



# GIC

GESTIÓN INTEGRAL DE CLIENTES PARA PEQUEÑAS EMPRESAS

JUAN DANIEL CABRERA DÉNIZ

**TUTOR**

ABRAHAM RODRIGUEZ RODRIGUEZ

TRABAJO DE FIN DE TÍTULO  
UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA

## ÍNDICE

<b>1 INTRODUCCIÓN .....</b>	<b>4</b>
<b>1.1 CONTEXTO .....</b>	<b>4</b>
<b>1.2 DESCRIPCIÓN DEL DOMINIO Y OBJETIVOS.....</b>	<b>5</b>
<b>1.3 METODOLOGÍA.....</b>	<b>9</b>
<b>1.4 ESTRUCTURA DE LA MEMORIA .....</b>	<b>10</b>
<b>2 ANÁLISIS.....</b>	<b>11</b>
<b>2.1 ANÁLISIS DE LA COMPETENCIA .....</b>	<b>11</b>
2.1.1 INVOICE SIMPLE .....	11
2.1.2 FACTUSOL.....	13
2.1.3 CONTASIMPLE .....	16
2.1.4 COMPARATIVA DE LA COMPETENCIA CONTRA GIC .....	18
<b>2.2 TABLA DE DUMMIES Y CARACTERÍSTICAS .....</b>	<b>19</b>
<b>2.3 REQUISITOS .....</b>	<b>20</b>
<b>2.4 UML Y DIAGRAMAS DE CASOS DE USO.....</b>	<b>21</b>
<b>2.5 MOCKUPS, DISEÑO DE LA INTERFAZ Y PALETA DE COLORES .....</b>	<b>24</b>
2.5.1 MOCKUPS .....	24
2.5.2 DISEÑO DE LA INTERFAZ.....	30
2.5.3 PALETA DE COLORES .....	39
<b>2.6 MODELO DE NEGOCIO .....</b>	<b>40</b>
<b>2.7 NORMATIVA Y LEGISLACIÓN .....</b>	<b>41</b>
<b>3 DESARROLLO .....</b>	<b>44</b>
<b>3.1 ALCANCE DE LA IMPLEMENTACIÓN.....</b>	<b>44</b>
<b>3.2 DISEÑO ARQUITECTÓNICO.....</b>	<b>46</b>
<b>3.3 MODELO DE LA BASE DE DATOS.....</b>	<b>47</b>
<b>3.4 TECNOLOGÍAS.....</b>	<b>56</b>
3.4.1 IONIC .....	56
3.4.2 APACHE CORDOVA .....	58
3.4.3 ANGULAR.....	58
3.4.4 TYPESCRIPT.....	59
3.4.5 SASS.....	59
3.4.6 FIREBASE .....	60



3.4.7 CHARTJS.....	61
3.4.8 PDFMAKE.....	61
<b>3.5 PRUEBAS Y DESPLIEGUE .....</b>	<b>62</b>
<b>4 CONCLUSIONES.....</b>	<b>65</b>
<b>5 DOCUMENTACIÓN .....</b>	<b>67</b>
<b>6 ANEXOS .....</b>	<b>69</b>



# 1 INTRODUCCIÓN

## 1.1 CONTEXTO

Todo profesional autónomo o empresa que realiza un servicio o vende un producto, está obligado a presentar una factura. La factura que se genera deberá tener un número representativo y único, información acerca del cliente, del proveedor o acreedor, fecha de generación de la factura, información detallada acerca del servicio o producto (cantidad, precio, descuento e impuesto entre otros campos) e información acerca de las retenciones a aplicar en el caso de que fuese necesario. Los campos descritos se pueden definir como los campos mínimos que deben representar una factura.

Dentro de las empresas, la gestión de la facturación siempre ha sido un proceso laborioso. Tengamos en cuenta que antes de la era de la informática y la información, toda información que se necesitaba generar de una factura tenía que ser detallada en papel y se debía escribir manualmente cada factura por separado con toda su información sin la capacidad de poder reutilizar información. Además, no sólo el proceso de generación de la factura era complejo, sino que su almacenaje, búsqueda y acceso llevaba costes asociados de almacenaje, distribución y personal

Estos procesos de creación, almacenaje y distribución de la información, aunque no eran complejos, sí que conllevaba grandes sumas de tiempo y dinero. Además, dichos procesos se complicaban a medida que las empresas eran de mayor tamaño, dado que los volúmenes de información iban creciendo considerablemente

La globalización de las empresas y el rápido crecimiento de estas ha supuesto que las empresas ya no sólo disponen de tiendas físicas para realizar sus negocios, sino que existe la posibilidad de montar negocios virtuales paralelamente a los físicos o incluso despreciando la parte física de la venta tradicional.

En la actualidad, el mercado en el que nos encontramos obliga a autónomos y pequeñas empresas a tener un volumen importante de información que deben gestionar intentando tener los mínimos costes posibles (tanto de personal como económico) y poder realizarlos de forma ágil y simple.

Al comienzo de la era de la información y la informática, comenzaron a aparecer pequeñas soluciones software que permitían generar facturas. Con el paso del tiempo ya permitían gestionar clientes y facturas, y si seguimos avanzando hasta día de hoy, las soluciones software ya permiten una gestión integral de la facturación de cualquier empresa.

Es importante destacar, que con el paso del tiempo la gestión ha pasado de estar localizada en la propia empresa a estar disponible en cualquier parte del mundo y en el momento que se necesario su uso o acceso. Todo gracias a internet y al rápido crecimiento de las soluciones hardware y software.

Todo esto implica que hoy en día las soluciones software que permiten la gestión de la facturación de las empresas, deben ser opciones ágiles, simples y que permitan llevar la gestión de forma cómoda por una o varias personas en dependencia del volumen del trabajo o de la propia empresa.



## 1.2 DESCRIPCIÓN DEL DOMINIO Y OBJETIVOS

El objetivo principal del aplicativo que se desea construir es la generación de facturas. Una factura está formada por una amplia colección de modelos e información que definen una transacción entre dos empresas. Definiremos una factura en GIC la que consta de las siguientes características:

- Información del cliente: Nombre, NIF o CIF, IBAN, dirección, ciudad, provincia, código postal, teléfono y correo electrónico.
- Información del proveedor: Nombre, NIF o CIF, IBAN, dirección, ciudad, provincia, código postal, teléfono y correo electrónico.
- Fechas:
  - Fecha de presentación: Fecha en la que ha sido presentada la factura.
  - Fecha prevista para el pago: Fecha en la que se prevé cobrar la factura basándose en los términos de pago establecidos y la fecha de presentación.
  - Fecha de creación: Fecha en la que se ha generado la factura en la plataforma y que no es visible por parte del usuario. Esta fecha es generada al guardar una factura en el sistema.
- Número de la factura: Identificación que establece el generador de la factura para poder identificar la factura.
- Términos de pago: Definen el método de pago (contado, transferencia, pagaré o talón entre otros) y los días previstos para el pago. Es habitual que los términos de pago vengan definidos por el cliente.
- Líneas de facturas: Las líneas de facturas definen el cuerpo de la factura y nos dan información de los artículos o servicios que se han generado en dicha factura. Los datos que una línea de factura debe tener son: Nombre, cantidad, unidad, precio, descuento e impuesto.
- Retención: Cantidad que se retiene por parte del estado correspondiente de la factura.
- Archivos adjuntos: Es habitual que una factura tenga documentos e imágenes adjuntas para extender la información existente en las líneas de factura. Además, se permitirá la subida de archivos o la posibilidad de sacar una foto a algún documento, trabajo o servicio en el momento que se esté generando o modificando la factura.
- Observaciones: La factura debe tener un campo de observaciones que el usuario podrá usar con diferentes fines en el caso de lo que necesite.

Con todos los puntos anteriores, podemos definir que tenemos un modelo de factura que puede cumplir los requisitos que la mayoría de los autónomos y pequeñas empresas que hagan uso del aplicativo van a necesitar. Es evidente, que a la hora de generar una factura tanto como en las líneas de factura como en la factura en general, se deberá permitir la creación de nuevos campos que extiendan el modelo de factura que estamos considerando. De forma que, aunque en principio el usuario no tenga los campos que necesita, los pueda añadir posteriormente. Cabe destacar que estos nuevos campos que se van a generar no sólo se deben permitir en caliente a la hora de generar una nueva factura, sino que se podrán parametrizar en la configuración de la factura para poder ser utilizados en el resto de las facturas.



En la factura, uno de los primeros datos a rellenar, será el cliente y es ahí cuando aparece el concepto de compañía. En la plataforma se podrán generar compañías para poder ser reutilizadas tantas veces como fuese necesario en presupuestos, facturas o pagos. Además de los atributos que serán visibles en una factura, se deberá poder almacenar el logotipo y otros datos de interés como puede ser el tipo de compañía que estamos creando. En la plataforma van a existir dos tipos importantes de compañía: los clientes y los proveedores. Entenderemos como clientes aquellos a los que vamos a presentar tanto presupuestos como facturas y como proveedores a aquellos que nos generan pagos dentro del aplicativo. Por tanto, es vital que a la hora de generar una compañía se defina alguno de los dos tipos establecidos o incluso ambos al mismo tiempo, para poder hacer uso de los datos de la compañía donde fuesen necesarios.

Otro de los campos vitales dentro de la factura, serán los productos o servicios. Estos suelen ser diferentes en cada línea de factura si estamos hablando de trabajos puntuales o sobre todo si hablamos de servicios específicos como poder “Lacado de un mueble”. Sin embargo, en la mayoría de las ocasiones, los usuarios que hagan uso de GIC harán uso de un catálogo de productos y servicios. El catálogo de productos y servicios facilitará al usuario la generación de las líneas de factura, dado que al generar un producto o servicio ya definirá tanto su precio como su impuesto como incluso su descuento. De forma que al generar la factura sólo tendrá que seleccionar que desea e indicar la cantidad para que se calcule automáticamente dicha línea de factura. Es vital que este catálogo sea simple y cómodo de usar, dado que esto hará que los usuarios inviertan el tiempo necesario en introducir los datos de sus productos y servicios dado que a posteriori les reducirá los tiempos de generación de presupuestos, facturas y pagos.

Otro punto significativo sobre los productos y servicios será la gestión de stock. El aplicativo debe ser capaz de informar al usuario en todo momento de si es posible añadir un producto en base a su existencia en el almacén o si es posible realizar un servicio teniendo en cuenta de que se tengan todas las herramientas disponibles para ello.

Una vez definidas las compañías y los productos o servicios. Llegado a este punto otro punto vital de la aplicación, será la generación de presupuestos. Esto se debe a que la mayoría de los autónomos y pequeñas empresas, no generan una factura directamente a sus clientes, sino que generan un presupuesto y en el caso de que este se acepte, se comienza el trabajo o se atiende el pedido y tras ello se generará la factura correspondiente. Por tanto, cuando se genere un presupuesto se debe dotar a GIC con la capacidad de convertir un presupuesto en una factura.

El aplicativo no sólo permitirá la generación a partir de un presupuesto, sino que también se permitirá generar una factura a partir de otra generada con anterioridad. Esto es importante, dado que en muchas ocasiones un autónomo o pequeña empresa presenta mensualmente la misma factura. Como es lógico, todos los campos de la factura serán los mismo salvo la fecha de presentación, el número identificativo de la factura y los documentos adjuntos de la factura anterior.

Una vez se genere un presupuesto o factura, es importante indicar cómo los usuarios de la plataforma podrán hacer llegar esa información a sus clientes. Para ello, GIC presenta varias vías de presentación de los presupuestos o facturas:



- Impresión en formato PDF: Una de las opciones más comunes en este tipo de situaciones es la generación de un PDF ya sea para ser impreso o para ser enviado vía telemática al cliente. GIC no sólo permitirá la generación de un PDF del presupuesto o factura, sino que además existirá una galería de plantillas que el usuario podrá escoger en la configuración de la factura. Además, a dichas plantillas se les podrá seleccionar la paleta de colores. Esto permitirá a los usuarios de GIC personalizar sus facturas de forma que se adapten lo máximo posible a la imagen corporativa de la compañía que las genera.
- Generación de una factura electrónica: Dependiendo del país desde donde se genere una factura, existe una factura electrónica que define cómo debe ser una factura y que información debe llevar dicho archivo. GIC permitirá la generación de facturas electrónicas de los países en los que llegue a operar, como puede ser España. En 2019 está previsto que incluso se implante la factura electrónica europea, por lo que una factura electrónica generada por GIC siguiendo su reglamentación será válida en toda Europa.
- Enlace público: Los usuarios de GIC, podrán generar enlaces públicos tanto de presupuestos como de sus facturas. Por tanto, se generará una URL que podrá ser visitada por el cliente para visualizar el presupuesto o factura que ha sido generado en su nombre. Además, se permitirá descargar tanto en PDF como en formato electrónico la factura desde el enlace público, de forma que el cliente tendrá acceso a la factura en todos sus formatos. Es importante destacar que los enlaces públicos deben tener un tiempo finito de funcionamiento, por lo que un presupuesto o factura de la que se genere un enlace público sólo será visible durante 15 días.
- Portal del cliente: Los usuarios de GIC podrán invitar a sus clientes a formar parte de GIC. En principio no tendrán la herramienta completa, sino un pequeño portal en el que puedan visualizar y descargar en diferentes formatos los presupuestos y facturas que se hayan generado a su nombre. De forma que un cliente podrá tener siempre sus presupuestos y facturas a mano y ser informado en tiempo real de que se le ha generado un presupuesto o factura mediante notificaciones. Esto incluso permite que se pueden generar avisos en el calendario del cliente de las fechas previstas para el pago de sus facturas, de la caducidad de un presupuesto o de las fechas en las que se le presentó una factura.

Un punto muy importante dentro del portal del cliente será la posibilidad que se le ofrecerá a los clientes de pagar una factura directamente desde la plataforma a través de una pasarela de pago. De forma que se simplifica enormemente el proceso de pago por parte de cliente, porque no tiene que tener información adicional del número de cuenta en la que tiene que realizar el ingreso y los detalles que lleva dicho ingreso. Además, esto implica que, si se realiza el pago de una factura, esta cambiará su estado automáticamente a pagada, con lo que los usuarios de GIC no tendrán que realizar el cambio manualmente y se simplifica más aún el proceso.

- Compartir un PDF o factura electrónica: Se podrá compartir una factura mediante las principales vías de comunicación: Correo electrónico, WhatsApp, Telegram, etc. Esto permitirá a los usuarios de GIC compartir directamente su



factura en formato PDF o electrónico con las plataformas que tenga instalada en su dispositivo.

Otro punto importante de GIC será la gestión de los pagos que los autónomos y pequeñas empresas que hagan uso del aplicativo deben realizar. Por tanto, se podrá registrar los datos (proveedor, nombre, identificación de la factura, detalle, total y fecha en la que se realizó el pago) de los pagos dentro de GIC. Cabe mencionar que se podrán categorizar los pagos haciendo uso de las categorías básicas y también de las categorías personalizadas. Esto permitirá que posteriormente se pueda dar una información detallada de los gastos que se están realizando en cada una de las categorías y dar al usuario una vista rápida de en cuál de las categorías está realizando una mayor inversión.

Como es visible, GIC ofrece la gestión de compañías, productos o servicios, presupuestos, facturas y pagos. Esto hace que el volumen de información que los usuarios que hagan uso del aplicativo guarden sea importante. Es por ello, que se generarán filtros y búsquedas que permitan encontrar los datos que sean necesarios con pocos pasos. A continuación, se detalla algunos de los filtros necesarios en dependencia de su modelo:

- Presupuestos y facturas: Se podrán buscar presupuestos y facturas tanto por cliente, fechas, número de factura o estado de la factura. Además, se ofrecerán diferentes filtros que ofrezcan un resultado más exhaustivo, como por ejemplo filtrar las facturas por rangos de totales.
- Compañías: Se podrán realizar búsquedas de compañías por nombre, NIF o CIF, correo electrónico o teléfono. Además, se podrán realizar filtros avanzados que reduzcan los resultados, como por ejemplo compañías que operan en una o varias provincias.
- Productos o servicios: Se podrán realizar búsquedas de productos o servicios por su nombre o descripción. Al igual que con los modelos anteriores, se podrá realizar filtros avanzados que den un mejor resultado de búsqueda.
- Pagos: Se podrán realizar búsquedas de pagos por cliente, identificación de la factura o fechas de pago. Los filtros avanzados también permitirán obtener mejores resultados de los pagos ofrecidos.

Cabe destacar que, de las búsquedas ofrecidas, se podrán generar informes en PDF que podrán ser descargados o compartidos. De esta forma, un usuario podrá generar fácilmente un informe de todas las facturas que le ha realizado a un cliente a lo largo de 2019. Esta opción es de gran utilidad para los usuarios que hagan uso de GIC, por ejemplo, un autónomo tiene muy fácil generar sus trimestres o una pequeña empresa puede generar el modelo 347 de forma cómoda y sencilla.

Además de los informes, en la vista principal existirán gráficas que rápidamente darán información visual a los usuarios de los estados de las facturas del mes, del total de pagos frente a facturas pagadas o de la cantidad de productos o servicios de los que la compañía dispone actualmente. Estas gráficas podrán ser configuradas para que muestre lo que cada usuario desea visualizar y de esta forma tener la información que necesita de una forma simple y atractiva.

Como conclusión, podemos definir que GIC es un aplicativo que da solución a todas aquellas necesidades que los autónomos y pequeñas empresas necesitan. Y lo hace siempre buscando la mejor solución para dichos usuarios, de forma que todos los





procesos de insertar compañías, productos o servicios, aunque son laboriosos se hacen de forma muy simple para que a posteriori se puedan generar presupuestos y facturas con muy pocos pasos. Además, GIC no sólo simplifica el proceso de generación, sino que reduce los procesos posteriores a la generación de la factura.

### 1.3 METODOLOGÍA

SCRUM es un proceso de desarrollo ágil y de buenas prácticas que es adoptado por un equipo de trabajo para el desarrollo colaborativo de un proyecto. La metodología SCRUM se basa en un conjunto de entregas parciales y regulares de proyecto con la meta final de llegar al punto deseado del proyecto.

Esta metodología de desarrollo está indicada para proyectos complejos, donde los requisitos son cambiantes o poco definidos, donde la innovación, competitividad, flexibilidad y productividad son fundamentales. También es ideal para proyectos en los que el cliente desea estar al tanto del proyecto, cuando los costes de tiempo y dinero del proyecto se están disparando o cuando se requiere identificar y solucionar problemas respecto al equipo de trabajo.

Para poder realizar la metodología SCRUM es imprescindible la colaboración y el contacto constante con el cliente, dado que es este el que prioriza los objetivos y la meta final que se desea obtener. Por tanto, al comienzo de cada iteración existe una reunión tanto con el cliente como con el equipo de trabajo para definir las historias (requisitos) que se deben llevar a cabo para completar la iteración.

Cada iteración, por normal general, será de dos o cuatro semanas y comenzará con una reunión del equipo donde una vez definidas las historias, se procederá a la selección de estas por cada miembro del equipo, de forma que la selección de las historias dependerá tanto de la prioridad de las mismas como del tiempo estimado para la realización de cada una de ellas, de forma que, la suma total de las horas seleccionadas por un miembro del equipo deben ser de las que dispone para trabajar.

Al final de cada iteración, se debe comprobar que las historias se han completado y que el tiempo empleado en ellas ha sido el que sea había establecido previamente. Además, es en este momento cuando se analiza si han existido problemas, si existen nuevos requisitos o si es necesaria cualquier modificación del proyecto.

Como se puede ver, SCRUM es ideal como metodología para proyectos que se desarrollan en equipo, dado que facilita la comunicación dentro del mismo y permite que gracias a su modelo iterativo, el proyecto que se está desarrollando llegue a buen puerto.

Las ventajas de SCRUM son muchas y diversas, es por ello, por lo que se ha decidido seguir una metodología SCRUM, pero adaptando la metodología para que sea desarrollada por una sola persona y no por un equipo. Por tanto, todas las iteraciones van a ser desarrolladas por una persona y las historias serán adaptadas a la jornada laboral de una persona, por lo que cada iteración semanal será de unas 40 horas.



## 1.4 ESTRUCTURA DE LA MEMORIA

La memoria que se muestra a continuación se ha estructurado de la siguiente forma:

- En la primera parte se hablará del análisis previo de la situación de la competencia; los requisitos previos, funcionales o no funcionales; diagramas de caso de uso y UML; los conceptos de diseño; modelos de negocio; normativa y legislación.
- En la segunda parte de la memoria se hablará del desarrollo propio del aplicativo. Por tanto, se definirá el alcance de la implementación, diseño arquitectónico, el modelo de la base de datos, las tecnologías que serán utilizadas y las pruebas a realizar para comprobar el correcto funcionamiento del aplicativo.
- En la tercera parte de la memoria, se realizarán todas las conclusiones tanto a nivel de objetivos conseguidos, como las personales acerca del proceso de desarrollo del aplicativo.



## 2 ANÁLISIS

### 2.1 ANÁLISIS DE LA COMPETENCIA

Para el estudio de la competencia, nos hemos centrado en varios aspectos de los aplicativos de la competencia. Es importante a la hora de realizar el estudio, no centrarse solamente en los aspectos negativos de la competencia, sino visualizar tanto los aspectos positivos como negativos.

#### 2.1.1 INVOICE SIMPLE



## Invoice Simple

Invoice Simple destaca por ser un aplicativo multiplataforma. Por tanto, podemos hacer uso del aplicativo por tres vías: Web, Android e iOS.

Desde la versión web de Invoice, no se muestran detalles de la cantidad de usuarios de los que dispone y su valoración del aplicativo. Cosa que si es visible desde su página de Android e iOS:

- Android: Actualmente tiene entre 500.000 y 1.000.000 de instalaciones. Además, tiene una valoración de 4,70 sobre 5,00 por parte de los usuarios respecto a las 18.441 valoraciones con las que cuenta.
- iOS: Aunque en la página oficial no informa del número de instalaciones que se han realizado. Si dispone de la valoración por parte de los usuarios que es de 5 estrellas por parte de los 2099 usuarios que han realizado la valoración.

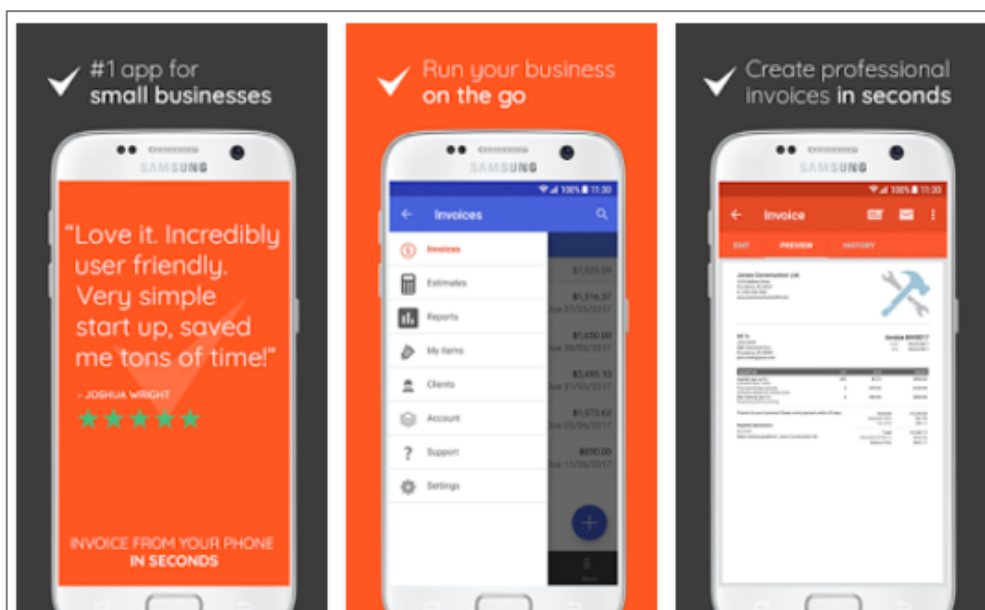


FIGURA 2.1.1.A – VISTA DEL APPLICATIVO INVOICE EN ANDROID



El uso del aplicativo es bastante intuitivo y cómodo de utilizar. Se pueden crear clientes registrándolos desde la aplicación o importarlos desde la agenda de contactos. En este punto, el registro de un cliente es bastante simple, aunque los campos que permite rellenar son bastantes escasos, dado que no permite detalles simples como añadir una foto al contacto, incluir un documento de identificación entre otros. Sin embargo, el hecho que permita importar usuarios desde la agenda de contactos es un gran punto a su favor, dado que incluso permite importarlos a la hora de realizar una factura o presupuesto.

