

# GESCOM

Aplicación Móvil Android para la Gestión de Comunidades de Vecinos

Mobile Android Application for Neighbor Communities Management

**GRADO EN INGENIERÍA INFORMÁTICA**  
**UNIVERSIDAD COMPLUTENSE DE MADRID**



**TRABAJO FIN DE GRADO EN INGENIERÍA INFORMÁTICA**

**01/06/2021**

**Autores:**

Luis Jesús Cepeda Caperos  
Teresa Fernández Merino

**Director**

Iván García-Magariño García

Este trabajo se lo dedicamos Luis Vázquez,  
por haber sido durante estos cuatro años  
de carrera una fuente de motivación.  
Gracias

## Agradecimientos

Gracias a Albert, Aitor, Clara, Álvaro, Alberto y Bea por su apoyo incondicional sin el cual este proyecto no hubiera sido posible.

## Resumen

En este Trabajo de Fin de Grado hemos desarrollado una aplicación para dispositivos Android que facilita la gestión, por parte del administrador, de las comunidades de vecinos. Nuestra aplicación, GesCom, concentra una serie de funcionalidades relacionadas con las actividades rutinarias en las comunidades de vecinos, como son las funcionalidades de realizar votaciones, publicar avisos o el registro de incidencias. Los datos de la comunidad son proporcionados por el Administrador.

Palabras clave: Aplicaciones móviles, Android, Comunidad, Comunicación, Java.

## Abstract

In this End of Degree Project we have developed an application for Android devices that facilitates the management from the administrator's side in neighbouring communities. Our application, GesCom, concretes a series of functionalities related with neighbour communities routines, with functionalities such as voting, publishing announcements or registering incidents. Application data is provided by the community Administrator.

Key words: Mobile applications, Android, Community, Communication, Java.

# Índice General

<b>Índice de Figuras</b>	<b>8</b>
<b>Índice de Tablas</b>	<b>10</b>
<b>Introducción</b>	<b>11</b>
1.1 Motivación	12
1.2 Objetivos	13
1.3 Estructura	14
<b>Introduction</b>	<b>15</b>
1.1 Motivation	16
1.2 Objectives	17
1.3 Structure	18
<b>Estado del arte</b>	<b>19</b>
2.1 Investigación del mercado	20
2.2 Análisis de los resultados	27
<b>Desarrollo de Gescom</b>	<b>28</b>
3.1 Tecnologías empleadas	29
3.1.1 Android Studio	29
3.1.2 GitHub	29
3.1.3 Java	29
3.1.4 LucidChart	30
3.1.5 Creately	30
3.1.6 Google Docs	31
3.1.7 Google Forms	31
3.1.8 MySQL	31
3.1.9 Trello	31
3.2 Casos de uso	32
3.2.1 Tipos de roles	32
3.2.2 Usuario sin cuenta	34
3.2.2 Miembro de la comunidad	36
3.2.4 Administrador de la comunidad	42
3.3 Diseño de la aplicación	51
3.3.1 DashBoard	51
3.3.2 Incidencias	52
3.3.3 Votaciones	54
3.3.4 Avisos	54
3.3.5 Registro de usuario	56
3.3.6 Registro de una comunidad	57
3.3.7 Login	59
3.3.8 Gestión de usuarios	60
3.4 Modelo de datos	61

3.4.1 Modelo entidad-relación	62
3.4.2 Implementación de la BBDD	65
Tabla de usuarios	65
Tabla de comunidades	65
Tabla de incidencias	65
Tabla de avisos	66
Tabla de votaciones	66
Tabla de registro de votaciones	66
3.5 Arquitectura	67
3.5.1 Modelo MVP	67
3.5.2 Patrón de diseño: Singleton	68
<b>Interacción con usuarios</b>	<b>70</b>
4.1 Fases de la interacción	71
4.2 Recolección de datos previos al desarrollo	71
4.3 Análisis de experiencias de usuario	73
<b>Conclusiones y trabajo futuro</b>	<b>82</b>
5.1 Conclusiones	83
5.2 Conclusions	83
5.3 Trabajo futuro	84
<b>Aportaciones individuales</b>	<b>87</b>
6.1 Luis Jesús Cepeda Caperos	88
Ingeniería de requisitos	88
Desarrollo	88
Memoria	88
6.2 Teresa Fernández	90
Ingeniería de requisitos	90
Desarrollo	90
Memoria	91
<b>Bibliografía y referencias</b>	<b>93</b>
<b>Anexo I: Cuestionario Aplicación Gestión de Comunidades</b>	<b>99</b>
<b>Anexo II: Encuesta final a los usuarios</b>	<b>101</b>
<b>Anexo III: Manual de usuario</b>	<b>103</b>

## Índice de Figuras

Figura 1 - Opción de delegar voto en TuComunidApp .....	20
Figura 2 - Pantalla de votación en Mi Comunidad .....	21
Figura 3 - Necesidad de un código especial para verificar el registro del usuario.....	22
Figura 4 - Documentos de actas archivados con índice de importancia.....	23
Figura 5 - Botón de prueba para cargar una comunidad con datos preestablecidos...24	
Figura 6 - Botón para aceptar términos y condiciones .....	25
Figura 7 - Sección de leyes y normas.....	26
Figura 8 - DashBoard .....	50
Figura 9 - Diagrama de clases DashBoard .....	51
Figura 10 - Actividad de incidencias del administrador y del usuario .....	52
Figura 11 - Diagrama de interacción actividad de Incidencias .....	52
Figura 12 - Vista de votaciones para administrador y vecino .....	53
Figura 13 - Pantalla de avisos de usuario administrador.....	54
Figura 14 - Pantalla de avisos usuario vecino .....	54
Figura 15 - Diagrama de clases de aviso .....	55
Figura 16 - Formulario de registro de usuario .....	56
Figura 17 - Formulario de registro de una comunidad .....	57
Figura 18 - Pantalla de información de registro .....	57
Figura 19 - Pantalla de inicio de sesión.....	58
Figura 20 - Visualización de los usuarios de la comunidad .....	59
Figura 21 - Añadir un usuario a la comunidad .....	59
Figura 22 - Editar un usuario de la comunidad.....	60
Figura 23 - Eliminar un usuario de la comunidad .....	60
Figura 24 - Modelo Entidad-Relación.....	62
Figura 25 - Modelo Entidad-Relación de Avisos .....	63
Figura 26 - Modelo Entidad-Relación de Votaciones.....	63
Figura 27 - Patrón MVP en el registro de usuario .....	67
Figura 28 - Singleton empleado para el almacenamiento del estado global .....	68
Figura 29 - Ejemplo de pregunta en cuestionario SUS.....	73
Figura 30 - Resultados obtenidos en frecuencia de uso .....	73
Figura 31 - Resultados obtenidos en complejidad .....	74
Figura 32 - Resultados obtenidos en facilidad de uso .....	75
Figura 33 - Resultados obtenidos en necesidad de ayuda técnica.....	75
Figura 34 - Resultados obtenidos en integración de funcionalidades.....	76
Figura 35 - Resultados obtenidos en inconsistencia.....	76
Figura 36 - Resultados obtenidos en rapidez de aprendizaje .....	77
Figura 37 - Resultados obtenidos en dificultad de uso .....	77
Figura 38 - Resultados obtenidos en confianza.....	78
Figura 39 - Resultados obtenidos en aprendizaje previo .....	79
Figura 40 - Pantallas de login, registro de usuario y registro de comunidad .....	102
Figura 41 - Pantalla principal para diferentes usuarios.....	103
Figura 42 - Visualización de votaciones .....	104

Figura 43 - Cerrar votación .....	104
Figura 44 - Reportar incidencia .....	105
Figura 45 - Marcación de una incidencia como gestionada.....	105
Figura 46 - Gestión de avisos .....	106
Figura 47 - Gestión de usuarios.....	106

## Índice de Tablas

Tabla 1 - Registrar comunidad de vecinos .....	33
Tabla 2 - Registrar usuarios .....	34
Tabla 3 - Iniciar sesión .....	35
Tabla 4 - Cambiar contraseña .....	36
Tabla 5 - Visualizar avisos .....	37
Tabla 6 - Reportar incidencias .....	38
Tabla 7 - Participar en votaciones .....	39
Tabla 8 - Visualizar información de la comunidad.....	40
Tabla 9 - Iniciar sesión como administrador .....	41
Tabla 10 - Cambiar contraseña como administrador .....	42
Tabla 11 - Borrar aviso .....	43
Tabla 12 - Editar avisos .....	44
Tabla 13 - Concluir votación.....	45
Tabla 14 - Visualizar clave comunidad.....	46
Tabla 15 - Añadir usuario a la comunidad .....	47
Tabla 16 - Editar usuario de la comunidad.....	48
Tabla 17 - Eliminar usuario de la comunidad .....	49

# **Capítulo 1**

## **Introducción**

## 1.1 Motivación

Con el avance de la tecnología, los teléfonos móviles han seguido evolucionando y cambiando nuestra forma de interactuar con el mundo, debido a que estos dispositivos presentan un gran potencial para mejorar la calidad de vida de sus usuarios [1]. En el año 2020 se estimó que 5.645 mil millones de personas tenían acceso a un dispositivo móvil, equivalente a casi el 70% de la población mundial [2]. A la gran cantidad de personas que poseen uno de estos dispositivos, hay que sumarle la cantidad de tiempo que interactuamos con los mismos. La media de tiempo empleado en su uso es de 161.95 minutos al día [3] y el número de interacciones diarias oscilan entre 10 a 200 [4].

A partir de la creación de los primeros dispositivos móviles inteligentes, en 1992, se han podido ver cambios radicales en la forma de comunicación de las personas o la realización de algunas actividades rutinarias en la vida cotidiana. Ya no solo llamamos, mandamos mensajes instantáneos o accedemos a redes sociales, sino que nuestros dispositivos nos permiten realizar pagos instantáneos, comprobar tiempos de llegada, realizar cualquier tipo de compra, reproducir contenidos multimedia o acceder a rutinas de ejercicio físico entre otras cosas.

Las aplicaciones móviles a día de hoy son accesibles desde cualquier tipo de dispositivo a través de los diferentes mercados y tiendas de aplicaciones. Entre estos mercados, encontramos dos de los más destacados, Apple Store y Google Play, que ofrecen gran cantidad de aplicaciones para usuarios de los sistemas operativos Android e iOS, y constituyen las plataformas para dispositivos inteligentes más populares [5].

La gran mayoría de funcionalidades y herramientas que ofrecen los dispositivos móviles inteligentes se llevan a cabo haciendo uso de diferentes aplicaciones cada vez más abundantes en el mercado. En el segundo cuarto del año 2020, Google Play contaba con 2.7 millones de aplicaciones, seguida por Apple Store, para dispositivos iOS, con 1.82 millones [6]. Aunque las primeras aplicaciones aparecieron hace 20 años, se aprecia una evolución casi exponencial de su creación y desarrollo, lo que motiva la expansión de este software a diferentes campos.

Actualmente tan solo en Madrid existen más de 200.000 comunidades de vecinos [7], muchas de las cuales, han tenido que adoptar medidas excepcionales en los últimos meses debido a la Covid-19[8, 9]. Este proceso ha supuesto nuevas recomendaciones de conducta y reglamentos con el fin de reducir los impactos y los riesgos en la salud de los vecinos. Durante el

desarrollo de este trabajo, haremos uso de tecnologías y conocimientos técnicos aprendidos para expandir el uso de los dispositivos móviles en comunidades de vecinos, facilitando de esta forma el intercambio de información, sin necesidad del contacto físico, entre miembros de la comunidad.

Actualmente la gestión de comunidades de vecinos se ha visto muy afectada por la presencia de la Covid-19, ya que la realización de actividades rutinarias entre propietarios pone en riesgo su salud y la de todos los vecinos[10].

Nuestra motivación es ayudar a las personas a realizar aquellas tareas que pueden considerar tediosas en un momento en el que usar las tecnologías para mantenernos conectados y protegidos es más importante que nunca. Para ello crearemos una aplicación móvil para Android, intuitiva, accesible, flexible y para cualquier tipo de usuario.

## 1.2 Objetivos

El objetivo principal de nuestro proyecto es desarrollar una aplicación para dispositivos móviles con sistema operativo Android, que permita llevar a cabo tareas típicas en una comunidad de vecinos de una forma cómoda y accesible para todos. Para ello desarrollaremos dicha aplicación con unas funcionalidades específicas previamente definidas. Estos objetivos incluyen:

- Desarrollo de una aplicación para dispositivos móviles con sistema operativo Android cuyas funcionalidades permitan a los usuarios realizar votaciones, reportar incidencias, crear y leer avisos, consultar información general y gestionar usuarios.
- Desarrollo e implementación de una base de datos donde almacenar toda la información de cada comunidad, los usuarios y aquellos datos que maneje nuestra aplicación.
- Uso del entorno de desarrollo integrado Android Studio, en el que haremos uso del lenguaje de programación Java para realizar un diseño orientado a objetos adecuado a nuestras necesidades.

## 1.3 Estructura

En este apartado se describe el contenido y la organización de los apartados de este documento:

- **Capítulo 1:** Incluye una explicación de las razones que motivaron la realización de este trabajo, los objetivos establecidos al comienzo del desarrollo y la estructura general de esta memoria.
- **Capítulo 2:** Incluye un estudio de mercado de las aplicaciones más relevantes que se pueden descargar en este momento, que presentan funcionalidades y naturalezas parecidas a GesCom además de un análisis de los resultados observados tras dicho estudio.
- **Capítulo 3:** Incluye una breve descripción de las tecnologías empleadas en el desarrollo de GesCom, junto con las razones de elección de las mismas. A continuación se presentan algunos de los casos de uso más relevantes que los usuarios pueden llevar a cabo en nuestra aplicación, seguido de diagramas de clase, el modelo de datos empleado y la arquitectura que presenta nuestra aplicación.
- **Capítulo 4:** Se trata de un análisis y exposición de los resultados obtenidos al llevar a cabo las diferentes interacciones con usuarios.
- **Capítulo 5:** En el que se exponen las conclusiones obtenidas durante el desarrollo de este proyecto y los posibles trabajos futuros de mejora, incremento o iteración de funcionalidades.
- **Capítulo 6:** Expone las aportaciones individuales de cada miembro del equipo.
- **Capítulo 7:** Presenta la bibliografía y las referencias a la misma hechas en este documento.

A continuación de este capítulo se incluyen los anexos. En estos, se presenta la transcripción de las encuestas realizadas a los usuarios tanto al comienzo (Anexo I) como al final (Anexo II) del desarrollo y un manual de usuario (Anexo III).

# **Chapter 1**

## **Introduction**

## 1.1 Motivation

With the advancement of technology, mobile phones have continued to evolve and change the way we interact with the world, given that these devices have great potential to improve the quality of life of their users [1]. In 2020 it was estimated that 5.645 billion people had access to a mobile device, equivalent to almost 70% of the world's population [2]. To the already large number of people that own one of these devices, we must also add the amount of time they interact with them. The mean time spent using a mobile device is 161.95 minutes per day [3] and the number of daily interactions ranges from 10 to 200 [4].

Since the creation of the first smart mobile devices in 1992, we've been able to see a drastic change in the way people communicate or perform their daily routine activities. We not only call, send instant messages and access social networks, but our devices also allow us to make instant payments, check arrival times, make any purchase we want, play multimedia content or access physical exercise routines to name a few.

Today's mobile applications are accessible from any type of device through the different markets and application stores. Among these markets, we find two of the most prominent being Apple Store and Google Play, which offer a large number of applications for users of Android and iOS operating systems, and are the most popular platforms for smart devices [5].

The vast majority of functionalities and tools offered by smart mobile devices are carried out using different applications that are increasingly abundant on the market. In the second quarter of 2020, Google Play had 2.7 million applications, followed by the Apple Store, for iOS devices, with 1.82 million [6]. Although the first applications were created 20 years ago, there is an almost exponential evolution of its creation and development, which motivates the expansion of this software to different fields.

Currently in Madrid alone there are more than 200,000 neighborhoods [7], many of which have had to adopt exceptional measures in recent months due to Covid-19 [8, 9]. This process has led to newly advised conducts and regulations in order to reduce any potential impacts and health risks on neighbors. During the development of this project, we will make use of technologies and technical knowledge learned to expand the use of mobile devices in neighboring communities, thus facilitating the exchange of information without the need for physical contact between community members.

Currently, the management of neighborhoods has been greatly affected by the presence of Covid-19, since performing routine activities involving owners puts their health and that of all neighbors at risk [10].

Our motivation is to help people perform tasks that they may consider tedious at a time when using technologies to stay connected and protected is more important than ever. To achieve this we will create a mobile application for Android that's intuitive, accessible, flexible and usable by a wide variety of users.

## 1.2 Objectives

The main purpose of our project is to develop an application for mobile devices with the Android operating system, that allows us to carry out typical tasks in a neighborhood community in a comfortable and accessible way for everyone. For this, we will develop said application with specific previously defined functionalities. These goals include:

- Development of an application for mobile devices with Android as their operating system whose functionalities allow users to vote, report incidents, create and read notices, consult general information and manage users.
- Development and implementation of a database where to store all the information of each community, the users and data managed by our application.
- Use of the Android Studio Integrated Development Environment, in which we will use the Java programming language to make an object-oriented design appropriate to our needs.

## 1.3 Structure

This section describes the content and organization of the sections of this document:

- Chapter 1: Includes an explanation of the reasons that motivated the realization of this work, the objectives established at the beginning of the development and the general structure of this report.
- Chapter 2: Includes a market study of the most relevant applications that can be downloaded at this time, which have functionalities and a nature similar to GesCom, as well as an analysis of the results observed after said study.
- Chapter 3: Includes a brief description of the technologies used in the development of GesCom, together with the reasons for choosing them. This is followed by some of the most relevant use cases that users can carry out in our application, class diagrams, the data model used and the architecture that our application presents.
- Chapter 4: This is an analysis and presentation of the results obtained by carrying out the different interactions with users.
- Chapter 5: In which the conclusions obtained during the development of this project and the possible future works of improvement, increase or iteration of functionalities are exposed.
- Chapter 6: Exposes the individual contributions of each team member.
- Chapter 7: Presents the bibliography and references to it made in this document.

The annexes are included following the last chapter. In these, the transcript of the surveys carried out to the users is presented both at the beginning (Annex I) and at the end (Annex II) of the development and a user manual (Annex III).

# **Capítulo 2**

## **Estado del arte**

## 2.1 Investigación del mercado

La finalidad de la aplicación es ayudar con tareas rutinarias de una comunidad de vecinos, y que la realización de éstas en un dispositivo móvil suponga una ayuda y no una inconveniencia. Por esta razón comenzamos una fase de investigación, haciendo un estudio de mercado de aplicaciones para dispositivos móviles con sistema operativo Android cuyas funcionalidades faciliten la gestión de comunidades de vecinos.

Las dos características de mayor relevancia a la hora de investigar han sido las funcionalidades de cada aplicación y el proceso de registro como nuevo usuario o como administrador de la propiedad. Se ha seguido el mismo proceso para cada aplicación probada. Todas ellas han sido descargadas en un teléfono móvil físico para probar sus funcionalidades. En caso de tener dificultades a la hora de realizar la descarga de la aplicación, se ha hecho uso de la máquina virtual de *smartphone* 'BlueStacks' [11].

Las figuras aportadas son capturas de las diferentes aplicaciones o las fotos proporcionadas por el emulador si se tuvo dificultades para acceder a las funcionalidades principales. En caso de no poder completar el proceso de registro para comprobar qué herramientas ofrece, hemos utilizado la descripción de la aplicación, las imágenes de muestra y los comentarios para determinar sus funcionalidades.

Las aplicaciones mencionadas a continuación se pueden encontrar en *Google Store* [12].

**TucomunidApp** [13] - publicada por Informatización de Empresas S.L.

Funcionalidades destacadas:

- Consulta de recibos.
- Reportar incidencias.
- Reserva de áreas comunes (piscinas, pistas de fútbol, etc.)
- Comunicación mediante mensajes entre el propietario y el inquilino de una vivienda.
- Proponer temas para reuniones de vecinos.
- Realizar votaciones y posibilidad de delegar el voto. [Figura 1]

Proceso de registro:

Al abrir la aplicación el usuario tiene la opción de registrarse o iniciar sesión. Tras introducir una dirección de correo electrónico, la aplicación pide un código que debe proporcionar el dueño de la comunidad. No se especifica cómo registrarse si eres el administrador de una comunidad o

con quien contactar. El inicio de sesión resulta confuso por las direcciones contradictorias que da la aplicación.



[Figura 1] - Opción de delegar voto en TuComunidadApp

**Mi Comunidad** [14] - publicada por Community Pro.

Funcionalidades destacadas:

- Diferentes visualizaciones dependiendo del tipo de usuario.
- Alertas sobre mantenimientos periódicos.
- Foro de vecinos.
- Juntas de la comunidad: Fecha, hora y tema a tratar.
- Votación de la junta dentro de la aplicación: Fecha y hora para votar, temas a votar con las opciones de “abstención” “a favor” o “en contra”. [Figura 2]
- Avisos sobre incidencias en la comunidad organizadas por categorías: calefacción, fontanería, electricidad, albañilería...
- Apartado de emergencias para contactar con el 112 sobre caídas, quemaduras, electrocución, intoxicación, incendios o explosiones.

Proceso de registro:

Para el registro de un nuevo usuario, la aplicación pide los datos del propietario de la vivienda: nombre, dirección, correo electrónico y teléfono. También hay otro apartado para introducir los datos del

administrador de la comunidad. No queda clara la obligación del usuario de rellenar todos los campos. Al intentar registrarnos en la aplicación aparece el mensaje “Por favor, inténtelo de nuevo más tarde”, y no permite seguir.



[Figura 2] - Pantalla de votación en Mi Comunidad

## ComunidadFeliz [15] - publicada por ComunidadFeliz

Funcionalidades destacadas:

- No hay un límite en el número de comunidades que se pueden registrar.
- Realizar pagos a través de la aplicación de gastos comunes, con historial y comprobantes de los pagos realizados.
- Reserva de espacios comunes por horas.
- Enviar mensajes de invitación a través de las redes sociales para los espacios comunes reservados.
- Notificaciones de la administración pública.
- Opción de mandar mensajes al administrador.

Proceso de registro:

Para verificar el registro en esta aplicación, el usuario debe seleccionar su país de residencia y posteriormente escanear con el teléfono un código QR que recibe por parte del administrador de la comunidad [Figura 3]. También tiene la opción de introducirlo manualmente. Aparece en pantalla un link para acceder a su página web, en caso de ser un administrador de una comunidad, el usuario debe estar dispuesto a pagar para dar de alta la comunidad en la aplicación, por el contrario, los vecinos no pagan gastos adicionales una vez registrada la comunidad y descargar la aplicación es gratis.



