

UNIVERSIDAD DEL VALLE DE GUATEMALA

Facultad de Ingeniería



Propuesta de automatización en la entrega de reportes del consumo de combustible de gasolinera móvil de una empresa agroindustrial

Trabajo de graduación presentado por Juan de Dios Raymundo López Gálvez para optar al grado académico de Licenciado en Tecnología de Sistemas Informáticos.

Guatemala,

2022

UNIVERSIDAD DEL VALLE DE GUATEMALA

Facultad de Ingeniería




Propuesta de automatización en la entrega de reportes del consumo de combustible de gasolinera móvil de una empresa agroindustrial

Trabajo de graduación presentado por Juan de Dios Raymundo López Gálvez para optar al grado académico de Licenciado en Tecnología de Sistemas Informáticos.


Guatemala,


2022


Vo.Bo. :

(f) 
Ing. Samuel Melquisedec Molina Donis
Asesor

Tribunal Examinador:

(f) 
Ing. Samuel Melquisedec Molina Donis
Asesor

(f) 
Ing. Eddyn Geovani Figueroa López
Examinador

(f) 
Ing. Mario Adolfo Sian Quisque
Director

Fecha de aprobación: Guatemala, 14 de diciembre de 2022

ÍNDICE

LISTA DE TABLAS	<i>iii</i>
LISTA DE ILUSTRACIONES.....	<i>iv</i>
RESUMEN	<i>vi</i>
I. INTRODUCCIÓN	<i>1</i>
II. OBJETIVOS.....	<i>2</i>
III. JUSTIFICACIÓN	<i>3</i>
IV. MARCO TEÓRICO	<i>4</i>
A. Dispositivos móviles.....	<i>4</i>
B. Características generales de los dispositivos móviles.....	<i>4</i>
C. Tipos de dispositivos móviles.....	<i>5</i>
D. Redes a las que puede acceder un dispositivo móvil	<i>6</i>
E. Sistemas operativos.....	<i>8</i>
F. Aplicaciones móviles.....	<i>12</i>
G. Aplicaciones nativas	<i>13</i>
H. Software development kit (SDK).....	<i>14</i>
I. Android SDK	<i>15</i>
J. Lenguaje de programación java.....	<i>16</i>
K. Entorno de desarrollo integrado (IDE).....	<i>19</i>
L. Android Studio	<i>19</i>

M.	Base de datos	21
N.	Modelo de datos	25
O.	Patrón de diseño MVC	27
V.	METODOLOGÍA	31
1.	Investigación.....	31
2.	Alcance	31
3.	Requerimientos	31
4.	Diagramas	35
5.	Diseño de propuesta de interfaces.....	36
VI.	RESULTADOS	50
VII.	ANÁLISIS DE RESULTADOS.....	51
VIII.	CONCLUSIONES	52
IX.	BIBLIOGRAFÍA	53
X.	ANEXOS.....	53
XI.	GLOSARIO	56

LISTA DE TABLAS

Tabla 01: Requerimiento funcional autenticación de usuario activo en la empresa.....	32
Tabla 02: Requerimiento funcional formulario interactivo para ingreso de nuevos despachos de combustible	32
Tabla 03: Requerimiento funcional reportes de despacho de combustible.....	32
Tabla 04: Requerimiento funcional, base de datos, manejo local de información.....	33
Tabla 05: Requerimiento funcional sincronización de base de datos local con servidor...	33
Tabla 06: Requerimiento no funcional interfaz intuitiva del sistema.....	34
Tabla 07: Requerimiento no funcional disponibilidad.....	34
Tabla 08: Requerimiento no funcional seguridad e integridad de datos.....	34
Tabla 09: Requerimiento no funcional rendimiento y desempeño.....	35
Tabla 10: Especificaciones diagrama caso de uso general	36
Tabla 11: Versiones de Android recomendadas	55

LISTA DE ILUSTRACIONES

Ilustración 01: Teléfono inteligente TCL 20 SE.....	6
Ilustración 02: Estación 4G.....	7
Ilustración 03: Sistemas operativos.....	8
Ilustración 04: Funciones de sistema operativo	9
Ilustración 05: Logo iOS	10
Ilustración 06: Logo Android	11
Ilustración 07: Tipos de aplicaciones móviles.....	13
Ilustración 08: Flujo de desarrollo en apps nativas	14
Ilustración 09: Portabilidad de java.....	17
Ilustración 10: Esquema elementos plataforma java.....	18
Ilustración 11: Los archivos de proyecto en la vista de Android	21
Ilustración 12: Base de datos en una organización.....	23
Ilustración 13: SGBD actuales.....	24
Ilustración 14: Modelo de datos de alto nivel.....	27
Ilustración 15: Modelo de datos de bajo nivel.....	27
Ilustración 16: Ciclo de vida del patrón MVC	30
Ilustración 17: Diagrama de caso de uso general	35
Ilustración 18: Diagrama entidad relación	36
Ilustración 19: Diseño de interfaz inicio de sesión	37
Ilustración 20: Diseño de interfaz menú principal	38
Ilustración 21: Diseño de interfaz de nuevo despacho de combustible.....	39
Ilustración 22: Diseño de interfaz de confirmar datos para nuevo despacho.....	40
Ilustración 23: Diseño de interfaz de transacción guardada correctamente.....	41
Ilustración 24: Diseño de interfaz de menú de reportes de despacho de combustible.....	42

Ilustración 25: Diseño de interfaz de pantalla para reporte despacho por bomba.....	43
Ilustración 26: Diseño de interfaz de pantalla para reporte despacho por empresa.....	44
Ilustración 27: Diseño de interfaz de pantalla para reporte despacho por tipo vehículo....	45
Ilustración 28: Diseño de interfaz para reporte despacho por tipo de combustible.....	46
Ilustración 29: Diseño de interfaz de pantalla de sincronizar base de datos manualmente	47
Ilustración 30: Diseño de interfaz de alerta de sincronización correcta.....	48
Ilustración 31: Diseño de interfaz de alerta de error en sincronización.....	49
Ilustración 32: Dispositivo utilizado usualmente por la empresa.....	55

RESUMEN

Para que una empresa crezca de forma sostenible es fundamental el fácil acceso a la información y, sobre todo, que esta sea fiable. En este escenario, la automatización es importante, ya que brinda un fácil acceso a la información, también ayuda a optimizar recursos, reducir fallas, etc.

Actualmente, la automatización es parte fundamental para el crecimiento en la agroindustria azucarera de Guatemala, ya que nos permite ganar tiempo en tareas operativas o tediosas como el llenado en papel de cierta información, ingreso de los documentos, el archivado, las búsquedas, etc. A su vez, acelera el proceso y lo convierte en algo más productivo, logrando así acortar tiempos importantes.

El presente proyecto tiene como objetivo presentar una propuesta en respuesta a la necesidad de automatizar el proceso de despacho y envío de información de una gasolinera móvil, por medio de una aplicación móvil para dispositivos Android, logrando así reducir costos en su totalidad en papel, lapiceros e impresiones, reducir tiempos de envío de información al sistema central, integridad y seguridad en los datos ingresados.

La propuesta plantea una aplicación móvil de fácil manejo y acceso, almacenar datos de manera local en una base de datos para lograr una integridad, con un método de autenticación del usuario final, garantizando que solo posea acceso a la información un usuario autorizado por la empresa.

I. INTRODUCCIÓN

Actualmente, las tecnologías de la información y comunicación han revolucionado la forma de comunicación y recuperación de información. En este contexto, las aplicaciones móviles proveen a sus usuarios un acceso ágil a ingreso y consulta de información desde sus dispositivos tecnológicos a información de un sistema informático.

El uso de herramientas tecnológicas es una realidad latente, que entra a formar parte fundamente de las empresas en sus procesos, automatizando estos mismos. Para esto, las herramientas preferidas resultan siendo los teléfonos inteligentes. Esto sucede por la necesidad de gestionar información en tiempo real.

Razón por la que se optó por realizar la presente propuesta en una aplicación móvil que automatice el ingreso de información obtenida en los despachos de combustible y consulta de información presentada por medio de reportes de despachos de combustibles de una gasolinera móvil, con el fin de tener información íntegra y segura, evitando pérdidas de información, el uso de papel, lapicero e impresiones para los reportes y hoja de despachos en su formato físico.

El proceso actual es pausado y redundante, dependiendo en su totalidad en la correcta escritura de los expendedores en la hoja de despachos, en su formato físico de las distintas bombas móviles de las gasolineras, estando en pausa la información hasta que las bombas móviles regresen a la empresa, para realizar la tabulación de la información al sistema de la empresa, obteniendo los reportes de los distintos despachos al finalizar la tabulación. Podemos notar que este proceso actual es tardado y consume recursos para llevarse a cabo, observando esta oportunidad de mejora se planteó esta propuesta de implementación de automatización.

II. OBJETIVOS

A. Objetivo general

1. Propuesta de automatización del proceso de generación de despachos de combustible de gasolineras móviles a través de una aplicación con sistema operativo Android.

B. Objetivos específicos

1. Automatizar la generación de boletas de despacho de múltiples tipos de combustible en las bombas móviles de cascos de las fincas, reduciendo el 100% el uso de papel.
2. Diseñar un prototipo de aplicación móvil para sistema operativo Android, con lenguaje de programación java para generación de boletas de despachos y consultas históricas de despacho de combustibles, para reducir el tiempo de digitación de boletas de despacho de 8 horas a 3 horas diarias durante el periodo de zafra.
3. Automatizar el envío del 100% de la información de despachos de combustible desde la aplicación móvil al ERP de la empresa por medio de una conexión en tiempo real por internet.

III. JUSTIFICACIÓN

La información siempre ha sido de vital importancia para medir, controlar y tomar decisiones de una empresa, el acceso a la información en tiempo real es un reto que al día de hoy muchas organizaciones no han podido resolver, por lo cual siguen con procesos obsoletos los cuales nos brindan información limitada o incompleta.

Los sistemas de información constituyen una herramienta en el ámbito organizacional, son importantes para el desarrollo de actividades, automatización de procesos, sosteniendo la integridad, seguridad y gestión de datos en tiempo real. Las aplicaciones móviles proveen a sus usuarios un acceso ágil desde sus dispositivos tecnológicos a información de un sistema informático.

Tomando en cuenta la temática de la información en tiempo real y la automatización de procesos, el presente trabajo responde con una propuesta de llevar un mejor control digital de ingreso de datos sobre nuevos despachos de combustible, reportes de consumo de combustible y agilización en la entrega de los mismos, esta propuesta es una aplicación móvil Android, que contiene un formulario con los campos necesarios para el llenado de una nueva orden de despacho, consulta de datos por medio de reportes con parámetros, la aplicación tendrá acceso a la red, el objetivo de este es identificar una conexión a internet para la carga de información al servidor del establecimiento.

Actualmente, se llevan en un formato de hojas de papel, para luego proceder a tabularlos en una hoja de Excel, con el peligro que en el proceso pueda suceder pérdida de algún despacho de combustible físico y con ello derive pérdida de información. Por lo tanto, reduce el riesgo en la pérdida de información, optimiza el proceso actual, elimina el formato físico en hojas de papel y a su vez reduce el tiempo para la entrega de nuevos despachos de combustible y la generación de reportes con datos actualizados.

IV. MARCO TEÓRICO

A. Dispositivos móviles

En los últimos años, buena parte del desarrollo en el mundo del cómputo se ha volcado hacia el modelo de cómputo representado, genéricamente, por los dispositivos móviles. Dado el interés que estas plataformas han despertado, se torna necesario abordar el tema, aunque sea más para anotar similitudes que diferencias con el resto de los equipos de cómputo. Para hacer esto, sin embargo, es necesario primero abordar la definición: ¿en qué consiste un dispositivo móvil, cuáles son los límites de su definición, qué fronteras se le pueden definir?

Es difícil encontrar límites claros y duros para lo que este concepto abarca; en el transcurso de esta sección se abordan las características de las computadoras diseñadas no solo en el nivel del hardware, sino de interfaz de usuario, para que su propietario las cargue consigo y las convierta en un asistente para sus actividades cotidianas, para la organización de su vida diaria. Partiendo de esta definición se tiene que un teléfono inteligente será tratado como dispositivo móvil, pero una computadora portátil no, puesto que su interfaz es la misma de una computadora estándar.