Además, el aplicativo permite crear presupuestos y facturas. A la hora de crearlas es bastante sencillo la forma de realizarlo, sin embargo, no es cómodo crearlas. Uno de los principales inconvenientes que se puede visualizar es la numeración de facturas o presupuestos, dado que se permite la modificación de estas y puede llevar a errores con usuarios poco experimentados con este tipo de software. Otro de los grandes errores puede llegar a ser que no permita generar una factura a partir de un presupuesto.

En cuanto a diseño del aplicativo, podemos decir que sigue las líneas marcadas por cada uno de los sistemas operativos. Sin embargo, no profundiza en el diseño por lo que acaba siendo demasiado simple y no llega a diferenciarse con cualquier aplicación que se genere para Android o iOS.

La base de datos está localizada en el dispositivo, permitiendo realizar copias de seguridad tanto en la tarjeta SD o en servicios de almacenamiento en la nube como pueden ser Dropbox o Google Drive. Este hecho se traduce que la aplicación no se puede utilizar desde varios dispositivos o usuarios con una base de datos en la nube.

La aplicación es gratuita y permite crear un par de facturas para comprobar el funcionamiento del aplicativo. Una vez creada dos o tres facturas, se puede tener la versión completa realizando un pago de 2,99€ mensuales o 29,99€ anuales. Como es visible, es una aplicación económica y bastante completa en cuanto a funciones para el precio mensual que tiene.

En grandes líneas Invoice es un aplicativo simple y funciona correctamente para usuarios que necesiten crear dos o tres facturas mensuales en las que existan pocos servicios o productos. Sin embargo, es muy poco funcional para empresas o autónomos que tenga un cierto volumen de facturación o incluso para los que aun teniendo poco volumen si que tienen una gran cantidad de productos.



### 2.1.2 FACTUSOL



Factusol destaca por ser un programa de escritorio bastante potente que permite tener los datos sincronizados o no, en dependencia de lo que el usuario decida que puede ser lo mejor para su compañía.

Desde su web no detallan en ningún momento la cantidad de usuarios ni compañías que hacen uso de su aplicativo. Dado que no dispone de ninguna versión en sistemas operativos móviles, tampoco es posible ver la valoración que los usuarios hacen en las principales tiendas móviles.

Uno de los puntos fuertes de Factusol se encuentra en su web y toda la información que muestra acerca de su producto. Esto es importante, dado que los usuarios que están decidiendo por qué gestor de sus facturas escoger, tienen en la web de Factusol toda la información sobre su producto y mostrado de forma que los posibles aspectos negativos del producto estén difuminados.

Factusol es un programa bastante complejo y con grandes opciones para realizar acciones de todo tipo. Esto significa que nos encontramos con un programa que da opciones para casi cualquier acción, lo que para medianas o grandes empresas es un punto a su favor. Sin embargo, para autónomos y pequeñas empresas, significa tener un aplicativo demasiado grande y con muchas opciones que jamás van a ser utilizadas.

Aparte de ser un aplicativo robusto y de gran tamaño, Factusol pertenece a un grupo de aplicaciones de la compañía Software del Sol. En la que se encuentran otros aplicativos como puede ser TPVSOL, GESTORSOL, CONTASOL, ESTIMASOL o NOMINASOL. Esto quiere decir que el grupo de aplicativos da respuesta a casi todas las necesidades que la gestión de una compañía requiere. Sin embargo, esto implica que para que se consiga llevar toda la gestión de una empresa se requiera varios aplicativos, con lo que aumentamos la complejidad de la gestión y el correspondiente aumento económico.

Factusol tiene dos modalidades, trabajo en local o en la nube. Evidentemente la versión del aplicativo limita mucho las opciones del usuario dado que sólo podrá acceder a sus datos y a su gestión dentro de la compañía, ya sea en un equipo o en varios siempre que la conexión sea local. La versión en la nube aumenta las posibilidades y permite gestionar los datos desde varios equipos y sincronizar los datos entre ellos. Es evidente que existen diferencias económicas entre las dos opciones:



Modalidades de contrato			
DIGITAL	ESTÁNDAR	PROFESIONAL	ÉLITE
<ul style="list-style-type: none"> <li>✓ Actualizaciones anuales</li> <li>✗ Soporte por email</li> <li>✗ Soporte por teléfono</li> <li>✗ Asistencia para redes locales</li> <li>✗ Acceso al Centro de Soporte Online</li> </ul>	<p style="background-color: #007bff; color: white; padding: 2px; text-align: center; font-weight: bold;">Recomendado para soporte por email</p> <ul style="list-style-type: none"> <li>✓ Actualizaciones anuales</li> <li>✓ Soporte por email</li> <li>✗ Soporte por teléfono</li> <li>✗ Asistencia para redes locales</li> <li>✓ Acceso al Centro de Soporte Online</li> </ul>	<p style="background-color: #fd7e14; color: white; padding: 2px; text-align: center; font-weight: bold;">Recomendado para soporte por teléfono</p> <ul style="list-style-type: none"> <li>✓ Actualizaciones anuales</li> <li>✓ Soporte por email</li> <li>✓ Soporte por teléfono</li> <li>✗ Asistencia para redes locales</li> <li>✓ Acceso al Centro de Soporte Online</li> </ul>	<ul style="list-style-type: none"> <li>✓ Actualizaciones anuales</li> <li>✓ Soporte por email</li> <li>✓ Soporte por teléfono</li> <li>✓ Asistencia para redes locales</li> <li>✓ Acceso al Centro de Soporte Online</li> </ul>
<b>150,00 € / Año</b>	<b>195,00 € / Año</b>	<b>310,00 € / Año</b>	<b>420,00 € / Año</b>
CONTRATAR	CONTRATAR	CONTRATAR	CONTRATAR

FIGURA 2.1.2.A – MODALIDAD DE CONTRATO DE FACTUSOL LOCAL

Modalidades de contrato			
SIN SOPORTE		CON SOPORTE	
MENSUAL	ANUAL	MENSUAL	ANUAL
	1 Usuario		1 Usuario
	2 Empresas		2 Empresas
<b>210,00 € / Año</b>		<b>420,00 € / Año</b>	
CONTRATAR		CONTRATAR	

FIGURA 2.1.2.B – MODALIDAD DE CONTRATO DE FACTUSOL EN LA NUBE

Como se pueden observar en las dos figuras anteriores, el modo en local (Figura 2.1.2.A) tiene varias modalidades de pago y estas se diferencian sobre todo en su soporte. Sin embargo, el modo en la nube (Figura 2.1.2.B) sólo tiene dos modalidades y ambas son bastante ya más caras que las dos primeras opciones en modo local. Es lógico, por otro lado, que el modo en la nube sea una opción más cara dado que es más completa y permite muchas más posibilidades.



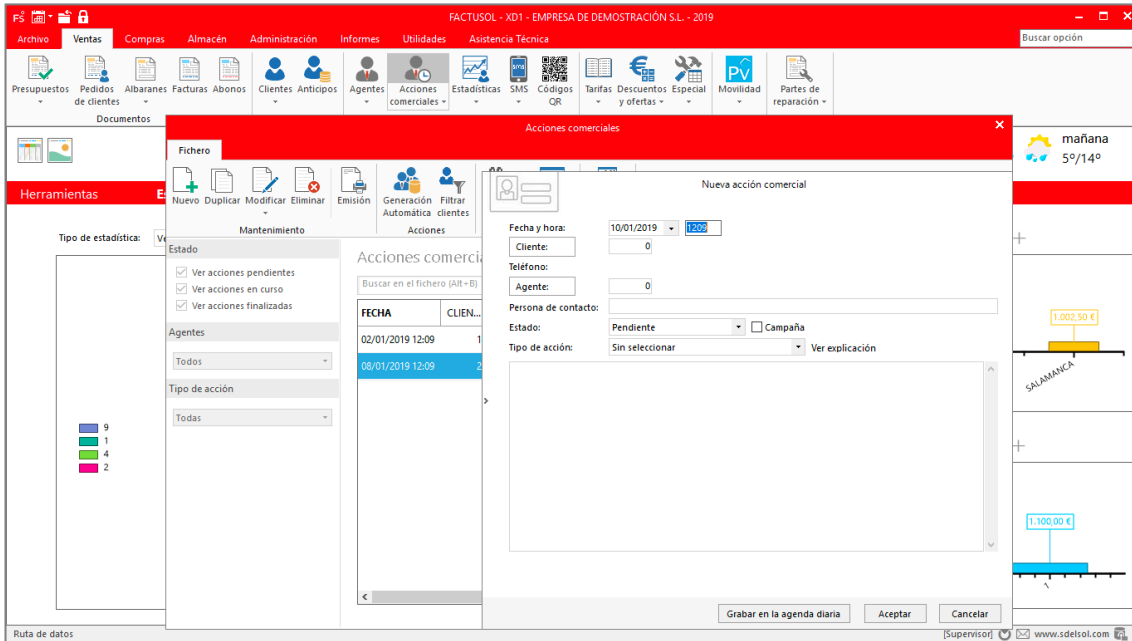


FIGURA 2.1.2.C – MODALIDAD DE CONTRATO DE FACTUSOL EN LA NUBE

La interfaz del aplicativo es similar a la que ofrece el paquete Microsoft Office, lo cual puede ser una ventaja dado que muchos usuarios que realicen la gestión tendrán que hacer uso de aplicativos como el Microsoft Word o Microsoft Excel que vienen incluidos dentro del paquete Office. Sin embargo, también es una desventaja porque hace que el aplicativo no tenga una interfaz propia y le resta personalidad al producto.

Factusol es un gran aplicativo y con muchas opciones que dan respuesta a que si todas las acciones que una empresa necesita para realizar su gestión. Sin embargo, lo que puede ser su gran punto a favor, también es su gran desventaja. Factusol tiene demasiadas opciones y para dominarlo se necesitan grandes dosis de tiempo y de lecturas de manuales y vídeos, o incluso accediendo a la propia formación que ellos ofrecen. Todo ello implica que el producto está destinado a grandes medianas y grandes empresas con altos volúmenes de información y de flujos de datos, pero no responde a la necesidad de autónomos y pequeñas empresas que necesitan disponer de su información siempre encima y que les de respuesta completa a las necesidades básicas sin ser demasiado complejo el aplicativo que estén utilizando.

## 2.1.3 CONTASIMPLE


 Contasimple

Contasimple es un aplicativo multiplataforma y es accesible desde tres vías distintas: Web, Android e iOS. Contasimple muestra desde su web que opera en nueve países y que dispone de más de ciento setenta mil usuarios. También en la web nos informa que ya dispone de más de diez millones de descargas entre todas sus plataformas.



FIGURA 2.1.3.A – VISTA DE CONTASIMPLE EN TODAS SUS VERSIONES

Desde el principio Contasimple informa que va dirigido hacia autónomos y pequeñas empresas, siendo de los pocos aplicativos que van dirigidos hacia este mercado y busca que su aplicativo cumpla todos los requisitos que dicho mercado tiene.

Una de las grandes ventajas de las que goza Contasimple es que sus facturas se encuentran en la nube en lo que ellos denominan “Disco duro virtual”, lo que permite que el usuario tenga acceso a sus facturas independientemente del dispositivo o plataforma mediante la cual esté accediendo.

Contasimple dispone de varios planes de pago (Figura 2.1.3.B) de los cuales el usuario podrá seleccionar el plan que se adapte a sus necesidades en cada momento. Cabe destacar la existencia de un plan gratuito que mientras no supere las limitaciones existentes podrá utilizar de por vida sin realizar pago alguno. Además, el mayor de sus planes



supone unos 108 euros anuales sin IVA, lo que supone que su mayor plan de pago no supondrá un gran coste para el usuario que esté haciendo uso del aplicativo.

	EL MÁS POPULAR	
<p style="text-align: center;"><b>Básico</b></p> <p style="text-align: center;"><b>B</b></p> <p style="text-align: center;"><b>0</b> €/mes</p> <p style="text-align: center;"><i>Para siempre</i></p>	<p style="text-align: center;"><b>Profesional</b></p> <p style="text-align: center;"><b>P</b></p> <p style="text-align: center;"><b>6</b> €/mes</p> <p style="text-align: center;"><i>(9 € facturado trimestralmente) IVA no incluido</i></p>	<p style="text-align: center;"><b>Ultimate</b></p> <p style="text-align: center;"><b>U</b></p> <p style="text-align: center;"><b>9</b> €/mes</p> <p style="text-align: center;"><i>(12€ facturado trimestralmente) IVA no incluido</i></p>
<p>50 Registros/año</p> <p>5 Productos</p> <p>10 Clientes / 10 Prov.</p> <p>10MB Disco virtual</p> <p>1 Remesa bancaria</p> <hr/> <p>✓ Exportaciones (Excel o PDF)</p> <p>✗ Personalización de facturas</p> <p>✗ Impuestos</p> <p>✗ Informes</p> <p>✗ Factura electrónica</p> <p style="text-align: center; border: 1px solid #ccc; padding: 5px;">Más información</p>	<p>500 Registros/año</p> <p>50 Productos</p> <p>Clientes / Prov. ilimitados</p> <p>100MB Disco virtual</p> <p>12 Remesas bancarias</p> <hr/> <p>✓ Exportaciones (Excel o PDF)</p> <p>✓ Personalización de facturas</p> <p>✓ Impuestos</p> <p>✓ Informes básicos</p> <p>✗ Factura electrónica</p> <p style="text-align: center; border: 1px solid #ccc; padding: 5px;">Más información</p>	<p>Registros ilimitados</p> <p>Productos ilimitados</p> <p>Clientes / Prov. ilimitados</p> <p>1GB Disco virtual</p> <p>Remesas bancarias ilim.</p> <hr/> <p>✓ Exportaciones (Excel o PDF)</p> <p>✓ Personalización de facturas</p> <p>✓ Impuestos</p> <p>✓ Informes avanzados</p> <p>✓ Factura electrónica</p> <p style="text-align: center; border: 1px solid #ccc; padding: 5px;">Más información</p>

FIGURA 2.1.3.B – PLANES DE CONTASIMPLE

Contasimple dispone de una interfaz propia y personalizada, que da a los usuarios la capacidad de reconocer el aplicativo sea cual sea la plataforma desde la que se esté utilizando. Sin embargo, no se trata de un diseño que podamos considerar limpio y desde cualquiera de las plataformas desde la que hagamos uso de Contasimple, podemos detectar que abusa de meter demasiada información en poco espacio. Esto se traduce en clics erróneos, sobre todo en dispositivos móviles de poca pantalla.

Contasimple permite realizar todas las acciones que un autónomo o pequeña empresa necesita realizar de forma simple y con una poca necesidad de aprendizaje. La aplicación es intuitiva y no tienen una cantidad infinita de opciones que hagan al usuario perderse entre opciones y características que en la mayoría de las ocasiones no son necesarias.

En líneas generales Contasimple es un aplicativo que cumple lo que promete y se encuentra en constante actualización y contacto con sus usuarios, esto es visible si visitamos su página en la Google Play. Además, está focalizada para autónomos y pymes, lo que la hace centrarse en su mercado y ofrecer resolver las necesidades de este. Sin embargo, la falta de un diseño mejor analizado y ejecutado hace que el aplicativo no tenga más calidad y otros productos del mercado por su diseño atractivo ganen más usuarios desde un inicio a la hora de que el usuario seleccione que producto desea.



### 2.1.4 COMPARATIVA DE LA COMPETENCIA CONTRA GIC

APLICATIVO	CREADA PARA AUTÓNOMOS Y PEQUEÑAS EMPRESAS	DISEÑO PERSONALIZADO	OPCIONES Y CARACTERÍSTICAS NECESARIAS	WEB O SISTEMA DE ESCRITORIO	APLICACIÓN MÓVIL	INFORMACIÓN ACERCA DE USO ACCESIBLE Y SIMPLE	PLANES DE PAGO DIFERENCIADOS	PLANES COMPLETOS DIFERENCIADOS POR FACTURAS Y NO POR CARACTERÍSTICAS
INVOICE	SI	NO	SI	SI	SI	NO	SI	NO
FACTUSOL	NO	NO	NO	SI	NO	NO	SI	SI
CONTASIMPLE	SI	SI	SI	SI	SI	NO	SI	NO
GIC	SI	SI	SI	SI	SI	SI	SI	SI

FIGURA 2.1.4.A – TABLA COMPARATIVA LA COMPETENCIA CONTRA GIC

Una vez analizada la competencia, podemos distinguir en qué aspectos se va a distinguir GIC frente a otras opciones del mercado. Esas opciones las podemos definir en tres bloques, aunque en realidad existirán otras características que lo aumentarán el interés de los usuarios de elegir a GIC frente al resto de opciones.

- GIC se encuentra enfocada a los autónomos y pequeñas empresas. La mayoría de las opciones se encuentran enfocadas para compañías de mayor tamaño y no tienen en cuenta las necesidades que necesitan el mercado hacia el que se quiere distinguir GIC.
- Diseño personalizado. Una de las grandes ventajas de las que gozará GIC es su diseño simple y personalizado. Esto hará que los usuarios se sientan cómodos con el uso del aplicativo y lo distingan de otras opciones porque realizar la gestión con GIC es muy cómodo y atractivo.
- Opciones justas y necesarias. Dado que la mayoría de los aplicativos quieren optar al mayor mercado posible, llenan sus aplicativos de opciones y características que la mayoría de los usuarios no usan o limitan su uso a una o dos ocasiones. GIC estará focalizada a realizar las acciones que los usuarios demanden y a diseñar el aplicativo para que con pocos pasos los usuarios consigan realizar las acciones que necesitan.

Además, una de las ventajas que debe tener GIC teniendo en cuenta el mercado al que nos disponemos a atender, será que deberá ofrecer planes y modelos asequibles para los usuarios. No sólo se trata de ofrecer varios modelos con precios distintos para llegar al mercado con precios que los usuarios puedan seleccionar, sino que dichos modelos deben ofrecer y ver que tipos de usuarios tienen y ofrecer lo que cada compañía necesita en ese momento. También es importante destacar que GIC se diferenciará porque sus modelos permitirán hacer todas las acciones en el aplicativo y que sólo diferenciará sus planes en cuanto a usuarios y cantidad de facturas mensuales y no limitará a los



usuarios a planes en los que faltan características para obligar a seleccionar planes de mayor coste por características básicas que los planes de menor coste nunca incluirán.

## 2.2 TABLA DE DUMMIES Y CARACTERÍSTICAS

NOMBRE	EDAD	AUTONO O EMPRESA	PRODUCTO O SERVICIO	ALMACÉN	IMPUESTO	EMPLEADOS	VOLUMEN DE FACTURAS	PRESUPUESTOS	DISPOSITIVO
CARLOS	36	AUTONOMO	SERVICIO	NO	IGIC IRPF	NO	8 MES	NO	MÓVIL
ANABEL	47	EMPRESA	PRODUCTO	SI	IGIC IRPF	SI	321 MES	SI	WEB
SEBASTIÁN	73	EMPRESA	PRODUCTO	SI	IVA IGIC IRPF	SI	674 MES	SI	WEB Y MÓVIL
GUIOMAR	27	AUTONOMO	PRODUCTO	SI	IGIC IRPF	SI	36 MES	NO	WEB
NEMROD	41	EMPRESA	SERVICIO	NO	IGIC IRPF	NO	18 ANUAL	NO	WEB Y MÓVIL
SORAYA	68	EMPRESA	PRODUCTOS Y SERVICIOS	SI	IVA IGIC IRPF	SI	1247 MES	SI	WEB Y MÓVIL

FIGURA 2.2.A – TABLA DE DUMMIES

La tabla de dummies nos proporciona la información necesaria de los posibles usuarios que van a hacer uso de nuestra solución software. Cada columna de las que se muestran da información de posibles requisitos tanto funcionales como no funcionales que nos van a ser necesarios incluir en la solución para poder ofrecer una alternativa competitiva.

Se nos hace necesario saber la edad del usuario, más que nada porque la edad del usuario implica que se trata de una generación acostumbrada al uso de las tecnologías de la información, como puede ser una franja de edad de los 20 a los 40 años, mientras que una franja de edad que supere los 60 años, por norma general no tendrá grandes conocimientos informáticos.

Otro dato que debemos conocer de nuestros usuarios es que si se trata de autónomos o empresas. Este dato nos permitirá diseñar la aplicación buscando hacerla más cómoda y simple para el mercado en el que tengamos más posibilidades.

Es importante visualizar si se trata de productos o servicios lo que vamos a facturar, dado que dependiendo de cuál sea, deberemos tener ciertos campos para poder almacenar correctamente la información del producto o servicio, para que al añadirlo a la factura tenga toda la información que sea necesaria sobre el mismo.

Además, va a ser muy importante conocer el tipo de impuesto a aplicar. En nuestro caso, el territorio español, vamos a tener como mínimo dos tipos de impuestos a aplicar en dependencia de en qué región nos encontremos: IGIC o IVA. Por lo que será necesario crear un modelo para impuesto y que el usuario pueda establecer su impuesto correctamente.

Una parte importante de la aplicación será la gestión de esta por parte de los empleados, lo que implica que una misma empresa podrá disponer de varios usuarios que realicen



facturas y/o albaranes. Por tanto, la aplicación deberá contar con roles que permitan al administrador definir que tipo de permisos tendrán sus empleados y las acciones que estos podrán realizar con la misma.

Es importante que la aplicación permita realizar presupuestos y definir una factura posteriormente a partir de un presupuesto. Esto es así, porque hoy en día muchos trabajos o ventas se producen tras enviar un presupuesto y la aceptación por parte del cliente.

Uno de los grandes atractivos de la aplicación será el uso de esta tanto de forma web como desde la aplicación móvil. Esto permitirá que los usuarios que usen el sistema puedan optar por el sistema operativo y dispositivo que deseen, sin verse limitados por el uso de un sistema que no es de su agrado.

### 2.3 REQUISITOS

A continuación, se detallan los requisitos funcionales del aplicativo:

- Crear un usuario (administrador) en plataforma.
- Modificación de los datos de la empresa.
- Invitación a otros usuarios / gestores de la compañía.
- Invitación a otras compañías.
- Gestión de los roles de los usuarios disponibles.
- Operaciones básicas (registrar, visualizar, editar y eliminar) de las compañías (cliente, proveedor o acreedor y con ambos perfiles)
- Operaciones básicas (registrar, visualizar, editar y eliminar) de los productos o servicios
- Gestión del stock.
- Operaciones básicas (registrar, visualizar, editar y eliminar) de los presupuestos.
- Operaciones básicas (registrar, visualizar, editar y eliminar) de las facturas.
- Generación de la factura a partir de un presupuesto o otra factura anterior.
- Generación de una factura a partir de una factura electrónica anterior.
- Adjunta documentos e imágenes a una factura.
- Descargar la versión electrónica española de la factura.
- Descargar un fichero comprimido con la factura en PDF y todos sus documentos e imágenes adjuntos.
- Operaciones básicas (registrar, visualizar, editar y eliminar) de los pagos.
- Imprimir en PDF tanto de los detalles de una compañía, artículo, servicio, pago, presupuesto o factura.
- Generación de plantillas de presupuestos y facturas para la impresión y descarga en PDF.
- Generación y descarga en formato electrónico de la factura, siguiendo las normas y pautas marcadas por la legislación española.
- Creación de gráficas en la pantalla principal a partir de los datos disponibles.
- Generación de informes acerca de los clientes, proveedores y acreedores registrados.



- Generación de informes de pagos, presupuestos y facturas.
- Filtros y búsquedas sobre productos, servicios, compañías y pagos
- Filtros y búsquedas avanzadas para los presupuestos y facturas.
- Compartir factura y presupuesto en las principales vías electrónicas de comunicación: Correo, WhatsApp, Telegram, etc.
- Sincronización de las fechas establecidas en los pagos, presupuestos y facturas en el calendario del dispositivo.
- Enlace público para la visualización de un presupuesto o factura.
- Visualización (terceros) de los presupuestos y facturas emitidas por los usuarios de la compañía.
- Descarga (terceros) de los presupuestos y facturas tanto en PDF como en formato de factura electrónica.
- Pasarela de pago (terceros) para el pago de la factura.
- Acceso a la aplicación sin conexión a internet.
- Sincronización de los datos locales con los disponibles en remoto.
- Copia de seguridad manual de los datos registrados.
- Sincronización y copia de seguridad automática con servicios en la nube: Google Drive, OneDrive, Dropbox...
- Traducción del aplicativo en inglés.

