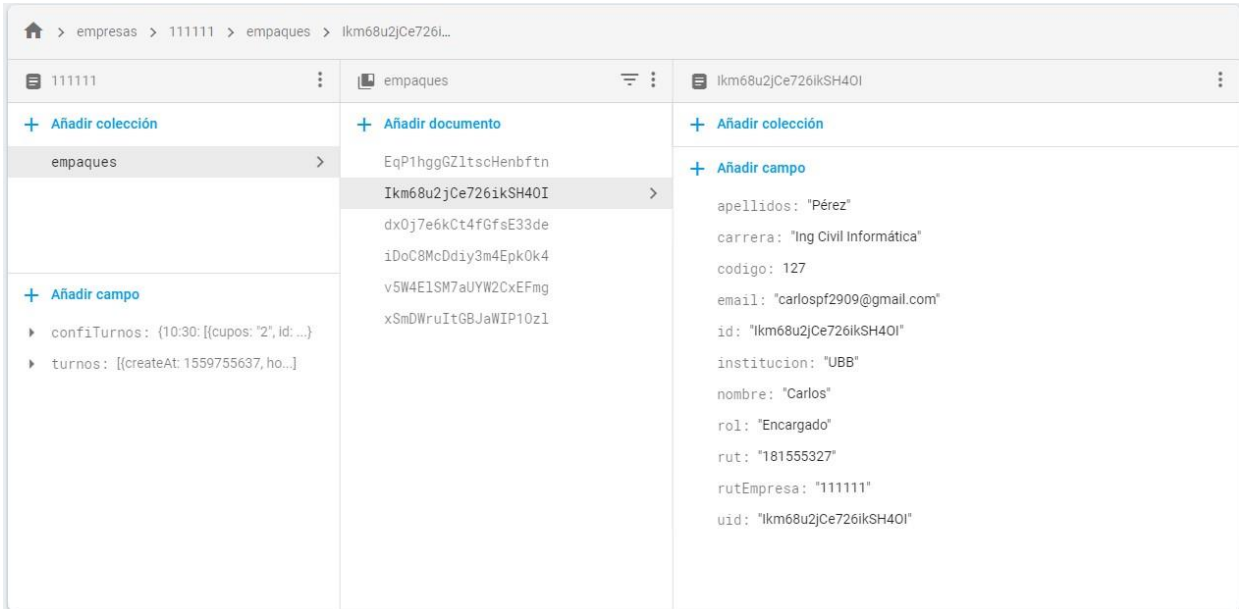


Ilustración 15: Modelo físico de Firebase, parte 2

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

7.2.2 Diseño de Servidor Web

A continuación, se muestra la Ilustración 16: Arquitectura de módulos y componentes, la cual es un esquema que señala algunos de los archivos que interactúan con el módulo central de la aplicación, AppModule. Dentro de los principales se pueden apreciar los siguientes.

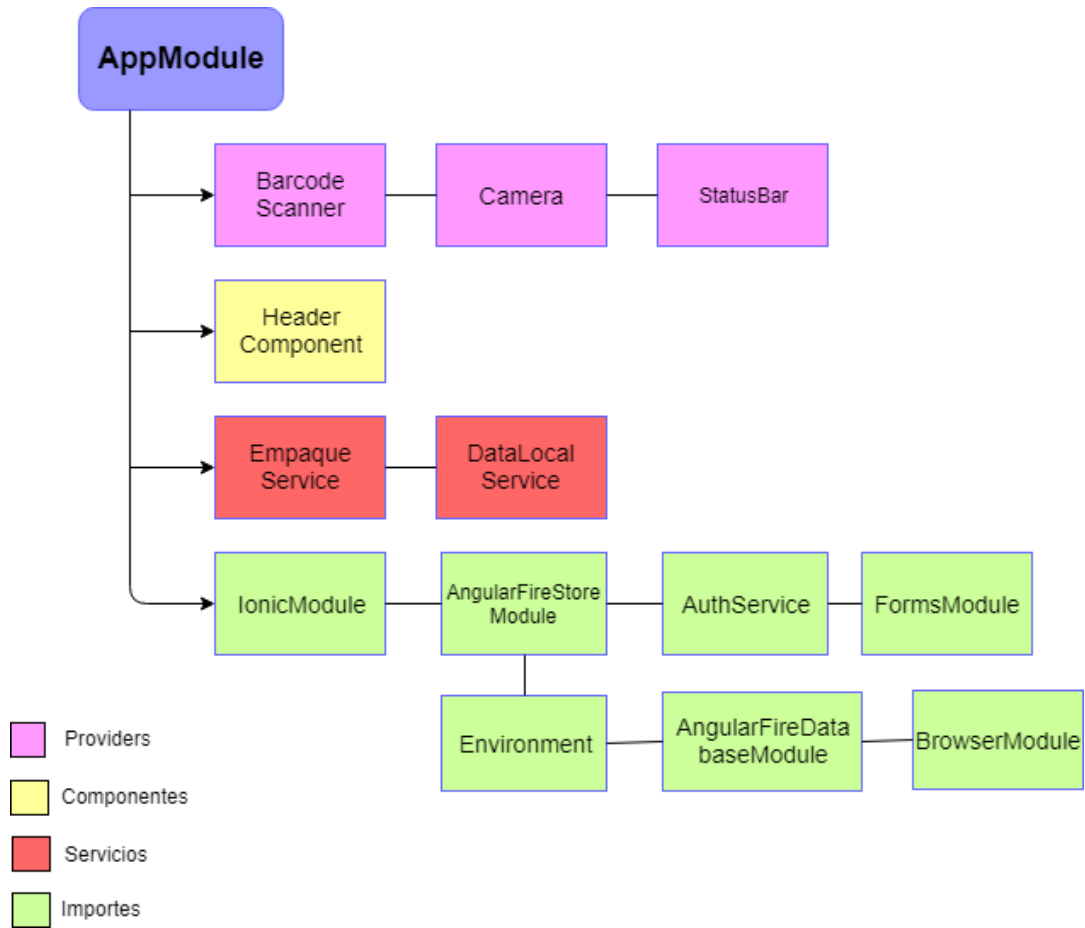


Ilustración 16: Arquitectura de módulos y componentes

Capítulo 8

Pruebas

8.1 Pruebas unitarias

Las pruebas unitarias son formas de comprobar el correcto funcionamiento de una unidad de código. Por ejemplo, en diseño estructurado o en diseño funcional, una función o un procedimiento. (Apiumhub, 2017).

Las características de las pruebas unitarias son:

- Automatizable: Los resultados se deben automatizar, de forma que podemos hacer las pruebas de forma individual o en grupos.
- Completas: Pequeños test sobre parte del código o funcionalidad, pero que al final, se deben comprobar en su totalidad.
- Repetible: En caso de repetir las pruebas, el resultado debe ser siempre el mismo.

8.1.1 escenarios de prueba

A continuación, se especifican los casos de pruebas sobre las principales funcionalidades de la aplicación. La lista de los escenarios definidos es:

1. Inicio sesión.
2. Registrar usuario (empaquete).
3. Tomar turno.
4. Configurar planilla.
5. Escanear código QR.

8.1.2 Metodología de pruebas

La metodología para llevar a cabo las pruebas es el enfoque de caja negra, la cual se encarga de verificar la funcionalidad sin tomar en cuenta la estructura interna del código. Básicamente se centra en las entradas y salidas del sistema.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

8.1.3 Pruebas funcionales

A continuación, se presentan las tablas de cada uno de los escenarios de prueba definidos anteriormente (8.1.1).

1. Escenario: Inicio sesión.

Fecha de prueba: 13-06-2019.

ID	Objetivo	Contexto	Entrada	Salida	Evento	Resultado	Evaluación
PU01	Determinar si un usuario tiene acceso al sistema.	Email y contraseña en Firebase.	Email y contraseña.	Se ingresa a la aplicación.	Al tocar iniciar.	Ingresa a Home.	Éxito.
PU02	Determinar si un usuario tiene acceso al sistema con datos erróneos.	Email y contraseña en Firebase.	Email o contraseña erróneas.	Mensaje de que los datos no son válidos.	Al completar los campos.	Se permanece en la página de Login.	Éxito.

Tabla 16: Prueba unitaria, Login.

2. Escenario: Registrar usuario (empaque).

Fecha de prueba: 13-06-2019.

ID	Objetivo	Contexto	Entrada	Salida	Evento	Resultado	Evaluación
PU03	Ingresar un usuario llenando todos los campos exigidos.		Nombres, apellidos, email, Rut, código, Carrera, institución y rol completados.	Se ingresa el usuario a Firebase.	Tocar botón guardar.	Se ve el usuario registrado en el listado de empaquetadores.	Éxito.
PU04	Ingresar un usuario dejando campos sin completar.		Nombres, apellidos, email, Rut, código, Carrera, institución o rol incompletos.	No se ingresa el usuario a Firebase.	Botón guardar desactivado.	Se marca con rojo campo incompleto.	Éxito

Tabla 17: Prueba unitaria, registrar usuario (empaque).

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

3. Escenario: **Tomar turno.**

Fecha de prueba: 13-06-2019.

ID	Objetivo	Contexto	Entrada	Salida	Evento	Resultado	Evaluación
PU05	Usuario toma turnos para la semana.	Seleccionar turnos deseados.	Turnos seleccionados.	Mensaje de turnos guardados con éxito.	Tocar botón “confirmar”	Turnos registrados en Firebase.	Éxito.
PU06	Usuario no toma turnos para la semana.	No selecciona turnos.	No hay selección de turnos	Ninguno.	No se toca el botón “confirmar”	No se registran turnos para el usuario en Firebase.	Éxito.

Tabla 18: Prueba unitaria, tomar turno.

4. Escenario: **Configurar planilla.**

Fecha de prueba: 13-06-2019.

ID	Objetivo	Contexto	Entrada	Salida	Evento	Resultado	Evaluación
PU07	Usuario “encargado” define cantidad de cupos para los turnos.	Ingresar número de cupos para turnos	Número de cupos.	Número de cupos en el turno.	Tocar botón “guardar”.	Planilla con número de cupos de cada turno.	Éxito.
PU08	Usuario “encargado” no define cantidad de cupos para algún turno.	No se ingresa número de cupos para algún turno.	Ninguna.	Mensaje e “N° cupos” en el turno.	Ninguno.	Turno sin cupos disponibles.	Éxito.

Tabla 19: Prueba unitaria, configurar planilla.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

5. Escenario: **Escanear código QR.**

Fecha de prueba: 13-06-2019.

ID	Objetivo	Contexto	Entrada	Salida	Evento	Resultado	Evaluación
PU09	Obtener dato existente mediante lectura de código QR.	El scanner lee código QR y obtiene un dato existente en base de datos	Dato capturado de Scanner.	Código y nombre de algún usuario de la organización.	Tocar botón "scan"	Mensaje de control de asistencia.	Éxito.
PU10	Obtener dato inexistente mediante lectura de código QR.	El scanner lee un dato que no pertenece a la base de datos.	Dato desconocido leído por scanner.	Ninguno.	Tocar botón "scan"	No se registra control de asistencia.	Fallo.

Tabla 20: Prueba unitaria, escanear código QR.

8.2 Pruebas de rendimiento

Para analizar el comportamiento de rendimiento de la aplicación hay que tener en cuenta diversos factores, como por ejemplo el tiempo de respuesta del servidor, la cantidad de usuarios simultáneos que soporta el servidor, la cantidad de peticiones simultaneas que soporta el servidor, la conectividad de internet, la capacidad de almacenamiento del servidor, etc.

Para el desarrollo de la aplicación, se posee una cuenta Firebase de pago llamado Flame Plan (ver **¡Error! No se encuentra el origen de la referencia.**) con a cual se tiene un amplio margen tanto de almacenamiento como de peticiones simultaneas.

Las necesidades de consumo que tiene la aplicación son claramente suplidos por el servidor Firebase.

A continuación, se detalla los aspectos a destacar:

1. Número de usuarios simultáneos: Se estima que por empresa existen en promedio 40 usuarios, y que en la región se logre implementar en 20 organizaciones, lo cual nos arroja un total de 800 usuarios que pueden acceder de manera simultánea al sistema. Firebase, con el plan que se tiene, soporta hasta 100.000 conexiones simultaneas, lo cual claramente cubre la necesidad.
2. Registros de usuarios en la base de datos: El registro de los usuarios no es un problema para el servidor de Firebase, ya que cuenta con 50 GB de almacenamiento para registro en su base de datos, lo cual es más que suficiente. Tomando en cuenta los datos del punto anterior, con 800 usuarios registrados, además de la información entregada en el “Anexo 4: Tamaño en Bytes por usuario registrado” que calcula 160 bytes por usuario, esto nos da un total de 128.000 Bytes almacenados por registro de usuarios, lo cual es equivalente a 1,024 Megabit.
3. Tiempo de respuesta: Esta puede variar de acuerdo a la conexión a internet que se tiene. En cualquiera de los casos, con una velocidad baja de internet los tiempos de respuesta son mínimos hasta la obtención de resultados.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

La

Ilustración 17: Prueba de rendimiento con JMeter. muestra una prueba realizada con el programa JMeter, el cual permite realizar simulaciones con el objetivo de saber los tiempos de respuesta de los servidores. Con esto se espera ejemplificar que el rendimiento del servidor de firebase.google.com posee una respuesta positiva frente a múltiples peticiones por parte de varios usuarios de manera simultánea.

