

Instituto Tecnológico de Costa Rica

Escuela de Ingeniería en Computación

**Informe final - “Go-Labs App: Una aplicación móvil para la administración e interacción con procesos internos de la empresa”**

Para optar por el título de Ingeniero en Computación con el grado académico de Bachiller

Gabriel Rojas Alfaro

San Carlos, Noviembre 2017

## Resumen ejecutivo

En el presente documento se analiza el estado final del proyecto de práctica profesional, realizado en la empresa Go-Labs. En esta, se da el problema de que se realizan procesos administrativos manualmente con una carga de tiempo considerable, por lo que se propone una solución para automatizar estos procesos y agilizar las tareas, por medio de una aplicación móvil: Go-Labs App. En este informe se pueden encontrar las secciones necesarias para detallar el proceso de desarrollo de la solución, aplicando los conocimientos adquiridos a lo largo de la carrera, mostrando avance y evolución conforme se obtiene retroalimentación de los productos de cada una de las fases de desarrollo.

Como se menciona, la solución será detallada por medio de diagramas, interfaces y descripciones generales, además de estar respaldada por el estado del arte, de forma que refleje lo que se trabajó durante la estadía en la empresa.

Por último, se obtienen las conclusiones y recomendaciones del proceso de desarrollo de la solución, describiendo generalmente el estado de la solución con respecto a los objetivos propuestos, los productos entregados y las tecnologías implementadas, cumpliendo con el objetivo de dar retroalimentación con respecto a la experiencia obtenida a lo largo del proceso de práctica profesional.

## Tabla de contenidos

1	Introducción	1
2	Contexto del proyecto	2
2.1	Datos de la empresa	2
2.2	Antecedentes del proyecto	3
3	Descripción del problema	4
3.1	Problema	4
3.2	Solución	4
3.3	Stakeholders	5
3.4	Necesidades y expectativas	6
3.5	Perspectiva, supuestos y dependencias	7
3.6	Requerimientos no funcionales	8
3.7	Características generales	8
4	Estado del arte	9
4.1	Aplicaciones híbridas	9
4.2	Ionic Framework	11
5	Análisis de riesgos	12
6	Objetivos y alcances del sistema	13
6.1	Objetivo general	13
6.2	Objetivos específicos	13
6.3	Alcances	13
7	Modelo de diseño	14
7.1	Arquitectura conceptual de la solución	14
7.2	Los modelos de subsistemas	16
7.3	Interfaces de usuario	17
7.4	Componentes y servicios	24
7.5	Diseño de base de datos	24
8	Plan de trabajo	25
9	Conclusiones y recomendaciones	28
10	Referencias	31

## Índice de figuras

Figura 2.1: Organigrama de la empresa	3
Figura 7.1: Diagrama de la arquitectura conceptual de la solución	15
Figura 7.2: Diagrama del modelo de subsistemas de la solución	16
Figura 7.3: Página de ingreso	18
Figura 7.4: Menú de navegación	18
Figura 7.5: Registro de productos	19
Figura 7.6: Historial de consumo	19
Figura 7.7: Administración de productos	20
Figura 7.8: Cierre de periodo	20
Figura 7.9: Visualización de solicitudes y días disponibles	21
Figura 7.10: Solicitud de periodo de vacaciones	21
Figura 7.11: Administración de solicitudes	22
Figura 7.12: Administración de días de vacaciones	22
Figura 7.13: Visualización de información de la empresa	23

## Índice de tablas

Tabla 3.1: Descripción de stakeholder N° 1	5
Tabla 3.2: Descripción de stakeholder N° 2	5
Tabla 3.3: Definición de stakeholder N° 3	5
Tabla 3.4: Definición de necesidad N° 1	6
Tabla 3.5: Definición de necesidad N° 2	6
Tabla 3.6: Descripción de la necesidad N° 3	7
Tabla 5.1: Descripción de riesgo N° 1	12
Tabla 5.2: Descripción de riesgo N° 2	12
Tabla 7.1: Descripción de la base de datos	25

# 1 Introducción

La empresa Go-Labs pretende automatizar varios procesos internos, entre los cuales se encuentran la administración de la pulpería interna, el control de periodos de vacaciones y la difusión de información acerca de la empresa. Para esto se desarrollará un producto software constituido por una aplicación web para funciones de administración de los procesos e información, y una aplicación móvil para consumo e interacción de ellos.

Cómo antecedente, estos procesos se llevan a cabo realizando procedimientos rudimentarios como cálculos sobre apuntes en hojas de papel, o correos electrónicos con peticiones sobre vacaciones. La expectativa de la plataforma consiste en la agilización de estos procesos por medio de su automatización, convirtiéndola así en la plataforma por defecto de la empresa, donde se lleven a cabo la mayor parte de sus procesos internos y sea el primer recurso de sus empleados.

En este documento, se podrá profundizar en la descripción del problema propuesto anteriormente. Para ello, se detalla además quienes serán los involucrados en el proyecto, describiendo sus objetivos y responsabilidades para que este sea exitoso. Además, se hace referencia a la perspectiva, supuestos y dependencias obtenidas, dando una descripción general de la solución.

Un aspecto importante es el respaldo de la solución propuesta, con respecto a las tecnologías y metodologías implementadas. En el estado del arte se podrán encontrar referencias a artículos en los cuales se indaga sobre las tecnologías propuestas, destacando eficazmente aspectos influyentes en la elección de estas.

Así mismo, se define el alcance del proyecto a partir de los objetivos propuestos, especificando también los riesgos que este supone. Estos últimos consisten generalmente en aspectos de conexión a internet y base de datos, pues son aspectos de los que la aplicación es altamente dependiente.

Por último, se puede encontrar aspectos específicos de la solución, como diagramas conceptuales, modelos de subsistemas, interfaces y diseños de la aplicación, así como el plan de trabajo propuesto para dar seguimiento al proceso de desarrollo.

## **2 Contexto del proyecto**

### **2.1 Datos de la empresa**

Go-Labs es una empresa que se dedica al Desarrollo de Software en la Zona Norte del país, la cual se caracteriza por brindar servicios tanto a clientes en Costa Rica como en el extranjero. Go-Labs exporta soluciones de software a clientes y aliados estratégicos en países como Estados Unidos, Canadá, España, Ecuador y otros.

El área de producción es la encargada de desarrollar los distintos programas (software) que son demandados por sus clientes, la cual se puede apreciar en la jerarquía mostrada en la Figura 2.1: Organigrama de la empresa. Esta área cuenta con un Gerente de Innovación, el cual busca involucrar a la empresa con potenciales colaboradores para Go-Labs. Además, esta área también incluye al Gerente de Producción, el cual tiene a su cargo a los Project Manager, los Architect y al equipo de producción (Junior Developer, Software Engineer I, II y III, y Senior Software Engineer).

El Project Manager es el encargado de administrar y gestionar a cada equipo de los proyectos que se encuentran en ejecución. Se encarga de liderar proyectos en progreso y también interactúa con el cliente. El Architect es el encargado de crear la arquitectura de los sistemas, la infraestructura que soportará el sistema que solicita el cliente, y hacerlo aún mejor que las expectativas que se tengan.

