



# GRADO EN INGENIERÍA INFORMÁTICA

## TRABAJO FINAL DE GRADO

---

Sistema de registro de visitas comerciales como  
herramienta para la toma de decisiones ejecutivas

---

Supervisor:

Autor:

Nacho Porcar Botas

Iván Serrano Lluch

Tutor académico:

Cristina Campos Sancho

Fecha de lectura: 18 de julio de 2019

Curso académico 2018/2019



## **RESUMEN**

El proyecto se ha realizado en la empresa SACMI IBERICA bajo un contrato profesional. Ha sido supervisado por el director del área de desarrollo del departamento de Informática de la empresa, Nacho Porcar, licenciado en Ingeniería Informática en la Universidad Jaume I (UJI). A su vez, este proyecto ha sido propuesto y consensuado por la gerencia de la empresa y el responsable del personal comercial. Finalmente, será evaluado como una exención en la asignatura *EII054 - Prácticas en Empresa y Proyecto Fin de Grado* del Grado en Ingeniería Informática de la UJI de Castellón.

El proyecto a realizar consta de varias partes, una de ellas es el desarrollo de una aplicación *Android* móvil que permita registrar las visitas realizadas por los comerciales a los clientes. Por otro lado, el desarrollo del *back-end*, con la funcionalidad de mantener la consistencia entre las bases de datos del sistema, y el desarrollo del *ERP* corporativo (*Enterprise Resource Planning*), para consultar las visitas registradas por los comerciales en las interfaces habituales de uso de los usuarios. Y finalmente, la explotación de los datos almacenados y el desarrollo de un *dashboard* interactivo para facilitar el proceso de toma de decisiones ejecutivas.

## **Palabras clave**

Comerciales, Aplicación, *ERP*, *Android*, *Node JS*, Móvil.

## **Keywords**

Commercials, Application, ERP, Android, Node JS, Mobile.



# Índice General

<b>Introducción</b> .....	9
1.1. Descripción del proyecto .....	9
1.2. Contexto y motivación del proyecto.....	10
1.3. Objetivos del proyecto .....	11
1.4. Tecnologías utilizadas .....	12
1.5. Estructura memoria.....	14
<b>Planificación</b> .....	17
2.1. Metodología.....	17
2.2. Secuenciar, organizar y temporalizar actividades.....	18
2.3. Estimación de recursos, costes y beneficios del proyecto .....	19
2.4. Seguimiento del proyecto.....	21
2.5. Gestión de riesgos .....	24
<b>Análisis</b> .....	29
3.1. Definición de requisitos .....	29
3.2. Diagrama de casos de uso .....	31
3.4. Estudio datos internos necesarios.....	43
3.5. Integración de datos del ERP hacia la APP.....	44
3.6. Integración de datos de la APP hacia el ERP .....	45
<b>Diseño</b> .....	47
4.1 Diseño APP .....	47
4.1.1. Diseño DB APP.....	47
4.1.2. Diseño Inicio Sesión, APP .....	50
4.1.3. Diseño registro visita comercial, APP .....	51
4.2. Diseño <i>back-end</i> .....	53
4.2.1 Diseño proceso integración datos, <i>ERP</i> → <i>APP</i> .....	53
4.2.2. Diseño proceso integración datos, <i>APP</i> → <i>ERP</i> .....	54
4.3. Diseño interfaz consulta de visitas, <i>ERP</i> .....	54
4.4. Diseño <i>dashboard Power BI</i> .....	55
<b>Implementación y pruebas</b> .....	57
5.1. Implementación APP .....	57
5.1.1. Inicio Sesión, APP.....	57
5.1.2. Implementación registro visita comercial, APP .....	59
5.2. Implementación <i>back-end</i> .....	64
5.2.1 Implementación del proceso de integración de datos, <i>ERP</i> → <i>APP</i> .....	64

5.2.2. Implementación del proceso de integración de datos, APP→ERP .....	66
5.3. Implementación interfaz de consulta de visitas, ERP .....	67
5.4. Implementación <i>dashboard Power BI</i> .....	68
5.5. Pruebas.....	70
5.5.1. Pruebas consistencia proyecto.....	71
5.5.2. Pruebas con usuarios finales .....	71
<b>Conclusiones</b> .....	73
<b>Bibliografía</b> .....	75

# Índice de Tablas

Tabla 1. Estimación recursos <i>software</i> .....	21
Tabla 2. Descripción de los riesgos.....	24
Tabla 3. Análisis de los riesgos.....	25
Tabla 4. Prevención y corrección de los riesgos.....	27
Tabla 5. Control de riesgos.....	28
Tabla 6. Caso de uso: Importar datos clientes y comerciales.....	32
Tabla 7. Requisitos del caso de uso: Importar datos clientes y comerciales.....	33
Tabla 8. Caso de uso: Iniciar sesión.....	34
Tabla 9. Requisitos del caso de uso: Iniciar sesión.....	35
Tabla 10. Caso de uso: Registrar visita a clientes.....	36
Tabla 11. Requisitos del caso de uso: Registrar visita a cliente.....	37
Tabla 12. Caso de uso: Exportar datos de las visitas de los comerciales.....	38
Tabla 13. Requisitos del caso de uso: Exportar datos de las visitas de los comerciales.....	39
Tabla 14. Caso de uso: Consultar visitas comerciales.....	40
Tabla 15. Requisitos del caso de uso: Consultar visitas comerciales.....	41
Tabla 16. Caso de uso: Consultar estadísticas de visitas comerciales.....	42
Tabla 17. Requisitos del caso de uso: Consultar estadísticas de visitas comerciales.....	43

# Índice de Figuras

Figura 1. Diagrama de Gantt.....	18
Figura 2. Diagrama de Gantt de seguimiento.....	23
Figura 3. Diagrama de casos de uso.....	31
Figura 4. Raíz proyecto Visitas Comerciales.....	48
Figura 5. Árbol padre Cliente.....	48
Figura 6. Árbol padre Comerciales.....	49
Figura 7. Árbol padre Visitas.....	50
Figura 8. Prototipo interfaz <i>LogIn</i> APP.....	51
Figura 9. Prototipo interfaz registro de visita APP.....	52
Figura 10. Prototipo interfaz visualización de datos almacenados APP.....	53
Figura 11. Interfaz <i>ERP Microsoft Dynamics NAV</i> .....	55
Figura 12. Prototipo <i>dashboard</i> de <i>Power BI</i> .....	56
Figura 13. Interfaz <i>Splash A</i> .....	58
Figura 14. Interfaz Inicio de Sesión APP.....	58
Figura 15. Interfaz registro de la visita.....	59
Figura 16. Interfaz registro almacenado de la visita.....	60
Figura 17. Lista seleccionable Clientes.....	61
Figura 18. Mensaje informativo I.....	62
Figura 19. Mensaje informativo II.....	62
Figura 20. Almacenaje de Visita.....	63
Figura 21. Confirmación de salida.....	64
Figura 22. Diseño lógico en estructura en estrella.....	70
Figura 23. <i>Dashboard</i> Interactivo.....	71



# Capítulo 1

## Introducción

El apartado de Introducción del presente Trabajo Final de Grado presenta el contexto de realización del proyecto, así como la motivación que ha impulsado su desarrollo. Seguidamente, se enumeran los objetivos planteados, las tecnologías utilizadas y la estructura del documento.

### 1.1. Descripción del proyecto

El presente proyecto está planteado por SACMI IBÉRICA. Más concretamente, está consensuado por el responsable de los comerciales y la gerencia de la empresa, y supervisado por el responsable del departamento de Informática.

El proyecto pretende desarrollar una herramienta que permita almacenar información sobre las visitas realizadas por los comerciales a los clientes, y poder explotar esta información junto con la información ya existente en la empresa para mejorar el proceso ejecutivo de toma de decisiones. Los datos se integrarán en el *ERP (Enterprise Resource Planning)* corporativo para que todos los usuarios de la empresa tengan acceso. Finalmente, se proporcionará la información a los ejecutivos de la empresa mediante un *dashboard* interactivo.

La herramienta a desarrollar para el registro de las vistas de los comerciales es una aplicación móvil *Android*. Por una parte se escogió el sistema operativo *Android* porque actualmente los usuarios utilizan móviles con dicho sistema operativo, y por otro lado, se decidió crear una aplicación móvil para que los comerciales pudieran trasladar el dispositivo fuera de nuestras instalaciones con la mayor comodidad posible.

La herramienta con la que presentaremos los datos analizados a los ejecutivos es un *dashboard* con *Power BI (Business Intelligence)*.

## 1.2. Contexto y motivación del proyecto

El proyecto presentado se ha desarrollado en la empresa SACMI IBERICA como un proyecto formativo. Cabe destacar que la empresa siempre ha tenido este proyecto en el punto de mira, pero no lo había realizado hasta la fecha por la falta de tiempo y de personal para llevarlo a cabo.

Para entender el por qué de la necesidad de implantar esta aplicación en la empresa, es necesario conocer de un modo general su contexto.

SACMI IBERICA se dedica principalmente a la venta y el mantenimiento de maquinaria industrial para el sector cerámico. Pertenece al grupo SACMI, un grupo formado por más de 80 empresas ubicadas por todo el mundo, que posee su sede principal en la ciudad de Imola (Italia), donde se produce la mayoría de la maquinaria ofertada. Esto implica que la empresa SACMI IBERICA no consta de un departamento de producción, la maquinaria es obtenida a través de proveedores afiliados. No obstante, la sede consta de los siguientes departamentos: departamento Comercial, dedicado a la venta de esta maquinaria; departamento Administrativo, encargado de la monetización de las ventas, los pagos a los clientes y los salarios internos; departamento de Recursos Humanos (RRHH), encargado de la contratación del personal; departamento de Compras y Ventas, responsable de las compras y las ventas de los recambios necesarios para el mantenimiento de la maquinaria; departamento de Asistencia, encargado de la reparación de la maquinaria del cliente; departamento de Almacén, responsable de la gestión de todo el material recepcionado en SACMI IBERICA; departamento Informático, encargado del mantenimiento y la gestión del *ERP* corporativo y del desarrollo y mantenimiento de nuevos proyectos tecnológicos prioritarios para la empresa; y finalmente el departamento Ejecutivo, encargado de la supervisión y organización de todos y cada uno de los departamentos de la empresa.

Todos estos departamentos utilizan para su gestión interna y almacenamiento de datos el *ERP* corporativo *Microsoft Dynamics NAV*, por lo que el personal puede disponer de los datos con facilidad, mantenerlos localizados y extraerlos para su análisis.

Hasta el momento, SACMI IBERICA no almacenaba información referente a las visitas realizadas por los comerciales, por lo que no se podía tener ningún control sobre ello. La principal motivación para desarrollar el proyecto fueron los beneficios que podía aportar a la empresa la implementación de una aplicación que permitiera almacenar información sobre las

visitas realizadas por los comerciales a los clientes, así como el análisis y la explotación de los datos almacenados mediante herramientas como *Power BI*.

A nivel personal de aprendizaje, el proyecto permite reforzar conocimientos afines a la línea del itinerario de Sistemas de la Información, como por ejemplo, el desarrollo de nuevas tecnologías, el desarrollo multiplataforma, la integración de la información, los *ERPs*, y el análisis y la explotación de los datos.

Finalmente, este proyecto será evaluado en la asignatura *EII054 - Prácticas en Empresa y Proyecto Fin de Grado* del Grado en Ingeniería Informática de la Universidad Jaume I (UJI) de Castellón como una exención.

### **1.3. Objetivos del proyecto**

*El principal objetivo de este proyecto es desarrollar una aplicación móvil para sistemas Android que sirva para almacenar información sobre las visitas realizadas por los comerciales a los clientes de la empresa y, junto con la explotación de estos datos almacenados y la información ya existente en la empresa, pretende ser una herramienta de análisis para los ejecutivos de la empresa mejorando los procesos de toma de decisiones ya existentes.*

El objetivo principal lo podemos desglosar en los siguientes subobjetivos:

1. Desarrollo de un sistema para la captación de datos sobre las visitas de los comerciales.
  - a. Permitir al usuario elegir el cliente que va a visitar.
  - b. Permitir introducir y almacenar información por parte del comercial, sobre la visita realizada.
2. Mayor control para el responsable de los comerciales y la gerencia sobre sus empleados.
  - a. Permitir consultar las visitas realizadas por los comerciales.
  - b. Permitir consultar los datos explotados.
  - c. Permitir detectar comerciales acomodados.
  - d. Premiar a comerciales implicados.
3. Mejorar el proceso de toma de decisiones ejecutivas que afectan al departamento comercial.

El alcance de este proyecto es, por tanto, proveer a la empresa de un sistema que permita almacenar información sobre las visitas de los comerciales realizadas a los clientes, con el fin de elaborar una herramienta de análisis y explotación de los datos, que permita mejorar el proceso de toma de decisiones ejecutivas.

## 1.4. Tecnologías utilizadas

Para mencionar las tecnologías a utilizar en este proyecto, es necesario diferenciar entre cuatro partes: la primera parte es la aplicación móvil, la segunda es el *ERP* corporativo, la tercera es el intercambio de información entre estas dos primeras partes mediante (*back-end*), y finalmente la herramienta de explotación de los datos almacenados por la aplicación (*dashboard* de *Power BI*).

En esta primera parte, se opta técnicamente por el desarrollo de una aplicación móvil *Android* en el *IDE (Integrated Development Environment) Android Studio* porque el departamento de informática de la empresa ya tiene experiencia con este *IDE* de desarrollo y supone un riesgo menos que tomar a la hora de implementar el proyecto. En cuanto a la base de datos que utiliza la aplicación, se optó por la plataforma *Firebase*, otra vez por la experiencia que tiene la empresa en esta tecnología y además por las características que ofrece como base de datos en tiempo real y servicios de autenticación de usuarios.

En la segunda parte se encuentra el *ERP* corporativo, *Microsoft Dynamics NAV*, que es el software de gestión interna que se utiliza en SACMI IBERICA, por lo que era obligado desarrollar las funcionalidades necesarias para que la información esté disponible para toda la empresa, en esta tecnología.

La tercera parte hace referencia a los procesos del intercambio de información entre las dos bases de datos, *Firebase* y la base de datos del *ERP* corporativo. Estos procesos de intercambio de información se han implementado con diversos *scripts* programados en *Node JS*, el cual se ha elegido por la experiencia del equipo de informática con él, porque es muy sencillo y rápido instalar y programar, la documentación es muy extensa, tiene una enorme cantidad de librerías listas para utilizar, la comunidad es la más extensa del mundo, y es *opensource*. El *IDE* elegido para programar estos *scripts* es *Visual Studio Code*, por la experiencia previa y porque el coste es *free*.

Finalmente, la cuarta parte, el *dashboard* de *Power BI*, que muestra la información tratada y analizada. Se ha escogido *Power BI* por su potencia, flexibilidad y facilidad de uso. También cabe mencionar que la empresa contaba con la licencia y, por tanto, trabajar con *Power BI* no supone ningún sobrecoste.

A continuación, se detallan los lenguajes, herramientas y utilidades de estas tecnologías:

### ***Android Studio***

*IDE* oficial para la plataforma *Android*. Proporciona emuladores *Android* y está basado en *IntelliJ*. Diseñado especialmente para el desarrollo en *Android*. Su descarga, instalación y uso es gratuito. Admite el lenguaje de programación Java para el desarrollo de las funcionalidades y el lenguaje de marcado *XML (Extensible Markup Language)* para el diseño de interfaces [1].

### ***Firebase Realtime Database***

*Firebase* es una plataforma para el desarrollo de aplicaciones móviles que pertenece a *Google*. *Firebase* cuenta con varios servicios para el análisis, gestión y desarrollo de proyectos, de los que se va a utilizar *Realtime Database*. Este servicio proporciona una base de datos no relacional, con estructura jerárquica en formato *JSON (Javascript Object Notation)*, que no se basa en tablas, alojada en la nube y accesible a tiempo real [2].

### ***Microsoft Dynamics NAV***

Es un *software ERP* que permite la gestión, planificación, integración y automatización de diferentes procesos dentro de una empresa [3]. Forma parte de la familia de productos *Microsoft Dynamics*. Es un software que cuenta con una parte cliente, que es la que utiliza el usuario final, otra parte que es el servidor de base de datos donde se almacena la información (solo utiliza *SQL Server*) y finalmente la parte del desarrollador, el cual consta del *IDE C/SIDE* que se utiliza para la gestión de los objetos de este entorno de desarrollo integrado. El lenguaje de programación que se utiliza en este *IDE* es *C/AL* [4].

## ***Node JS***

Entorno de ejecución en tiempo real multiplataforma, de código abierto, de la parte del servidor, basado en *ECMAScript*, asíncrono y basado en el motor V8 de *Google Chrome*. Coloquialmente conocido como *JavaScript* de la parte del servidor [5].

## ***Visual Studio Code***

*Visual Studio Code* es un editor de código fuente desarrollado por *Microsoft*. Entre otras cosas incluye soporte para la depuración, control de versiones integrado con *Git*, resaltado de sintaxis, control de autocompletado, refactorización del código y terminal integrada [6]. Se ha utilizado en el proyecto como entorno de desarrollo para programar los procesos *Node JS* de intercambio de información entre las dos bases de datos.