[Figura 3] - Necesidad de un código especial para verificar el registro del usuario

**SomosVecinos** [16] - publicada por BiblioEteca Technologies.

Funcionalidades destacadas:

- Alertas de incidencias
- Notificaciones de reuniones
- Teléfonos de contacto de otros vecinos
- Archivos de las diferentes leyes
- Archivos de las diferentes actas con diferentes categorías según su importancia (normal o urgente) [Figura 4]

Proceso de registro:

La aplicación muestra directamente los campos para iniciar sesión. En la sección de registro pide nombre de usuario, apellidos, contraseña, la cual se debe introducir dos veces para verificarla, correo electrónico, teléfono de contacto y dirección. Resulta confuso saber si quien debe registrarse es el usuario, o el administrador de la comunidad. Tras introducir los datos aparece el mensaje de “usuario registrado con éxito” pero al intentar iniciar sesión presenta al usuario un mensaje de error. No está claro cómo utilizar la aplicación. En la última línea de la descripción de la aplicación en la App Store, se encuentran las instrucciones que no aparecían en la aplicación y una dirección de correo electrónico a la que debe contactar el administrador para registrar la comunidad de vecinos.



[Figura 4] - Documentos de actas archivados con índice de importancia


**MyComunidad** [17] - publicada por VultureSoft.

Funcionalidades destacadas:

- Botón de prueba en la pantalla de registro [Figura 5]
- Lista de contactos de los miembros de la comunidad de vecinos
- Incidencias de la comunidad, organizadas en categorías: abiertas, cerradas, mías, todas
- Posibilidad de añadir una foto cuando creas una nueva incidencia
- Reservas de áreas comunes por horas, es posible reservar más de una plaza en una misma reserva.
- Tablón de anuncios con noticias importantes de la comunidad
- Normas de la comunidad y de las áreas comunes
- Posibilidad de recibir las noticias del tablón de anuncios por email

Proceso de registro:

Una única pantalla para iniciar sesión. No aparece información en la aplicación de cómo darse de alta, en la *App Store* informan de llevar a cabo el registro en su página web, pero no hay referencia a la misma. Existe un botón "prueba" en la aplicación para interactuar con un modelo de prueba de una comunidad.



MyComunidad

Usuario  
admin

Contraseña  
.....

Recordar contraseña

ENTRAR

¿Tienes alguna pregunta?

PRUEBA

[Figura 5] - Botón de prueba para cargar una comunidad con datos preestablecidos

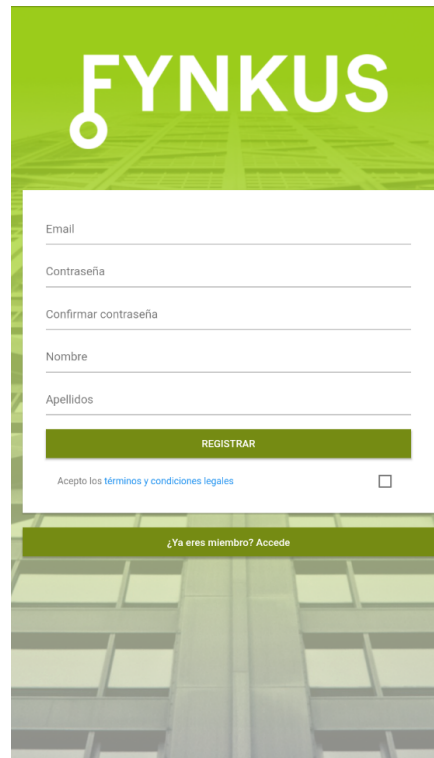
**Fynkus** [18] - publicada por Fynkus.

Funcionalidades destacadas:

- Saldo de las propiedades
- Comprobar gastos de contabilidad
- Notificaciones de juntas generales y actas
- Importe de la morosidad de la comunidad
- Gastos totales que ha tenido la comunidad

Proceso de registro:

Al intentar registrarse el usuario, se le redirecciona a la página web de la empresa para tramitar el pago del servicio. Permite usar sus servicios tanto en la web como a través de la aplicación móvil. El usuario debe aceptar los términos y condiciones de la aplicación en el proceso de registro [Figura 6].



[Figura 6] - Botón para aceptar términos y condiciones

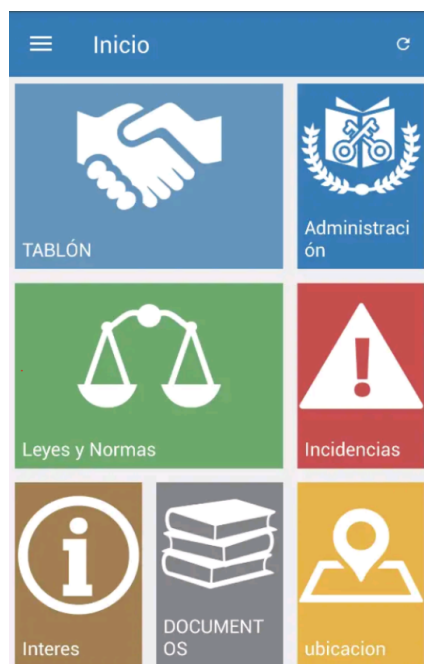
### **Tu Comunidad** [19] - publicada por Tumiapps.

#### Funcionalidades destacadas:

- Teléfonos de contacto
- Normas de la comunidad
- Tablón de anuncios con información general
- Horarios de instalaciones de uso común
- Marcar incidencias, con posibilidad de adjuntar una fotografía
- Reglamento y leyes de la vivienda [Figura 7]
- Opción de contactar con el administrador de la comunidad

#### Proceso de registro:

Al iniciar la aplicación, el usuario se encuentra con que su única opción es iniciar sesión. Para ello le pide introducir el código de tu comunidad. No hay establecida ninguna medida de seguridad, ya que con saber el código de una comunidad tienes acceso a ella, el usuario no tiene obligación de verificar su identidad.



[Figura 7] - Sección de leyes y normas

## 2.2 Análisis de los resultados

Entre las aplicaciones analizadas observamos que muchas de ellas comparten funcionalidades, como la reserva de áreas comunes, un tablón de anuncios, reportar incidencias o recibir notificaciones de reuniones de vecinos. También se aprecia que el proceso de registro resulta muy confuso y no presenta instrucciones claras para el usuario.

No hay aplicaciones en el mercado según lo analizado con funcionalidades más simples y con instrucciones e interfaces claras para el usuario. Además no hemos encontrado aplicaciones recientemente actualizadas que incorporen funciones como notificar de vecinos en cuarentena o herramientas similares que puedan ayudar a reducir el contacto entre vecinos.

Un aspecto de negocio que potenciará nuestra aplicación es la sencillez y gratuidad de todas las funcionalidades, además de la ausencia de anuncios. Por otro lado, hemos intentado que nuestra aplicación maneje el menor número de datos sensibles posible, equilibrando así la seguridad de la aplicación con la confianza del usuario.

# **Capítulo 3**

## **Desarrollo de Gescom**

## 3.1 Tecnologías empleadas

A lo largo de este apartado se van a exponer aquellas herramientas implicadas en el desarrollo de GesCom acompañadas de las razones por las cuales se ha hecho uso de las mismas y no de otras alternativas.

### 3.1.1 Android Studio

Se trata de una herramienta que proporciona al usuario un entorno dirigido al desarrollo de aplicaciones para dispositivos móviles con sistema operativo Android.

Nos decantamos por usar esta herramienta ya que es apta para ordenadores con los sistemas operativos Windows, Linux o macOS. Además el desarrollo en esta herramienta se realiza en código nativo con el Kit de Desarrollo de Software [20], por lo que se logra un gran aumento de la funcionalidad integrable en el código. Al hacer uso de esta herramienta, el mercado al que aspira nuestra aplicación se adapta a aquellos dispositivos con sistema Android, los cuales acaparan la gran mayoría contabilizando el 80% del mercado de dispositivos móviles [21].

### 3.1.2 GitHub

Dado que el desarrollo de este proyecto se iba a realizar entre dos personas, vimos clara la necesidad de un repositorio conjunto que permitiese el desarrollo tanto del código como de la memoria en paralelo.

Como respuesta a esta necesidad creamos un repositorio en GitHub, el cual hace uso de la tecnología Git, empleado para el control de versiones, que permite la descentralización, facilitando así el desarrollo de un proyecto y la rectificación de errores en el código [22, 23].

A continuación se proporciona el link al repositorio empleado:

<https://github.com/teresitaFerme/GesCom.git>

### 3.1.3 Java

En el mundo del desarrollo de aplicaciones Android los lenguajes más empleados son Java y Kotlin, siendo este último una evolución del primero [24,25]. La gran ventaja del uso de estos lenguajes es la posibilidad de hacerlos coexistir en una misma aplicación.

Para la realización de nuestra aplicación, el lenguaje de programación más empleado será Java combinado junto con XML para la gestión de las interfaces[26, 27]. La decisión de emplear Java como lenguaje principal ha sido tomada en base a nuestros conocimientos previos del lenguaje, ya que aquellas limitaciones de Java que soluciona el lenguaje Kotlin, no presentan un problema para el desarrollo de nuestra aplicación [28].

Java es un lenguaje de programación orientado a objetos que fue diseñado siguiendo el principio de “escribe una vez, úsalo en cualquier sitio” [29], lo que hizo que este lenguaje pudiera emplearse en todas las plataformas que soportan Java sin tener que realizar recompilación. Originalmente Java fue desarrollado por James Gosling en Sun Microsystems, el cual fue adquirido por la conocida Oracle [30].

Por otra parte, el uso de Java en nuestra aplicación, facilita la implementación de patrones arquitectónicos y de diseño en el código de nuestra aplicación, lo que favorece a la eficiencia y buena organización del mismo.

Otras ventajas que ofrece el uso de un lenguaje orientado a objetos son la reusabilidad de clases para distintos proyectos o partes de un programa, la mantenibilidad, la modificabilidad y la fiabilidad del código, ya que se pueden hacer tests de manera totalmente independiente.

### 3.1.4 LucidChart

Se trata de una herramienta web gratuita que proporciona una interfaz gráfica para el desarrollo de todo tipo de diagramas y gráficos [31]. Esta herramienta nos permite hacer de manera sencilla e intuitiva diagramas de flujo y diagramas de clases.

### 3.1.5 Creately

Al igual que la anteriormente mencionada herramienta, Creatively [32] es una herramienta web gratuita que permite la implementación de gran variedad de gráficos y diagramas. Hemos utilizado esta herramienta para la implementación del modelo entidad relación.

### 3.1.6 Google Docs

Se trata de un editor de texto gratuito que forma parte de la suite web ofimática gratuita de Google que incluye otros editores como un editor para hojas de cálculo o un programa para la elaboración de presentaciones[33]. Este editor de texto permite el trabajo colaborativo y paralelo en un documento, lo que soluciona el problema de trabajar en grupo en esta memoria.

### 3.1.7 Google Forms

Esta herramienta es un software de administración de cuestionarios ofrecido por Google, al igual que Google Docs, en su suite web gratuita de editores [34, 35]. Una de las ventajas de uso de esta herramienta es el volcado automático de los datos a hojas de cálculo, lo que permite un análisis ordenado y rápido de las respuestas obtenidas en un cuestionario.

### 3.1.8 MySQL

Sistema de gestión de bases de datos relacionales, comúnmente empleado en el desarrollo de aplicaciones web debido a que se trata de un sistema de código abierto distribuido bajo la licencia GPL (General Public License) [36, 37]. Este sistema hace uso de un lenguaje de peticiones estructurado (Structured Query Language), para la introducción, edición y eliminación de datos de la base de datos relacional [38]. Además constituye una combinación de servidor y línea de comandos de cliente.

El empleo de este sistema se debe a su fácil uso, implementación y a la naturaleza de los datos que se tratan en GesCom, los cuales son fácilmente aplicables a una base de datos relacional que organiza los mismos en tablas cuya información puede estar relacionada entre sí.

### 3.1.9 Trello

Todos los proyectos de desarrollo de software siguen algún marco de trabajo estructurado para garantizar que se alcanzan los objetivos. En el desarrollo de GesCom nos hemos decantado por el uso de la metodología SCRUM combinada con la metodología Kanban [39, 40]. Estas metodologías son ampliamente utilizadas en el desarrollo de aplicaciones ya que forman parte del conjunto de metodologías ágiles [41].

La metodología SCRUM tiene como objetivo garantizar el éxito del proyecto con respecto al tiempo, el coste y el alcance del mismo. De la misma manera, Kanban resulta una metodología exitosa en la industria de desarrollo de software, principalmente en proyectos en los que debe haber un desarrollo continuo y entregas habituales de productos [42].

Trello es una herramienta gratuita que permite la creación de tableros de organización de proyectos que se llevan a cabo mediante metodologías ágiles [43], en nuestro caso la metodología SCRUM. En nuestro caso la metodología empleada ha sido iterativa e incremental.

El desarrollo de este proyecto se ha realizado en diferentes sprints que combinan el desarrollo de código con la redacción de la memoria, por ello se han creado dos tableros diferentes orientados a cada una de estas dos partes.

## 3.2 Casos de uso

### 3.2.1 Tipos de roles

Para obtener los diferentes casos de uso es necesario entender primero los diferentes tipos de usuarios que pueden interactuar con la aplicación.

**Usuario sin cuenta:** Son aquellos usuarios que abren la aplicación por primera vez. Es el usuario que dispone del menor número de funcionalidades. Podrán registrarse como un nuevo usuario de una comunidad, siempre que el administrador les haya proporcionado el código de la comunidad de vecinos. Si realizan este proceso con éxito pasarán a ser “miembro de la comunidad”. También podrán registrar una nueva comunidad, una vez introducidos los datos adecuados serán “administrador de la comunidad”. Por último tendrán la opción de iniciar sesión o recuperar su contraseña, pero al ser usuarios que no disponen de una cuenta todavía no podrán realizar ninguna de estas dos tareas.

**Miembro de la comunidad:** Son aquellos usuarios que se registraron con éxito como residentes de una comunidad de vecinos. Dentro de su propia comunidad de vecinos tienen acceso a diferentes

funcionalidades. Pueden participar en votaciones y ver los resultados de las mismas, y tienen la opción de reportar incidencias. También pueden ver el historial de las incidencias que hayan reportado anteriormente. Además pueden visualizar el tablón con avisos de la comunidad.

**Administrador de la comunidad:** Son aquellos usuarios que se han registrado como administradores de una comunidad de vecinos. Como administradores pueden visualizar el código de su comunidad, el cual podrán enviar a los miembros de la comunidad para que puedan completar el proceso de registro. Como administrador tienen acceso a funcionalidades únicas. Un administrador de la comunidad puede crear, borrar y editar los avisos, estos avisos son visualizados por todos los miembros dentro de la misma comunidad. El administrador recibirá todas las incidencias reportadas por los miembros y tiene la posibilidad de gestionarlas. Además podrá crear y cerrar las votaciones de la comunidad al igual que visualizar los resultados.

### 3.2.2 Usuario sin cuenta

UC - 01	Registrar comunidad de vecinos																	
Versión	1.0																	
Descripción	Un usuario sin cuenta podrá registrar una comunidad de vecinos.																	
Prioridad	Alta																	
Estabilidad	Alta																	
Precondición	N/A																	
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El usuario abre la aplicación.</td> </tr> <tr> <td>2</td> <td>El usuario selecciona la opción de registrar comunidad.</td> </tr> <tr> <td>3</td> <td>El usuario introduce la información solicitada para cada campo que se muestra en pantalla.</td> </tr> <tr> <td>4</td> <td>El usuario pulsa el botón de registrar.</td> </tr> <tr> <td>5</td> <td>El sistema valida que los datos proporcionados son correctos.</td> </tr> <tr> <td>7</td> <td>El sistema muestra una pantalla con información de cómo añadir un usuario.</td> </tr> <tr> <td>8</td> <td>El usuario pulsa el botón para finalizar el registro.</td> </tr> </tbody> </table>	Paso	Acción	1	El usuario abre la aplicación.	2	El usuario selecciona la opción de registrar comunidad.	3	El usuario introduce la información solicitada para cada campo que se muestra en pantalla.	4	El usuario pulsa el botón de registrar.	5	El sistema valida que los datos proporcionados son correctos.	7	El sistema muestra una pantalla con información de cómo añadir un usuario.	8	El usuario pulsa el botón para finalizar el registro.	
	Paso	Acción																
	1	El usuario abre la aplicación.																
	2	El usuario selecciona la opción de registrar comunidad.																
	3	El usuario introduce la información solicitada para cada campo que se muestra en pantalla.																
	4	El usuario pulsa el botón de registrar.																
	5	El sistema valida que los datos proporcionados son correctos.																
	7	El sistema muestra una pantalla con información de cómo añadir un usuario.																
	8	El usuario pulsa el botón para finalizar el registro.																
Postcondición	La comunidad queda registrada en la base de datos. El usuario podrá utilizar el nombre de usuario y contraseña que introdujo durante el proceso de registro para iniciar sesión como administrador de la comunidad. Se generará un código de comunidad y una cuenta de usuario por cada miembro dentro de esa comunidad.																	
Excepciones	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>5</td> <td>Se encuentran campos sin rellenar.</td> </tr> <tr> <td>5</td> <td>El número de documento de identidad no tiene un formato correcto.</td> </tr> </tbody> </table>	Paso	Acción	5	Se encuentran campos sin rellenar.	5	El número de documento de identidad no tiene un formato correcto.											
	Paso	Acción																
	5	Se encuentran campos sin rellenar.																
5	El número de documento de identidad no tiene un formato correcto.																	
Dependencias	N/A																	
Comentarios	La opción de iniciar sesión se encuentra en la primera pantalla al abrir la aplicación. El usuario que registra la comunidad será registrado como administrador de la comunidad.																	

Tabla 1 - Registrar comunidad de vecinos

UC - 02	Registrar usuario															
Versión	1.0															
Descripción	Un usuario sin cuenta podrá registrarse como miembro dentro de una comunidad de vecinos.															
Prioridad	Alta															
Estabilidad	Alta															
Precondición	El usuario ha recibido el código de la comunidad por parte del administrador de la comunidad.															
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El usuario abre la aplicación .</td> </tr> <tr> <td>2</td> <td>El usuario selecciona la opción de registrar usuario.</td> </tr> <tr> <td>3</td> <td>El usuario introduce la información solicitada para cada campo que se muestra en pantalla.</td> </tr> <tr> <td>4</td> <td>El usuario pulsa el botón de confirmar.</td> </tr> <tr> <td>5</td> <td>El sistema valida que los datos proporcionados son correctos.</td> </tr> <tr> <td>6</td> <td>El sistema registra al usuario dentro de la comunidad de vecinos.</td> </tr> </tbody> </table>	Paso	Acción	1	El usuario abre la aplicación .	2	El usuario selecciona la opción de registrar usuario.	3	El usuario introduce la información solicitada para cada campo que se muestra en pantalla.	4	El usuario pulsa el botón de confirmar.	5	El sistema valida que los datos proporcionados son correctos.	6	El sistema registra al usuario dentro de la comunidad de vecinos.	
	Paso	Acción														
	1	El usuario abre la aplicación .														
	2	El usuario selecciona la opción de registrar usuario.														
	3	El usuario introduce la información solicitada para cada campo que se muestra en pantalla.														
	4	El usuario pulsa el botón de confirmar.														
	5	El sistema valida que los datos proporcionados son correctos.														
	6	El sistema registra al usuario dentro de la comunidad de vecinos.														
Postcondición	El usuario queda registrado en la base de datos. Con el código de la comunidad el usuario queda registrado dentro de esa comunidad de vecinos. El usuario podrá utilizar el nombre de usuario y contraseña que introdujo durante el proceso de registro para iniciar sesión como miembro de la comunidad.															
Excepciones	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>5</td> <td>Se encuentran campos sin rellenar.</td> </tr> <tr> <td>5</td> <td>El número de documento de identidad no tiene un formato correcto.</td> </tr> </tbody> </table>	Paso	Acción	5	Se encuentran campos sin rellenar.	5	El número de documento de identidad no tiene un formato correcto.									
	Paso	Acción														
	5	Se encuentran campos sin rellenar.														
5	El número de documento de identidad no tiene un formato correcto.															
Dependencias	UC-01, el código de comunidad generado se usa como campo para el registro de un miembro de la comunidad.															
Comentarios	La opción de iniciar sesión se encuentra en la pantalla principal nada más abrir la aplicación.															

Tabla 2 - Registrar usuarios

## 3.2.2 Miembro de la comunidad

UC - 03	Iniciar sesión	
Versión	1.0	
Descripción	Un miembro de la comunidad podrá iniciar sesión dentro de la aplicación.	
Prioridad	Alta	
Estabilidad	Alta	
Precondición	El usuario ha creado una cuenta.	
Secuencia normal	Paso	Acción
	1	El usuario abre la aplicación .
	2	El usuario introduce su nombre de usuario, su contraseña y pulsa iniciar sesión.
	3	El sistema valida que los datos proporcionados son correctos.
	4	El sistema registra una nueva comunidad y guarda en caché los datos básicos que ha introducido el usuario.
Postcondición	El sistema carga los datos asociados al usuario y muestra la pantalla principal una vez completado el proceso de iniciar sesión.	
Excepciones	Paso	Acción
	3	No se completa el proceso de iniciar sesión. Los datos proporcionados no corresponden a ningún usuario existente.
Dependencias	Formato de datos guardados en la base de datos, usuario y contraseña cifrada.	
Comentarios	La opción de iniciar sesión se encuentra en la pantalla principal nada más abrir la aplicación.	