Claro, esta definición –indudablemente rápida e imperfecta– deja una gran área gris, y permite cierta ambigüedad. Por ejemplo, las más recientes versiones de algunos entornos de usuario (notablemente, la interfaz primaria de Windows 8, o los entornos GNOME y Unity de Linux) buscan unificar la experiencia, incorporando conceptos del multitáctil a los escritorios y acercando los casos de uso. Tómense, pues, estos lineamientos como meramente indicativos (Gunnar Wolf, Estaban Ruiz, Federico Bergero, Erwin Meza, 2015.).

B. Características generales de los dispositivos móviles

Una gran cantidad de dispositivos electrónicos se clasifican actualmente como dispositivos móviles, desde teléfonos hasta tabletas, pasando por dispositivos como lectores de RFID. Con tanta tecnología clasificada como móvil, puede resultar complicado determinar cuáles son las características de los dispositivos móviles.

Antes de describir detalladamente algunos dispositivos móviles, vamos a concretar el concepto de dispositivo tratado en esta asignatura. A continuación, detallamos las características esenciales que tienen los dispositivos móviles:

- Son aparatos pequeños.
- La mayoría de estos aparatos se pueden transportar en el bolsillo del propietario o en un pequeño bolso.
- Tienen capacidad de procesamiento.
- Tienen conexión permanente o intermitente a una red.
- Tienen memoria (RAM, tarjetas Micro SD, flash, etc.).
- Normalmente, se asocian al uso individual de una persona, tanto en posesión como en operación, la cual puede adaptarlos a su gusto.
- Tienen una alta capacidad de interacción mediante la pantalla o el teclado.

En la mayoría de los casos, un dispositivo móvil puede definirse con cuatro características que lo diferencian de otros dispositivos que, aunque pudieran parecer similares, carecen de algunas de las características de los verdaderos dispositivos móviles. Estas cuatro características son:

- 1) Movilidad
- 2) Tamaño reducido
- 3) Comunicación inalámbrica
- 4) Interacción con las personas

(Julián David Morillo Pozo, 2011)

C. Tipos de dispositivos móviles

El término dispositivo móvil cubre un amplio rango de dispositivos electrónicos de consumo. Normalmente, por dispositivo móvil nos referimos a un dispositivo que puede conectarse a Internet. No obstante, algunas veces también se clasifican cámaras digitales y reproductores MP3 o MP4 estándares como dispositivos móviles. Uno de estos dispositivos es el teléfono inteligente (Julián David Morillo Pozo, 2011).

1. Teléfonos inteligentes

Los smartphones o teléfonos inteligentes son teléfonos que soportan más funciones que un teléfono común. Entre estas funciones suelen encontrarse la de gestor de correo electrónico, la funcionalidad completa de organizador personal, y suelen estar pensados para acceder de manera continua a Internet. Actualmente, se les añade como función común la posibilidad de instalar programas adicionales.

Uno de los mayores atractivos de los smartphones es su simplicidad. El usuario medio puede tener su dispositivo funcionando en cuestión de minutos sin tener que preocuparse de una configuración complicada.

Estos teléfonos inteligentes vienen integrados la mayoría con los teclados software, nos referimos a los teclados que los dispositivos con pantalla táctil pueden presentarnos por pantalla para introducción de texto, ya sea mediante la utilización de un lápiz o con nuestros propios dedos. (Julián David Morillo Pozo, 2011).



Ilustración 1: Teléfono inteligente TCL 20 SE

(TCL, 2021)

D. Redes a las que puede acceder un dispositivo móvil

Muchos teléfonos móviles disponibles actualmente soportan tanto tecnologías celulares como otras tecnologías de banda ancha inalámbricas. Por ejemplo, los

dispositivos Samsung, Huawei, Motorola, iPhone, entre otros la proporcionan (Julián David Morillo Pozo, 2011).

1. Red 4G

La red 4G estará totalmente basada en el protocolo IP (En inglés internet protocol). Es un sistema de sistemas y una red de redes. Se pretende alcanzar la convergencia entre redes de cables e inalámbricas para proporcionar velocidades de acceso de entre 100 Mbps en movimiento y 1 Gbps en reposo, y mantener la calidad de servicio punta a punta, así como la seguridad para poder ofrecer servicios de cualquier clase en cualquier momento y en cualquier lugar con el mínimo coste posible (Julián David Morillo Pozo, 2011).

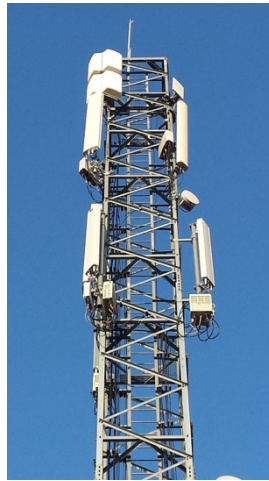


Ilustración 2: Estación 4G

(Ibanobitz, 2015)

El WWRF (Wireless World Research Forum) define 4G como una red que funciona en la tecnología de Internet que se combina con otros usos y tecnologías (tales como Wifi y WiMAX). 4G no es una tecnología o estándar definido, sino una colección de tecnologías y protocolos que permiten el máximo rendimiento con la red inalámbrica más barata (Julián David Morillo Pozo, 2011).

E. Sistemas operativos

El sistema operativo es el principal programa que se ejecuta en toda computadora de propósito general.

Los hay de todo tipo, desde muy simples hasta terriblemente complejos, y entre más casos de uso hay para el cómputo en la vida diaria, más variedad habrá en ellos.

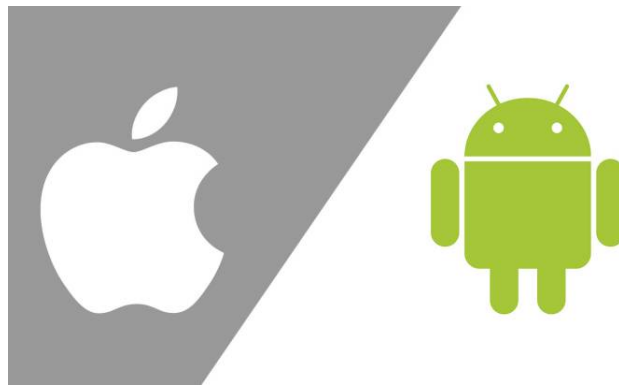


Ilustración 3: sistemas operativos

(Cesar Otero, 2022)

A lo largo del presente texto, no se hace referencia al sistema operativo como lo ve o usa el usuario final, o como lo vende la mercadotecnia — el ambiente gráfico, los programas que se ejecutan en este, los lenguajes de programación en los cuales están desarrollados y en que más fácilmente se puede desarrollar para ellos, e incluso el conjunto básico de funciones que las bibliotecas base ofrecen son principalmente clientes del sistema operativo — se ejecutan sobre él, y ofrecen sus interfaces a los usuarios (incluidos, claro, los desarrolladores). La diferencia en el uso son solo — cuando mucho — consecuencias del diseño de un sistema operativo (Gunnar Wolf, Estaban Ruiz, Federico Bergero, Erwin Meza, 2015.)

1. Funciones y objetivos del sistema operativo

El sistema operativo es el único programa que interactúa directamente con el hardware de la computadora. Sus funciones primarias son:

- a. Abstracción: Los programas no deben tener que preocuparse de los detalles de acceso a hardware, o de la configuración particular de una computadora. El sistema operativo se encarga de proporcionar una serie de abstracciones para que los programadores puedan enfocarse en resolver las necesidades particulares de sus usuarios. Un ejemplo de tales abstracciones es que la información está organizada en archivos y directorios (en uno o muchos dispositivos de almacenamiento).
- b. Administración de recursos: Un sistema de cómputo puede tener a su disposición una gran cantidad de recursos (memoria, espacio de almacenamiento, tiempo de procesamiento, etc.), y los diferentes procesos que se ejecuten en él compiten por ellos. Al gestionar toda la asignación de recursos, el sistema operativo puede implementar políticas que los asignen de forma efectiva y acorde a las necesidades establecidas para dicho sistema.
- c. Aislamiento: En un sistema multiusuario y multitarea, cada proceso y cada usuario no tendrá que preocuparse por otros que estén usando el mismo sistema —Idealmente, su experiencia será la misma que si el sistema estuviera exclusivamente dedicado a su atención (aunque fuera un sistema menos poderoso). Para implementar correctamente las funciones de aislamiento hace falta que el sistema operativo utilice hardware específico para dicha protección.

(Gunnar Wolf, Estaban Ruiz, Federico Bergero, Erwin Meza, 2015.)

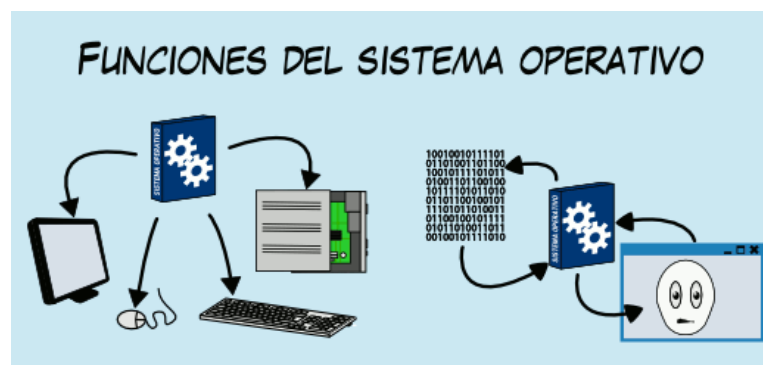


Ilustración 4: Funciones de sistema operativo

(CC-BY-SA)

2. Sistemas operativos para teléfonos inteligentes

Actualmente, los principales sistemas operativos para teléfonos inteligentes son:

a. iOS

El sistema operativo de Apple, y diseñado exclusivamente para el hardware producido por dicha compañía. Fue el primero en implementar la interfaz usuario multitáctil y, en buena medida, se puede ver como el responsable de la explosión y universalización en el uso de dispositivos móviles. Al igual que el sistema operativo que emplean para sus equipos de escritorio, Mac OS X, iOS está basado en el núcleo Darwin, derivado de FreeBSD, un sistema libre tipo Unix (Gunnar Wolf, Estaban Ruiz, Federico Bergero, Erwin Meza, 2015.)



(Apple In., 2017)

b. Android

Diseñado por la compañía Google, basa la mayor parte de su operación en software libre (un núcleo Linux, máquina virtual Java, y muchas de las bibliotecas de sistema comunes en sistemas Linux), agregando una capa de servicios propietarios. La estrategia de Google ha sido inversa a la de Apple: en vez de fabricar sus propios dispositivos, otorga licencias para el uso de este sistema operativo a prácticamente todos los fabricantes de hardware, con lo que la amplia mayoría de los modelos de teléfonos inteligentes y tabletas corren sobre Android (Gunnar Wolf, Estaban Ruiz, Federico Bergero, Erwin Meza, 2015.)



(Google LLC, 2019)

El sistema permite programar aplicaciones en una variación de Java. El sistema operativo proporciona todas las interfaces necesarias para desarrollar aplicaciones que accedan a las funciones del teléfono (como el GPS, las llamadas, la agenda, etc.) de una forma muy sencilla en un lenguaje de programación muy conocido como es Java.

Esta sencillez, junto a la existencia de herramientas de programación gratuitas, hace que una de las cosas más importantes de este sistema operativo sea la cantidad de aplicaciones disponibles, que extienden casi sin límites la experiencia del usuario (Alejandro Nieto Gonzalez, 2011).

Dentro de la propia arquitectura del sistema, podemos destacar los principales componentes de Android:

Núcleo Linux: El núcleo del sistema es Linux y actúa como una capa de abstracción entre el hardware del dispositivo y las aplicaciones instaladas. Además, el sistema operativo de Google depende de Linux para otros servicios básicos como la seguridad, gestión de memoria, gestión de procesos, pila de red o controladores.

Runtime: El sistema operativo de Google para dispositivos móviles incluye un conjunto de bibliotecas que proporcionan la mayor parte de las funciones disponibles en las bibliotecas base del lenguaje de programación Java. Cada aplicación Android corre su propio proceso con su instancia a la máquina virtual Dalvik. Esta máquina ejecutaba hasta la versión 5.0 archivos en formato .dex, pero a partir de esa versión se utiliza el ART, que compila totalmente al momento de instalación de la aplicación.

Bibliotecas: El sistema operativo Android incluye un conjunto de bibliotecas de C o C++ que son utilizadas por varios componentes del sistema. Estas características se exponen a los desarrolladores a través del marco de las aplicaciones de Android. Entre estas bibliotecas, caben destacar System C, bibliotecas de medios, de gráficos 3D o SQLite, entre otras.