A continuación, se detallan los requisitos no funcionales:

- No se puede imprimir en PDF una factura sin registrarlos previamente.
- No se puede descargar en ningún formato una factura sin ser registrada.
- Los enlaces públicos no podrán ser accesibles tras 15 días de su generación.
- No se debe permitir modificar una factura si su estado es aceptada o rechazada.

## 2.4 UML Y DIAGRAMAS DE CASOS DE USO

El diagrama de bases de datos UML permite la visión gráfica de las entidades de la base de datos y como éstas se van a relacionar entre sí. Con herramientas de alto nivel incluso se puede pasar de dicho diagrama UML a la generación o actualización de la base de datos. Además, estos diagramas UML sirven como herramienta para conocer cómo está construida la base de datos sin tener que investigar las tablas y los datos que están generados dentro de ellas.

En la figura 2.4.A se muestra el diagrama UML de base de datos que seguirá GIC. Tal y como se observa, existen dos entidades importantes en la plataforma: los usuarios y las facturas, dado que de ellas es de donde más relaciones se distinguen.

Un usuario puede tener las siguientes relaciones:



- Puede formar parte de una o varias compañías. Por norma general, un usuario formará parte de una única compañía, pero puede darse el caso de que un gestor quiera gestionar varias compañías con un mismo usuario.
- Puede generar ningún o muchos productos o servicios. El usuario tiene la opción de generar los impuestos para usarlos posteriormente en las facturas o no generar ninguno y generarlos todos en caliente en la factura. Además, a los productos o servicios se les puede agregar un impuesto o dejarlo sin agregar por si a posteriori se quiere agregar a la hora de generar la factura.
- Puede generar ningún o muchos pagos. Es evidente que el usuario tiene la posibilidad de agregar sus pagos, por tanto, si no desea gestionar los pagos nunca generará un pago dentro del aplicativo. Pero también tiene la posibilidad de agregar todos los pagos que desee.
- Puede generar ningún o muchos presupuestos o facturas: La acción principal de un usuario dentro del aplicativo, será la generación de presupuestos y facturas. Por tanto, normalmente al entrar al aplicativo un usuario puede no tener facturas o presupuestos generados, pero con el paso del tiempo y por norma general generará gran cantidad de estos.

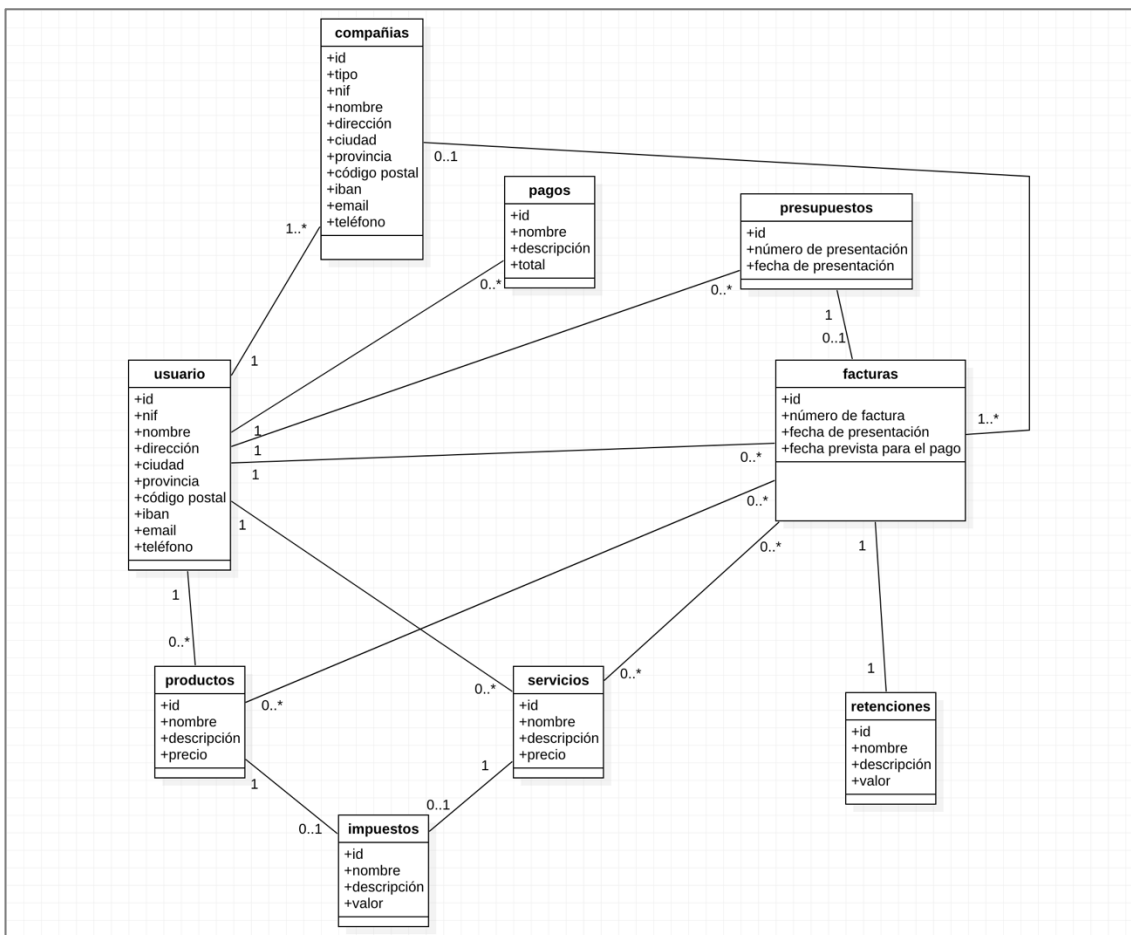


FIGURA 2.4.A – DIAGRAMA UML DE GIC

Una factura puede tener las siguientes relaciones:

- Una a ninguna o a una. Un presupuesto puede convertirse en una factura o jamás convertirse en una factura.
- Una factura puede tener ninguno o muchos productos. Una factura puede tener un producto o un servicio, pero no ambos, con lo que cabe la posibilidad de que alguno sea cero. Además, en una factura pueden existir tanto productos como servicios y ambos en grandes cantidades.
- Una factura debe tener una retención. Todas las facturas deben tener una retención, aunque esta sea 0 deberá aparecer reflejada.

Por otro lado, los diagramas de casos de uso reflejan las acciones que los perfiles que existan pueden realizar en el aplicativo. A continuación, se va a explicar el diagrama de casos de uso (Figura 2.4.B) basado en las necesidades que presenta el prototipo que se va a realizar (definido en el punto 3.1 de esta memoria) y los perfiles y acciones que pueden llevar estos.

El único perfil que tendremos en el prototipo será el de usuario que definiremos como “CLIENTE” y podrá realizar las operaciones CRUD sobre su perfil de compañía (en este caso no tendrá la posibilidad de eliminar su perfil), facturas, presupuestos, compañías, pagos, productos y servicios. Además, un usuario podrá descargar sus facturas o presupuestos y compartir su perfil, facturas y presupuestos.

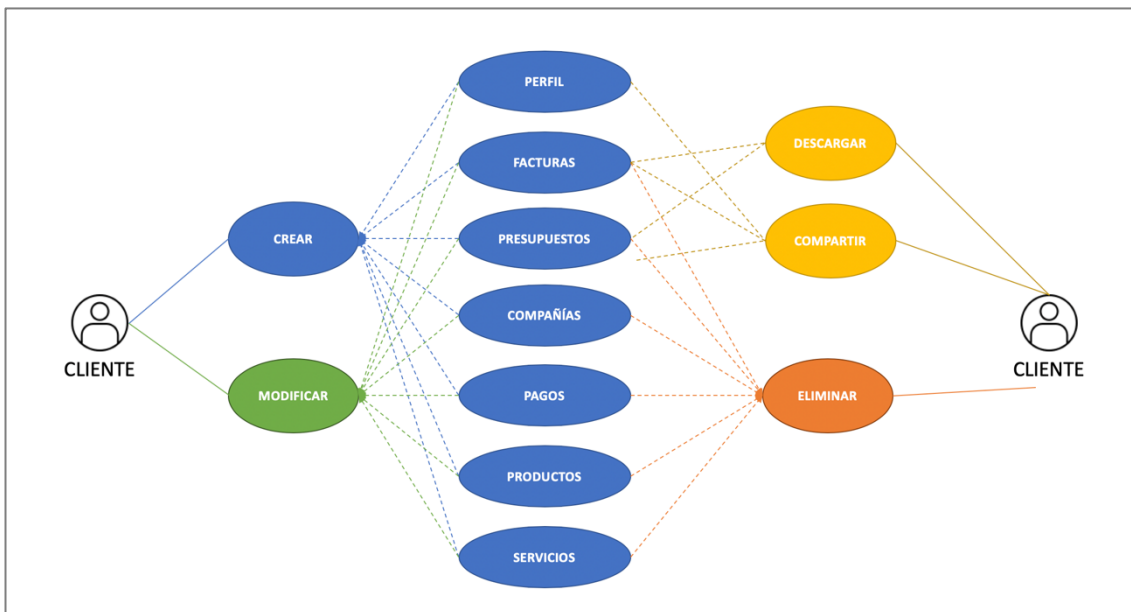


FIGURA 2.4.B – DIAGRAMA DE CASOS DE USO DEL PROTOTIPO DE GIC



## 2.5 MOCKUPS, DISEÑO DE LA INTERFAZ Y PALETA DE COLORES

### 2.5.1 MOCKUPS

Los mockups sirven para comenzar a definir el diseño que tendrá el aplicativo. Es importante diseñar los mockups de forma que representen lo que quiere crear y siempre buscando que sean lo más simple posibles, dado que a posteriori cuando se genere el diseño final de la interfaz es cuando se le añadirán los detalles necesarios para que sean todo lo completos que se requieran.

Para los mockups, se ha tenido en cuenta cuál es objetivo del prototipo y el alcance que se va a realizar (definidos completos en el punto 3.1 de esta memoria). En todos ellos se puede apreciar que se busca que el aplicativo sea lo más simple y que cualquier usuario con pocos conocimientos en tecnología sea capaz de navegar y usar la aplicación sin ningún tipo de problemas:

- En el login del aplicativo, se mostrará dos inputs en los que el usuario podrá introducir el correo y la contraseña tanto para iniciar sesión como para registrarse en la plataforma.



Mockup de la interfaz de inicio de sesión (login) para el aplicativo GIC. El diseño muestra un encabezado con el título "GIC - INICIO DE SESIÓN". Debajo del encabezado, hay dos campos de entrada de texto: "CORREO" con el placeholder "Escriba su correo" y "CONTRASEÑA" con el placeholder "Escriba su contraseña". Debajo de los campos de entrada, hay dos botones de acción: "INICIAR SESIÓN" y "REGISTRARSE".

FIGURA 2.5.1.A – MOCKUP DE LA VISTA DEL LOGIN





- Home: Mostrará las gráficas generadas a partir de los datos de facturas y pagos gestionados en el aplicativo. Además, tendrá el menú del aplicativo en la zona izquierda y sólo mostrará el Home y las 5 entidades que manejamos: Facturas, compañías, productos y servicios, pagos y mi compañía.

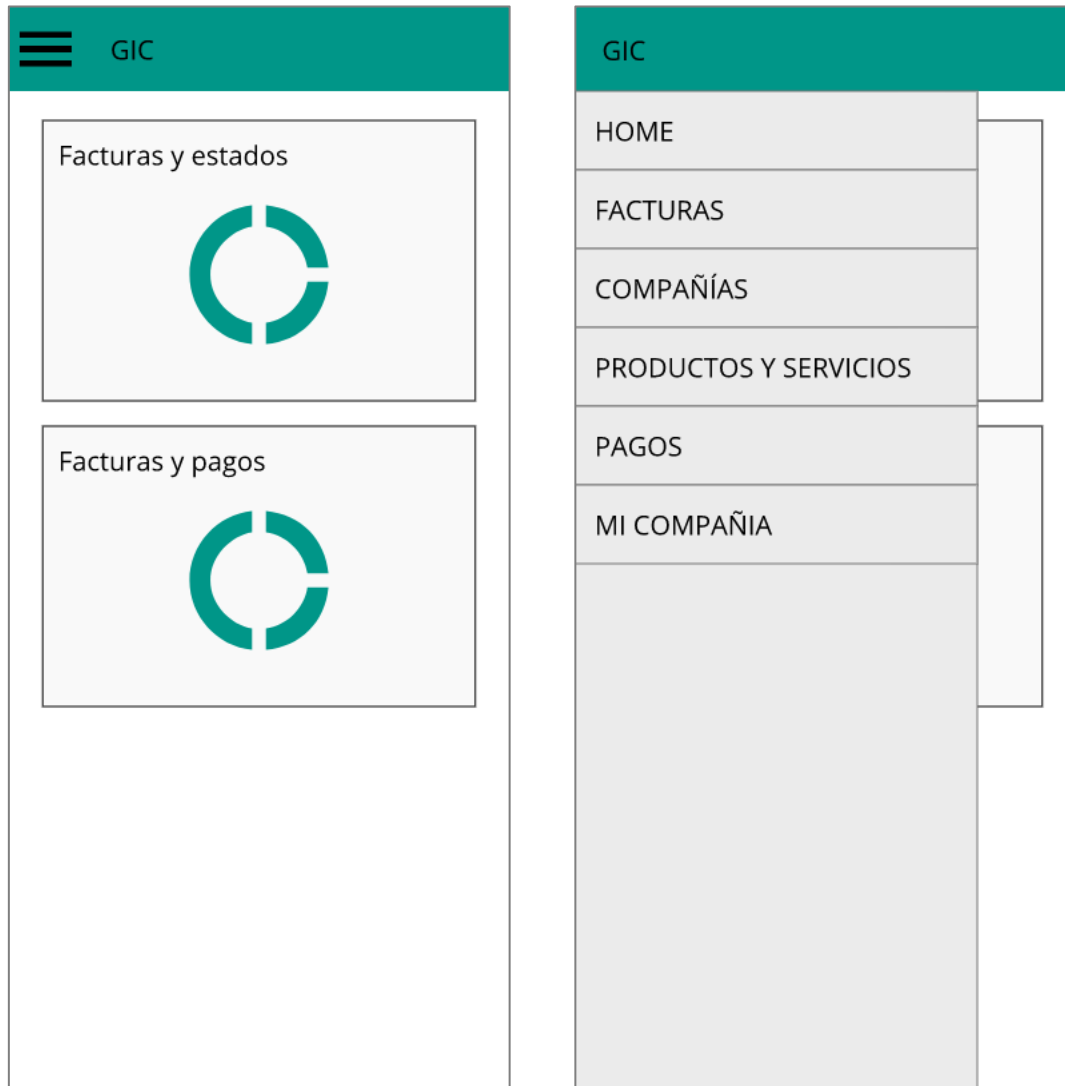
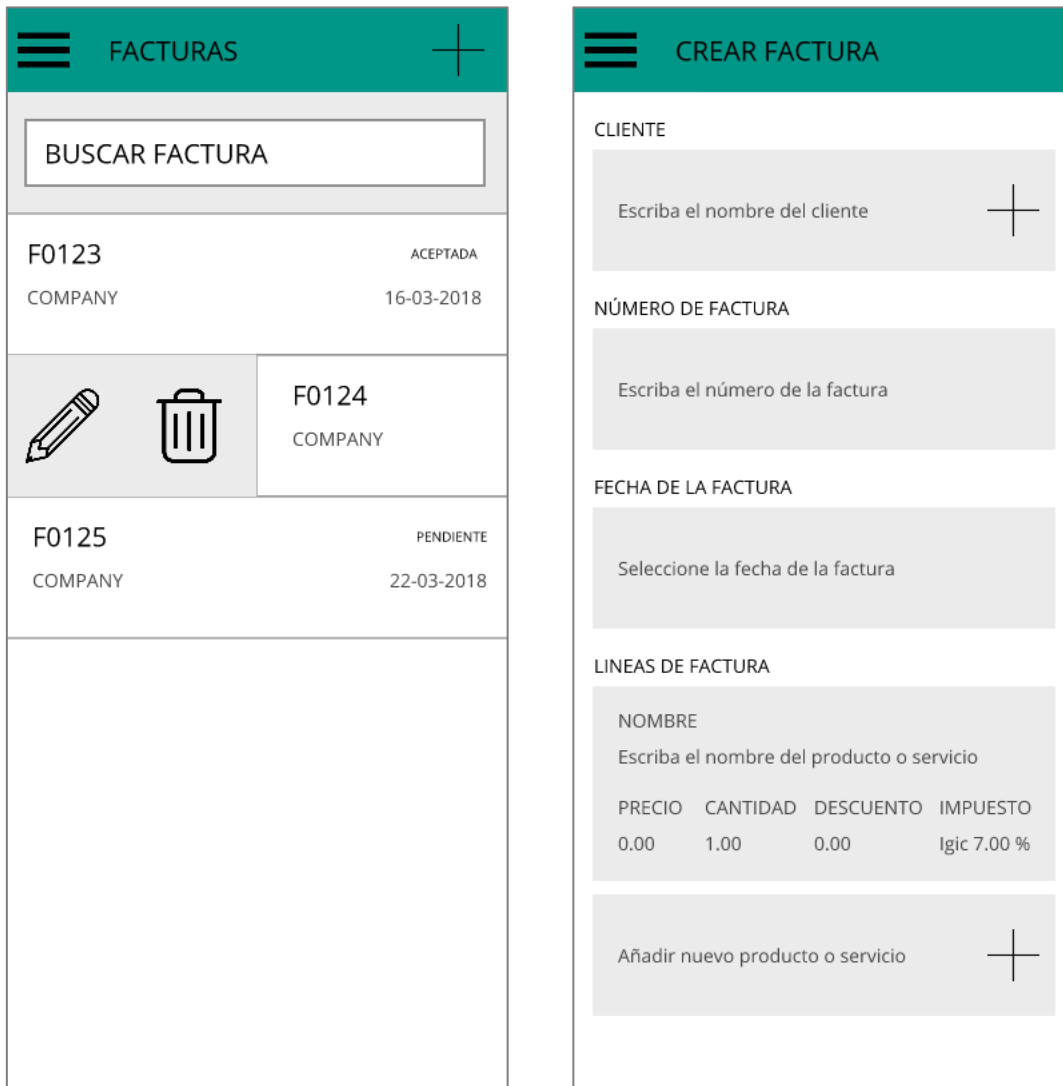


FIGURA 2.5.1.B – MOCKUP DE LA HOME Y EL MENÚ

- Facturas: En vista del listado de las facturas es donde aparecerá el botón de agregar nueva factura y el listado de las facturas ya generadas. De las facturas generadas se podrá ver un mínimo de información (número de la factura, estado, cliente y la fecha de presentación) y al deslizar sobre ella se podrá visualizar las posibles acciones sobre dicha factura.



The mockup consists of two side-by-side panels. The left panel, titled 'FACTURAS', shows a search bar 'BUSCAR FACTURA' and a list of three invoices:

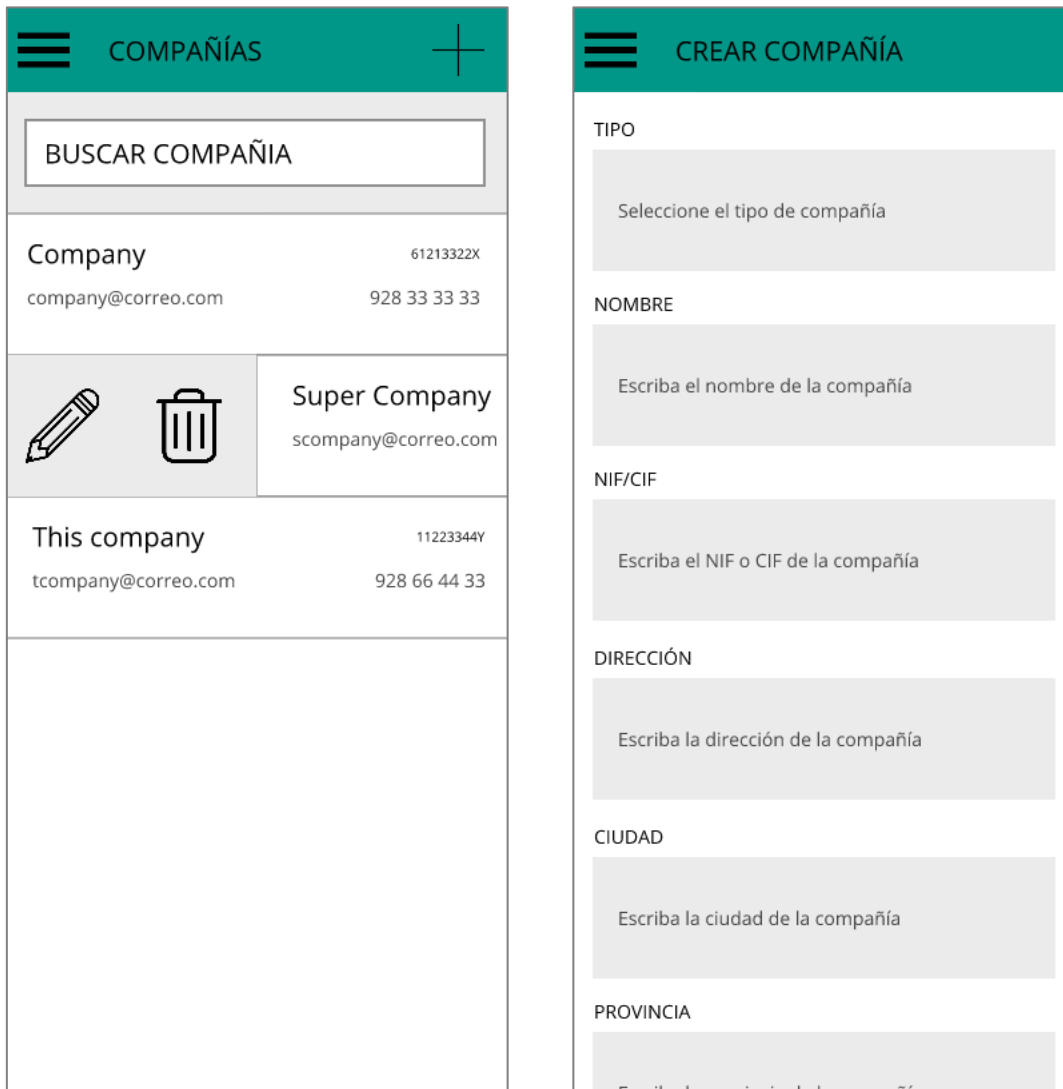
ID	COMPANY	STATUS	DATE
F0123	COMPANY	ACEPTADA	16-03-2018
F0124	COMPANY		
F0125	COMPANY	PENDIENTE	22-03-2018

The right panel, titled 'CREAR FACTURA', is a form with the following sections:

- CLIENTE:** Input field 'Escriba el nombre del cliente' with a plus icon.
- NÚMERO DE FACTURA:** Input field 'Escriba el número de la factura'.
- FECHA DE LA FACTURA:** Input field 'Seleccione la fecha de la factura'.
- LINEAS DE FACTURA:**
  - Section 'NOMBRE' with input 'Escriba el nombre del producto o servicio'.
  - Table with columns: PRECIO, CANTIDAD, DESCUENTO, IMPUESTO. Values: 0.00, 1.00, 0.00, Igtc 7.00 %.
  - Input field 'Añadir nuevo producto o servicio' with a plus icon.

FIGURA 2.5.1.C – MOCKUP DE LAS FACTURAS Y LA CREACIÓN DE LA FACTURA



- **Compañías:** En vista del listado de las compañías es donde aparecerá el botón de agregar nueva compañía y el listado de las compañías ya generadas. De las compañías generadas se podrá ver un mínimo de información (nombre, NIF o CIF, correo y teléfono de la compañía) y al deslizar sobre ella se podrá visualizar las posibles acciones sobre dicha compañía.