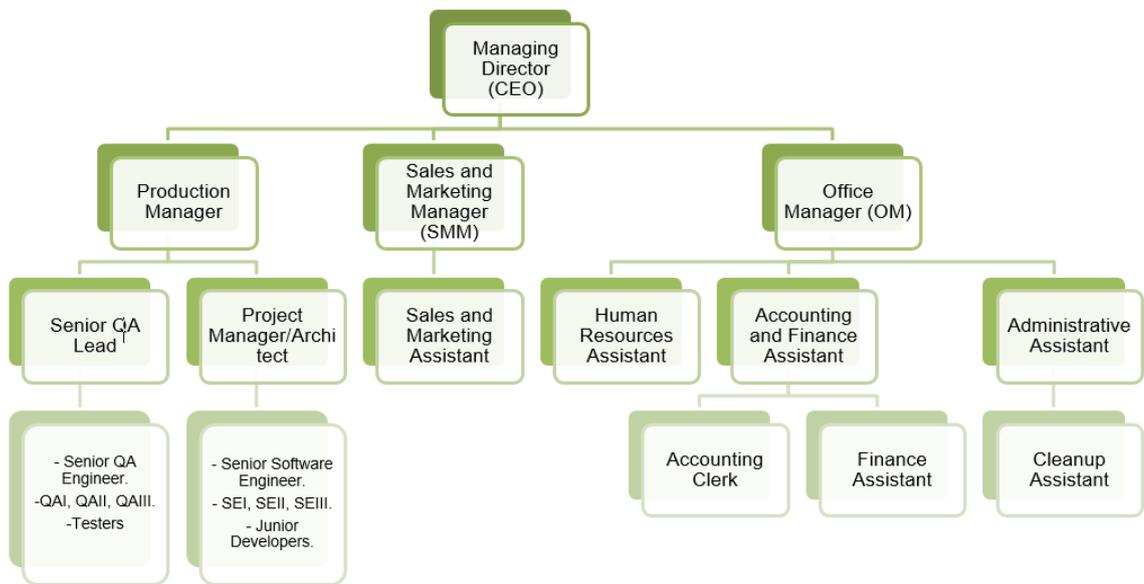


Figura 2.1: Organigrama de la empresa

## 2.2 Antecedentes del proyecto

Actualmente, en la empresa se encuentra un servicio de pulpería interna, donde el personal podrá comprar productos como refrescos, galleta y snacks. Estos deberán apuntar en una hoja el consumo semanal, para que luego este se rebaje de su salario. El o la encargada de recursos humanos deberá calcular el consumo de cada colaborador para así realizar las debidas deducciones. Este proceso requiere de esfuerzo y tiempo extra, pues el encargado deberá contar los gastos de los empleados uno por uno y realizar las sumas pertinentes.

La aplicación pretende agilizar el proceso tanto para los empleados como para el encargado de recursos humanos, pues contará con una aplicación móvil donde los colaboradores podrán agregar las compras que han hecho mientras verifican la lista de precios y su respectivo acumulado, así como su respectiva sección de administrador donde se realizarán los cálculos automáticamente y mostrará los datos necesarios para ejecutar las deducciones.

Se pretende además agilizar otros procesos como la solicitud de vacaciones, donde actualmente se debe enviar un correo electrónico con la solicitud y esperar la respuesta, y difusión de información de la empresa, donde datos como los valores, la misión e información se transmiten por medio de carteles y panfletos.

## **3 Descripción del problema**

### **3.1 Problema**

Algunos de los procesos administrativos de la compañía como la gestión de vacaciones, la difusión de información de la empresa y el manejo de la pulpería interna representan una carga de tiempo considerable en el personal de recursos humanos.

La inversión de tiempo en estos procesos reduce las posibilidades del personal de recursos humanos para atender otras labores estratégicas que pueden impactar muy positivamente en la organización.

### **3.2 Solución**

La solución debe proporcionar al personal interesado una plataforma rápida, eficiente y de fácil acceso que les permita gestionar los procesos internos de la empresa. De esta forma, estas operaciones requerirán menos carga a los empleados al reducir el tiempo de ejecución y de espera entre cálculos y respuestas, además del ahorro de recursos como el papel.

Dentro de estos procesos se encuentra la administración de la pulpería interna, donde se incluirán tareas como el registro de productos consumidos y el cálculo del total a rebajar, además de la administración de los días de vacaciones, donde se podrá solicitar el periodo específico que se desea tomar así como el rechazo o la aprobación de estos, y la administración de la información de la empresa, donde se mostrarán y editarán datos generales como valores, misión y visión.

### 3.3 Stakeholders

Tabla 3.1: Descripción de stakeholder N° 1

<b>Nombre</b>	Gabriel Rojas
<b>Rol</b>	Junior Developer / Practicante
<b>Labores</b>	Desarrollo de proyectos
<b>Responsabilidades</b>	Desarrollo del proyecto asignado en base a los requerimientos definidos.
<b>Objetivos</b>	Aplicar el conocimiento adquirido a lo largo de la carrera con el fin de obtener un producto de calidad.

Tabla 3.2: Descripción de stakeholder N° 2

<b>Nombre</b>	Ester Vargas
<b>Rol</b>	Aseguramiento de la calidad
<b>Labores</b>	Preparar planes de prueba Ejecutar planes de prueba
<b>Responsabilidades</b>	Reporte de fallos y defectos del sistema.
<b>Objetivos</b>	Realizar las pruebas de calidad a lo largo del desarrollo del proyecto.

Tabla 3.3: Definición de stakeholder N° 3

<b>Nombre</b>	Carlos Rojas
<b>Rol</b>	Architect
<b>Labores</b>	Asignación de tareas Supervisión de operaciones
<b>Responsabilidades</b>	Dar seguimiento al desarrollo del proyecto
<b>Objetivos</b>	Aportar ideas respecto al uso correcto de las tecnologías aplicadas

### 3.4 Necesidades y expectativas

Tabla 3.4: Definición de necesidad N° 1

<b>Necesidad</b>	Administración de la pulpería interna
<b>Prioridad</b>	Alta
<b>Problema</b>	Se requiere llevar control de los productos consumidos por cada empleado, de forma que se pueda realizar la debida reducción al salario en cada periodo de pago.
<b>Solución actual</b>	Anotación en hojas de papel, realizando marcas en una tabla correspondiente al producto consumido. Se calcula la reducción contando marcas en relación a la lista de precios.
<b>Solución propuesta</b>	Se habilitará el módulo de una aplicación móvil que permita a los empleados escanear o añadir los productos consumidos a su cuenta, realizando automáticamente el cálculo del consumo total por empleado.

Tabla 3.5: Definición de necesidad N° 2

<b>Necesidad</b>	Administración de días de vacaciones
<b>Prioridad</b>	Media
<b>Problema</b>	Se requiere administrar las solicitudes de vacaciones de los empleados y la comunicación del estado del proceso de cada una.
<b>Solución actual</b>	Los empleados solicitan su periodo de vacaciones por medio de correos electrónicos, donde los administradores deberán responder a estos con su respectiva decisión.
<b>Solución propuesta</b>	Se desarrollará un módulo para una aplicación móvil que permita a los empleados visualizar y solicitar sus periodos de vacaciones, y a los administradores aprobar o rechazarlas, dando retroalimentación inmediata a ambos.

Tabla 3.6: Descripción de la necesidad N° 3

<b>Necesidad</b>	Control de información general de la empresa
<b>Prioridad</b>	Baja
<b>Problema</b>	Se desea promover y reforzar los valores, la misión y la visión de la empresa, manteniéndolos siempre al alcance en el momento que estos los quieran consultar.
<b>Solución actual</b>	Se colocan carteles en las paredes del establecimiento con los valores, misión y visión de la empresa impresos en ellos.
<b>Solución propuesta</b>	Se diseñará un módulo para una aplicación móvil donde se puedan visualizar los diferentes datos de la empresa, además de su parte administrativa donde se controle el texto a mostrar.