La prueba se realizó con 100 usuarios realizando peticiones de manera simultánea y este, tardo 8 segundos en procesar cada petición.

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time(ms)
13	22:10:07.798	Thread Group 1-25	HTTP Request	2567	Success	165552	246	229	19
14	22:10:08.898	Thread Group 1-52	HTTP Request	1485	Success	165552	246	574	13
15	22:10:07.561	Thread Group 1-17	HTTP Request	2825	Success	165552	246	219	18
16	22:10:08.154	Thread Group 1-37	HTTP Request	2235	Success	165552	246	386	167
17	22:10:08.035	Thread Group 1-33	HTTP Request	2356	Success	165552	246	494	42
18	22:10:08.525	Thread Group 1-49	HTTP Request	1870	Success	165552	246	233	15
19	22:10:08.784	Thread Group 1-58	HTTP Request	1812	Success	165552	246	610	129
20	22:10:07.828	Thread Group 1-26	HTTP Request	2577	Success	165552	246	744	17
21	22:10:08.274	Thread Group 1-41	HTTP Request	2136	Success	165552	246	273	49
22	22:10:08.007	Thread Group 1-32	HTTP Request	2404	Success	165552	246	512	70
23	22:10:08.422	Thread Group 1-46	HTTP Request	1996	Success	165552	246	230	16
24	22:10:08.363	Thread Group 1-44	HTTP Request	2066	Success	165552	246	221	15
25	22:10:07.506	Thread Group 1-15	HTTP Request	2931	Success	165552	246	228	18
26	22:10:09.070	Thread Group 1-67	HTTP Request	1414	Success	165552	246	479	276
27	22:10:08.303	Thread Group 1-42	HTTP Request	2219	Success	165552	246	237	20
28	22:10:08.393	Thread Group 1-45	HTTP Request	2132	Success	165552	246	219	16
29	22:10:09.110	Thread Group 1-59	HTTP Request	1467	Success	165552	246	456	247
30	22:10:09.079	Thread Group 1-68	HTTP Request	1543	Success	165552	246	489	278
31	22:10:08.842	Thread Group 1-50	HTTP Request	1819	Success	165552	246	560	72
32	22:10:08.601	Thread Group 1-52	HTTP Request	2067	Success	165552	246	219	15
33	22:10:07.729	Thread Group 1-22	HTTP Request	2941	Success	165552	246	216	17
34	22:10:08.572	Thread Group 1-51	HTTP Request	2102	Success	165552	246	218	14
35	22:10:08.630	Thread Group 1-53	HTTP Request	2087	Success	165552	246	223	17
36	22:10:09.458	Thread Group 1-80	HTTP Request	1311	Success	165552	246	219	14
37	22:10:07.946	Thread Group 1-30	HTTP Request	2838	Success	165552	246	576	130
38	22:10:07.859	Thread Group 1-27	HTTP Request	2941	Success	165552	246	693	219
39	22:10:07.300	Thread Group 1-9	HTTP Request	3515	Success	165552	246	413	167
40	22:10:07.450	Thread Group 1-14	HTTP Request	3372	Success	165552	246	260	19
41	22:10:09.459	Thread Group 1-81	HTTP Request	1368	Success	165552	246	218	13
42	22:10:07.680	Thread Group 1-21	HTTP Request	3190	Success	165556	246	223	15
43	22:10:08.543	Thread Group 1-50	HTTP Request	2472	Success	165552	246	220	19
44	22:10:08.720	Thread Group 1-56	HTTP Request	2314	Success	165552	246	867	190

Ilustración 17: Prueba de rendimiento con JMeter.

8.3 Pruebas de usuario

8.3.1 Usabilidad

La usabilidad en una aplicación móvil es muy importante a la hora de entregar un producto, ya que si este aspecto está bien definido y diseñado es un punto a favor para la aplicación. Para asegurar una correcta usabilidad, se procura usar colores adecuados, títulos apropiados y una navegación corta entre vistas.

Se realizaron pruebas a usuarios del supermercado Jumbo Chillán, las cuales arrojaron un resultado positivo para una versión beta de la aplicación de gestión.

Se les propuso a 5 usuarios que probaran la aplicación (3 encargados, 2 empaques). Entre las actividades que tenía que realizar un encargado están:

Para encargados

N°	Actividad	%Completado
1	Ingresar al menú de inicio de la aplicación	100
2	Escanear un código QR	66
3	Generar un listado para ordenar la toma de turnos y publicarlo	66

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

4	Ver registro de empaques	100
5	Agregar un empaque al sistema	100
6	Ingresar cantidad de cupos a un determinado turno	33
7	Tomar turnos	100
8	Ver información de contacto del desarrollador	100
9	Visualizar los turnos tomados	66

Tabla 21: Pruebas de usuario, encargados.

Se puede apreciar en la Tabla 21: Pruebas de usuario, encargados. que las actividades 2, 3 y 9 se completaron por 2/3 (66%) de los usuarios, lo cual es un rango aceptable. Además, son actividades que se pueden mejorar a futuro en la aplicación.

Para empaquetadores

Entre las actividades para los empaquetadores están las siguientes:

N°	Actividad	%Completado
1	Ingresar al menú de inicio de la aplicación.	100
2	Tomar turnos.	100
3	Información de contacto del desarrollador	100
4	Ver listado de orden para toma de turno.	100

Tabla 22: Pruebas de usuario, empaques.

La Tabla 22: Pruebas de usuario, empaques. muestra un resultado satisfactorio completando todas las actividades en su cabalidad.

El detalle de las bitácoras aplicadas a los usuarios de manera individual se puede apreciar en el Anexo 5: Pruebas de usuario, Jumbo Chillán.

8.3.2 Eficiencia

En este punto, el aspecto a medir es la eficiencia en los tiempos de respuesta de las funcionalidades, ya sea en la respuesta del servidor, como en la navegación entre las distintas vistas de la aplicación. Los usuarios, al probar las funcionalidades, no señalaron quejas al respecto. En las observaciones se detalla más que todo aspecto de visual y de definición de conceptos.

La aplicación entrega respuestas de no más allá de 3 segundos (dependiendo de la conectividad de internet), por lo que las respuestas cumplen con el intervalo esperado por el usuario.

8.4 Conclusiones de pruebas

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

Como se logra ver en las pruebas realizadas, la mayor parte de estas fueron exitosas, sin embargo, aún hay cabos sueltos que se pueden mejorar y pulir para un funcionamiento más adecuado. Aspectos como las acciones a tomar cuando un usuario no selecciona sus turnos, son las que se pueden mejorar con las recomendaciones de encargados de organizaciones de empaquetadores.

En cuanto a la seguridad, se considera una aplicación que cumple con las reglas y condiciones mínimas apropiadas para la confidencialidad y disponibilidad de los datos mediante un filtro de acceso (Login). Además, como dato anexo, mediante código, usando métodos de Angular (CanActivate), se especificó que solo se puede acceder a las vistas de la aplicación (excepto login) estando autenticado.

Como conclusión, gracias a estas pruebas se puede observar las funcionalidades que operan correctamente y las que se pueden mejorar, para así desarrollar un producto de mayor calidad.

Conclusiones

El desarrollo de un proyecto de gran tamaño es muy difícil cuando no se planifica ni se gestiona de manera correcta, por ende, el utilizar una metodología como la iterativo incremental ayuda a organizar de mejor manera la forma de trabajo durante el ciclo de vida del proyecto, desde definir los requisitos, hasta las pruebas finales del software. Por otro lado, el conocer herramientas que simplifiquen el desarrollo, es esencial para avanzar de manera más rápida y eso se ha comprobado realizando este proyecto.

Con respecto a la arquitectura usada en el desarrollo, el modelo MV-VM (Modelo-Vista Vista-Modelo), es una muy buena opción al momento de usar el Framework Angular, ya que angular trabaja de una modalidad un tanto diferente a MVC. Angular es basado en componentes, por ende, los modelos interactúan entre las vistas para relacionar los datos que son obtenidos desde un servidor web (Firebase).

Desde la perspectiva de un servidor web en la nube, se puede concluir que la elección de una nueva tecnología como las bases de datos NoSQL es una opción muy buena si se desea trabajar con tecnologías emergentes. En relación a lo aprendido en el desarrollo del de la aplicación, una base de datos no relacional es

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

mucho más simple de utilizar que una base de datos SQL, pero la desventaja está en la documentación, ya que no hay una métrica o estándar que permita modelar una base de datos NoSQL.

Realizando las pruebas se puede concluir que la aplicación es bastante robusta y con resultados exitosos en sus funcionalidades. Además, se puede concluir que las mejoras que se pueden hacer a la aplicación son más orientadas a la interfaz gráfica.

El desarrollo del proyecto me ha proporcionado bastante experiencia enriquecedora con respecto a nuevas tecnologías y herramientas, como lo son Angular, Ionic, Firebase, JavaScript y herramientas complementarias que ayudan a realizar la documentación.

Finalmente, se puede concluir que el proyecto fue realizado de buena manera, cumpliendo con los objetivos definidos y las experiencias personales propuestas.

Trabajos futuros

Se consideran como trabajos futuros, aquellas funcionalidades que se puedan agregar o mejorar alguna ya existente, esto con el fin de mejorar el servicio entregado por la aplicación PackinApp. A continuación, se detalla las siguientes funcionalidades que se contemplan a futuro.

- Implementar un módulo que permita realizar cambios de turnos en la planilla entre usuarios.
- Se puede mejorar la interfaz gráfica, esto para que sea más llamativa para los usuarios.
- Mejorar las validaciones de todos los campos que soliciten datos.
- Implementar un módulo que permita designar cajas a los empaquetadores.
- Mejorar información mostrada en informes de asistencia.
- Exportar la planilla de turnos en formato PDF o PNG.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

- Incluir un modo tutorial al acceder por primera vez a la aplicación, para así evitar confusión a usuarios nuevos.

Bibliografía

- A.G., S. d. (Enero de 2019). Obtenido de <http://www.supermercadosdechile.cl/estadisticas/>
- Alder, D. B. (2000). *Draw.io*. Obtenido de <https://about.draw.io/>
- Alvarez, M. A. (04 de 12 de 2017). *Servicios-Angular*. Obtenido de <https://desarrolloweb.com/articulos/servicios-angular.html>
- Angular oficial*. (s.f.). Obtenido de <https://angular.io/docs>
- Apiumhub. (24 de 08 de 2017). Obtenido de <https://apiumhub.com/es/tech-blog-barcelona/beneficios-de-las-pruebas-unitarias/>
- Eich, B. (2016). *JavaScript*. Obtenido de <https://www.javascript.com/>
- Google. (s.f.). *Firestore*. Obtenido de <https://firebase.google.com/?hl=es-419>
- Ionic Framework*. (2019). Obtenido de <https://ionicframework.com/>
- ISO25000*. (s.f.). Obtenido de <https://iso25000.com/index.php/normas-iso-25000?limit=4&limitstart=0>
- Larsson, A. (2011). *wikipedia*. Obtenido de [https://es.wikipedia.org/wiki/Dia_\(programa\)](https://es.wikipedia.org/wiki/Dia_(programa))

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

- Limited, B. (2019). *Bizagi*. Obtenido de <https://www.bizagi.com/es>
- Mazuelos, j. (2017). Obtenido de <http://katakuntur.jeanmazuelos.com/es>
- Microsoft. (2016). *Office*. Obtenido de <https://www.office.com/?omkt=es-cl>
- Pressman, R. S. (2010). *Ingeniería del software Un enfoque practico*. Mcgraw-Hill.
- Ricart, S. (2018). *Survio*. Obtenido de <https://www.survio.com/es/>
- Ruiz, M. (2017). Obtenido de <https://openwebinars.net/blog/que-es-firebase-de-google/>
- Shaw, D. G. (1994). *An introduction to Software Architecture*.
- Sommerville, I. (2005). *Ingeniería del software*.
- Twitter. (2018). *Bootstrap*. Obtenido de <https://getbootstrap.com/>
- w3schools. (s.f.). *HTML* . Obtenido de https://www.w3schools.com/html/html_intro.asp