**COMPAÑÍAS**

BUSCAR COMPAÑÍA

Company	61213322X
company@correo.com	928 33 33 33

  Super Company  
scompany@correo.com

This company	11223344Y
tcompany@correo.com	928 66 44 33

**CREAR COMPAÑÍA**

TIPO  
Seleccione el tipo de compañía

NOMBRE  
Escriba el nombre de la compañía

NIF/CIF  
Escriba el NIF o CIF de la compañía

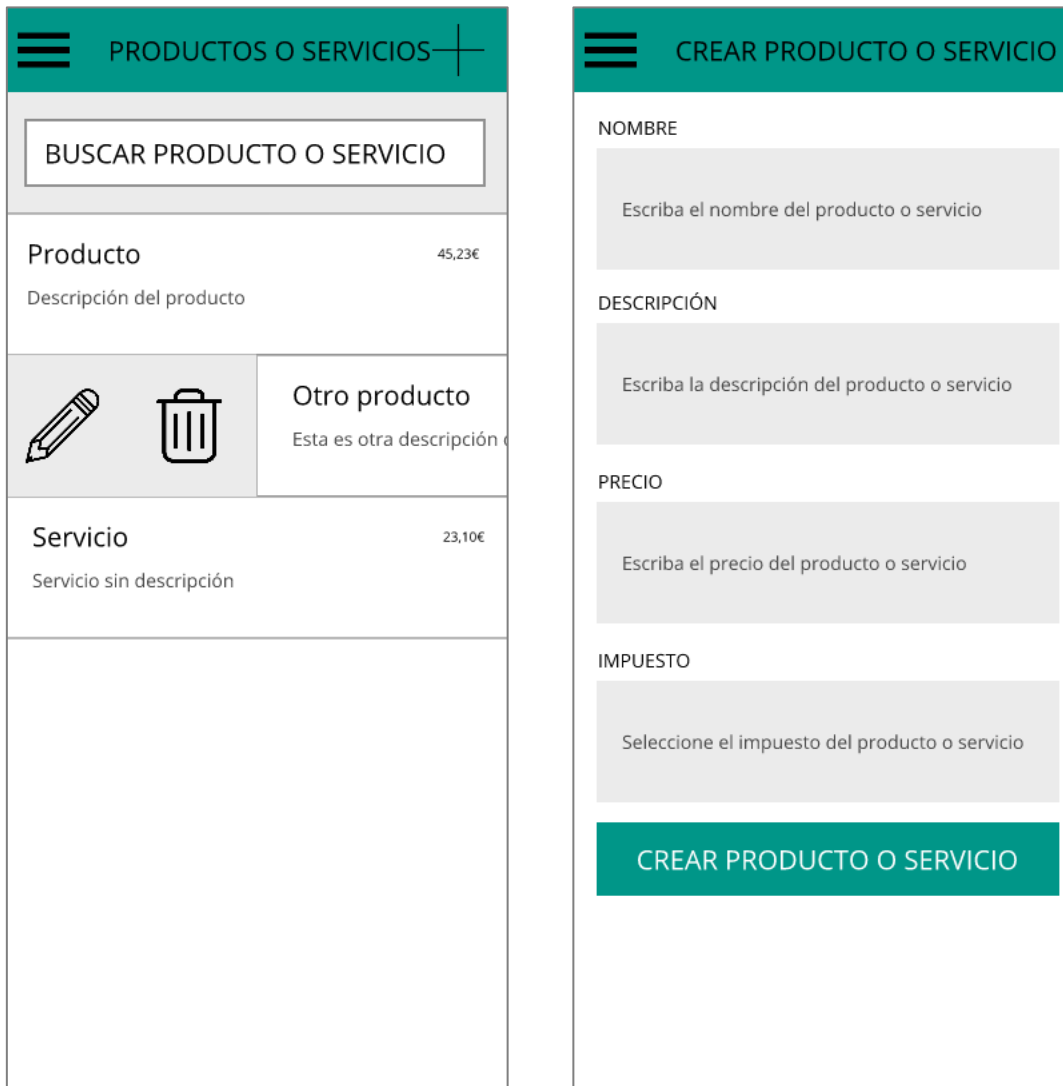
DIRECCIÓN  
Escriba la dirección de la compañía

CIUDAD  
Escriba la ciudad de la compañía

PROVINCIA  
Escriba la provincia de la compañía

FIGURA 2.5.1.D– MOCKUP DE LAS COMPAÑÍAS Y LA CREACIÓN DE LA COMPAÑÍA

- **Productos y servicios:** En vista del listado de los productos y servicios es donde aparecerá el botón de agregar nuevo producto o servicio y el listado de productos o servicios ya generados. De los productos y servicios generados se podrá ver un mínimo de información (nombre, descripción y precio del producto) y al deslizar sobre ella se podrá visualizar las posibles acciones sobre el producto o servicio.



The figure consists of two side-by-side mockups of a mobile application interface.

**Left Mockup: PRODUCTOS O SERVICIOS**

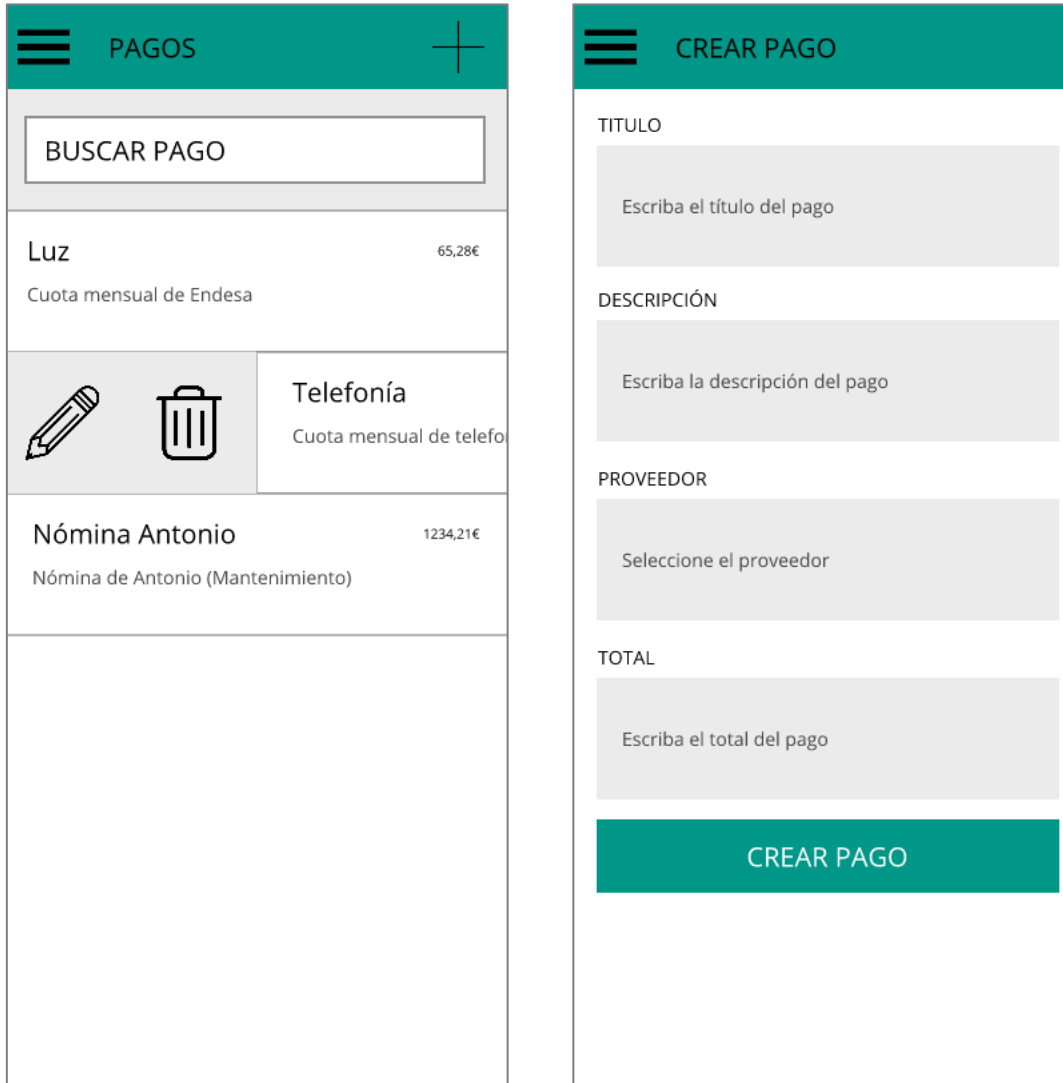
- Header: PRODUCTOS O SERVICIOS
- Search bar: BUSCAR PRODUCTO O SERVICIO
- Product list:
  - Producto** (Price: 45,23€): Descripción del producto
  - Otro producto** (Price: Esta es otra descripción): Includes edit (pencil) and delete (trash) icons.
  - Servicio** (Price: 23,10€): Servicio sin descripción

**Right Mockup: CREAR PRODUCTO O SERVICIO**



- Form fields:
  - NOMBRE**: Escriba el nombre del producto o servicio
  - DESCRIPCIÓN**: Escriba la descripción del producto o servicio
  - PRECIO**: Escriba el precio del producto o servicio
  - IMPUESTO**: Seleccione el impuesto del producto o servicio
- Submit button: CREAR PRODUCTO O SERVICIO

FIGURA 2.5.1.E – MOCKUP DE LOS PRODUCTOS Y SERVICIOS, Y LA CREACIÓN DE ESTOS

- **Pagos:** En vista del listado de los pagos es donde aparecerá el botón de agregar nuevo pago y el listado pagos ya generados. De los pagos generados se podrá ver un mínimo de información (nombre, descripción y total del pago) y al deslizar sobre ella se podrá visualizar las posibles acciones sobre el pago.



The image shows two mobile application screens. The left screen, titled 'PAGOS', features a search bar labeled 'BUSCAR PAGO' and a list of payment items. The right screen, titled 'CREAR PAGO', is a form for creating a new payment with fields for 'TITULO', 'DESCRIPCIÓN', 'PROVEEDOR', and 'TOTAL', followed by a 'CREAR PAGO' button.

PAGOS	
BUSCAR PAGO	
Luz	65,28€
Cuota mensual de Endesa	
 	<b>Telefonía</b> Cuota mensual de telefo
Nómina Antonio	1234,21€
Nómina de Antonio (Mantenimiento)	

CREAR PAGO	
TITULO	Escriba el titulo del pago
DESCRIPCIÓN	Escriba la descripción del pago
PROVEEDOR	Seleccione el proveedor
TOTAL	Escriba el total del pago
CREAR PAGO	

FIGURA 2.5.1.F – MOCKUP DE LOS PAGOS Y LA CREACIÓN DEL PAGO

- **Mi compañía:** En esta vista el usuario podrá modificar los datos de su compañía que posteriormente se verán reflejados en las facturas cuando se vayan generando.

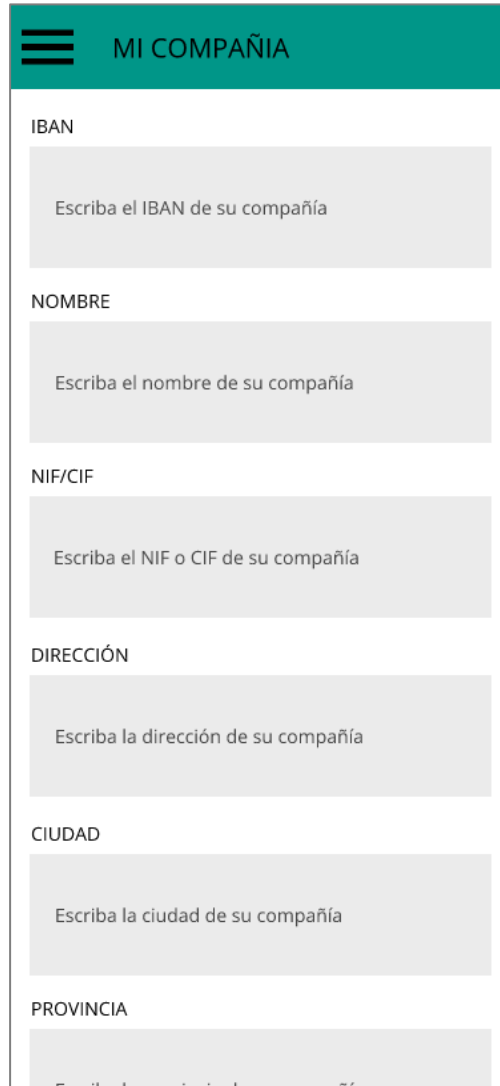


FIGURA 2.5.1.G – MOCKUP DE MI COMPAÑÍA

## 2.5.2 DISEÑO DE LA INTERFAZ

Al comenzar con el desarrollo, se ha intentado seguir la línea marcada por los mockups mostrados en el apartado anterior y buscar la funcionalidad ante el diseño. Una vez logrado la funcionalidad, se ha buscado diferenciar a GIC del resto de aplicaciones y buscar que no parezca una aplicación tipo de dentro de cada uno de los sistemas operativos.

Para lograr diferenciar GIC del resto, se ha utilizado una paleta de colores (apartado 2.5.3 de esta memoria) y se han diseñado estilos para los inputs y los labels, de forma que toda la aplicación aplique el mismo estilo sin importar de qué sistema operativo estemos hablando. Además, se ha establecido un diseño redondeado a todos los elementos para que sigan el conjunto y apliquen el mismo diseño.

Como se puede mostrar en las figuras que aparecen a continuación, todo el aplicativo sigue una misma línea de diseño y permite disfrutar al usuario de continuidad y seguridad a lo largo de todas las vistas del aplicativo. Es muy importante esto, dado que desde que el usuario domine una de las acciones ya podrá hacer uso de todo el aplicativo, dado



que ha sido construido en base al mismo diseño y la misma forma de trabajar en todas sus vistas, de forma que se muy fácil el adaptarse y comenzar a usarlo.