### 3.5 Perspectiva, supuestos y dependencias

Del producto se espera que sea una aplicación emblema para la empresa, donde se reúnan los procesos más importantes y sea de uso cotidiano para los integrantes de ella. Se espera que su uso agilice estos procesos de manera que se perciba un ahorro de tiempo y recursos.

#### Supuestos

- Los usuarios utilizarán dispositivos con los sistemas operativos Android y iOS.
- Los dispositivos contarán con una cámara que les permita escanear códigos de barras.
- La empresa proveerá licencias y planes de las herramientas necesarias para el desarrollo y mantenimiento del producto.

#### Dependencias

- Alta dependencia a la conexión a internet.
- Alta dependencia a la estabilidad del servidor de bases de datos de Firebase.

### 3.6 Requerimientos no funcionales

- El sistema debe ser escalable, con el objetivo de poder expandir su alcance con respecto a futuros procesos que se puedan incluir.
- El sistema debe ser modular, de modo que los procesos se encuentren separados entre sí, y la modificación de uno no afecte a los demás.
- El sistema debe ser eficiente, aprovechando los recursos de manera que se perciba la agilización de los procesos.
- EL sistema debe ser seguro, de forma que los datos y operaciones se encuentren resguardados de ataques y pérdidas.

### 3.7 Características generales

El producto consistirá en una aplicación para dispositivos móviles donde se encuentren los procesos de la empresa separados por módulos:

- Pulpería: Los empleados tendrán la posibilidad de registrar los productos que consumen de la pulpería interna, ya sea manualmente o por medio de códigos de barras, de manera que los costos se calculan automáticamente por cada usuario. Existirá dentro de este módulo una sección para los administradores, donde podrán manejar los detalles de cada productos, además de verificar el consumo total de cada empleado con el fin de realizar las rebajas necesarias.
- Vacaciones: Los empleados podrán realizar solicitudes por medio de la aplicación, donde su respectiva sección para los administradores les permitirá aceptarlas o rechazarlas.
- Información: Los empleados podrán consultar datos generales de la empresa como los valores, la misión y la visión. Esta información podrá ser editada por los administradores.

Para abarcar las necesidades con respecto a los dispositivos móviles de los empleados, se hará uso del framework Ionic, el cual ofrece herramientas para el desarrollo de aplicaciones híbridas.

Además, implementará aspectos básicos de seguridad donde se le permita únicamente a los empleados registrarse en la aplicación.

## **4 Estado del arte**

### **4.1 Aplicaciones híbridas**

En los últimos años se ha dado un enorme crecimiento en el uso de los dispositivos móviles, lo cual incluye 'tablets' y teléfonos inteligentes. En el artículo de Malavolta [1] se hace referencia a un punto de por qué se dio este salto en la tecnología móvil, y es que todo es gracias al uso de las aplicaciones móviles, pues conforma gran parte del uso de los medios digitales.

Este avance supone una oportunidad para los desarrolladores de tener un lugar en el mercado, pues aún habiendo hasta dos millones de aplicaciones móviles en el mercado (en el momento de la publicación del artículo) sigue existiendo una gran demanda por software que cumpla con las necesidades que solo un dispositivo móvil puede satisfacer, o al menos haría del proceso mucho más eficiente.

Sin embargo, tanto Malavolta como Heitkötter [2], concuerdan en que hay una brecha que dificulta el desarrollo de las aplicaciones móviles. Esta brecha es la de los sistemas operativos de los dispositivos móviles, y es que el conflicto consiste en que estos sistemas difieren entre ellos significativamente y las herramientas de desarrollo son específicas para cada plataforma. Al existir un mercado tan fragmentado como el de los dispositivos móviles en cuanto a sistemas operativos, siendo los más populares Android y iOS, provoca que si un desarrollador quiere abarcar una audiencia más amplia, tendrá que desarrollar la misma aplicación tantas veces como plataformas quiera incluir.

Para dar la solución a este problema, surgió la opción de desarrollar aplicaciones híbridas, o multiplataforma, las cuales se pudieran programar una sola vez, pero pudieran ser compiladas para varias plataformas. Esto es posible gracias a las ventajas de las tecnologías web, como lo son HTML, CSS y Javascript. Ahora la discusión no solo consiste en decidir cual sistema operativo se ajusta mejor a las necesidades, sino que se añade la posibilidad de utilizar aplicaciones híbridas, eso sí, con sus respectivas ventajas y desventajas con respecto a las aplicaciones nativas. Es por eso que actualmente existen muchos frameworks y herramientas de desarrollo de aplicaciones híbridas, las cuales también proponen sus ventajas y desventajas.

Dentro de los frameworks de desarrollo de aplicaciones híbridas se encuentran varios ejemplos como PhoneGap, Xamarin y Ionic. Cada una difiere de la otra en diferentes puntos, como la arquitectura y el lenguaje. Por ejemplo, la herramienta de Microsoft, Xamarin, utiliza C# en sus aplicaciones para Android, iOS y Windows Phone, mientras que PhoneGap y Ionic se basan en las últimas tecnologías web como CSS3, HTML5, Sass y Javascript, ambas potenciadas por Cordova.

Vilček y Jakopec [3] realizaron un estudio en el que comparaban características entre diferentes frameworks de desarrollo de aplicaciones híbridas, en relación además a los de aplicaciones nativas. Dentro de los criterios de evaluación utilizados se encuentran los sistemas operativos de las máquinas de desarrollo soportados, las plataformas móviles soportadas, el lenguaje de programación, la documentación y comunidad oficial, la velocidad y complejidad de instalación del framework, y la complejidad de desarrollo de las aplicaciones. Una conclusión importante que se debe destacar para tomar decisiones sobre qué tipo de aplicaciones utilizar, es que en cuestiones de desempeño y experiencia del usuario las aplicaciones nativas obtienen un mejor resultado, mientras que las híbridas se caracterizan por requerir menos recursos en desarrollo con un mayor alcance de audiencia.

## 4.2 Ionic Framework

De acuerdo a la comparación antes mencionada, los mismos autores concluyeron que, entre los tres frameworks analizados (Ionic, PhoneGap y NativeScript), Ionic fue el que obtuvo mejores resultados en promedio. Comparado con PhoneGap, el cual también obtuvo muy buenos resultados en el análisis, tiene como desventaja que no presenta tanta compatibilidad en cuanto a plataformas, pues solo compila para las plataformas Android y iOS, además de Windows 10 Universal Apps a partir de la versión 2, mientras que PhoneGap abarca muchas más plataformas: Android, iOS, BlackBerry 10, Windows and Windows Phone, Firefox OS, Ubuntu, Tizen y Amazon Fire OS. Por otro lado, el uso de PhoneGap para el desarrollo de aplicaciones híbridas es (a la fecha de la publicación del artículo) imposible.

Tomando en consideración que el producto propuesto por este informe va dirigido únicamente a las plataformas Android y iOS, además de que el ambiente de desarrollo utilizará como sistema operativo Linux, se puede reforzar la decisión de utilizar el framework Ionic para su desarrollo de acuerdo a los análisis antes propuestos. Además, Bosnic et. al [4] presenta en su paper su experiencia al desarrollar aplicaciones con Ionic y Cordova. Explica en primer lugar cómo el potencial de Ionic se enfoca en el uso de Apache Cordova como base para el manejo de plug ins e interacción con la plataforma, a la vez que utiliza Angular para organizar la lógica y arquitectura de la aplicación, utilizando el patrón MVC, y simplificando en el proceso gran parte del desarrollo del front-end de la aplicación.