Tabla 3 - Iniciar sesión

UC - 04	Cambiar contraseña																	
Versión	1.0																	
Descripción	Un miembro de la comunidad podrá cambiar su contraseña.																	
Prioridad	Alta																	
Estabilidad	Alta																	
Precondición	El usuario ha creado una cuenta como miembro de una comunidad de vecinos.																	
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El usuario abre la aplicación e inicia sesión.</td> </tr> <tr> <td>2</td> <td>El usuario despliega el menú de opciones y accede al apartado de ajustes.</td> </tr> <tr> <td>3</td> <td>El sistema muestra los ajustes del sistema.</td> </tr> <tr> <td>4</td> <td>El usuario accede a la sección de cambiar contraseña.</td> </tr> <tr> <td>5</td> <td>El usuario introduce su contraseña actual e introduce dos veces su nueva contraseña y confirma sus modificaciones.</td> </tr> <tr> <td>6</td> <td>El sistema valida la petición solicitada.</td> </tr> <tr> <td>7</td> <td>El sistema actualiza la contraseña del usuario.</td> </tr> </tbody> </table>	Paso	Acción	1	El usuario abre la aplicación e inicia sesión.	2	El usuario despliega el menú de opciones y accede al apartado de ajustes.	3	El sistema muestra los ajustes del sistema.	4	El usuario accede a la sección de cambiar contraseña.	5	El usuario introduce su contraseña actual e introduce dos veces su nueva contraseña y confirma sus modificaciones.	6	El sistema valida la petición solicitada.	7	El sistema actualiza la contraseña del usuario.	
	Paso	Acción																
	1	El usuario abre la aplicación e inicia sesión.																
	2	El usuario despliega el menú de opciones y accede al apartado de ajustes.																
	3	El sistema muestra los ajustes del sistema.																
	4	El usuario accede a la sección de cambiar contraseña.																
	5	El usuario introduce su contraseña actual e introduce dos veces su nueva contraseña y confirma sus modificaciones.																
	6	El sistema valida la petición solicitada.																
	7	El sistema actualiza la contraseña del usuario.																
Postcondición	La nueva contraseña del usuario se ha actualizado. Se guarda dicha contraseña cifrada en la base de datos.																	
Excepciones	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>6</td> <td>No se completa el proceso de actualizar la contraseña. La contraseña introducida coincide con la contraseña actual.</td> </tr> </tbody> </table>	Paso	Acción	6	No se completa el proceso de actualizar la contraseña. La contraseña introducida coincide con la contraseña actual.													
	Paso	Acción																
6	No se completa el proceso de actualizar la contraseña. La contraseña introducida coincide con la contraseña actual.																	
Dependencias	Contraseña almacenada en la base de datos.																	
Comentarios	N/A																	

Tabla 4 - Cambiar contraseña

UC - 05	Visualizar avisos									
Versión	1.0									
Descripción	Un miembro de la comunidad podrá visualizar los avisos publicados de su comunidad.									
Prioridad	Alta									
Estabilidad	Media									
Precondición	El usuario dispone de una cuenta.									
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El usuario abre la aplicación e inicia sesión.</td> </tr> <tr> <td>2</td> <td>El usuario despliega el menú de opciones y accede al apartado de avisos.</td> </tr> <tr> <td>3</td> <td>El sistema muestra los avisos de la comunidad.</td> </tr> </tbody> </table>		Paso	Acción	1	El usuario abre la aplicación e inicia sesión.	2	El usuario despliega el menú de opciones y accede al apartado de avisos.	3	El sistema muestra los avisos de la comunidad.
Paso	Acción									
1	El usuario abre la aplicación e inicia sesión.									
2	El usuario despliega el menú de opciones y accede al apartado de avisos.									
3	El sistema muestra los avisos de la comunidad.									
Postcondición	El usuario visualizará en pantalla los avisos de la comunidad.									
Excepciones	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>3</td> <td>La pantalla de aviso no contiene ningún aviso. El sistema muestra un mensaje alternativo para informar al usuario de que no hay avisos en este momento.</td> </tr> </tbody> </table>		Paso	Acción	3	La pantalla de aviso no contiene ningún aviso. El sistema muestra un mensaje alternativo para informar al usuario de que no hay avisos en este momento.				
Paso	Acción									
3	La pantalla de aviso no contiene ningún aviso. El sistema muestra un mensaje alternativo para informar al usuario de que no hay avisos en este momento.									
Dependencias	Avisos publicados por el administrador de la comunidad.									
Comentarios	N/A									

Tabla 5 - Visualizar avisos

UC - 06	Reportar incidencias														
Versión	1.0														
Descripción	Un miembro de la comunidad podrá reportar incidencias al administrador.														
Prioridad	Alta														
Estabilidad	Media														
Precondición	El usuario dispone de una cuenta.														
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El usuario abre la aplicación e inicia sesión.</td> </tr> <tr> <td>2</td> <td>El usuario despliega el menú de opciones y accede al apartado de incidencias.</td> </tr> <tr> <td>3</td> <td>El sistema muestra la interfaz del apartado de incidencias.</td> </tr> <tr> <td>4</td> <td>El usuario pulsa el símbolo de añadir incidencia y procede a rellenar los datos solicitados.</td> </tr> <tr> <td>5</td> <td>El sistema valida los datos introducidos.</td> </tr> <tr> <td>6</td> <td>El sistema agrega con éxito la incidencia.</td> </tr> </tbody> </table>	Paso	Acción	1	El usuario abre la aplicación e inicia sesión.	2	El usuario despliega el menú de opciones y accede al apartado de incidencias.	3	El sistema muestra la interfaz del apartado de incidencias.	4	El usuario pulsa el símbolo de añadir incidencia y procede a rellenar los datos solicitados.	5	El sistema valida los datos introducidos.	6	El sistema agrega con éxito la incidencia.
Paso	Acción														
1	El usuario abre la aplicación e inicia sesión.														
2	El usuario despliega el menú de opciones y accede al apartado de incidencias.														
3	El sistema muestra la interfaz del apartado de incidencias.														
4	El usuario pulsa el símbolo de añadir incidencia y procede a rellenar los datos solicitados.														
5	El sistema valida los datos introducidos.														
6	El sistema agrega con éxito la incidencia.														
Postcondición	La nueva incidencia se guarda en la base de datos, el usuario podrá visualizar todas las incidencias que ha realizado.														
Excepciones	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>5</td> <td>No se permite añadir la incidencia. No se han rellenado todos los campos solicitados.</td> </tr> </tbody> </table>	Paso	Acción	5	No se permite añadir la incidencia. No se han rellenado todos los campos solicitados.										
Paso	Acción														
5	No se permite añadir la incidencia. No se han rellenado todos los campos solicitados.														
Dependencias	N/A														
Comentarios	N/A														

Tabla 6 - Reportar incidencias

UC - 07	Participar en votaciones																	
Versión	1.0																	
Descripción	Un miembro de la comunidad puede participar en las votaciones de la comunidad, votando la opción que considere apropiada.																	
Prioridad	Alta																	
Estabilidad	Media																	
Precondición	El usuario dispone de una cuenta. Hay una votación en proceso.																	
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El usuario abre la aplicación e inicia sesión.</td> </tr> <tr> <td>2</td> <td>El usuario despliega el menú de opciones y accede al apartado de votaciones.</td> </tr> <tr> <td>3</td> <td>El sistema muestra la interfaz del apartado de votaciones.</td> </tr> <tr> <td>4</td> <td>El usuario selecciona si está a favor o en contra según el título y descripción de la votación.</td> </tr> <tr> <td>5</td> <td>El sistema muestra una pestaña para pedir al usuario que confirme su voto.</td> </tr> <tr> <td>6</td> <td>El usuario confirma su voto.</td> </tr> <tr> <td>7</td> <td>El sistema registra el voto del usuario.</td> </tr> </tbody> </table>		Paso	Acción	1	El usuario abre la aplicación e inicia sesión.	2	El usuario despliega el menú de opciones y accede al apartado de votaciones.	3	El sistema muestra la interfaz del apartado de votaciones.	4	El usuario selecciona si está a favor o en contra según el título y descripción de la votación.	5	El sistema muestra una pestaña para pedir al usuario que confirme su voto.	6	El usuario confirma su voto.	7	El sistema registra el voto del usuario.
Paso	Acción																	
1	El usuario abre la aplicación e inicia sesión.																	
2	El usuario despliega el menú de opciones y accede al apartado de votaciones.																	
3	El sistema muestra la interfaz del apartado de votaciones.																	
4	El usuario selecciona si está a favor o en contra según el título y descripción de la votación.																	
5	El sistema muestra una pestaña para pedir al usuario que confirme su voto.																	
6	El usuario confirma su voto.																	
7	El sistema registra el voto del usuario.																	
Postcondición	Se actualizan los resultados de la votación en la base de datos. Se muestran los datos actualizados de la votación al usuario.																	
Excepciones	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>6</td> <td>El usuario cancela la confirmación del voto. El sistema volverá a la secuencia normal 3, donde el usuario podrá volver a elegir qué votar.</td> </tr> </tbody> </table>		Paso	Acción	6	El usuario cancela la confirmación del voto. El sistema volverá a la secuencia normal 3, donde el usuario podrá volver a elegir qué votar.												
Paso	Acción																	
6	El usuario cancela la confirmación del voto. El sistema volverá a la secuencia normal 3, donde el usuario podrá volver a elegir qué votar.																	
Dependencias	Posibilidad de votación establecida por el administrador.																	
Comentarios	Solo se permite un voto por participante.																	

Tabla 7 - Participar en votaciones

UC - 08	Visualizar información de la comunidad									
Versión	1.0									
Descripción	Un miembro de la comunidad puede visualizar información de la comunidad a la que pertenece.									
Prioridad	Alta									
Estabilidad	Alta									
Precondición	El usuario dispone de una cuenta.									
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El usuario abre la aplicación e inicia sesión.</td> </tr> <tr> <td>2</td> <td>El usuario despliega el menú de opciones y accede al apartado de información.</td> </tr> <tr> <td>3</td> <td>El sistema muestra la interfaz del apartado con la información de la comunidad.</td> </tr> </tbody> </table>		Paso	Acción	1	El usuario abre la aplicación e inicia sesión.	2	El usuario despliega el menú de opciones y accede al apartado de información.	3	El sistema muestra la interfaz del apartado con la información de la comunidad.
Paso	Acción									
1	El usuario abre la aplicación e inicia sesión.									
2	El usuario despliega el menú de opciones y accede al apartado de información.									
3	El sistema muestra la interfaz del apartado con la información de la comunidad.									
Postcondición	El usuario puede visualizar la información de la comunidad.									
Excepciones	N/A									
Dependencias	La información introducida por el administrador al registrar la comunidad.									
Comentarios	N/A									

Tabla 8 - Visualizar información de la comunidad

### 3.2.4 Administrador de la comunidad

UC - 09	Iniciar sesión como administrador											
Versión	1.0											
Descripción	Un administrador de la comunidad podrá iniciar sesión dentro de la aplicación.											
Prioridad	Alta											
Estabilidad	Alta											
Precondición	El usuario ha creado una cuenta.											
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El administrador abre la aplicación .</td> </tr> <tr> <td>2</td> <td>El administrador introduce su nombre de usuario, su contraseña y pulsa iniciar sesión.</td> </tr> <tr> <td>3</td> <td>El sistema valida que los datos proporcionados son correctos.</td> </tr> <tr> <td>4</td> <td>El sistema registra una nueva comunidad y guarda en caché los datos básicos que ha introducido el usuario.</td> </tr> </tbody> </table>		Paso	Acción	1	El administrador abre la aplicación .	2	El administrador introduce su nombre de usuario, su contraseña y pulsa iniciar sesión.	3	El sistema valida que los datos proporcionados son correctos.	4	El sistema registra una nueva comunidad y guarda en caché los datos básicos que ha introducido el usuario.
Paso	Acción											
1	El administrador abre la aplicación .											
2	El administrador introduce su nombre de usuario, su contraseña y pulsa iniciar sesión.											
3	El sistema valida que los datos proporcionados son correctos.											
4	El sistema registra una nueva comunidad y guarda en caché los datos básicos que ha introducido el usuario.											
Postcondición	El sistema carga los datos asociados al administrador y muestra la pantalla principal una vez completado el proceso de iniciar sesión.											
Excepciones	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>3</td> <td>No se completa el proceso de iniciar sesión. Los datos proporcionados no corresponden a ningún administrador existente.</td> </tr> </tbody> </table>		Paso	Acción	3	No se completa el proceso de iniciar sesión. Los datos proporcionados no corresponden a ningún administrador existente.						
Paso	Acción											
3	No se completa el proceso de iniciar sesión. Los datos proporcionados no corresponden a ningún administrador existente.											
Dependencias	Formato de datos guardados en la base de datos, usuario y contraseña cifrada.											
Comentarios	La opción de iniciar sesión se encuentra en la pantalla principal nada más abrir la aplicación.											

Tabla 9 - Iniciar sesión como administrador

UC - 10	Cambiar contraseña como administrador																	
Versión	1.0																	
Descripción	Un administrador de la comunidad podrá cambiar su contraseña.																	
Prioridad	Alta																	
Estabilidad	Alta																	
Precondición	El administrador ha creado una cuenta como miembro de una comunidad de vecinos.																	
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El administrador abre la aplicación e inicia sesión.</td> </tr> <tr> <td>2</td> <td>El administrador despliega el menú de opciones y accede al apartado de ajustes.</td> </tr> <tr> <td>3</td> <td>El sistema muestra los ajustes del sistema.</td> </tr> <tr> <td>4</td> <td>El administrador accede a la sección de cambiar contraseña.</td> </tr> <tr> <td>5</td> <td>El administrador introduce su contraseña actual e introduce dos veces su nueva contraseña y confirma sus modificaciones.</td> </tr> <tr> <td>6</td> <td>El sistema valida la petición solicitada.</td> </tr> <tr> <td>7</td> <td>El sistema actualiza la contraseña del usuario.</td> </tr> </tbody> </table>	Paso	Acción	1	El administrador abre la aplicación e inicia sesión.	2	El administrador despliega el menú de opciones y accede al apartado de ajustes.	3	El sistema muestra los ajustes del sistema.	4	El administrador accede a la sección de cambiar contraseña.	5	El administrador introduce su contraseña actual e introduce dos veces su nueva contraseña y confirma sus modificaciones.	6	El sistema valida la petición solicitada.	7	El sistema actualiza la contraseña del usuario.	
	Paso	Acción																
	1	El administrador abre la aplicación e inicia sesión.																
	2	El administrador despliega el menú de opciones y accede al apartado de ajustes.																
	3	El sistema muestra los ajustes del sistema.																
	4	El administrador accede a la sección de cambiar contraseña.																
	5	El administrador introduce su contraseña actual e introduce dos veces su nueva contraseña y confirma sus modificaciones.																
	6	El sistema valida la petición solicitada.																
7	El sistema actualiza la contraseña del usuario.																	
Postcondición	La nueva contraseña del administrador se ha actualizado. Se guarda dicha contraseña cifrada en la base de datos.																	
Excepciones	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>6</td> <td>No se completa el proceso de actualizar la contraseña. La contraseña introducida coincide con la contraseña actual.</td> </tr> </tbody> </table>	Paso	Acción	6	No se completa el proceso de actualizar la contraseña. La contraseña introducida coincide con la contraseña actual.													
	Paso	Acción																
6	No se completa el proceso de actualizar la contraseña. La contraseña introducida coincide con la contraseña actual.																	
Dependencias	Contraseña almacenada en la base de datos.																	
Comentarios	N/A																	

Tabla 10 - Cambiar contraseña como administrador

UC - 11	Borrar avisos													
Versión	1.0													
Descripción	Un administrador de la comunidad podrá borrar un aviso que se muestra a los miembros de esa comunidad.													
Prioridad	Alta													
Estabilidad	Media													
Precondición	El administrador dispone de una cuenta.													
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El administrador abre la aplicación e inicia sesión.</td> </tr> <tr> <td>2</td> <td>El administrador despliega el menú de opciones y accede al apartado de incidencias.</td> </tr> <tr> <td>3</td> <td>El sistema muestra la interfaz del apartado de avisos.</td> </tr> <tr> <td>4</td> <td>El usuario pulsará el icono de la papelera y confirmará su decisión.</td> </tr> <tr> <td>5</td> <td>El sistema borrará el aviso seleccionado.</td> </tr> </tbody> </table>		Paso	Acción	1	El administrador abre la aplicación e inicia sesión.	2	El administrador despliega el menú de opciones y accede al apartado de incidencias.	3	El sistema muestra la interfaz del apartado de avisos.	4	El usuario pulsará el icono de la papelera y confirmará su decisión.	5	El sistema borrará el aviso seleccionado.
	Paso	Acción												
	1	El administrador abre la aplicación e inicia sesión.												
	2	El administrador despliega el menú de opciones y accede al apartado de incidencias.												
	3	El sistema muestra la interfaz del apartado de avisos.												
	4	El usuario pulsará el icono de la papelera y confirmará su decisión.												
5	El sistema borrará el aviso seleccionado.													
Postcondición	El aviso se eliminará de la base de datos y no se mostrará a los demás miembros de la comunidad de vecinos.													
Excepciones	N/A													
Dependencias	Avisos ya creados.													
Comentarios	N/A													

Tabla 11 - Borrar avisos

UC - 12	Editar avisos													
Versión	1.0													
Descripción	Un administrador de la comunidad podrá editar un aviso que se muestra a los miembros de esa comunidad.													
Prioridad	Media													
Estabilidad	Media													
Precondición	El administrador dispone de una cuenta.													
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El administrador abre la aplicación e inicia sesión.</td> </tr> <tr> <td>2</td> <td>El administrador despliega el menú de opciones y accede al apartado de incidencias.</td> </tr> <tr> <td>3</td> <td>El sistema muestra la interfaz del apartado de avisos.</td> </tr> <tr> <td>4</td> <td>El administrador pulsará el icono del lápiz, edita la información del aviso y confirma su decisión.</td> </tr> <tr> <td>5</td> <td>El sistema actualiza los datos del aviso.</td> </tr> </tbody> </table>		Paso	Acción	1	El administrador abre la aplicación e inicia sesión.	2	El administrador despliega el menú de opciones y accede al apartado de incidencias.	3	El sistema muestra la interfaz del apartado de avisos.	4	El administrador pulsará el icono del lápiz, edita la información del aviso y confirma su decisión.	5	El sistema actualiza los datos del aviso.
Paso	Acción													
1	El administrador abre la aplicación e inicia sesión.													
2	El administrador despliega el menú de opciones y accede al apartado de incidencias.													
3	El sistema muestra la interfaz del apartado de avisos.													
4	El administrador pulsará el icono del lápiz, edita la información del aviso y confirma su decisión.													
5	El sistema actualiza los datos del aviso.													
Postcondición	El aviso queda actualizado en la base de datos.													
Excepciones	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>5</td> <td>No se ha podido actualizar la información. Se han detectado caracteres especiales en el texto introducido.</td> </tr> </tbody> </table>		Paso	Acción	5	No se ha podido actualizar la información. Se han detectado caracteres especiales en el texto introducido.								
Paso	Acción													
5	No se ha podido actualizar la información. Se han detectado caracteres especiales en el texto introducido.													
Dependencias	Información existente del aviso guardada en la base de datos.													
Comentarios	N/A													

Tabla 12 - Editar avisos

UC - 13	Concluir votación													
Versión	1.0													
Descripción	Un administrador de la comunidad podrá concluir las votaciones.													
Prioridad	Media													
Estabilidad	Media													
Precondición	El administrador dispone de una cuenta. Miembros de la comunidad han enviado incidencias desde sus cuentas.													
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El administrador abre la aplicación e inicia sesión.</td> </tr> <tr> <td>2</td> <td>El administrador despliega el menú de opciones y accede al apartado de votación.</td> </tr> <tr> <td>3</td> <td>El sistema muestra la interfaz del apartado de votaciones.</td> </tr> <tr> <td>4</td> <td>El administrador pulsa el botón de cerrar votación.</td> </tr> <tr> <td>5</td> <td>El sistema concluye con éxito la votación.</td> </tr> </tbody> </table>		Paso	Acción	1	El administrador abre la aplicación e inicia sesión.	2	El administrador despliega el menú de opciones y accede al apartado de votación.	3	El sistema muestra la interfaz del apartado de votaciones.	4	El administrador pulsa el botón de cerrar votación.	5	El sistema concluye con éxito la votación.
	Paso	Acción												
	1	El administrador abre la aplicación e inicia sesión.												
	2	El administrador despliega el menú de opciones y accede al apartado de votación.												
	3	El sistema muestra la interfaz del apartado de votaciones.												
	4	El administrador pulsa el botón de cerrar votación.												
	5	El sistema concluye con éxito la votación.												
Postcondición	El administrador finaliza el periodo de votación con éxito.													
Excepciones	N/A													
Dependencias	Cambios en la información almacenada en la base de datos.													
Comentarios	N/A													

Tabla 13 - Concluir votación

UC - 14	Visualizar clave comunidad												
Versión	1.0												
Descripción	Un administrador de la comunidad podrá visualizar el código de la comunidad.												
Prioridad	Alta												
Estabilidad	Alta												
Precondición	El administrador dispone de una cuenta.												
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El administrador abre la aplicación e inicia sesión.</td> </tr> <tr> <td>2</td> <td>El administrador despliega el menú de opciones y accede al apartado de ajustes.</td> </tr> <tr> <td>3</td> <td>El sistema muestra la interfaz del apartado de ajustes.</td> </tr> <tr> <td>4</td> <td>El administrador pulsa sobre la opción “mostrar código comunidad”.</td> </tr> <tr> <td>5</td> <td>El sistema muestra un mensaje con el código de la comunidad de vecinos.</td> </tr> </tbody> </table>	Paso	Acción	1	El administrador abre la aplicación e inicia sesión.	2	El administrador despliega el menú de opciones y accede al apartado de ajustes.	3	El sistema muestra la interfaz del apartado de ajustes.	4	El administrador pulsa sobre la opción “mostrar código comunidad”.	5	El sistema muestra un mensaje con el código de la comunidad de vecinos.
Paso	Acción												
1	El administrador abre la aplicación e inicia sesión.												
2	El administrador despliega el menú de opciones y accede al apartado de ajustes.												
3	El sistema muestra la interfaz del apartado de ajustes.												
4	El administrador pulsa sobre la opción “mostrar código comunidad”.												
5	El sistema muestra un mensaje con el código de la comunidad de vecinos.												
Postcondición	Se muestra con éxito el mensaje al administrador.												
Excepciones	N/A												
Dependencias	Código de la comunidad generado al registrar la comunidad de vecinos.												
Comentarios	El mensaje se muestra como una ventana emergente en pantalla, este código lo emplearán los miembros de la comunidad al iniciar sesión por primera vez.												