Marco del trabajo de aplicaciones: El entorno de Google permite que los desarrolladores tengan acceso a las mismas API del entorno de trabajo utilizadas por las aplicaciones base. Y es que la arquitectura de Android está diseñada para simplificar la reutilización de componentes. Es decir, cualquier aplicación puede publicar sus capacidades y que otras aplicaciones puedan reutilizarlas dentro de unas reglas de seguridad.

Aplicaciones: Android cuenta con ciertas aplicaciones base que permiten el uso de las funciones básicas de un dispositivo como son, correo electrónico, mensajes de texto, SMS, calendario, mapas, navegador, contactos y otros. Aplicaciones desarrolladas en lenguaje Java (Roberto Adeva, 2022).

F. Aplicaciones móviles

En la era de la información e inmediatez en la que nos encontramos las organizaciones dependen en gran medida de las aplicaciones tecnológicas para llevar a cabo tareas cotidianas como enviar mails, revisar estados de cuentas, recibir y enviar informes, entre otras actividades que en la actualidad son realizadas a través de aplicaciones para dispositivos móviles.

Las aplicaciones móviles han ido incursionando a partir de inserción de lo que se denomina Smartphone a nuestra cotidianidad, donde cada vez es más frecuente su utilización para comunicarnos y ejecutar tareas de distinta índole. Para entender que es una aplicación móvil, se puede decir que las aplicaciones móviles son programas diseñados para ejecutarse en teléfonos, tablets y otros dispositivos móviles, y que, permiten al usuario

realizar actividades profesionales, acceder a servicios, mantenerse informado, entre otras posibilidades y usos, según sea el fin para el cual se desarrollan.

En la actualidad se han desarrollado todo tipo de aplicaciones con distintos fines y objetivos que van más allá de las prestaciones de los primeros teléfonos y en la actualidad se centran en la productividad personal, profesional y empresarial, ya que las apps se integran a los distintos sistemas informáticos de las organizaciones que permiten acceder e interactuar con los datos e información de los usuarios a través de las aplicaciones móviles.

Se han desarrollado tres tipos de aplicaciones las cuales son: app nativa, aplicaciones web y aplicaciones híbridas, son el resultado de la combinación entre las aplicaciones nativas y las webs apps que se desarrollan usando tecnologías web tales como: HTML, CSS, JavaScript que se compila y empaqueta de tal forma que el resultado final es una app para dispositivos móviles (Galo Puetate, José Luis Ibarra).

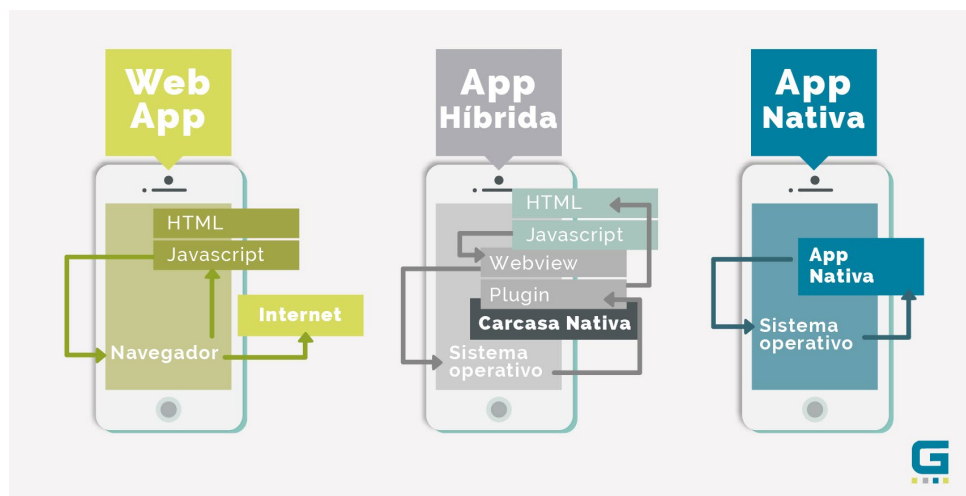


Ilustración 7: Tipos de aplicaciones móviles

(Gsoft informática, 2019).

G. Aplicaciones nativas

Las Apps nativas son aquellas aplicaciones que están desarrolladas para un equipo o plataforma determinada. Es decir, funciona en el equipo sin necesidad de ningún programa externo, ya que se ha desarrollado en el lenguaje de programación específico de

cada equipo. El término de App Nativa está habitualmente asociado a los dispositivos móviles y, por tanto, hay Apps Nativas para cada sistema operativo como iOS o Android.

La principal ventaja de las Apps Nativas es que se adapta al 100% al dispositivo y puede utilizar sin mayores problemas todas las funcionalidades del mismo (Rendimiento gráfico, GPS, cámara, acelerómetro).

El principal inconveniente es, si se quiere disponer de la aplicación para los diferentes sistemas, hay que duplicar el proyecto una vez por cada sistema en el que se quiera disponer de la app (iOS, Android) y muchas veces también con la versión del mismo. Eso aumenta mucho tanto los tiempos de desarrollo como los costes del mismo.

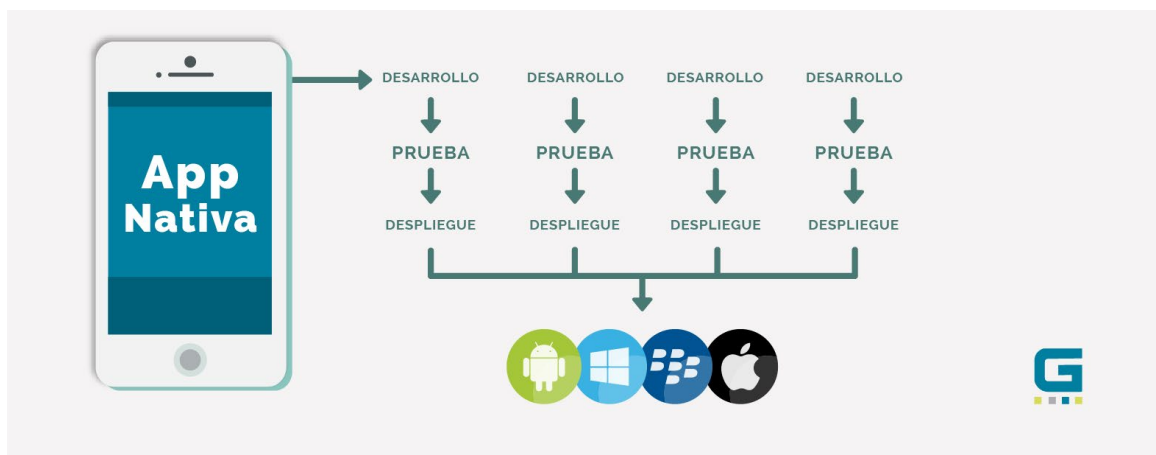


Ilustración 8: Flujo de desarrollo en apps nativas (Gsoft informática, 2019).

H. Software development kit (SDK)

Es un conjunto de herramientas que ayudan a la programación de aplicaciones para un entorno tecnológico particular.

Es decir, las aplicaciones desarrolladas sobre el SDK estarán destinadas a algún sistema operativo, plataforma hardware, consola de videojuegos o paquete de software en especial. Son muchos los recursos que puede contener un SDK (Marianna Rolfo, 2022).

1. Recursos de SDK

- Una interfaz de programación de aplicaciones (API): Puede verse como una abstracción del funcionamiento interno del entorno sobre el que vamos a trabajar. Se trata de un conjunto de funciones, rutinas, estructuras de datos, clases y variables que nos permiten manipular el mecanismo de la plataforma sin conocerlo internamente.
- Un entorno de desarrollo integrado (IDE): Un editor que nos ayuda a escribir fácilmente el código fuente del programa. Generalmente, también brinda una interfaz amigable para dos aplicaciones fundamentales:
 - Debugger: Permite testear el programa en cada paso de su ejecución.
 - Compilador: Traduce el código fuente a lenguaje de máquina, obteniendo así un programa ejecutable.
- Código de ejemplo y otra documentación: Como punto de partida para empezar a desarrollar aplicaciones.
- Un emulador del entorno: Por ejemplo, si desarrollamos una aplicación para móviles desde una computadora de escritorio, nos permite saber cómo la vería el usuario final.

I. Android SDK

Android SDK (Software Development Kit) es un conjunto de herramientas de desarrollo que se utilizan para desarrollar aplicaciones para la plataforma Android. Este SDK proporciona una selección de herramientas necesarias para crear aplicaciones de Android y garantiza que el proceso se desarrolle de la forma más fluida posible.

El SDK de Android es una colección de herramientas de desarrollo de software y bibliotecas necesarias para desarrollar aplicaciones de Android. Cada vez que Google lanza una nueva versión de Android o una actualización, también se lanza un SDK correspondiente que los desarrolladores deben descargar e instalar.

El JDK (Java Development Kit) es un SDK (Software Dev Kit). Se utiliza para crear software / aplicaciones en Java y, por supuesto, incluye JRE (Java Runtime Edition)

para ejecutar ese software. Si solo desea ejecutar una aplicación Java, descargue solo el JRE (CompuHoy).

J. Lenguaje de programación java

Los lenguajes de programación son idiomas artificiales diseñados para expresar cálculos y procesos que serán llevados a cabo por ordenadores. Un lenguaje de programación está formado por un conjunto de palabras reservadas, símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones. El proceso de programación consiste en la escritura, compilación y verificación del código fuente de un programa.

Java es un lenguaje de programación desarrollado por Sun Microsystems. Fue presentado en la segunda mitad del año 1995 y desde entonces se ha convertido en un lenguaje de programación muy popular. Es un lenguaje muy valorado porque los programas Java se pueden ejecutar en diversas plataformas con sistemas operativos como Windows, Mac OS, Linux o Solaris. James Gosling, el director del equipo de trabajo encargado de desarrollar Java, hizo realidad la promesa de un lenguaje independiente de la plataforma. Se buscaba diseñar un lenguaje que permitiera programar una aplicación una sola vez que luego pudiera ejecutarse en distintas máquinas y sistemas operativos. Para conseguir la portabilidad de los programas Java se utiliza un entorno de ejecución para los programas compilados. Este entorno se denomina Java Runtime Environment (JRE). Es

gratuito y está disponible para los principales sistemas operativos. Esto asegura que el mismo programa Java pueda ejecutarse en Windows, Mac OS, Linux o Solaris.



Ilustración 9: Portabilidad de java

(Jorge Martínez Ladrón de Guevara).

Los programas Java se compilan a un lenguaje intermedio, denominado Bytecode. Este código es interpretado por la máquina virtual de Java del entorno de ejecución (JRE) y así se consigue la portabilidad en distintas plataformas. El JRE es una pieza intermedia entre el código Bytecode y los distintos sistemas operativos existentes en el mercado. Un programa Java compilado en Bytecode se puede ejecutar en sistemas operativos como Windows, Linux, Mac OS, Solaris, BlackBerry OS, iOS o Android utilizando el entorno de ejecución de Java (JRE) apropiado.

Una de las características más importantes de los lenguajes de programación modernos es la portabilidad. Como se ha comentado antes, un programa es portable cuando es independiente de la plataforma y puede ejecutarse en cualquier sistema operativo y dispositivo físico.

La evolución del lenguaje de programación Java ha sido muy rápida. La plataforma de desarrollo de Java, denominada Java Development Kit (JDK), se ha ido ampliando y cada vez incorpora a un número mayor de programadores en todo el mundo. En realidad, Java no solo es un lenguaje de programación. Java es un lenguaje, una plataforma de desarrollo, un entorno de ejecución y un conjunto de librerías para desarrollo de programas sofisticados. Las librerías para desarrollo se denominan Java Application Programming Interface (Java API).

El siguiente esquema muestra los elementos de la plataforma Java, desde el código fuente, el compilador, la API de Java, los programas compilados en Bytecode y el entorno de ejecución de Java. Este entorno de ejecución (JRE) y la máquina virtual (JVM) permiten que un programa compilado Java se ejecute en distintos sistemas operativos (Jorge Martínez Ladrón de Guevara).