## ***Power BI***

Servicio de análisis de negocios desarrollado por *Microsoft*. Permite proporcionar visualizaciones interactivas con una interfaz lo suficientemente simple para que los usuarios finales creen sus propios informes y *dashboards*. También proporciona servicios basados en la nube y ofrece capacidades de almacenamiento de datos [7].

# **1.5. Estructura memoria**

**Capítulo 1. Introducción.** La introducción del proyecto sitúa al lector en el contexto en que se ha desarrollado el proyecto, es decir, la empresa y las motivaciones que han llevado a plantearlo, y una breve descripción del mismo. Además, se plantean los objetivos, el alcance del trabajo y las tecnologías utilizadas.

**Capítulo 2. Planificación.** El capítulo 2 comprende la metodología utilizada y la organización cronológica de las actividades, así como los costes, beneficios y riesgos de su implementación.

**Capítulo 3. Análisis, Diseño, Implementación y Pruebas.** En este capítulo, se realiza la definición de requisitos y el estudio de los datos necesarios para el desarrollo del sistema.

**Capítulo 4. Diseño.** Este capítulo muestra el diseño de los componentes del proyecto, junto a una serie de prototipos previos.

**Capítulo 5. Implementación y pruebas.** En este apartado se desarrollan las fases de implementación y pruebas finales del proyecto.

**Capítulo 6. Conclusiones.** Este apartado recoge las principales conclusiones obtenidas de la realización del presente Trabajo Final de Grado.





## Capítulo 2

# Planificación

En este apartado se exponen las metodologías utilizadas en el proyecto, la planificación temporal, los costes estimados, el seguimiento del proyecto y la gestión de los riesgos establecidos.

### 2.1. Metodología

SACMI IBERICA aborda nuevos proyectos tecnológicos respondiendo a la prioridad que le impone el gerente de la empresa al proyecto, siempre consensuadamente con todas las partes implicadas.

Una vez seleccionado el proyecto, el responsable de informática es el encargado de supervisar el proyecto y dividirlo en las fases que considere conveniente. En este caso, al tratarse de un proyecto en el que las partes intervinientes se localizan en el mismo edificio, con facilidad para concertar reuniones con la frecuencia requerida, y al tratarse de un proyecto interno ya definido, se optó por una metodología predictiva.

Para esta metodología, durante la toma de requisitos se utilizaron reuniones; una reunión inicial con todas las partes implicadas por separado, y otra reunión todos juntos para terminar de refinar la primera toma de requisitos. Posteriormente para la revisión de la implementación, se hizo un análisis y diseño utilizando *UML (Unified Modeling Language)* como técnica de modelado, y para validar los desarrollos se fueron presentando prototipos a los usuarios finales con frecuencia, pero sin ninguna reunión formal, únicamente intercambios de opiniones y anotaciones.

## 2.2. Secuenciar, organizar y temporalizar actividades

La cronología de las actividades del proyecto se presenta mediante un diagrama de Gantt, que se muestra en la figura 1.

Figura 1. Diagrama de Gantt.

<b>Proyecto</b>	<b>64 days</b>	<b>Mon 02/07/18</b>	<b>Thu 27/09/18</b>	
<b>Inicio</b>	<b>6 days</b>	<b>Mon 02/07/18</b>	<b>Mon 09/07/18</b>	
Definir proyecto con supervisor	1 day	Mon 02/07/18	Mon 02/07/18	
Estudio del caso	3 days	Tue 03/07/18	Thu 05/07/18	3
Definir objetivos y alcance	1 day	Fri 06/07/18	Fri 06/07/18	4
Definir requisitos	1 day	Mon 09/07/18	Mon 09/07/18	5
<b>Planificación</b>	<b>13 days</b>	<b>Mon 09/07/18</b>	<b>Wed 25/07/18</b>	
Definir metodología	1 day	Mon 09/07/18	Mon 09/07/18	5
Secuenciar actividades	2 days	Tue 10/07/18	Wed 11/07/18	8
Estimar recusos, costes y beneficios del proyecto	2 days	Thu 12/07/18	Fri 13/07/18	9
Gestión de Riesgos	1 day	Mon 16/07/18	Mon 16/07/18	10
Definición tecnologías utilizadas	1 day	Tue 17/07/18	Tue 17/07/18	11
Formación arquitectura tecnológica organización	6 days	Wed 18/07/18	Wed 25/07/18	12
<b>Modelización</b>	<b>46 days</b>	<b>Thu 26/07/18</b>	<b>Thu 27/09/18</b>	
<b>Análisis</b>	<b>22 days</b>	<b>Thu 26/07/18</b>	<b>Fri 24/08/18</b>	
Definición de requisitos	3 days	Thu 26/07/18	Mon 30/07/18	13
Diagramas casos de uso	2 days	Tue 31/07/18	Wed 01/08/18	16
Requisitos de datos	1 day	Thu 02/08/18	Thu 02/08/18	17
Estudio datos internos necesarios	4 days	Fri 03/08/18	Wed 08/08/18	18
Estudio importación datos APP	4 days	Thu 09/08/18	Tue 14/08/18	19
Formación en las TIC a utilizar	8 days	Wed 15/08/18	Fri 24/08/18	20
<b>Diseño</b>	<b>8 days</b>	<b>Mon 27/08/18</b>	<b>Wed 05/09/18</b>	
<b>Diseño APP</b>	<b>3 days</b>	<b>Mon 27/08/18</b>	<b>Wed 29/08/18</b>	
Diseño DB APP	1 day	Mon 27/08/18	Mon 27/08/18	21
Diseño Inicio Session, APP	1 day	Tue 28/08/18	Tue 28/08/18	24
Diseño registro visita comercial, APP	1 day	Wed 29/08/18	Wed 29/08/18	25
<b>Diseño back-end</b>	<b>3 days</b>	<b>Tue 28/08/18</b>	<b>Thu 30/08/18</b>	
Diseño proceso de integración de datos, ERP → APP	1 day	Tue 28/08/18	Tue 28/08/18	24
Diseño proceso de integración de datos, APP → ERP	1 day	Thu 30/08/18	Thu 30/08/18	26
Diseño interfaz de consulta de visitas, ERP	1 day	Fri 31/08/18	Fri 31/08/18	29
Diseño dashboard Power BI	1 day	Mon 03/09/18	Mon 03/09/18	30
Verificación y validación	2 days	Tue 04/09/18	Wed 05/09/18	31
<b>Implementación</b>	<b>16 days</b>	<b>Thu 06/09/18</b>	<b>Thu 27/09/18</b>	
<b>Implementación APP</b>	<b>10 days</b>	<b>Thu 06/09/18</b>	<b>Wed 19/09/18</b>	
Implementación Inicio Session, APP	2 days	Thu 06/09/18	Fri 07/09/18	32
Implementación registro visita comercial, APP	8 days	Mon 10/09/18	Wed 19/09/18	35
<b>Implementación back-end</b>	<b>14 days</b>	<b>Thu 06/09/18</b>	<b>Tue 25/09/18</b>	
Implementación proceso de integración de datos, ERP → APP	3 days	Thu 06/09/18	Mon 10/09/18	32
Implementación proceso de integración de datos, APP → ERP	4 days	Thu 20/09/18	Tue 25/09/18	36
Implementación interfaz consultas visitas ERP	1 day	Wed 26/09/18	Wed 26/09/18	39
Implementación dashboard Power BI	1 day	Thu 27/09/18	Thu 27/09/18	40
<b>Pruebas</b>	<b>5 days</b>	<b>Thu 20/09/18</b>	<b>Wed 26/09/18</b>	
Pruebas funcionamiento	2 days	Thu 20/09/18	Fri 21/09/18	35;36
Pruebas con usuarios finales	2 days	Mon 24/09/18	Tue 25/09/18	43
Verificación y validación	1 day	Wed 26/09/18	Wed 26/09/18	44

La Figura 1 representa el diagrama de Gantt Inicial del proyecto, que representa las distintas fases por las que se prevé que pase el proyecto durante su desarrollo.

## 2.3. Estimación de recursos, costes y beneficios del proyecto

En este apartado se estiman los recursos necesarios para llevar a cabo el proyecto, y a su vez, el coste que generan estos recursos.

Inicialmente, es necesario recordar que el proyecto desarrollado no es para la venta a ningún cliente, sino que es para el uso interno en la organización, por lo que no se obtendrá ninguna monetización al final de la entrega del proyecto. Por otra parte, sí se deben estimar los recursos necesarios, y el coste que supone para la organización el desarrollo del proyecto.

### Recursos

Los recursos necesarios son de tres tipos: recursos humanos, recursos físicos y de hardware y recursos de software.

En cuanto a los recursos humanos, se encuentran implicadas varias partes de la empresa. Una de ellas es el departamento comercial, que serán los usuarios finales de la aplicación, por lo que será necesario tener interacción constante con ellos. Por otro lado, el sector ejecutivo consulta el *dashboard* que les ayuda en el proceso de toma de decisiones, por lo que deberá validar el diseño y funcionamiento de este. Finalmente, se encuentra el equipo informático, que es el encargado del desarrollo de todo el sistema.

En referencia a los recursos físicos y hardware, el equipo informático necesita un *PC* y un teléfono móvil *Android* para probar las funciones de la aplicación móvil. También se precisa de los servidores de la empresa para alojar nuestro pequeño *back-end*, y además, es el lugar donde residen los datos corporativos.

Finalmente, los recursos de software requeridos para llevar a cabo el desarrollo de este proyecto, se pueden dividir en cuatro partes: la parte del desarrollo de la aplicación móvil, la parte de la explotación de los datos, la parte de desarrollo con el *ERP* corporativo y la parte del *back-end*.

En la parte del desarrollo de la aplicación móvil, se utiliza el *IDE* de desarrollo *Android Studio* para la implementación de la parte gráfica y funcional de la aplicación. También se

utiliza la plataforma *Firebase*, en la que se almacenan los datos de la aplicación, y se registran y autentican los usuarios.

Por otro lado, la parte de la explotación de los datos se ha realizado mediante la herramienta *Power BI*, y para el desarrollo con el ERP corporativo se utiliza el desarrollador de *Microsoft Dynamics NAV* junto con su base de datos *SQL Server*.

En cuanto al desarrollo con el ERP corporativo se utiliza el desarrollador de *Microsoft Dynamics NAV* junto con su base de datos *SQL Server*.

Por último, en la parte del *back-end* se han utilizado la tecnología *Node JS* para implementar la funcionalidad y el editor de código *Visual Studio Code* como *IDE* de desarrollo.

## **Coste**

El proyecto realizado, como hemos mencionado anteriormente, es para uso interno de la organización, por lo que los costes se calcularán en función del gasto económico que ha supuesto a la organización la realización de este proyecto.

Para el cálculo de los recursos personales es necesario tener en cuenta el coste indirecto de los sueldos de los empleados, pero como únicamente se ha destinado a un miembro del equipo informático trabajando a tiempo completo en el proyecto, solo se calculará el coste de este, ya que resto de participantes intervienen mínimamente. Por tanto, el coste de los recursos personales para la organización es de 10€ la hora, haciendo una jornada de 8 horas diarias durante 66 días, el coste total es de 5.280€.

En cuanto al coste de los recursos físicos o hardware, el desarrollador del proyecto contaba previamente con un *PC* y un dispositivo móvil en las condiciones adecuadas para poder desarrollar el proyecto. Además, los servidores de la empresa tenían el suficiente espacio libre para almacenar el *back-end*, siendo que el coste de esta parte de 0€.

Finalmente, los gastos extraídos de los recursos de *software*, se encuentran desglosados en las cuatro partes principales mencionadas en el apartado Recursos.

La parte del desarrollo de la aplicación móvil ha tenido un coste de 0€, ya que el *IDE Android Studio* es gratuito y la plataforma *Firebase* también cuenta con una versión gratuita que

cumple con los requerimientos necesarios. Sin embargo, en la parte que utilizamos la herramienta *Power BI*, sí que tiene un coste de licencia, pero la organización ya disponía de licencias cedidas por el grupo SACMI con un único pago de 100€.

En cuanto al coste del desarrollo en el *ERP Microsoft Dynamics NAV* ha sido 0€. Esto se debe a que es una herramienta de uso diario en SACMI IBERICA, por lo que las licencias requeridas se encontraban disponibles. El único problema que se podría encontrar sería en el caso de que hubieran *Tables* o *Pages* libres contratadas; en este caso, se debería pagar al *partner* para que las añadiera, pero este no es el caso ya que tenemos espacio de sobra.

En el *back-end* se ha utilizado la tecnología *Node JS* que es *open source* y el editor de código *Visual Studio Code*, que es gratuito. Siendo de 0€ el coste de esta parte.

Tabla 1. Estimación recursos *software*.

<b>Producto <i>Software</i></b>	<b>Precio</b>
<i>IDE Android Studio</i>	0€
Plataforma <i>Firebase</i>	0€
<i>Power BI</i>	100€
Desarrollador <i>Microsoft Dynamics NAV</i>	0€
<i>Node JS</i>	0€
<i>Visual Studio Code</i>	0€

La Tabla 1 muestra los precios de cada producto de software que se han utilizado para el desarrollo de este proyecto.

A modo de resumen, el coste total del proyecto sería el sueldo del desarrollador del sistema, que son 5.280€, y el coste de la licencia de la herramienta *Power BI*, que era de 100€, alcanzado un total de 5.380€.

## 2.4. Seguimiento del proyecto

El diagrama de Gantt se modificó según las necesidades y/o modificaciones que surgieron durante la realización del proyecto. En la figura 2 se muestra una comparación entre el diagrama

de Gantt inicial y final, que permite observar un leve retraso de tres días entre las actividades planificadas previamente y las actividades realizadas finalmente.

Como se aprecia en la Figura 2, Diagrama de Gantt de seguimiento, el proyecto ha alargado su duración en tres días. Este pequeño retraso en la planificación no ha supuesto ningún problema adicional, ya que no existía un plazo de entrega establecido.

Esta demora se ha producido por la inversión de un día más en la formación en la arquitectura tecnológica de la organización, puesto que, aunque inicialmente no estaba contemplada, resultó una excelente forma de familiarizarse con el sistema. Dicho cambio en la planificación del proyecto se ha producido en la fase de Planificación, viéndose así arrastrado durante todo el proyecto. Esto es debido entre otros factores a la metodología predictiva elegida.

La otra alteración que ha sufrido la planificación del proyecto ha sido en la fase de Implementación, más concretamente en la tarea de implementación del apartado encargado del registro de la visita comercial. Este cambio ha supuesto un aumento de dos días, y ha sido causado por la escasa experiencia en la plataforma *Firebase* por parte del equipo, sobre todo, en cuanto a la consulta y extracción de datos de la base de datos, por lo que se ha precisado ampliar los conocimientos referentes a la plataforma.

Por tanto, el retraso total con respecto a la planificación inicial ha sido de tres días, causado generalmente por la búsqueda de información con la finalidad de obtener una formación mayor, y como se ha comentado anteriormente, no ha supuesto ningún problema.

Figura 2. Diagrama de Gantt de seguimiento.