Anexo 1: Encuesta a Jumbo Chillán

Encuesta aplicada a empaquetadores Jumbo Chillan

Pregunta 1: Se puede apreciar en la Ilustración 18: Encuesta a empaquetador, pregunta 1. Que la gran mayoría de los encuestados prefieren hacer uso de una aplicación móvil por sobre una página web.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

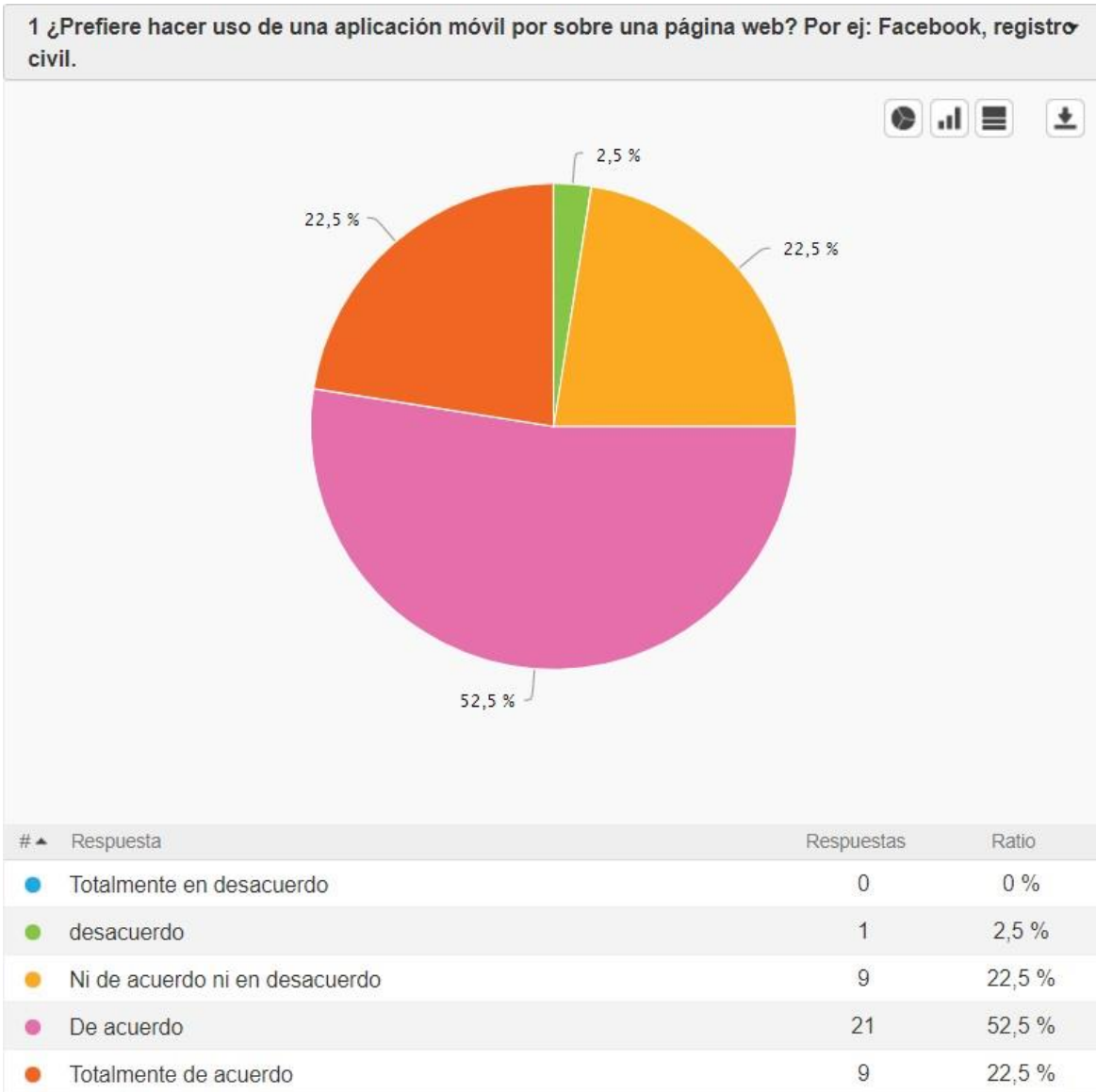


Ilustración 18: Encuesta a empaquetador, pregunta 1.

Pregunta 2: Se puede apreciar, en Ilustración 19: Encuesta a empaquetador, pregunta 2. que el 57,5% de los encuestados está de acuerdo en que una aplicación móvil mejoraría la transparencia de la toma de turno, mientras que el resto o no está de acuerdo o tiene una opinión relativa, esto se puede deber a que desconocen el cómo es el funcionamiento de la aplicación con respecto a la toma de turnos.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

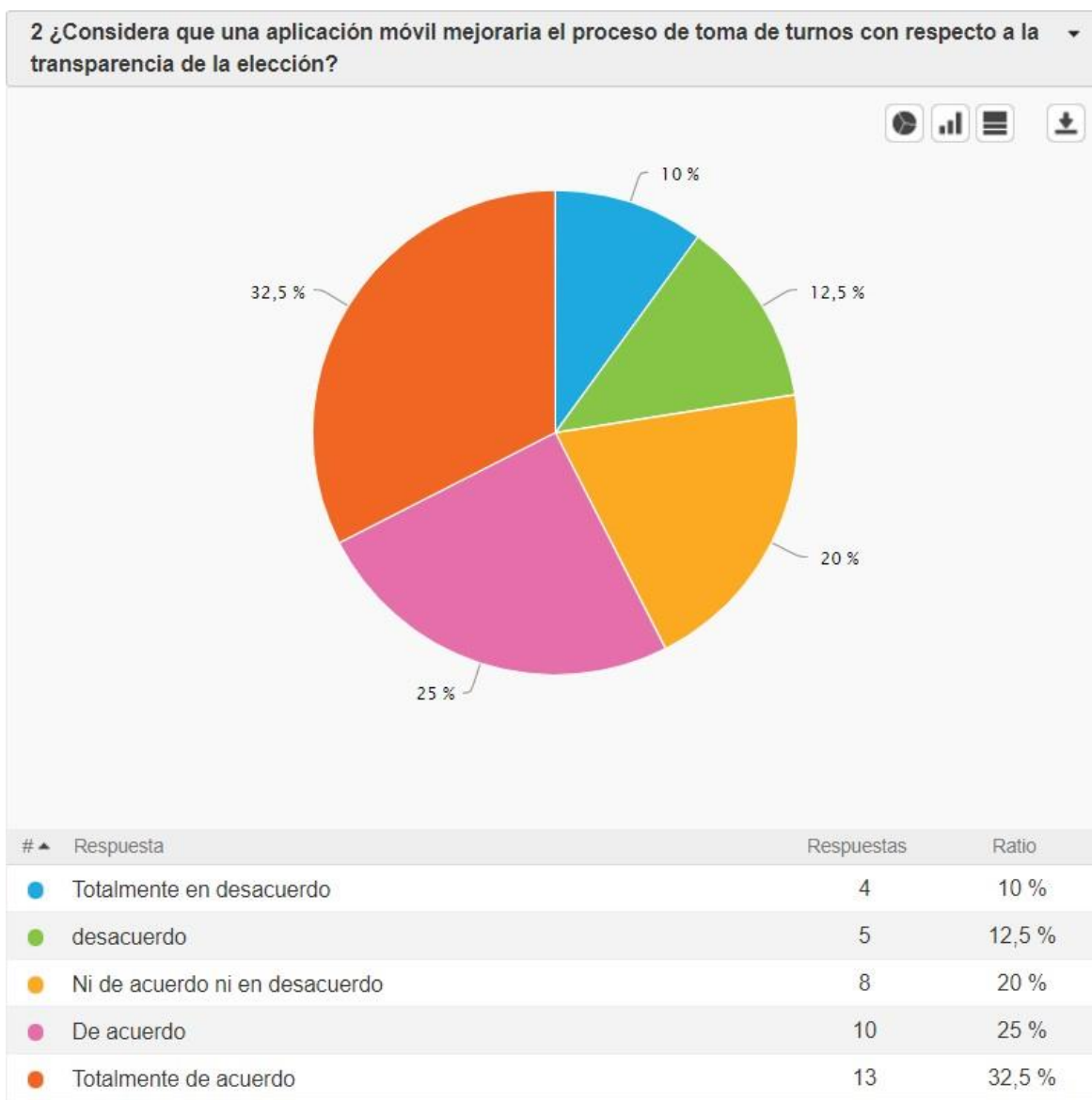


Ilustración 19: Encuesta a empaquetador, pregunta 2.

Pregunta 3: Se puede apreciar en la Ilustración 20: Encuesta a empaquetador, pregunta 3. que casi todos los encuestados consideran que tienen un amplio manejo con aplicaciones móviles.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

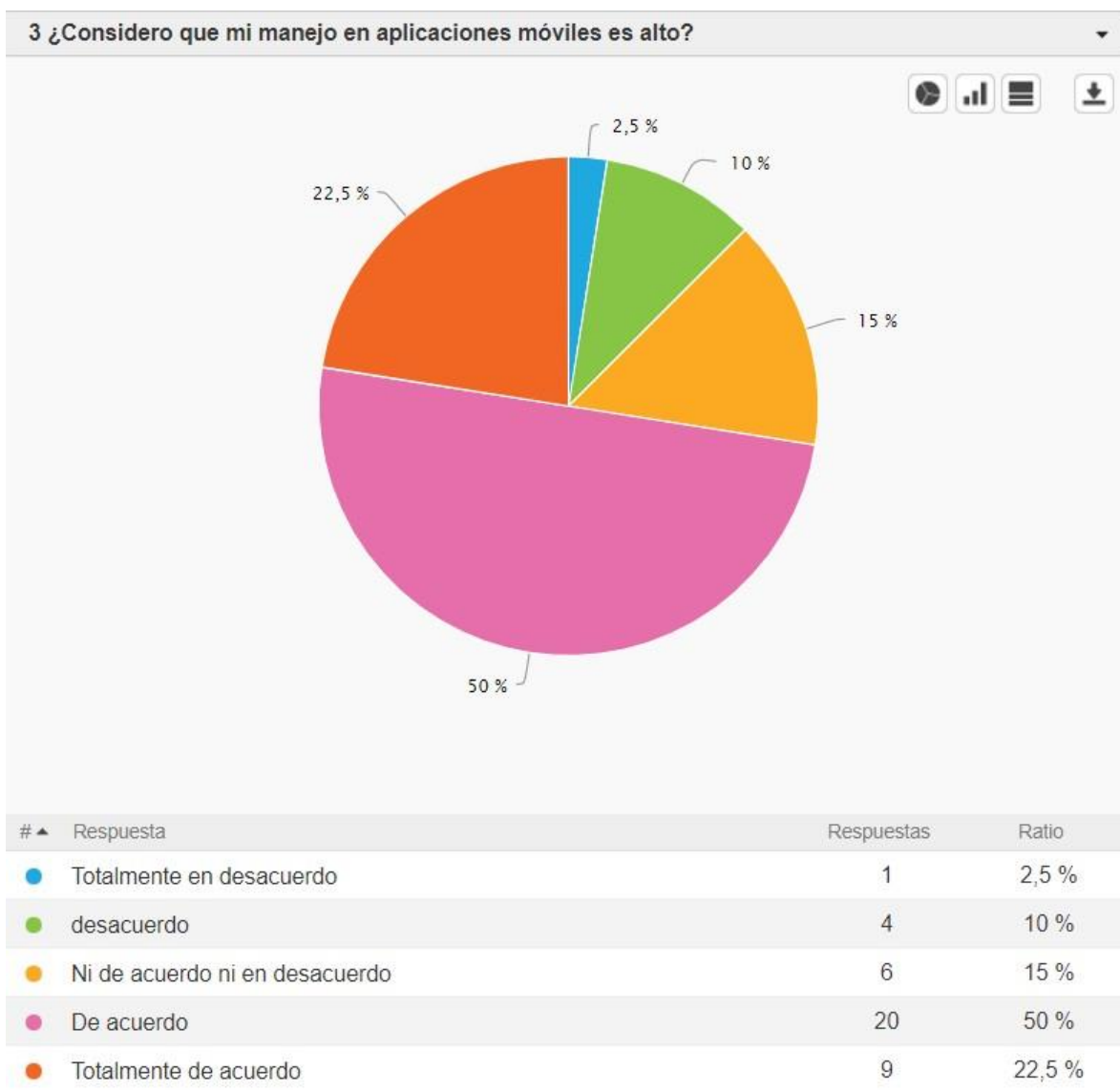


Ilustración 20: Encuesta a empaquetador, pregunta 3.