Tabla 14 - Visualizar clave comunidad

UC - 15	Añadir usuario a la comunidad															
Versión	1.0															
Descripción	Un administrador de la comunidad podrá añadir un nuevo usuario a la comunidad.															
Prioridad	Alta															
Estabilidad	Media															
Precondición	El administrador dispone de una cuenta.															
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El administrador abre la aplicación e inicia sesión.</td> </tr> <tr> <td>2</td> <td>El administrador despliega el menú de opciones y accede al apartado de gestionar usuarios.</td> </tr> <tr> <td>3</td> <td>El sistema muestra la interfaz del apartado de gestión de usuarios.</td> </tr> <tr> <td>4</td> <td>El administrador pulsa sobre la opción “añadir usuario”.</td> </tr> <tr> <td>5</td> <td>El administrador introduce los campos solicitados y pulsa “añadir usuario”</td> </tr> <tr> <td>6</td> <td>El sistema valida los datos y añade un nuevo usuario a la comunidad.</td> </tr> </tbody> </table>		Paso	Acción	1	El administrador abre la aplicación e inicia sesión.	2	El administrador despliega el menú de opciones y accede al apartado de gestionar usuarios.	3	El sistema muestra la interfaz del apartado de gestión de usuarios.	4	El administrador pulsa sobre la opción “añadir usuario”.	5	El administrador introduce los campos solicitados y pulsa “añadir usuario”	6	El sistema valida los datos y añade un nuevo usuario a la comunidad.
Paso	Acción															
1	El administrador abre la aplicación e inicia sesión.															
2	El administrador despliega el menú de opciones y accede al apartado de gestionar usuarios.															
3	El sistema muestra la interfaz del apartado de gestión de usuarios.															
4	El administrador pulsa sobre la opción “añadir usuario”.															
5	El administrador introduce los campos solicitados y pulsa “añadir usuario”															
6	El sistema valida los datos y añade un nuevo usuario a la comunidad.															
Postcondición	Se añade un nuevo usuario a la comunidad de vecinos.															
Excepciones	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>6</td> <td>Se encuentran campos sin rellenar.</td> </tr> <tr> <td>6</td> <td>El número de documento de identidad no tiene un formato correcto.</td> </tr> <tr> <td>6</td> <td>Ya existe un usuario con ese DNI</td> </tr> </tbody> </table>		Paso	Acción	6	Se encuentran campos sin rellenar.	6	El número de documento de identidad no tiene un formato correcto.	6	Ya existe un usuario con ese DNI						
Paso	Acción															
6	Se encuentran campos sin rellenar.															
6	El número de documento de identidad no tiene un formato correcto.															
6	Ya existe un usuario con ese DNI															
Dependencias	Códigos de identificación de otros usuarios. No pueden existir dos usuarios con el mismos DNI.															
Comentarios	N/A															

Tabla 15 - Añadir usuario a la comunidad

UC - 16	Editar usuario a la comunidad															
Versión	1.0															
Descripción	Un administrador de la comunidad podrá editar los datos de un usuario existente de la comunidad.															
Prioridad	Alta															
Estabilidad	Media															
Precondición	El administrador dispone de una cuenta.															
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El administrador abre la aplicación e inicia sesión.</td> </tr> <tr> <td>2</td> <td>El administrador despliega el menú de opciones y accede al apartado de gestionar usuarios.</td> </tr> <tr> <td>3</td> <td>El sistema muestra la interfaz del apartado de gestión de usuarios.</td> </tr> <tr> <td>4</td> <td>El administrador pulsa sobre el icono de un lápiz.</td> </tr> <tr> <td>5</td> <td>El administrador actualiza los campos solicitados y pulsa "actualizar usuario".</td> </tr> <tr> <td>6</td> <td>El sistema valida los datos y añade un nuevo usuario a la comunidad.</td> </tr> </tbody> </table>		Paso	Acción	1	El administrador abre la aplicación e inicia sesión.	2	El administrador despliega el menú de opciones y accede al apartado de gestionar usuarios.	3	El sistema muestra la interfaz del apartado de gestión de usuarios.	4	El administrador pulsa sobre el icono de un lápiz.	5	El administrador actualiza los campos solicitados y pulsa "actualizar usuario".	6	El sistema valida los datos y añade un nuevo usuario a la comunidad.
	Paso	Acción														
	1	El administrador abre la aplicación e inicia sesión.														
	2	El administrador despliega el menú de opciones y accede al apartado de gestionar usuarios.														
	3	El sistema muestra la interfaz del apartado de gestión de usuarios.														
	4	El administrador pulsa sobre el icono de un lápiz.														
	5	El administrador actualiza los campos solicitados y pulsa "actualizar usuario".														
	6	El sistema valida los datos y añade un nuevo usuario a la comunidad.														
Postcondición	Se modifican los datos del usuario de la comunidad de vecinos.															
Excepciones	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>6</td> <td>Se encuentran campos sin rellenar.</td> </tr> <tr> <td>6</td> <td>El número de documento de identidad no tiene un formato</td> </tr> <tr> <td>6</td> <td>Ya existe un usuario con ese DNI</td> </tr> </tbody> </table>		Paso	Acción	6	Se encuentran campos sin rellenar.	6	El número de documento de identidad no tiene un formato	6	Ya existe un usuario con ese DNI						
	Paso	Acción														
	6	Se encuentran campos sin rellenar.														
	6	El número de documento de identidad no tiene un formato														
6	Ya existe un usuario con ese DNI															
Dependencias	Códigos de identificación de otros usuarios. No pueden existir dos usuarios con el mismos DNI.															
Comentarios	No es posible modificar el nombre de usuario, tan solo puede cambiarse el identificador y el localizador.															

Tabla 16 - Editar usuario a la comunidad

UC - 17	Eliminar usuario a la comunidad													
Versión	1.0													
Descripción	Un administrador de la comunidad podrá eliminar un usuario existente de la comunidad.													
Prioridad	Alta													
Estabilidad	Media													
Precondición	El administrador dispone de una cuenta.													
Secuencia normal	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El administrador abre la aplicación e inicia sesión.</td> </tr> <tr> <td>2</td> <td>El administrador despliega el menú de opciones y accede al apartado de gestionar usuarios.</td> </tr> <tr> <td>3</td> <td>El sistema muestra la interfaz del apartado de gestión de usuarios.</td> </tr> <tr> <td>4</td> <td>El administrador pulsa sobre el icono de una papelera y verifica la ventana de diálogo emergente.</td> </tr> <tr> <td>5</td> <td>El sistema elimina el usuario seleccionado de la comunidad de vecinos.</td> </tr> </tbody> </table>		Paso	Acción	1	El administrador abre la aplicación e inicia sesión.	2	El administrador despliega el menú de opciones y accede al apartado de gestionar usuarios.	3	El sistema muestra la interfaz del apartado de gestión de usuarios.	4	El administrador pulsa sobre el icono de una papelera y verifica la ventana de diálogo emergente.	5	El sistema elimina el usuario seleccionado de la comunidad de vecinos.
Paso	Acción													
1	El administrador abre la aplicación e inicia sesión.													
2	El administrador despliega el menú de opciones y accede al apartado de gestionar usuarios.													
3	El sistema muestra la interfaz del apartado de gestión de usuarios.													
4	El administrador pulsa sobre el icono de una papelera y verifica la ventana de diálogo emergente.													
5	El sistema elimina el usuario seleccionado de la comunidad de vecinos.													
Postcondición	Se muestra con éxito el mensaje al administrador.													
Excepciones	<table border="1"> <tbody> <tr> <td>4</td> <td>El usuario rechaza la ventana emergente, cancelando el proceso de eliminar un usuario.</td> </tr> </tbody> </table>		4	El usuario rechaza la ventana emergente, cancelando el proceso de eliminar un usuario.										
4	El usuario rechaza la ventana emergente, cancelando el proceso de eliminar un usuario.													
Dependencias	N/A													
Comentarios	Se elimina con éxito el usuario de la base de datos, también se borran la participación del usuario dentro de la aplicación, esto incluye cosas como las incidencias reportadas.													

Tabla 17 - Eliminar usuario a la comunidad

### 3.3 Diseño de la aplicación

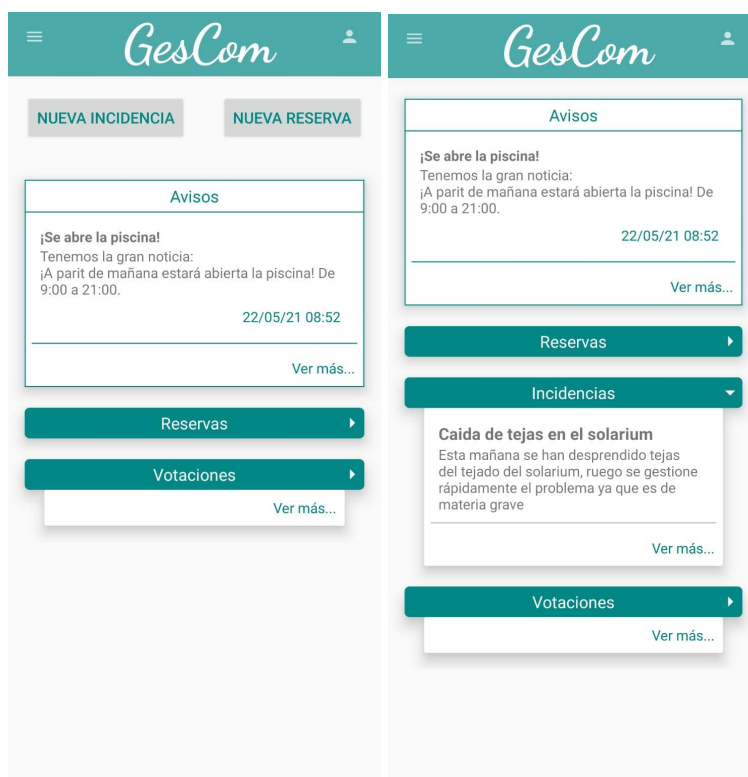
En este apartado se expone mediante una serie de diagramas de clases y de interacción, la estructura de GesCom a nivel de código. Cada subapartado consistirá en la descripción y exposición de la estructura de las funcionalidades principales de la aplicación. Una explicación más detallada se puede encontrar en el Anexo III.

#### 3.3.1 DashBoard

Se trata de la actividad principal de la aplicación, en ella se muestran diferentes atajos según el rol del usuario que ha iniciado la sesión. Se muestran también varios listados de las principales funcionalidades: avisos, incidencias, votaciones y reservas.

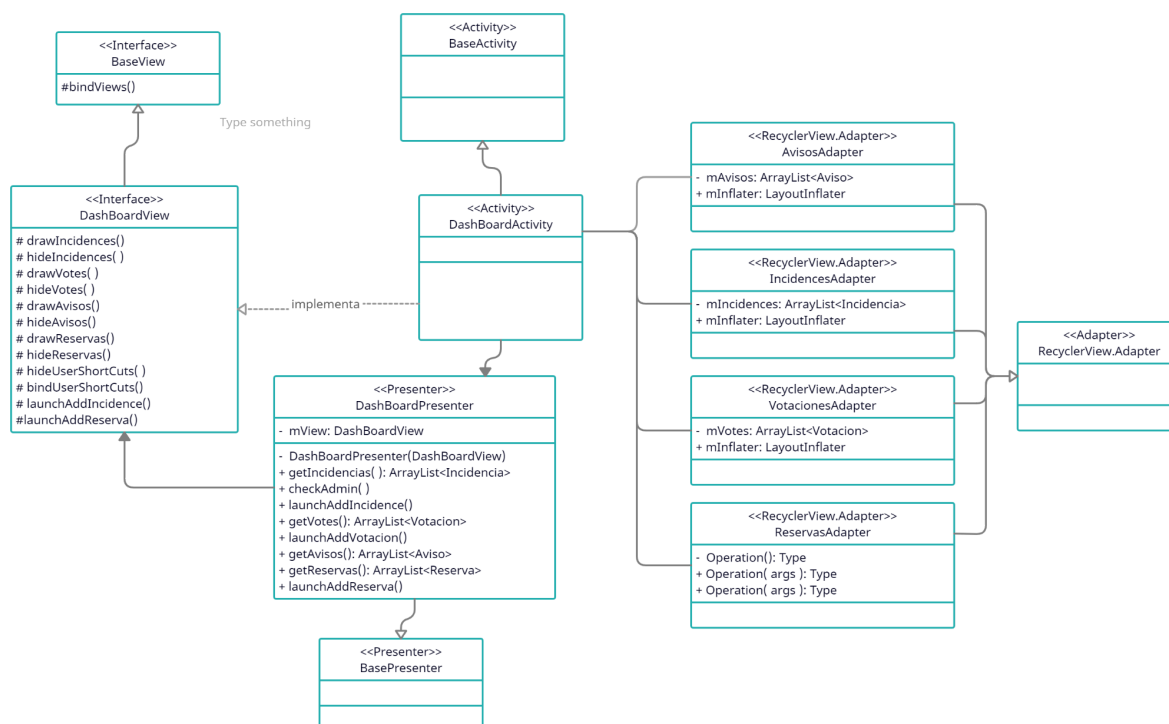
En caso de haber iniciado sesión un usuario administrador, los atajos que se muestran son los de crear aviso, visualizar incidencias, acceder a las votaciones actuales para gestionarlas y visualizar los ajustes.

Sin embargo, si el usuario no tiene el rol de administrador, entonces se muestran atajos para reportar nuevas incidencias o realizar nuevas reservas en las diferentes instalaciones de la comunidad. Estas diferencias se pueden apreciar en la Figura 8.



[Figura 8] - DashBoard

La estructura interna y las interacciones entre diferentes clases que se dan a la hora de llevar a cabo una acción en esta pantalla se pueden apreciar en la Figura 9, que presenta un diagrama de clases.

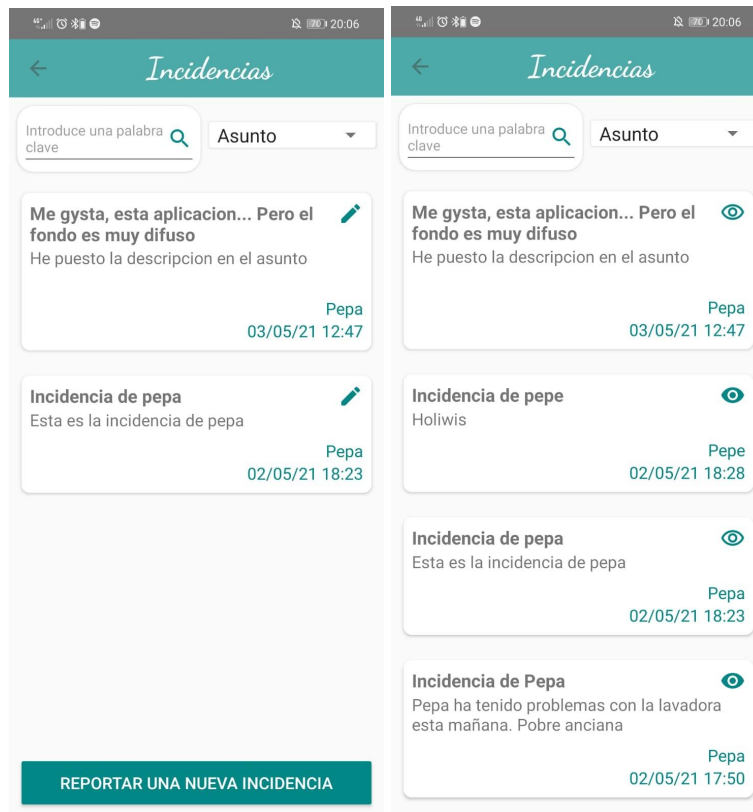


[Figura 9] - Diagrama de clases Dashboard

### 3.3.2 Incidencias

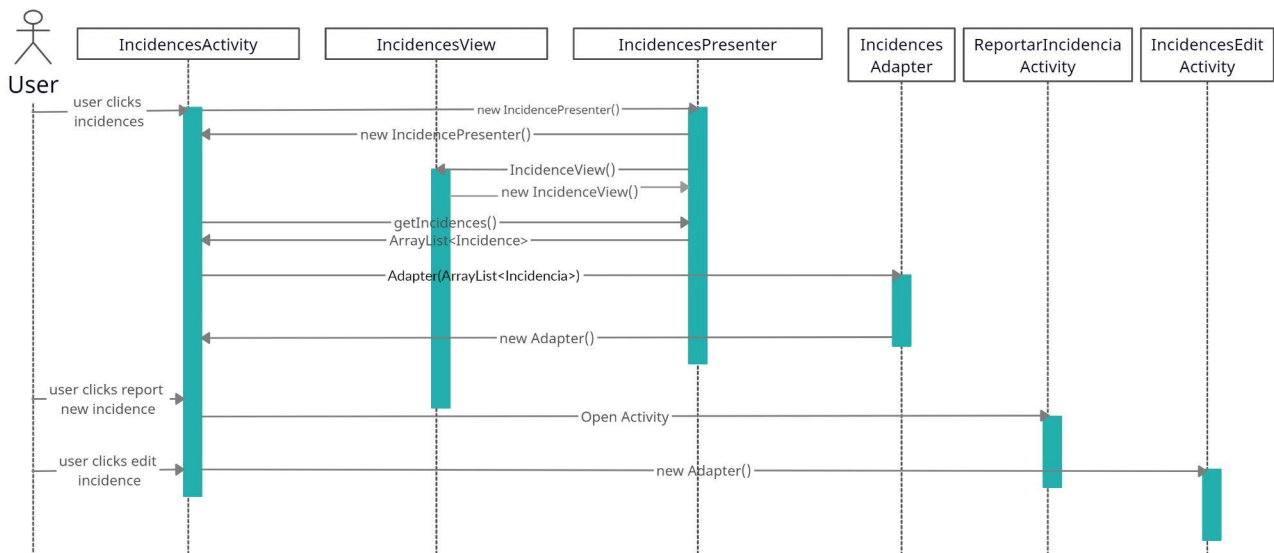
Esta funcionalidad engloba los casos de uso de reportar incidencia, ver incidencias, editar incidencias y filtrar incidencias. En esta actividad el usuario administrador puede ver todas las incidencias ordenadas por proximidad de fecha que le han reportado los vecinos de su comunidad. Por otra parte el usuario normal, puede reportar una nueva incidencia o editar las incidencias anteriormente reportadas, las cuales se mostrarán editadas para el administrador.

Para facilitar la gestión de las incidencias por parte de ambos usuarios, se ha implementado la funcionalidad de filtrar las incidencias, y al administrador se le ha proporcionado una forma de marcar las incidencias que ya han sido gestionadas, como se puede ver en la Figura 10.



[Figura 10] - Actividad de incidencias del administrador y del usuario

Para un mejor entendimiento de las funcionalidades proporcionadas por esta actividad, se presenta a continuación un diagrama de interacción en la Figura 11.



[Figura 11] - Diagrama de interacción actividad de Incidencias

### 3.3.3 Votaciones

Dentro de este apartado encontramos todo lo relacionado con la gestión de las votaciones por parte del administrador, como por ejemplo la creación o cierre de las mismas. Los usuarios sin rol de administrador pueden enviar sus votos a las votaciones y consultar los resultados de las últimas votaciones, junto con su asunto y descripción.

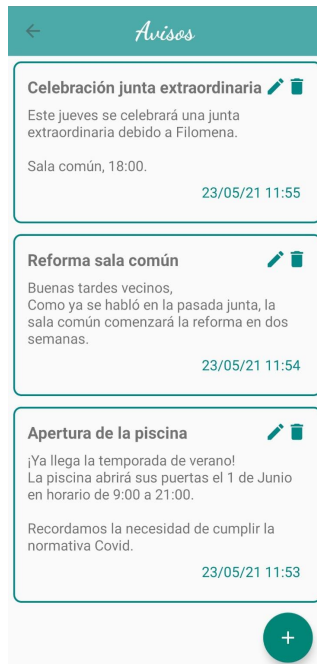
Para enviar su voto los usuarios deberán seleccionar el botón que se observa en la Figura 12. Una vez seleccionado, un diálogo de alerta les dará a elegir entre votar a favor o en contra. Tras haber enviado su voto, los usuarios no podrán cambiarlo o votar una segunda vez.



[Figura 12] - Vista de votaciones para administrador y vecino

### 3.3.4 Avisos

Este apartado engloba todas aquellas funcionalidades relacionadas con los avisos. En el caso de un usuario administrador, este podrá crear nuevos avisos, editarlos o eliminarlos, además de poder visualizar todos los avisos ordenados por fecha de publicación. Esta última opción, tal como se muestra en la Figura 14, es la que puede llevar a cabo un usuario vecino.



[Figura 13] - Pantalla de avisos de usuario administrador

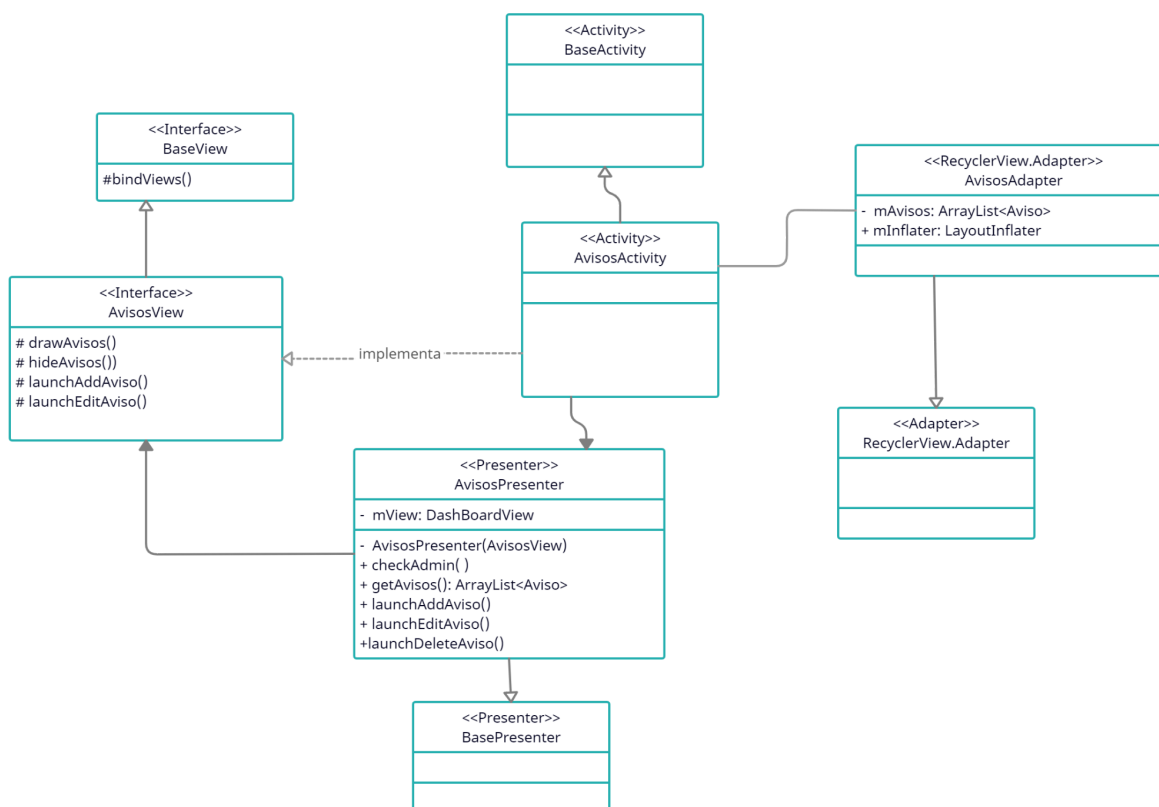
En el caso de esta funcionalidad, a continuación, en la Figura 13, mostramos una captura de pantalla de la actividad de avisos para un usuario administrador, ya que en esta actividad se pueden visualizar todos aquellos casos de uso que se pueden llevar a cabo.



[Figura 14] - Pantalla de avisos usuario vecino

Como se puede observar en la Figura 15, la actividad de avisos sigue el patrón MVP (Modelo Vista-Presentador). Al igual que el resto de vistas

y presentadores extienden de unas clases base, lo que facilita que si en un futuro, todas las clases que extienden de cada una de las bases necesiten una misma funcionalidad o método, sería muy sencillo incrementar la funcionalidad en todas las clases simplemente añadiendo a la clase base.



[Figura 15] - Diagrama de clases de aviso

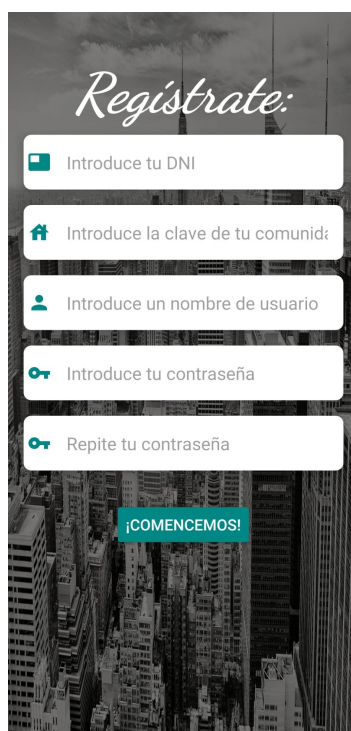
### 3.3.5 Registro de usuario

Una de las funcionalidades de la que la gran mayoría de usuarios harán uso es la de registro de usuario, ya que todo usuario que no tenga el rol de administrador, deberá registrarse como usuario.