<b>Proyecto</b>	<b>67 days</b>	<b>Mon 02/07/18</b>	<b>Tue 02/10/18</b>	
<b>Inicio</b>	<b>6 days</b>	<b>Mon 02/07/18</b>	<b>Mon 09/07/18</b>	
Definir proyecto con supervisor	1 day	Mon 02/07/18	Mon 02/07/18	
Estudio del caso	3 days	Tue 03/07/18	Thu 05/07/18	3
Definir objetivos y alcance	1 day	Fri 06/07/18	Fri 06/07/18	4
Definir requisitos	1 day	Mon 09/07/18	Mon 09/07/18	5
<b>Planificación</b>	<b>14 days</b>	<b>Mon 09/07/18</b>	<b>Thu 26/07/18</b>	
Definir metodología	1 day	Mon 09/07/18	Mon 09/07/18	5
Secuenciar actividades	2 days	Tue 10/07/18	Wed 11/07/18	8
Estimar recussos, costes y beneficios del proyecto	2 days	Thu 12/07/18	Fri 13/07/18	9
Gestión de Riesgos	1 day	Mon 16/07/18	Mon 16/07/18	10
Definición tecnologías utilizadas	1 day	Tue 17/07/18	Tue 17/07/18	11
Formación arquitectura tecnológica organización	7 days	Wed 18/07/18	Thu 26/07/18	12
<b>Modelización</b>	<b>48 days</b>	<b>Fri 27/07/18</b>	<b>Tue 02/10/18</b>	
<b>Analisis</b>	<b>22 days</b>	<b>Fri 27/07/18</b>	<b>Mon 27/08/18</b>	
Definición de requisitos	3 days	Fri 27/07/18	Tue 31/07/18	13
Diagramas casos de uso	2 days	Wed 01/08/18	Thu 02/08/18	16
Requisitos de datos	1 day	Fri 03/08/18	Fri 03/08/18	17
Estudio datos internos necesarios	4 days	Mon 06/08/18	Thu 09/08/18	18
Estudio importación datos APP	4 days	Fri 10/08/18	Wed 15/08/18	19
Formación en las TIC a utilizar	8 days	Thu 16/08/18	Mon 27/08/18	20
<b>Diseño</b>	<b>8 days</b>	<b>Tue 28/08/18</b>	<b>Thu 06/09/18</b>	
<b>Diseño APP</b>	<b>3 days</b>	<b>Tue 28/08/18</b>	<b>Thu 30/08/18</b>	
Diseño DB APP	1 day	Tue 28/08/18	Tue 28/08/18	21
Diseño Inicio Session, APP	1 day	Wed 29/08/18	Wed 29/08/18	24
Diseño registro visita comercial, APP	1 day	Thu 30/08/18	Thu 30/08/18	25
<b>Diseño back-end</b>	<b>3 days</b>	<b>Wed 29/08/18</b>	<b>Fri 31/08/18</b>	
Diseño proceso de integración de datos, ERP → APP	1 day	Wed 29/08/18	Wed 29/08/18	24
Diseño proceso de integración de datos, APP → ERP	1 day	Fri 31/08/18	Fri 31/08/18	26
Diseño interfaz de consulta de visitas, ERP	1 day	Mon 03/09/18	Mon 03/09/18	29
Diseño dashboard Power BI	1 day	Tue 04/09/18	Tue 04/09/18	30
Verificación y validación	2 days	Wed 05/09/18	Thu 06/09/18	31
<b>Implementación</b>	<b>18 days</b>	<b>Fri 07/09/18</b>	<b>Tue 02/10/18</b>	
<b>Implementación APP</b>	<b>12 days</b>	<b>Fri 07/09/18</b>	<b>Mon 24/09/18</b>	
Implementación Inicio Session, APP	2 days	Fri 07/09/18	Mon 10/09/18	32
Implementación registro visita comercial, APP	10 days	Tue 11/09/18	Mon 24/09/18	35
<b>Implementación back-end</b>	<b>16 days</b>	<b>Fri 07/09/18</b>	<b>Fri 28/09/18</b>	
Implementación proceso de integración de datos, ERP → APP	3 days	Fri 07/09/18	Tue 11/09/18	32
Implementación proceso de integración de datos, APP → ERP	4 days	Tue 25/09/18	Fri 28/09/18	36
Implementación interfaz consultas visitas ERP	1 day	Mon 01/10/18	Mon 01/10/18	39
Implementación dashboard Power BI	1 day	Tue 02/10/18	Tue 02/10/18	40
<b>Pruebas</b>	<b>5 days</b>	<b>Tue 25/09/18</b>	<b>Mon 01/10/18</b>	
Pruebas funcionamiento	2 days	Tue 25/09/18	Wed 26/09/18	35;36
Pruebas con usuarios finales	2 days	Thu 27/09/18	Fri 28/09/18	43
Verificación y validación	1 day	Mon 01/10/18	Mon 01/10/18	44

La Figura 2 representa mediante un diagrama de Gantt las distintas fases por las que finalmente pasa el proyecto.

## 2.5. Gestión de riesgos

En el siguiente apartado se desarrolla uno de los elementos clave a la hora de asegurar el éxito del proyecto, la gestión de riesgos, medido en términos de cumplimiento de plazos, costes, alcance funcional y calidad final de la solución. Esto aumenta la probabilidad del impacto de los acontecimientos positivos y disminuye la probabilidad y el impacto de los acontecimientos negativos del proyecto.

Para realizar un control adecuado de los riesgos, es necesario hacer el seguimiento durante las fases que componen el proyecto, realizando el análisis necesario al inicio de cada fase y también al final, así se podrán extraer conclusiones válidas para el futuro.

Los riesgos identificados en este proyecto se muestran en la Tabla 2, y se describe su análisis en la Tabla 3.

Tabla 2. Descripción de los riesgos.

<b>ID</b>	<b>Descripción del Riesgo</b>	<b>Tipo de Riesgo</b>
R01	Cambio en la prioridad de los proyectos	Riesgo de Proyecto
R02	Abandono temporal miembro equipo	Riesgo del Proyecto
R03	Falta experiencia con las herramientas utilizadas	Riesgo de Producto/Proyecto
R04	Conflicto partes implicadas	Riesgo de Proyecto
R05	Estimación costos fuera del alcance de la realidad	Riesgo de Proyecto

La Tabla 2 muestra las descripciones de los distintos riesgos del proyecto.



Tabla 3. Análisis de los riesgos.

ID	Análisis Riesgo
R01	<p><b>Magnitud:</b> Alta.</p> <p><b>Descripción:</b> Al desarrollar un proyecto en una empresa que no se dedica íntegramente al desarrollo tecnológico, es posible que surjan nuevos proyectos a los que la gerencia les dé mayor prioridad, con lo que sería necesario pasar a dedicar el 100% de la disponibilidad al nuevo proyecto.</p> <p><b>Impacto:</b> El cambio a otro proyecto supone un retraso total del proyecto. La envergadura de este nuevo proyecto estará relacionada directamente con el retraso y además, al volver a retomar el proyecto, será necesario volver a estudiar el estado en el que se había quedado.</p> <p><b>Indicadores:</b> Clientes con nuevas peticiones, encontrar necesidades internas que necesitan cubrirse rápidamente.</p>
R02	<p><b>Magnitud:</b> Alta, cuando afecta a un miembro del departamento informático. Muy alta cuando afecta a varios miembros del departamento informático.</p> <p><b>Descripción:</b> Si algún miembro del equipo informático tiene que abandonar temporalmente el proyecto, se vería afectado, ya que para su correcto desarrollo son necesarios los conocimientos de todos los miembros del equipo.</p> <p><b>Impacto:</b> Teniendo en cuenta que la entrega puede posponerse, la falta de disponibilidad de personal no supone una pérdida de calidad en el producto.</p> <p><b>Indicadores:</b> No existen, ya que al estar causado por factores externos la proyecto, son muy difíciles de predecir.</p>
R03	<p><b>Magnitud:</b> Media</p> <p><b>Descripción:</b> El encargado de desarrollar este proyecto (Iván Serrano Lluch) no tiene experiencia en ninguna de las tecnologías que se van a utilizar para desarrollar el proyecto, ni de las que ya son utilizadas por la organización, por lo que necesitará una formación previa y un asesoramiento por parte del departamento de informática.</p> <p><b>Impacto:</b> La planificación guía todo el desarrollo del proyecto, por lo que un error en la misma puede incidir directamente en sus resultados. No obstante, el hecho de poder posponer la fecha de la entrega permite dedicar más tiempo a la formación.</p> <p><b>Indicadores:</b> Diferencias entre el desarrollo real del proyecto y la planificación estimada.</p>

Continuación Tabla 3. Análisis de los riesgos.

<b>ID</b>	<b>Análisis Riesgo</b>
R04	<p><b>Magnitud:</b> Media</p> <p><b>Descripción:</b> Aparición de problemas y discrepancias entre los miembros del proyecto. Falta de acuerdo en las decisiones tomadas.</p> <p><b>Impacto:</b> Si los desacuerdos no son rápidamente resueltos se pueden provocar retrasos en la planificación.</p> <p><b>Indicadores:</b> Mucho tiempo dedicado a decisiones concretas, énfasis en las posturas enfrentadas, número de enfrentamientos con respecto a una misma decisión. Malas relaciones existentes con anterioridad.</p>
R05	<p><b>Magnitud:</b> Media</p> <p><b>Descripción:</b> Se sobreestiman o subestiman los costes involucrados en el desarrollo del producto de <i>software</i>.</p> <p><b>Impacto:</b> Puede generar que el equipo entre en períodos de sobrecarga de trabajo o periodos de ausencia del mismo, lo que a su vez puede conllevar a un deterioro en la calidad.</p> <p><b>Indicadores:</b> El equipo trabaja más o menos horas de las inicialmente programadas, y se presentan quejas al jefe del proyecto o pedidos de redimensionamiento.</p>

La Tabla 3 muestra el análisis de los distintos riesgos contemplados en el proyecto.

Tras el análisis, se reflejaron en la Tabla 4 una serie de acciones de prevención y de corrección de los riesgos.

Finalmente, el control de los riesgos y su seguimiento a lo largo del proyecto, se han reflejado en la Tabla 5, siendo:

- Responsable: Persona o personas asignadas a la implantación de las acciones preventivas y/o correctoras
- Fecha Terminación: Fecha límite en la cual todas las acciones anteriormente descritas deban haber sido ejecutadas por el responsable o responsables asignados.
- Estado: Estado actual del riesgo y de las acciones preventivas y/o correctoras.

Tabla 4. Prevención y corrección de los riesgos.

ID	Plan de Prevención/Acción	Plan de Corrección/Contingencia
R01	Generación de documentación sobre el estado del proyecto.	El equipo de desarrollo deberá generar una serie de documentos sobre el estado del proyecto en las diversas fases, así como los factores a tener en cuenta que han intervenido.
R02	Tratar de cumplir las metas y los objetivos antes de lo estimado en la planificación, siempre que sea posible, para que una ausencia puntual de un miembro del equipo no suponga un retraso importante.	<p>El equipo de desarrollo tratará de cubrir el trabajo no realizado por el miembro del proyecto que no pueda trabajar. En caso necesario, dejarán de realizarse tareas menos importantes para centrarse en las principales.</p> <p>Se tratará de reajustar la planificación del proyecto, causando el menor número de cambios posible.</p>
R03	Realizar reuniones entre los miembros del proyecto para evaluar el desarrollo, consultar al tutor y dedicar más tiempo a la formación.	En caso necesario, después del estudio, se reajustará la planificación del proyecto añadiendo más tiempo a las fases de formación.
R04	Cada vez que se fije un punto de dirección en el proyecto, es necesario aclarar todo tipo de dudas y contar con la aceptación de todos los miembros del proyecto.	<p>Teniendo en cuenta que podrían surgir desacuerdos entre los miembros, se establecerá previamente una política de toma de decisiones en caso de desacuerdo:</p> <p>“Las cuestiones relativas a requisitos se tratarán conjuntamente con todas las partes implicadas, que serán quienes tomen las decisiones, con un predominio de la decisión escogida por los ejecutivos de la empresa”.</p>
R05	Realizar estimaciones de los costes en función de diversas herramientas utilizadas como <i>Power BI</i> , <i>Android Studio</i> , <i>Firebase</i> , ampliaciones de licencias del <i>ERP</i> corporativo, etc.	Redimensionar el proyecto a lo largo de su desarrollo, y añadir o eliminar funcionalidades según precise.

La Tabla 4 representa una serie de medidas de prevención y corrección de los distintos riesgos contemplados en el proyecto.

Tabla 5. Control de riesgos.

<b>ID</b>	<b>Responsable</b>	<b>Fecha de Terminación</b>	<b>Estado</b>
R01	Departamento Informático	Fin del Proyecto	Terminado
R02	Gerencia y departamento informático	Fin del Proyecto	Terminado
R03	Departamento Informático	Fin del Proyecto	Terminado
R04	Gerencia y departamento Informático	Fin del Proyecto	Terminado
R05	Departamento Informático	Fin del Proyecto	Terminado

La Tabla 5 de control de riesgos, muestra los responsables de llevar a cabo la gestión de cada uno de los riesgos contemplados en este proyecto.

## Capítulo 3

# Análisis

El análisis abarca las actividades que se han llevado a cabo para identificar, documentar los requisitos y validarlos con los usuarios finales de la aplicación. También incluye el estudio de los datos internos necesarios, el estudio de la integración de los datos entre el *ERP* y la aplicación móvil, y las tecnologías utilizadas en todo el proyecto. A continuación se incluye la documentación.

### 3.1. Definición de requisitos

Para la obtención de requisitos se han utilizado las técnicas de las entrevistas y las historias de usuario, como mencionamos a continuación.

En este proyecto, la gerencia y los comerciales de la empresa actúan como clientes, ya que el sistema a desarrollar ha sido consensuado a partir de las necesidades percibidas a lo largo de los años.

Para que transmitieran sus demandas, se concertaron tres reuniones. Una primera reunión con la gerencia de la empresa para aclarar los objetivos y alcance del proyecto, la cual terminó cuando queden claras y documentadas las peticiones; una segunda reunión con los comerciales, quienes serán los usuarios finales de la aplicación móvil; y finalmente, una última reunión inicial con la asistencia de todas las partes implicadas, para poner en común todos los puntos y empezar la fase de análisis del proyecto.

En estas reuniones se generaron una serie de documentos de requisitos que sirvieron de punto de partida para empezar a desarrollar. Estos documentos se fueron modificando a medida que se presentaron los modelos *UML* y los prototipos a los usuarios finales del sistema.

El resultado de estas reuniones se refleja en el siguiente texto:

“La aplicación debe cargar automáticamente del inicio de sesión los datos del comercial, en este caso el usuario de la aplicación.

Otra de sus funciones es permitir seleccionar al cliente que comercial va a visitar.

Finalmente, para poder registrar y almacenar una visita a un cliente, el comercial tiene que imputar una descripción sobre esta visita. Tras completar estos tres campos mencionados (comercial, cliente y datos sobre la visita) la aplicación ya permitirá registrar la visita.

La gerencia de la empresa ha estipulado estrictamente que el registro de la visita se tiene que imputar después de la visita realizada con el comercial, siendo esta fecha la que tiene que almacenar la aplicación.

Hasta este punto, la aplicación contiene todas las visitas imputadas por los comerciales en su base de datos en la nube, pero hasta que esta información no sea trasladada al *ERP* corporativo no aportará demasiados beneficios. Para ello, para poder trasladar la información desde la aplicación hasta las instalaciones se necesitará un pequeño *back-end* en la sede de la empresa.

Seguidamente, para que los comerciales y los altos cargos ejecutivos de la empresa tengan acceso a la visualización de la información, se crearán las tablas y *pages* necesarios donde se almacena y se visualiza la información respectivamente (objetos del desarrollador de *Microsoft Dynamics NAV*).

Finalmente, una vez que los datos puedan ser trasladados a la sede, se requerirá de la herramienta *Power BI* para poder explotarlos y obtener la información necesaria para poder mejorar el proceso de toma de decisiones de los cargos directivos de la empresa”.

En referencia a las historias de usuario, a continuación se muestran algunas de las tareas que tienen que realizar. Estas funcionalidades se han obtenido de las peticiones realizadas por parte de los comerciales y la gerencia de la empresa.

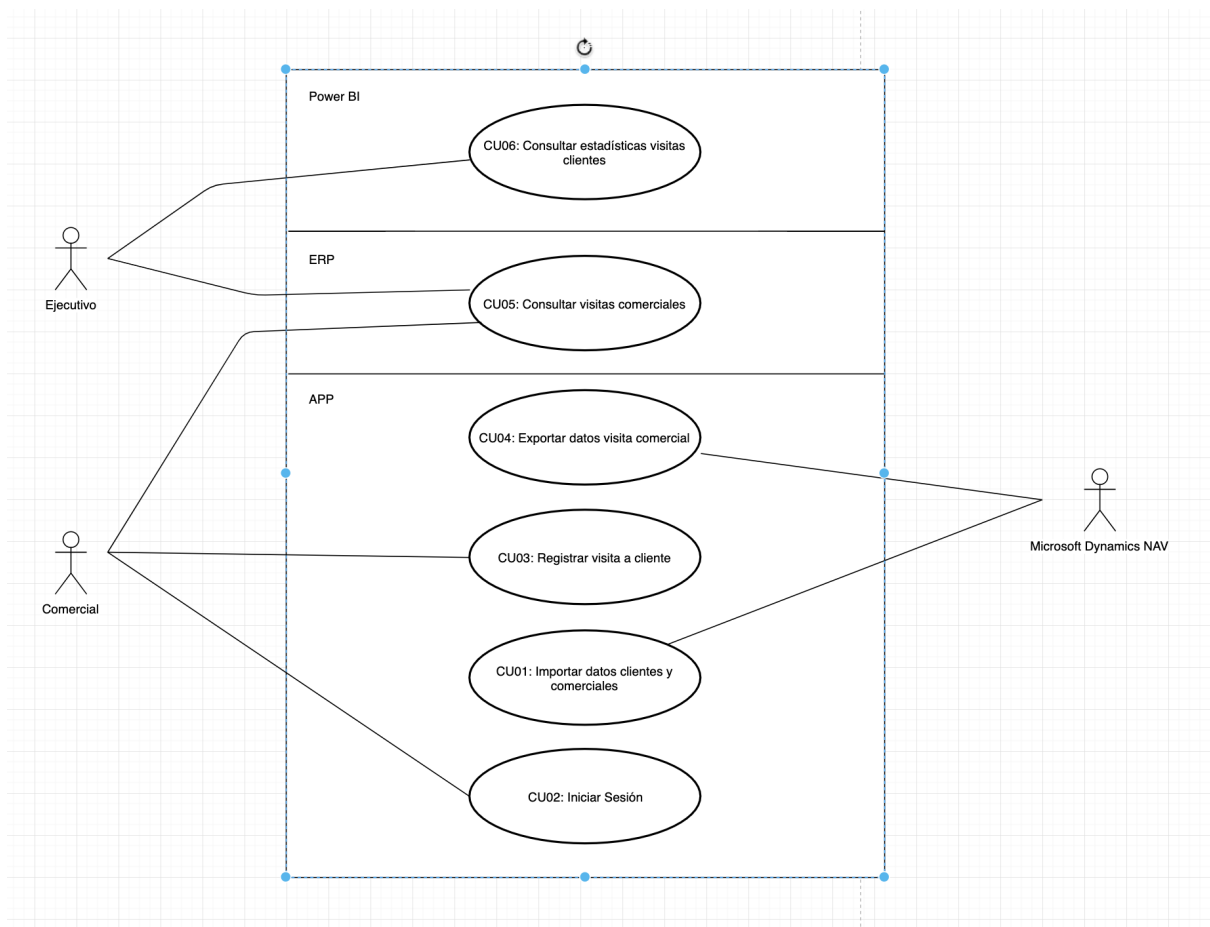
- Como comercial que acabo de terminar una visita con un cliente, quiero registrar esta visita en la aplicación móvil.
- Como gerente de la organización quiero visualizar solo las visitas que está realizando un comercial en concreto.
- Como gerente de la organización quiero consultar información sobre la relación de las ventas de un comercial en concreto y las visitas que realiza a los clientes.
- Como comercial quiero visualizar las visitas que están realizando mis compañeros.