Como conclusión, Bosnic analiza el desempeño de su aplicación en comparación a otras aplicaciones nativas. Obtiene como resultado que la aplicación desarrollada con Ionic y Cordova (y las aplicaciones híbridas en general), no se desempeñan tan bien como las nativas, pues presentan un mayor consumo de memoria, procesador y tiempo de respuesta. Esto se evidencia principalmente en aplicaciones complejas, donde se deben mostrar muchos componentes y animaciones, pues estas no acceden directamente a los recursos de la plataforma, deben cargar más librerías y generalmente utilizan el WebView de los dispositivos, el cual requiere mucha memoria por sí solo.

## 5 Análisis de riesgos

Tabla 5.1: Descripción de riesgo N° 1

<b>Nombre</b>	Pérdida de la conexión de internet
<b>Categoría</b>	Tecnológico
<b>Posible causa</b>	Corte de luz
<b>Impacto (I)<sup>1</sup></b>	8
<b>Prob. de ocurrencia (P)</b>	2
<b>Exposición al riesgo (I*P)</b>	16
<b>Estrategia de evasión</b>	Mantener los servidores conectados a respaldos eléctricos.
<b>Estrategia de mitigación</b>	Uso de datos móviles.
<b>Estrategia de contingencia</b>	Uso de datos móviles.

Tabla 5.2: Descripción de riesgo N° 2

<b>Nombre</b>	Pérdida de la base de datos
<b>Categoría</b>	Tecnológico
<b>Posible causa</b>	Eliminación accidental
<b>Impacto (I)</b>	10
<b>Prob. de ocurrencia (P)</b>	1
<b>Exposición al riesgo (I*P)</b>	10
<b>Estrategia de evasión</b>	
<b>Estrategia de mitigación</b>	Realizar respaldos periódicos de la base de datos.
<b>Estrategia de contingencia</b>	Utilizar los respaldos realizados para restablecer la base de datos.

<sup>1</sup> La escala utilizada se representa del 1 al 10, donde 1 es el menos probable y 10 el más probable.

## 6 Objetivos y alcances del sistema

### 6.1 Objetivo general

1. Automatizar los procesos administrativos que realiza Go-Labs, específicamente la gestión de vacaciones, información del marco estratégico y la pulpería interna mediante el desarrollo de una plataforma móvil, que contribuya en la manipulación eficiente de los recursos de la empresa.

### 6.2 Objetivos específicos

1. Proponer una solución mediante historias de usuario para el software de gestión de vacaciones, información y pulpería interna de Go-Labs.
2. Diseñar la arquitectura necesaria para el software de gestión de vacaciones y pulpería interna de Go-Labs.
3. Implementar el diseño para el software de gestión de vacaciones, información y pulpería interna de Go-Labs.
4. Aplicar las pruebas requeridas al software de gestión de vacaciones y pulpería interna de Go-Labs.

### 6.3 Alcances

#### Módulo de pulpería

##### Prioridad: alta

- Se permitirá registrar los productos consumidos de la pulpería interna.
- Se podrá verificar el total consumido.
- Se podrá consultar un historial de productos consumidos.
- Se implementará una sección administrativa que podrá:
  - Agregar, editar y eliminar productos.
  - Verificar el total consumido de cada uno de los usuarios registrados.
  - Realizar un cierre de periodo donde se respalden los datos y se reinicien los valores.

## Módulo de vacaciones

### Prioridad: media

- Se permitirá solicitar un día específico de vacaciones.
- Se permitirá consultar la cantidad de días de vacaciones disponibles actualmente.
- Se implementará una sección administrativa que podrá:
  - Visualizar todas las solicitudes de los usuarios.
  - Aprobar, rechazar o deducir las solicitudes.
  - Consultar y modificar los días disponibles de cada uno de los usuarios.

## Módulo de información

### Prioridad: baja

- Se permitirá consultar información general de la empresa, específicamente los valores, la misión y la visión.
- Se implementará una sección administrativa que podrá editar la información a mostrar.

## 7 Modelo de diseño

### 7.1 Arquitectura conceptual de la solución

La arquitectura de la solución consiste, tal y como se puede apreciar en la Figura 7.1: Diagrama de la arquitectura conceptual de la solución, en una simple interacción cliente - servidor, donde el cliente se basa en una aplicación móvil y el servidor será una aplicación web de Firebase.

El cliente móvil, al ser desarrollado con el framework de Ionic, implementa las metodologías y patrones de diseño de Angular. Además, con la interacción incluida con Apache Cordova, aprovecha los plug-ins disponibles para realizar las conexiones con el servidor.

La base de datos se aloja en un servidor de Firebase, el cual provee funciones de autenticación y copias de seguridad de la base de datos. Para realizar la comunicación entre la aplicación móvil y Firebase, Cordova provee un plug-in llamado AngularFire2, el cual brinda las funciones necesarias para realizar las consultas al servidor, siendo necesario únicamente agregar los parámetros de configuración facilitados por Firebase. Para realizar las consultas, este plug-in implementa observables, a los cuales se pueden suscribir para obtener datos en tiempo real. Así mismo, el plug-in encapsula las funciones de inserción, modificación y eliminación de datos, además de la autenticación y manejo de usuarios.

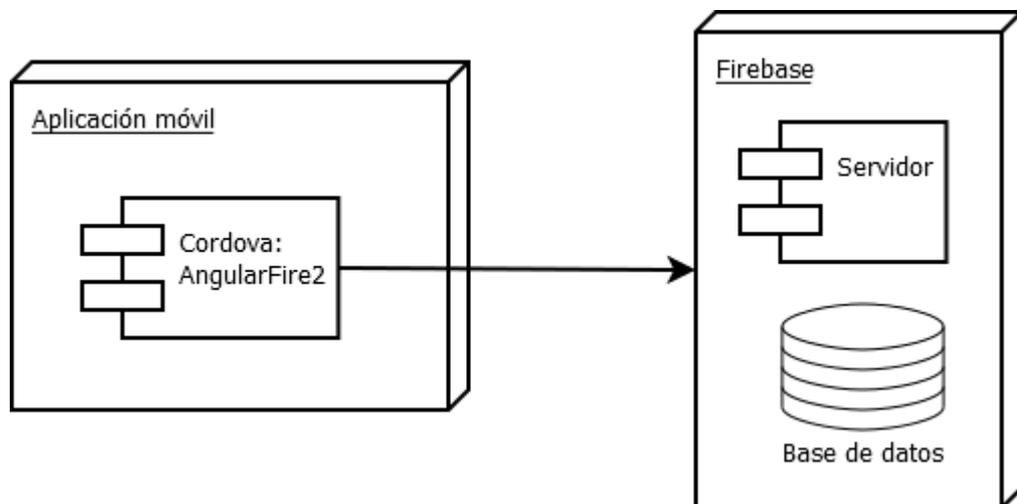
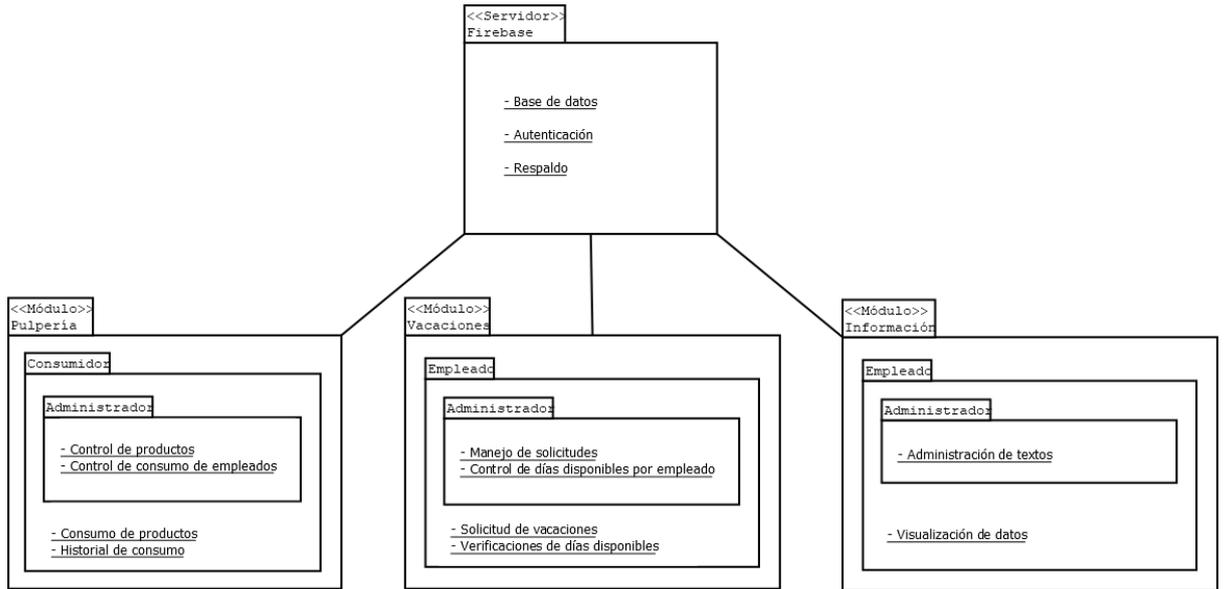