Para llevar a cabo esta funcionalidad, y por tanto el caso de uso de registro de usuario, el usuario deberá pertenecer a una comunidad previamente registrada por el administrador de la misma.

Si el usuario cumple la anterior condición, entonces el administrador debe ponerse en contacto con el susodicho para hacerle llegar la clave aleatoriamente generada, que permite comprobar que un usuario pertenece a esa comunidad.

Tras rellenar la batería de campos que se presentan en la Figura 16, la aplicación realiza una serie de validaciones. Entre ellas encontramos la comprobación de que para la comunidad cuya clave es igual a la introducida por el usuario, contiene un usuario anteriormente registrado por el administrador. Otra validación que lleva a cabo el Presentador es comprobar que no existe otro usuario con el nombre de usuario introducido.

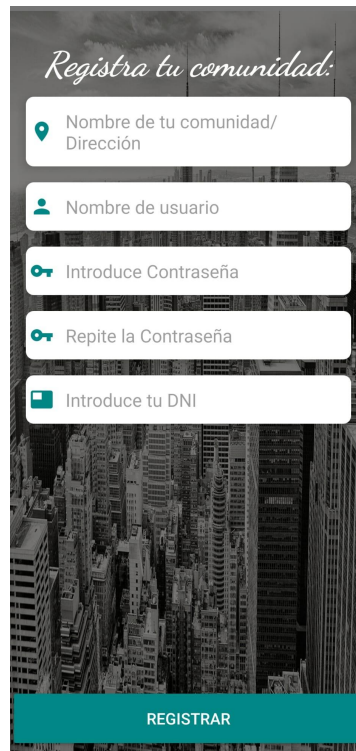
El formulario de registro de usuario se muestra sobre un fondo de una ciudad. El título "Regístrate:" está en una fuente cursiva blanca. Hay cinco campos de entrada con iconos a la izquierda: un cuadrado para el DNI, una casa para la clave de la comunidad, una persona para el nombre de usuario, una llave para la contraseña y otra llave para repetir la contraseña. Debajo de los campos hay un botón verde con el texto "¡COMENCEMOS!".

[Figura 16] - Formulario de registro de usuario

### 3.3.6 Registro de una comunidad

Esta funcionalidad es una de las más relevantes de la aplicación, ya que es necesario llevarla a cabo con el fin de poder hacer uso del resto de las funcionalidades que presenta nuestra aplicación.

Para obtener un resultado exitoso a la hora de registrar su comunidad, el administrador debe rellenar los campos del formulario que se muestran en la Figura 17. Estos campos deben cumplir una serie de validaciones, como por ejemplo: el DNI debe tener ocho caracteres numéricos y una letra, ambas contraseñas deben coincidir y no se permitirá el uso de caracteres especiales.

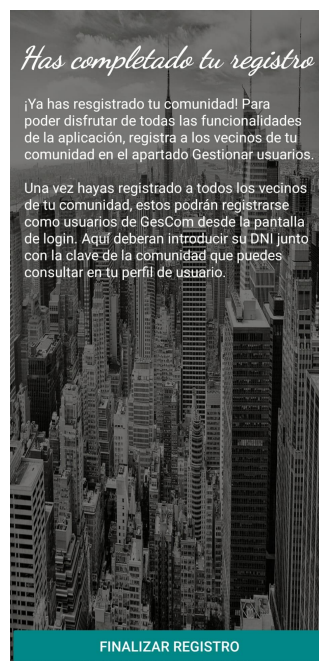


The image shows a mobile application screen for registering a community. At the top, the text "Registra tu comunidad." is written in a cursive font. Below this, there are five white input fields with rounded corners, each containing a small icon and a label: a location pin icon for "Nombre de tu comunidad/ Dirección", a person icon for "Nombre de usuario", a key icon for "Introduce Contraseña", another key icon for "Repite la Contraseña", and a document icon for "Introduce tu DNI". At the bottom of the screen, there is a teal button with the word "REGISTRAR" in white capital letters. The background of the screen is a grayscale image of a dense city skyline.

[Figura 17] - Formulario de registro de la comunidad

Una vez que el usuario rellena todos los campos y pulsa el botón de registrar, se le dirige a una pantalla, mostrada en la Figura 18, que informa de cómo debe añadir nuevos usuarios a la comunidad.

Esta funcionalidad es explicada con mayor profundidad en el apartado de Gestión de usuarios.



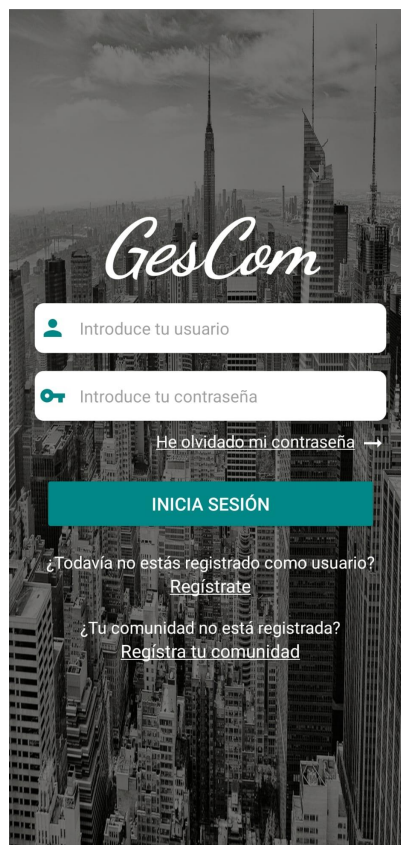
[Figura 18] - Pantalla de información de registro

### 3.3.7 Login

Se trata de la segunda pantalla que ve el usuario al abrir la aplicación. En esta pantalla el usuario tiene las opciones de iniciar sesión, registrarse como usuario, registrar una nueva comunidad o modificar su contraseña actual.

En el caso de iniciar sesión, el usuario debe rellenar su usuario y contraseña en los campos del formulario y presionar el botón de aceptar. En caso de que los datos introducidos pasen las diferentes validaciones, el usuario será reconducido al dashboard de nuestra aplicación.

Como se ve en la Figura 19, hemos optado por un diseño bastante sencillo, que hereda el fondo del splash de la aplicación.

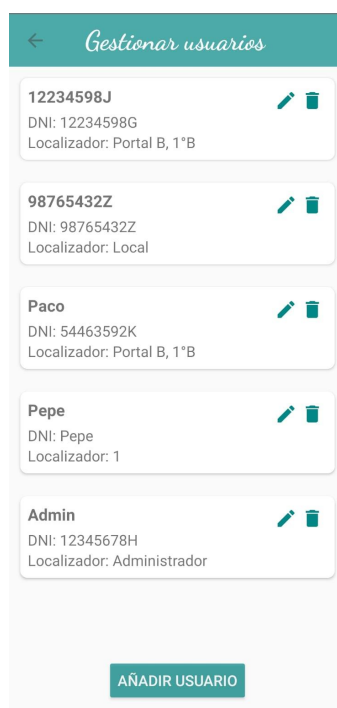


[Figura 19] - Pantalla de inicio de sesión

### 3.3.8 Gestión de usuarios

Esta funcionalidad es exclusiva para usuarios administradores, ya que solo ellos deben poder gestionar los usuarios de la comunidad. La “Gestión de usuarios” incluye los casos de uso de añadir usuarios a la comunidad, editarlos o eliminarlos.

El administrador de una comunidad accede a esta funcionalidad desde el menú principal de la aplicación. La primera pantalla que se le muestra, Figura 20, le permite visualizar todos los usuarios registrados en esa comunidad.

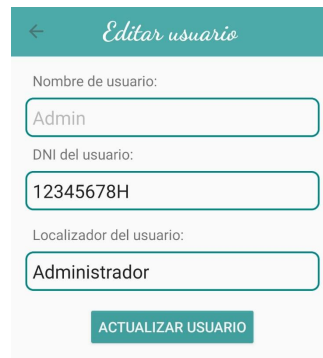


[Figura 20] - Visualización de los usuarios de la comunidad

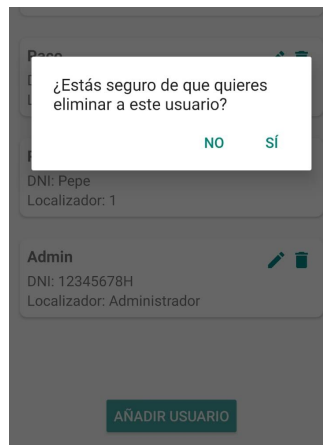
Una vez que el usuario ha accedido a esta pantalla, podrá realizar acciones como añadir un nuevo usuario, editar un usuario existente o eliminarlo, tal y como se muestra en las Figuras 21, 22 y 23 respectivamente. En el caso de editar un usuario, el administrador únicamente podrá modificar el DNI y el localizador de dicho vecino.



[Figura 21] - Añadir un nuevo usuario a la comunidad



[Figura 22] - Editar un usuario de la comunidad



[Figura 23] - Eliminar un usuario de la comunidad

### 3.4 Modelo de datos

Con este modelo de datos nuestra intención es describir la estructura de la base de datos implementada, especificando el tipo de datos y la forma en la que relacionamos dichos datos. Además comentaremos las condiciones de integridad impuestas a la estructura para garantizar una representación más exacta.

### 3.4.1 Modelo entidad-relación

Debido a la naturaleza de nuestra base de datos y el objetivo de este capítulo, nos hemos decantado por el uso de un modelo de datos conceptuales orientado a hacer una representación objetiva y clara de aquellos elementos que forman parte de la aplicación y las posibles relaciones entre los mismos.

En este modelo apreciamos elementos básicos, denominados entidades, que pueden ir acompañados de atributos y relaciones. Las entidades específicas de GesCom y sus atributos se pueden contemplar en la Figura 24.

El sistema guarda información sobre la comunidad de vecinos del usuario que ha iniciado sesión además de guardar información sobre dicho usuario. Los usuarios se identifican por un identificador, pertenecen a una sola comunidad y viven en la vivienda especificada por el localizador. Además el usuario administrador debe introducir los DNIs de los miembros de la comunidad, y a su vez estos eligen su nombre de usuario y contraseña.

Una comunidad está formada por muchos usuarios que pertenecen a ella, y cada comunidad tiene un solo usuario administrador, identificado por el identificador de usuario. Por último las comunidades contienen una clave alfanumérica que permite a los usuarios registrarse en esa misma comunidad.

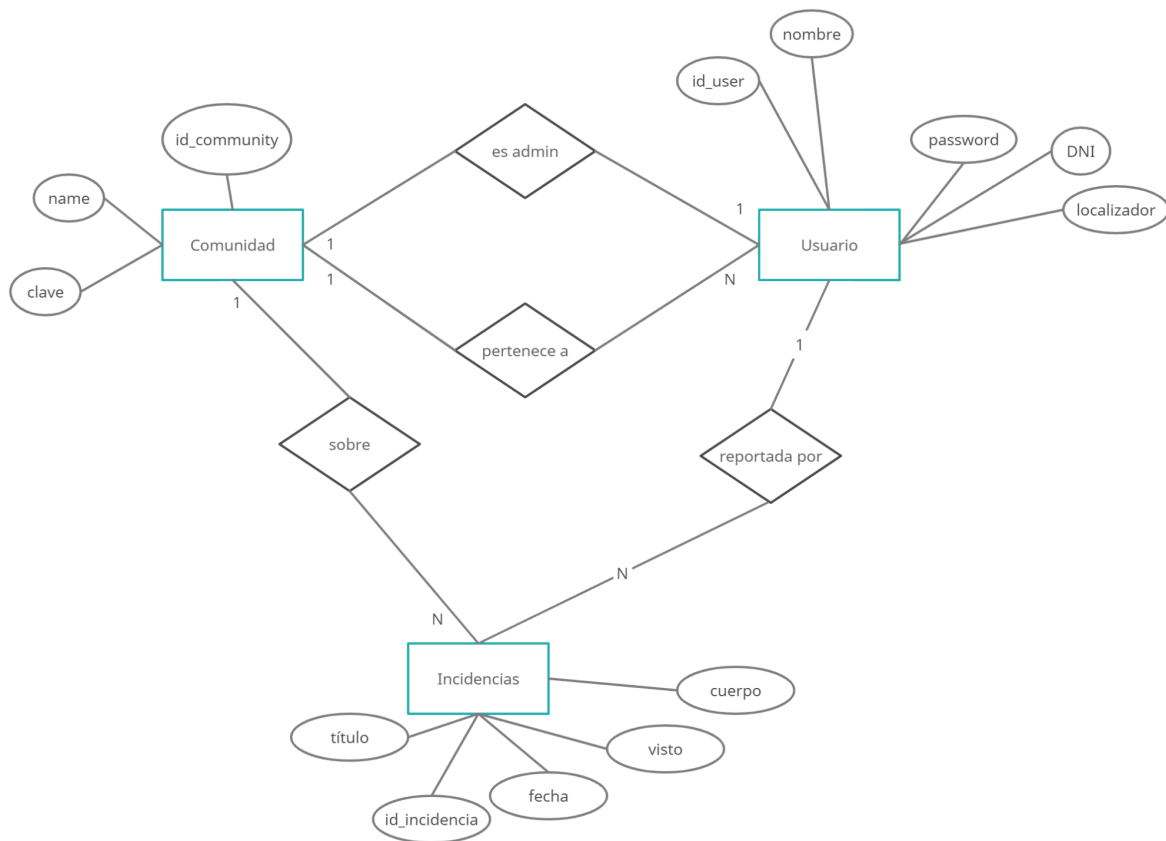
Los usuarios pueden reportar incidencias al usuario administrador, las cuales se guardan con un identificador único. Además se almacena, tal y cómo se indica en la Figura 24 el identificador del usuario que reporta la incidencia, la fecha y hora, el título y el cuerpo de la incidencia, el identificador de la comunidad a la que pertenece el usuario que reporta y por último un booleano que permite saber si el administrador ha gestionado o no esa incidencia.

Como en la mayoría de comunidades de vecinos, el administrador puede convocar votaciones, en este caso virtuales. Por lo tanto se guardan en nuestra base de datos con su título y descripción, el identificador de la comunidad que ha sido convocada para votar. Se guardan también los votos a favor y en contra y un booleano que nos permite saber si la votación ha sido cerrada por el administrador o sigue abierta para que voten los vecinos.

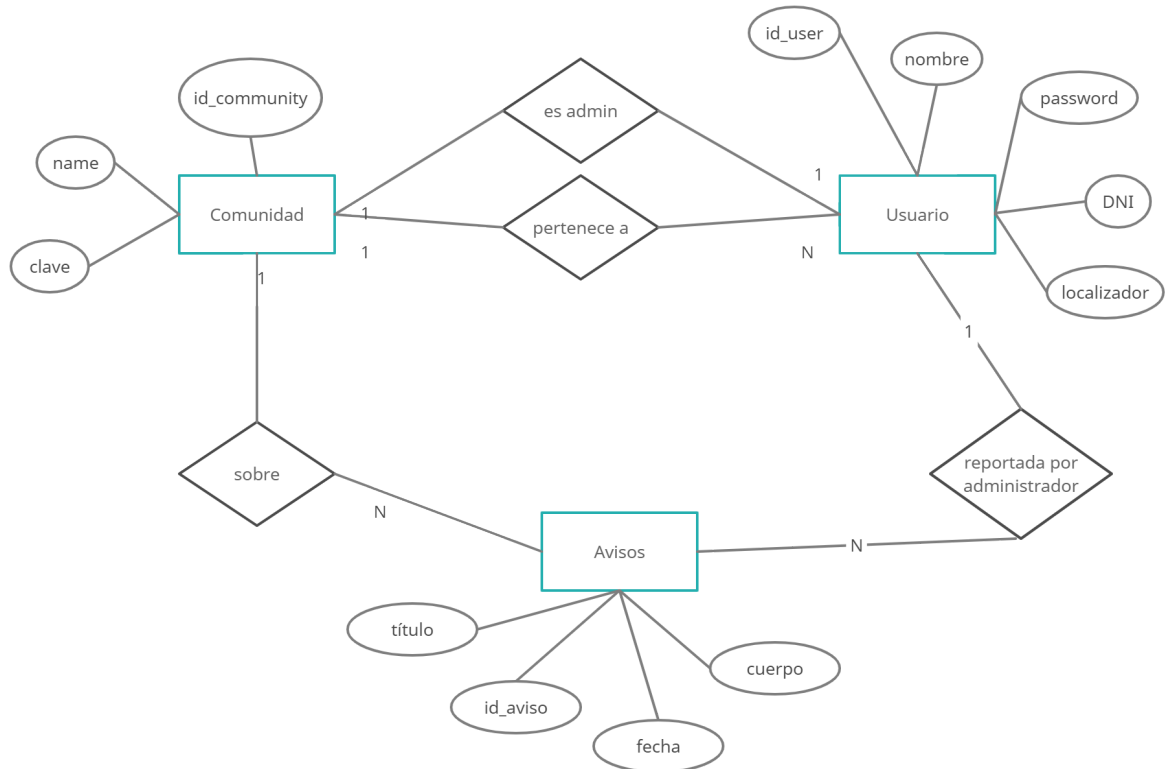
Debido a la necesidad de controlar que un usuario solo envíe un voto a cada votación, se realiza un registro que relaciona al usuario con la votación a la que ha entregado su voto.

El administrador de cada comunidad puede crear nuevos avisos, visibles para todos los usuarios, por ello se almacenan el título y la descripción del aviso, el identificador de la comunidad asociada y la fecha y hora de registro del aviso.

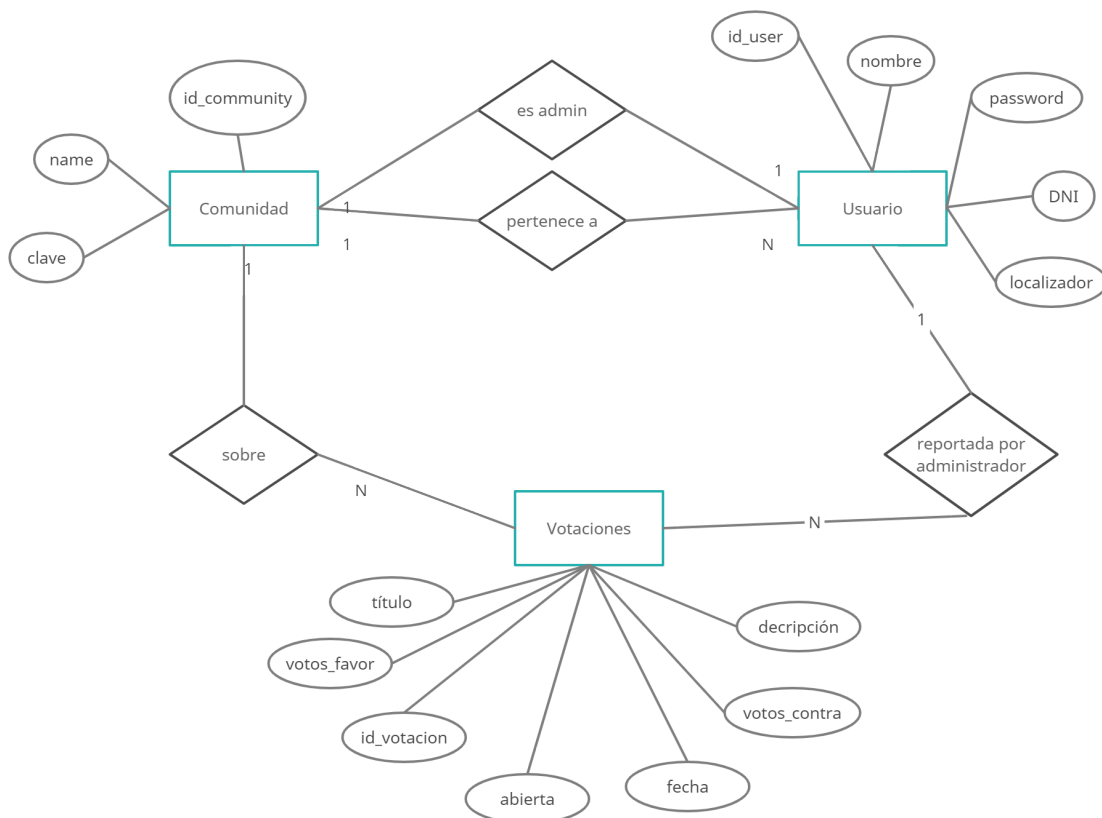
Debido al uso de un modelo de datos conceptuales, consideramos que la mejor manera de representar y exponer la estructura de nuestra base de datos es presentando las siguientes Figuras (24, 25 y 26). Estas Figuras muestran los modelos entidad-relación seguidos durante la implementación y uso de la base de datos, modelo actualmente empleado en la gran mayoría de técnicas de ingeniería del software [44].



[Figura 24] - Modelo Entidad-Relación Incidencias



[Figura 25] - Modelo Entidad-Relación Avisos



[Figura 26] - Modelo Entidad-Relación Votaciones

### 3.4.2 Implementación de la BBDD

El modelo entidad relación descrito en el apartado anterior ha sido implementado en una base de datos SQL. A continuación se describen el total de tablas implementadas y las posibles relaciones entre ellas.

#### Tabla de usuarios

Contiene toda la información necesaria para la correcta identificación de cada usuario y está formada por las siguientes columnas:

- id: Identificador para cada usuario de la aplicación.
- username: Nombre con el que se dan a conocer en la aplicación.
- community: Identificador de la comunidad a la que pertenece.
- password: Contraseña para su cuenta de usuario.
- dni: Documento nacional de identidad del usuario.
- localizer: Representa la vivienda en la que habita cada usuario. En el caso de ser administrador el localizador es “Administrador”.

#### Tabla de comunidades

Esta tabla contiene información básica sobre cada comunidad. Alberga las siguientes columnas:

- id: Identifica cada comunidad registrada en la aplicación.
- nombre: Representa el nombre por el cual se conoce la comunidad.
- id\_admin: El identificador del usuario administrador de la comunidad.
- key: La clave aleatoriamente generada para el registro de los usuarios.

#### Tabla de incidencias

Contiene la información necesaria para relacionar correctamente cada incidencia con la comunidad asociada o el usuario que reportó la incidencia:

- id: El identificador de cada incidencia reportada.
- título: Breve frase que resume la naturaleza de la incidencia.
- cuerpo: Descripción detallada de la naturaleza de la incidencia.
- id\_comunidad: Identificador de la comunidad asociada a la incidencia.
- id\_usuario: Identificador del usuario que reporta la incidencia.

- nombre\_de\_usuario: Nombre del usuario que reporta la incidencia.
- fecha: Fecha en la que se reporta la incidencia.
- hora: Hora en la que se reporta la incidencia.
- visto: Booleano que indica si la incidencia ha sido gestionada por el administrador.