## 3.2. Diagrama de casos de uso

Seguidamente, se realiza el diagrama de casos de uso del proyecto para definir las interacciones entre los usuarios y el sistema, y así se obtiene una visión de las funcionalidades del proyecto.

En la Figura 3 se muestra el diagrama de casos de uso que representa el escenario del producto, con sus actores y la interacción entre estos [8].

Figura 3. Diagrama de casos de uso.



La Figura 3 representa el diagrama de casos de uso; en él, intervienen tres actores, que son: por una parte los actores comercial, ejecutivo y *ERP* corporativo *Microsoft Dynamics NAV*, y por otra parte, los seis casos de uso del proyecto representados en sus respectivos ecosistemas.

El actor comercial, después de autenticarse, se encarga de imputar la visita que realiza al cliente con la aplicación móvil. Esta información se almacena en la nube y se sincroniza con el

*ERP* corporativo. Este actor también puede consultar sus visitas mediante las interfaces desarrolladas en el *ERP* corporativo.

El actor ejecutivo es el responsable de consultar las estadísticas de las visitas de los comerciales a los clientes mediante el *dashboard Power BI* que le será proporcionado. También puede consultar las visitas de los comerciales directamente desde el *ERP* corporativo.

Finalmente, el *ERP* corporativo *Microsoft Dynamics NAV* actuará como actor cuando importe los datos de las visitas de los comerciales desde la aplicación y cuando exporte los datos corporativos de los clientes y los comerciales hacia la aplicación.

A continuación, las Tablas 6 – 17 muestran las descripciones de cada uno de los casos de uso que intervienen en el proyecto y los requisitos de datos de estos.

Tabla 6. Caso de uso: Importar datos clientes y comerciales.

#### **Descripción del caso de uso: CU01 Importar datos clientes y comerciales**

---

<b>Identificador:</b> CU01	<b>Fecha creación:</b> 17/07/2018
<b>Nombre:</b> Importar datos clientes y comerciales	<b>Fecha revisión:</b> 18/07/2018
<b>Autor:</b> Iván Serrano Lluch	<b>Fecha aprobación:</b> 23/07/2018
<b>Fuente:</b> <i>ERP Microsoft Dynamics NAV</i>	<b>Versión:</b> 0
<b>Descripción:</b> El <i>ERP Microsoft Dynamics NAV</i> enviará los datos corporativos necesarios sobre de los clientes y los comerciales a la aplicación.	

---

#### **Pasos secuencia normal:**

1. Ejecución cada 30 minutos.
2. Comprueba si los campos de los clientes y los comerciales están sincronizados.
3. Si no están sincronizados, se envía la información pendiente de actualizar a la aplicación.

#### **Precondición:**

Estar en ejecución y tener acceso a internet.

---

En la Tabla 6 se muestra detalladamente la descripción del caso de uso CU01 Importar datos de los clientes.



Tabla 7. Requisitos del caso de uso: Importar datos clientes y comerciales.

<b>Requisitos de datos</b>	
<b>Identificador:</b> CU01	<b>Fecha creación:</b> 17/07/2018
<b>Nombre:</b> Importar datos clientes y comerciales	<b>Fecha revisión:</b> 18/07/2018
<b>Autor:</b> Iván Serrano Lluch	<b>Fecha aprobación:</b> 23/07/2018
<b>Fuente:</b> <i>ERP Microsoft Dynamics NAV</i>	<b>Versión:</b> 0
<b>Tipos:</b> Datos de entrada	
<b>Datos específicos a mantener:</b> nomComercial, numRecursoCoemrcial, emailCoemrcial, nomCliente, idCliente, idClienteNAV, kilometrosHastaCliente, minutoHastaCliente, provinciaCliente, poblacionCliente, paisCliente, aliasCliente, idDelegacionCliente, clienteMarcador, sincronizado.	
<b>Detalle:</b>	
<b>emailComercial:</b> Es el correo que utilizamos para autenticar, por lo que es un campo obligatorio.	
<b>idDelegacionCliente:</b> Puede estar vacío ya que no todos los clientes son delegaciones.	
<b>clienteMarcador:</b> Puede estar vacío ya que los clientes marcador solo escasos.	
<b>sicnronizado:</b> Tendrá el valor 0 cuando aún no se haya sincronizado el cliente en <i>Firebase</i> , y el valor 1 cuando ya esté sincronizado.	
<b>Comentarios:</b>	
Los datos ya están en el formato deseado ya que son importados a la aplicación móvil desde la base de datos <i>SQL Server</i> del <i>ERP</i> corporativo.	

La Tabla 7 muestra detalladamente los requisitos de datos de CU01 Importar datos clientes y comerciales.

Tabla 8. Caso de uso: Iniciar sesión.

**Descripción del caso de uso: CU02 Iniciar Sesión**

---

<b>Identificador:</b> CU02	<b>Fecha creación:</b> 17/07/2018
<b>Nombre:</b> Iniciar Sesión	<b>Fecha revisión:</b> 18/07/2018
<b>Autor:</b> Ivan Serrano Lluch	<b>Fecha aprobación:</b> 23/07/2018
<b>Fuente:</b> Comercial	<b>Versión:</b> 0
<b>Descripción:</b> El sistema permitirá acceder a la aplicación mediante el inicio de sesión.	

---

**Pasos secuencia normal:**

1. Rellenar el campo de usuario con el correo.
2. Rellenar la el campo contraseña.
3. *LogIn*.

**Excepción:**

Si el usuario no pone un usuario válido se le informará de lo ocurrido.

**Precondición:**

El usuario debe estar previamente registrado en la plataforma.

---

**Comentarios:** Si el usuario es un usuario integrado en el dominio de la empresa, no es necesario que se registre desde la aplicación, ya que su usuario se corresponderá con el del dominio.

En la Tabla 8 se muestra detalladamente la descripción del caso de uso CU02 Iniciar Sesión.

Tabla 9. Requisitos del caso de uso: Iniciar sesión.

<b>Requisitos de datos</b>	
<b>Identificador:</b> CU02	<b>Fecha creación:</b> 17/07/2018
<b>Nombre:</b> Iniciar Sesión	<b>Fecha revisión:</b> 18/07/2018
<b>Autor:</b> Ivan Serrano Lluch	<b>Fecha aprobación:</b> 23/07/2018
<b>Fuente:</b> Comercial	<b>Versión:</b> 0
<b>Tipos:</b> Datos de entrada	
<b>Datos específicos a mantener:</b> emailComercial, contraseñaComercial	
<b>Detalle:</b>	
<b>emailComercial:</b> Este campo debe tener formato de correo electrónico, además de coincidir con el correo de la cuenta del dominio corporativo.	
<b>contraseñaComercial:</b> Este campo debe coincidir con la contraseña del dominio corporativo del usuario.	
<b>Comentarios:</b> El equipo informático será el responsable de que los comerciales utilicen las mismas cuentas del dominio corporativo (con las que ya cuenta la empresa) para la autenticación en la aplicación.	

La Tabla 9 muestra detalladamente los requisitos de datos de CU02 Iniciar Sesión.

Tabla 10. Caso de uso: Registrar visita a clientes.

**Descripción del caso de uso: CU03 Registrar visita a clientes**

---

<b>Identificador:</b> CU03	<b>Fecha creación:</b> 17/07/2018
<b>Nombre:</b> Registrar visita a cliente	<b>Fecha revisión:</b> 18/07/2018
<b>Autor:</b> Iván Serrano Lluch	<b>Fecha aprobación:</b> 23/07/2018
<b>Fuente:</b> Comercial	<b>Versión:</b> 0
<b>Descripción:</b> El sistema permitirá registrar la visita realizada a un cliente en el momento que considere, mediante una aplicación instalada en su teléfono móvil.	

---

**Pasos secuencia normal:**

1. Seleccionar un cliente de la lista desplegable.
2. Escribir una descripción sobre la visita realizada.
3. Comprobar que los datos introducidos son correctos.
4. Finalizar transacción.

**Excepción:**

Si el usuario no rellena algún campo la aplicación no le permitirá acceder a la parte de la aplicación que permite almacenar la visita.

**Precondición:**

El usuario tiene que registrarse con la ayuda y consenso del departamento de informática antes de utilizar la aplicación.

---

**Comentarios:** La lista desplegable de la que se selecciona al cliente tiene un filtro por texto para facilitar su búsqueda.

En la Tabla 10 se muestra detalladamente la descripción del caso de uso CU03 Registrar visitas a clientes.

Tabla 11. Requisitos del caso de uso: Registrar visita a cliente.

<b>Requisitos de datos</b>	
<b>Identificador:</b> CU03	<b>Fecha creación:</b> 17/07/2018
<b>Nombre:</b> Registrar visita a cliente	<b>Fecha revisión:</b> 18/07/2018
<b>Autor:</b> Iván Serrano Lluch	<b>Fecha aprobación:</b> 23/07/2018
<b>Fuente:</b> Comercial	<b>Versión:</b> 0
<b>Tipos:</b> Datos de entrada	
<b>Datos específicos a mantener:</b> nombreComercial, nombreCliente, descripcionVisita, fechaVisita	
<b>Detalle:</b>	
<b>nombreCliente:</b> Se podrá seleccionar de una lista donde aparecen todos los clientes.	
<b>descripcionVisita:</b> El texto introducido deberá tener una longitud máxima de 250 caracteres.	
<b>fechaVisista:</b> El comercial no podrá seleccionar la fecha de la visita, sino que se almacenará automáticamente cuando se guarde la visita.	
<b>Comentarios:</b>	
Por petición de la gerencia, los comerciales solo podrán registrar las visitas a clientes inmediatamente después de terminar la reunión con este, de ahí que el usuario no pueda seleccionar la fecha de la visita.	

La Tabla 11 muestra detalladamente los requisitos de datos de CU03 Registrar visitas a cliente.

Tabla 12. Caso de uso: Exportar datos de las visitas de los comerciales.

<b>Descripción del caso de uso: CU04 Exportar datos visitas comerciales</b>	
<b>Identificador:</b> CU04	<b>Fecha creación:</b> 17/07/2018
<b>Nombre:</b> Exportar datos visitas comerciales	<b>Fecha revisión:</b> 18/07/2018
<b>Autor:</b> Iván Serrano Lluch	<b>Fecha aprobación:</b> 23/07/2018
<b>Fuente:</b> <i>ERP Microsoft Dynamics NAV</i>	<b>Versión:</b> 0
<b>Descripción:</b> El <i>ERP Microsoft Dynamics NAV</i> recibirá de la aplicación los datos de las visitas realizadas por los comerciales.	
<b>Pasos secuencia normal:</b>	
<ol style="list-style-type: none"><li>1. El equipo informático ejecuta el <i>script</i>.</li><li>2. El <i>script</i> se queda a la espera de que salte el evento cuando se inserta una visita</li><li>3. Obtiene la información desde la base de datos en tiempo real de <i>Firebase</i>.</li><li>4. La almacena en la base de datos del <i>ERP</i>.</li></ol>	
<b>Precondición:</b>	
Es necesario contar con un <i>script</i> creado en <i>Node JS</i> corriendo en el servidor de la empresa como si fuera un proceso de <i>Windows</i> . Este <i>script</i> se encarga de enviar la información actualizada hacia el <i>ERP</i> corporativo.	

En la Tabla 12 se muestra detalladamente la descripción del caso de uso CU04 Exportar datos visitas comerciales.

Tabla 13. Requisitos del caso de uso: Exportar datos de las visitas de los comerciales.

<b>Requisitos de datos</b>	
<b>Identificador:</b> CU04	<b>Fecha creación:</b> 17/07/2018
<b>Nombre:</b> Exportar datos visitas comerciales	<b>Fecha revisión:</b> 18/07/2018
<b>Autor:</b> Iván Serrano Lluch	<b>Fecha aprobación:</b> 23/07/2018
<b>Fuente:</b> <i>ERP Microsoft Dynamics NAV</i>	<b>Versión:</b> 0
<b>Tipos:</b> Datos de salida	
<b>Datos específicos a mantener:</b> nombreComercial, recursoComercial, nombreCliente, idClienteNAV, descripciónVisita, fechaVisita, sincronizado	
<b>Detalle:</b>	
<b>sincronizado:</b> Tendrá el valor 0 cuando aún no se haya insertado la visita en el <i>ERP</i> corporativo, y el valor 1 cuando ya esté insertado en el sistema corporativo.	
<b>Comentarios:</b> Este caso de uso se ejecuta cuando se almacena la visita, manteniendo la consistencia entre la base de datos de la aplicación y la base de datos del <i>ERP</i> corporativo.	

La Tabla 13 muestra detalladamente los requisitos de datos de CU04 Exportar datos visitas comerciales.

Tabla 14. Caso de uso: Consultar visitas comerciales.

<b>Descripción del caso de uso: CU05 Consultar visitas comerciales</b>	
<b>Identificador:</b> CU05	<b>Fecha creación:</b> 17/07/2018
<b>Nombre:</b> Consultar visitas comerciales	<b>Fecha revisión:</b> 18/07/2018
<b>Autor:</b> Ivan Serrano Lluch	<b>Fecha aprobación:</b> 23/07/2018
<b>Fuente:</b> Comercial, Ejecutivo	<b>Versión:</b> 0
<b>Descripción:</b> El sistema permitirá consultar y modificar las visitas realizadas por los comerciales mediante interfaces en el <i>ERP</i> corporativo, habilitadas para los comerciales y los ejecutivos.	
<b>Pasos secuencia normal:</b>	
<ol style="list-style-type: none"><li>1. El usuario accede al ordenador de trabajo.</li><li>2. El usuario se autentica en el dominio.</li><li>3. El usuario accede a la Aplicación de Escritorio <i>Microsoft Dynamics NAV</i>.</li><li>4. El usuario se desplaza por el menú hasta la pantalla Consultar Visitas Comerciales.</li><li>5. El usuario consulta las visitas de los comerciales.</li></ol>	
<b>Excepción:</b>	
Si el usuario no tiene el ROL necesario, se le deniega el acceso.	
<b>Precondición:</b>	
Estar autenticado en el dominio de la organización.	
<b>Comentarios:</b> La pantalla de consulta cuenta con los filtros básicos de <i>Microsoft Dynamics NAV</i> .	

La Tabla 14 muestra detalladamente la descripción del caso de uso CU05 Consultar visitas comerciales.



Tabla 15. Requisitos del caso de uso: Consultar visitas comerciales.

<b>Requisitos de datos</b>	
<b>Identificador:</b> CU05	<b>Fecha creación:</b> 17/07/2018
<b>Nombre:</b> Consultar visitas comerciales	<b>Fecha revisión:</b> 18/07/2018
<b>Autor:</b> Ivan Serrano Lluch	<b>Fecha aprobación:</b> 23/07/2018
<b>Fuente:</b> Comercial, Ejecutivo	<b>Versión:</b> 0
<b>Tipos:</b> Datos a mantener	
<b>Datos específicos a mantener:</b> nombreComercial, recursoComercial, nombreCliente, idClienteNAV, descripciónVisita, fechaVisita, visitaVentaRecambio, visitaVentaMaquinaria	
<b>Detalles:</b>	
<b>visitaVentaMaquinaria:</b> Este campo sirve para identificar la visita que ha supuesto una venta de maquinaria. Los comerciales lo modifican desde la interfaz del <i>ERP</i> cuando tengan la información.	
<b>visitaVentaRecambio:</b> Este campo sirve para identificar la visita que ha supuesto una venta de recambios. Los comerciales lo modifican desde la interfaz del <i>ERP</i> cuando tengan la información.	
<b>Comentarios:</b> Mostramos los mismos datos que insertamos en el CU04, excepto el campo sincronizado, que no tiene ninguna relevancia para el usuario.	

La Tabla 15 muestra detalladamente los requisitos de datos de CU05 Consultar visitas comerciales.

Tabla 16. Caso de uso: Consultar estadísticas de visitas comerciales.