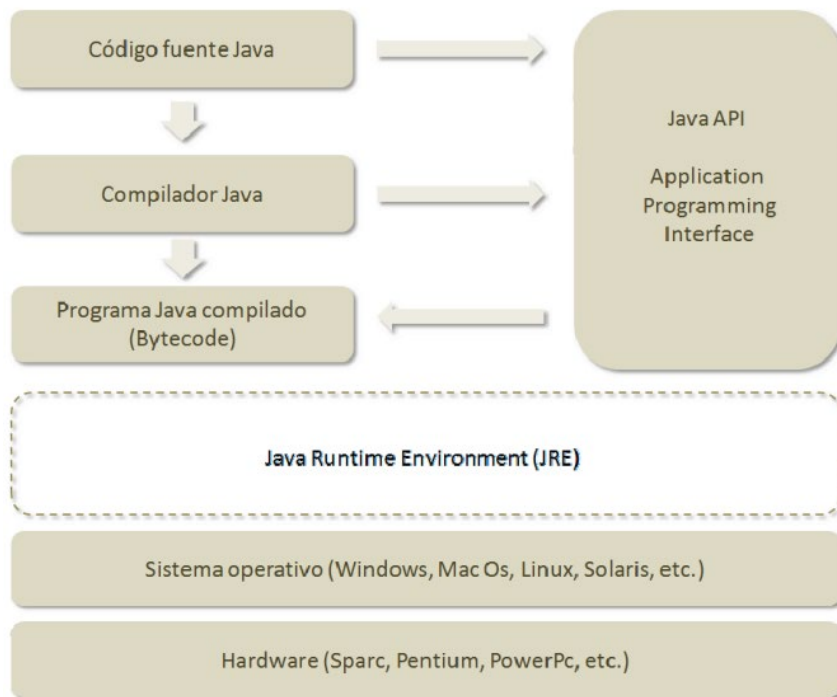


Ilustración 10: Esquema elementos de plataforma java (Jorge Martínez Ladrón de Guevara).

K. Entorno de desarrollo integrado (IDE)

Un entorno de desarrollo integrado (IDE) es un sistema de software para el diseño de aplicaciones que combina herramientas comunes para desarrolladores en una sola interfaz de usuario gráfica (GUI). Generalmente, un IDE cuenta con las siguientes características:

- Editor de código fuente: editor de texto que ayuda a escribir el código de software con funciones como el resaltado de la sintaxis con indicaciones visuales, el relleno automático específico para el lenguaje y la comprobación de errores a medida que se escribe el código.
- Automatización de compilaciones locales: herramientas que automatizan tareas sencillas y repetitivas como parte de la creación de una compilación local del software para su uso por parte del desarrollador, como la compilación del código fuente de la computadora en un código binario, el empaquetado de ese código y la ejecución de pruebas automatizadas.
- Depurador: programa que sirve para probar otros programas y mostrar la ubicación de un error en el código original de forma gráfica.

(Red Hat, 2019)

L. Android Studio

Android Studio es el entorno de desarrollo integrado (IDE) oficial para el desarrollo de apps para Android y se basa en IntelliJ IDEA. Además del potente editor de códigos y las herramientas para desarrolladores de IntelliJ, Android Studio ofrece incluso más funciones que aumentan tu productividad cuando desarrollas apps para Android, como las siguientes:

- Un sistema de compilación flexible basado en Gradle.
- Un emulador rápido y cargado de funciones.

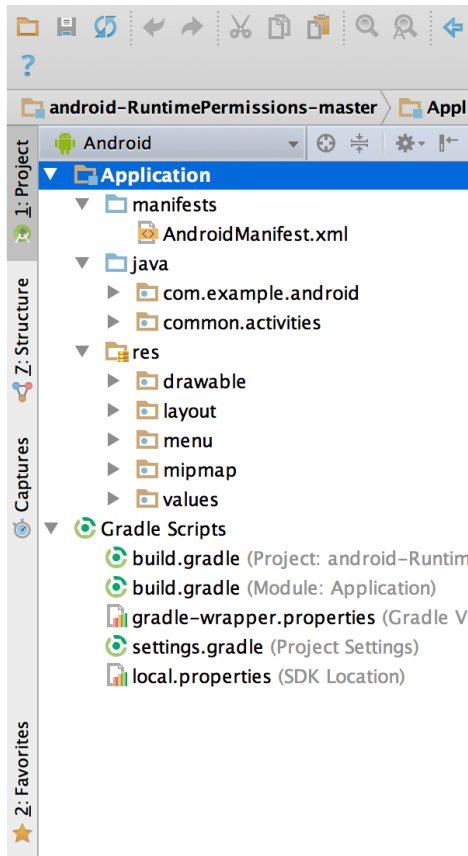
- Un entorno unificado donde puedes desarrollar para todos los dispositivos Android.
- Aplicación de cambios para insertar cambios de código y recursos a la app en ejecución sin reiniciarla.
- Integración con GitHub y plantillas de código para ayudarte a compilar funciones de apps comunes y también importar código de muestra.
- Variedad de marcos de trabajo y herramientas de prueba.
- Herramientas de Lint para identificar problemas de rendimiento, usabilidad y compatibilidad de versiones, entre otros.
- Compatibilidad con C++ y NDK.
- Compatibilidad integrada con Google Cloud Platform, que facilita la integración con Google Cloud Messaging y App Engine

1. Estructura del proyecto

Cada proyecto de Android Studio incluye uno o más módulos con archivos de código fuente y archivos de recursos. Entre los tipos de módulos se incluyen los siguientes:

- Módulos de apps para Android.
- Módulos de biblioteca
- Módulos de Google App Engine

De manera predeterminada, Android Studio muestra los archivos de tu proyecto en la vista de proyecto de Android, como se ve en la Ilustración 11. Esta vista está organizada en módulos para que puedas acceder rápidamente a los archivos fuente clave de tu proyecto (Developers Android, 2022).



*Ilustración 11: Los archivos de proyecto en la vista de Android
(Developers Android, 2022)*

Puedes ver todos los archivos de compilación en el nivel superior de Secuencias de comando de Gradle y cada módulo de app contiene las siguientes carpetas:

- manifests: contiene el archivo AndroidManifest.xml.
- java: contiene los archivos de código fuente Java, incluido el código de prueba de JUnit.
- res: contiene todos los recursos sin código, como diseños XML, strings de IU e imágenes de mapa de bits.

M. Base de datos

Una base de datos es un conjunto de datos almacenados en memoria externa que están organizados mediante una estructura de datos. Cada base de datos ha sido diseñada

para satisfacer los requisitos de información de una empresa u otro tipo de organización, como por ejemplo, una universidad o un hospital.

Antes de existir las bases de datos se trabajaba con sistemas de ficheros. Los sistemas de ficheros surgieron al informatizar el manejo de los archivadores manuales para proporcionar un acceso más eficiente a los datos almacenados en los mismos. Un sistema de ficheros sigue un modelo descentralizado, en el que cada departamento de la empresa almacena y gestiona sus propios datos mediante una serie de programas de aplicación escritos especialmente para él. Estos programas son totalmente independientes entre un departamento y otro, y se utilizan para introducir datos, mantener los ficheros y generar los informes que cada departamento necesita. Es importante destacar que en los sistemas de ficheros, tanto la estructura física de los ficheros de datos como la de sus registros, están definidas dentro de los programas de aplicación.

Cuando en una empresa se trabaja con un sistema de ficheros, los departamentos no comparten información ni aplicaciones, por lo que los datos comunes deben estar duplicados en cada uno de ellos. Esto puede originar inconsistencias en los datos. Se produce una inconsistencia cuando copias de los mismos datos no coinciden: dos copias del domicilio de un cliente pueden no coincidir si solo uno de los departamentos que lo almacenan ha sido informado de que el domicilio ha cambiado.

Otro inconveniente que plantean los sistemas de ficheros es que cuando los datos se separan en distintos ficheros, es más complicado acceder a ellos, ya que el programador de aplicaciones debe sincronizar el procesamiento de los distintos ficheros implicados para garantizar que se extraen los datos correctos. Además, ya que la estructura física de los datos se encuentra especificada en los programas de aplicación, cualquier cambio en dicha estructura es difícil de realizar. El programador debe identificar todos los programas afectados por el cambio, modificarlos y volverlos a probar, lo que cuesta mucho tiempo y está sujeto a que se produzcan errores. A este problema, tan característico de los sistemas de ficheros, se le denomina también falta de independencia de datos lógica-física.

Una base de datos se puede percibir como un gran almacén de datos que se define y se crea una sola vez, y que se utiliza al mismo tiempo por distintos usuarios. En una base de datos todos los datos se integran con una mínima cantidad de duplicidad. De este modo, la base de datos no pertenece a un solo departamento, sino que se comparte por toda la organización. Además, la base de datos no solo contiene los datos de la organización, también almacena una descripción de dichos datos. Esta descripción es lo que se denomina metadatos, se almacena en el diccionario de datos o catálogo y es lo que permite que exista independencia de datos lógica-física (Mercedes Marqués, 2011).

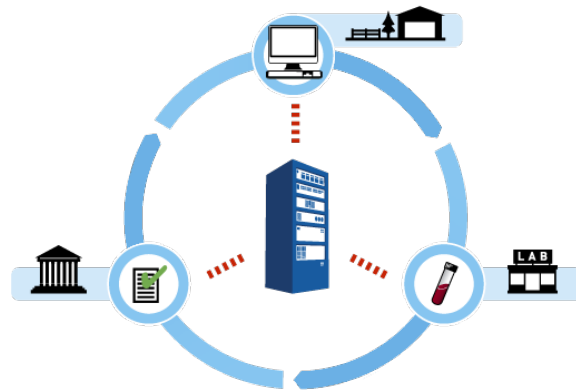


Ilustración 12: Base de datos en una organización

(Cristian J., 2019)

1. Sistema de gestión de base de datos

El sistema de gestión de la base de datos (en adelante SGBD) es una aplicación que permite a los usuarios definir, crear y mantener la base de datos, además de proporcionar un acceso controlado a la misma. Se denomina sistema de bases de datos al conjunto formado por la base de datos, el SGBD y los programas de aplicación que dan servicio a la empresa u organización.



Ilustración 13: SGBD actuales

(Diarlu, 2019)

El modelo seguido con los sistemas de bases de datos es muy similar al modelo que se sigue en la actualidad para el desarrollo de programas con lenguajes orientados a objetos, en donde se da una implementación interna de un objeto y una especificación externa separada. Los usuarios del objeto solo ven la especificación externa y no se deben preocupar de cómo se implementa internamente el objeto. Una ventaja de este modelo, conocido como abstracción de datos, es que se puede cambiar la implementación interna de un objeto sin afectar a sus usuarios, ya que la especificación externa no se ve alterada. Del mismo modo, los sistemas de bases de datos separan la definición de la estructura física de los datos de su estructura lógica, y almacenan esta definición en la base de datos. Todo esto es gracias a la existencia del SGBD, que se sitúa entre la base de datos y los programas de aplicación.

Generalmente, un SGBD proporciona los servicios que se citan a continuación:

- El SGBD permite la definición de la base de datos mediante un lenguaje de definición de datos. Este lenguaje permite especificar la estructura y el tipo de los datos, así como las restricciones sobre los datos.
- El SGBD permite la inserción, actualización, eliminación y consulta de datos mediante un lenguaje de manejo de datos
- El SGBD proporciona un acceso controlado a la base de datos mediante:

- Un sistema de seguridad, de modo que los usuarios no autorizados no puedan acceder a la base de datos.
- Un sistema de integridad que mantiene la integridad y la consistencia de los datos.
- Un sistema de control de concurrencia que permite el acceso compartido a la base de datos.
- Un sistema de control de recuperación que restablece la base de datos después de que se produzca un fallo del hardware o del software.
- Un diccionario de datos o catálogo, accesible por el usuario, que contiene la descripción de los datos de la base de datos.

(Mercedes Marqués, 2011).

N. Modelo de datos

Una de las características fundamentales de los sistemas de bases de datos es que proporcionan cierto nivel de abstracción de datos, al ocultar las características sobre el almacenamiento físico que la mayoría de usuarios no necesita conocer. Los modelos de datos son el instrumento principal para ofrecer dicha abstracción a través de su jerarquía de niveles. Un modelo de datos es un conjunto de conceptos que sirven para describir la estructura de una base de datos, es decir, los datos, las relaciones entre los datos y las restricciones que deben cumplirse sobre los datos. Los modelos de datos contienen también un conjunto de operaciones básicas para la realización de consultas (lecturas) y actualizaciones de datos. Además, los modelos de datos más modernos incluyen mecanismos para especificar acciones compensatorias o adicionales que se deben llevar a cabo ante las acciones habituales que se realizan sobre la base de datos.