Figura 7.1: Diagrama de la arquitectura conceptual de la solución

## 7.2 Los modelos de subsistemas



**Figura 7.2: Diagrama del modelo de subsistemas de la solución**

El sistema se divide en 3 subsistemas principales que se alimentan de un servidor. Como se puede observar en la Figura 7.2: Diagrama del modelo de subsistemas de la solución, estos 3 subsistemas tienen comportamientos muy similares, pues cada uno encapsula funciones generales y funciones dedicadas a un administrador. Además, el servidor de Firebase se encarga de la autenticación de la aplicación en general y de la conexión en tiempo real de cada uno de los módulos con la base de datos.

En el módulo de pulpería se puede encontrar el comportamiento relacionado a la administración de productos y el consumo de cada empleado, automatizando cálculos y manteniendo datos personalizados.

El módulo de vacaciones se encarga de controlar los periodos de vacaciones de cada empleado de la empresa, donde las funciones de administrador permite manejar solicitudes de manera eficiente.

El módulo de información está constituido por funciones de visualización y manejo de información pública de la empresa, de forma que los empleados tengan fácil acceso a datos que los motiven en las ideologías de la empresa.

### 7.3 Interfaces de usuario

Para mostrar el funcionamiento general de la aplicación, a continuación se podrán encontrar las interfaces de los módulos de pulpería, vacaciones e información. Estas pantallas muestran solo las funciones prominentes de los diferentes módulos, obviando pantallas con mensajes de alerta y formularios.

## Vistas generales:



Figura 7.3: Página de ingreso

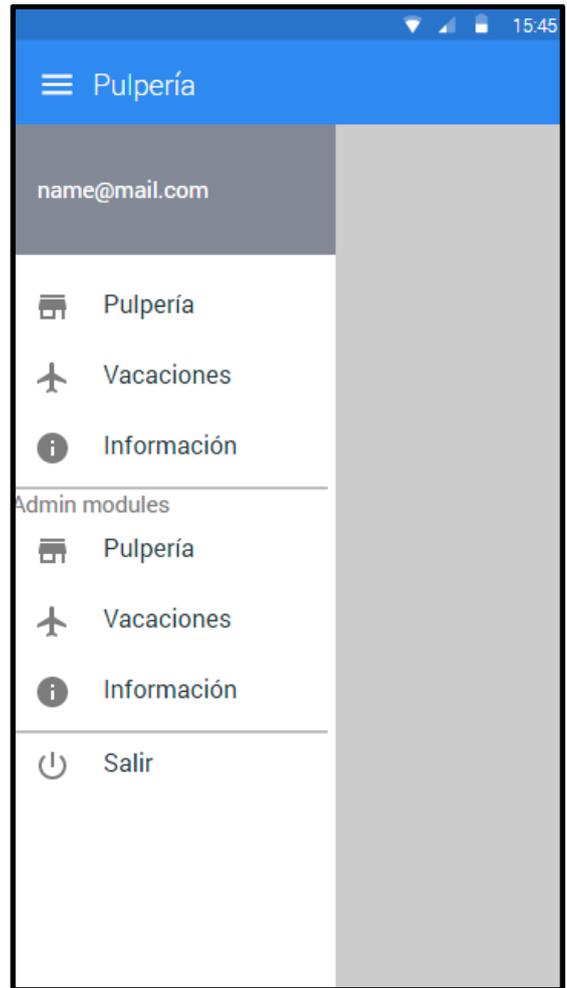


Figura 7.4: Menú de navegación

La navegación de la aplicación comenzará en una pantalla de inicio, donde se ingresará por medio del correo del dominio de Go-Labs.net. A partir de ahí, se accederá a las diferentes secciones de la aplicación por medio de un menú lateral.