<b>Descripción del caso de uso: CU06 Consultar estadísticas de visitas comerciales</b>	
<b>Identificador:</b> CU06	<b>Fecha creación:</b> 17/07/2018
<b>Nombre:</b> Consultar estadísticas de visitas comerciales	<b>Fecha revisión:</b> 18/07/2018
<b>Autor:</b> Iván Serrano Lluch	<b>Fecha aprobación:</b> 23/07/2018
<b>Fuente:</b> Ejecutivo	<b>Versión:</b> 0
<b>Descripción:</b> El sistema debe permitir consultar estadísticas sobre las visitas de los comerciales y la relación con la venta de maquinaria mediante un <i>dashboard</i> en <i>Power BI</i> .	
<b>Pasos secuencia normal:</b>	
<ol style="list-style-type: none"><li>1. Acceder a la aplicación de escritorio <i>Power BI</i>.</li><li>2. Elegir el proyecto indicado.</li><li>3. Consultar estadísticas.</li><li>4. Gestionar filtros.</li><li>5. Exploración datos.</li></ol>	
<b>Comentarios:</b> El usuario tendrá filtros y un <i>dashboard</i> muy amigable, que habrá sido validado previamente con él.	

La Tabla 16 muestra detalladamente la descripción del caso de uso CU06 Consultar estadísticas de visitas comerciales.

Tabla 17. Requisitos del caso de uso: Consultar estadísticas de visitas comerciales.

<b>Requisitos de datos</b>	
<b>Identificador:</b> CU06	<b>Fecha creación:</b> 17/07/2018
<b>Nombre:</b> Consultar estadísticas de visitas comerciales	<b>Fecha revisión:</b> 18/07/2018
<b>Autor:</b> Iván Serrano Lluch	<b>Fecha aprobación:</b> 23/07/2018
<b>Fuente:</b> Comercial, Ejecutivo	<b>Versión:</b> 0
<b>Tipos:</b> Datos a mantener.	
<b>Datos específicos a mantener:</b> nombreComercial, recursoComercial, nombreCliente, codigoCliente, descripciónVisita, fechaVisita, tablaClientes, tablaVentasMaquinaria, tablaVentaRecambios y la tablaComerciales.	
<b>Comentarios:</b> Necesita estar conectado a una fuente de datos para obtener información. En este caso se conectara a la fuente que contiene toda la información de la empresa, la base de datos del <i>ERP</i> corporativo.	

En la Tabla 17 se muestra detalladamente los requisitos de datos de CU06 Consultar estadísticas de visitas comerciales.

### 3.4. Estudio datos internos necesarios

En SACMI IBERICA, al igual que en muchas otras organizaciones, los datos internos con relevancia están almacenados en los servidores de la propia sede, más concretamente en la base de datos del *ERP* corporativo. Esta es una base de datos *SQL Server*, la cual es necesario estudiar para proporcionar los datos necesarios a tres de las partes que forman el sistema: la aplicación móvil, las interfaces del *ERP* corporativo y la herramienta de consulta de datos explotados para el proceso de toma de decisiones ejecutivas.

En cuanto a la aplicación móvil es necesario cubrir los siguientes casos de uso: importar datos clientes y comerciales, exportar datos visitas, registrar visita a cliente e iniciar sesión. Por tanto, las tablas del *ERP* corporativo que es conveniente estudiar para obtener datos en esta parte, son las de “Comerciales” y “Clientes”. También es necesario crear la tabla “Visitas” donde se almacenarán as visitas que registran los comerciales con la aplicación.

Referente a las interfaces del *ERP* corporativo, se deben mostrar las visitas realizadas por los comerciales a los clientes, por lo que será necesaria la tabla “Visitas” mencionada anteriormente.

Finalmente, con la herramienta de explotación de datos elaboramos una serie de paneles de información, que permitan la interacción del usuario para que pueda representar la información cumpliendo sus necesidades de consulta. Estos paneles analíticos necesitarán información de diversas tablas de la empresa para poder representar las peticiones de los ejecutivos, las cuales responden a la siguiente consulta: la relación entre las máquinas/recambios vendidos y el número de visitas comerciales realizadas a los clientes. Por lo que las tablas a consultar serán “VentaMaquinaria”, “VentaRecambios”, “Comerciales”, “Visitas” y “Clientes” del *ERP* corporativo.

Tras el estudio detallado de los datos internos requeridos, se puede observar que todas estas tablas están interconectadas entre sí, formado un sistema muy factible para someterse a la explotación de datos si las disponemos en una estructura de diagrama en estrella.

### **3.5. Integración de datos del ERP hacia la APP**

Una vez localizados los datos corporativos necesarios para la APP, es necesario integrar estos datos en la nueva base de datos de la aplicación móvil y mantener la consistencia entre las bases de datos del *ERP* y la de la aplicación móvil.

Para integrar estos datos es conveniente tener en cuenta que esta aplicación consta de una base de datos en la plataforma *Firebase*, la cual contiene, entre otras cosas, una base de datos de *JSON* en tiempo real, no relacional y en la nube, que será utilizada para alojar los datos de la aplicación *Android*.

Cabe recordar que los datos corporativos de los que disponemos están en formato *SQL* (*Structured Query Language*) y para integrarlos en la base de datos se deben de transformar. Para esto, se creó un pequeño proceso con *Node JS* que transforma los datos corporativos a una estructura *JSON* y los inserta en la base de datos en la nube de *Firebase*.

### **3.6. Integración de datos de la APP hacia el ERP**

Una vez los datos se encuentran cargados en la aplicación móvil, conviene pensar qué hacer con los datos que genere esta aplicación.

En este caso la aplicación sirve para imputar visitas a clientes, por lo que esta aplicación genera datos como: qué comercial ha realizado la visita, a qué cliente, la descripción y el motivo de la visita y la fecha de la misma.

Una vez almacenados estos datos en la base de datos de la aplicación en formato *JSON*, se trasladan estos registros almacenados al *ERP* de la empresa, mediante un *webservice SOAP (Simple Object Access Protocol)* que se publica desde el *ERP* de la empresa. Después un proceso con Node JS será el encargado de transformar los datos en un formato relacional y usar el *webservice* para enviarlos al *ERP*.



## Capítulo 4

# Diseño

En este apartado se muestra cómo se han diseñado los diferentes casos de uso del proyecto y la base de datos de la aplicación móvil. En cuanto al prototipado de diseño de las interfaces gráficas, han sido realizadas mediante prototipos en papel y evaluadas por parte de los usuarios finales, que serán explicadas posteriormente.

### 4.1 Diseño APP

Este apartado se centra en el diseño de la aplicación móvil. Más concretamente en el diseño de la base de datos, la pantalla de inicio de sesión y la interfaz de registro de visita comercial.

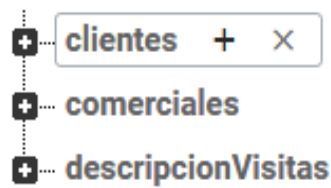
#### 4.1.1. Diseño DB APP

La elección para la plataforma de la base de datos de la aplicación ha sido *Firebase Realtime Database*, la cual es una base de datos en tiempo real, no relacional y almacenada en la nube. En esta base de datos, los datos se almacenan como objetos *JSON*, es decir, esta base de datos puede contextualizarse como un árbol *JSON* alojado en la nube. Por tanto, para diseñarla es necesario pensar cómo almacenar los datos en una estructura *JSON*.

Como la aplicación trabajará con los datos de los clientes, los comerciales y las visitas, se han almacenado estos datos jerárquicamente dentro de la raíz `proyectoVisitasComerciales`. De esta raíz cuelgan los árboles clientes, comerciales y visitas como se muestra en la figura 4.

A su vez, dentro del árbol padre `Cientes`, se almacenan todos los clientes ordenados por su código de relevancia para la organización.

Figura 4. Raíz proyectoVisitasComerciales.



La Figura 4 representa los padres de cada entidad utilizada en el proyecto. Estas entidades contienen datos de los clientes y comerciales de la sede, y datos de la visita realizada.

Dentro de cada cliente se pueden encontrar sus datos en detalle, como se puede apreciar en la figura 5.

Figura 5. Árbol padre Cliente.



La Figura 5 representa la entidad Cliente con los campos que contiene.

Dentro del árbol padre Comerciales se almacenan todos los comerciales que utilizan la aplicación. Dentro de cada comercial, se pueden encontrar a su vez sus datos personales, que



son su correo, su nombre y su código identificativo en la organización, tal y como se muestra en la Figura 6.

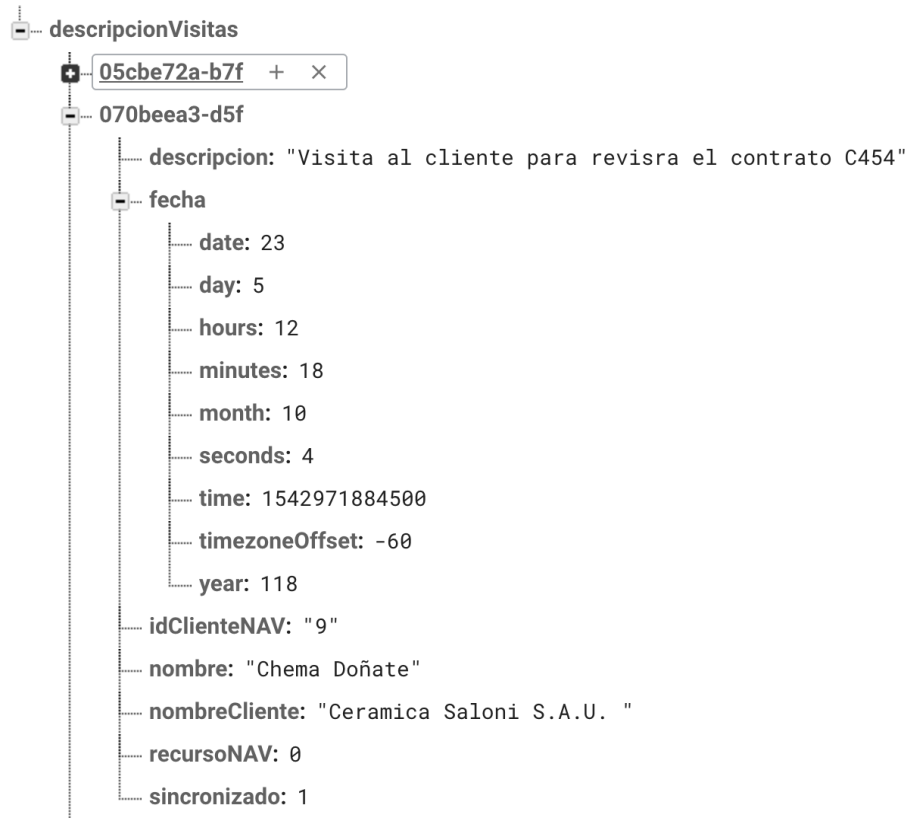
Figura 6. Árbol padre Comerciales.



La Figura 6 representa la entidad Comercial con los datos que contiene.

Dentro del árbol Visitas se encuentran almacenadas todas las visitas realizadas por los comerciales. Estas visitas almacenan detalles de la visita como la fecha de realización, la descripción de la visita, el comercial que la ha realizado, el cliente, el código identificativo en la organización del cliente y el comercial, etc. Esto se puede observar en la Figura 7.

Figura 7. Árbol padre Visitas.



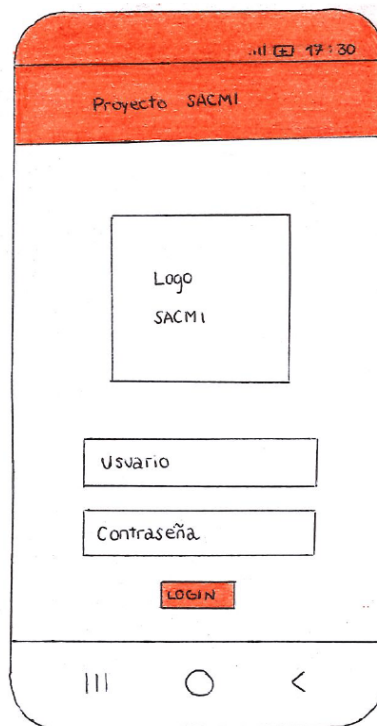
La Figura 7 representa la entidad Visita con la información que contiene, que ha sido previamente registrada por el comercial.

#### 4.1.2. Diseño Inicio Sesión, APP

Para realizar el diseño de la pantalla de inicio de la aplicación se ha aplicado la técnica de los prototipos en papel.

Se realizó un prototipo en papel y se validó con los comerciales, que son los usuarios finales de la aplicación. En este prototipo se mostró la pantalla inicial de la aplicación móvil donde los usuarios tienen que autenticarse para iniciar sesión y poder registrar una visita comercial. Esta interfaz gráfica contiene en la parte superior de la pantalla, el logo de la empresa, y en la parte inferior de la pantalla se pueden encontrar los campos que los usuarios tienen que rellenar para autenticarse. Estos campos son el correo y la contraseña. Finalmente, se encuentra el botón para hacer el *LogIn*. El diseño de este prototipo se muestra en la Figura 8.

Figura 8. Prototipo interfaz *LogIn* APP



La Figura 8 representa el prototipo de la interfaz del *LogIn* de la aplicación móvil. En ella se puede encontrarla ubicación del logo de la empresa, los campos usuario y contraseña a rellenar, y el botón de *LogIn*.

### 4.1.3. Diseño registro visita comercial, APP

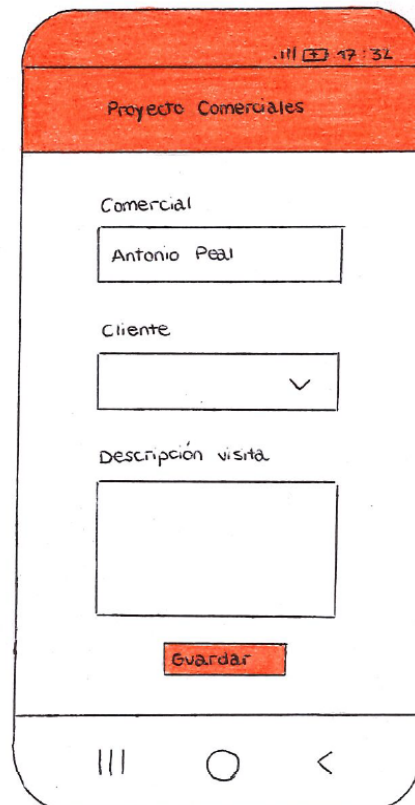
Para realizar el diseño de la pantalla principal de la aplicación que se encarga de registrar las visitas realizadas por los comerciales a los clientes se ha aplicado la técnica de los prototipos en papel.

El prototipo realizado fue validado junto los comerciales, que son los usuarios finales de la aplicación.

Este prototipo consta de dos interfaces. Una primera interfaz donde se registran y son almacenados los datos de la visita, y otra posterior donde se pueden consultar los datos introducidos a modo de informe.

La primera interfaz consta de un campo que contiene el nombre del comercial que ha iniciado sesión en la aplicación, más abajo un desplegable donde se puede seleccionar el cliente al que se ha realizado la visita, seguido de un campo de texto donde se puede escribir una descripción de la visita. Finalmente se encuentra el botón que almacena la visita. El diseño completo de la interfaz se muestra en la figura 9.

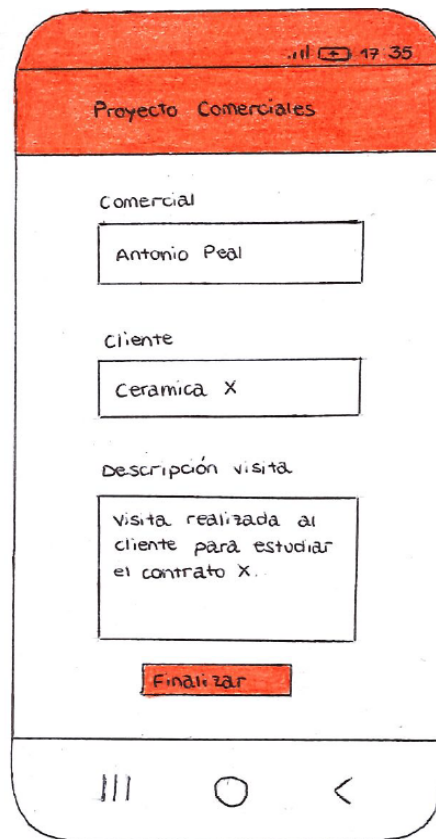
Figura 9. Prototipo interfaz registro de visita APP.



La Figura 9 muestra el prototipo de la interfaz de registro de visita. En ella, se puede encontrar el nombre del comercial que ha iniciado sesión, un campo seleccionable del que elegir el cliente, otro campo donde cumplimentar la descripción de la visita y un botón para guardar y almacenar la visita.

En la segunda interfaz se pueden observar los mismos datos que introducidos en la primera interfaz añadiendo la fecha del momento en que se ha registrado la visita, que tiene que coincidir con la fecha de finalización de la reunión/visita. Esto está estipulado por la gerencia de la organización. El diseño de la interfaz se muestra en la Figura 10.

Figura 10. Prototipo interfaz visualización de datos almacenados APP.