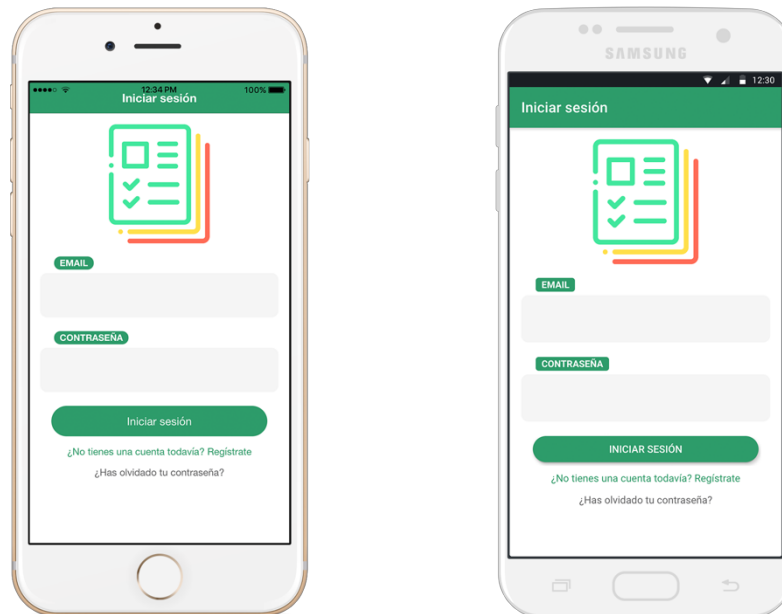


FIGURA 2.5.2.A – VISTA DEL DISEÑO DE LA PATALLA DE INICIO

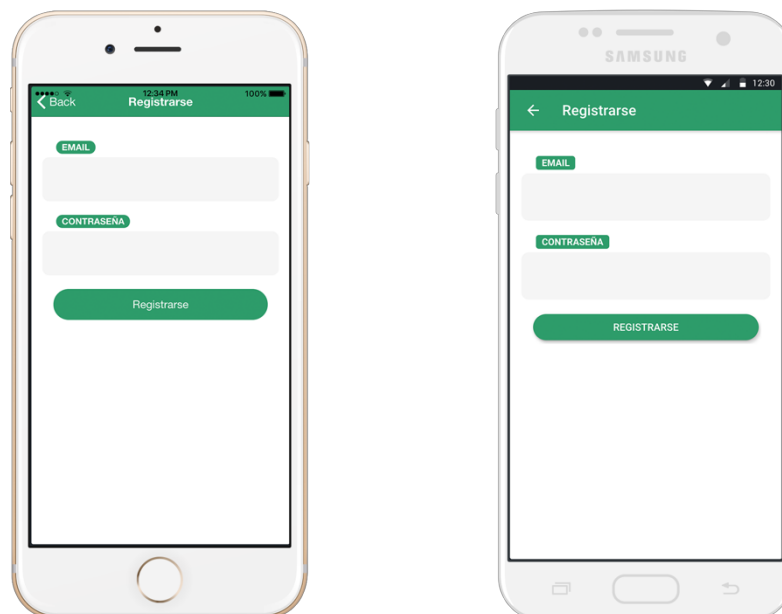


FIGURA 2.5.2.B – VISTA DEL DISEÑO DE LA PANTALLA DE REGSITRO



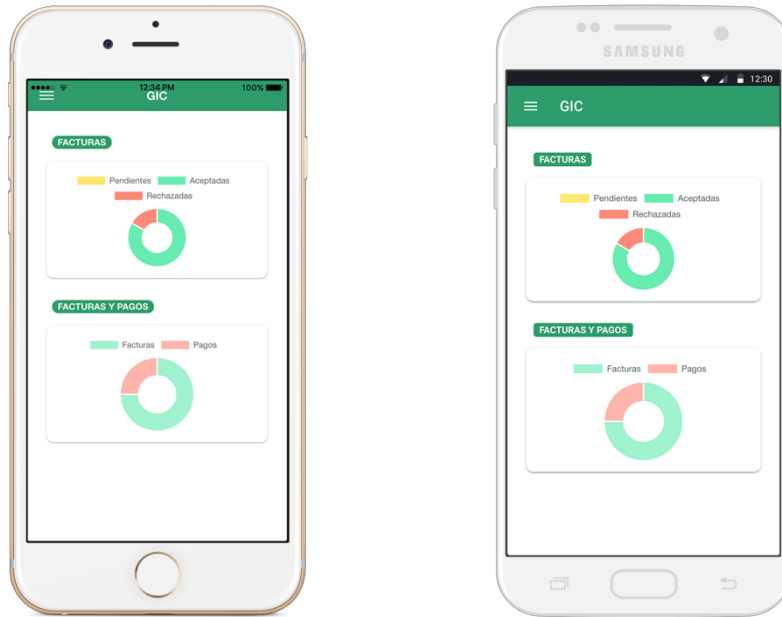


FIGURA 2.5.2.C – VISTA DEL DISEÑO DE LA PATALLA DE HOME

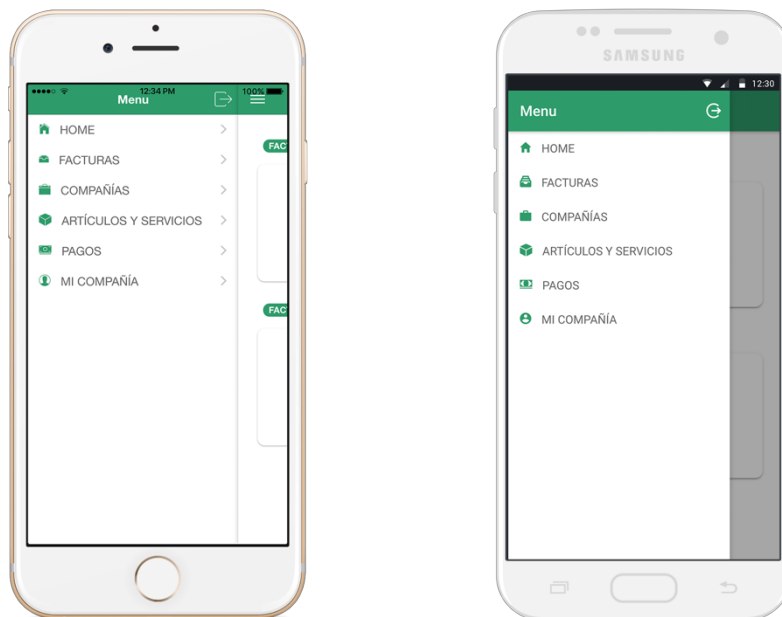


FIGURA 2.5.2.D – VISTA DEL DISEÑO DEL MENU



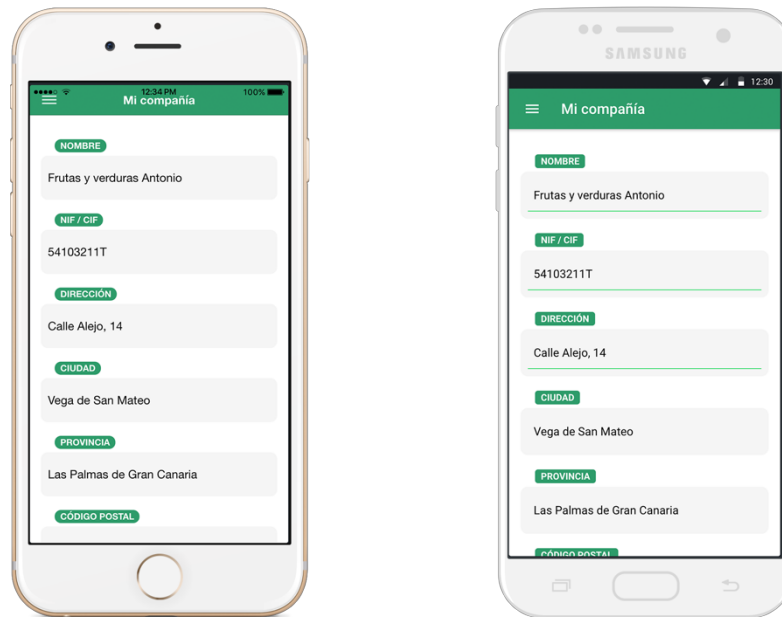


FIGURA 2.5.2.E – VISTA DEL DISEÑO DE LA PATALLA DE MI COMPAÑÍA

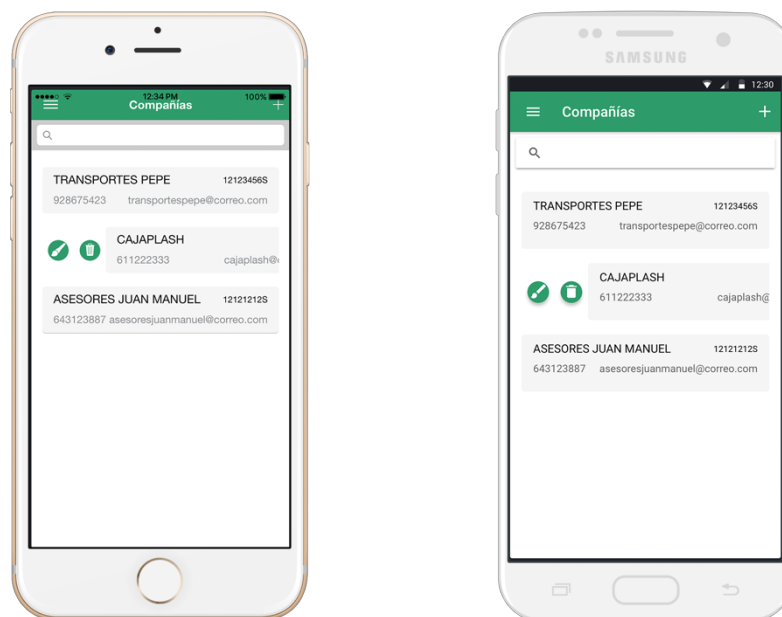


FIGURA 2.5.2.F – VISTA DEL DISEÑO DE LA PATALLA DEL LISTADO DE COMPAÑÍAS

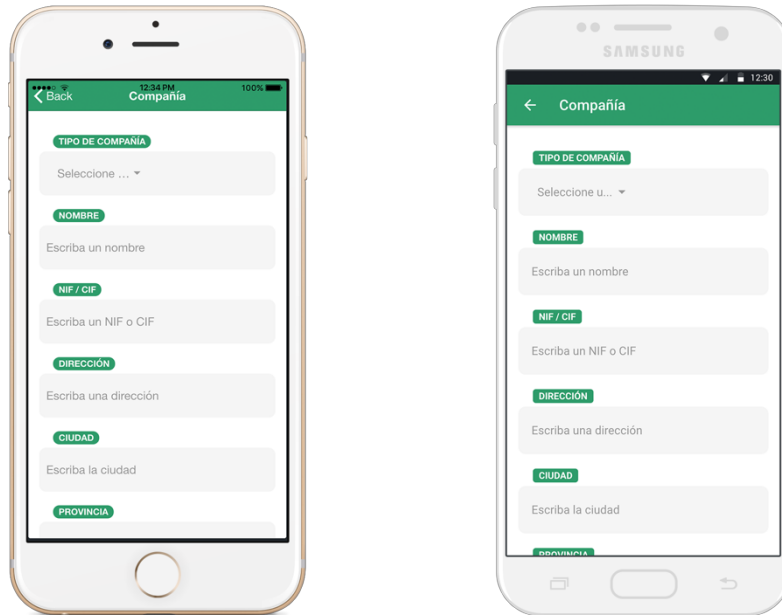


FIGURA 2.5.2.G – VISTA DEL DISEÑO DE LA PATALLA DE NUEVA COMPAÑÍA

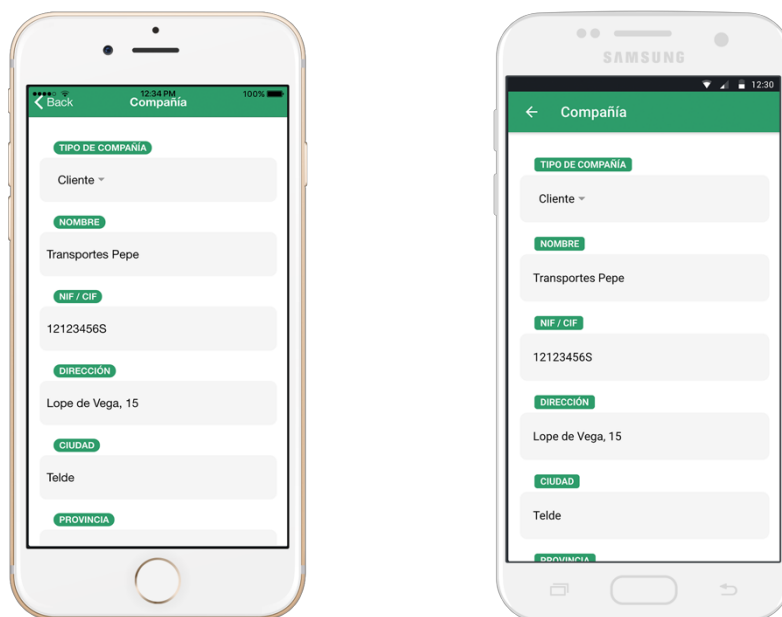


FIGURA 2.5.2.H – VISTA DEL DISEÑO DE LA PATALLA DETALLA DE LA COMPAÑÍA

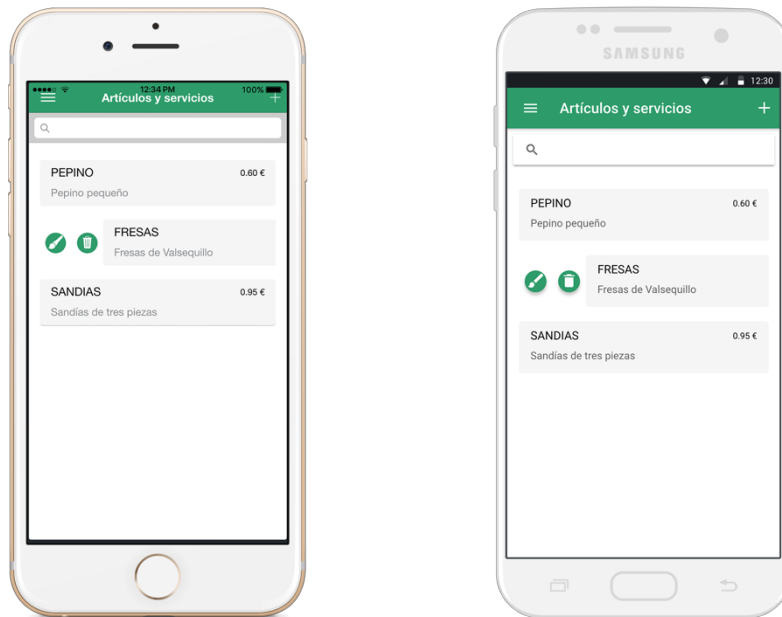


FIGURA 2.5.2.I – VISTA DEL DISEÑO DE LA PATALLA DEL LISTADO DE PRODUCTOS Y SERVICIOS

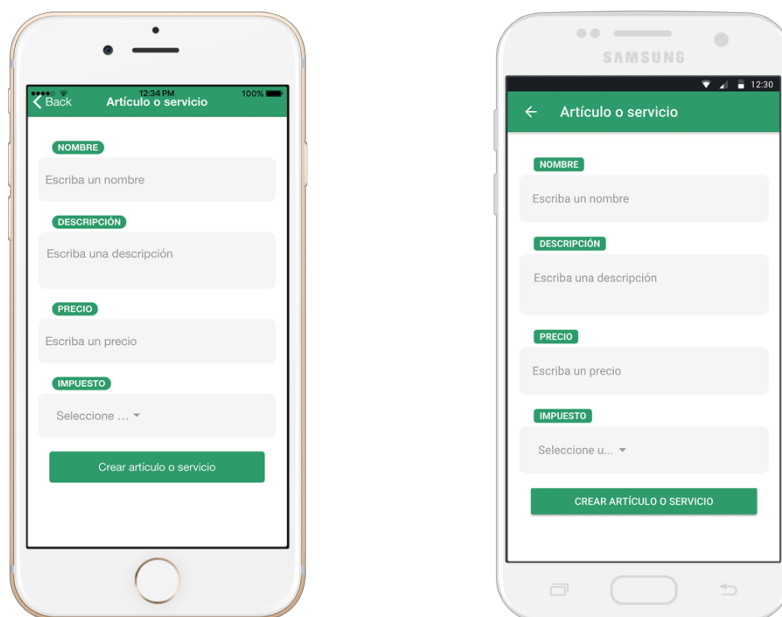


FIGURA 2.5.2.J – VISTA DEL DISEÑO DE LA PATALLA DE NUEVO PRODUCTO O SERVICIO

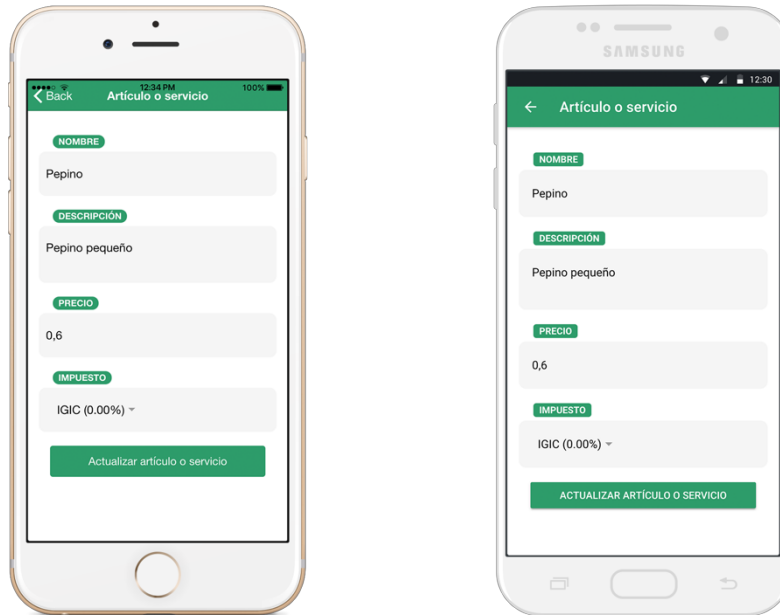


FIGURA 2.5.2.K – VISTA DEL DISEÑO DE LA PANTALLA DETALLE DEL PRODUCTO O SERVICIO

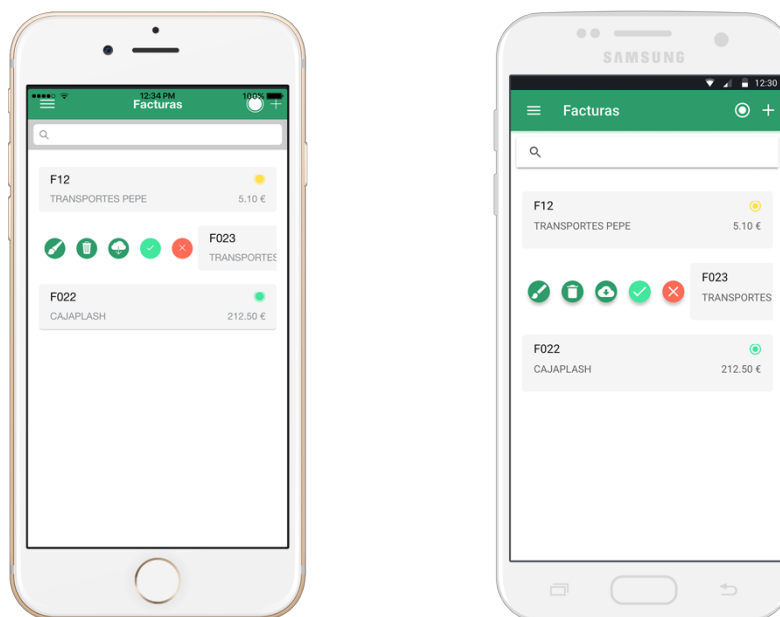


FIGURA 2.5.2.L – VISTA DEL DISEÑO DE LA PANTALLA DEL LISTADO DE LAS FACTURAS

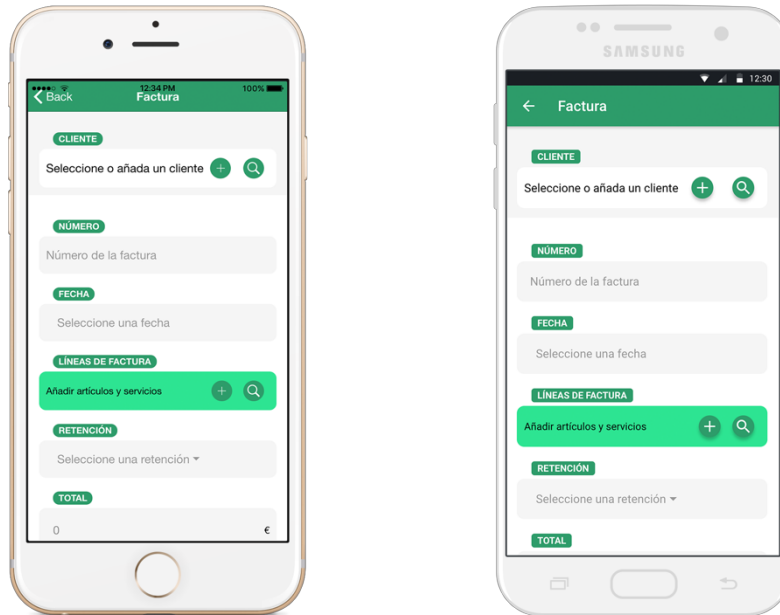


FIGURA 2.5.2.M – VISTA DEL DISEÑO DE LA PANTALLA DE CREAR FACTURA

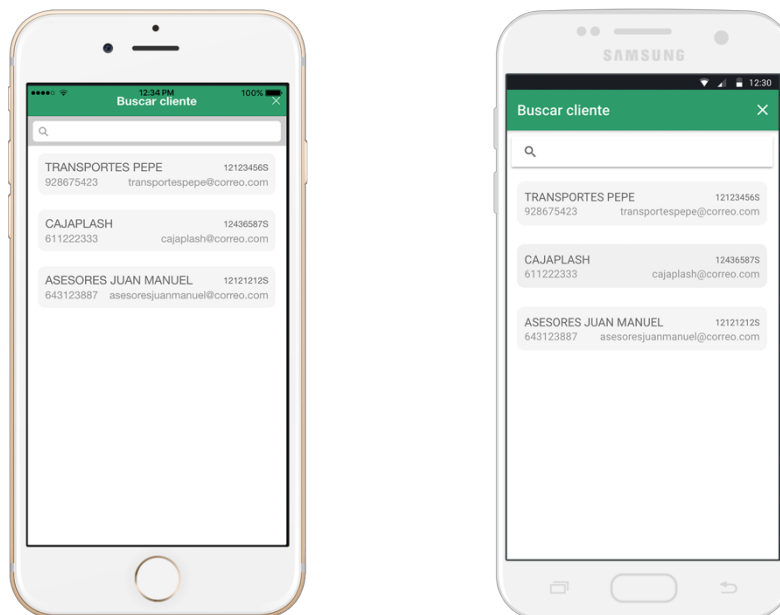


FIGURA 2.5.2.N – VISTA DEL DISEÑO DEL MODAL DE AÑADIR CLIENTE A LA FACTURA

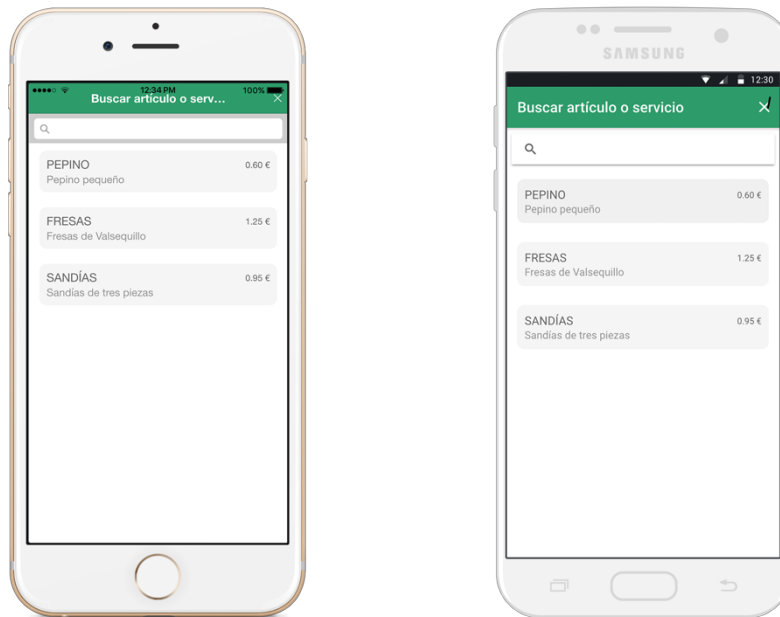


FIGURA 2.5.2.L – VISTA DEL DISEÑO DEL MODAL DE AÑDIR UN ARTÍCULO O SERVICIO A LA FACTURA

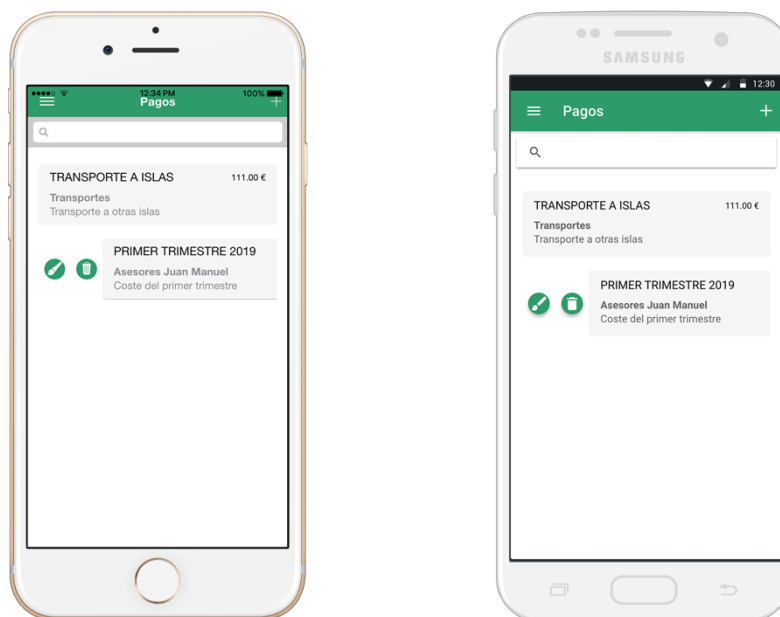


FIGURA 2.5.2.M – VISTA DEL DISEÑO DEL DISEÑO DE LA PANTALLA DEL LISTADO DE PAGOS

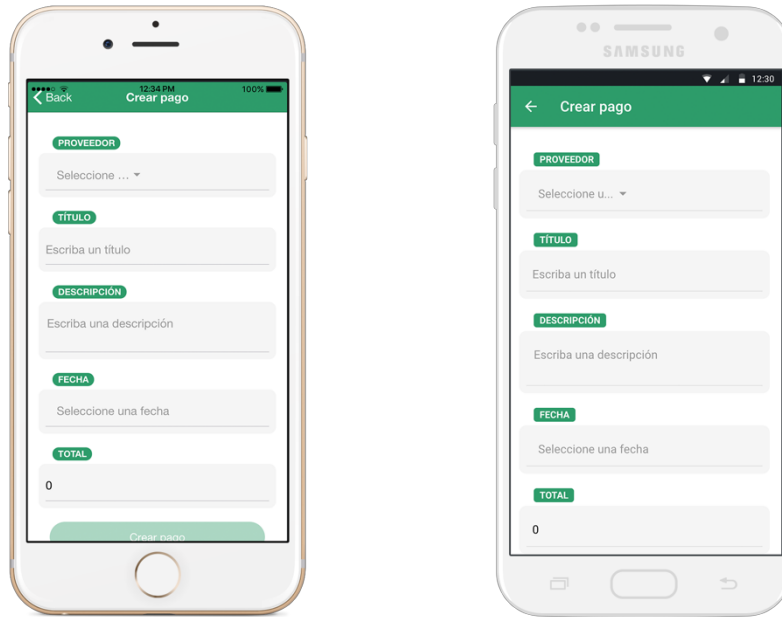


FIGURA 2.5.2.N – VISTA DEL DISEÑO DEL DISEÑO DE LA PATALLA DE CREAR PAGO

### 2.5.3 PALETA DE COLORES

PRIMARIO	SECUNDARIO	ADVERTENCIA	ALERTA	INFORMACIÓN
45, 156, 106	66, 231, 157	255, 225, 74	255, 107, 87	103, 103, 104

Es importante destacar el por qué del uso de esta paleta de colores, dado que, aunque pueda parecer un detalle simple, los usuarios valoran que la paleta elegida sea homogénea y que tenga sentido dentro del aplicativo.

Como se puede apreciar, los colores elegidos cazan entre ellos y se puede entender el uso de estos en la aplicación:

- El color principal del aplicativo será un verde oscuro. Dado que se trata de una plataforma que controlará las facturas y gastos de una compañía, es importante que el color principal de la sensación al usuario de que todas sus finanzas funcionan correctamente y de ahí la elección de un color verde.
- El color secundario es un color verde más vivo, utilizado para marcar las facturas como aceptas y para resaltar ciertas acciones importantes del aplicativo



- El color de advertencia es un amarillo y su uso principal será para marcar las facturas como pendientes hasta su cambio de estado hacia aceptadas o rechazadas.
- El color de alerta es un rojo, principalmente por la estrecha relación que este color tiene con una alerta o algo importante de visualizar, como pueden ser pagos o resaltar los mismos en gráficas y que se puede apreciar fácilmente por el usuario que haga uso del aplicativo en el caso de que fuese necesaria alguna acción suya.
- Por último, se hace necesario el uso de algún color plano para mostrar información y que haga cómoda la lectura de los datos por parte de los usuarios de la plataforma.

Como se aprecia la elección de estos cinco colores viene dada en gran medida por la sensación de que todo está funcionando y viene estrechamente relacionada con los tres estados que el aplicativo permite definir a una factura: aceptada, pendiente y rechazada.



Es por ello por lo que los colores dan forma al logotipo de la aplicación y definen exactamente la principal meta a conseguir, que no es otra que la gestión cómoda y simple de las facturas.

## 2.6 MODELO DE NEGOCIO

Antes de definir el modelo de negocio, es importante destacar que el aplicativo está destinado a autónomos y pequeñas empresas. Por tanto, tenemos que tener en cuenta que los recursos económicos de los que disponen este tipo de empresas no suelen ser muy grandes, sobre todo en los comienzos de estas.

El modelo de negocio que se va a utilizar es el de suscripción mensual con pago anual y en el que existirá varios modelos, de forma que el usuario pueda escoger la opción de pago que se adapte a sus necesidades:

- Gratuita: Un usuario y hasta 10 facturas mensuales. Limitada.
- S: Un usuario y hasta 100 facturas mensuales.
- M: Dos usuarios y hasta 250 facturas mensuales.
- L: Tres usuarios y hasta 500 facturas mensuales.
- XL: Tres usuarios sin límite de facturas mensuales.





La existencia de un modelo gratuito se debe a la necesidad que existe de probar por parte de los clientes la plataforma y la aplicación antes de suscribirse a uno de los modelos de pago. La versión gratuita viene limitada para evitar que existan usuarios que puedan generar facturas indefinidamente sin pasarse a una de las modalidades de pago. Donde las limitaciones son:

- No poder descargar ni compartir facturas ni presupuestos en PDF o en formatos electrónicos.
- En la vista pública de los presupuestos o facturas estará desactivada la opción de descargar en PDF o formato electrónico.
- No se podrán generar más de 10 compañías.
- No se podrán generar más de 10 productos o servicios.
- No tendrá la gestión de stock.
- No existirá la opción de activar los portales para los clientes a los que se generen presupuestos o facturas. Esto implica que tampoco tendrá la opción de pago por pasarela de pago para terceros.

Cabe mencionar que todas las modalidades de pago disponen de las mismas características y sólo se diferencian por la cantidad de usuarios y de facturas que se permiten generar en el aplicativo. Es importante que el usuario escoja su modelo de pago por estos motivos y no porque los modelos básicos no le permiten realizar todas las acciones sobre las facturas y obligarlo a seleccionar modelos de mayor coste obligatoriamente.

En cuanto a las modalidades de pago, el usuario podrá pasar a una de mayor tamaño abonando la diferencia (de meses que le queden por pagar) de precio existente entre ambos modelos. También podrá disminuir su modelo, abonando previamente la diferencia entre ambos modelos como sanción.

En las modalidades de pago no se han definido los precios de momento debido a que aún el proyecto se encuentra en una fase inicial con el desarrollo de un prototipo que hace uso de todas las tecnologías y plataformas de la que hará uso el producto final. Es por ello, que todavía no conocemos todos los gastos derivados del desarrollo y mantenimiento del aplicativo, por lo que no podemos definir los precios de las modalidades de pago.

## 2.7 NORMATIVA Y LEGISLACIÓN

La normativa y legislación en un aplicativo como el que estamos desarrollando es muy importante. Esto se debe principalmente a que estamos generando facturas y estas deben tener un mínimo de datos para considerarse una factura legal y aceptable:

- Numeración de la factura. La factura debe estar numerada para poder ser identificada fácilmente en caso de ser necesaria su búsqueda.
- Fecha de presentación. La factura debe reflejar la fecha en la que ha sido presentada al cliente.



- Datos del proveedor. Al presentar una factura, el proveedor está obligado a presentar todos sus datos correctamente al cliente en la factura. Son importantes el NIF o CIF de la empresa, su nombre, dirección y datos de contacto.
- Datos del cliente. Al igual que sucede con los datos del proveedor, los datos del cliente también tienen que estar perfectamente detallados para que se pueda identificar perfectamente a las dos partes.
- Líneas de factura. Las líneas de factura deben cumplir ciertos requisitos como pueden ser mostrar el precio unitario del artículo o servicio, su base imponible o el descuento que lleva asociado en el caso de que lo llevase. Es muy importante el impuesto que lleve asociada una línea de factura, y esto es porque si no se encuentra bien definido, la factura ya no es válida.
- Retenciones. En el caso de que la factura lleve algún tipo de retención, esta debe quedar reflejada perfectamente en la factura.

Como se puede observar, todos los aspectos legales que debe cumplir una factura están incluidos en la factura que se puede generar en el prototipo que se está desarrollando, dado que para que se un prototipo válido deberá cumplir todos los aspectos normativa y legislación que se exigen para la generación de una factura.

Así mismo, una vez generada una factura, es importante que, al descargar la factura en cualquier formato, esta cumpla también todos los aspectos legales. Por tanto, al generar una factura en PDF, la factura debe ser legible y contener todos los aspectos que hemos detallado anteriormente. Así mismo, si lo que se desea es generar una factura electrónica, no se debe generar un formato personalizado, sino seguir las pautas marcadas por el estado en cuestión. En el anexo, se podrá encontrar una guía del formato que debe tener una factura electrónica y como se debe generar la misma para que cumpla la legislación española en este caso, por lo que también es interesante que al aplicativo se le vayan añadiendo la posibilidad de descargar la factura electrónica en diversos formatos en dependencia de en qué país se esté generando la factura.