Los modelos de datos se pueden clasificar dependiendo de los tipos de conceptos que ofrecen para describir la estructura de la base de datos, formando una jerarquía de niveles. Los modelos de datos de alto nivel, o modelos conceptuales, disponen de conceptos muy cercanos al modo en que la mayoría de los usuarios percibe los datos, mientras que los modelos de datos de bajo nivel, o modelos físicos, proporcionan conceptos que describen los detalles de cómo se almacenan los datos en el ordenador. Los conceptos de los modelos físicos están dirigidos al personal informático, no a los usuarios finales.

Entre estos dos extremos se encuentran los modelos lógicos, cuyos conceptos pueden ser entendidos por los usuarios finales, aunque no están demasiado alejados de la forma en que los datos se organizan físicamente. Los modelos lógicos ocultan algunos detalles de cómo se almacenan los datos, pero pueden implementarse de manera directa en un SGBD.

Los modelos conceptuales utilizan conceptos como entidades, atributos y relaciones. Una entidad representa un objeto o concepto del mundo real como, por ejemplo, un cliente de una empresa o una de sus facturas. Un atributo representa alguna propiedad de interés de una entidad como, por ejemplo, el nombre o el domicilio del cliente. Una relación describe una interacción entre dos o más entidades, por ejemplo, la relación que hay entre un cliente y las facturas que se le han realizado.

Los modelos físicos describen cómo se almacenan los datos en el ordenador: el formato de los registros, la estructura de los ficheros (desordenados, ordenados, agrupados) y los métodos de acceso utilizados (índices, tablas de dispersión).

A la descripción de una base de datos mediante un modelo de datos se le denomina esquema de la base de datos. Este esquema se especifica durante el diseño, y no es de esperar que se modifique a menudo. Sin embargo, los datos que se almacenan en la base de datos pueden cambiar con mucha frecuencia: se insertan datos, se actualizan, se borran, etc. (Mercedes Marqués, 2011)

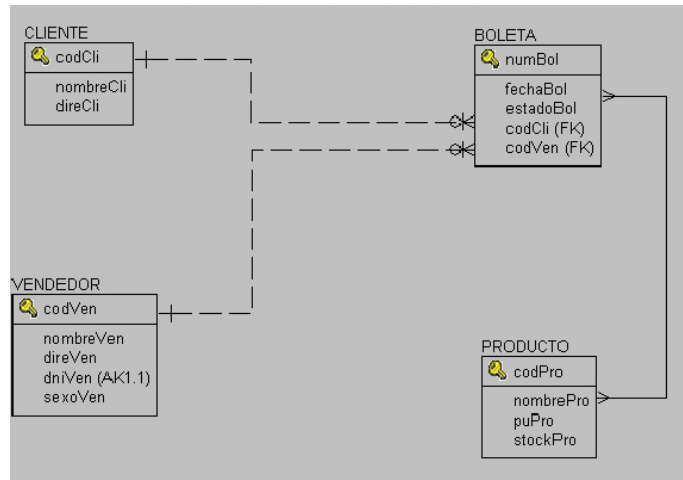


Ilustración 14: Modelo de datos de alto nivel

(Ray Sánchez, 2016)

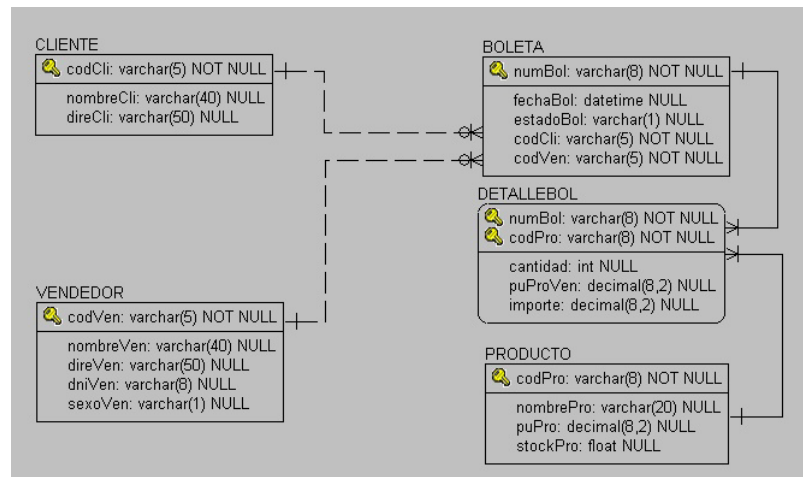


Ilustración 15: Modelo de datos de bajo nivel

(Ray Sánchez, 2016)

O. Patrón de diseño MVC

MVC son las siglas de modelo-vista-controlador, uno de los patrones más utilizados por ser la base sobre la que se asientan distintos gestores de contenido.

Para entender lo que significa debemos empezar por explicar qué es un patrón arquitectónico: con este término nos referimos a la forma de organizar los componentes de un sistema aplicando ciertas normas de diseño para fortalecer su usabilidad y prepararlo para su evolución.

Se trata de la forma de organizar los componentes de un sistema para conseguir que la calidad del software sea satisfactoria, cumpliendo los siguientes atributos:

- Configurabilidad: que el usuario pueda realizar ciertos cambios en el sistema.
- Integrabilidad de los módulos independientes del sistema
- Integridad: que la información solo pueda ser modificada por quien esté autorizado y de manera controlada.
- Interoperabilidad con otros sistemas.
- Modificabilidad con vistas al futuro.
- Mantenibilidad sencilla.
- Portabilidad: que se pueda ejecutar en distintos software y hardware.
- Reusabilidad: que la estructura pueda ser utilizada en futuros sistemas.
- Escalabilidad: que el sistema se pueda ampliar.
- Testeabilidad: facilidad para ser sometido a pruebas que aseguren que el sistema falla cuando es lo que se espera.
- Confidencialidad: no se permite el acceso no autorizado.
- Cumplimiento de la funcionalidad requerida.
- Seguridad externa e interna: prevenir ataques de terceros o usos no autorizados.

(Código Nexa)

1. Estructura del patrón MVC

El patrón MVC es un patrón de arquitectura de software encargado de separar la lógica del negocio de la interfaz del usuario y es el más utilizado en aplicaciones Web, ya que facilita la funcionalidad, mantenibilidad y escalabilidad del sistema de forma sencilla, a la vez que permite no mezclar lenguajes de programación en el mismo código.

La programación de la aplicación se separa en tres componentes con sus respectivas responsabilidades: un modelo, una o varias vistas, y uno o varios controladores.

El modelo: El modelo implementa la lógica de la aplicación, es decir, almacena todos los datos y el estado de la aplicación y tiene los métodos que manipulan esos datos. Por ejemplo, en una aplicación bancaria el modelo lo forman aquellas clases que representan las cuentas, transacciones, etc. Es decir, los datos importantes para el dominio del problema, así como los métodos que permiten manipularlos. El modelo no es “consciente” de la vista y el controlador (no los referencia directamente).

La vista: La vista es la interfaz de usuario. Muestra al usuario una representación visual del modelo, sus datos y estado, tomándolos directamente del modelo. También contiene los elementos de la interfaz que permiten al usuario interactuar con el programa, tales como botones y menús. Sin embargo, no es tarea de la vista implementar cómo se deben comportar esos elementos.

El controlador: El controlador es el intermediario entre la vista y el modelo. Es quien controla las interacciones del usuario solicitando los datos al modelo y entregándolos a la vista para que esta lo presente al usuario (Código Nexa).

2. Funcionamiento del patrón MVC

El funcionamiento básico del patrón MVC es el siguiente:

- a. El usuario realiza una petición que captura el controlador.
- b. El controlador llama al modelo correspondiente.
- c. El modelo solicita la información a la base de datos.
- d. El modelo recoge la información de la base de datos.
- e. El controlador recibe la información
- f. El controlador procesa y envía la información a la vista
- g. La vista entrega al usuario la información de forma “humanamente legible”.

(Código Nexa)

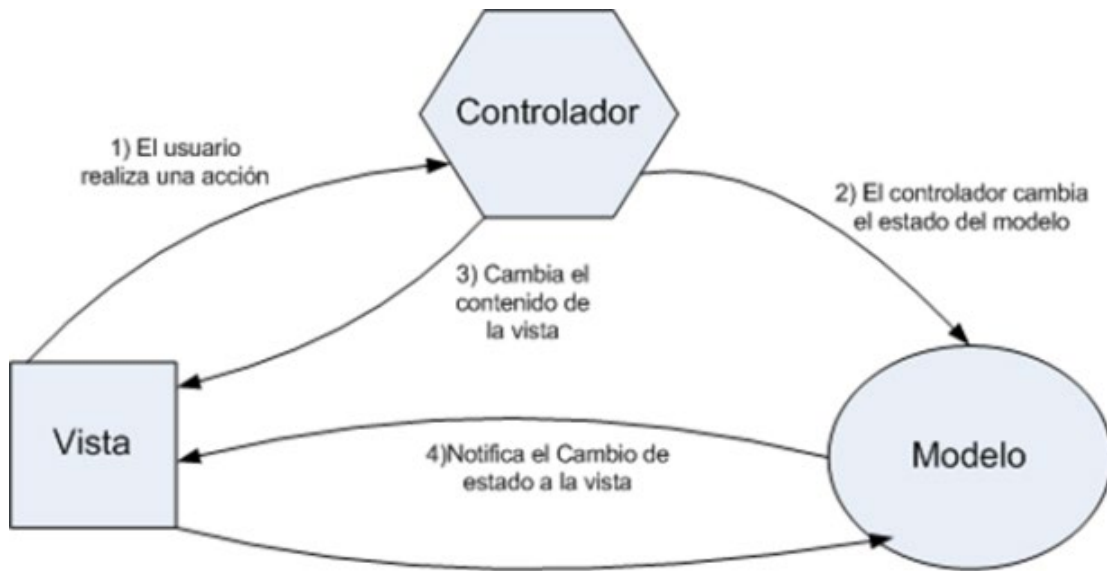


Ilustración 16: Ciclo de vida del patrón MVC

(codesystems, 2012)

V. METODOLOGÍA

A. Investigación

Para este proyecto es positivo comprender cómo la automatización aplicada a través de la tecnología, beneficia grandemente a la industria, para este caso en concreto una empresa agroindustrial del sector sur de Guatemala, en el proceso de captura de datos para el despacho de una gasolinera móvil, lo cual facilitara la generación de despachos a los distintos usuarios internos de la empresa.

La investigación consistió en comprender la funcionalidad de la gasolinera móvil, usuarios finales y departamento de combustible y lubricantes de la empresa, esto con el fin de diseñar una solución funcional y apegada a sus procesos actuales, con el fin de evitar que la propuesta sea disruptiva para los usuarios finales en sus procesos, se realizó reunión con el departamento de combustible y lubricantes de la empresa para toma de requerimientos, con los requerimientos se procedió a realizar la propuesta junto a un prototipo de la aplicación móvil.

B. Alcance

El alcance de la propuesta planteada es permitir al usuario debidamente registrado en la empresa, acceso a generar y consultar despachos de gasolina a las distintas unidades asignadas a la bomba de despacho, durante el periodo vigente de zafra (cosecha de caña) por medio de una aplicación móvil de sistema operativo Android.

C. Requerimientos

1. Requerimientos funcionales

No. de requerimiento funcional:	01
Nombre del requerimiento:	Autenticación de usuario activo en la empresa
Características:	El usuario deberá de identificarse para acceder a la funcionalidad de la aplicación, este debe estar activo.

Descripción del requerimiento:	El usuario debe estar previamente registrado por T.I. de la empresa, con una contraseña segura, la cual debe ingresar para acceder, el usuario tendrá que ser activo.
Prioridad del requerimiento:	Alta

Tabla 01 Requerimiento funcional autenticación de usuario activo en la empresa

No. de requerimiento funcional:	02
Nombre del requerimiento:	Formulario interactivo para ingreso de nuevos despachos de combustible.
Características:	Formulario interactivo para el ingreso de información necesario para el registro de una nueva orden de despacho a un cliente.
Descripción del requerimiento:	Formulario interactivo con campos necesarios para el registro de un nuevo despacho: No. bomba, empresa asociada a despachar, frente, vehículo a despachar, nombre del conductor, tipo de combustible, cantidad a despachar.
Prioridad del requerimiento:	Alta

Tabla 02 Requerimiento funcional formulario interactivo para ingreso de nuevos despachos de combustible.