#### Tabla de avisos

Esta tabla presenta menor número de columnas ya que solo el administrador de cada comunidad puede crear un nuevo aviso, con lo cual nos vale con tener información para identificar la comunidad asociada:

- id: El identificador de cada aviso creado.
- título: Breve frase que resume la naturaleza del aviso.
- cuerpo: Descripción más detallada de la naturaleza del aviso.
- id\_comunidad: Identificador de la comunidad asociada al aviso. Permite la identificación del administrador que creó el aviso.
- fecha: Fecha de creación del aviso.
- hora: Hora de creación del aviso.

#### Tabla de votaciones

Se trata de una de las tablas que más información almacena. Presenta las siguientes columnas:

- id: Identifica cada una de las votaciones registradas en la aplicación.
- título: Breve descripción del motivo de la elección.
- descripción: Explicación más detallada de las opciones a elegir.
- id\_comunidad: Identificador de la comunidad asociada.
- votos\_favor: Número de usuarios que han votado a favor.
- votos\_contra: Número de usuarios que han votado en contra.
- abierta: Booleano que indica si la votación sigue o no admitiendo votos.
- fecha: fecha de publicación de la votación.

#### Tabla de registro de votaciones

La necesidad de esta tabla surge del debido control de los votos de una votación. Los usuarios solo pueden votar una única vez para cada votación.

- id: Identifica cada uno de los registros efectuados.
- id\_votacion: Identificador de la votación abierta asociada al registro.
- id\_usuario: Identificador del usuario que ha enviado su voto.

## 3.5 Arquitectura

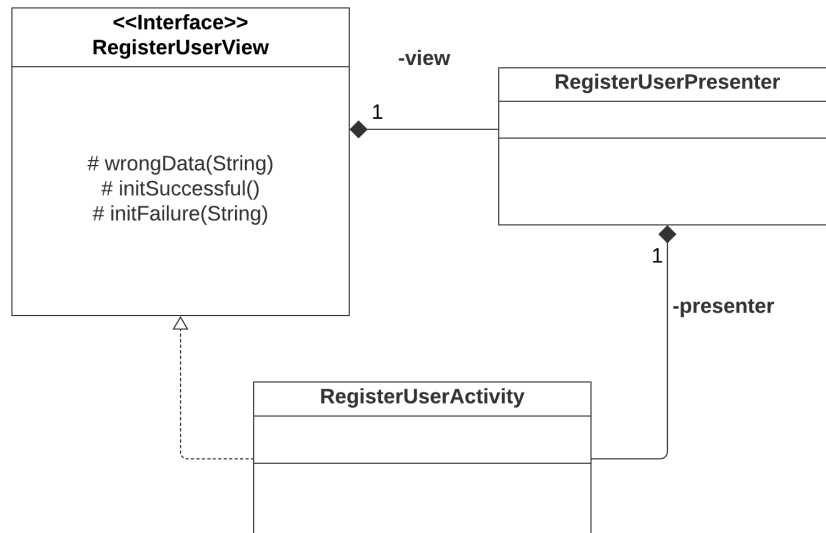
A continuación se describen los patrones y técnicas empleados tanto en el diseño como en el desarrollo de nuestra aplicación con el fin de obtener una aplicación correctamente estructurada. Además se acompaña la explicación técnica de los patrones y diseños, de una explicación de su uso, ya que los usos de estos pueden diferir mucho [45].

### 3.5.1 Modelo MVP

A lo largo de los últimos años, los desarrolladores de aplicaciones se han visto obligados a desarrollar con gran rapidez y bajo unas condiciones cada vez más exigentes en términos de interfaz de usuario [46]. Por lo anteriormente expuesto, el uso del patrón Modelo Vista-Presentador ha crecido, ya que presenta una alternativa de gran eficiencia y poco coste.

El Modelo Vista Presentador (Model View Presenter) es un patrón arquitectónico comúnmente empleado en el desarrollo de aplicaciones Android [47]. Se trata de un patrón derivado del Modelo Vista Controlador, cuyo principal objetivo es separar la lógica de negocio de la aplicación de las vistas (Activities) de la misma [48].

La razón por la cual hemos empleado este patrón es por su gran éxito a la hora de desarrollar interfaces de usuario, ya que en el mismo, la vista no tiene conocimiento alguno de modelo, como se puede observar en la Figura 27 a continuación.



[Figura 27] - Patrón MVP en el registro de usuario.

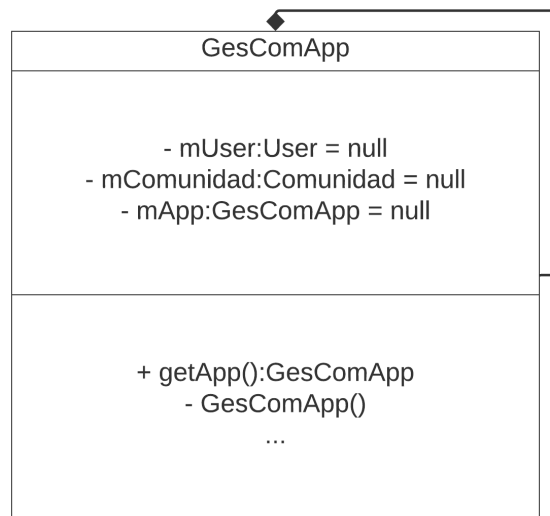
### 3.5.2 Patrón de diseño: Singleton

Los patrones de diseño son comunmente empleados en la creación de aplicaciones de gran calidad. Estos patrones son soluciones generales para problemas durante el desarrollo de software [49].

Este patrón de diseño restringe la instancia de una clase a una sola instancia, por ello se emplea en aquellos casos en los que tan solo una instancia de una clase es necesaria para la correcta implementación de una aplicación[50].

Este patrón se caracteriza por esconder el constructor de la clase y por definir un método final, estático y público, como se observa en la Figura 28 que devuelve la instancia de la clase en caso de haber sido creada con anterioridad. En caso contrario crea una nueva instancia haciendo uso de un constructor privado.

En el caso de nuestra aplicación, hemos hecho uso de este patrón mayoritariamente para aquellas clases que definen objetos empleados en el almacenamiento de información persistente.



[Figura 28] - Singleton empleado para el almacenamiento del estado global

# **Capítulo 4**

## **Interacción con usuarios**

## 4.1 Fases de la interacción

Nuestro fin es desarrollar una app innovadora y que presente nuevas funcionalidades. Para lograr este objetivo hemos comenzado recopilando datos de usuarios potenciales mediante un cuestionario y un estudio de mercado.

Tras el análisis de la información recolectada, nuestra aplicación debe facilitar las actividades que se llevan a cabo de forma rutinaria en las diferentes fincas, como pueden ser recordatorios de las juntas de vecinos, reportar incidencias o irregularidades, reservar áreas comunes y tener las noticias más relevantes en un tablón de anuncios. Todo ello mientras proporcionamos una herramienta sencilla, clara, eficaz y gratuita.

Nuestra última fase de interacción con usuarios comienza con una prueba de la aplicación en un usuario con rol de administrador o de usuario vecino. Tras esta experiencia de usuario, los participantes rellenaron una encuesta implementada en Google Forms que contenía las preguntas del conocido test de usuario “System usability Scale (SUS)” [51], que se encuentran descritas en el Anexo II.

## 4.2 Recolección de datos previos al desarrollo

Para tener una mejor idea de las necesidades que tienen los usuarios a los que va dirigida nuestra aplicación, realizamos una encuesta online, plasmada en el Anexo I. El objetivo de esta primera encuesta es aprender más sobre los usuarios, su situación actual de vivienda, los intereses y desintereses acerca de las funcionalidades de una aplicación móvil para la gestión de comunidades y datos sobre la competencia que exista en este sector del mercado.

Categorizamos las preguntas según los datos que nos parecían más relevantes. Fuera de la aplicación, queríamos saber cuántas personas conviven en una misma residencia, si pertenecen a una comunidad de vecinos y si son propietarios de la vivienda o inquilinos. Para saber más acerca de la competencia de otras compañías, era importante saber si los encuestados ya estaban haciendo uso de una aplicación para gestionar la comunidad de vecinos. En caso afirmativo estábamos interesados en saber de qué aplicación se trata y qué funcionalidades proporciona al usuario. Para aquellos que no hacen uso de una, era importante saber qué gestiones dentro de una comunidad resultaban más frustrante, y qué funcionalidades consideran los usuarios útiles y cuales les parecería menos interesante si estuvieran usando una aplicación móvil.

Se redactó la encuesta haciendo uso de la herramienta Google Form, y fue distribuida mediante la aplicación de mensajería de Whatsapp. Con la finalidad de tener un mayor número de respuestas, el cuestionario creado constaba de un máximo de 10 preguntas, dependiendo de las respuestas elegidas. Antes de analizar los resultados mantuvimos la encuesta activa con un límite de siete días desde la fecha de emisión de la encuesta. La encuesta estuvo activa hasta el 28 de febrero del 2021.

Con los datos obtenidos continuamos con la exposición y análisis de los resultados.

Un 66,9% afirman ser el propietario de la vivienda y el 29,9% viven en un piso de alquiler, con un total del 96,8% entre los dos. Tendremos que considerar el poder especificar la situación de vivienda en la aplicación.

Además el 86,4% viven en una comunidad de vecinos. Con este elevado porcentaje podemos saber que la gran mayoría de las respuestas de los encuestados reflejarán la opinión de posibles futuros clientes.

El 92,5% respondió que no hace uso de ninguna aplicación móvil o web a la hora de gestionar su comunidad de vecinos. Son muchas las personas que no hacen uso de una aplicación o herramienta externa digital.

De aquellas personas que sí utilizan una aplicación el 55,6% utilizaba el sistema operativo android. De las herramientas que les ofrecía a estos usuarios, más de la mitad podían reportar incidencias y ver los recibos de gastos de la comunidad. Ninguna persona respondió que le permitía ver qué vecinos se encontraban en cuarentena y tan solo el 20% podían hacer uso de esta aplicación de forma gratuita.

Como temas más frustrantes y lentos en una comunidad fueron seleccionadas: las reuniones de vecinos (63,8%), las juntas de la comunidad (51,5%), las votaciones de la comunidad (36,9%) y la notificación de incidencias (31,5%). Estas respuestas tuvieron una relación directa con la siguiente pregunta en el que marcaron qué herramientas de una aplicación facilitan la interacción en su comunidad de vecinos. A un 44% le gustaría poder votar sobre temas de la comunidad, a un 57,6% poder reportar incidencias y un 37,6% poder hablar con otros vecinos por chat y un 30,4% poder realizar las reuniones por videollamada.

Otras funcionalidades votadas por más del 25% incluyen recibir notificaciones de vecinos en cuarentena, ver los cargos de incidencias y derramas, ver los recibos de los cargos de la comunidad y que la aplicación sea gratuita.

En las últimas dos preguntas se les pedían a los usuarios funcionalidades adicionales que les parecieran imprescindibles en la aplicación. Un gran número de los encuestados quieren una aplicación sencilla, simple y rápida donde se puedan reportar incidencias, hacer reuniones a través de videollamadas, que se pueda votar y ver las notificaciones recientes un un tablón de anuncios.

Si bien anteriormente un 37,6% votó que quería un chat de vecinos, el 44,9% respondió explícitamente por escrito que no querían ningún tipo de chat ni videollamadas entre vecinos. Con un elevado número de personas eliminaría la sencillez de la aplicación con mensajes constantes o notificaciones innecesarias y molestas.

### 4.3 Análisis de experiencias de usuario

Con el objetivo de analizar de manera objetiva la usabilidad de nuestra aplicación, hemos llevado a cabo unas pruebas con usuarios, seguidas de la respuesta al cuestionario de la Escala de Usabilidad del Sistema por parte de los mismos.

La primera parte de estas experiencias de usuario consiste en que éste lleve a cabo una serie de casos de uso propuestos por el miembro del proyecto. Además de realizar los casos de uso propuestos, los usuarios también han contado con un tiempo de uso libre de la aplicación.

Durante esta primera fase, los usuarios han tomado el tiempo que han necesitado para llevar a cabo los casos de uso.

Tras la finalización de esta parte, todos los usuarios han contestado a un cuestionario en Google Forms, en el que han respondido a las preguntas del test de la Escala de Usabilidad del Sistema que se presentan en el Anexo II.

Las respuestas a estas preguntas eran de tipo numérico, como se observa en la Figura 29, significando el número 1 que el usuario estaba en desacuerdo con lo que el enunciado expresa, y 5 lo contrario. Siendo los valores intermedios proporcionales. A continuación procedemos al análisis de las respuestas del cuestionario.

Encuentro GesCom innecesariamente compleja (1 - Muy en desacuerdo, 5 - Totalmente de acuerdo) \*

1

2

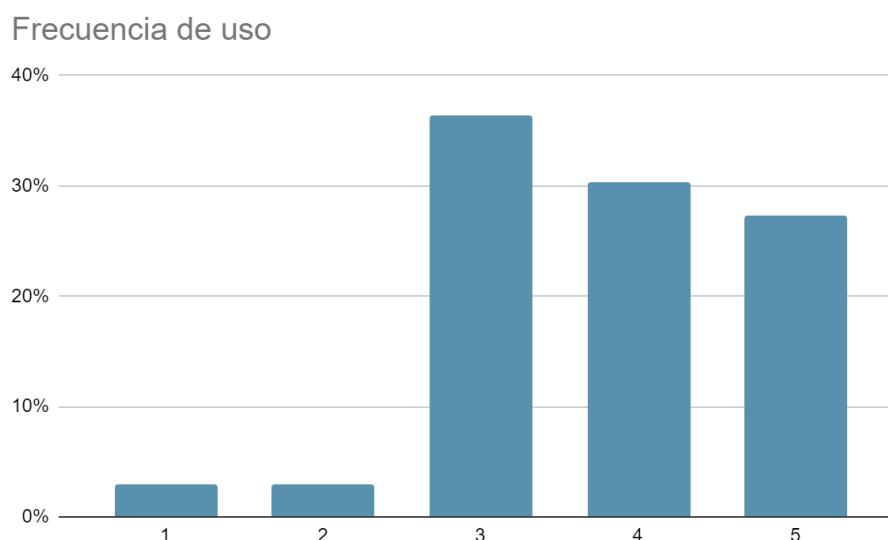
3

4

5

[Figura 29] - Ejemplo de pregunta en cuestionario SUS

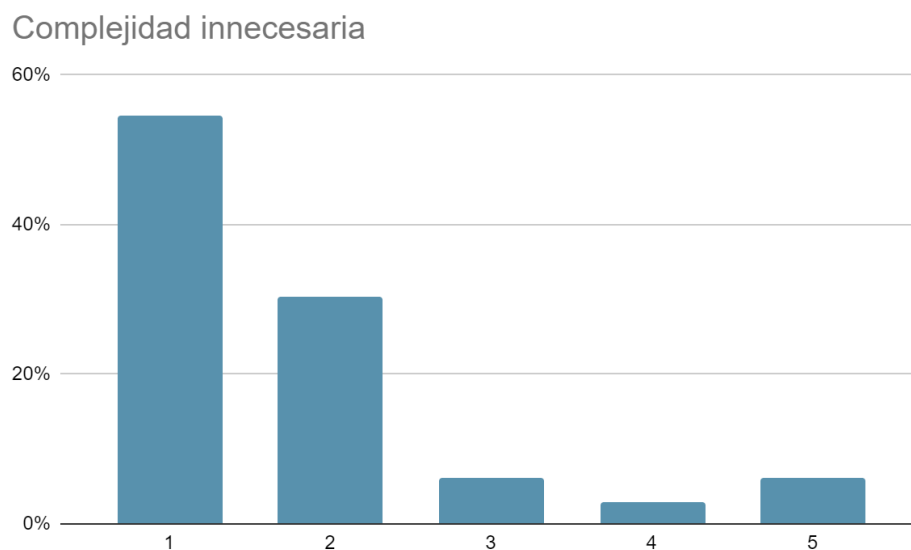
La primera pregunta consistía en analizar cuán frecuentemente usarían los usuarios GesCom. Contemplando en la Figura 30 los resultados obtenidos, vemos que un 27,3% de los encuestados usarían frecuentemente la aplicación, seguidos de un 30,3%, que usaría la aplicación menos frecuentemente. Teniendo en cuenta las funcionalidades que presenta nuestra aplicación, contemplamos los resultados obtenidos de manera positiva.



[Figura 30] - Resultados obtenidos en frecuencia de uso

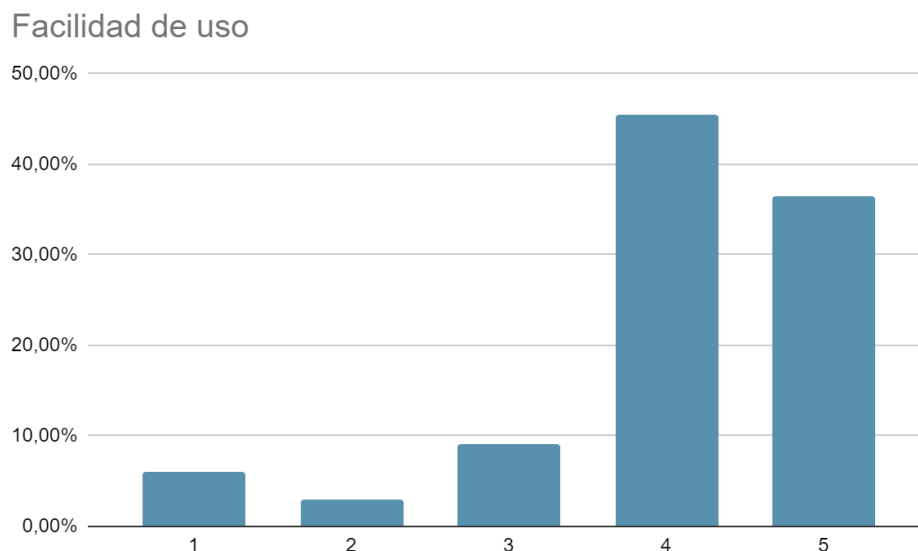
A continuación, los encuestados tenían la oportunidad de expresar su opinión acerca de la complejidad de la aplicación. El 54,5% de ellos no vio ningún tipo de complejidad. Por otro lado, el porcentaje de usuarios que respondieron en el intervalo del 3 al 5, es menor a un 15%. Aquellos encuestados que expresaron alguna complejidad en la aplicación, solían coincidir con aquellos usuarios que hacen un menor uso de aplicaciones para dispositivos móviles.

Los resultados obtenidos en esta pregunta de la encuesta, reflejados en la Figura 31, fueron satisfactorios, y en ellos vemos reflejado que hemos alcanzado uno de los objetivos que pretendíamos con nuestra aplicación: la simplicidad, ya que como se ha visto en el estudio de mercado, la gran mayoría de aplicaciones en este campo del mercado, presentan mayores dificultades para los usuarios.



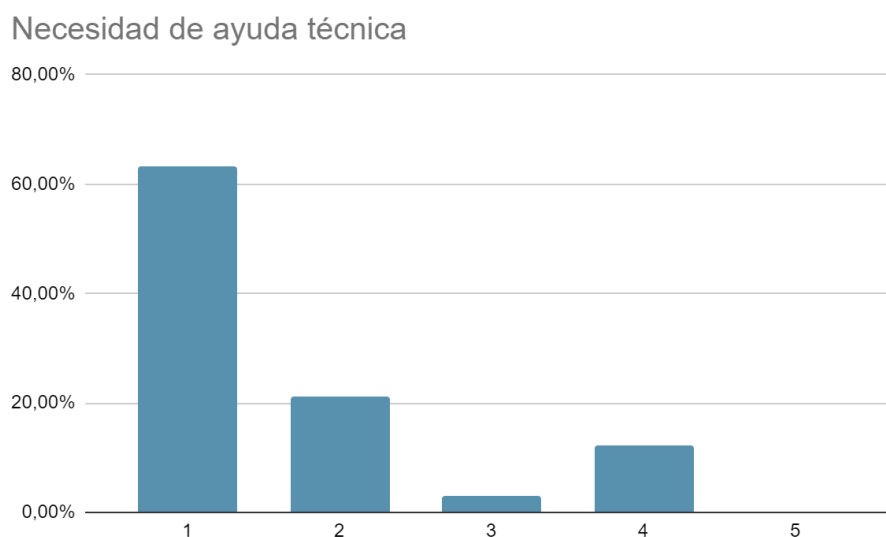
[Figura 31] - Resultados obtenidos en complejidad

La siguiente pregunta del formulario trataba sobre la facilidad de uso de GesCom, y en ella hemos obtenido resultados inversos a la pregunta anterior, tal y como se observa en la Figura 32. El 45,5% de los encuestados estaban totalmente de acuerdo con que GesCom es fácil de usar, y al igual que en la pregunta anterior, el porcentaje de personas que no contemplan la aplicación como fácil de usar, no llega a alcanzar el 15%.



[Figura 32] - Resultados obtenidos en facilidad de uso

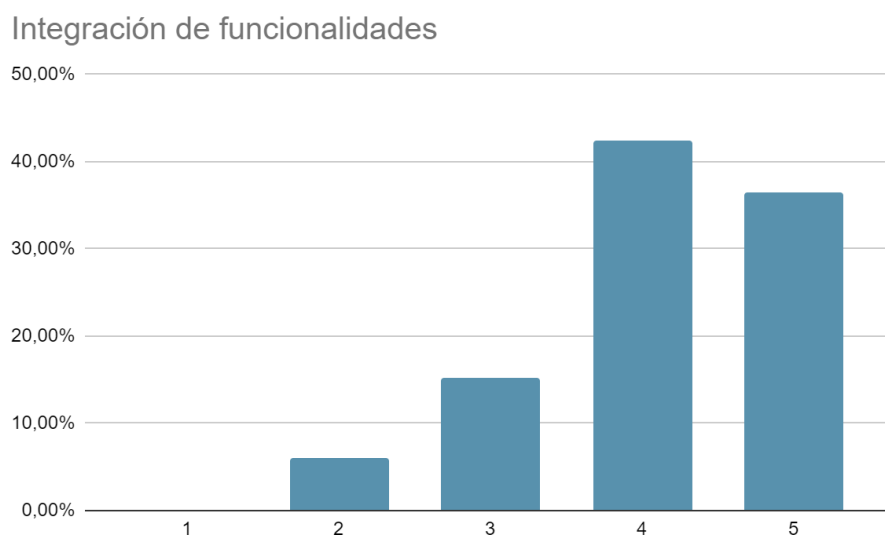
Cuando los encuestados fueron cuestionados sobre la necesidad de ayuda, por parte de una persona con conocimientos técnicos a la hora de usar GesCom, obtenemos el mayor porcentaje obtenido, apreciable en la Figura 33, siendo este de un 63,6%, de usuarios seleccionando la opción más positiva. Este porcentaje de usuarios está en completo desacuerdo con el enunciado.