Otro aspecto fundamental que debe cumplir el aplicativo es la GDPR (General Data Protection Regulation o Reglamento General de Protección de Datos Europeo), que trata de cubrir y proteger la privacidad de los ciudadanos europeos. Esto permite que los ciudadanos europeos tengan derechos respecto a sus datos como puede ser:

- Derecho de acceso. Cualquier usuario del aplicativo podrá solicitar si se tiene información suya. En caso afirmativo, podrá solicitar información acerca de donde se almacena dicha información y para qué y cómo se está utilizando sus datos.
- Derecho al olvido. Cualquier usuario del aplicativo podrá solicitar que se borren sus datos, ya sea porque ya no quiere formar parte del aplicativo, porque ya



hayan recogidos datos suyos sin su consentimiento o cuando se hayan obtenido esos datos de forma ilícita.

- Derecho a la portabilidad. Cualquier usuario del aplicativo podrá solicitar que se porten sus datos hacia otro aplicativo o hacia otra compañía.

Es importante para cumplir todos los derechos que tienen los usuarios del aplicativo, que se informe al usuario de forma simple de para qué y cómo se van a utilizar sus datos. A la hora de aceptar la GPDR dentro del aplicativo no hay que premarcar ninguna opción, sino que debe ser el usuario el que marque todas ellas para asegurar que está entendiendo y aceptando la GPDR. Además, es muy importante tener un canal de comunicación directa y rápida con los usuarios del aplicativo para informar en caso de ataque informático o cualquier cambio sobre las políticas del aplicativo al menor tiempo posible y que el usuario siempre esté informado de la situación de sus datos. Es vital ser un aplicativo transparente y cumplir la GPDR en todos sus puntos de cara al usuario para ser un aplicativo fiable de cara a la cantidad y calidad de datos que almacenamos y manejamos de los usuarios y sus compañías.



## 3 DESARROLLO

### 3.1 ALCANCE DE LA IMPLEMENTACIÓN

El desarrollo de una aplicación como la que estamos tratando, es largo y laborioso. Es muy importante definir fases de desarrollo y establecer cuáles son las prioridades y que desarrollos pueden ser posteriores a tener una versión lanzada, estable y utilizada por usuarios reales.

De cara a la implementación, lo primordial será la generación de una factura. La factura está compuesta por diferentes componentes, modelos e información. Es por ello por lo que es necesario que datos consideramos mínimos para una factura:

1. Información acerca del cliente: Nombre, NIF o CIF, dirección, ciudad, provincia, código postal, teléfono y correo electrónico.
2. Información acerca del proveedor: Nombre, NIF o CIF, dirección, ciudad, provincia, código postal, teléfono y correo electrónico.
3. Número de la factura y fecha de presentación.
4. Líneas de factura (producto o servicio): Nombre, cantidad, precio, descuento e impuesto a aplicar.
5. Retención por aplicar a la factura.
6. Total de la factura. Que será calculado en base a las líneas de factura introducidas y los impuestos y retenciones definidos.
7. Estado de la factura: pendiente, aceptada o rechazada.

Definidos los puntos anteriores, podemos dar por definida la factura y los datos mínimos que necesitaremos para llegarla a generar. Por tanto, tomaremos como factura inicial del aplicativo la que esté formada por los siete puntos anteriores y será la factura sobre la que permitiremos realizar todas las operaciones CRUD (crear, leer, actualizar y eliminar).

Además de las operaciones CRUD sobre la factura, se hace necesario permitir en una fase inicial el cambio de estados en la factura. Dado que sobre el estado de la factura se define el estado de la compañía que hace uso del aplicativo. En una primera versión, se permitirá realizar un cambio de estado hacia cualquiera de sus posibles estados. Sin embargo, en futuras versiones esto vendrá parametrizado por la configuración que desee realizar el cliente, por lo que se podrá restringir los cambios de estado. Por ejemplo, una factura que ya ha sido aceptada o rechazada no podrá volver al estado de pendiente. Así mismo, los estados iniciales serán pendiente, aceptada o rechazada. Estos estados irán creciendo a medida que el aplicativo se desarrolle en siguientes desarrollos.

Una factura no sirve de nada si no se puede llevar al cliente. Por tanto, en la primera fase se podrá descargar una versión simplificada en PDF. Aunque la factura en PDF será simple y contendrá toda la información, es importante, dotar al aplicativo en próximas versiones de plantillas y selección de colores para que un usuario, podrá configurar que plantilla desea utilizar y con la que se sienta más identificado.

Como es lógico, a la hora de presentar una factura se pueden definir los productos o servicios. Pero es de gran utilidad que dichos productos o servicios, puedan ser generados con anterioridad y que se puedan añadir a la factura de forma cómoda. Es por ello



por lo que, en una primera versión de la aplicación, se podrán realizar operaciones CRUD sobre productos y servicios, que constarán de las siguientes características:

1. Nombre.
2. Descripción.
3. Precio.
4. Impuesto.

Aunque estos son los datos que tomaremos como válidos sobre los productos o servicios en una fase inicial. En futuras versiones se deberá permitir distinguir a la hora de generar un servicio o producto, generar categorías, agregar imágenes o establecer por defecto un producto o servicio siempre en la factura.

Al igual que sucede con los productos o servicios, se hace necesario la generación de compañías para que estas puedan ser utilizadas posteriormente en la factura de una forma cómoda y sencilla. En la versión inicial de GIC, se podrán almacenar el tipo, nombre, NIF o CIF, dirección, ciudad, código postal, provincia, correo y teléfono de las compañías. Por lo que una vez definido lo que entendemos como compañía en la versión inicial, se podrán realizar todas las operaciones CRUD que fuesen necesarias sobre las compañías.

Para poder generar facturas en la fase inicial y dejar un primer alcance completado de forma que se pueda utilizar la plataforma de forma real, necesitamos definir los datos del proveedor de la factura. Como es lógico, estos datos (nombre, dirección, ciudad, código postal, provincia, correo y teléfono) no es necesario generarlos en cada factura, sino que se generarán automáticamente en la factura a partir de los datos que se generen desde “Mi compañía”. Cabe mencionar que estos datos podrán ser modificados en cualquier momento que sea necesario para el usuario actualizar los datos de su compañía.

El último modelo que se generará en el alcance inicial será el de pagos. Es importante definir los pagos que los usuarios puedan tener un control simple de sus entradas y gastos. De los gastos bastará con definir el proveedor, título, descripción, fecha y total. Es por lo que, definidos los gastos, se desarrollarán todas las operaciones CRUD sobre los pagos.

Se generarán gráficas en la pantalla principal de la aplicación para que, mediante información gráfica y simple, los usuarios que usan el aplicativo puedan tener una comparativa de las facturas y sus estados o de las facturas frente a los pagos.

Como es comprensible, es necesario generar métodos de autenticación y almacenado de los datos que se puedan generar de las operaciones CRUD que se van a realizar. Para todo ello, en una fase inicial se hará uso de Firebase, con lo que se permitirá los siguientes puntos:

- Registrar nuevo usuario.
- Iniciar sesión.
- Recuperar contraseña.
- Realizar las operaciones CRUD de las facturas, compañías, productos o servicios y pago sobre el almacenamiento en tiempo real que provee Firebase.



Con todo lo mencionado en este apartado generado, GIC podrá ya ser útil para autónomos y pequeñas empresas. De esta forma, los usuarios ya pueden tener acceso a una aplicación en iOS o Android que les permita gestionar todos sus datos de forma simple y cómoda.

Es evidente, que este sólo es el primer paso de cara a generar de una plataforma web y una aplicación para los principales sistemas operativos móviles. Se ha decidido comenzar por la aplicación móvil y no por la plataforma web, dado que es importante dotar a los usuarios de un aplicativo que pueda ser utilizado desde cualquier sitio con acceso a redes móviles.

Para ir finalizando con la definición del alcance, se debe generar un logotipo para la aplicación que resuma y defina que aplicativo estamos utilizando. Además, es imprescindible generar un diseño personalizado para GIC que sea distintivo para que sea fácil diferenciarlo de la competencia y muestre una interfaz característica.

### 3.2 DISEÑO ARQUITECTÓNICO

Para el diseño arquitectónico del aplicativo, nos basaremos en el uso del modelo MVC (modelo, vista y controlador), dado que es el que Ionic define de base en gran medida por su uso de Angular.

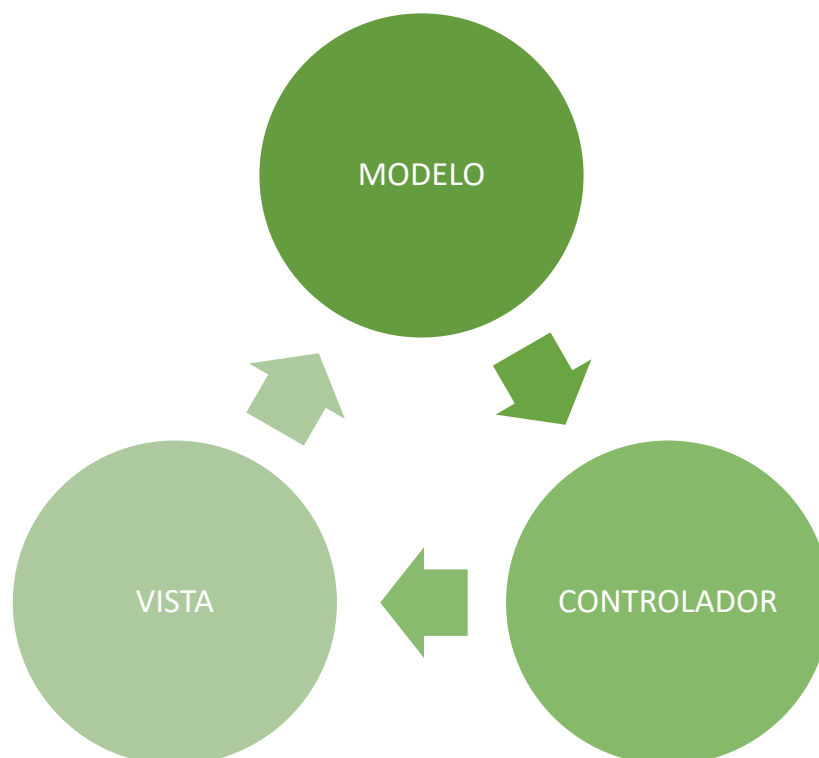


FIGURA 3.2.A – MODELO MVC (MODELO, VISTA Y CONTROLADOR)

El modelo-vista-controlador es un patrón de diseño que separa los datos, la lógica y las interfaces de usuario. A continuación, se detalla cada una de las capas por separado:



1. **Modelo:** Es la capa encargada de los datos y será la encargada de realizar toda la comunicación con los sistemas de bases de datos. En nuestro caso, el modelo serán servicios que crearemos para cada uno de los modelos de datos que disponemos (facturas, pagos, compañías y productos o servicios) y que serán los encargados de realizar las operaciones CRUD dentro de Firebase.
2. **Vista:** Es la capa que presenta a los usuarios los modelos anteriormente definidos ya procesados. Por tanto, dentro de Ionic podremos definir tanto la maquetación en HTML como el diseño de estilos en SCSS de cada una de nuestras páginas.
3. **Controlador:** Es la capa que se encuentra entre el modelo y la vista, y es la encargada de procesar los datos puros que llegan desde los servicios y convertirlos en datos procesados que puedan ser mostrados a los usuarios mediante las vistas.

### 3.3 MODELO DE LA BASE DE DATOS

Para el modelado de la base de datos, en la fase inicial y con vistas a la generación de un prototipo, lo recomendable será el uso de bases de datos jerárquicas. Las bases de datos jerárquicas tienen una forma similar a la de un árbol, por lo que un padre podrá tener muchos hijos, pero un hijo sólo podrá tener un padre.

Dado que en el prototipo sólo generaremos un usuario por compañía y bajo este vendrá toda la información que genere, utilizaremos los usuarios como nodos raíces y de ellos caerán todos los nodos hijos, de forma que:

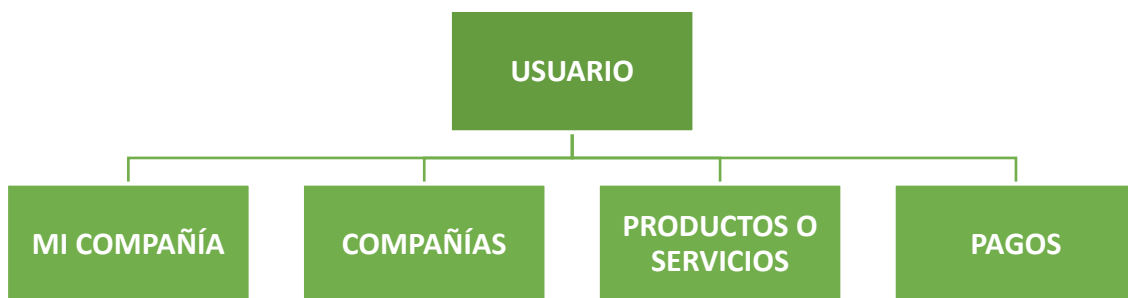


FIGURA 3.3.A – JERARQUÍA DE LAS ENTIDADES EN EL APLICATIVO

Por tanto, en un caso de una compañía real, podríamos tener un contenido en la base de datos similar a:

```
{
  "GwByoRQgzBhurqcvkHgv3yb0umf2" : {
    "companies" : {
      "1544972901936" : {
        "address" : "Calle Alberto, 16",
        "city" : "Telde",
        "email" : "company@correo.com",
        "id" : 1544972901936,
        "name" : "Company",
        "nif" : "12345678X",
        "phone" : "928112233",
        "postalCode" : "35214",
        "province" : "Las Palmas de Gran Canaria",
        "type" : "1"
      }
    },
    "concepts" : {
      "1544966894134" : {
        "description" : "Cuota por hora en servicios de administración",
        "id" : 1544966894134,
        "name" : "Hora de administración",
        "price" : "10.25",
        "tax" : 3
      },
      "1544967483492" : {
        "description" : "Cuota por hora en servicios de contabilidad",
        "id" : 1544967483492,
        "name" : "Hora de contabilidad",
        "price" : "14.65",
        "tax" : 3
      }
    }
  }
}
```





```
    }  
  },  
  "invoices" : {  
    "1545297794159" : {  
      "concepts" : [ {  
        "concept" : {  
          "description" : "Cuouta por hora en servicios de contabilidad",  
          "id" : 1544967483492,  
          "name" : "Hora de contabilidad",  
          "price" : "14.65",  
          "tax" : 3  
        },  
        "discount" : 0,  
        "price" : "14.65",  
        "quantity" : "4",  
        "tax" : 3,  
        "total" : 62.76  
      } ],  
      "created" : 1545297794159,  
      "customer" : {  
        "address" : "Calle Alberto, 16",  
        "city" : "Telde",  
        "email" : "company@correo.com",  
        "id" : 1544972901936,  
        "name" : "Company",  
        "nif" : "12345678X",  
        "phone" : "928112233",  
        "postalCode" : "35214",  
        "province" : "Las Palmas de Gran Canaria",  
        "type" : "1"  
      },  
      "customerName" : "Company",
```



```
    "date" : "2018-12-20",
    "id" : 1545297794159,
    "invoice" : "201218A",
    "state" : "aceptada",
    "total" : "53.25",
    "withholding" : 1
  }
},
"mycompany" : {
  "address" : "Avenida Almendros, 17",
  "city" : "Vega de San Mateo",
  "email" : "enforma@correo.com",
  "name" : "En forma",
  "nif" : "87654321X",
  "phone" : "928332211",
  "postalCode" : "35320",
  "province" : "Las Palmas de Gran Canaria"
},
"payments" : {
  "1545310999412" : {
    "date" : "2018-12-20",
    "description" : "Cuota mensual por ADSL y teléfono fijo.",
    "id" : 1545310999412,
    "name" : "Cuota internet",
    "total" : "98.75"
  }
}
}
```

FIGURA 3.3.B – JSON DE EJEMPLO DE LOS DATOS ALMACENADOS EN FIREBASE

El ejemplo anterior, muestra un caso real de datos almacenados en Firebase siguiendo la estructura jerárquica que se ha definido. Como se ve, sigue la estructura de JSON



(JavaScript Object Notation) en la que se definen los nodos por la dupla clave-valor y los nodos siguen la estructura ordenada de un árbol simple.

Además, un punto importante es que los datos no están relacionados entre sí, no existe todavía esa relación que si se da entre las bases de datos relacionales. Por ejemplo, una factura no tiene relación con las compañías o con los productos y servicios, sino que simplemente almacena una copia de estos, de forma que, se podría modificar un cliente y producto y en la factura que ya se ha generado estos datos aparecerán inmutables y tal como se definieron en el momento de generar la factura.

Como es lógico, a medida que GIC crezca y pase la fase de prototipo, será ideal cambiar este modelo de bases de datos a una relacional, en la que se pueden relacionar las entidades entre sí sin tener que recorrer toda la base de datos. Por ejemplo, desde que en GIC se permita el inicio de sesión de varios usuarios de la misma compañía, seguir con un modelo jerárquico hace que el acceso a los datos en base de datos sea complejo y lento, problema que con una base de datos relacional sería solucionado instantáneamente y ayudaría a escalar mejor al aplicativo.

El modelo entidad relación define las entidades de datos y sus interrelaciones y propiedades. En GIC, se va a trabajar con las siguientes entidades de datos: Compañías, artículos o servicios, pagos, presupuestos y facturas. A continuación, se pueden ver dichas entidades y todas sus propiedades.