No. de requerimiento funcional:	03
Nombre del requerimiento:	Reportes de despachos de combustibles
Características:	Realizar reportes de despachos de combustibles con distintos filtros de información.
Descripción del requerimiento:	Generación de reportes para consulta de órdenes de despachos realizadas con anterioridad, mostrando resultados según sus filtros: empresa asociada despachada, departamento, frente, vehículos, tipo de combustible, no. bomba.
Prioridad del requerimiento:	Alta

Tabla 03 Requerimiento funcional reportes de despachos de combustible

No. de requerimiento funcional:	04
Nombre del requerimiento:	Base de datos para el manejo local de los datos.
Características:	Utilizar una base de datos SQLite para el manejo estructurado de manera local de los datos.
Descripción del requerimiento:	Base de datos SQLite para guardar de forma estructurada datos, tales como: la información de usuarios autorizados, catálogos. Manteniendo una integridad.
Prioridad del requerimiento:	Alta

Tabla 04 Requerimiento funcional base de datos manejo local de información

No. de requerimiento funcional:	05
Nombre del requerimiento:	Sincronización de base de datos local con servidor.
Características:	Sincronizar la base de datos local cuando se detecte un servicio de conexión a internet con la base de datos de producción de la empresa, por medio de un micro servicio.
Descripción del requerimiento:	Sincronizar la base de datos local con la base de datos de la empresa en producción cuando se detecte un servicio de conexión a internet, con el fin de alimentar la base de datos con el envío de información de nuevos despachos de combustibles realizados y actualizar en la base de datos local los catálogos disponibles.
Prioridad del requerimiento:	Alta

Tabla 05 Requerimiento funcional, sincronización de base de datos, local con servidor.

2. Requerimientos no funcionales

No. de requerimiento no funcional:	01
Nombre del requerimiento:	Interfaz intuitiva del sistema.
Características:	La aplicación presenta una interfaz de usuario sencilla e intuitiva para el fácil manejo y fácil adaptabilidad a los usuarios de la misma.

Descripción del requerimiento:	La aplicación móvil presenta una interfaz intuitiva y de fácil uso, para la adaptabilidad temprana del usuario.
Prioridad del requerimiento:	Alta

Tabla 06 Requerimiento no funcional interfaz intuitiva del sistema

No. de requerimiento no funcional:	02
Nombre del requerimiento:	Disponibilidad
Características:	El usuario debe de hacer registros y reportes en cualquier momento determinado.
Descripción del requerimiento:	La aplicación debe estar disponible las 24 horas del día, para el registro de nuevos despachos y la consulta de reportes.
Prioridad del requerimiento:	Alta

Tabla 07 Requerimiento no funcional disponibilidad

No. de requerimiento no funcional:	03
Nombre del requerimiento:	Seguridad e integridad de datos
Características:	La aplicación garantizará a los usuarios la seguridad e integridad de datos a utilizar.
Descripción del requerimiento:	La aplicación garantizará que los datos estén íntegro en su totalidad con una base de datos local, aplicando seguridad sobre los mismos con la validación de usuario activo y previamente registrado.
Prioridad del requerimiento:	Alta

Tabla 08 Requerimiento no funcional seguridad e integridad de datos

No. de requerimiento no funcional:	04
Nombre del requerimiento:	Rendimiento y desempeño
Características:	La aplicación debe garantizar a los usuarios un desempeño adecuado y resultados en un tiempo prudente, según consultas y registros.

Descripción del requerimiento:	Garantizar el correcto desempeño de la aplicación, optimizando los procesos, lo cual nos ayuda a garantizar el rendimiento adecuado en un tiempo de respuesta de 5 a 15 segundos según el tipo de consulta.
Prioridad del requerimiento:	Alta

Tabla 09 Requerimiento no funcional rendimiento y desempeño.

D. Diagramas

1. Caso de uso

a. General

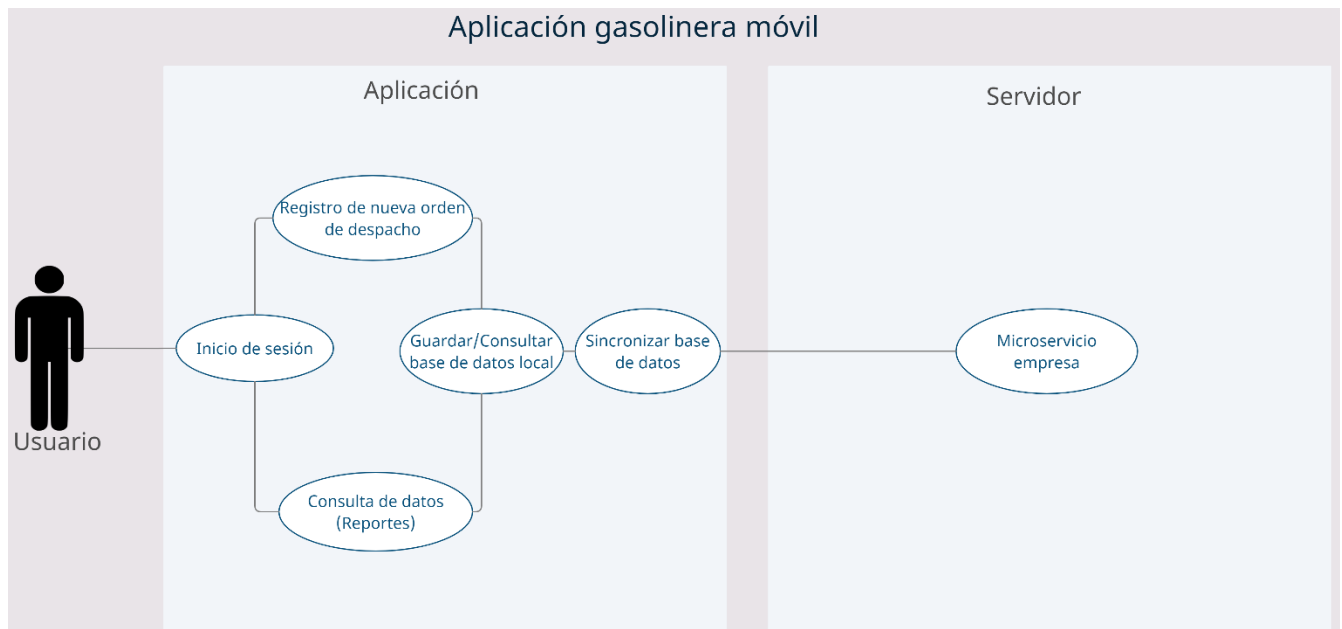


Ilustración 17 Diagrama de caso de uso general

Caso de uso general aplicación gasolinera móvil	
Responsable:	Usuario final.
Descripción:	El usuario final accede a las funciones de la aplicación por medio de sus credenciales: usuario y contraseña.
Actividades:	<ul style="list-style-type: none"> • Registro de un nuevo despacho de combustible. • Consulta de datos (reportes) con filtros de información. • Guardar o consultar datos en la base de datos local. • Sincronizar la base de datos local con un micro servicio de la empresa.
Visualizaciones:	<ul style="list-style-type: none"> • Formulario de registro de un nuevo despacho.

	<ul style="list-style-type: none"> • Ver la información de la consulta, según los filtros utilizados. • Botón de sincronización
--	---------------------------------------------------------------------------------------------------------------------------------------------------------

Tabla 10 Especificaciones diagrama caso de uso general

2. Entidad relación

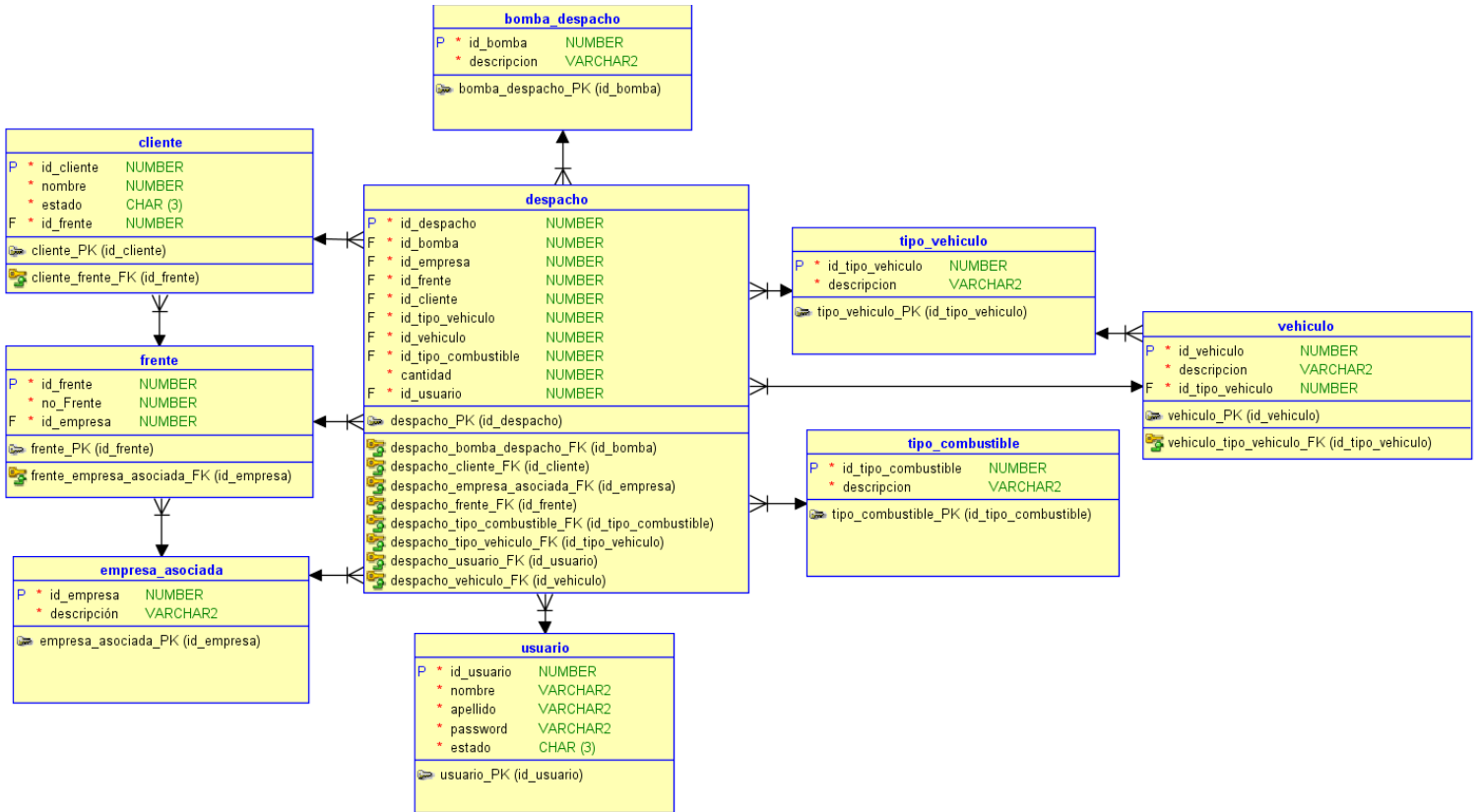


Ilustración 18 Diagrama entidad relación

E. Diseño de propuesta de interfaces.

1. Inicio de sesión.



Ilustración 19: Diseño de interfaz inicio de sesión.

2. Menú principal.



Ilustración 20: Diseño de interfaz menú principal.

3. Ingreso de nuevo despacho de combustible.

Ingreso nuevo despacho de combustible

Bomba
Seleccione ▼

Empresa **Frente**
Seleccione ▼ Seleccione ▼

Nombre conductor
Seleccione ▼

Tipo de vehículo **Vehículo**
Seleccione ▼ Seleccione ▼

Tipo de combustible **Cantidad**
Seleccione ▼ Ingrese cantidad

Ingresar despacho




Ilustración 21: Diseño de interfaz de nuevo despacho combustible.

4. Alerta de confirmación de datos para nuevo despacho.

Ingreso nuevo despacho de combustible

Bomba
Seleccione ▼

Empresa **Frente**
Seleccione ▼ Seleccione ▼

Nombre conductor
Seleccione ▼

Tipo de vehículo **Vehículo**
Seleccione ▼ Seleccione ▼

Tipo de combustible **Cantidad**
Seleccione ▼ Ingrese cantidad

Por favor revisa la información ingresada si es correcta confirma, caso contrario presiona el botón cancelar e ingresa los datos correctos.