[Figura 33] - Resultados obtenidos en necesidad de ayuda técnica

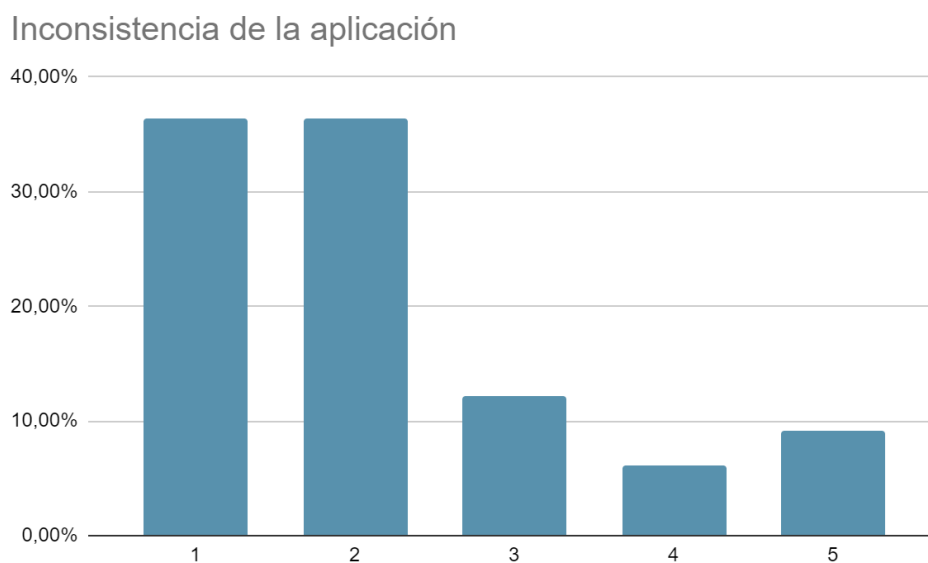
Para continuar con el cuestionario, los usuarios fueron preguntados si en su opinión, las funcionalidades de la aplicación estaban bien integradas. Los resultados obtenidos en esta pregunta no fueron menos satisfactorios que los

anteriores, ya que más del 70% de los usuarios encuestados estaban bastante o muy de acuerdo con la afirmación, como se puede observar en la Figura 34.



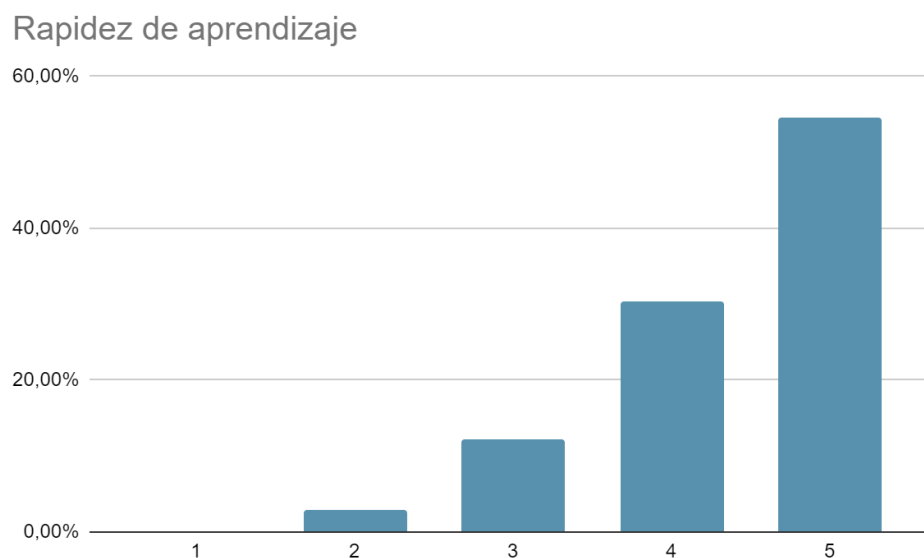
[Figura 34] - Resultados obtenidos en integración de funcionalidades

El 70% de los usuarios que realizaron las experiencias no estaban de acuerdo con que nuestra aplicación sea inconsistente. Consideramos que estas cifras son bastante significativas, sin embargo, se profundizará más en los aspectos que hicieron que casi el 30% de los usuarios, como se ve en la Figura 35, concibieran GesCom como inconsistente.



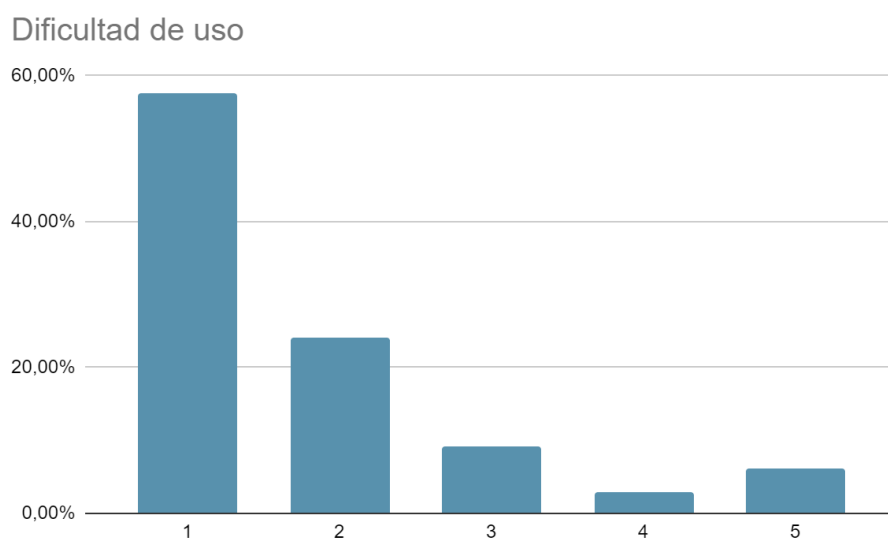
[Figura 35] - Resultados obtenidos en inconsistencia

Ya que nuestra aplicación está diseñada para ser simple e intuitiva, consideramos que los resultados que muestra la Figura 36, nos permiten confirmar que es una aplicación que ha satisfecho sus objetivos, permitiendo así que personas de variada edad y experiencia de uso tecnológico, puedan aprender rápidamente cómo llevar a cabo los diferentes casos de uso.



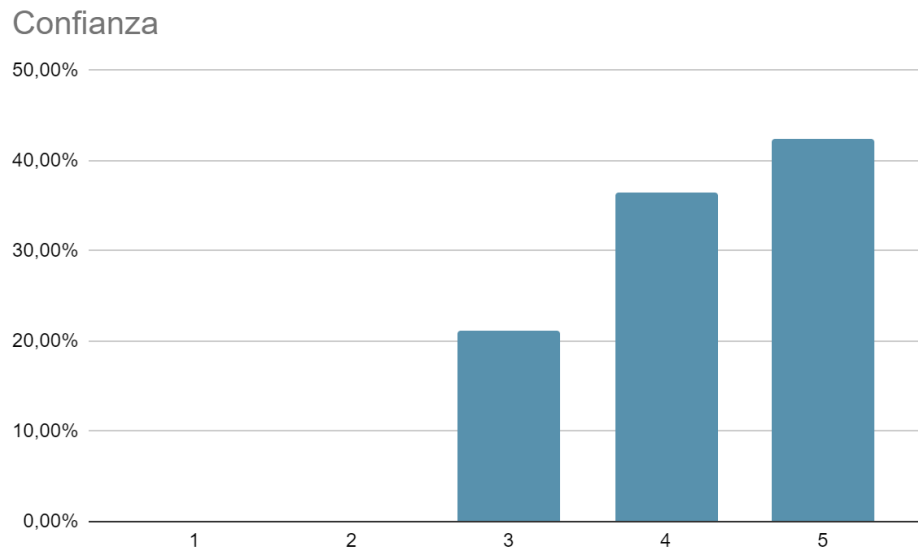
[Figura 36]- Resultados obtenidos en rapidez de aprendizaje

Los resultados obtenidos en esta siguiente pregunta corroboran las conclusiones extraídas de los resultados obtenidos en la pregunta anterior. Un 80% de los encuestados, tal y como se ve en la Figura 37, están muy en desacuerdo con la afirmación “Encuentro que GesCom es muy difícil de usar”.



[Figura 37] - Resultados obtenidos en dificultad de uso

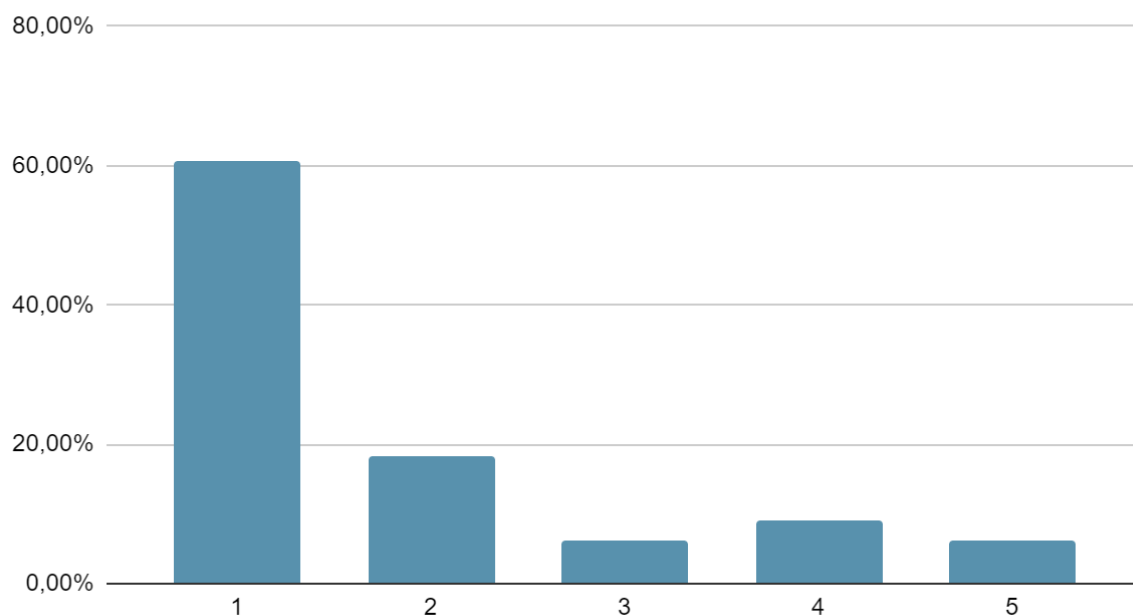
Tal y como se muestra en la Figura 38, los resultados obtenidos son significativamente satisfactorios, ya que ningún usuario experimentó un nivel de confianza menor que la media al usar nuestra aplicación. Otro aspecto positivo es que el porcentaje es proporcional al valor, por lo tanto hay más usuarios que hayan experimentado mucha confianza en comparación con los que han experimentado un grado de confianza menor, y así sucesivamente.



[Figura 38] - Resultados obtenidos en confianza

Los resultados obtenidos sobre la necesidad de realizar un aprendizaje previo al uso de la aplicación son satisfactorios, ya que, como muestra la Figura 39, más de un 60% de los usuarios encuestados consideran que no hay ninguna necesidad de aprender nada nuevo antes de realizar casos de uso en la aplicación.

### Necesidad de aprendizaje previo



[Figura 39] - Resultados obtenidos en aprendizaje previo

A continuación, realizaremos un análisis global de todos los datos obtenidos en el cuestionario sobre la Escala de Usabilidad del Sistema, con el fin de evaluar si los objetivos principales de nuestra aplicación han sido alcanzados durante el desarrollo de GesCom.

Para llevar a cabo esta evaluación global, comenzaremos por asignar a cada respuesta su mismo valor, por ejemplo, “Muy en desacuerdo” recibiría un valor de 1, mientras que “Totalmente de acuerdo” recibiría el valor de 5.

Tras asignar estos valores calculamos la suma de los valores en las preguntas impares por una parte, y de las preguntas pares por otra parte. A partir de estos valores obtenidos sacamos dos variables X e Y para obtener la puntuación:

$$X = \text{Sum of the points for all odd-numbered questions} - 5$$

$$Y = 25 - \text{Sum of the points for all even-numbered questions}$$

$$\text{SUS Score} = (X + Y) \times 2.5$$

La puntuación SUS media obtenida es de 87.5, lo que significa que siguiendo la guía de interpretación de puntuaciones SUS, nuestra aplicación presenta una calificación de tipo excelente.



# **Capítulo 5**

## **Conclusiones y trabajo futuro**

## 5.1 Conclusiones

En este Trabajo de Fin de Grado se ha desarrollado una aplicación móvil que facilita una gestión más cómoda y sencilla dentro de una comunidad de vecinos.

Los administradores introducen los datos de la comunidad y sus miembros durante el proceso de registro, tendrán acceso a un tablón donde publicar avisos, podrán gestionar votaciones y recibir incidencias por parte de otros miembros de la comunidad. Los miembros de la comunidad a su vez podrán reportar incidencias que encuentren, participar en las votaciones de la comunidad y ver información de la comunidad.

Se ha desarrollado nuestra aplicación “GesCom” con la ayuda de Android Studio, herramienta empleada por primera vez para el desarrollo de un proyecto de este calibre. Haciendo uso de Java hemos podido utilizar un lenguaje orientado a objetos, a esto hay que sumarle la implementación del “model view-presenter” como uno de los patrones de diseño utilizados. Nos ha facilitado la estructuración del código y ha sido esencial para ahorrar tiempo a la hora de intentar resolver posibles conflictos durante la creación del proyecto.

Durante el desarrollo de este proyecto se han cumplido los objetivos principales del mismo y se ha comprendido la complejidad de diseñar un sistema que presenta diferentes frentes abiertos y que reúne conocimientos de diferentes materias. La aplicación GesCom recoge una serie de conocimientos adquiridos a lo largo de la carrera, y puestos en práctica gracias a la realización del presente trabajo.

## 5.2 Conclusions

In this Final Degree Project we have developed a mobile application that facilitates a more comfortable and simple management within a community of neighbors.

Administrators enter the data of both the community and its members during the registration process, they will have access to a board where they can post notices, they will be able to manage the voting process and receive incidents from other members of the community. Community members in turn will be able to report incidents they find, participate in community votes and view community information.

Our application named “GesCom” has been developed with the help of Android Studio, a tool we have used for the first time to develop a project of this caliber.

Using Java we have been able to use an object-oriented language, additionally we have added the implementation of the "model view-presenter" as one of the used design patterns. It has made it easier for us to structure the code and has been essential to save time when trying to resolve possible conflicts during the creation of the project.

During the development of this project, the main objectives have been met and we have understood the complexity of designing a system that presents different open fronts and that gathers knowledge from different subjects. The GesCom application collects a series of knowledge acquired throughout the career that is put into practice thanks to the completion of this work.

### 5.3 Trabajo futuro

Finalizado el desarrollo para el alcance de nuestro proyecto, cabe destacar el gran número de ampliaciones y evoluciones que se podrían llevar a cabo en nuestro código. A continuación se listan algunos de los trabajos futuros y las funcionalidades que se podrán llevar a cabo para aumentar el alcance de GesCom:

- Delegación del voto: En la versión implementada el miembro de la comunidad tiene derecho a realizar una votación, se podría implementar una funcionalidad para permitir delegar el voto en otro miembro de la comunidad en caso de no poder votar durante la franja horaria establecida.
- Reserva de instalaciones deportivas y espacios comunes de la comunidad: Se podría implementar una funcionalidad que permita a los miembros reservar zonas comunes o instalaciones en franjas horarias determinadas. El administrador podría variar el número de plazas por instalación y las horas de disponibilidad de las mismas.
- Administración de múltiples comunidades por parte de un mismo usuario: en casos en los que un administrador de la comunidad ha sido contratado por la misma, y no reside en ella, puede darse el caso de que este usuario administre dos o más comunidades.
- Pago de derramas y consulta de recibos: Los miembros de la comunidad podrían visualizar los pagos o trámites ya cobrados dentro de la aplicación. De esta forma podrían tener un historial de los gastos dentro de la comunidad y la fecha en la que se realizaron dichos trámites.

- Información de contacto de incidencias en la aplicación: sería de gran uso implementar un apartado donde los usuarios tengan acceso a la información de los diferentes servicios contratados por la comunidad, tales como: empresas del agua y de la luz, empresa de la calefacción, servicio de seguridad, etc...
- Posibilidad de adjuntar una foto a incidencias: Actualmente un miembro puede redactar una descripción de la incidencia que desea reportar. En una versión actualizada se podría permitir al usuario adjuntar una foto a la incidencia.
- Posibilidad de adjuntar archivos a avisos: Actualmente el usuario administrador puede añadir una descripción al aviso que publica, pero en muchos casos, sería de gran utilidad para el administrador, el poder adjuntar archivos relacionados, archivos informativos o aclaratorios de dicho aviso.
- Pertenencia de un usuario a dos o más comunidades: para casos en los que un usuario es propietario de más de una vivienda y debe ser activo tanto en su comunidad de residencia como en el resto.
- Existencia de varios usuarios convivientes: en el caso de que un usuario sea copropietario de la vivienda, sería de conveniencia la implementación de diferentes sesiones para la gestión de la misma.



# **Capítulo 6**

## **Aportaciones individuales**

## 6.1 Luis Jesús Cepeda Caperos

### Ingeniería de requisitos

- Obtención y análisis de las respuestas de la encuesta inicial  
Creación y distribución inicial del cuestionario inicial, además del posterior análisis de los resultados.
- Investigación bibliográfica  
Investigación y análisis de fuentes externas, citas y referencias para ayudar a profundizar en los conceptos expuestos.
- Definición previa de los roles de usuario  
Análisis de los posibles usuarios que podrían interactuar con la aplicación y los roles que se les debería asignar para evitar conflictos.

### Desarrollo

- Funcionalidad de avisos  
Diseño de la interfaz e implementación de la funcionalidad de avisos de miembros de la comunidad y administradores.
- Creación y actualización de tableros de metodología ágil  
Estructuración y revisión de la implementación de los tableros de metodología ágil para la distribución de trabajo.
- Realización de experiencias de usuario  
Proceso de verificación de transición de vistas y funcionamiento de las acciones realizadas por un usuario a la hora de navegar por la aplicación.

### Memoria

- Estructuración de la memoria  
Distribución en partes de los diferentes apartados y subapartados en los que debería consistir este documento. Con posibilidad de modificación futura en caso de necesitar añadir detalles adicionales.
- Investigación y redacción de la introducción  
Proceso de investigación de citas y redacción del apartado de la introducción.

- Redacción del estudio de mercado  
Proceso de investigación de aplicaciones del mercado similares y análisis de los resultados obtenidos. Elaboración del apartado del estudio de mercado.
- Redacción de usuarios del sistema  
Redacción de los diferentes usuarios que pueden interactuar con la aplicación y las diferentes acciones que pueden tomar cada uno de ellos.
- Implementación y desarrollo de casos de uso  
Creación e implementación de las tablas de casos de uso, especificando las funcionalidades e interacciones que puede hacer un usuario con la aplicación.
- Redacción de conclusiones y trabajos futuros  
Detallado de las conclusiones obtenidas tras el desarrollo de este proyecto y trabajos futuros que se podrían realizar sobre nuestro proyecto para poder mejorar y optimizar en una versión posterior.
- Redacción y análisis de los resultados en interacciones con usuarios  
Con los datos obtenidos de usuarios utilizando nuestra aplicación, procedió a redactar un análisis de los resultados.
- Traducción de la introducción y conclusión  
Redacción en inglés de los apartados de introducción y conclusión además de la verificación ortográfica de los textos.
- Redacción y estructuración del manual de usuario  
Creación del manual de usuario para intentar ofrecer una idea del proceso de funcionamiento de la aplicación. Una guía que detalle las diferentes acciones que se pueden realizar explicando las transiciones entre diferentes vistas.

## 6.2 Teresa Fernández

### Ingeniería de requisitos

- Obtención de respuestas para el cuestionario inicial  
Difusión y análisis de los resultados obtenidos en el cuestionario inicial para la especificación de requisitos.
- Investigación de patrones arquitectónicos  
Recopilación de información sobre el desarrollo profesional de aplicaciones móviles y aquellos patrones aplicados al mismo.

### Desarrollo

- Diseño e implementación de la BBDD  
Estructuración y diseño de la base de datos e implementación de la misma para la aplicación.
- Estructuración de las *activities*  
Diseño y encapsulación de los métodos y variables de las diferentes *activities*.
- Implementación de adaptadores  
Desarrollo del código de los adaptadores empleados en aquellas vistas cuya implementación depende del número de objetos a pintar.
- Implementación funcionalidad de avisos  
Implementación de la lógica de edición y eliminación de avisos. Diseño del componente de aviso.
- Implementación funcionalidad de perfil de usuario  
Diseño de la pantalla de visualización de perfil. Desarrollo e implementación de la lógica y el flujo de navegación.
- Implementación funcionalidad de votaciones  
Diseño de la pantalla de votaciones y de la visualización de las mismas desde el dashboard. Desarrollo e implementación de la lógica y el flujo de navegación. Realización de todas las pantallas.
- Implementación funcionalidad de ajustes  
Diseño de la pantalla de ajustes e implementación del código.
- Implementación funcionalidad de gestión de usuarios

Desarrollo del adaptador de los usuarios y de las lógicas de edición y eliminación de las mismas.

## Memoria

- Descripción de las tecnologías empleadas  
Investigación previa, análisis y posterior descripción de las tecnologías empleadas a lo largo del proyecto.
- Implementación del modelo entidad-relación  
Diseño y realización del diagrama del modelo entidad-relación junto con la descripción y explicación del mismo.
- Implementación y análisis de los patrones de diseño y arquitectónicos  
Investigación previa a su uso. Análisis y explicación de los diferentes patrones de diseño y arquitectura.
- Análisis Escala de Usabilidad del Sistema  
Ejecución de los Test para el análisis de la usabilidad del sistema y posterior entrevista con los usuarios.
- Diagramas de interacción y de clases  
Diseño de los diagramas de interacción y de clases para aquellas funcionalidades en las que los mismos ayudan a la comprensión del código implementado.
- Exposición y análisis del diseño de la aplicación  
Explicación del diseño tanto del código como de la interfaz implementados, para cada una de las funcionalidades principales de la aplicación.
- Investigación bibliográfica  
Investigación de artículos científicos relacionados con el desarrollo de nuestra aplicación. Posterior redacción y referenciación de dichos artículos en la bibliografía.
- Investigación y redacción de la introducción  
Proceso de investigación de citas y redacción del apartado de la introducción.