Pregunta 4: La Ilustración 21: Encuesta a empaquetador, pregunta 4. demuestra que un alto porcentaje de encuestados indica que tiene acceso a internet móvil, con lo cual, no tendrían problema para hacer uso de una aplicación que necesite internet.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

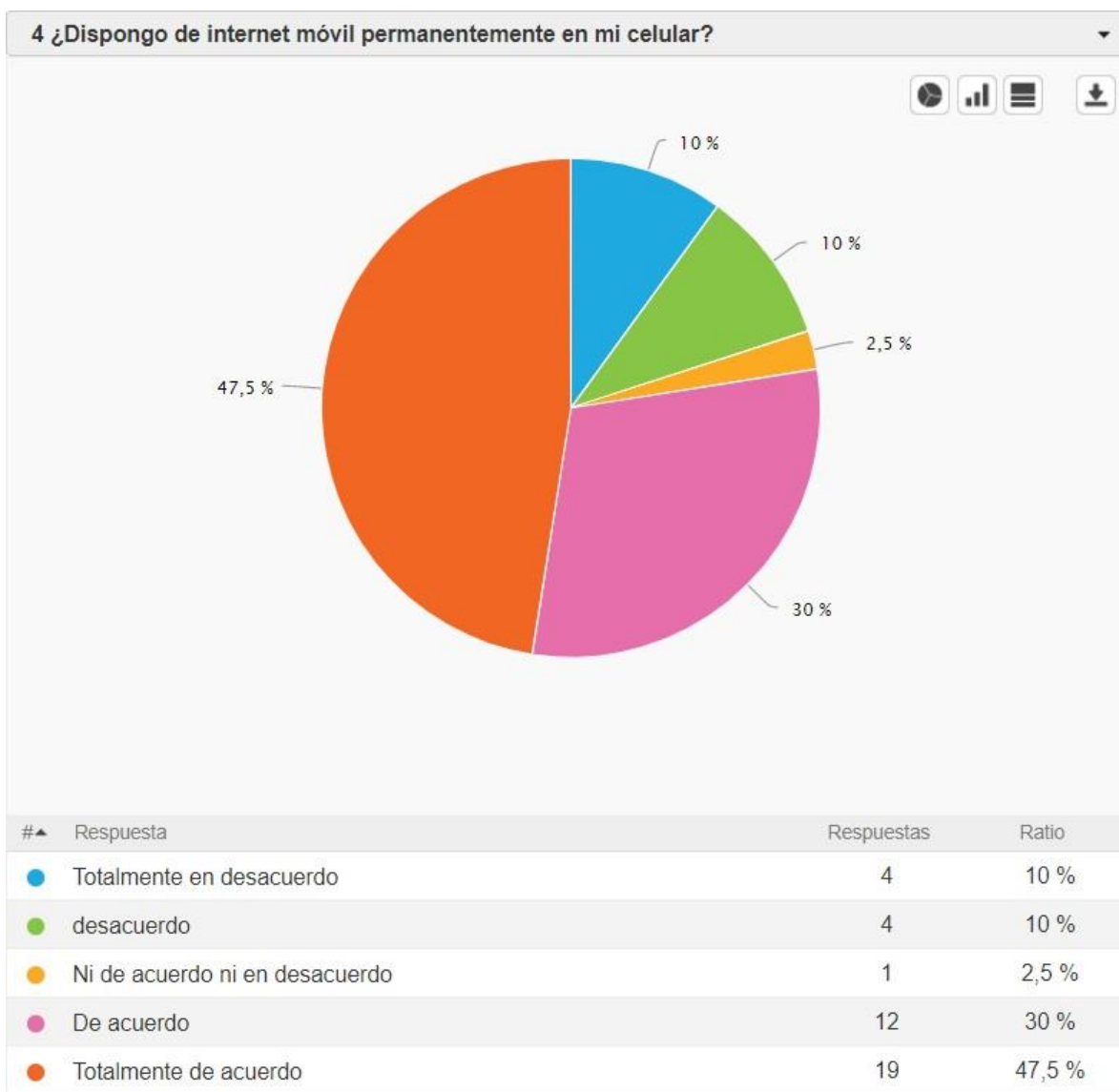


Ilustración 21: Encuesta a empaquetador, pregunta 4.

Pregunta 5: En la Ilustración 22: Encuesta empaquetador, pregunta 5. se puede observar que el 72,5% de los encuestados consideran que es apropiado actualizar o emigrar los procesos de gestión de las organizaciones de empaquetadores a nuevas tecnologías.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

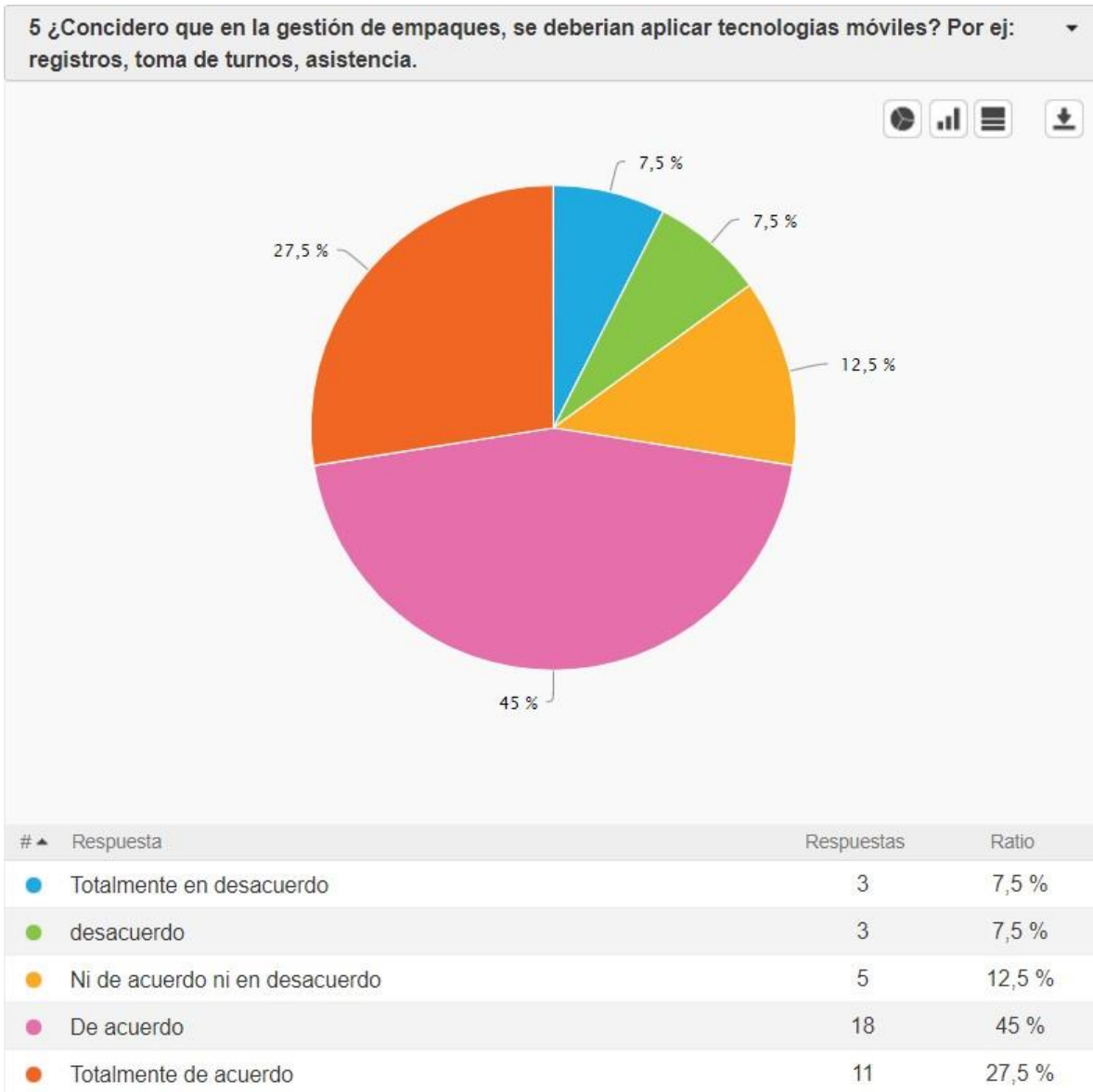


Ilustración 22: Encuesta empaquetador, pregunta 5.

Pregunta 6: la Ilustración 23: Encuesta empaquetador, pregunta 6. refleja que el 70% de los encuestados están interesados en probar una nueva modalidad para el proceso de toma de turnos, con lo cual, una aplicación que cumpla esa función es bienvenida para las organizaciones de empaquetadores.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

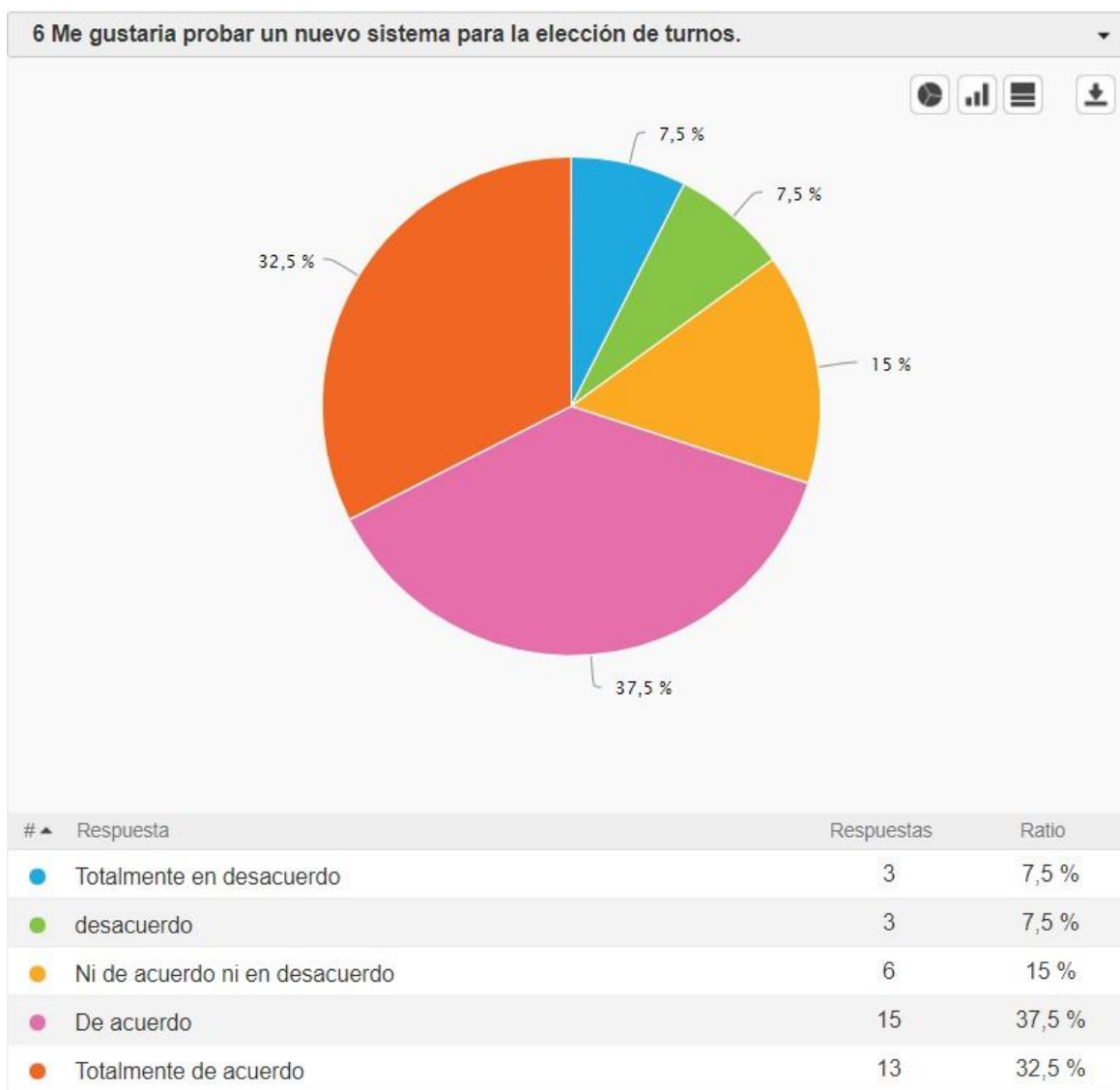


Ilustración 23: Encuesta empaquetador, pregunta 6.

Encuesta aplicada a encargados de organización Jumbo Chillan

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

Pregunta 1: La Ilustración 24: Encuesta encargados, pregunta 1. muestra que el 75% de los encargados de esta organización consideran que la gestión de empaquetadores mejoraría la forma de registro de datos haciendo uso de una aplicación que respalde dicha información.