La Figura 10 muestra el prototipo de la interfaz de visualización de datos almacenados. En ella, se observa el nombre del comercial que ha registrado la visita, el cliente al que se le ha realizado la visita, la descripción de la visita y un botón para finalizar la transacción.

## 4.2. Diseño *back-end*

Este apartado se centra en el diseño de los procesos que conforman el *back-end*, los cuales se encargan de la integración de la información entre la base de datos no relacional de la aplicación móvil y la base de datos relacional del ERP corporativo.

### 4.2.1 Diseño proceso integración datos, *ERP* → *APP*

En este punto se va a diseñar un proceso en *Node JS*, que exporte desde el *ERP* corporativo datos internos sobre clientes y comerciales, y los importe a la aplicación móvil. Este proceso se tendrá que conectar a la base de datos del *ERP* corporativo y a la base de datos

de la aplicación para realizar el intercambio de datos. También es necesario tener en cuenta que se debe saber qué clientes ya se han sincronizado con la aplicación cuando añadimos algún cliente al *ERP* corporativo. Para ello, se ha añadido un campo sincronizado en la tabla clientes del *ERP* de la empresa que informará de si el cliente está ya introducido en la aplicación o no.

Este proceso se alojará en los servidores de la empresa como un servicio integrado de *Windows*, haciendo transparente en el servidor y pudiendo acceder a él cada vez que se necesite ejecutar, por lo que no se requiere de ninguna interfaz gráfica.

#### **4.2.2. Diseño proceso integración datos, APP → ERP**

Se trata de diseñar un proceso en *Node JS* que ejercerá de *back-end*, el cual se conecta a la base de datos de la aplicación, con el fin de que extraiga las visitas realizadas por los comerciales y las inserte en la base de datos del *ERP* corporativo. Este evento se producirá gracias a un campo sincronizado que mantendrá la consistencia entre las dos bases de datos.

Este proceso envía a la aplicación los datos de las visitas de los comerciales realizadas a los clientes de la organización, para que puedan ser gestionados, y se aloja en los servidores de la empresa como un servicio integrado de *Windows*, haciendo transparente en el servidor.

### **4.3. Diseño interfaz consulta de visitas, ERP**

Para el diseño de la interfaz de consultas de visitas se aplica la técnica de los prototipos, con las herramientas de desarrollo del *ERP Microsoft Dynamics NAV*.

En esta interfaz se muestran el comercial que ha realizado la visita, el cliente con el que se ha reunido, la descripción de la visita, la fecha de la visita, un identificador de si la visita ha supuesto una venta de maquinaria o recambios, el código identificativo del comercial del *ERP*, y el código identificativo de cliente del *ERP*. Además, en la parte superior se pueden encontrar los filtros. El diseño de la interfaz se muestra en la Figura 11.

Figura 11. Interfaz *ERP Microsoft Dynamics NAV*.

nombre	descripcion	recursoNAV	fechaVisita	codigoCliente	nombreCliente	ventaRecambio	ventaMaquinaia
Juanito Gomanio	Visita realizada para concretar los puntos del posible contrato C22553	RC36	12/02/2019	C243	EL MOLINO	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Juanito Gomanio	Visita realizada para concretar los puntos del posible contrato C22553	RC36	10/01/2019	C456	ARTUR	<input type="checkbox"/>	<input type="checkbox"/>
Juanito Gomanio	Visita realizada para concretar los puntos del posible contrato C23534343	RC36	30/05/2019	C33	EL POZO	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Mona Lee	Visita realizada para concretar los puntos del posible contrato C2353	RC30	12/12/2018	C4543	AZULEV	<input type="checkbox"/>	<input type="checkbox"/>
Juana Archi	Visita realizada para concretar los puntos del posible contrato C2353454	RC37	12/09/2018	C755	FABRESA	<input type="checkbox"/>	<input type="checkbox"/>
Manu Tino	Visita realizada para concretar los puntos del posible contrato C2353635	RC3	05/05/2018	C243	EL MOLINO	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Manu Tino	Visita realizada para concretar los puntos del posible contrato C23536354	RC3	06/06/2018	C755	FABRESA	<input type="checkbox"/>	<input type="checkbox"/>
Mona Lee	Visita realizada para concretar los puntos del posible contrato C2353635	RC30	04/04/2018	C33	EL POZO	<input type="checkbox"/>	<input type="checkbox"/>
Mona Lee	Visita realizada para concretar los puntos del posible contrato C23536353	RC30	04/05/2018	C33	EL POZO	<input type="checkbox"/>	<input type="checkbox"/>
Juana Archi	Visita realizada para concretar los puntos del posible contrato C2353635	RC37	03/03/2018	C456	ARTUR	<input type="checkbox"/>	<input type="checkbox"/>
Juana Archi	Visita realizada para concretar los puntos del posible contrato C23536355	RC37	03/03/2018	C456	ARTUR	<input checked="" type="checkbox"/>	<input type="checkbox"/>

En la Figura 11 se representa la interfaz del *ERP Microsoft Dynamics NAV*, donde los usuarios consultarán las visitas de los comerciales.

Esta interfaz no ha sido validada con el usuario final, ya que sigue el formato estándar del *software ERP Microsoft Dynamics NAV* y los usuarios están familiarizados con este producto.

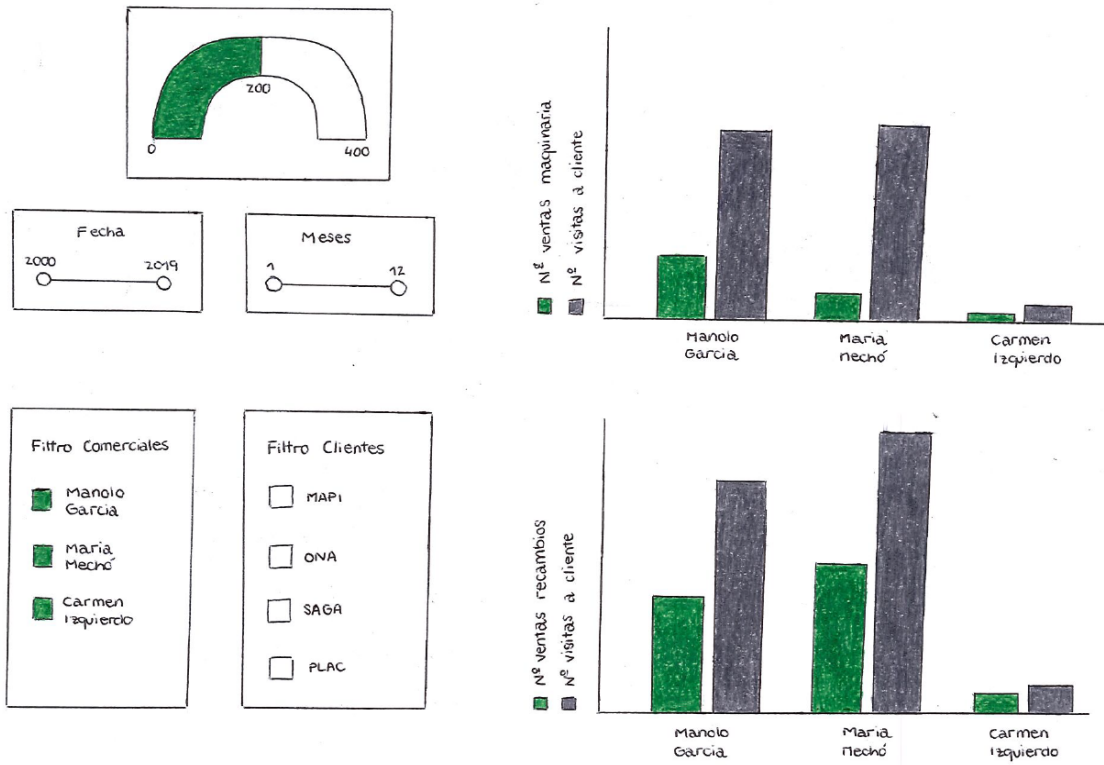
#### 4.4. Diseño *dashboard Power BI*

Para realizar el diseño del *dashboard* de apoyo a la toma de decisiones ejecutivas, se ha aplicado la técnica de los prototipos en papel. El prototipo se puede observar en la Figura 12.

En este prototipo se muestran un indicador, dos gráficas y cuatro filtros. El primer indicador, situado en la parte superior izquierda, representa el coste total de la cantidad de ventas realizadas por la organización, un dato relevante para el equipo ejecutivo. Seguidamente en el lado derecho del prototipo se encuentran, en la parte superior, un gráfico que representa el número de ventas de maquinaria por comercial y número de visitas realizadas por comercial, mientras que en la parte inferior se representa el número de ventas de recambios por comercial y el número de visitas realizadas por comercial.

En cuanto a los filtros, se observan filtros de año, de mes, de clientes y de comerciales. Todos estos filtros se pueden combinar entre sí ofreciendo información más precisa y específica.

Figura 12. Prototipo *dashboard* de *Power BI*.



La Figura 12 representa el diseño del prototipo del *dashboard* de *Power BI*, formado por un indicador de costes, dos gráficas sobre ventas y visitas realizadas por los comerciales, y cuatro filtros (año, mes, clientes y comerciales).



## Capítulo 5

# Implementación y pruebas

Este apartado es una descripción técnica del proyecto y en él se puede ver cómo se han implementado los diferentes casos de uso del producto y cómo se han realizado las distintas pruebas del sistema, así como detalles técnicos sobre el desarrollo del sistema. Cabe mencionar que todos los datos y la información tratada en las imágenes de este apartado son generados, por lo que cualquier parecido con la realidad es pura coincidencia.

### 5.1. Implementación APP

Este punto se centra en la implementación de la parte de la aplicación móvil. En él, encontramos los detalles de la implementación de la interfaz del inicio de sesión y sus funcionalidades, y la interfaz de registro de visitas y sus funcionalidades.

#### 5.1.1. Inicio Sesión, APP

En este punto se va a implementar una aplicación móvil que permita iniciar sesión a un usuario registrado. Para ello, se ha utilizado el *IDE* de desarrollo *Android Studio*, y dentro de este, para desarrollar las interfaces gráficas y las funcionalidades se han utilizado el lenguaje de programación *Java* y el lenguaje de marcado *XML* respectivamente.

El registro de los usuarios que pueden acceder a la aplicación móvil, se hará previamente desde la plataforma *Firebase*. Como ya se ha comentado, esta plataforma también tiene varios servicios de autenticación de usuarios.

Para agregar esta librería de autenticación de usuarios de *Firebase* en el *IDE* de *Android Studio* al proyecto, conviene realizar una serie de pasos, que proporcionan las funcionalidades necesarias para poder importar la librería y autenticar desde el código de la aplicación móvil a los usuarios. Para añadir esta librería al proyecto, se deben seguir los siguientes pasos [11]:

1. Iniciar sesión en la consola de *Firebase*.
2. Registrar tu app en *Firebase*.
3. Agregar un archivo de configuración de *Firebase* en el *Gradle* (sistema de compilación) de *Android Studio*.
4. Agregar los *SDK* (Kit de Desarrollo de *Software*) de *Firebase* a la APP.

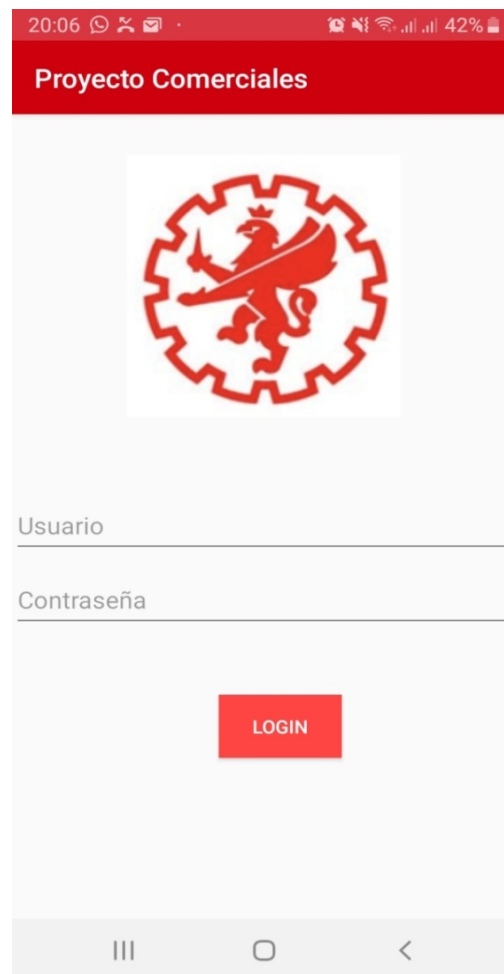
En referencia a los datos encapsulados, tratados y gestionados en estas interfaces, como por ejemplo el correo del usuario o la contraseña, todos ellos están utilizando el *SDK* de *Android Studio*, que ofrece las *APIs* (*Application Programming Interface*) de *Android* para facilitar el desarrollo de aplicaciones móviles *Android*.

También cabe destacar que se ha implementado una pantalla de *Splash*, que suaviza el acceso a la aplicación haciéndola más agradable. Esta interfaz se muestra en la figura 13. Seguidamente en la figura 14 se encuentra la interfaz de inicio de sesión de la aplicación.

Figura 13. Interfaz *Splash* APP



Figura 14. Interfaz Inicio de Sesión APP

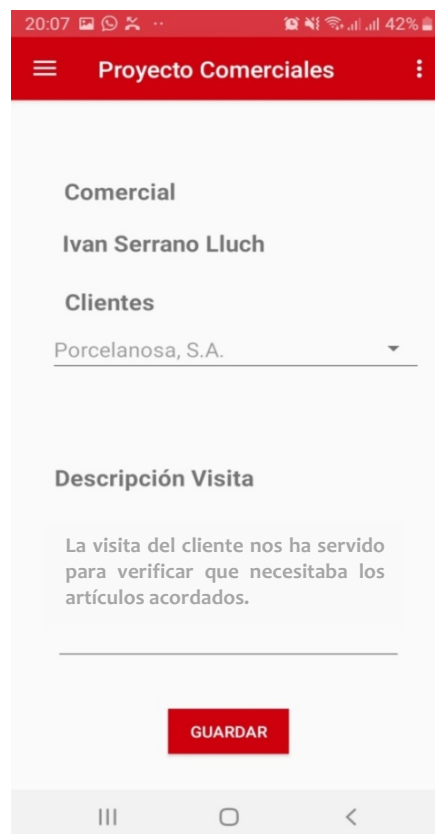


## 5.1.2. Implementación registro visita comercial, APP

En este punto seguimos con el desarrollo de la aplicación móvil. El inicio de sesión de la aplicación ya está implementado, por lo que ahora es necesario centrarse en la parte principal de la aplicación, que es el registro de visitas comerciales. Esta parte principal se pueden diferenciar a su vez dos partes; una primera donde los comerciales podrán registrar y almacenar una visita, y una segunda interfaz que mostrará la información almacenada como ya se ha comentado en la fase de diseño.

En cuanto a los datos que se requieren en estas interfaces, la primera interfaz requiere gestionar los datos del nombre del comercial, el recurso del comercial, el nombre del cliente, el código identificativo del cliente, la descripción de la visita y la fecha en la que se ha realizado la visita; mientras que la segunda interfaz solo requiere gestionar los datos del nombre del comercial, el nombre del cliente, la descripción de la visita y la fecha. Estas interfaces se muestran en las Figuras 15 y 16 respectivamente.

Figura 15. Interfaz registro de la visita.



The screenshot shows a mobile application interface with a red header bar containing the text "Proyecto Comerciales". Below the header, the form is organized into sections: "Comercial" with the name "Ivan Serrano Lluch", "Clientes" with a dropdown menu showing "Porcelanosa, S.A.", and "Descripción Visita" with the text "La visita del cliente nos ha servido para verificar que necesitaba los artículos acordados." At the bottom of the form is a red button labeled "GUARDAR". The status bar at the top shows the time as 20:07 and battery level at 42%.

La Figura 15 representa una interfaz donde se registran los datos de la visita comercial, es decir, comercial que realiza la visita, cliente y una breve descripción.

Figura 16. Interfaz registro almacenado de la visita.

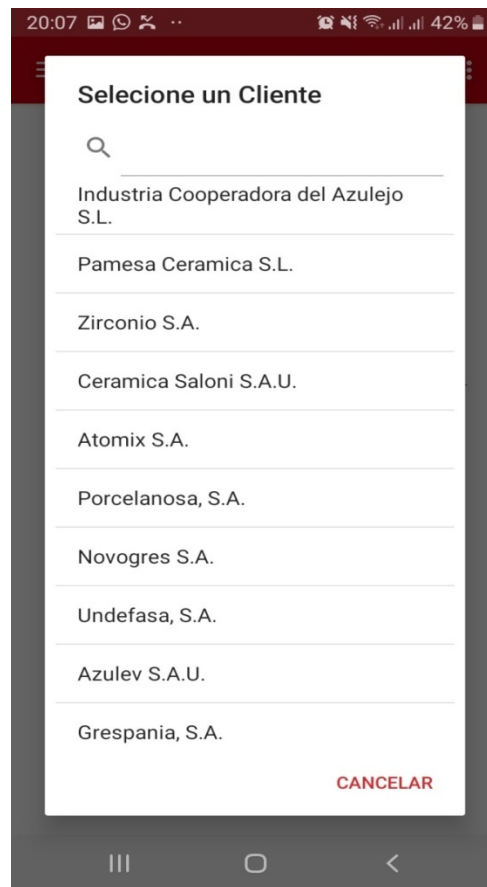


La Figura 16 muestra una interfaz donde se representan los datos de la visita comercial que se ha almacenado previamente en el sistema.