## Módulo de pulpería:

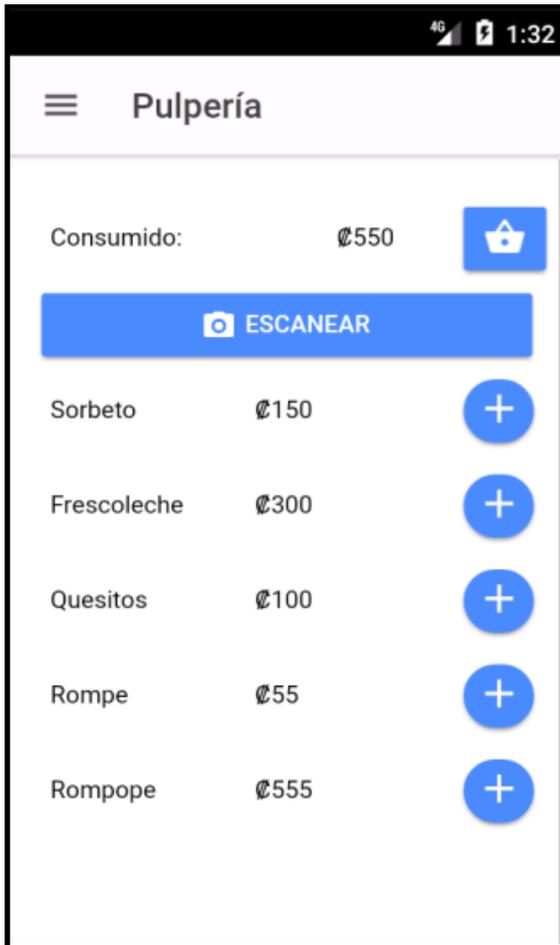


Figura 7.5: Registro de productos

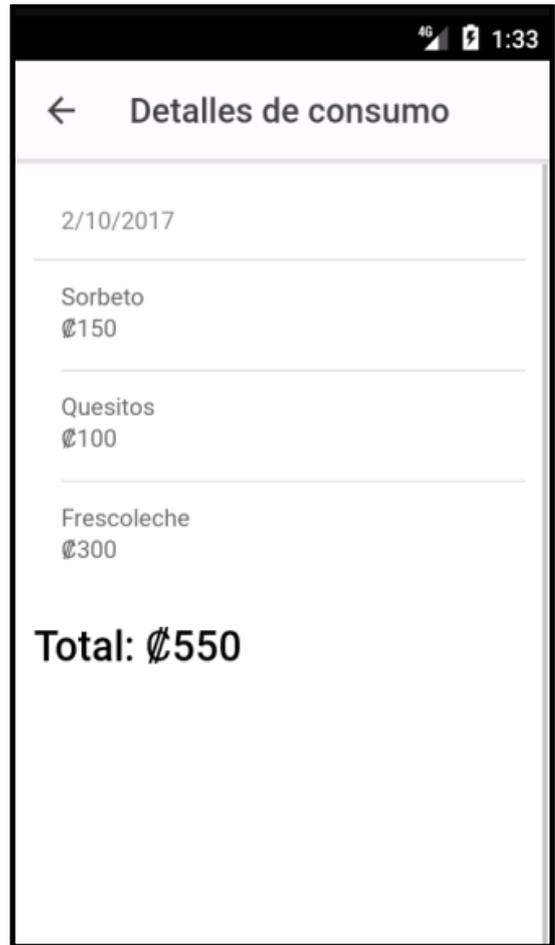


Figura 7.6: Historial de consumo

El módulo de la pulpería le permitirá al empleado comprar productos de dos maneras; seleccionando el producto individual o escaneando un código de barras. Además, este podrá ver un historial de las compras realizadas en ese periodo, agrupadas por fecha y calculando el total consumido.

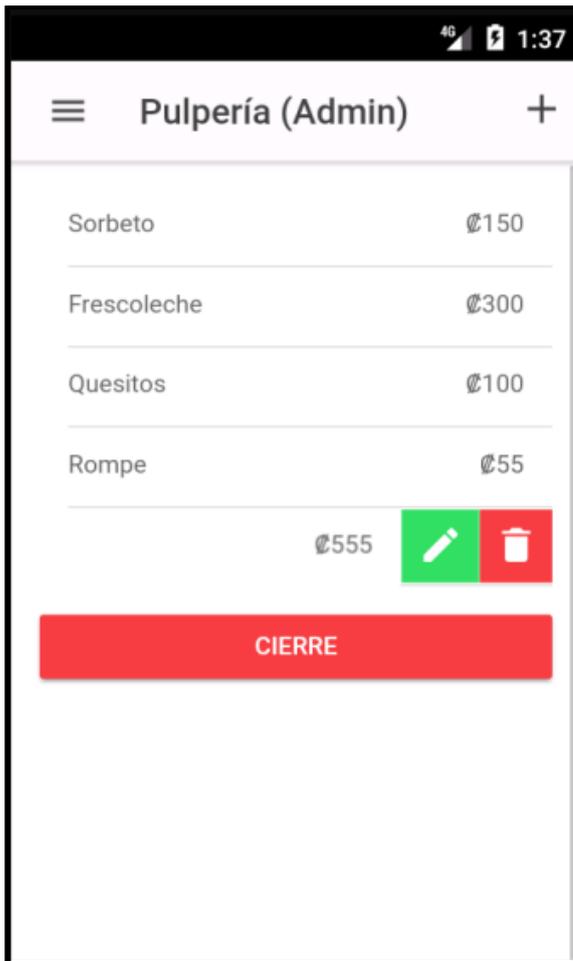


Figura 7.7: Administración de productos

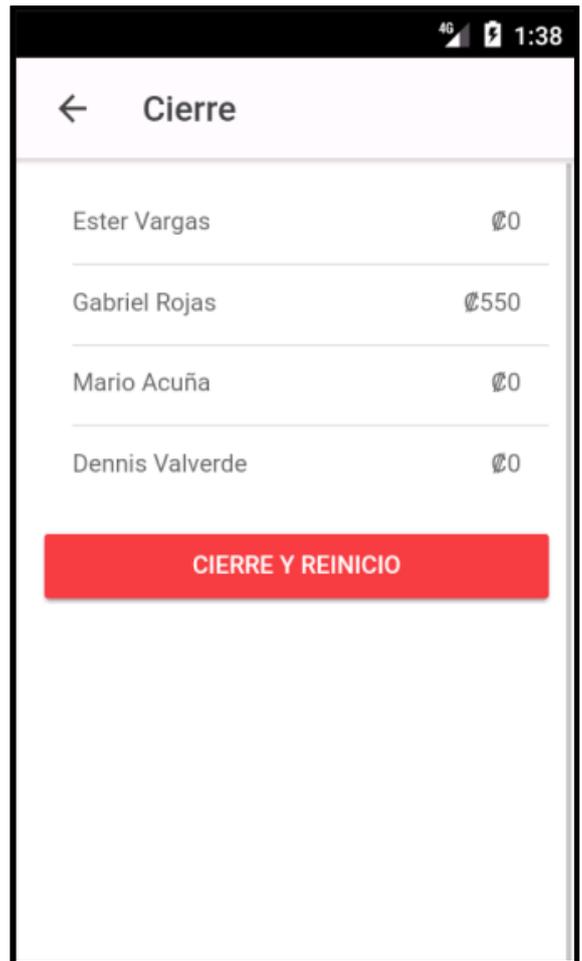


Figura 7.8: Cierre de periodo

En la sección del administrador de la pulpería se permite visualizar la lista de productos, agregar, modificar y eliminar los mismos. Las opciones se mostrarán al realizar el gesto de deslizamiento en el ítem deseado. Además, por este medio se accede a la función de cierre de periodo, donde se podrá visualizar el total consumido por cada empleado para realizar las respectivas deducciones, y reiniciar los datos para el siguiente periodo.

## Módulo de vacaciones:



Figura 7.9: Visualización de solicitudes y días disponibles



Figura 7.10: Solicitud de periodo de vacaciones

Los empleados podrán solicitar días de vacaciones seleccionando la fecha deseada en un calendario, donde se les permitirá seleccionar el período del día. Las solicitudes se mostrarán en diferentes colores con respecto a su estado; gris si están pendientes, con borde verde como aprobada, relleno verde como consumida y rojo como rechazada.