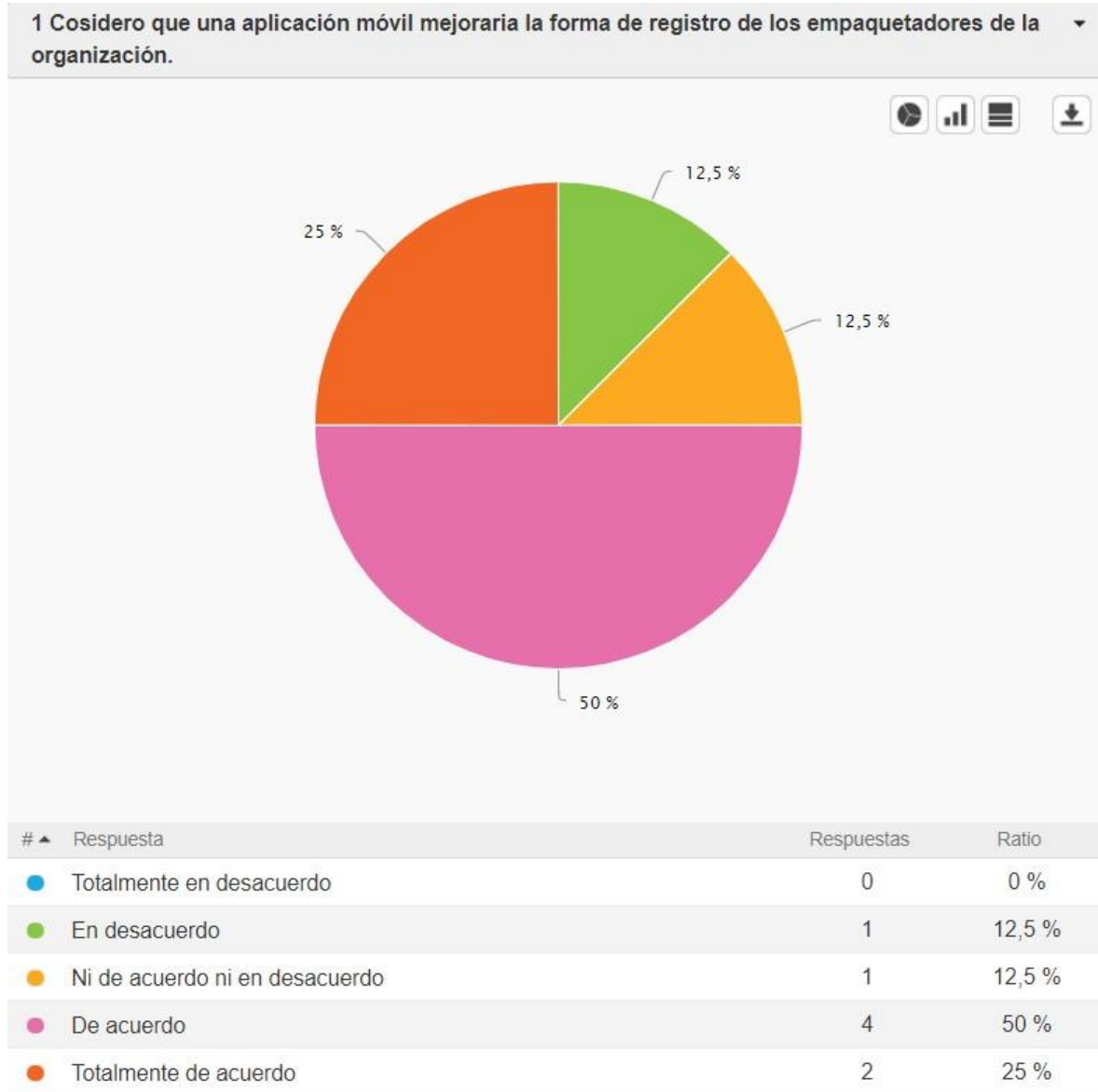


Ilustración 24: Encuesta encargados, pregunta 1.

Pregunta 2: El 100% de los encargados señala al menos que está de acuerdo con que puede hacer uso de una aplicación que permita gestionar la organización. Se puede apreciar en la Ilustración 25: Encuesta encargados, pregunta 2.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

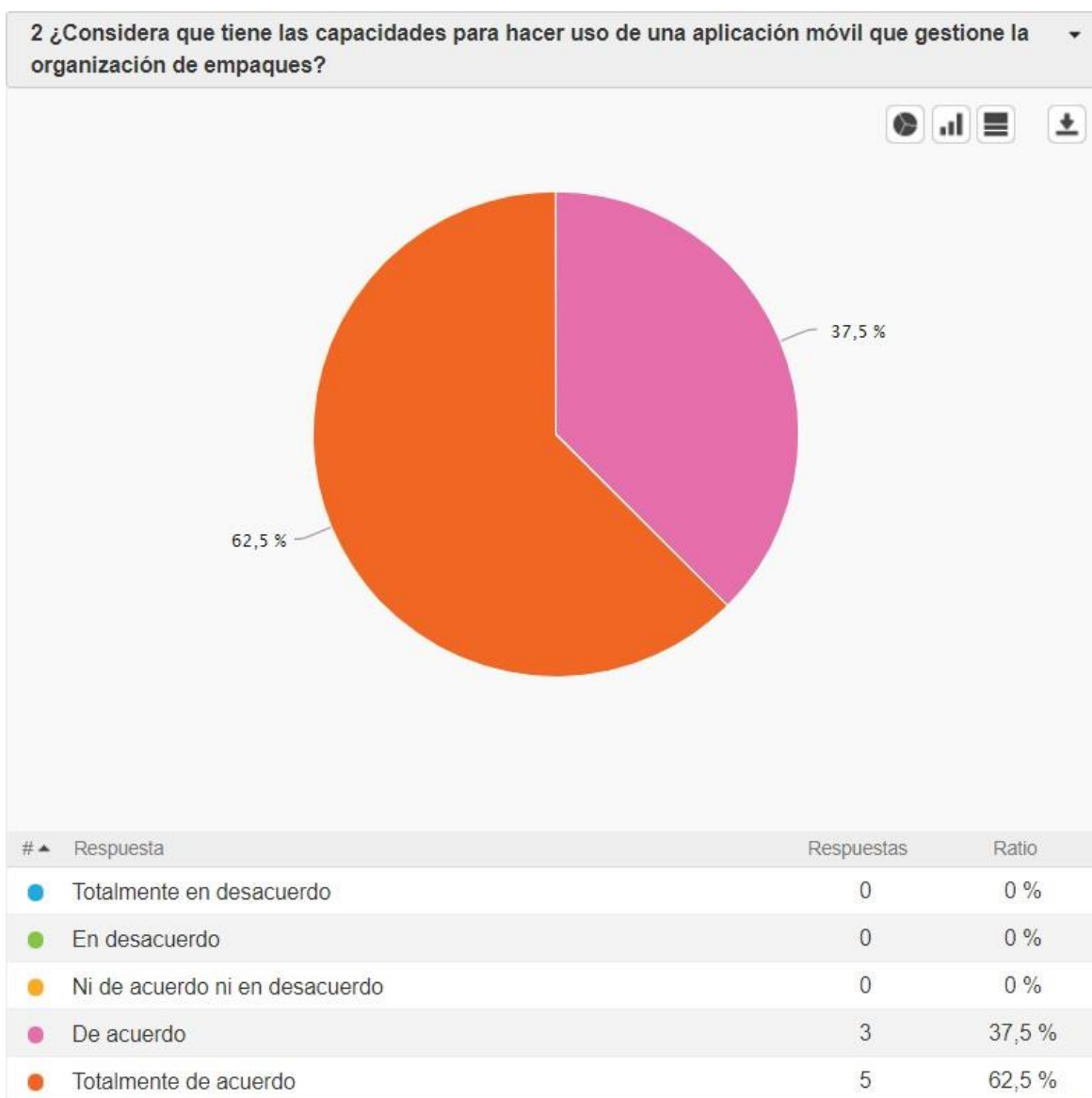


Ilustración 25: Encuesta encargados, pregunta 2.

Pregunta 3: La Ilustración 26: Encuesta encargados, pregunta 3. refleja que el 62,5% de los encargados considera que es necesaria alguna herramienta o

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

aplicación que permita el control de los empaquetadores con respecto a la asistencia.

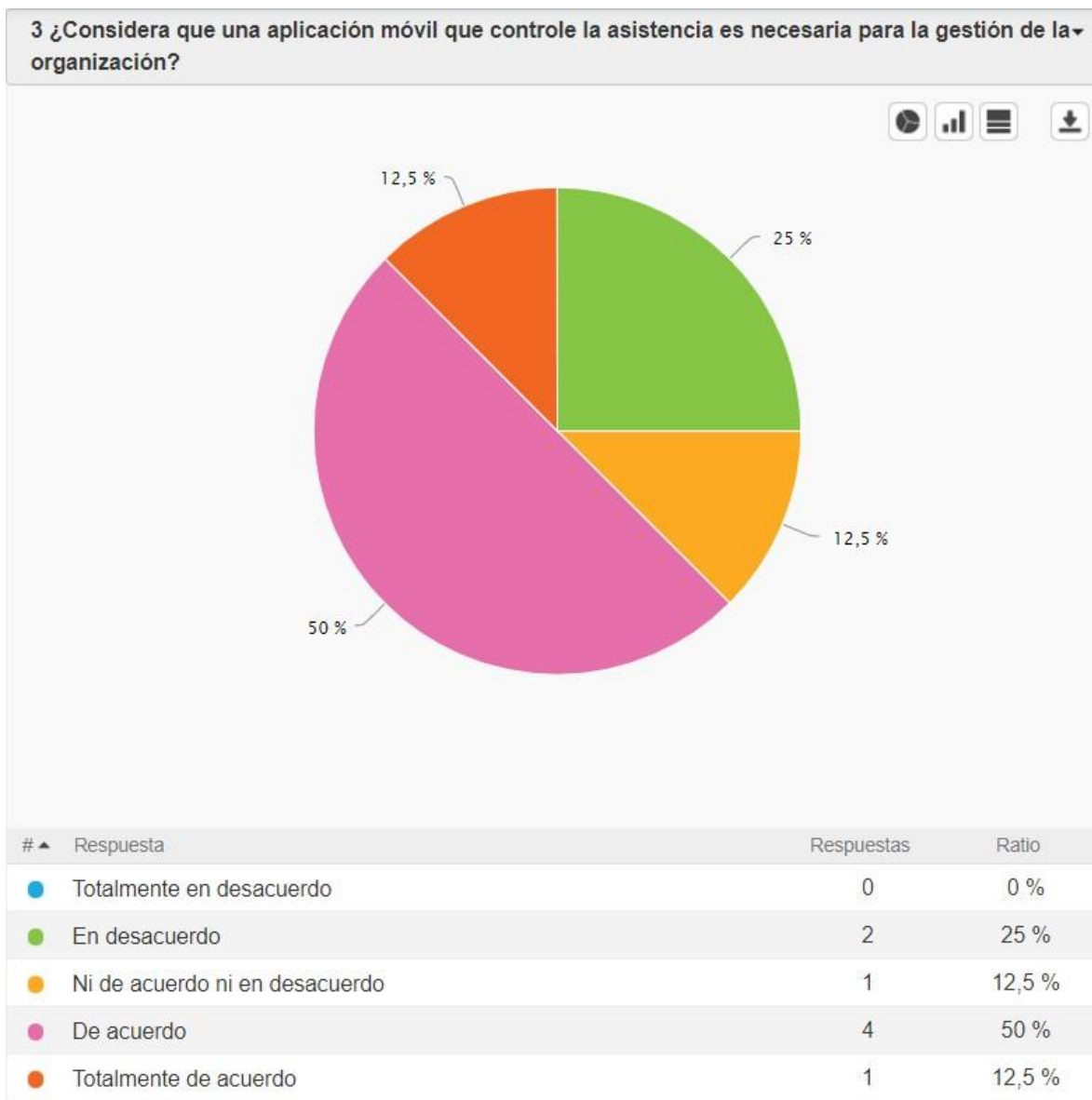


Ilustración 26: Encuesta encargados, pregunta 3.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

Pregunta 4: la Ilustración 27: Encuesta encargados, pregunta 4. muestra que el 75% de los encargados piensa que es adecuado buscar soluciones que modernicen la gestión de la organización.