Confirmar **Cancelar**

Ilustración 22 Diseño de interfaz de confirmar datos para nuevo despacho.

5. Notificación de transacción guardada correctamente.

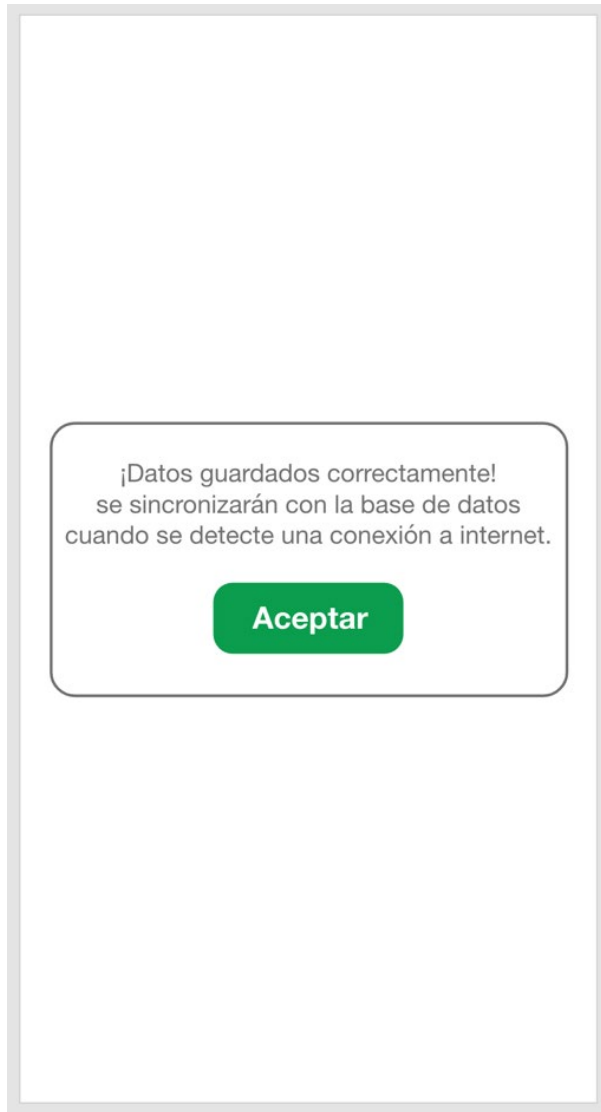


Ilustración 23: Diseño de interfaz de transacción guardada correctamente.

6. Menú de reportes de despacho de combustible.

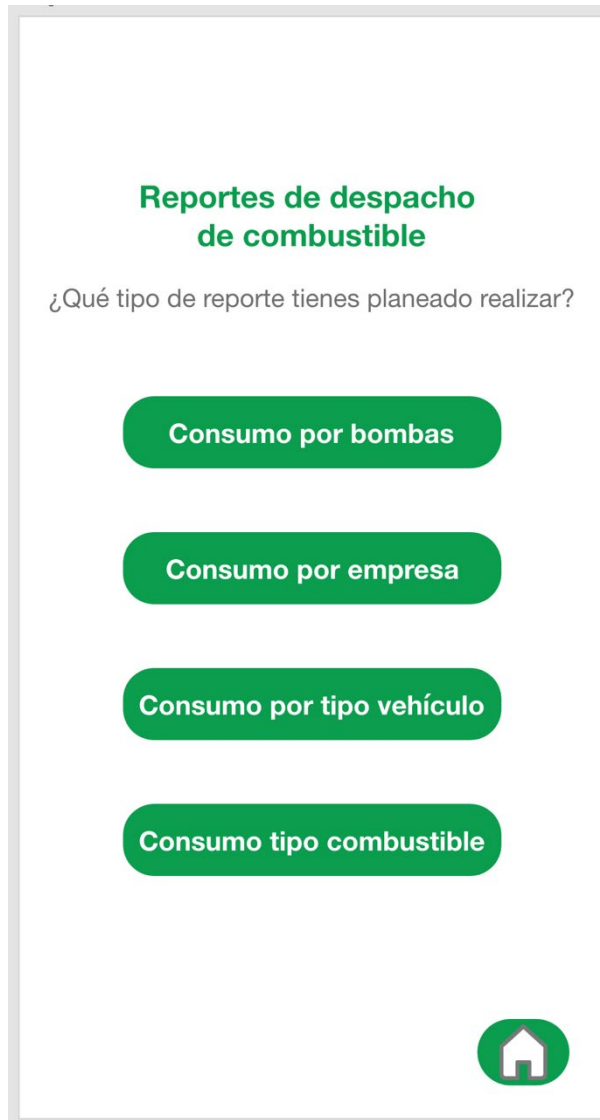


Ilustración 24: Diseño de interfaz de menú de reportes de despacho de combustible.

7. Pantalla para reporte despacho por bomba.

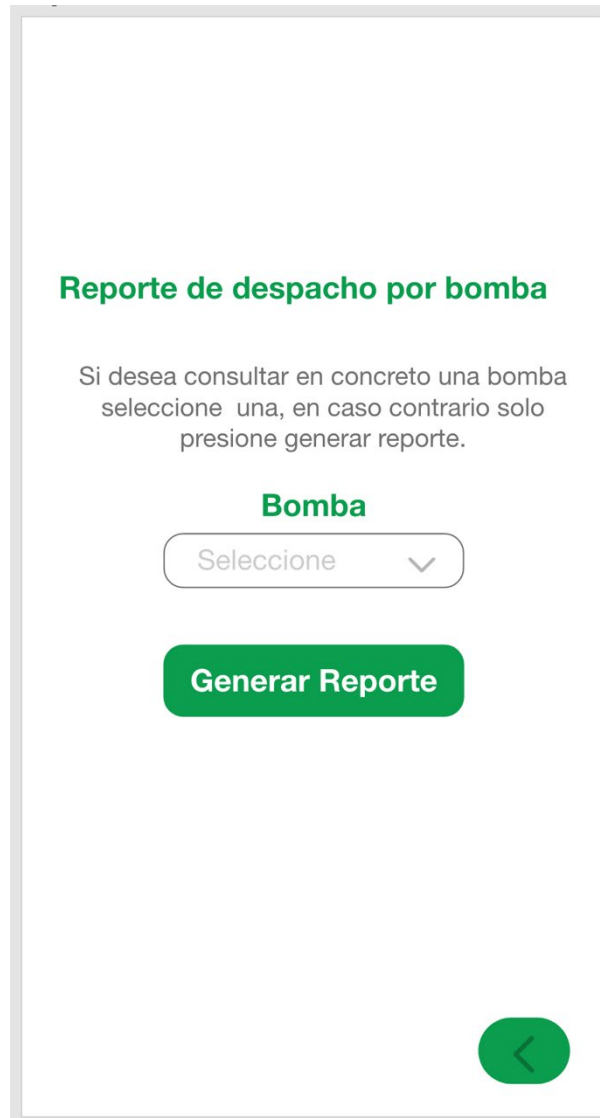


Ilustración 25: Diseño de interfaz de pantalla para reporte despacho por bomba.

8. Pantalla para reporte despacho por empresa.

Reporte de despacho por empresa

Si desea consultar en concreto una empresa seleccione una, en caso contrario solo presione generar reporte.

Empresa

Seleccione ▼

Generar Reporte

<

The image shows a mobile application interface for generating a report by company. It features a title 'Reporte de despacho por empresa' in green. Below the title is a paragraph of instructions: 'Si desea consultar en concreto una empresa seleccione una, en caso contrario solo presione generar reporte.' There is a dropdown menu labeled 'Empresa' with the placeholder text 'Seleccione' and a downward arrow. Below the dropdown is a prominent green button with the text 'Generar Reporte'. At the bottom right corner, there is a green circular button with a white left-pointing arrow, serving as a back navigation control.

Ilustración 26: Diseño de interfaz de pantalla para reporte despacho por empresa.

9. Pantalla para reporte despacho por tipo vehículo.

The screenshot shows a mobile application interface for generating a report. At the top, the title "Reporte de despacho por tipo vehículo" is displayed in green. Below the title, a paragraph of text reads: "Si desea consultar en concreto un tipo de vehículo seleccione una, en caso contrario solo presione generar reporte." Underneath this text, there is a section titled "Tipo vehículo" in green, followed by a dropdown menu with the text "Seleccione" and a downward arrow. Below the dropdown is a prominent green button with the text "Generar Reporte" in white. In the bottom right corner of the screen, there is a green circular button with a white left-pointing arrow.

Ilustración 27: Diseño de interfaz de pantalla para reporte despacho por tipo vehículo.

10. Pantalla para reporte despacho por tipo de combustible.

The screenshot shows a mobile application interface for generating a report. At the top, the title "Reporte de despacho por tipo de combustible" is displayed in green. Below the title, a paragraph of text reads: "Si desea consultar en concreto un tipo de Combustible seleccione una, en caso contrario solo presione generar reporte." Underneath this text, the label "Tipo combustible" is shown in green. Below the label is a dropdown menu with the text "Seleccione" and a downward arrow icon. A prominent green button with the text "Generar Reporte" is positioned below the dropdown. In the bottom right corner of the screen, there is a green circular button containing a white left-pointing arrow.

Ilustración 28: Diseño de interfaz para reporte despacho por tipo de combustible.

11. Pantalla de sincronizar base de datos manualmente.

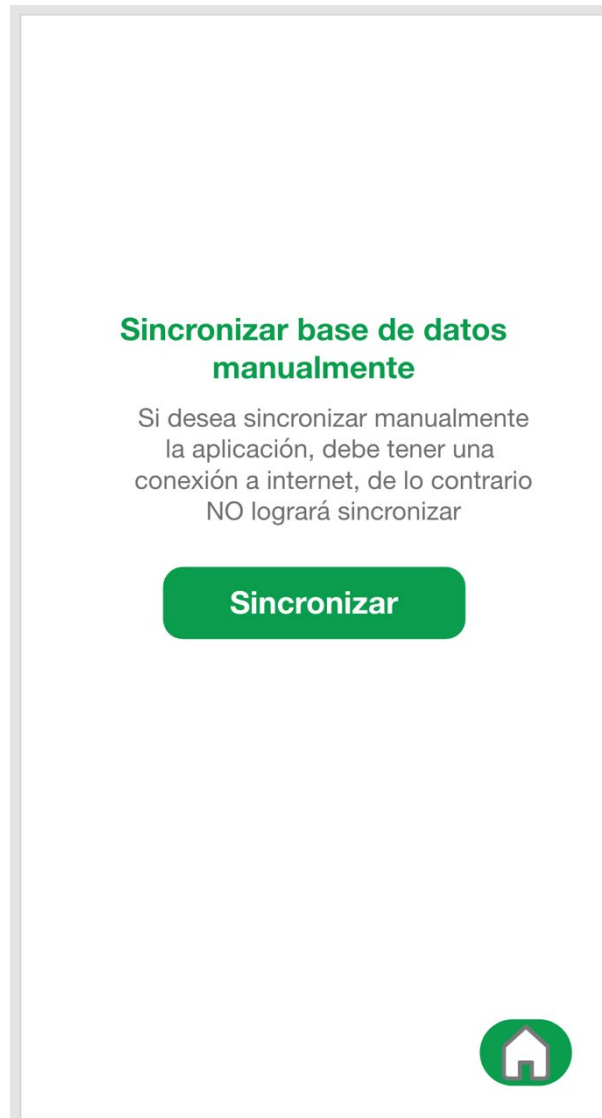


Ilustración 29: Diseño de interfaz de pantalla de sincronizar base de datos manualmente.