Figura 7.11: Administración de solicitudes

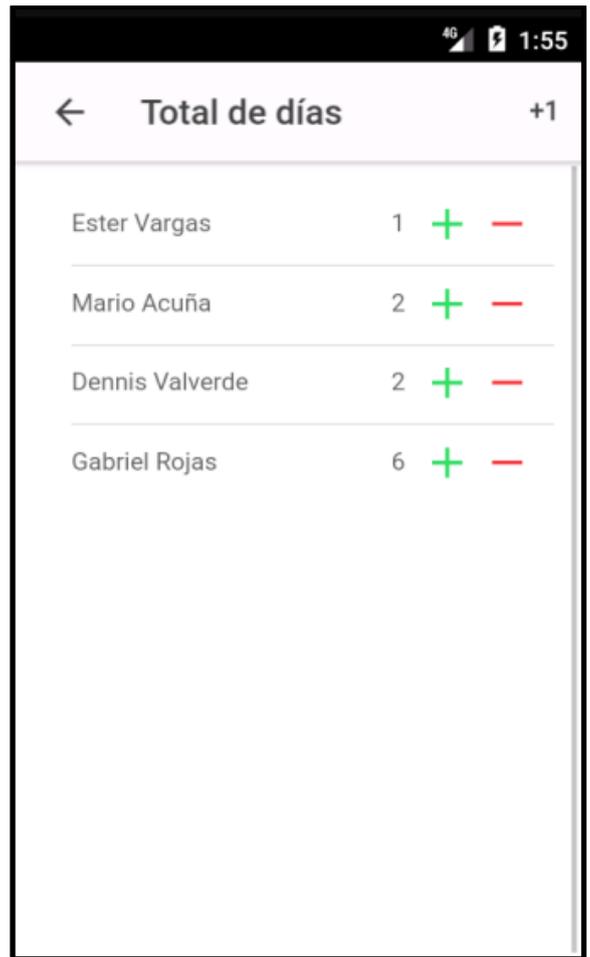


Figura 7.12: Administración de días de vacaciones

En la sección del administrador del módulo de vacaciones se podrá visualizar todas las solicitudes realizadas por los empleados, así como rechazarlas o aprobarlas por medio de opciones accesibles con un gesto de deslizamiento. Las solicitudes se mostrarán con el mismo sistema de colores que el del empleado. El administrador también podrá controlar los días de vacaciones disponibles de cada empleado, ya sea manualmente para cada uno o aumentando uno para a todos, por ejemplo, en el caso de un nuevo periodo.

## Módulo de información:

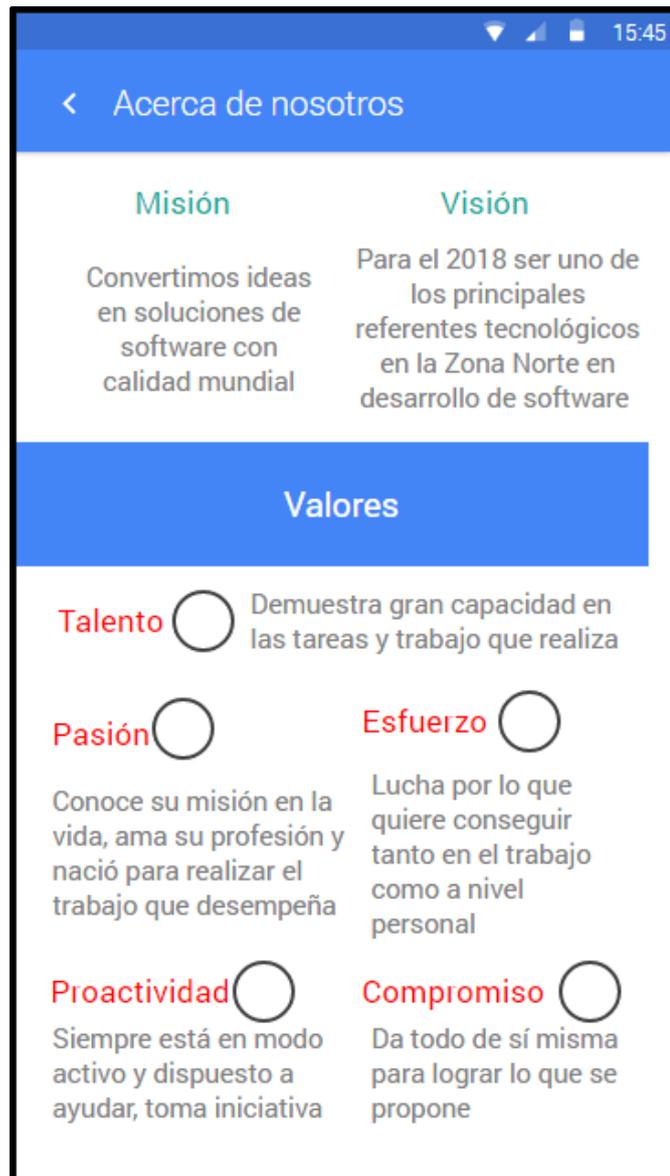


Figura 7.13: Visualización de información de la empresa

Esta pantalla es principalmente informativa, donde el usuario podrá visualizar tanto los valores como la misión y visión establecidas por la empresa.

## 7.4 Componentes y servicios

**Ionic:** Se utiliza el framework Ionic para estructurar y diseñar la aplicación móvil. Se aprovecha la interacción con Angular y Cordova para el control de formularios, componentes y servicios. Ionic Framework permite el desarrollo de la aplicación híbrida con tecnologías web (HTML, CSS y JavaScript) para reducir costos de programación específicos para cada plataforma. En esta aplicación se aprovechan los componentes y directivas proporcionadas por el framework para agilizar el proceso de desarrollo.

**Cordova:** Se utiliza el framework de Apache Cordova para administrar los plugins de la aplicación que interactúan con las funciones nativas de cada plataforma. Se utilizan componentes de comunicación con el servidor de Firebase, escáner de código de barras y visualización de calendarios.

**Firebase:** El servidor de Firebase administra la base de datos de la aplicación, suministrando funciones de suscripción en tiempo real de los datos almacenados. Cuenta además con funciones de autenticación compatibles con Google OAuth, lo cual agrega la posibilidad de restringir el acceso a la aplicación a correos del dominio de Google, por ejemplo, correos específicos para los empleados de la empresa.

## 7.5 Diseño de base de datos

Para el manejo de base de datos se decidió utilizar la herramienta Firebase. En esta se aloja una base de datos no-relacional, por lo que no cuenta con modelo de entidad-relación.

Cada módulo de la aplicación accede a sus respectivas colecciones, por ejemplo, el módulo de pulpería accede a la colección de productos, cada uno con su nombre, precio y código de barras, y las ventas de cada empleado.

A continuación, se describen los objetos más importantes almacenados en las colecciones de la base de datos:

**Tabla 7.1: Descripción de la base de datos**

Nombre	Atributos
Usuario	<ul style="list-style-type: none"> <li>● Nombre</li> <li>● Email</li> <li>● 'Es administrador'</li> </ul>
Producto	<ul style="list-style-type: none"> <li>● Código</li> <li>● Nombre</li> <li>● Precio</li> </ul>
Ventas	<ul style="list-style-type: none"> <li>● Usuario</li> <li>● Total</li> <li>● Lista de productos</li> <li>● Fecha</li> </ul>
Vacaciones	<ul style="list-style-type: none"> <li>● Usuario</li> <li>● Días disponibles</li> <li>● Lista de peticiones</li> <li>● Última modificación</li> </ul>
Petición	<ul style="list-style-type: none"> <li>● Usuario</li> <li>● Fecha</li> <li>● Periodo</li> <li>● Estado</li> </ul>
Información	<ul style="list-style-type: none"> <li>● Misión</li> <li>● Visión</li> <li>● Talento</li> <li>● Pasión</li> <li>● Esfuerzo</li> <li>● Proactividad</li> <li>● Compromiso</li> </ul>

## 8 Plan de trabajo

Para organizar y ejecutar el plan de trabajo se utilizará la metodología Kanban, donde se asignarán las tareas por módulo y se actualizarán sus estados conforme se trabajan en ellas. Se utilizará además la herramienta YouTrack para aplicar esta metodología.

El Backlog actual del producto contiene las siguientes tareas ordenadas por prioridad:

### **Módulo de pulpería:**

- Instalación y configuración del ambiente de desarrollo
- Diseño de mockups de las páginas de inicio, registro y menú de navegación.
- Implementación de módulo de Log In.
- Implementación del menú de navegación.
- Diseño del mockup y base de datos del módulo de pulpería.
- Desarrollo e implementación de la vista principal de la pulpería.
- Desarrollo e integración de funciones de escaneo de códigos de barra.
- Desarrollo e implementación de la página del historial de consumo.
- Diseño del mockup del módulo de administración de la pulpería.
- Desarrollo e implementación de la vista principal del administrador de la pulpería.
- Desarrollo e integración de funciones de ingresar, editar y eliminar productos.
- Desarrollo e implementación de funciones de cierre y reinicio de periodo.