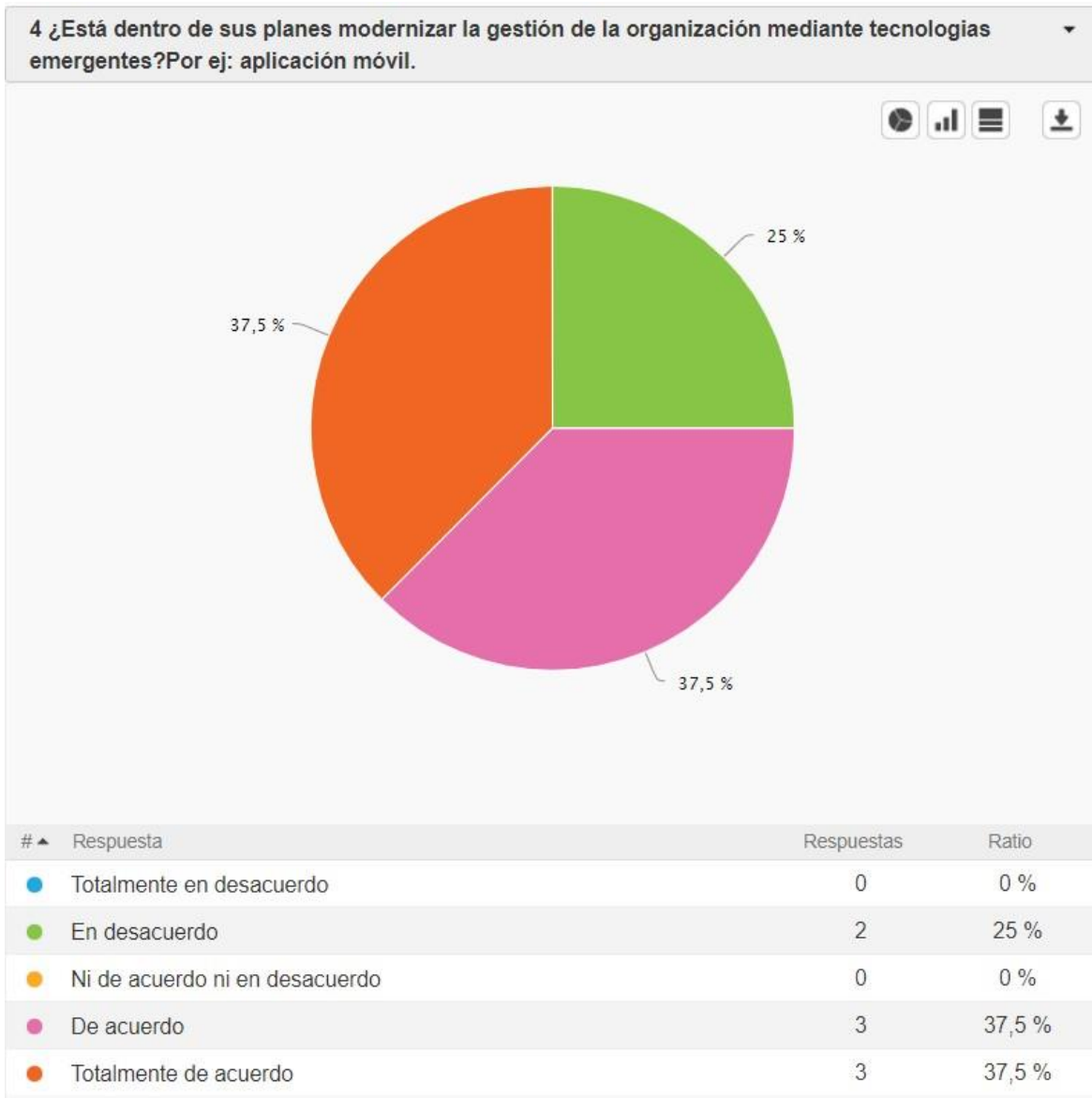


Ilustración 27: Encuesta encargados, pregunta 4.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

Pregunta 5: la Ilustración 28: Encuesta encargados, pregunta 5. muestra que el 87,5% de los encargados considera que la acción de toma de turnos, para los empaquetadores, sería mejor realizarla de manera online, es decir mediante sus SmartPhone desde cualquier lugar.

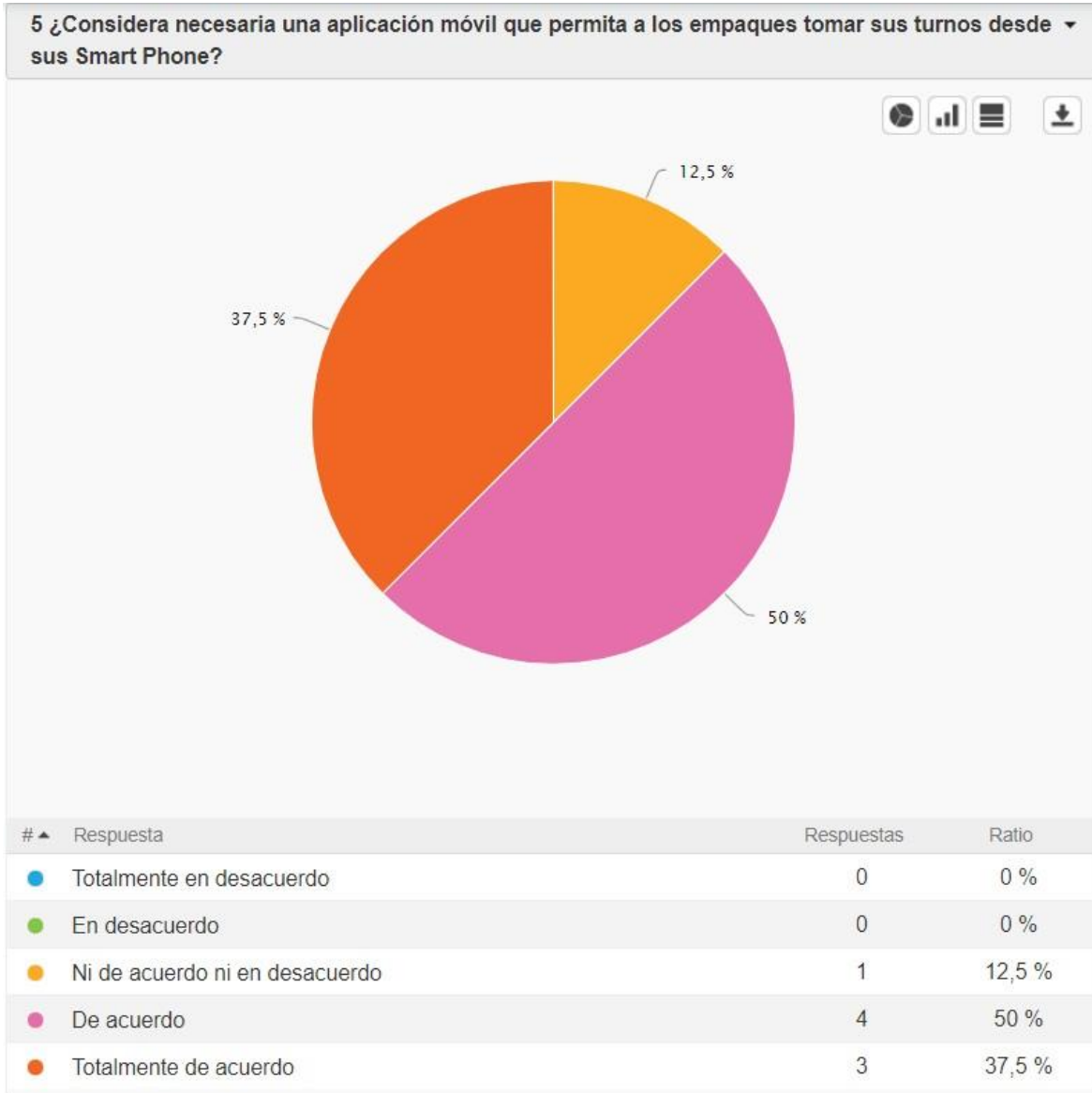


Ilustración 28: Encuesta encargados, pregunta 5.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

Anexo 2: MockUps



Ilustración 29: MockUp Login (ambos usuarios).

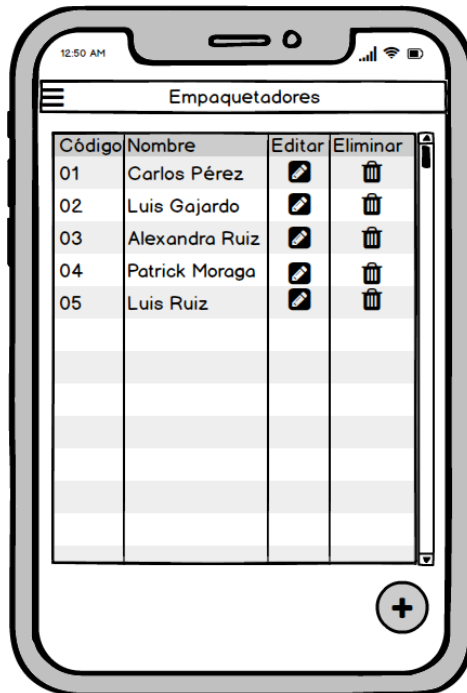


Ilustración 30: MockUp, Listado de empaques.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

The image shows a mobile application interface for adding a packager. The screen is titled "Agregar Empaquetador". It features a form with six input fields, each with a label to its left: "Nombres:", "Apellidos:", "Rut:", "Email:", "Carrera:", and "Institución:". Below the form, there are two buttons: "Guardar" (Save) and "Salir" (Exit). The status bar at the top shows the time as 12:50 AM and various system icons.

Ilustración 31: MockUp, Formulario para agregar empaquetador.

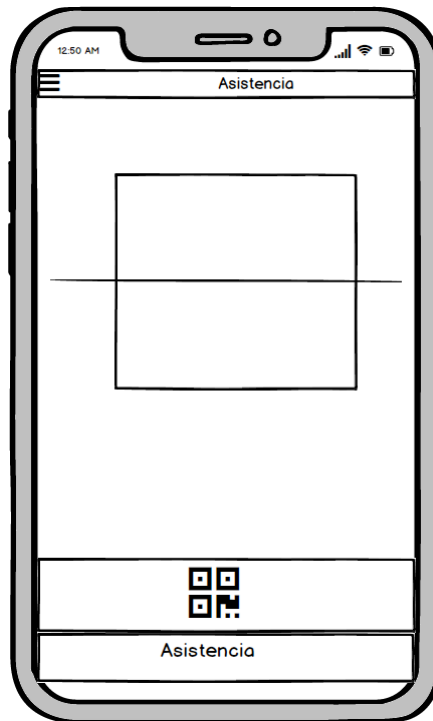


Ilustración 32: MockUp, Scanner QR para control de asistencia.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

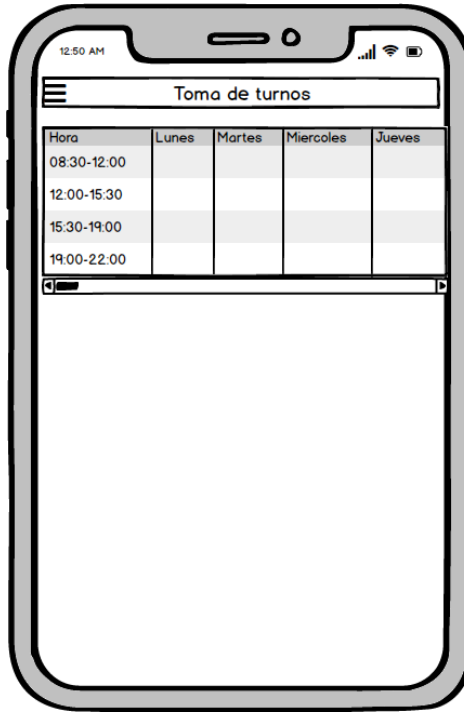


Ilustración 33: MockUp, Toma de turnos (ambos usuarios).

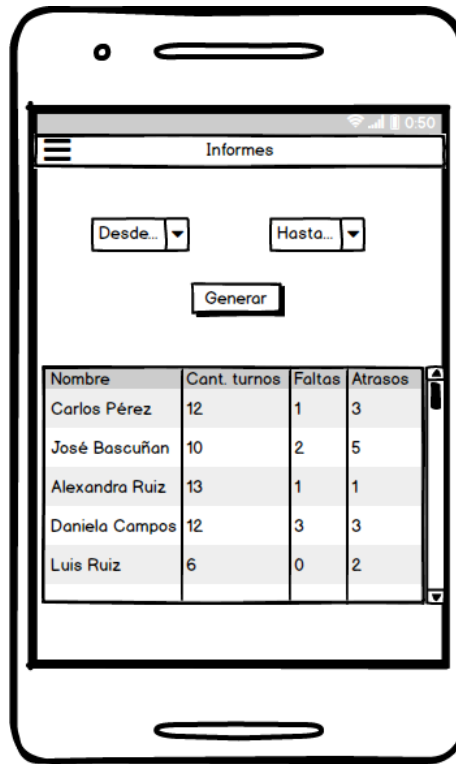


Ilustración 34: MockUp, Generar informe.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

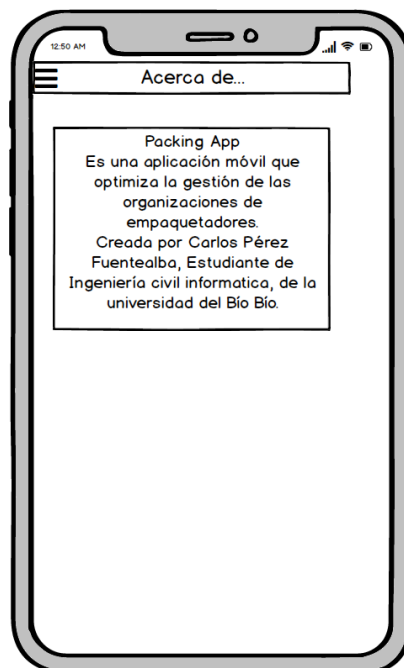


Ilustración 35: MockUp, Acerca de.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

Anexo 3: Diseño de interfaz gráfica

La Ilustración 36: Interfaz gráfica, Login, muestra el login para hacer ingreso a la aplicación, para esto, se debe tener un usuario y contraseña. El usuario es el correo que el empaque entrega a la hora de ser registrado en la organización.