12. Alerta de sincronización correcta.

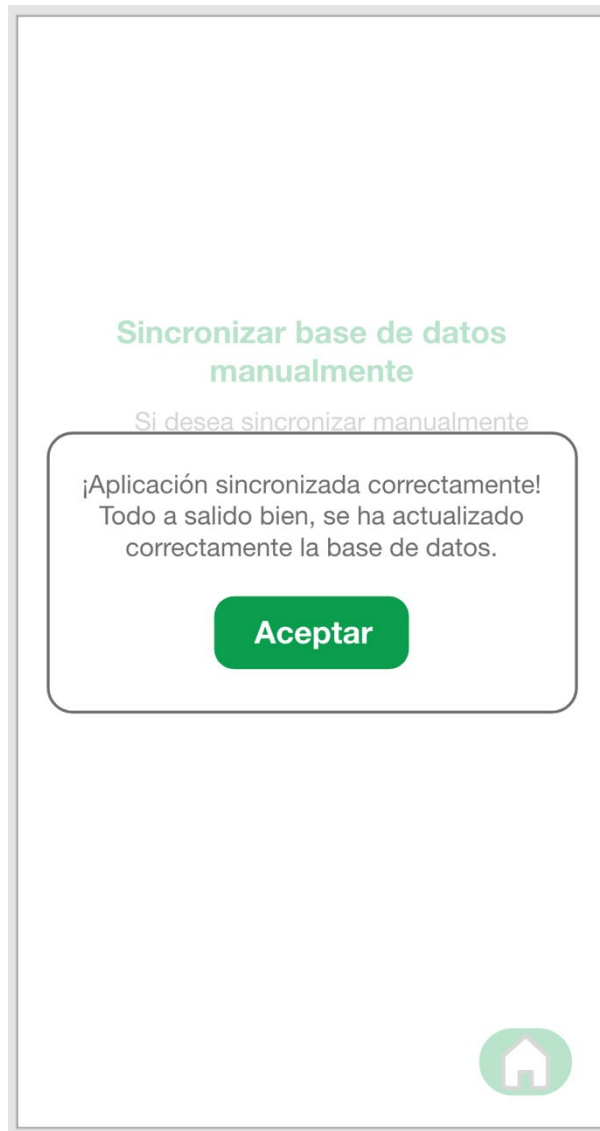


Ilustración 30: Diseño de interfaz de alerta de sincronización correcta.

13. Alerta de error en sincronización.

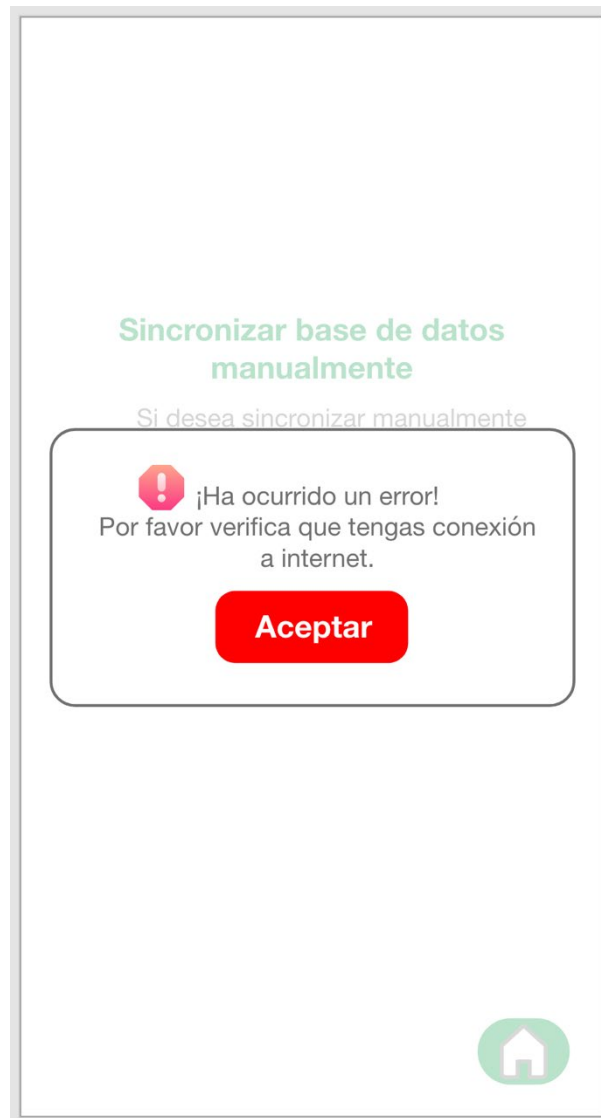


Ilustración 31: Diseño de interfaz de alerta de error en sincronización.

VI. RESULTADOS

Los resultados de la presente propuesta son los siguientes:

1. Prototipo de aplicación móvil Android con navegación dentro de las distintas pantallas. Una aplicación móvil debido a la accesibilidad y la practicidad de utilizar un teléfono móvil inteligente, con la finalidad de una aplicación sencilla y de fácil uso para el usuario final.
2. Acceso a la red del teléfono para realizar el uso de internet para la conexión entre la aplicación y servicio web de la empresa, con dos tipos de sincronización, automática o cuando el usuario final lo requiera sincronizar los datos. Lo cual logra obtener información actualizada y al día.
3. Se suprime el uso de papel para la generación de boletas impresas, con la propuesta de prototipo de aplicación móvil, al mismo tiempo se obtuvo una integridad de datos utilizando una base de datos en la aplicación localmente, en este caso SQLite, lo cual asegura la integridad y seguridad sobre los datos, con la autenticación por medio de una contraseña y un usuario.
4. Reducción de tiempo en un estimado de 5 horas diarias en la tabulación. Lográndolo con la tabulación directamente del expendedor, suprimiendo del proceso la tabulación de despachos en oficinas centrales, automatizando así mismo envío de información directamente desde la aplicación móvil al servicio web de la empresa.

VII. ANÁLISIS DE RESULTADOS

1. Las aplicaciones móviles sin duda alguna son de gran ayuda en la industria actual, ya que con un teléfono móvil inteligente se obtiene beneficios de una computadora en un dispositivo compacto, de menor tamaño, práctico y de menor dificultad de uso, integrando una aplicación móvil sencilla e intuitiva, es una solución viable y de no mucho costo.
2. La comunicación por medio de internet sin duda alguna es una gran herramienta de comunicación, la cual nos permite envío y recepción de datos de manera inmediata, nos permiten sincronizar datos de manera eficiente, se permite al usuario realizar una sincronización manual, por si en un caso en medio de la sincronización se realiza algún fallo en la señal.
3. Suprimir el uso de papel sin duda alguna es de gran ayuda en muchos aspectos, entre ellos se encuentra no la tala de árboles, reducción de costos que le conlleva a la empresa utilizar formatos físicos de formularios, ya que se evita costos de impresión, lapiceros, y hojas de papel, mitigando error humano en la tabulación de los datos físicos a digital, obteniendo una mayor integridad de datos y seguridad sobre los mismos.
4. La gestión de tiempos es de suma importancia en los procesos de las industrias, la reducción de los mismos es beneficioso sobre todo en procesos que conllevan información para la toma de decisiones, la reducción de 5 horas diarias en la tabulación de los datos en el proceso ayudara a obtener información actualizada.

VIII. CONCLUSIONES

1. Automatizar procesos con una aplicación móvil para dispositivos inteligentes con sistema operativo Android, es una solución idónea, ya que nos permite facilidad de acceso a ingreso y obtención de información de manera ágil y segura, para la propuesta descrita en este proyecto redujo en tiempo en el proceso de la generación de despachos de combustibles de gasolineras móviles, con un formulario dinámico en la información, de acuerdo con los datos que se estén seleccionando.
2. Transformar boletas de formato físico a digital por medio de la aplicación móvil, contribuyo a la reducción del 100% en el uso de papel, impresiones de boletas y la perdida de las boletas en su formato físico, logrando así tanto una mitigación de perdida de información y reduciendo el 100% el uso de papel para las boletas de despachos por un formulario dinámico.
3. Por medio de la propuesta de la aplicación móvil descrita, se logra suprimir el formato físico, por lo cual implica que directamente el expendedor tabule la información al sistema, logrando así la reducción de 8 horas a un estimado de 3 horas diarias el tiempo de digitación de boletas de despacho de combustible durante el periodo de zafra.
4. Se logró automatizar el 100% del envío de la información de despachos de combustibles, por medio de la propuesta de aplicación móvil, con acceso a la red, al identificar una conexión a internet se envía la información al servicio web de la empresa, para la actualización de su ERP.

IX. BIBLIOGRAFÍA

- Adeva, Roberto. (2022). ¿Qué es Android?: todo sobre el sistema operativo de Google. Obtenido de: <https://www.adslzone.net/reportajes/software/que-es-android/>
- Bergero, Federico; Meza Erwin, Ruiz Esteban. y Wolf, Gunnar. 2015. *Fundamentos de sistemas operativos*. Universidad Nacional Autónoma de México. URL: https://ru.iiec.unam.mx/2718/1/sistemas_operativos.pdf
- Cadenas, Ricardo. (2019). ¿Qué necesito? ¿Web apps, app nativa o app híbrida? Obtenido de: <https://www.gsoft.es/articulos/que-necesito-web-apps-app-nativa-o-app-hibrida/>
- Code Systems. Comprendiendo el patrón model, view, controller. Obtenido de: <https://codesystems.wordpress.com/2012/02/26/comprendiendo-el-patron-model-view-controller-mvc/>
- Código Nexo. Tutorial patrón MVC. Obtenido de: <https://www.codigonexo.com/wp-content/uploads/2014/06/Curso-completo-MVC.pdf>
- CompuHoy.com. ¿Qué se entiende por SDK de Android? Obtenido de: <https://www.compuhoy.com/que-se-entiende-por-sdk-de-android/>
- Developers Android. (2022). Introducción a Android Studio. Obtenido de: <https://developer.android.com/studio/intro?hl=es-419>
- Gonzalez, Alejandro Nieto. (2011). ¿Qué es Android? Obtenido de: <https://www.xatakandroid.com/sistema-operativo/que-es-android>

- Ibarra, José Luis; Puetate, Galo. 2020. *Aplicaciones móviles híbridas*. Centro de publicaciones PUCE. URL: <https://www.pucesi.edu.ec/webs2/wp-content/uploads/2021/02/Aplicaciones-M%C3%B3viles-H%C3%ADbridas-2020.pdf>
- Ladrón de Guevara, Jorge Martínez. (s.f.). *Fundamentos de programación en Java*. Editorial EME. URL: <https://www.tesuva.edu.co/phocadownloadpap/Fundamentos%20de%20programcion%20en%20Java.pdf>
- Marqués, Mercedes. 2011. *Base de datos*. Publicacions de la Universitat Jaume I. URL: <https://bdigital.uvhm.edu.mx/wp-content/uploads/2020/05/Bases-de-Datos.pdf>
- Red Hat. (2019). El concepto de IDE. Obtenido de: <https://www.redhat.com/es/topics/middleware/what-is-ide>
- Rolfo, M. (2022). SDK ¿Cuál es su importancia? Obtenido de: <https://codigoencasa.com/como-crear-tu-propio-sd-vs-api/>

X. ANEXOS

A. Versiones de Android recomendadas.

Versiones de Android recomendadas	
Android 10:	Es el décimo lanzamiento principal, su lanzamiento se realizó el 3 de septiembre de 2019, el 31 de enero de 2020 los fabricantes estaban obligados a lanzar los nuevos dispositivos con esta versión de fábrica, es el sistema operativo de fábrica utilizado en los dispositivos que usualmente usa la empresa.
Android 11:	Es la undécima versión principal, fue lanzado el 8 de septiembre de 2022, por su gran adaptabilidad los fabricantes de nuevos dispositivos comenzaron a lanzar Android 11 desde fábrica a las semanas de su lanzamiento. Un gran porcentaje de usuarios finales de la empresa actualizaron sus dispositivos a la versión de Android 11.

Tabla 11: Versiones de Android recomendadas

B. Dispositivos utilizado usualmente por la empresa

Xiaomi Redmi 9

Características técnicas

Pantalla
6.53", 1080 x 2340 pixels

Cámara Principal
Cuádruple
13MP+8MP +5MP+2MP

Batería
5020 mAh

Dimensiones
Perfil: 9.1 mm
Peso: 198 g



AP
Mediatek Helio G80 2GHz

RAM
3GB/4GB

Almacenamiento
32GB/64GB
microSD

OS
Android 10

Ilustración 32: Dispositivo utilizados usualmente por la empresa

XI. GLOSARIO

1. Automatizar: aplicar procedimientos automáticos en la realización de un proceso o en una industria.
2. Digitar: incorporar datos a una computadora u otro dispositivo electrónico utilizando el teclado.
3. ERP: sistemas de información gerenciales que integran y manejan muchos de los negocios asociados con las operaciones de producción y de los aspectos de distribución de una compañía en la producción de bienes o servicios.
4. Expendedores: persona que expende determinados productos.
5. Prototipo: primer modelo que sirve como representación o simulación del producto final y que permite verificar el diseño y confirmar que cuenta con las características específicas planteadas.
6. Sincronizar: tipo de integración que mantiene la coherencia de los datos con la base de datos.