Los datos requeridos por la primera interfaz los obtenemos de la siguiente manera:

- Los datos de los comerciales y de los clientes se encuentran almacenados en la base de datos de la aplicación móvil, por lo que es imprescindible conectarse a *Firestore* para obtener el comercial que realiza la visita y volverse a conectar para obtener los clientes, y cargarlos en el desplegable seleccionable. Dicho desplegable se muestra en la Figura 17. Esta conexión a *Firestore* se puede hacer efectiva gracias a que *Android* ya se encuentra añadido al proyecto, el *SDK* de *Firestore*. De esta manera, utilizando la *API* de *Firestore*, se podrán realizar consultas a la base de datos y obtener los datos necesarios.
- En cuanto a los datos de la descripción de la visita, el usuario es el encargado de rellenarlos.
- Los datos de la fecha de la visita se imputan automáticamente en el momento que se almacena la visita, con la respectiva fecha.

Figura 17. Lista seleccionable Clientes.



La Figura 17 representa la lista seleccionable del campo cliente. En la parte superior de la imagen se encuentra un filtro de texto que facilita la búsqueda del cliente visitado. Estos clientes se han cargado de la base de datos de la aplicación.

Una vez ya están rellenos todos los campos en esta primera interfaz, se procederá a almacenar la visita. Para ello, existe un botón que almacena la visita directamente en la base de datos de la aplicación y además envía los datos almacenados a la segunda interfaz, así se ahorra una consulta a la base de datos en esta segunda pantalla.

Los datos requeridos de esta segunda pantalla se obtienen de la interfaz anterior, que se los envía con la utilización de la *API* de *Android*.

Finalmente, en esta segunda pantalla únicamente se tienen que mostrar los datos a modo informe. Además, se ha añadido un botón que devuelve a la pantalla principal, finalizando la transacción de la operación del usuario.

También cabe destacar que se ha implementado la primera interfaz para que si no se ha rellenado algún campo obligatorio, como el nombre del cliente o la descripción de la visita, no será posible almacenar la visita e indicará lo ocurrido. También se ha optimizado la navegación lógica de la aplicación móvil, de esta manera no se retrocede a pantallas con poca relevancia, ni se accede a interfaces de un solo uso. Asimismo, se han implementado mensajes informativos de las acciones que se van realizando, por ejemplo, cuando un cliente intenta salir de la aplicación le pregunta que si está seguro de salir de la aplicación, o cuando va a almacenar la visita vuelve a pedir confirmación, aunque los campos estén correctamente registrados. Algunas de estas funcionalidades se representan en las Figuras 18, 19, 20 y 21 de las interfaces de la aplicación móvil.

Figura 18. Mensaje informativo I.

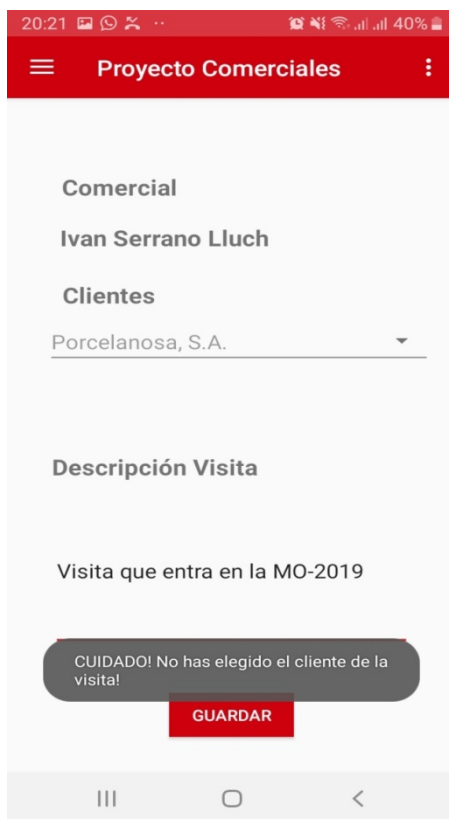
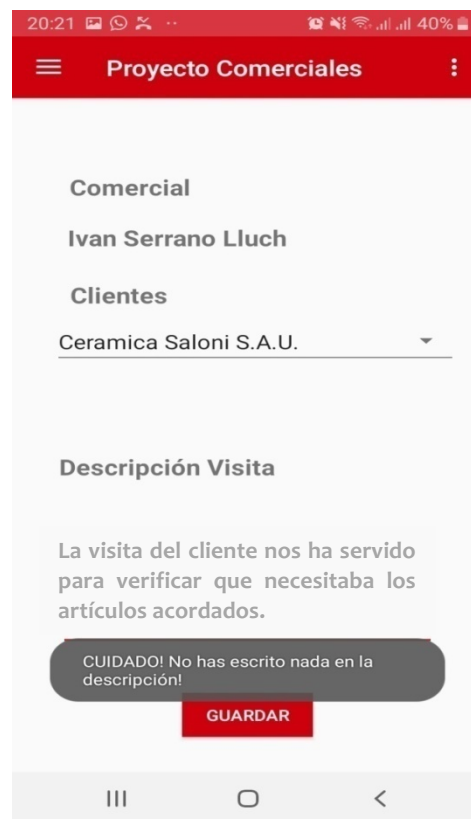
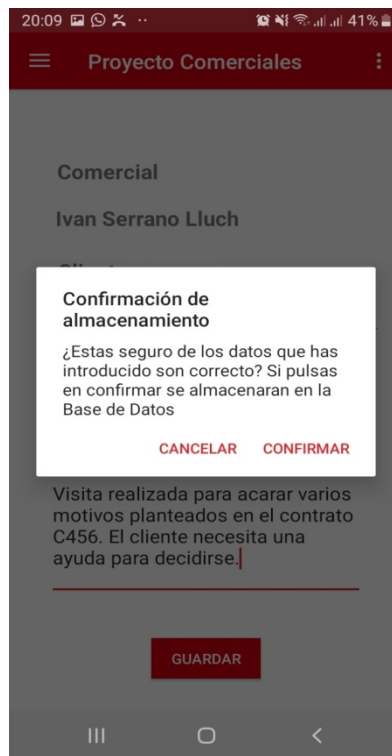


Figura 19. Mensaje informativo II.



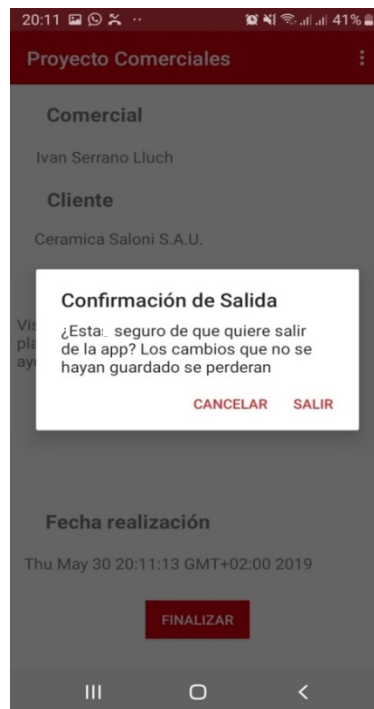
Las Figuras 18 y 19 representan la pantalla principal de la aplicación cuando se intenta almacenar una visita sin haber elegido un cliente o sin haber escrito nada en la descripción de la visita.

Figura 20. Almacenaje de Visita.



La Figura 20 muestra la interacción con la aplicación cuando se selecciona el botón guardar y se intenta almacenar una visita presuntamente válida. La aplicación pide una validación.

Figura 21. Confirmación de salida.



En la Figura 21 se muestra se la interacción del usuario con la aplicación cuando el usuario intenta salir.

En referencia a los datos encapsulados, tratados y gestionados en estas interfaces, como por ejemplo el nombre del comercial, los clientes, la descripción de la visita o la fecha, utilizan todos el *SDK* de *Android Studio*, el cual ofrece las *APIs* de *Android* para facilitar el desarrollo de aplicaciones móviles *Android*.

## 5.2. Implementación *back-end*

Este apartado se centra en la implementación del *back-end* de la aplicación móvil. En él, se pueden encontrar los detalles de la implementación de los procesos encargados de la integración de la información entre la base de datos de la aplicación y la base de datos del *ERP* corporativo.

### 5.2.1 Implementación del proceso de integración de datos, *ERP*→*APP*

En este punto se debe implementar un proceso que forma parte de un pequeño *back-end* en *Node JS* que importe datos desde la base de datos *SQL Server* del *ERP* corporativo *Microsoft Dynamics NAV* hasta la base de datos de la aplicación móvil que está alojada en la plataforma *FireBase*.

Este proceso conecta y mantiene sincronizadas a las dos bases de datos. Primeramente, la base de datos *SQL Server* del *ERP* corporativo, que se utilizará en este caso para extraer los datos de los comerciales y los clientes de la organización. Y por otro lado, la base de datos de la aplicación móvil que está en la plataforma *Firebase*, formada, entre otras capas, por la base de datos en tiempo real. La sincronización se realiza mediante un campo sincronizado que tienen la entidad Comercial y la entidad Cliente, el cual tiene un 0 si no se ha sincronizado y un 1 cuando esté sincronizado.

En este proceso cabe tener en cuenta que cada una de estas dos bases de datos almacena la información con un formato diferente, es decir, la plataforma *FireBase* almacena los datos en formato *JSON*, siendo una base de datos no relacional, mientras que la base de datos *SQL Server*



del *ERP* corporativo es una base de datos relacional que almacena los datos de manera estructurada. Por lo que será necesario realizar una transformación de la información almacenada para poder importarla de una base de datos a la otra y mantener la consistencia entre bases de datos.

Este proceso se implementa utilizando la tecnología *Node JS*, que ofrece un entorno *JavaScript* del lado del servidor como se ha mencionado en el apartado de las tecnologías utilizadas.

Para la implementación de este proceso se precisa de la instalación de una serie de librerías para cubrir las conexiones a las bases de datos mediante el gestor de paquetes *NPM* (*Node Package Manager*) de *Node JS*. Este gestor de paquetes también está explicado en el apartado de tecnologías utilizadas.

### **Librerías utilizadas**

- Paquete *firebase-admin*: Módulo *NPM Node JS*. Cliente completo de *JavaScript* de *Firebase*. Incluye soporte para la autenticación, para la base de datos en tiempo real, almacenamiento y mensajería en la nube. Permite el acceso a los servicios de *Firebase* desde entornos privilegiados [8].
- Paquete *mysql*: Controlador *Node JS* para *mysql*. Permite conexiones a distintas bases de datos [9].

### **Funcionamiento general del proceso, en el *Back-end***

Finalmente este proceso que forma parte del *back-end*, estará en ejecución en los servidores de la empresa. Cada 30 minutos, consultará los datos de la tabla Clientes y Comerciales de la base de datos *SQL Server* del *ERP* corporativo que no estén sincronizados, gracias a unos métodos que nos ofrece la librería *mysql*. Estos datos extraídos serán tratados para convertirlos a un formato *JSON* entendible para la base de datos en tiempo real de la plataforma *FireBase*. Una vez convertidos estos datos serán enviados a la base de datos de la aplicación gracias a unos métodos que nos ofrece la librería *firebase-admin*.

## 5.2.2. Implementación del proceso de integración de datos, APP→ERP

En este punto también es necesario implementar un proceso que forma parte de un pequeño *back-end* con *Node JS* que importa datos desde la base de datos de la aplicación móvil, alojada en la plataforma *Firebase*, hasta la base de datos *SQL Server* del *ERP* corporativo *Microsoft Dynamics NAV*.

Este proceso conecta y mantiene sincronizadas las dos bases de datos. Por un lado, a la base de datos en tiempo real de la aplicación, alojada en la plataforma *Firebase* que se utilizará en este caso para extraer los datos de las visitas comerciales. Y por otro lado, a la base de datos *SQL Server* del *ERP* corporativo *Microsoft Dynamics NAV*, en la que se insertarán los registros de las visitas. La sincronización se realiza mediante un campo sincronizado que tiene la entidad *Visita*, la cual tiene un 0 si no se ha sincronizado y un 1 cuando esté sincronizado.

En este proceso también se debe tener en cuenta que cada una de estas dos bases de datos almacena la información con un formato diferente, es decir, la plataforma *Firebase* almacena los datos en formato *JSON*, siendo una base de datos no relacional, mientras que la base de datos *SQL Server* del *ERP* corporativo es una base de datos relacional que almacena los datos de manera estructurada. Por lo que se deberá realizar una transformación de los datos para poder importarlos de una base de datos a la otra y mantener la consistencia entre bases de datos, pero esta vez a la inversa que en el otro proceso del *back-end* mencionado anteriormente.

En cuanto a la importación de los datos hasta la base de datos *SQL Server* del *ERP* corporativo, se realizará mediante un *webservice SOAP*. Este *webservice* está publicado en el *ERP* corporativo. También está implementado en el *ERP* corporativo la inserción de estos datos en la tabla *Visitas* del *ERP*. Esta implementación se ha realizado mediante el *IDE* desarrollador *Microsoft Dynamics NAV* en uno de sus objetos llamados *CodeUnit*.

Para la implementación del proceso de integración de datos de la APP hasta el ERP se deberán instalar las siguientes librerías:

- Paquete *firebase-admin*: Módulo *NPM Node JS*. Cliente completo de *JavaScript* de *Firebase*. Incluye soporte para la autenticación, para la base de datos en tiempo real, almacenamiento y mensajería en la nube. Permite el acceso a los servicios de *Firebase* desde entornos privilegiados [9].
- Paquete *httpntlm*: Biblioteca de *Node JS* utilizada para realizar la autenticación *NTLM* (*Windows New Technology Local Area Network Manager*) de *Windows* mediante *HTTP*

(Protocolo de Transferencia de Hipertexto) [12]. De esta manera se autentican la conexión contra el *ERP* a la hora de la inserción.

- Paquete *soap-ntlm*: Esta biblioteca permite conectarse a servicios web mediante el protocolo estándar *SOAP* [13]. De esta manera, el *back-end* se conecta con el *ERP* corporativo y le envía la información para que este la gestione.

### **Funcionamiento general del proceso, en el *back-end***

Finalmente, este proceso que forma parte del *back-end*, estará ejecutándose en los servidores de la empresa de forma constante. Cuando se inserte una visita comercial, y salta el evento ofrecido por la *API* de *Firebase* que controla el campo sincronizado, consultará los datos de *Firebase* de las visitas realizadas por los comerciales que no estén sincronizados, y estos datos extraídos serán insertados en la base de datos *SQL Server* del *ERP* corporativo gracias a un *webservice SOAP*. Para finalizar, el *ERP* corporativo será el encargado de gestionar los datos que reciba insertándose en la tabla correspondiente.

## **5.3. Implementación interfaz de consulta de visitas, *ERP***

En este punto se va a desarrollar una interfaz en el *ERP* corporativo *Microsoft Dynamics NAV* que permita consultar las visitas de los clientes que están almacenadas en la base de datos *SQL Server*. La interfaz es una pantalla básica del *ERP* que listara todas las visitas de los respectivos clientes.

Para el desarrollo de esta interfaz se utiliza el *IDE* de desarrollo de *Microsoft Dynamics NAV*. Esta plataforma cuenta con el *Object Designer*, que es la herramienta que todo programador de este *ERP* utiliza para crear, modificar o eliminar objetos.

El *Object Designer* cuenta con los siguientes objetos: *Table*, *Page*, *Report*, *CodeUnit*, *Query*, *XMLPort* y *MenuSuite*.

En el proyecto se van a utilizar los objetos *Table*, *Page* y *CodeUnit* de entre los que provisiona *Microsoft Dynamics NAV*, pero para la implementación de la interfaz de consulta de visitas en el *ERP* sólo se utilizan el objeto *Table*, que sirve para relacionar los datos que se

encuentran almacenados en la base de datos *SQL Server* del *ERP* corporativo con el desarrollador de *Microsoft Dynamics NAV*, y el objeto *Page*, que sirve para implementar la parte visual de la información.