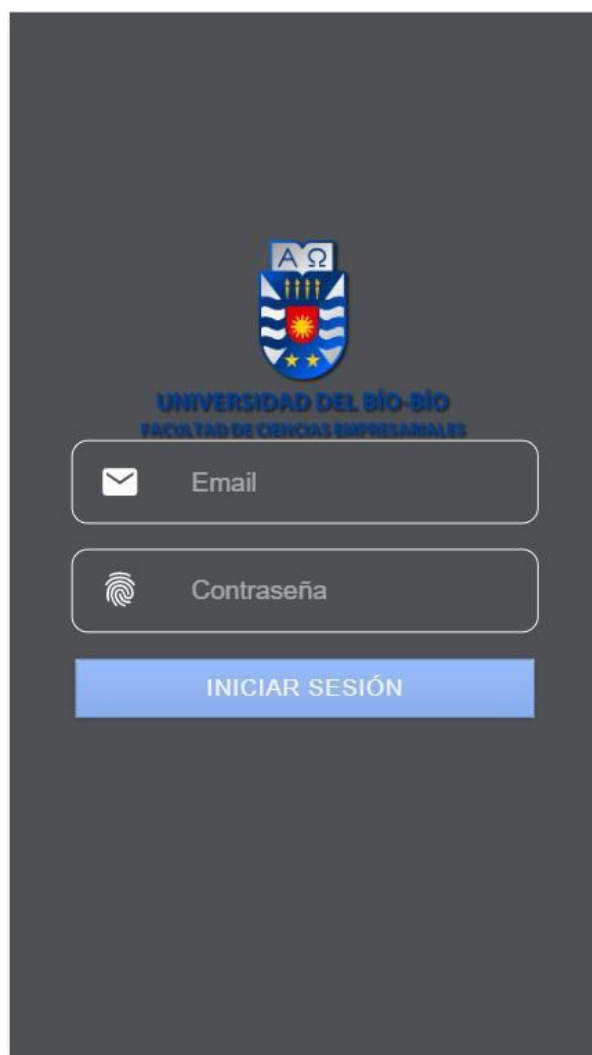


Ilustración 36: Interfaz gráfica, Login

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

La Ilustración 37: Interfaz gráfica, Home muestra el Home de la aplicación, es la pantalla inicial una vez el usuario se haya autenticado.

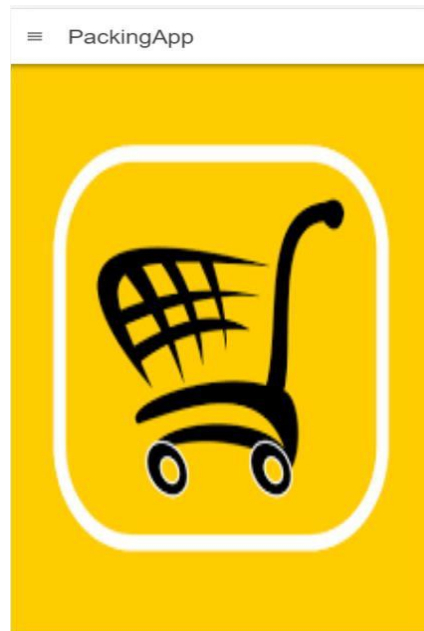


Ilustración 37: Interfaz gráfica, Home

La Ilustración 38: Interfaz gráfica, Generar listado muestra la vista de generar listado para establecer el orden de toma de turnos. Esta se publica en la vista “ver listado” para los empaquetadores.



Ilustración 38: Interfaz gráfica, Generar listado

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

La Ilustración 39: Interfaz gráfica, Ver listado muestra el listado publicado por el encargado para que el empaquetador tenga conocimiento del listado generado.

← Listado De La Semana		
N°	Código	Nombre
1	11	teresa
2	98	Veronica
3	127	Carlos
4	97	Luis
5	12	patrick
6	100	Alexandra
7	99	Jaime
8	200	Alonso

Ilustración 39: Interfaz gráfica, Ver listado

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

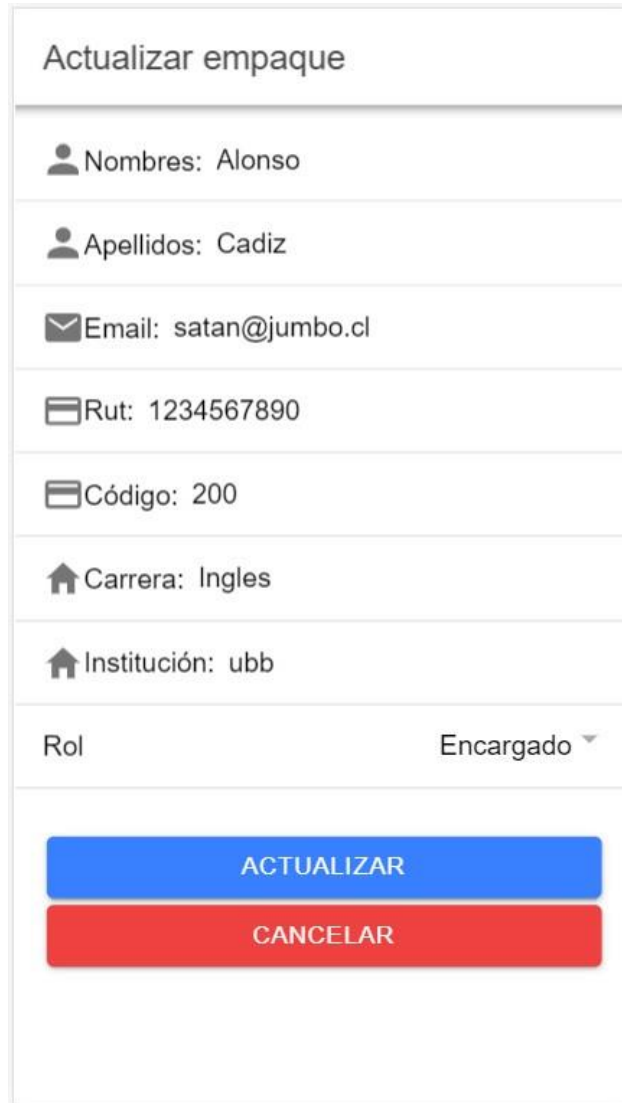
La Ilustración 40: Interfaz gráfica, Lista de empaques muestra una tabla con el registro de los empaquetadores que se encuentran en la organización, muestran datos como el código, nombre, apellido, rut, carrera y dos botones de opciones tanto para eliminar como para editar. Esta vista, además, posee un botón ubicado en la esquina inferior derecha que es para ir al formulario y agregar un empaquetador a la base de datos.

Código	Nombre	Apellido	Rut
12	patrick	Moraga	1234
127	Carlos	Pérez	1815
100	Alexandra	Ruiz	1877
99	Jaime	Rebolledo	5885
97	Luis	Ruiz	1812
11	teresa	roble	1234
98	Veronica	Fuentealba	1063
200	Alonso	Cadiz	1234

Ilustración 40: Interfaz gráfica, Lista de empaques

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

La Ilustración 41: Interfaz gráfica, Actualizar información de empaquetador. muestra el formulario de actualización de información de un determinado empaquetador o encargado. Se puede apreciar que el formato es similar al “registrar empaque” con la diferencia que al actualizar. Los campos están llenos con la información del empaque a editar. Se pueden modificar los campos deseados y posteriormente se da click en el botón “Actualizar” para guardar los cambios.



Actualizar empaque

👤 Nombres: Alonso
👤 Apellidos: Cadiz
✉ Email: satan@jumbo.cl
🇨🇱 Rut: 1234567890
🇨🇱 Código: 200
🏠 Carrera: Ingles
🏠 Institución: ubb

Rol Encargado ▾

ACTUALIZAR
CANCELAR

Ilustración 41: Interfaz gráfica, Actualizar información de empaquetador.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

La Ilustración 42: Interfaz gráfica, Menú encargado. muestra la barra de menú para el usuario “encargado” el cual posee acceso a todas las opciones de la aplicación.

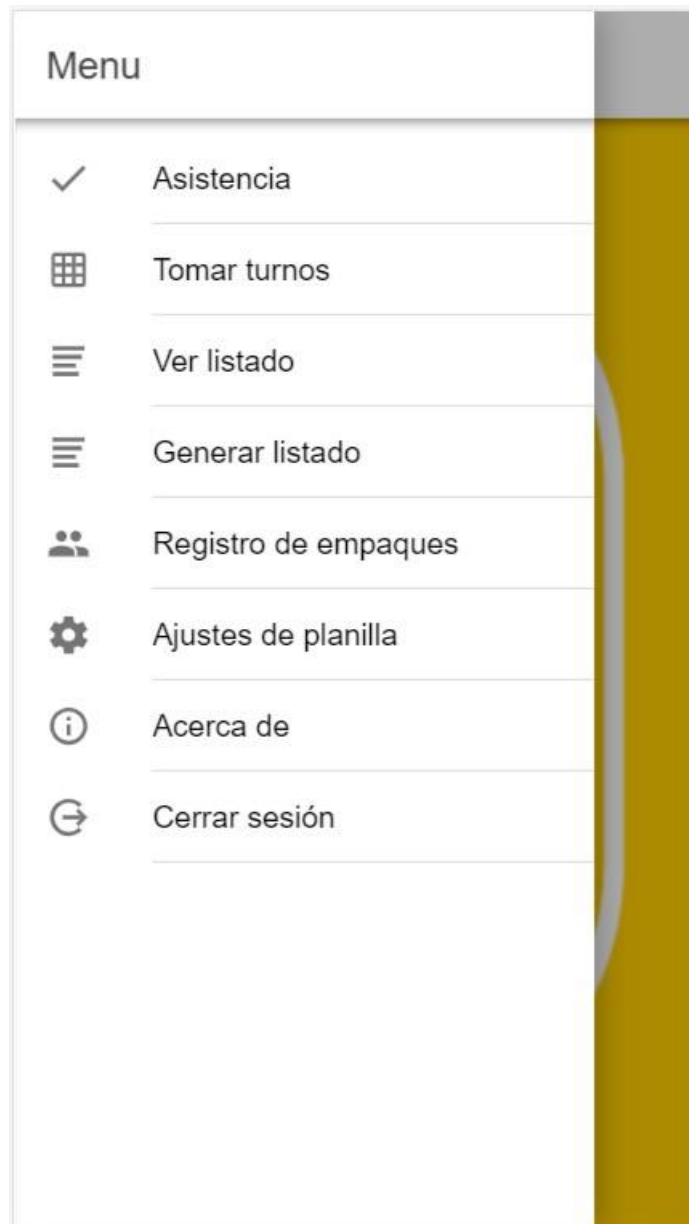


Ilustración 42: Interfaz gráfica, Menú encargado.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

La Ilustración 43: Interfaz gráfica, lector código Qr muestra la opción de capturar el código QR de un empaque para posteriormente registrarlo en la vista de “informe”. Se da click en el botón “QR escaner”, se lee el código y automáticamente se registra la información que se puede visualizar en el tab informe.