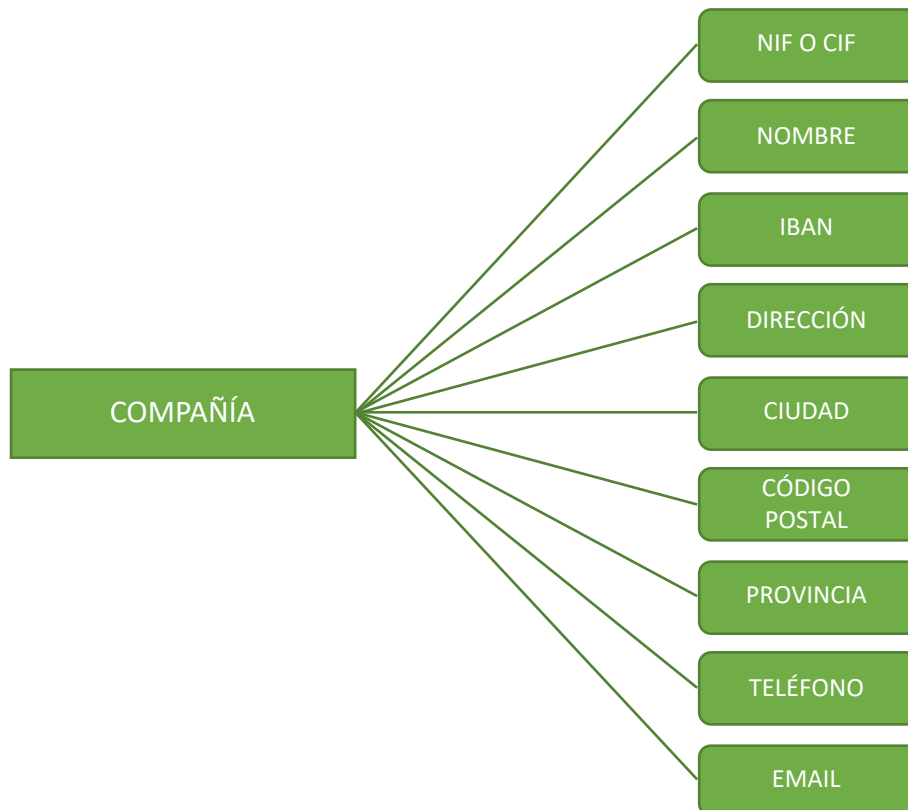


FIGURA 3.3.C – MODELO ER DE LA ENTIDAD COMPAÑÍA

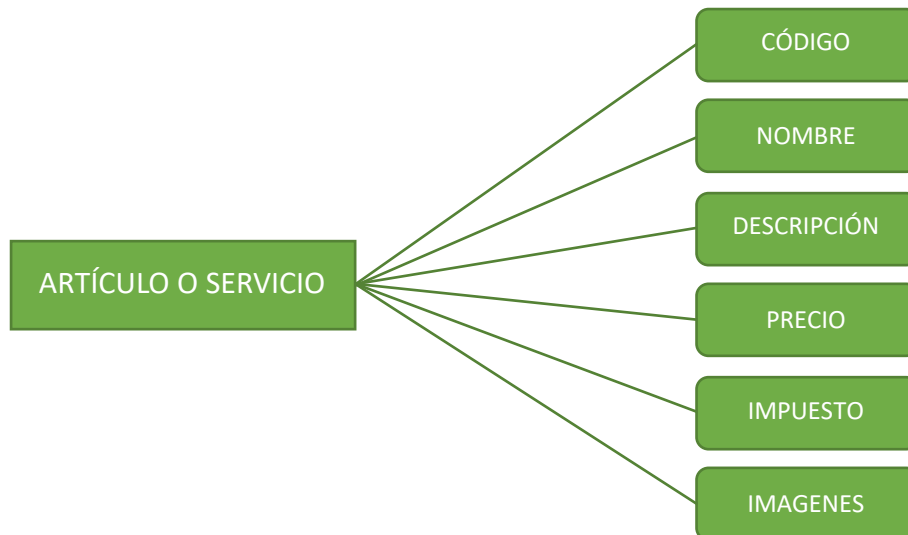


FIGURA 3.3.D – MODELO ER DE LA ENTIDAD ARTÍCULO O SERVICIO

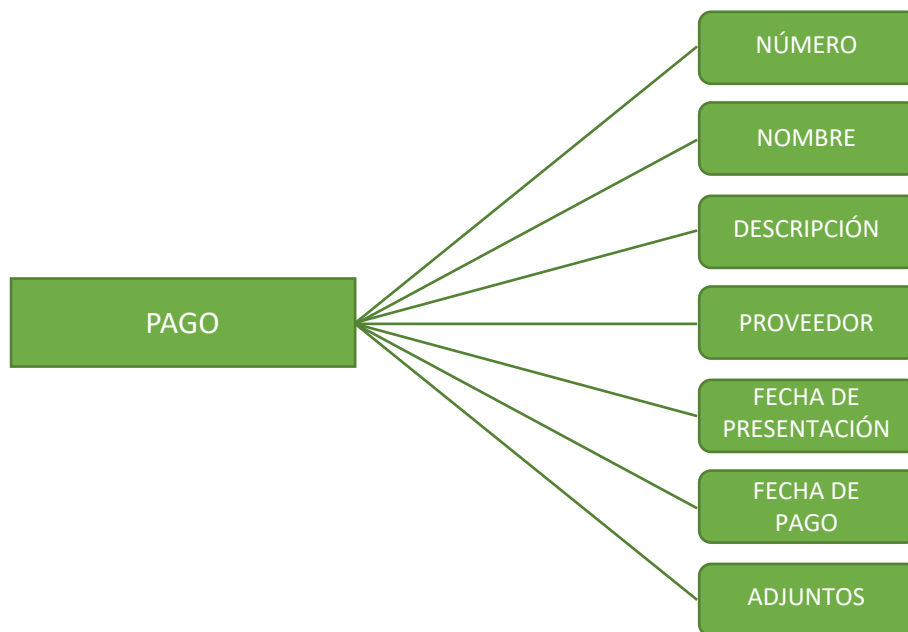


FIGURA 3.3.E – MODELO ER DE LA ENTIDAD PAGO

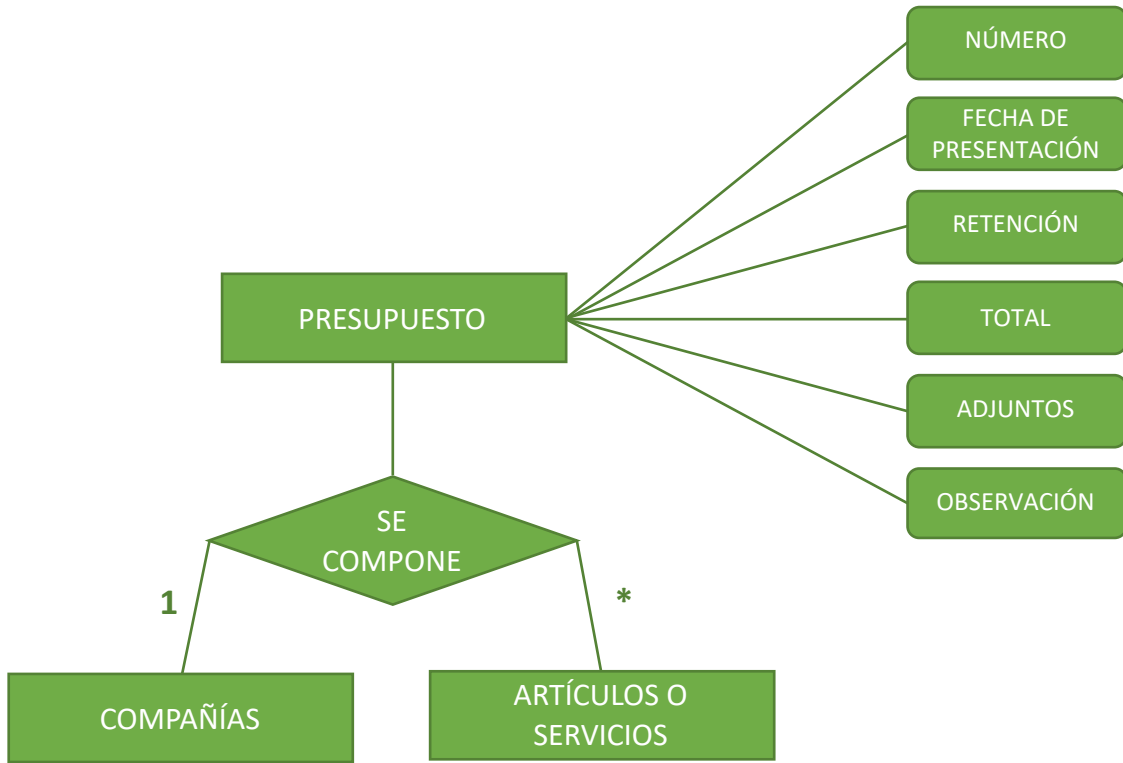


FIGURA 3.3.F – MODELO ER DE LA ENTIDAD PRESUPUESTO

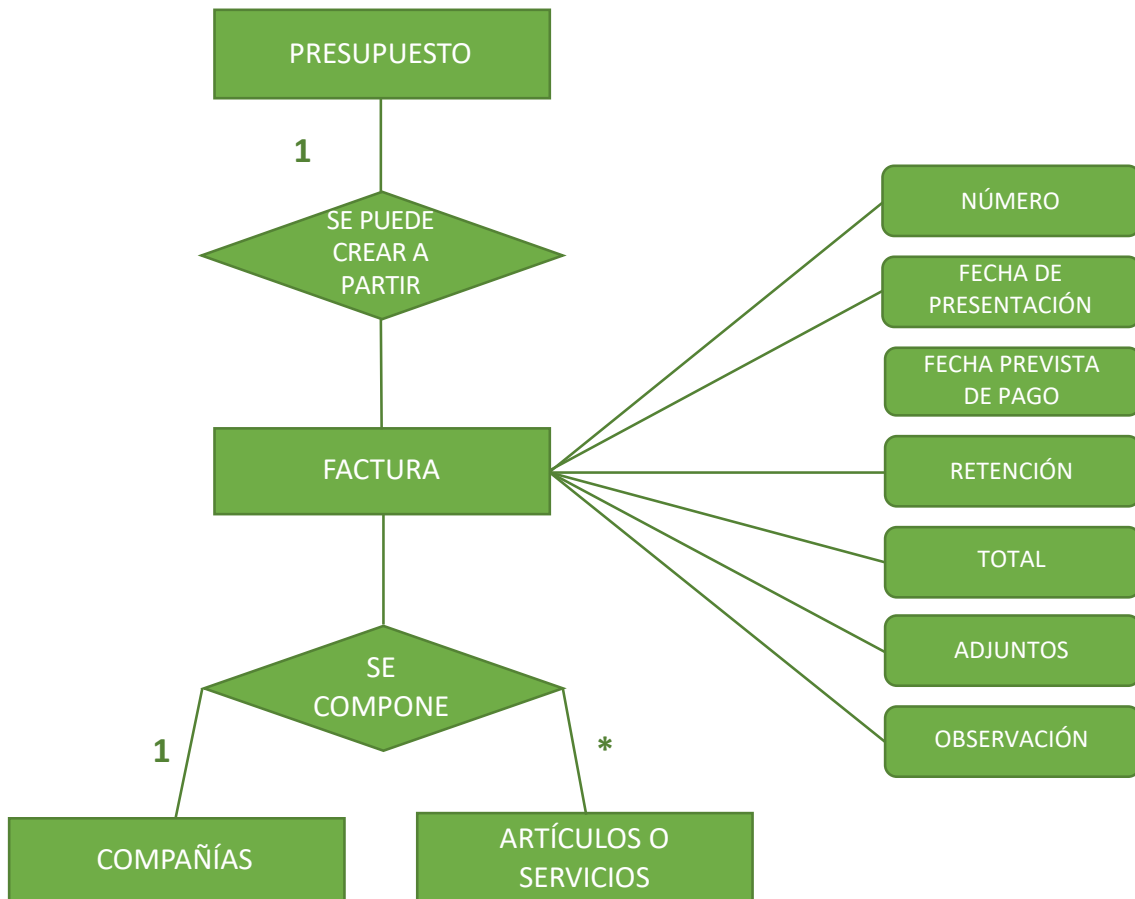


FIGURA 3.3.G – MODELO ER DE LA ENTIDAD FACTURA

Como es visible, salvo los presupuestos (Figura 3.3.F) y las facturas (Figura 3.3.G), todas las demás entidades sólo están formadas por propiedades. Sin embargo, estas dos están formadas por propiedades y a su vez tienen una compañía y tantos artículos o servicios como fuesen necesarios. Además, una factura puede ser generada a partir del contenido de un presupuesto.

Una vez definidas las entidades, vamos a comprobar las interrelaciones existentes entre los usuarios y dichas entidades. Por tanto, a continuación, se muestran los diagramas entidad relación existentes al realizar las operaciones CRUD entre las entidades y los usuarios:

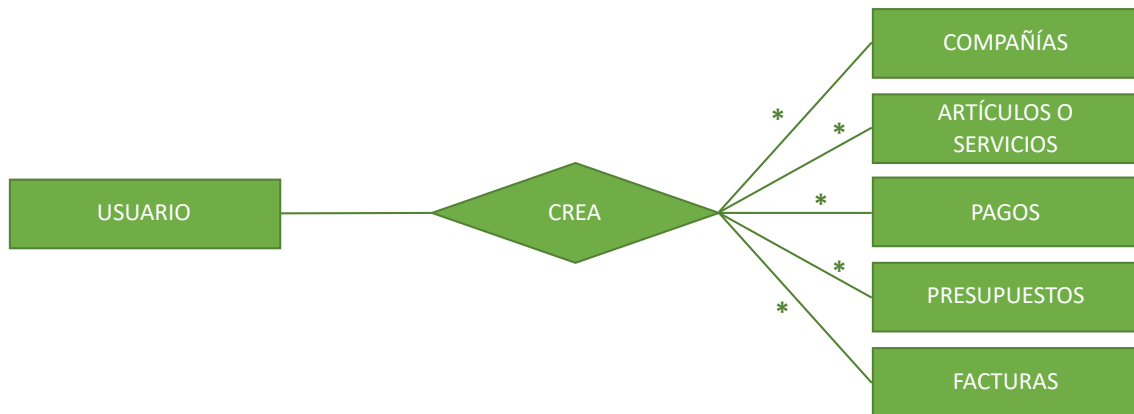


FIGURA 3.3.H – MODELO ER DE LAS OPERACIONES DE CREACIÓN EN GIC

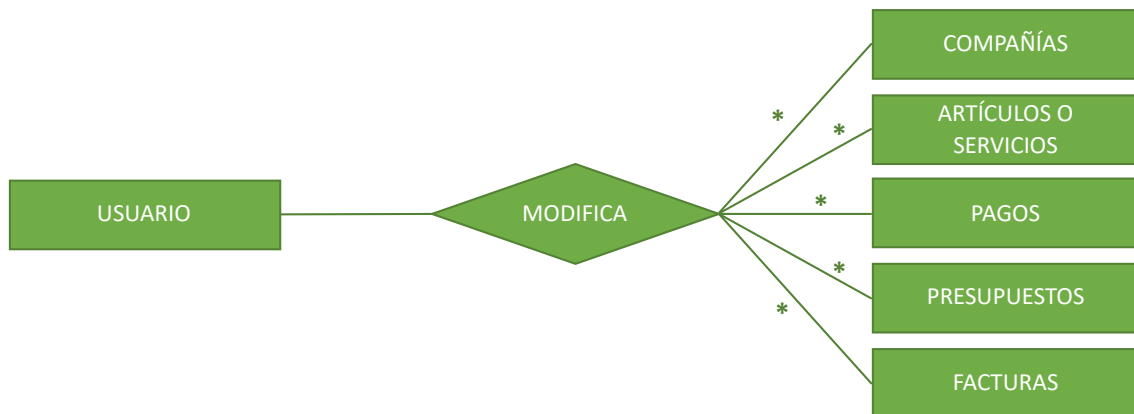


FIGURA 3.3.I – MODELO ER DE LAS OPERACIONES DE MODIFICACIÓN EN GIC

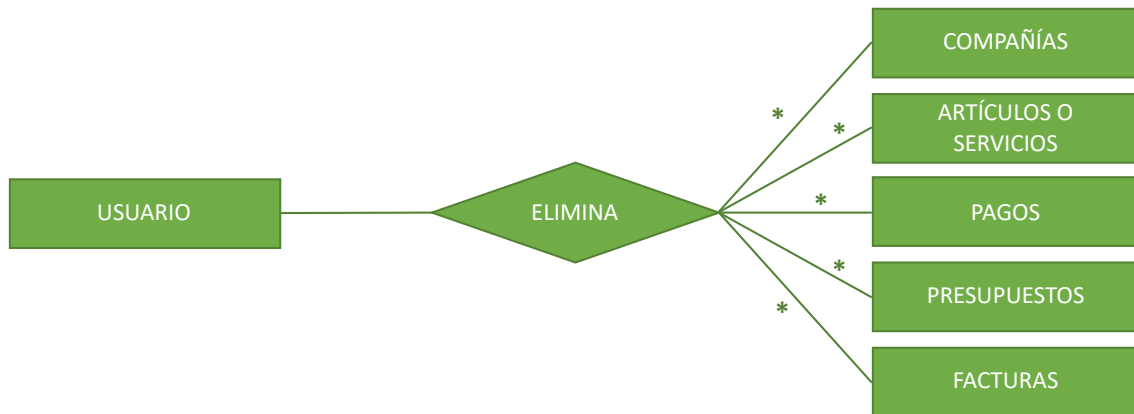


FIGURA 3.3.J – MODELO ER DE LAS OPERACIONES DE ELIMINACIÓN EN GIC

Además, de las operaciones CRUD, los usuarios podrán realizar otro tipo de acciones que se definen en los siguientes modelos:

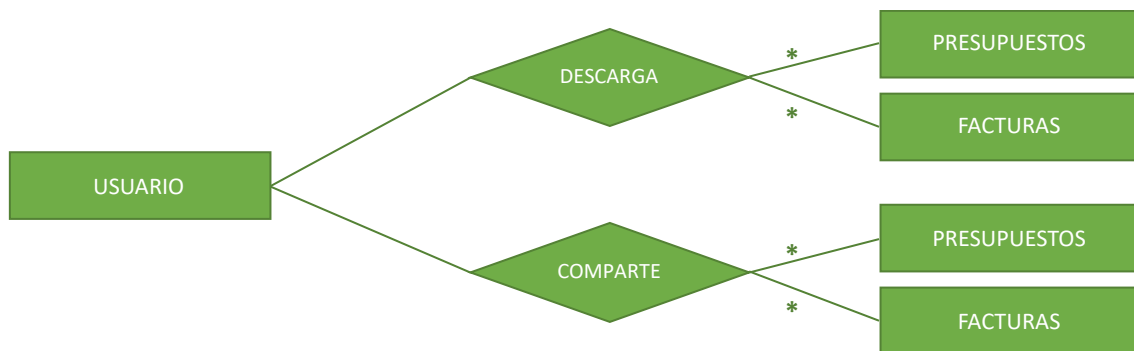


FIGURA 3.3.K – MODELO ER DE LAS OPERACIONES DE DESCARGAR Y COMPARTIR

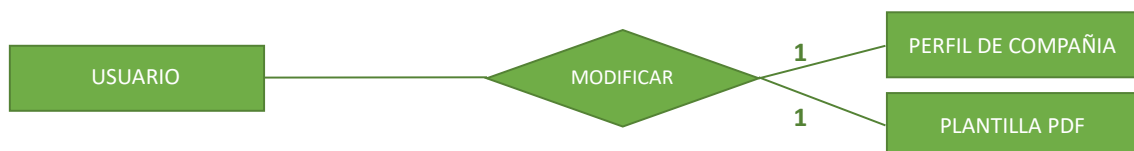


FIGURA 3.3.L – MODELO ER DE LAS OPERACIONES DE MODIFICAR PERFIL Y PLANTILLA

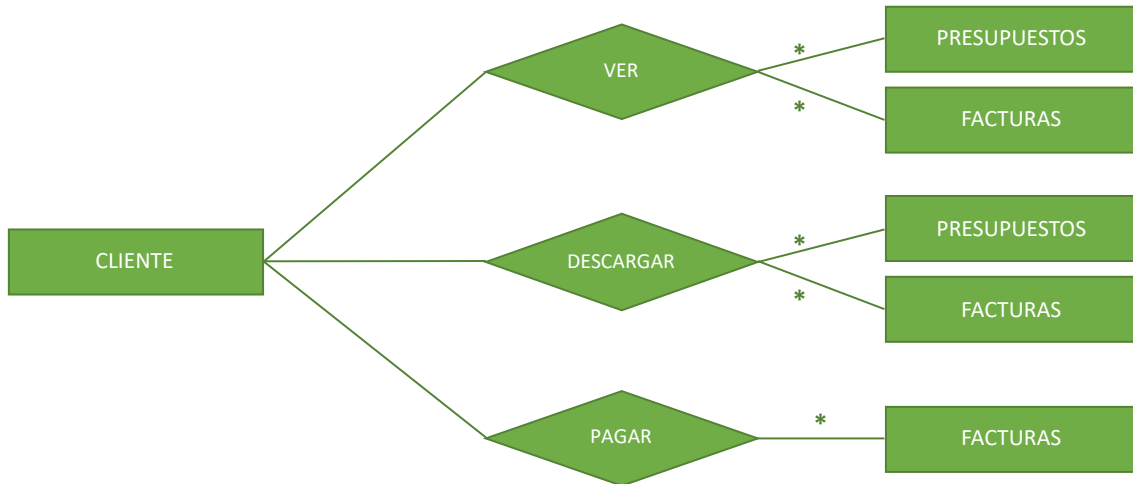


FIGURA 3.3.M – MODELO ER DE LAS OPERACIONES DE VISUALIZACIÓN, DESCARGA Y PAGO

De forma que con los modelos entidad relación mostrados, hemos definido las principales acciones que podrán llevar a cabo los usuarios y las principales propiedades que definen a las entidades que existirán dentro de GIC.

### 3.4 TECNOLOGÍAS

Para el desarrollo del prototipo y establecer las bases de cara a futuros desarrollos incrementales del aplicativo, se han utilizado las siguientes tecnologías:

#### 3.4.1 IONIC



Para el desarrollo de la aplicación de GIC, se podría haber generado una aplicación nativa por separado para cada uno de los sistemas operativos móviles. Aunque supone que estamos sacándole el máximo partido a cada uno de los sistemas operativos, también supone unos costes económicos y de tiempos importantes de cara al desarrollo en un entorno real. Para evitar este problema y poder tener el fin, que no es otro que tener el aplicativo en el máximo posible de sistemas operativos, nos hemos decantado por la opción de desarrollar el aplicativo bajo Ionic.

Ionic es un framework pensando para el desarrollo de aplicaciones híbridas, pero que también está diseñado para generar páginas webs e incluso aplicaciones de escritorio. Ionic hace uso de Angular, lo que permite la generación de aplicaciones móviles haciendo uso de tecnologías webs. Esto quiere decir que, por ejemplo, para la generación





de las vistas o páginas, haremos uso de tecnologías básicas de diseño como web como pueden ser HTML, CSS y JavaScript. Esto permite que el desarrollo de código no sea complejo y es amigable para los desarrolladores que se dedican a la rama del desarrollo web. Además, abre una vía cómoda para desarrollar en paralelo tanto las aplicaciones móviles como la web.

Una de las grandes ventajas de las que goza Ionic es que permite el uso de componentes ya diseñados y construidos para su uso inmediato. Estos componentes normalmente son componentes básicos como puede ser un menú de navegación. Los componentes están diseñados de forma que, si son usados durante el desarrollo, la interfaz que utilizarán será la marcada por el sistema operativo. Es decir, si utilizamos el componente Ion-List, por ejemplo, su vista y estilo cambiará dado que utilizará los patrones de diseño de Material Design si es en Android, el marcado por Apple si se trata de iOS o el definido por Microsoft para Windows Phone.

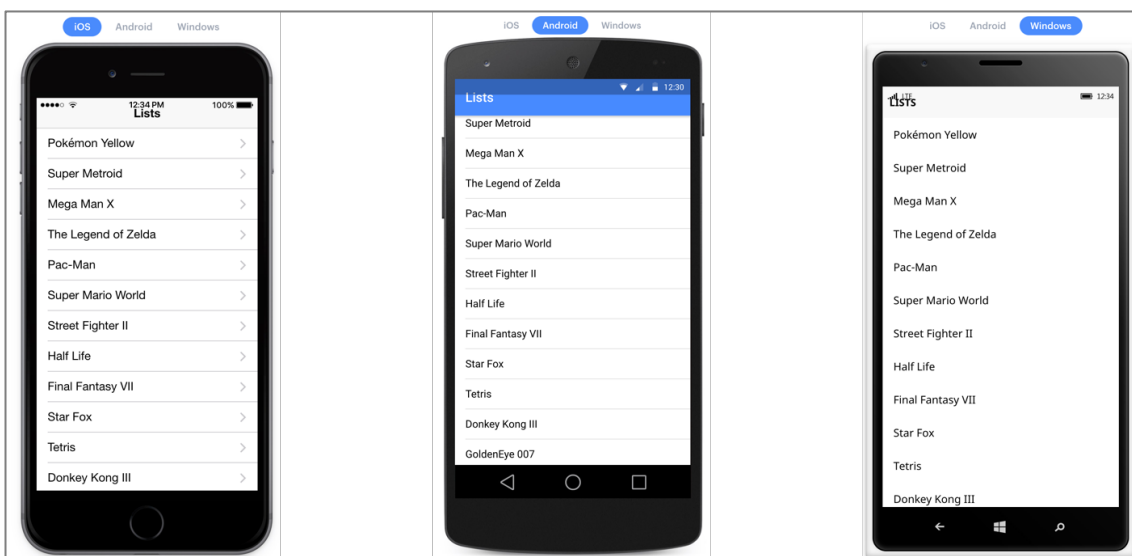


FIGURA 3.4.1.a – VISTA DEL COMPONENTE ION LIST

Los componentes de Ionic son muy fáciles de usar y la documentación asociada a ellos es extensa y completa. Esto apunta a que es fácil comprobar si existe el componente necesario para alguna acción en concreto o es necesario el desarrollo de dicho componente. En el caso de ser necesario un desarrollo complementario, al estar usando Angular como base, en Ionic se podrá generar de forma cómoda cualquier componente complementario.

En conclusión, Ionic es un framework que nos permitirá la generación de aplicaciones híbridas sin tener en cuenta hacia qué sistema operativo está pensada la aplicación y haciendo uso de tecnologías web ya conocidas. Esto unido a la cantidad de componentes existentes dentro de Ionic y cómo estos se adaptan al sistema operativo al dispositivo que esté usando el usuario, hacen que Ionic se muestre como una tecnología imprescindible en casos como el que nos ocupa en donde se desea generar tanto una aplicación móvil multiplataforma y otra versión web para los usuarios de escritorio.

### 3.4.2 APACHE CORDOVA



Gracias a Ionic se puede generar todos las páginas y componentes necesarios para la visualización y lógica de una aplicación híbrida. Sin embargo, existen acciones como el acceso a los ficheros, la cámara o GPS del dispositivo que son necesarios en una aplicación móvil.

Para el acceso al dispositivo a los componentes del dispositivo desde el que se está haciendo uso del aplicativo, Ionic hace uso de Cordova. Cordova es un conjunto de librerías y plugins que permiten el uso de los componentes nativos del dispositivo. Así mismo, Cordova también es la tecnología que nos permitirá generar la versión móvil en los diferentes sistemas operativos del desarrollo realizado con Ionic y con tecnologías webs.

### 3.4.3 ANGULAR



Angular es un framework para JavaScript desarrollado por Google. Angular es fundamental en nuestro caso dado que estamos haciendo uso de Ionic y este lo usa como base. Angular permite la generación de contenido web de forma simple gracias a las herramientas que aporta.

Un componente web construido con Angular funciona en modo reactivo, es decir, no es necesario recargar la página para que este realice las acciones necesarias, incluso las posibles transacciones que desee realizar entre el Front-end y el Back-end del aplicativo, dado que todo es dinámico y dichas conexiones se podrían realizar mediante AJAX. Por tanto, Angular permite generar aplicaciones web de una sola página y asimismo permite hacer uso del modelo MVC (modelo, vista y controlador) en las aplicaciones web.

Otra de las grandes ventajas de Angular es que nos permite generar componentes de forma simple y sencilla, e incluso gracias a su enrutamiento, aunque estemos en una sola página, podremos dar la sensación de aplicativo web con varias rutas sin tenerlas realmente conectadas a un controlador que nos muestre la vista que corresponda.

Por todo esto, Angular se convierte en una de las piedras fundamentales de este proyecto, dado que no sólo sus ventajas hacen que sea una herramienta potente, sino que



al estar integrado completamente dentro de Ionic, hará que las aplicaciones y plataformas web que se generen con ellas sea completas, funcionales y simples de construir.

#### 3.4.4 TYPESCRIPT

# TypeScript

Otra de las tecnologías importante de las que vamos a hacer uso es TypeScript. TypeScript es un superset de JavaScript, es decir, aunque programemos en TypeScript el código que se generará posteriormente será JavaScript y los dispositivos no notarán que se trata de código TypeScript, sino que entenderán que todo ha sido construido con JavaScript.

TypeScript surgió de la necesidad que aparece cuando se desean realizar aplicativos grandes, robustos y escalables. Por tanto, TypeScript implementa herramientas en su lenguaje para que puedan cumplir esta meta.

Algunas de las ventajas de TypeScript es que, al ser un lenguaje compilado, nos podremos dar cuenta de errores antes de su ejecución, hace uso de un tipado estático, está orientado a objetos, permitir aplicaciones también del lado del servidor o la flexibilidad que ofrece si en ciertas ocasiones nos deseamos acercar más al JavaScript más puro.

En el caso que nos ocupa, es importante el uso de TypeScript dado que Angular está fabricado con él y la lógica de las páginas y los servicios que generemos en Ionic están contruidos también bajo Ionic.

#### 3.4.5 SASS



Las hojas de estilo CSS aparecieron como la forma en la que aplicar estilo a los archivos HTML, de forma que se pudiesen separar de las etiquetas HTML y permitir un desarrollo más cómodo tanto del código HTML como del estilo que le dará forma. Sin embargo, con el paso del tiempo, la generación de los estilos en CSS se ha convertido en algo demasiado engorroso de cara al desarrollo, sobre todo de cara a aquellas aplicaciones o plataformas en las que se desea tener un estilo personalizado y diferenciador.

SASS ofrece grandes ventajas que no ofrece CSS, como puede ser la posibilidad de agregar una variable y hacer que el estilo del aplicativo o plataforma sea dinámico en relación con dichas variables. Este simple punto, permite la generación de paletas de colores simples que los usuarios puedan ir variando en el aplicativo en relación con su gusto.