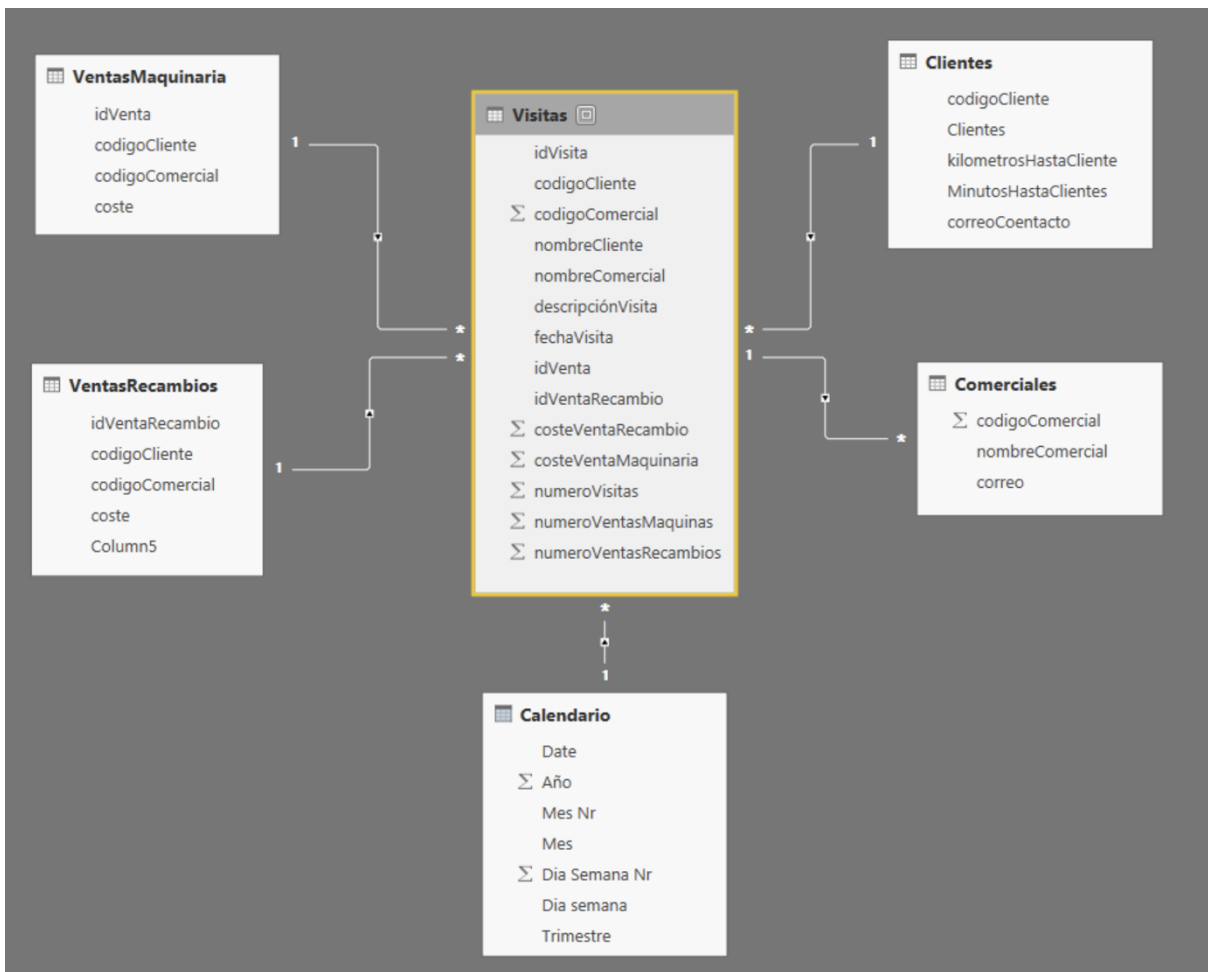
El proceso de implementación ha sido el siguiente:

1. Primero se creó el objeto *Table*, que almacenará los datos. Para ello, se necesitará añadir una numeración a la tabla, en *Microsoft Dynamics NAV*. Las tablas que se crean después de la implantación se numeran a partir de la numeración 50.000. Seguidamente, se añadirá la interpretación del contenido de la tabla, es decir, los campos y sus tipos. Esto directamente generará las respectivas tablas en la base de datos *SQL Server*.
2. Finalmente, se creó el objeto *Page*, definido por la lógica del diseño del *ERP*, por lo que lo único que fue necesario realizar fue enlazarla con la tabla creada, indicar que es de tipo lista y directamente se generó la interfaz del *ERP*.

## **5.4. Implementación *dashboard Power BI***

*Power BI* solo necesita que se le abastezca con datos para empezar a trabajar. Esto se realiza con mucha facilidad con su interfaz visual e intuitiva. Por tanto, se conectó mediante *Power BI* con las tablas y los campos que se requerían de la base de datos *SQL Server* del *ERP* y se cargaron estos datos en la herramienta *Power BI* para poder ser explotados. Una vez ya creadas las tablas con su contenido en la herramienta, se implementaron las relaciones en una estructura de diagrama de estrella para la fácil explotación de los datos. Este diagrama en estrella consta de una tabla de hechos que contiene los datos para el análisis, y las tablas de dimensiones, las cuales acompañan a la tabla de hechos y determinan los parámetros de los que depende la tabla de hechos. Por lo que en este esquema en estrella, la tabla *Visitas* actúa como la tabla de hechos, siendo las tablas que la rodean las dimensiones. Entre las tablas de dimensiones, se encuentra la tabla autogenerada calendario, la cual se utiliza para obtener detalles en los filtros de las fechas. Este esquema de diseño en estrella se puede apreciar en la Figura 22.

Figura 22. Diseño lógico en estructura en estrella.



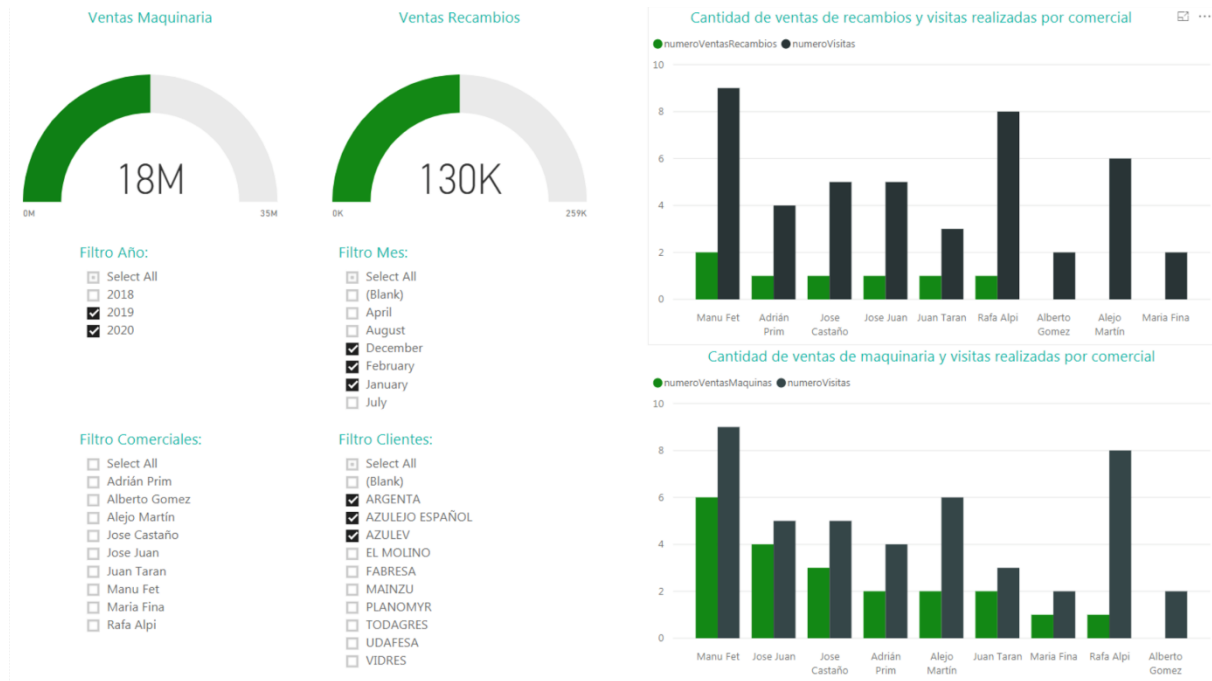
La Figura 22 representada el diseño lógico dispuesto en estructura de estrella donde se puede observar la tabla de visitas como una tabla de hechos. El resto de tablas que la rodean que son las dimensiones de este diagrama en estrella.

En la Figura 22 se puede observar que la tabla de hechos *Visitas* tiene una serie de campos como son: `costeVentaMaquina`, `costeVentaRecambio`, `numVisitas`, `numVentaMaquinaria` y `numVentaRecambio`. Son campos que se han trasladado desde las respectivas tablas de dimensiones para poder contabilizar ciertos datos que se necesitan en la tabla de hechos, como el número de visitas de los comerciales, el número de visitas que han supuesto ventas y el que no, y el coste de las ventas respecto a las visitas de los comerciales.

La siguiente fase de implementación gráfica del *dashboard* fue sencilla, ya que el trabajo más tedioso se había realizado en la fase de diseño. Para implementar el *dashboard* se utilizó la interfaz visual e interactiva que proporciona *Power BI* para facilitar el diseño, la cual permite

seleccionar iconos y llevarlos hasta el panel de visualización de *Power BI* donde serán representados con facilidad y rapidez. El *dashboard* resultado se representa en la Figura 23.

Figura 23. *Dashboard* interactivo.



En la Figura 23 se muestra el *dashboard* realizado con la herramienta *Power BI*, que se utiliza para la mejora del proceso de toma de decisiones ejecutivas. En él se puede encontrar la facturación obtenida con la venta de maquinaria y número de visitas por comercial. También se encuentra la facturación obtenida con la venta de recambios y número de visitas por comercial.

La implementación respeta el diseño del prototipo inicial, excepto el gráfico que representaba el total de ventas que ha sido sustituido por dos gráficos que representan la división de las ventas de la maquinaria y los recambios.

## 5.5. Pruebas

Este apartado contiene las pruebas realizadas sobre el proyecto. Contiene tanto las pruebas de consistencia interna del proyecto, llevadas a cabo por el equipo informático, como las pruebas realizadas con los usuarios finales del sistema.

### **5.5.1. Pruebas consistencia proyecto**

En esta parte ya se ha implementado el proyecto, por lo que es hora de que el departamento Informático pruebe el funcionamiento de todas las partes del sistema. Para ello, se necesita instalar las aplicaciones en los terminales, registrar visitas y ver cómo se comporta el sistema.

En esta parte se han encontrado varios problemas, como por ejemplo, en la aplicación móvil había pantallas que volvían a pantallas que no tenían relevancia para el usuario o también a pantallas que no eran coherentes volver a ser accedidas. También se han encontrado problemas con la fecha en uno de los procesos que conforman el *back-end*. Estos pequeños contratiempos fueron resueltos con rapidez sin tener que alargar el tiempo estimado en esta fase.

### **5.5.2. Pruebas con usuarios finales**

Una vez realizadas las primeras pruebas por parte del departamento de informática y después de corregir los errores encontrados en la fase anterior, se entregó la aplicación móvil a los comerciales para que la probaran durante unos días. También se les enseñó la interfaz del *ERP* corporativo donde pueden encontrar las visitas que han realizado. Por otra parte, al equipo ejecutivo se le presentó su *dashboard* en *Power BI*.

Una vez finalizados estos días de prueba, se llegó a la conclusión de que el funcionamiento del sistema fue el esperado, ya que no surgió ningún error adicional ni obtuvimos ninguna queja por parte de los usuarios.





## Capítulo 6

# Conclusiones

Como conclusiones a este trabajo final de grado se puede extraer que:

1. Se ha desarrollado un sistema que permite registrar visitas realizadas por los comerciales y que estas visitas se almacenen junto a los datos relevantes de la empresa, es decir, en la base de datos del *ERP* corporativo.
2. Los ejecutivos y los propios comerciales de la empresa pueden consultar y controlar las visitas que realizan los clientes de la empresa, todo esto mediante las interfaces del *ERP* corporativo, con cuyo uso ya se encuentran familiarizados.
3. En cuanto al proceso de ayuda a la toma de decisiones ejecutivas, se ha mejorado significativamente, ofreciendo un *dashboard* visual e interactivo, que muestra información a los ejecutivos como por ejemplo la relación del número de visitas realizadas por cada comercial y las ventas que han realizado.

De esta manera, hemos podido desarrollar un sistema para la organización que contenga información acerca de las visitas realizadas por los comerciales, ya que hasta el momento no quedaba constancia acerca de la relación comercial-cliente. También hemos provisto a los ejecutivos de una plataforma que le sirva de ayuda en el proceso de toma de decisiones ejecutivas, motivando a estos para futuras peticiones en esta misma línea de trabajo.

Para concluir, considero que he obtenido grandes beneficios tanto a nivel formativo como a nivel profesional y personal. En el ámbito formativo, he conseguido ampliar mis conocimientos de desarrollador de aplicaciones *Android*, de desarrollador de la parte del *back-end* con *Node JS*, y en el uso de la herramienta *Power BI* para la creación de *dashboards* interactivos y la explotación de información. Durante el grado en Ingeniería informática, de todas las tecnologías utilizadas en el proyecto, únicamente había podido conocer la herramienta *Power BI*. Por tanto, la realización de este proyecto me ha servido para observar y ampliar mi formación referente a nuevas tecnologías existentes en el mercado. También cabe mencionar que la utilización de la plataforma *Firebase* como base de datos de la aplicación móvil me ha servido para explorar nuevas opciones de bases de datos no relacionales con características muy interesantes como las capas de *back-end* de gestión de usuarios o la base de datos en tiempo

real. Esta última ha sido todo un gran descubrimiento, ya que nos permite manejar desde la parte del *back-end* los datos en tiempo real, es decir, con *Firebase* no necesitamos programar ningún cron, ni similares para que ejecute tareas en un cierto instante temporal, sino que se obtienen los cambios mediante unos eventos que actúan como observadores de la base de datos. Cuando en *Firebase Real Time* ocurre algún cambio, estos eventos son capaces de informar de que ha ocurrido y de qué ha cambiado.

En el ámbito profesional, el proyecto se ha desarrollado en una etapa de tres meses durante mi primer contrato con una empresa que cuenta con un departamento informático, que ha tenido una duración de 19 meses. Esto me ha permitido obtener conocimiento tanto sobre el funcionamiento general de la organización, como del funcionamiento de los procesos a seguir en el desarrollo de un proyecto en el departamento Informático. Los demás miembros del equipo contaban con mayores conocimientos sobre las tecnologías, herramientas y demás medios utilizados en el proyecto, que han sabido transmitirme con éxito.

En el ámbito personal, este proyecto me ha servido para crecer como persona. Tras esta experiencia, me siento más capaz y válido para llevar a cabo nuevos proyectos tecnológicos, y esto me motiva para en un futuro seguir formándome. Finalmente, destacar que el supervisor del proyecto Nacho Porcar quedó gratamente satisfecho con el resultado final del sistema, y con mi aportación y mi rendimiento durante la realización, por lo que estoy contento con el resultado.

# Bibliografía

[1] Colaboradores de Wikipedia. (2019, 14 mayo). Entorno de desarrollo integrado para la plataforma Android. Recuperado 9 junio, 2019, de [https://es.wikipedia.org/wiki/Android\\_Studio](https://es.wikipedia.org/wiki/Android_Studio)

[2] Firebase, G. O. O. G. L. E. (s.f.). FirebaseRealtimeDatabase| Recuperado 9 junio, 2019, de <https://firebase.google.com/docs/database?hl=es-419>

[3] VINOD KUMAN GARG y N.K. VENKIAKRISHNAN. (2012). ENTERPRICE RESOURCE PLANNING—Concepts and Practice, 2nd ed. New Delhi: PHI LearningPrivateLimeted.

[4] Colaboradores de Wikipedia. (2019c, 15 mayo). El programa de gestión empresarial Navision es la solución de software de gestión ERP para las pequeñas y medianas empresas. Recuperado 9 junio, 2019, de [https://es.wikipedia.org/wiki/Microsoft\\_Dynamics\\_NAV](https://es.wikipedia.org/wiki/Microsoft_Dynamics_NAV)

[5] Fundación de Node.js, N. O. D. E. J. S. (s.f.). Documentación | Node.js. Recuperado 9 junio, 2019, de <https://nodejs.org/es/docs/>

[6] Colaboradores de Wikipedia. (2019d, 5 junio). editor de código fuente desarrollado por Microsoft. Recuperado 9 junio, 2019, de [https://es.wikipedia.org/wiki/Visual\\_Studio\\_Code](https://es.wikipedia.org/wiki/Visual_Studio_Code)

[7] Documentación de Power BI - Power BI. (s.f.). Recuperado 9 junio, 2019, de <https://docs.microsoft.com/es-es/power-bi/>

[8] CRAIG LARMAN. (2013). UML y patrones. Prentice Hall

[9] npm: firebase-admin. (2019, 23 mayo). Recuperado 9 junio, 2019, de <https://www.npmjs.com/package/firebase-admin>

[10] npm: mysql. (2019, 18 abril). Recuperado 9 junio, 2019, de <https://www.npmjs.com/package/mysql>

[11] Agrega Firebase al proyecto de Android | Firebase. (s.f.). Recuperado 9 junio, 2019, de <https://firebase.google.com/docs/android/setup?hl=es-419>

[12] npm: httpntlm. (2018, 21 mayo). Recuperado 9 junio, 2019, de <https://www.npmjs.com/package/httpntlm>

[13] npm: soap-ntlm. (2015, 2 febrero). Recuperado 9 junio, 2019, de <https://www.npmjs.com/package/soap-ntlm>