# **Capítulo 7**

## **Bibliografía y referencias**

- [1] Ibrahim Osman Adam, Muftawu Dzang Alhassan, [The effect of mobile phone penetration on the quality of life](#), Telecommunications Policy, Volume 45, Issue 4, 2021, 102109, ISSN 0308-5961.
- [2] Gadzama, Wadzani & Joseph, Bitrus & Aduwamai, Ngubdo. (2019). [Global Smartphone Ownership, Internet Usage And Their Impacts On Humans](#). Research journal's Journal of Communications Networks, vol. 1, 2017.
- [3] Montag, C., Błaszczewicz, K., Sariyska, R. *et al.* [Smartphone usage in the 21st century: who is active on WhatsApp?](#). *BMC Res Notes* 8, 331 (2015).
- [4] Hossein Falaki, Ratul Mahajan, Srikanth Kandula, Dimitrios Lymberopoulos, Ramesh Govindan, and Deborah Estrin. 2010. [Diversity in smartphone usage](#). In Proceedings of the 8th international conference on Mobile systems, applications, and services (MobiSys '10). Association for Computing Machinery, New York, NY, USA, 179–194.
- [5] Shivi Garg, Niyati Baliyan, [Comparative analysis of Android and iOS from security viewpoint](#), Computer Science Review, Volume 40, 2021,100372,ISSN 1574-0137,
- [6] [Number of apps available in leading app stores 2020](#), Statista Research Department, May 11, 2021.
- [7] Comunidad de propietarios, [https://es.wikipedia.org/wiki/Comunidad\\_de\\_propietarios](https://es.wikipedia.org/wiki/Comunidad_de_propietarios). Recuperado el día 17 de Mayo de 2021.
- [8] Professor M. G. Worster, Dept of Applied Mathematics & Theoretical Physics|Centre for Mathematical Science, [Journal of Fluid Mechanics](#) , Volume 903 , 25 November 2020 , F1.
- [9] Restricciones Covid-19 en comunidades de vecinos. <https://www.comunidad.madrid/noticias/2021/01/26/informamos-200000-comunidades-vecinos-restricciones-adoptar-frente-covid-19>. Recuperado el día 17 de Mayo de 2021.
- [10] Bhagat, R., Davies Wykes, M., Dalziel, S., & Linden, P. (2020). [Effects of ventilation on the indoor spread of COVID-19](#). *Journal of Fluid Mechanics*, 903, F1.
- [11] BlueStacks. <https://en.wikipedia.org/wiki/BlueStacks>. Recuperado el día 20 de Mayo de 2021.

- [12] Google Play. [https://en.wikipedia.org/wiki/Google\\_Play](https://en.wikipedia.org/wiki/Google_Play). Recuperado el día 20 de Mayo de 2021.
- [13] TuComunidApp, <https://tucomunidapp.com/> . Recuperado el día 16 de Mayo de 2021.
- [14] MiComunidad, <https://cutt.ly/obCDYKG>. Recuperado el día 16 de Mayo de 2021.
- [15] ComunidadFeliz, <https://cutt.ly/MbCDAQe>. Recuperado el día 16 de Mayo de 2021.
- [16] SomosVecinos, <https://cutt.ly/vbCDMEd>. Recuperado el día 16 de Mayo de 2021.
- [17] MyComunidad, <https://cutt.ly/YbCD5qW>. Recuperado el día 16 de Mayo de 2021.
- [18] Fynkus, <https://cutt.ly/AbCD9Ch>. Recuperado el día 16 de Mayo de 2021.
- [19] Tu Comunidad, <https://cutt.ly/JbCFq2T>. Recuperado el día 16 de Mayo de 2021.
- [20] Kit de Desarrollo de Software.  
[https://en.wikipedia.org/wiki/Software\\_development\\_kit](https://en.wikipedia.org/wiki/Software_development_kit). Recuperado el día 20 de Mayo de 2021.
- [21] Bryan Pon, Timo Seppälä, Martin Kenney, [Android and the demise of operating system-based power: Firm strategy and platform control in the post-PC world](#), Telecommunications Policy, Volume 38, Issue 11, 2014, Pages 979-991, ISSN 0308-5961.
- [22] Github, <https://es.wikipedia.org/wiki/GitHub>. Recuperado el día 17 de Mayo de 2021.
- [23] Github Actions. <https://github.com/features/actions>. Recuperado el día 17 de Mayo de 2021.
- [24] Luca Ardito, Riccardo Coppola, Giovanni Malnati, Marco Torchiano, [Effectiveness of Kotlin vs. Java in android app development tasks](#), Information and Software Technology, Volume 127, 2020, 106374, ISSN 0950-5849,

[25] Zbigniew M. Sikora, Introduction, Editor(s): Zbigniew M. Sikora, In The Morgan Kaufmann Series in Data Management Systems, [Java: Practical Guide For Programmers](#), Morgan Kaufmann, 2003, Pages 1-6, ISBN 9781558609099.

[26] Lenguaje de programación Java.  
[https://en.wikipedia.org/wiki/Java\\_\(programming\\_language\)](https://en.wikipedia.org/wiki/Java_(programming_language)). Recuperado el día 20 de Mayo de 2021.

[27] XML. <https://en.wikipedia.org/wiki/XML>. Recuperado el día 20 de Mayo de 2021.

[28] Lenguaje de programación Kotlin.  
[https://en.wikipedia.org/wiki/Kotlin\\_\(programming\\_language\)](https://en.wikipedia.org/wiki/Kotlin_(programming_language)). Recuperado el día 20 de Mayo de 2021.

[29] Pietro H. Guzzi, [Computing Languages for Bioinformatics: Java](#), Editor(s): Shoba Ranganathan, Michael Gribskov, Kenta Nakai, Christian Schönbach, Encyclopedia of Bioinformatics and Computational Biology, Academic Press, 2019, Pages 206-208, ISBN 9780128114322.

[30] Sun Microsystems. [https://en.wikipedia.org/wiki/Sun\\_Microsystems](https://en.wikipedia.org/wiki/Sun_Microsystems). Recuperado el día 20 de Mayo de 2021.

[31] LucidChart. <https://www.lucidchart.com/pages/es>. Recuperado el día 17 de Mayo de 2021.

[32] Creately. <https://creately.com/presscoverage>. Recuperado el día 17 de Mayo de 2021.

[33] Google Doc. [https://en.wikipedia.org/wiki/Google\\_Docs](https://en.wikipedia.org/wiki/Google_Docs). Recuperado el día 20 de Mayo de 2021.

[34] Google Forms, <https://www.google.es/intl/es/forms/about>. Recuperado el día 16 de Mayo de 2021.

[35] Google. <https://en.wikipedia.org/wiki/Google>. Recuperado el día 20 de Mayo de 2021.

[36] Jan L. Harrington, 1 - Installing and Running MySQL, Editor(s): Jan L. Harrington, In The Morgan Kaufmann Series in Data Management Systems, [SQL Clearly Explained \(Second edition\)](#), Morgan Kaufmann, 2003, Pages 3-8, ISBN 9781558608764.

[37] General Public License.

[https://en.wikipedia.org/wiki/GNU\\_General\\_Public\\_License](https://en.wikipedia.org/wiki/GNU_General_Public_License). Recuperado el día 20 de Mayo de 2021.

[38] Structured Query Language. <https://en.wikipedia.org/wiki/SQL>. Recuperado el día 20 de Mayo de 2021.

[39] SCRUM. <https://www.scrum.org/resources/what-is-scrum>. Recuperado el día 20 de Mayo de 2021.

[40] Kanban.

<https://kanbanize.com/kanban-resources/getting-started/what-is-kanban>.

Recuperado el día 20 de Mayo de 2021.

[41] Kresna Dwi Prasetya, Suharjito, Devriady Pratama, [Effectiveness Analysis of Distributed Scrum Model Compared to Waterfall approach in Third-Party Application Development](#), Procedia Computer Science, Volume 179, 2021, Pages 103-111, ISSN 1877-0509.

[42] Muhammad Ovais Ahmad, Denis Dennehy, Kieran Conboy, Markku Oivo, [Kanban in software engineering: A systematic mapping study](#), Journal of Systems and Software, Volume 137, 2018, Pages 96-113, ISSN 0164-1212,

[43] Trello, <https://trello.com/about>. Recuperado el día 17 de Mayo de 2020.

[44] Luis González Jiménez, [REERM: Reenhancing the entity–relationship model](#), Data & Knowledge Engineering, Volume 58, Issue 3, 2006, Pages 410-435, ISSN 0169-023X.

[45] Kevin Lano, Shekoufeh Kolahtouz-Rahimi, Sobhan Yassipour-Tehrani, Mohammadreza Sharbaf, [A survey of model transformation design patterns in practice](#), Journal of Systems and Software, Volume 140, 2018, Pages 48-73, ISSN 0164-1212.

[46] Dan Dan Li, Xiao Yan Liu, [Research on MVP Design Pattern Modeling Based on MDA](#), Procedia Computer Science, Volume 166, 2020, Pages 51-56, ISSN 1877-0509.

[47] Modelo Vista-Presentador.

<https://en.wikipedia.org/wiki/Model–view–presenter>. Recuperado el día 20 de Mayo de 2021.

[48] Modelo Vista-Controlador.

<https://en.wikipedia.org/wiki/Model–view–controller>. Recuperado el día 20 de Mayo de 2021.

[49] Martin Reddy, Chapter 3 - Patterns, Editor(s): Martin Reddy, [API Design for C++](#), Morgan Kaufmann, 2011, Pages 65-104, ISBN 9780123850034.

[50] Patrón de diseño Singleton. [https://en.wikipedia.org/wiki/Singleton\\_pattern](https://en.wikipedia.org/wiki/Singleton_pattern). Recuperado el día 20 de Mayo de 2021.

[51] “System Usability Scale”. <https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html>. Recuperado el día 20 de Mayo de 2021.

## Anexo I: Cuestionario Aplicación Gestión de Comunidades

### **1. Información de la situación de residencia de los participantes.**

1.1 ¿Cuántas personas conviven en total en tu lugar de residencia?

- 1
- 2
- 3
- 4
- 5+

1.2 ¿Cuántas personas conviven en total en tu lugar de residencia?

- Propietario de la vivienda
- Piso en alquiler
- Otra...

1.3 ¿Tiene una comunidad de vecinos?

- Sí → sección 2
- No → Enviar formulario

### **2. Información de aplicaciones similares empleadas por los usuarios**

2.1 ¿Hace uso de alguna aplicación móvil o web para la gestión de su comunidad de vecinos?

- No → sección 4
- Fynkus
- Colindar
- Community Pro
- APP comunicarte
- FiacAPP
- Insulae
- App Vecinos en Red
- Otra

### **3. Características de la aplicación empleada y sistema operativo**

3.1 En caso de emplear una app, elija la plataforma en la que se puede usar:

- iOS
- Android
- Ambos
- Web

3.2 En caso de hacer uso de alguna aplicación móvil o web para la gestión de su comunidad de vecinos. Seleccione aquellas características que ofrece:

- Chat de comunicación entre vecinos

- Votación de la comunidad
- Notificación de vecinos en cuarentena
- Videollamadas para realizar reuniones y juntas
- Reportar incidencias
- Cargo del gasto de incidencias/derramas
- Recibos de gastos de la comunidad
- Reserva de espacios de áreas comunas por hora
- Uso gratuito
- Otra...

#### **4. Valoración del usuario de las funciones de una aplicación para comunidades de vecinos**

4.1 ¿Qué procesos le parecen lentos o frustrantes en una comunidad de vecinos?

- Las reuniones de vecinos
- La notificación de incidencias al administrador o personal de mantenimiento
- Las votaciones de la comunidad
- La reserva de manera física de las instalaciones de la comunidad
- La comunicación cara a cara con vecinos
- Las juntas de la comunidad
- Otra...

4.2 ¿Qué herramientas facilitarían la interacción en su comunidad de vecinos si usara una aplicación móvil?

- Chat de comunicación entre vecinos
- Votación de la comunidad
- Videollamadas para realizar reuniones y juntas
- Notificación de vecinos en cuarentena
- Reportar incidencia
- Cargo del gasto de incidencias/derramas
- Recibos de gastos de la comunidad
- Reserva de espacios de áreas comunes por hora (Ej.:piscina)
- Uso gratuito
- Otra...

4.3 ¿Qué funcionalidad(es) considera IMPRESCINDIBLE a la hora de utilizar una aplicación móvil para la gestión de comunidades de vecinos?

Texto de respuesta larga

4.4 ¿Qué funcionalidad(es) considera NO ÚTILES a la hora de utilizar una aplicación móvil para la gestión de comunidades de vecinos?

Texto de respuesta larga

## Anexo II: Encuesta final a los usuarios

Nombre del formulario: Escala de usabilidad de GesCom.

Descripción: Para cada una de las siguientes afirmaciones, marque una casilla que describa mejor sus reacciones a GesCom. (1- Muy en desacuerdo, 5 - Totalmente de acuerdo)

Creo que usaría este GesCom frecuentemente

- 1
- 2
- 3
- 4
- 5

Encuentro este GesCom innecesariamente complejo.

- 1
- 2
- 3
- 4
- 5

Creo que el GesCom fue fácil de usar.

- 1
- 2
- 3
- 4
- 5

Creo que necesitaría ayuda de una persona con conocimientos técnicos para usar este GesCom.

- 1
- 2
- 3
- 4
- 5

Las funciones de este GesCom están bien integradas.

- 1
- 2
- 3
- 4
- 5

Creo que el GesCom es muy inconsistente.

- 1
- 2
- 3
- 4
- 5

Imagino que la mayoría de la gente aprendería a usar este GesCom en forma muy rápida.

- 1
- 2
- 3
- 4
- 5

Encuentro que el GesCom es muy difícil de usar.

- 1
- 2
- 3
- 4
- 5

Me siento confiado al usar este GesCom.

- 1
- 2
- 3
- 4
- 5

Necesité aprender muchas cosas antes de ser capaz de usar este GesCom.

- 1
- 2
- 3
- 4
- 5

## Anexo III: Manual de usuario

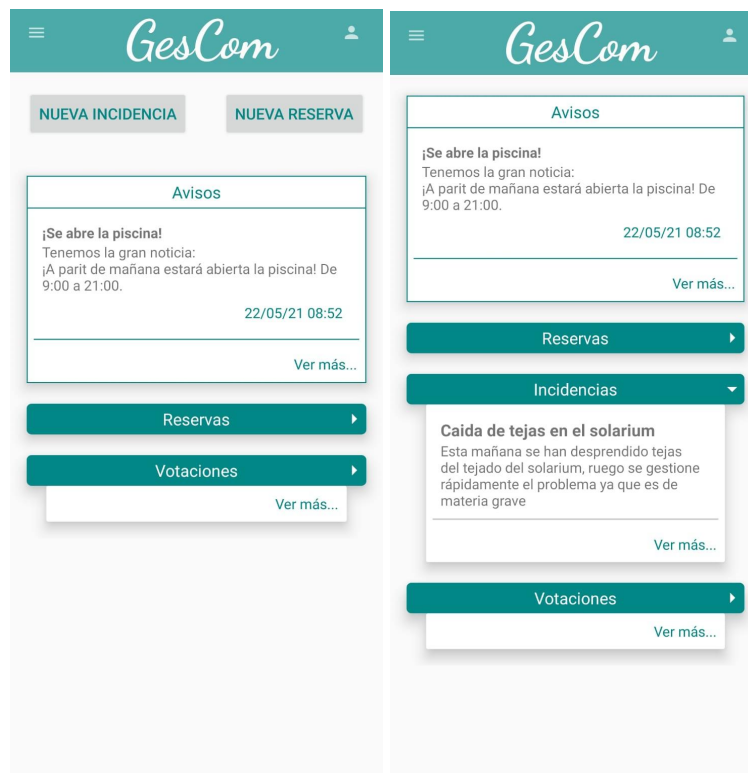
Una vez descargado el archivo APK obtenido del repositorio, se abre la aplicación y se muestra la pantalla de inicio de sesión, en la que también encontramos las opciones de registrarse como usuario, registrar una comunidad o recuperar la contraseña, presentadas en la Figura 40.



[Figura 40] - Pantallas de login, registro de usuario y registro de una comunidad.

En caso de no disponer de una cuenta, el usuario deberá registrarse como usuario clicando en “Regístrate”, haciendo uso de la clave proporcionada previamente por el administrador de su comunidad. Si el usuario es un administrador cuya comunidad no está registrada, entonces el usuario podrá registrar la misma en “Registra tu comunidad”, introduciendo el nombre de la comunidad y creándose un perfil de administrador.

En la pantalla principal de la aplicación el usuario puede visualizar los últimos avisos, votaciones y reservas registradas en la base de datos, es decir, las más recientes. Si se trata de un usuario administrador, éste podrá también ver las últimas incidencias reportadas por los vecinos de la comunidad. Por otro lado, el usuario vecino, tendrá acceso a dos atajos: reportar una nueva incidencia y realizar una nueva reserva.



[Figura 41] - Pantalla principal para diferentes usuarios

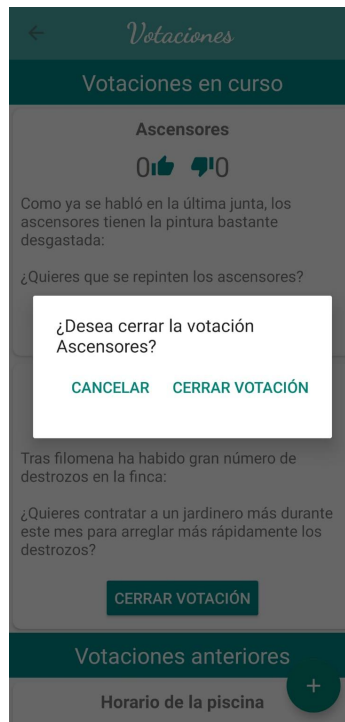
Desde la pantalla principal, como se ve en la Figura 41, el usuario puede desplegar el menú principal en la esquina superior izquierda o acceder a su perfil de usuario en el icono de la esquina superior derecha.

Desde el menú principal el usuario podrá acceder a la gran mayoría de funcionalidades de la aplicación. Accediendo a votaciones un usuario podrá participar en aquellas situadas en el apartado de “votaciones en curso”. Tras seleccionar si está a favor o en contra tendrá que confirmar la ventana emergente. También podrá visualizar votaciones anteriores junto con sus resultados, Figura 42.



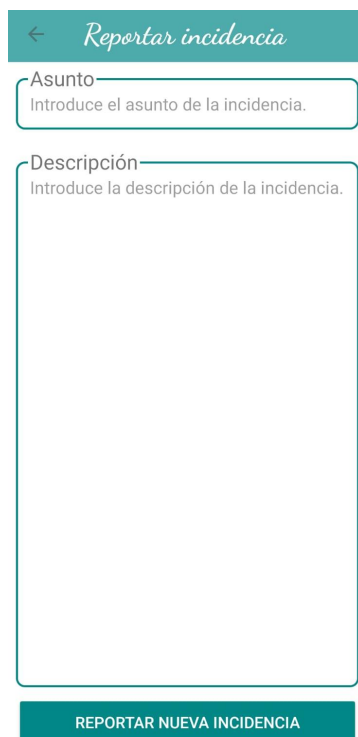
[Figura 42] - Visualización de votaciones

El administrador tendrá la posibilidad de añadir una nueva votación incluyendo el título y la descripción. Una vez abierta podrá cerrar la votación cuando lo desee, impidiendo que otros miembros voten, Figura 43.



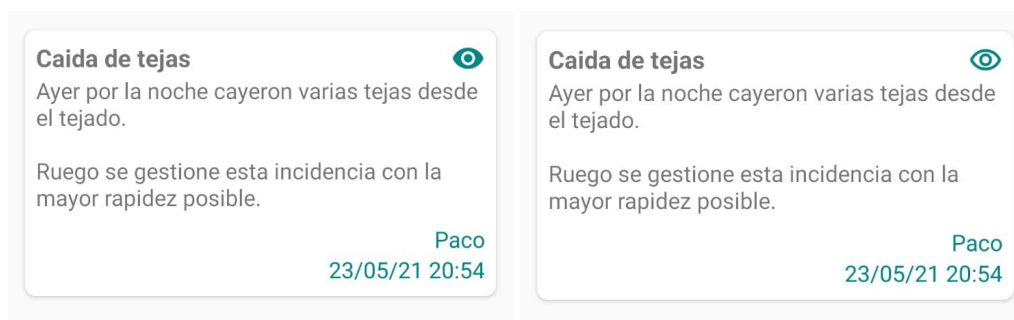
[Figura 43] - Cerrar votación

Además se podrá acceder al apartado de incidencias. Un miembro de la comunidad puede reportar incidencias pulsando sobre el botón de la esquina inferior derecha. Tras introducir los datos puedes confirmar la incidencia, Figura 44, que será enviada al administrador.



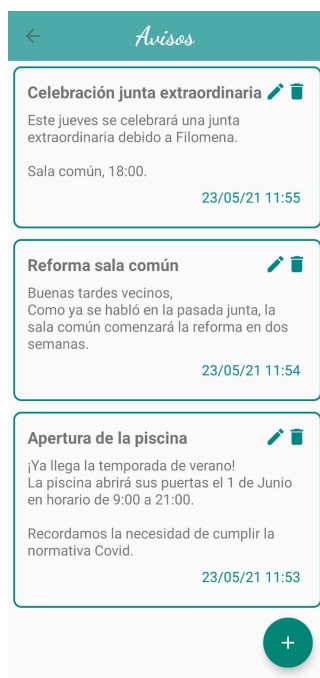
[Figura 44] - Reportar incidencia

El administrador puede visualizar todas las incidencias que le mandan los miembros de la comunidad, y marcarlas como visualizadas pulsando sobre el icono del ojo, Figura 45.



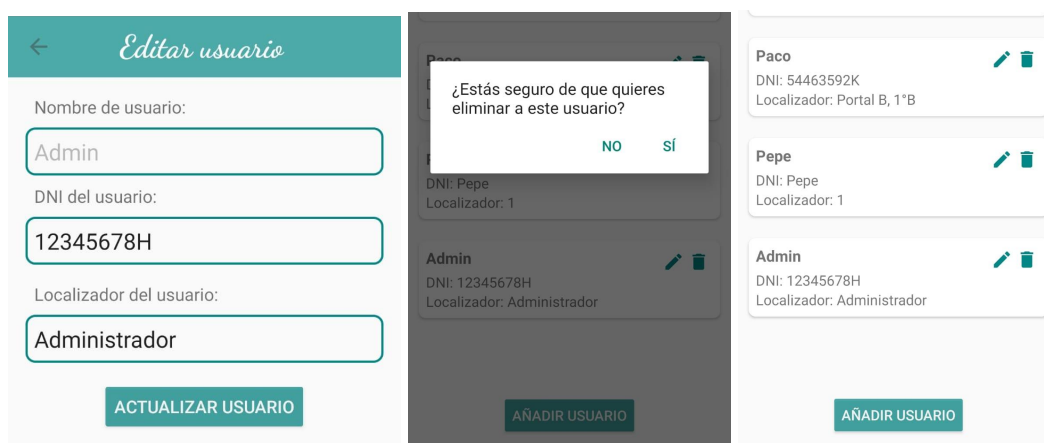
[Figura 45] - Marcación de una incidencia como gestionada

En el apartado de avisos el administrador puede crear, editar y eliminar avisos, que se mostrarán al resto de miembros de la comunidad, Figura 46. Los miembros que accedan a esta sección podrán ver los avisos publicados, ordenados por orden cronológico.



[Figura 46] - Gestión de avisos

El administrador podrá acceder a la sección de gestionar usuarios, donde podrá añadir, editar y eliminar miembros de la comunidad. Para añadir deberá pulsar en la parte inferior en “añadir usuario” y rellenar los campos solicitados. Para editar se puede pulsar sobre el icono con forma de lápiz, y tras rellenar los datos se debe pulsar “actualizar usuario”. Por último podrá eliminar usuarios pulsando sobre el icono de una papelera y verificando la decisión en la ventana emergente, Figura 47.



[Figura 47] - Gestión de usuarios