Otro de los factores importantes de SASS son los mixins y la herencia, que permite que no repitamos código de estilo innecesariamente. Los mixins son el elemento perfecto para la generación de código CSS3, dado que tenemos que añadir en ciertas ocasiones los prefijos de los navegadores en los que trabajamos. De forma que podemos generar un mixin que resuelva esto de forma cómoda y que su resultado cumpla las condiciones que establece CSS. Además, la herencia que permite SASS hace que podamos extender etiquetas, clases e identificadores de forma cómoda y simple sin tener que replicar código dentro de nuestras hojas de estilo

Por último, otra de las grandes ventajas de SASS es que nos libra del problema de desarrollar en CSS con su indentación y hacer uso de anidación, permitiendo desarrollar un código más simple y legible.

Por todos los motivos detallados, la utilización de SASS en el proyecto se hace de vital importancia, dado que nos permite un desarrollo más simple y estructurado, además de dar al aplicativo de una interfaz personalizada y diferenciadora.

#### 3.4.6 FIREBASE



Para los procesos de autenticación y almacenado de datos se ha hecho uso de la plataforma Firebase. Firebase es una plataforma de desarrollo en la nube para plataformas webs y aplicaciones móviles, que provee de una API para realizar procesos de autenticación, almacenado de datos o procesos de analíticas entre otros.

En nuestro caso, haremos de uso de Firebase para realizar los procesos de autenticación y almacenado de datos en tiempo real. Para poder hacer uso de los servicios que Firebase provee, es necesario generar una cuenta de desarrollador y dentro de esta se podrán generar todos los proyectos necesarios.

Al generar un proyecto dentro de Firebase, ya es posible hacer uso de su API para poder realizar los procesos de autenticación o almacenado de datos. Firebase permite una rápida y simple conexión, y como punto a favor, ofrece una documentación completa de sus funciones y cómo conectar Firebase con las diversas opciones de desarrollo existentes.

Otro de los puntos fuertes de Firebase es la existencia de una consola en la que se puede visualizar y configurar los diversos productos que ofrece. Por lo que parte de la configuración del aplicativo se realizará desde dicha consola.



Cabe mencionar que Firebase es escalable en cuanto a precios y que se podrán seleccionar planes en dependencia de los usuarios y necesidades que vaya teniendo el aplicativo a lo largo del tiempo. Esto permite que incluso desde el prototipo se pueda hacer uso de Firebase con su plan gratuito y poder realizar un prototipo completo sin necesidad de tener costes adicionales.

### 3.4.7 CHARTJS



Una de las partes más importantes del aplicativo será la de mostrar al usuario los datos procesados de cómo va su compañía. Es importante destacar que esos datos no pueden estar presentados como tablas, dado que su lectura y comprensión podría llevar tiempo y no sería un enfoque adecuado para la plataforma. Lo ideal en estos casos es mostrar una versión simplificada de los datos de esas tablas en forma de gráfica.

En GIC se ha hecho uso de Chart.js, que es una librería en JavaScript que hace uso de HTML5 y Canvas para la generación de gráficas en base a los datos que se deseen. Una de las ventajas del uso de esta librería reside en que las gráficas que se muestran son dinámicas y dado que en el prototipo y en la versión final de GIC los datos se almacenan y muestran en tiempo real, si un usuario está visualizando la página del Home y se generase el pago de una factura, la gráfica correspondiente a los estados de las facturas se actualizaría de forma automática mostrando en tiempo real los datos.

### 3.4.8 PDFMAKE



Es vital dotar al aplicativo de la generación de archivos PDF. Como norma, los ficheros en PDF se suelen generar desde el Back-end o desde una API construida en PHP o Java, entre otros lenguajes. Para evitar el uso de este tipo de lenguajes, se ha optado por hacer uso de PdfMake, se trata de librería en JavaScript que permite la generación de ficheros PDF completamente en JavaScript.

Desde PdfMake se pueden realizar configurar por completo el PDF que se pretende generar, desde generar texto, tablas, columnas hasta insertar imágenes y vectores o

modificar los encabezados y los pies de las páginas, pudiendo modificar el estilo en cualquiera de los elementos que se inserten.

Su uso es simple y aunque es laborioso generar todo el contenido de una factura dado que el contenido de una factura es grande y la generación en JavaScript puede llevar un tiempo dado que línea a línea se debe generar el elemento y sus estilos, PdfMake permite generar el PDF en un tiempo reducido y con un resultado de buena calidad.

### 3.5 PRUEBAS Y DESPLIEGUE

Para la realización de las pruebas pertinentes del prototipo se ha hecho uso de un componente importante de Ionic, Ionic Lab. Este componente permite compilar y ejecutar un servidor desde consola para visualizar y ejecutar el código desarrollado.

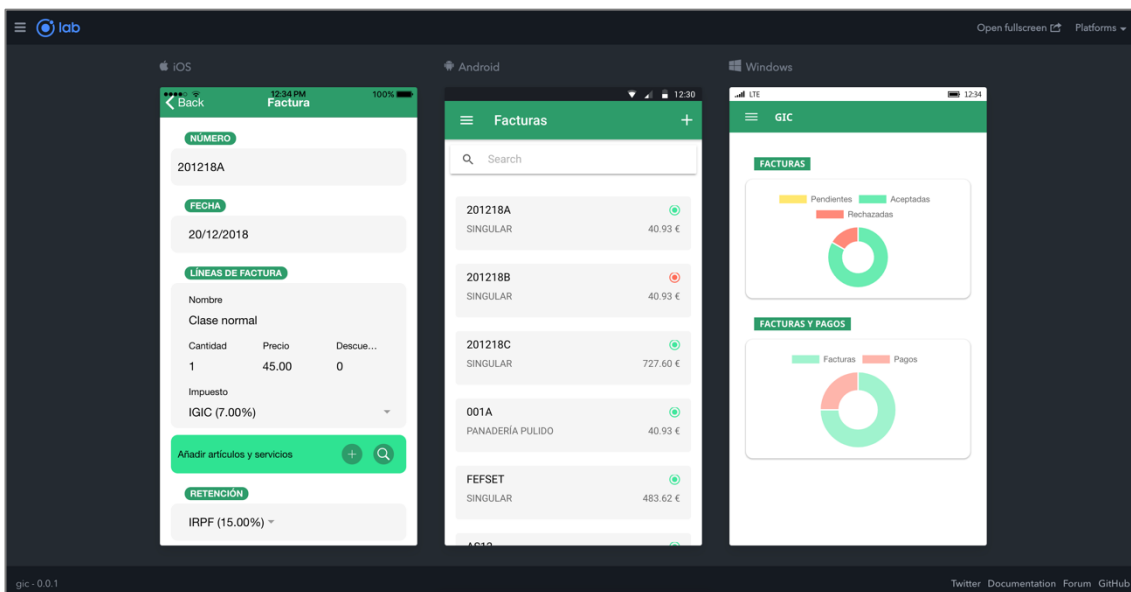


FIGURA 3.5.A – VISUALIZACIÓN EN IONIC LAB DE GIC

Una de las grandes ventajas del Ionic Lab reside en poder visualizar como se verá el aplicativo en todos los sistemas operativos. Además, permite interactuar con el aplicativo como la haría un usuario desde su dispositivo móvil.

Además, también Ionic permite poder compilar y ejecutar un servidor dentro del propio dispositivo. Para esto basta con habilitar las opciones de desarrollador del móvil Android en el que se desea ejecutar la prueba y lanzar el comando. Para iOS funciona de manera similar, pero para poder tener una cuenta de desarrollador habrá que realizar un desembolso de 99\$ por año. Es por ello por lo que para el prototipo se han realizado el testeo del aplicativo desde Ionic Lab.

Para testear y realizar las pruebas necesarias, se han generado datos de prueba de todas las entidades existentes en el aplicativo. De forma que se han realizado todas las operaciones CRUD sobre facturas, pagos, compañías y productos o servicios. Además, se ha comprobado que los filtros y búsquedas devuelven los resultados previstos.

Otras pruebas realizadas han sido la búsqueda de los errores del aplicativo al realizar las operaciones CRUD, como pueden ser intentar la creación de entidades sin rellenar los



formularios correctamente. Estas pruebas son necesarias dado que, en entornos reales, los usuarios al usar el aplicativo no rellenan los formularios correctamente y es necesario que se compruebe que el aplicativo responde ante este tipo de casos.

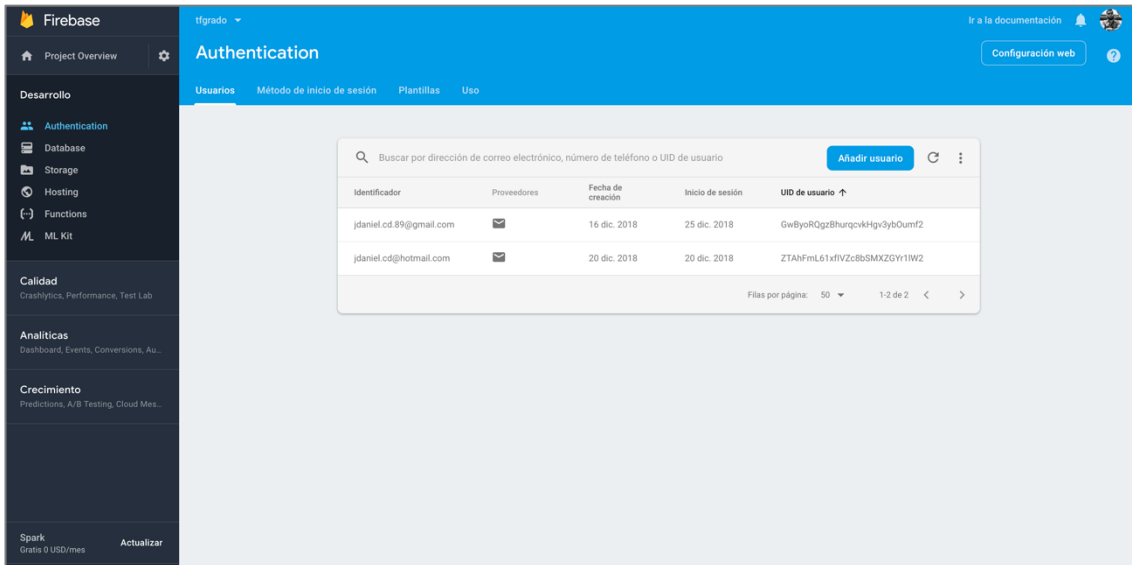


FIGURA 3.5.B – VISTA DE LA CONSOLA DE FIREBASE CON LA AUTENTICACIÓN

Otro punto importante del testeo es el de comprobar que los usuarios se han registrado correctamente o que los datos se han almacenado correctamente en la nube. Para ello, Firebase nos provee de una consola a la que podemos acceder y ver todos los datos desde allí. Incluso desde la consola de Firebase se pueden realizar acciones como crear usuarios o modificar los datos en la base de datos, para poder comprobar luego que el aplicativo está actualizando en tiempo real los datos o otras acciones que sean necesarias comprobar.

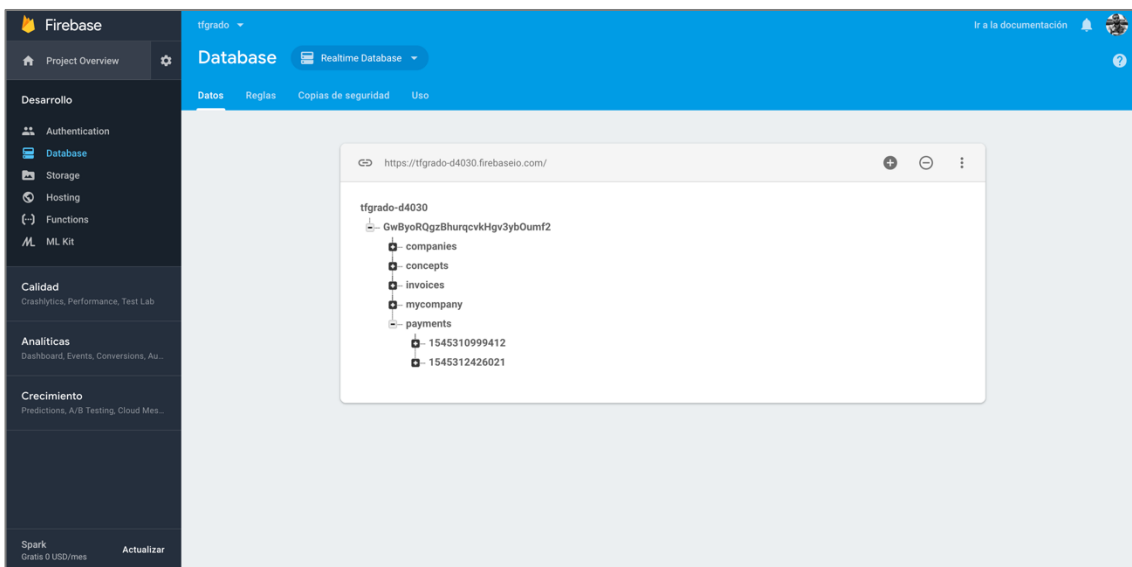


FIGURA 3.5.C – VISTA DE LA CONSOLA DE FIREBASE CON LA BASE DE DATOS

También es importante comentar que desde la consola se realizará toda la configuración necesaria para el uso de los productos que se estén usando de Firebase. Por ejemplo, si estamos trabajando con la autenticación, desde la consola podremos activar o desactivar los métodos de acceso que permitimos dentro del aplicativo o las plantillas que se enviarán a los correos electrónicos de los usuarios en el caso de activar la opción de recuperación de la contraseña.

Para finalizar las pruebas y comenzar con el despliegue, se ha generado una APK para Android de forma que se pueda testear el aplicativo en móviles Android y se pueda comenzar con la fase de despliegue.

En una fase inicial de GIC y con el prototipo finalizado. Es necesario definir que despliegue se va a realizar, aunque esté fuera del alcance del prototipo y de este trabajo. Lo ideal de cara a un despliegue correcto del aplicativo, será necesario generar el prototipo como aplicación tanto para Android como para iOS, de forma que se pueda enviar dichas aplicaciones a usuarios reales. Estos usuarios reales estarán controlados y formarán parte de la beta del aplicativo.

Es necesario que los usuarios que van a hacer uso del aplicativo en su fase beta estén en contacto con los desarrolladores para que se pueda comprobar que la interacción con la aplicación está siendo correcta y que el objetivo del aplicativo se está cumpliendo en todo momento. También es importante que en esta fase beta se anule la posibilidad de registro dentro del aplicativo y que el registro se lleve directamente desde Firebase por parte del equipo de desarrollo, para poder tener controlados a los usuarios que hacen uso del aplicativo y evitar posibles accesos indeseados a GIC.

Una vez evaluada la situación de la beta y comprobar que el aplicativo está solucionando los problemas reales de los usuarios, se podría comenzar con la subida del aplicativo a las principales tiendas de los sistemas operativos: App Store y Google Play. Es importante destacar que, aunque se suba a las tiendas y se permita su descarga, no se trata de un producto real, sino un prototipo y esto se deberá informar a los usuarios de ello. Es en este punto cuando se volverá a activar el registro y se permitirá el uso del prototipo con todas sus funcionalidades. Esto servirá para poder comprobar la acogida que tiene el aplicativo y la respuesta que tienen ante el producto los usuarios finales que estén haciendo uso del aplicativo.

Llegados a este punto, es importante tener muy en cuenta el feedback que recibamos parte de los usuarios que estén haciendo uso de la beta y corregir errores además de realizar las implementaciones que fuesen necesarias por nuevas el feedback recibido. Es vital responder y actuar rápido frente a los problemas que se les presenten a los usuarios que hagan uso de la beta para que no pierdan la esperanza en la plataforma y acudan a la competencia para buscar mejores resultados.

Una vez se considere que la beta está finalizada, se podrá comenzar a monetizar el aplicativo e iniciar con el modelo de negocio que se ha definido en puntos anteriores. Nunca se deberá comenzar con el modelo de negocio y la monetización de este sin haber pasado por una fase de testeo y otra de producto en estado beta, dado que es evidente que al inicio del aplicativo no está todo funcionando como debería ni las funcionalidades que se han generado son todas las que el usuario final necesita y que por ello poner en el mercado un aplicativo sin la calidad necesaria implicaría que el fracaso del producto.





## 4 CONCLUSIONES

Como conclusión para esta memoria, se puede dar cumplido el objetivo del prototipo. La existencia del prototipo prueba la mayoría de los objetivos con los que se ha realizado ya que muestra el gran potencial al que puede llegar GIC si se mantiene esta línea de desarrollo.

GIC en este estado ha hecho valor sus puntos fuertes frente a las alternativas del mercado. Ya como prototipo muestra que cumple la mayoría de los requisitos que son necesarios para autónomos y pequeñas empresas, además, de mostrar una interfaz propia y con un diseño que permite al usuario distinguirla de las demás opciones de mercado. La paleta de colores seleccionada permite que todo el aplicativo mantenga una interfaz acorde y que todos sus elementos tengan características entre ellos que permitan seguir una concordancia entre un elemento y otro.

Un punto para destacar dentro del prototipo es su facilidad de uso. No es necesario disponer de grandes manuales ni de un punto de información al que acceder para comenzar a utilizar el aplicativo. Nada más acceder al aplicativo, cualquier usuario detecta cuáles son los pasos que debe realizar para generar compañías, artículos o servicios, pagos o facturas. Este punto es vital, dado que demuestra que en el diseño del aplicativo ha sido realizado correctamente y que permiten que el uso diario del aplicativo sea simple y cómodo.

El prototipo realizado, ya se puede comenzar a utilizar como fase beta del proyecto. Esto permitirá no sólo testear el producto con los desarrolladores, sino encontrar clientes reales a los que comenzar a dar la beta y que puedan testear el producto. Esto permitirá evaluar si el análisis de requisitos realizado correctamente y si tras el feedback de los usuarios, ver si es necesario reformular ciertos requisitos. Esto es vital, dado que todavía nos encontramos en la fase del prototipo y será más fácil realizar cambios en esta fase que esperar a un producto ya en producción. También esta fase, nos permitirá comprobar si el volumen de datos volcado por los usuarios de sus empresas y compañías en la plataforma es muy grande y si es necesario cambiar hacia otros sistemas de almacenamiento o si es viable seguir haciendo uso de Firebase.

Tras una fase de testeo por parte compañías reales, será necesario definir si es necesario la generación de una API que controle desde Back-end todo el aplicativo. En estos momentos, no es necesario para el prototipo. Sin embargo, en fases futuras habría que evaluar si conviniese realizar una API robusta que de todas las respuestas a los frontales existentes. Esto también abrirá otras vías de cara a que otras compañías puedan conectarse a consumir y difundir información a través de nuestra API. Esto podría ser un gran avance para el proyecto, dado que la comunicación con terceros será vital si el proyecto crece y expande sus posibilidades.

GIC como producto tiene un gran potencial. Es evidente pensar que existen muchas opciones en el mercado y que es difícil conseguir que empresas y usuarios que ya están haciendo uso de otras plataformas, cambien la que estén usando por GIC. Sin embargo, GIC se ha desarrollado pensando en el usuario y como este interactúa en su día a día



con la gestión de sus facturas y demás partes de su negocio. Esto quiere decir que GIC debe mantener este punto de vista a la hora de seguir expandiendo sus posibilidades, para que nunca pierda el valor de ser una aplicación sencilla y útil para el usuario, y por la que paga un precio razonable y que se ajusta a sus posibilidades.

En un futuro, otro punto de vista a revisar serán las tecnologías que se han utilizado en este prototipo y la necesidad de utilizar nuevas para el desarrollo de ciertas funciones. Por ejemplo, en el caso de que se generase una API como se mencionaba con anterioridad, esta podría estar construida con Laravel, Symfony o Java. El aplicativo en su versión web podría ser simplemente un frontal desarrollado bajo Angular, Vue o React entre otros. Incluso se podría reconsiderar la opción de generar el aplicativo en versiones móviles bajo otro framework o incluso considerar la realización de cada versión del aplicativo bajo el software nativo de cada uno de los sistemas operativos móviles. Todos estos aspectos tendrán que ser considerados para realizar un crecimiento razonable del aplicativo en su conjunto e intentar no dar pasos en falsos que retrasen al operativo.

En lo personal, la realización de este prototipo me ha supuesto primero un cambio en la visión del análisis de un proyecto y las fases que deben pasar antes de comenzar con el desarrollo de un sistema software. Es vital no pasar directamente a desarrollar ningún software sin pararse a pensar qué deseamos construir y cuáles serán las necesidades de los usuarios cuando hagan uso del aplicativo.

Otra de las cosas que ha mejorado este proyecto en mi forma de ver el mundo del desarrollo, está en que por mucho que queramos llegar a conocer todos los campos del desarrollo es muy difícil manejarlos todos en un proyecto. Por ejemplo, en esta memoria se tocan muchas partes del desarrollo como pueden ser un análisis de la competencia, modelado de datos, bases de datos o uso de diversas tecnologías. Esto implica que debemos conocer cómo funciona el desarrollo y estar interesados en hacer crecer nuestros conocimientos generales de todos los campos sin dejar de concretar en alguno de los campos. En mi caso personal, he avanzado mucho en el conocimiento de lenguajes de programación e intentar comprender cada vez más como funcionan en profundidad y no en intentar conocer muchos sin adentrarte en sus entrañas, pero al realizar esta memoria me he dado cuenta de que estoy descuidando partes tan importantes del desarrollo como puede ser el análisis o la generación y administración de las bases de datos.

Este tipo de proyectos te hacen crecer no sólo a nivel profesional, sino profesional, a valorar lo importante que son cada una de las partes tanto del proyecto como las del equipo de desarrollo. Además, este tipo de proyectos te enseña que cualquier proyecto software que se quiera llevar a cabo tendrá que tener varios componentes si queremos hacer que funcione: un buen análisis, una buena elección de las tecnologías, un buen diseño y un buen equipo serán imprescindibles. Pero existe algo que es vital para que un proyecto funcione, el esfuerzo de todas las partes para dar lo máximo posible en su campo y hacer que el proyecto no sólo sea un buen diseño o una buena funcionalidad, sino una suma de muchas partes pensadas y trabajadas para que el proyecto sea lo mejor que cabía imaginar.



## 5 DOCUMENTACIÓN

Para la realización de esta memoria se han utilizado las siguientes fuentes. Cabe destacar que todos los enlaces webs se han comprado por última vez el 30 de diciembre de 2018, para comprobar que se encuentran todos disponibles.

- Angular
  - <https://angular.io/>
  - <https://angular.io/docs>
- ChartJS
  - <https://www.chartjs.org/docs/latest/>
  - <https://github.com/chartjs/Chart.js>
- Contasimple:
  - <https://www.contasimple.com/>
  - <https://play.google.com/store/apps/details?id=com.v2msoft.contasimple>
  - <https://itunes.apple.com/us/app/contasimple/id427583930?ls=1&mt=8>
- Factusol:
  - <https://www.sdelsol.com/programa-facturacion-factusol/>
  - <https://www.sdelsol.com/factusol360/>
  - <https://www.sdelsol.com/formacion/>
- Firebase
  - <https://firebase.google.com/docs/>
  - <https://firebase.google.com/docs/auth/>
  - <https://firebase.google.com/docs/database/>
- Invoice Simple:
  - <https://www.invoicesimple.com/>
  - [https://play.google.com/store/apps/details?id=com.aadhk.woinvoice&referrer=utm\\_source%3Dinvoicesimple%26utm\\_medium%3Dorganic%26utm\\_content%3Dfooter%26utm\\_campaign%3Dwebsite](https://play.google.com/store/apps/details?id=com.aadhk.woinvoice&referrer=utm_source%3Dinvoicesimple%26utm_medium%3Dorganic%26utm_content%3Dfooter%26utm_campaign%3Dwebsite)
  - <https://itunes.apple.com/app/apple-store/id694831622?mt=8>
- Ionic:
  - <https://ionicframework.com/>
  - <https://ionicframework.com/docs/components/>
  - <https://ionicframework.com/docs/api/>
  - <https://ionicframework.com/docs/native/>
- PdfMake
  - <http://pdfmake.org/#/>
  - <https://pdfmake.github.io/docs/>



- SASS
  - <https://sass-lang.com/>
  - [https://sass-lang.com/documentation/file.SASS\\_REFERENCE.html](https://sass-lang.com/documentation/file.SASS_REFERENCE.html)
- TypeScript
  - <https://www.typescriptlang.org/>
  - <https://www.typescriptlang.org/docs/home.html>



## 6 ANEXOS

- MANUAL DE USUARIO
- FORMATO DE LA FACTURA ELECTRÓNICA ESPAÑOLA