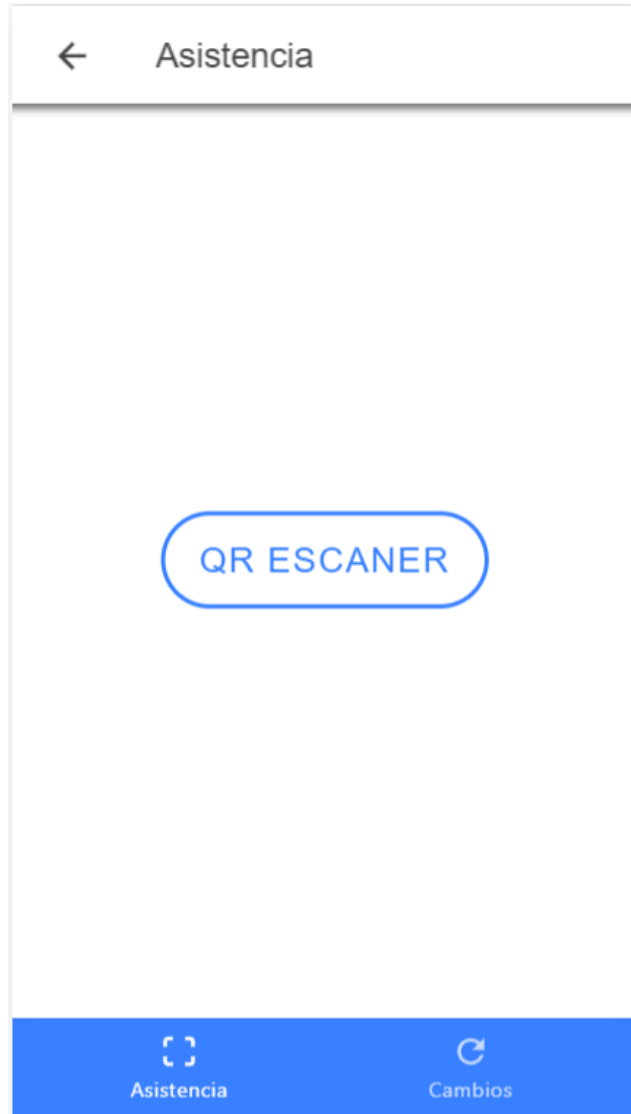


Ilustración 43: Interfaz gráfica, lector código Qr

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

La Ilustración 44: Interfaz gráfica, informe de asistencia. muestra el como se va registrando la información capturada del código QR. Esta información sirve para el control de la asistencia, como también para saber que empaques llegan atrasados a sus turnos ya que se registra la hora exacta de la lectura de código. De momento solo se registra la información, la cual queda estática, pero posteriormente se espera mejorar esta funcionalidad para entregar una acción más dinámica.

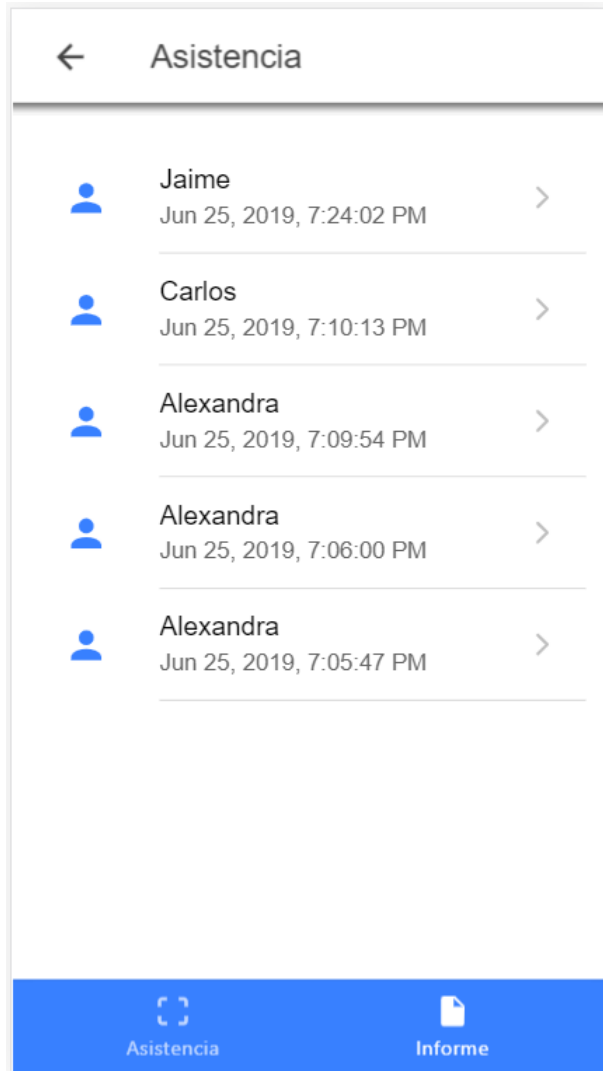


Ilustración 44: Interfaz gráfica, informe de asistencia.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

La Ilustración 45: Interfaz gráfica, toma de turno usuario encargado. muestra la toma de turnos que realiza el usuario Encargado, el cual tiene la misma forma que el usuario Empaque, con la diferencia de que el primero tiene la visibilidad de un botón “limpiar” la cual le permite borrar todos los turnos que se hayan tomado. Esta acción es de utilidad al momento de iniciar otra toma de turnos, ya que se necesita que los cupos estén libres para la próxima toma de turnos.

Hora	Lunes	Martes	Miércoles	Jue
8:30	0/ 1 +	0/ 1	0/ 1	0/ 2
9:30	0/ 1	2/ 2 100 100	0/	0/
10:30	0/ 2	1/ 10 127	0/ 5	0/
12:00	0/ 4	0/ 4	0/	0/
13:30	0/ 2	0/	0/	0/
15:30	0/ 5	0/	0/	0/
17:00	0/ 5	1/ 8 100	0/	0/

Ilustración 45: Interfaz gráfica, toma de turno usuario encargado.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

La Ilustración 46: Interfaz gráfica, Acerca de muestra la interfaz de “Acerca de”, la cual entrega información sobre la aplicación, por ejemplo, el nombre del desarrollador, el motivo por el cual se desarrolló el proyecto y el Email de contacto.



Ilustración 46: Interfaz gráfica, Acerca de

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

Anexo 4: Tamaño en Bytes por usuario registrado

El registrar usuarios en la base de datos y manejar un gran volumen de ellos es una tarea simple para un servidor potente como Firebase, por ende, se desea saber cuánto peso tiene registrar un usuario en la base de datos para así calcular un estimado de peso total almacenado en la base de datos.

A continuación, se especifican los atributos de un usuario registrado en la base de datos:

- Nombre: string = 20 Bytes.
- Apellidos: string = 20 Bytes.
- Rut: string = 10 Bytes.
- Código: int = 4 Bytes.
- Id: string = 20 Bytes.
- Uid: string = 20 Bytes.
- Email: string = 30 Bytes.
- RutEmpresa: string = 6 Bytes.
- Carrera: string = 20 Bytes.
- Institución: string = 20 Bytes.
- Rol: string = 10 Bytes.

Esto nos arroja un total de **160 Bytes** por usuario registrado.

La Ilustración 47: Firebase, Campos (atributos) de usuarios muestra la base de datos con los campos (atributos) que tiene cada registro de usuario.

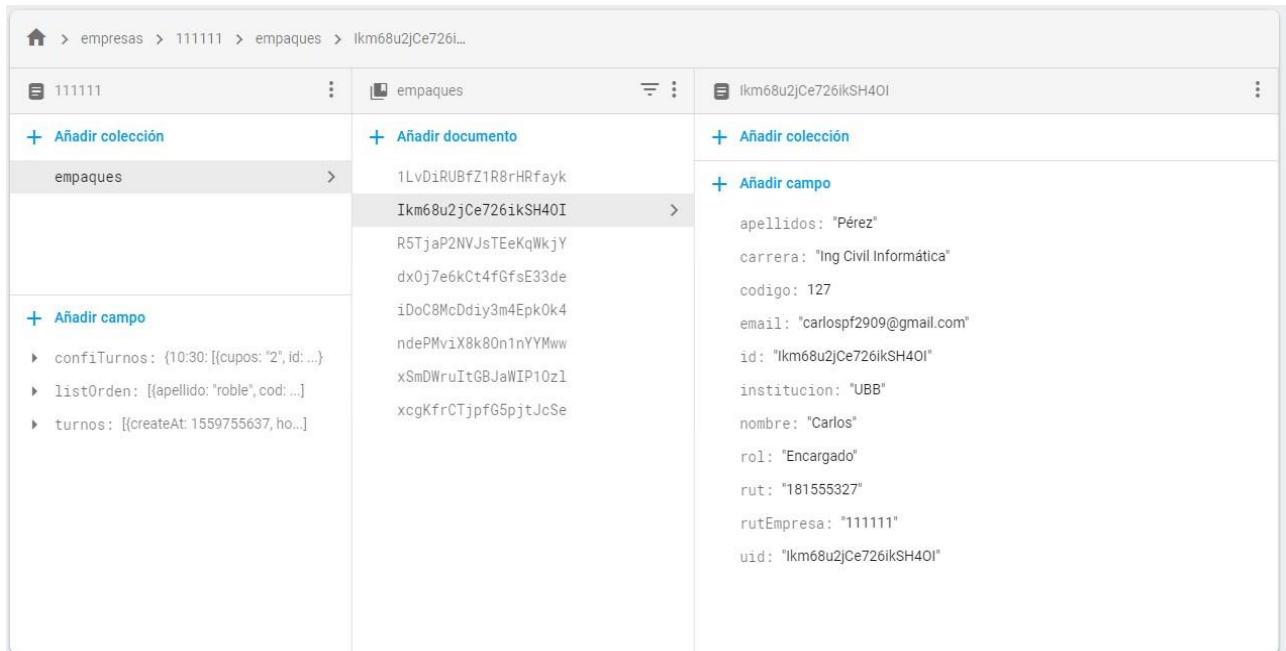


Ilustración 47: Firebase, Campos (atributos) de usuarios

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

Anexo 5: Pruebas de usuario, Jumbo Chillán

A continuación, se muestra las bitácoras que se les otorgo a 5 usuarios (3 encargados, 2 empaques), para que hicieran uso de la aplicación y llevaran a cabo una serie de acciones que involucran las funcionalidades de la aplicación.

Para encargados:

1)

N°		Actividad
1	✓	Ingresar al menú de inicio de la aplicación.
2	✓	Escanear un Código QR.
3	✓	Generar un listado para ordenar la toma de turnos y publicarlo.
4	✓	Ver registro de empaques.
5	✓	Agregar un empaque al sistema.
6	-	Ingresar cantidad de cupos a un determinado turno.
7	✓	Tomar turnos.
8	✓	Información de contacto del desarrollador.
9	✓	Visualizar turnos tomados.

Observaciones:

Definiciones de los términos del menú ya que son parecidos.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

2)

N°		Actividad
1	✓	Ingresar al menú de inicio de la aplicación.
2	-	Escanear un Código QR.
3	✓	Generar un listado para ordenar la toma de turnos y publicarlo.
4	✓	Ver registro de empaques.
5	✓	Agregar un empaque al sistema.
6	-	Ingresar cantidad de cupos a un determinado turno.
7	✓	Tomar turnos.
8	✓	Información de contacto del desarrollador.
9	✓	Visualizar turnos tomados.

Observaciones:

Bastante simple de usar, pero el lector de qr es una funcionalidad que no sabía para que se podía usar, por eso me costó saber dónde estaba. Se podría agregar más colores para mejorar la fachada. En general bien.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

3)

N°		Actividad
1	✓	Ingresar al menú de inicio de la aplicación.
2	✓	Escanear un Código QR.
3	-	Generar un listado para ordenar la toma de turnos y publicarlo.
4	✓	Ver registro de empaques.
5	✓	Agregar un empaque al sistema.
6	✓	Ingresar cantidad de cupos a un determinado turno.
7	✓	Tomar turnos.
8	✓	Información de contacto del desarrollador.
9	-	Visualizar turnos tomados.

Observaciones:

Se ve bastante bien, el nombre de las opciones podría mejorar un poco (para el listado). Un tutorial al iniciar la aplicación sería ideal.
Las opciones que tiene la app están justas y necesarias para lo que se necesita de manera básica.

Para empaques:

1)

N°		Actividad
1	✓	Ingresar al menú de inicio de la aplicación.
2	✓	Tomar turnos.
3	✓	Información de contacto del desarrollador.
4	✓	Ver listado de orden para toma de turnos.

Observaciones:

Es muy simple de usa y se ve bien la toma de turnos, cumple con lo que necesita un empaque. La información de quien hizo la app igual es bueno que este ahí.

Aplicación móvil para gestión de empaquetadores en comercios de grandes superficies

2)

N°		Actividad
1	✓	Ingresar al menú de inicio de la aplicación.
2	✓	Tomar turnos.
3	✓	Información de contacto del desarrollador.
4	✓	Ver listado de orden para toma de turnos.

Observaciones:

Se podría agregar más colores en la toma de turnos, por ejemplo, en los turnos que ya estén llenos. Tiene pocas opciones, así que es fácil saber dónde están las cosas. Como empaque considero que es una app útil y que cumple la función para tomar turnos.