### ***Módulo de vacaciones:***

- Diseño de mockups y base de datos del módulo de vacaciones.
- Desarrollo e implementación de la vista principal del módulo de vacaciones.
- Desarrollo e integración de funciones de solicitud de vacaciones.
- Diseño de mockups del módulo de administración de vacaciones.
- Desarrollo e implementación de la vista principal del módulo de administración de vacaciones.
- Desarrollo e integración de funciones de aprobación o rechazo de solicitudes.
- Desarrollo e implementación de vista de consulta de solicitudes aprobadas.
- Desarrollo e integración de función para marcar solicitudes como leídas/rebajadas.
- Desarrollo e implementación de la vista de consulta del total de días de vacaciones de cada empleado.

- Desarrollo e integración de funciones de aumento y rebajo de días de vacaciones.
- Desarrollo e integración de funciones de rebajo de días de vacaciones.
- Desarrollo e integración de funciones de respaldo de cambios de estado en las solicitudes y días de vacaciones.

#### **Módulo de información:**

- Diseño de mockups y base de datos del módulo de información.
- Desarrollo e implementación de la vista principal del módulo de información.
- Desarrollo e integración de funciones de conexión con base de datos del módulo de información.
- Desarrollo e implementación de la vista principal del módulo de administración de información.
- Desarrollo e integración de funciones de modificación de textos.

## 9 Conclusiones y recomendaciones

Los objetivos y el alcance propuesto para este proyecto fueron cumplidos satisfactoriamente, finalizando con éxito el proceso de práctica profesional al entregar un producto que cumple con las expectativas propuestas por la empresa.

En conclusión, para el primer objetivo se propuso la solución de crear una aplicación móvil con el fin de automatizar los procesos de la empresa. Si bien no se utilizaron historias de usuario como se menciona, se definió el proyecto por medio de un 'backlog' de tareas, siguiendo los lineamientos y costumbres de la empresa.

Con respecto al segundo objetivo, se diseñó el sistema, flujo e interfaz por medio de 'mockups'. También se realizó un diseño preliminar de la base de datos en una representación JSON, ajustándose adecuadamente a la propuesta del uso de una base de datos no-relacional.

Para el tercer objetivo, se logró la implementación de todos los módulos propuestos en el alcance, además de su respectiva conexión con el servicio de Firebase.

Por último, el proceso de pruebas de calidad fue llevado a cabo en conjunto con el departamento de Aseguramiento de la Calidad. Se realizaron 99 casos de prueba en dos escenarios diferentes: 56 con conexión a internet y 43 sin conexión a internet. En el caso del comportamiento con la conexión a internet, se probaron casos como las validaciones en los formularios y las funcionalidades de los botones. Por otro lado, en el escenario sin conexión a internet se validó la estabilidad de la aplicación. En total se reportaron 18 defectos y fallos, de los cuales todos se corrigieron, dando como resultado un estado de éxito en el 100% de las pruebas.

Para cumplir con lo propuesto, fueron influyentes varios aspectos. En primer lugar, el uso de la metodología Kanban propició una organización adecuada de las tareas, pues se dio prioridad y límites para estas, y junto con la herramienta Youtrack, permitía al Project Manager mantener supervisión sobre el proceso de cada una de las tareas.

Youtrack implementa esta metodología por medio de cuatro columnas: en espera, en proceso, corregido/finalizado y verificado, en las cuales se colocan las tareas según su estado. Además, a estas tareas se les puede asignar otros atributos como la prioridad, el encargado, el tiempo estimado, entre otros, que complementan la organización del trabajo.

Como parte del seguimiento, al final del día se debía registrar la cantidad de horas trabajadas en cada tarea, y en la sección de reportes el PM podía analizar el tiempo invertido en cada tarea en un periodo determinado. Esto le permitía al PM obtener un resumen laboral al final de la semana y dar retroalimentación al respecto. De esta forma, se omite la necesidad de realizar reuniones de avance diarias o semanales, sino cada vez que se consideraba necesario de acuerdo al análisis de los reportes.

Por otro lado, la orientación por parte de los compañeros y supervisores con respecto a metodologías y buenas prácticas contribuyó no sólo al desarrollo eficiente de la solución, sino también a la calidad del producto final.

Como parte de los resultados, se pudo apreciar la ventaja de implementar las tecnologías propuestas en la solución. Para el desarrollo de la aplicación móvil, el framework de Ionic propició el uso de componentes simples y técnicas de creación de interfaces que agilizaron el proceso en gran medida, junto con los plug-ins disponibles para Cordova, pues facilitaron el desarrollo de la parte lógica de la aplicación, empleando en este caso componentes para el escaneo de códigos de barras, formularios con calendarios y conexión con la base de datos.

Así mismo, Firebase demostró ser viable para su uso en aplicaciones pequeñas, pues brinda funcionalidades útiles como la base de datos no relacional en tiempo real y la autenticación al costo de la mínima configuración. Sin embargo, se encontró como desventaja de la poca flexibilidad en cuanto a proveedores externos de autenticación, esto al intentar integrar los servicios de autenticación de Google para restringir el dominio de los correos de ingreso de los usuarios, pues requirió de más tiempo de lo estimado en cuanto a tareas de configuración.

Una recomendación en cuanto al diseño de la base de datos, es la implementación de una lista de valores en el módulo de información del marco estratégico de la empresa, pues en caso que se decida agregar o eliminar algún valor, el código debería ser actualizado para adaptarse a este cambio.

Con respecto a la experiencia, resulta interesante realizar una comparación entre las actividades como estudiante y como practicante. Cumplir con un horario más estricto influye en el enfoque al proyecto en el que se trabaja, mientras que los horarios dispersos del estudio requerían mayor esfuerzo para la correcta organización de las tareas. Por otro lado, se puede apreciar cómo se puede poner en práctica lo aprendido en la carrera, por ejemplo, implementando buenas prácticas de desarrollo y técnicas de aprendizaje.

## 10 Referencias

- [1] I. Malavolta, "Web-Based Hybrid Mobile Apps: State of the Practice and Research Opportunities", en 2016 IEEE/ACM International Conference on Mobile Software Engineering and Systems (MOBILESoft), mayo de 2016, págs. 241-242. doi: 10.1109/MobileSoft.2016.050.
- [2] H. Heitkötter, S. Hanschke y T. A. Majchrzak, "Evaluating Cross-Platform Development Approaches for Mobile Applications", en Web Information Systems and Technologies: 8th International Conference, WEBIST 2012, Porto, Portugal, April 18-21, 2012, Revised Selected Papers, J. Cordeiro y K.-H. Krempels, eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, págs. 120-138, isbn: 978-3-642-36608-6. doi: 10.1007/978-3-642-36608-6\_8. dirección: [https://doi.org/10.1007/978-3-642-36608-6\\_8](https://doi.org/10.1007/978-3-642-36608-6_8).
- [3] T. Vilček y T. Jakopec, "Comparative analysis of tools for development of native and hybrid mobile applications", en 2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), mayo de 2017, págs. 1516-1521. doi: 10.23919/MIPRO.2017.7973662.
- [4] S. Bosnic, I. Papp y S. Novak, "The development of hybrid mobile applications with Apache Cordova", en 2016 24th Telecommunications Forum (TELFOR), nov. de 2016, págs. 1-4. doi: 10.1109/TELFOR.2016.7818919.