

APLICACIÓN WEB PARA LA ADMINISTRACIÓN DE PROPIEDAD HORIZONTAL
IMPLEMENTADO EN EL CONJUNTO RESIDENCIAL HACIENDA EL ROSAL
UBICADO EN FUNZA CUNDINAMARCA

KATHERINE AGUIRRE TIQUE
LEIDY YOHANA RODRÍGUEZ FUENTES

Proyecto de grado para optar por el título de Ingeniero de Sistemas y Computación
Modalidad: Desarrollo de software

Director de Proyecto
Ingeniera: MARÍA NURY ESCOBAR GUZMÁN
Directora Metodológica
Profesora: Rosalba Cruz Cepeda

UNIVERSIDAD ANTONIO NARIÑO
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS
BOGOTÁ D.C.
2020

APLICACIÓN WEB PARA LA ADMINISTRACIÓN DE PROPIEDAD HORIZONTAL
IMPLEMENTADO EN EL CONJUNTO RESIDENCIAL HACIENDA EL ROSAL
UBICADO EN FUNZA CUNDINAMARCA

KATHERINE AGUIRRE TIQUE
LEIDY YOHANA RODRÍGUEZ FUENTES

UNIVERSIDAD ANTONIO NARIÑO
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA DE SISTEMAS
BOGOTÁ D.C.
2020

DEDICATORIA Y AGRADECIMIENTOS

A mis padres, siendo las personas más importantes en mi vida, a quienes les agradezco por todos sus esfuerzos y apoyo el cual me permitió lograr cosas importantes. Igual que a todas las personas que me han ayudado y han hecho que el proyecto se realice con éxito.

Katherine Aguirre Tique

DEDICATORIA Y AGRADECIMIENTOS

Agradezco primeramente a Dios, a mi familia, a mi compañera de proyecto de grado, a los docentes de la carrera de ingeniería de sistemas de la facultad y a la directora, por el esfuerzo, los aprendizajes adquiridos y contribución al proceso de la construcción de este trabajo de grado.

Leidy Yohana Rodríguez Fuentes

CONTENIDO

	pág.
RESUMEN.....	12
INTRODUCCIÓN.....	13
1. PLANTEAMIENTO DEL PROBLEMA.....	15
1.1. DESCRIPCIÓN DEL PROBLEMA.....	15
1.2. FORMULACIÓN DEL PROBLEMA.....	16
1.3. JUSTIFICACIÓN.....	16
1.4. OBJETIVOS.....	17
1.4.1. Objetivo general.....	17
1.4.2. Objetivos específicos.....	18
1.5. ALCANCE Y LIMITACIONES DEL PROYECTO.....	18
1.5.1. Alcance.....	18
1.5.2. Limitaciones.....	20
2. ASPECTOS METODOLÓGICOS.....	22
2.1. DESCRIPCIÓN DE LA METODOLOGÍA INCREMENTAL.....	22
2.2. INCREMENTOS IMPLEMENTADOS EN EL PROYECTO.....	22
2.3. FASES DE LA METODOLOGÍA INCREMENTAL.....	23
2.3.1. Análisis de requerimientos.....	24
2.3.2. Diseño.....	24
2.3.3. Desarrollo.....	25
2.3.4. Pruebas.....	25
2.3.5. Despliegue.....	25
3. MARCO DE REFERENCIA.....	26
3.1. MARCO TEÓRICO.....	26
3.1.1. CONJUNTO RESIDENCIAL HACIENDA EL ROSAL.....	26
3.1.2. CONCEPTOS DE DESARROLLO DE SOFTWARE.....	27
3.2. ESTADO DEL ARTE.....	30
3.3. MARCO LEGAL.....	33

4.	DESARROLLO DEL PROYECTO	34
4.1.	INCREMENTO 1.....	34
4.1.1.	ANÁLISIS DE REQUERIMIENTOS	35
4.1.1.1.	Historias de usuario	35
4.1.1.2.	Requerimientos no funcionales	36
4.1.2.	DISEÑO	37
4.1.2.1.	Arquitectura de la aplicación.....	37
4.1.2.2.	Arquitectura de la aplicación en la nube AWS.....	38
4.1.2.3.	Diagrama de componentes.....	39
4.1.2.4.	Diagrama de clases	39
4.1.2.5.	Diagrama de secuencia	41
4.1.2.6.	Diagrama de historias de usuario	42
4.1.2.7.	Diagrama de actividades	43
4.1.2.8.	Diseño de la base de datos	43
4.1.3.	DESARROLLO.....	44
4.1.4.	PRUEBAS	45
4.1.5.	DESPLIEGUE	45
4.2.	INCREMENTO 2.....	48
4.2.1.	ANÁLISIS DE REQUERIMIENTOS	48
4.2.1.1.	Historias de usuario	48
4.2.2.	DISEÑO	49
4.2.2.1.	Diagrama de clases	49
4.2.2.2.	Diagramas de secuencia	49
4.2.2.3.	Diagrama de historias de usuario	49
4.2.2.4.	Diagrama de actividades	50
4.2.2.5.	Diseño de la base de datos	50
4.2.3.	DESARROLLO.....	51
4.2.4.	PRUEBAS	51
4.2.5.	DESPLIEGUE	52
4.3.	INCREMENTO 3.....	54
4.3.1.	ANÁLISIS DE REQUERIMIENTOS	54
4.3.1.1.	Historias de usuario	54

4.3.2. DISEÑO	55
4.3.2.1. Diagrama de clases	55
4.3.2.2. Diagramas de secuencia	56
4.3.2.3. Diagramas de historias de usuario	56
4.3.2.4. Diagrama de actividades	57
4.3.2.5. Diseño de la base de datos	57
4.3.3. DESARROLLO.....	57
4.3.4. PRUEBAS.....	58
4.3.5. DESPLIEGUE	58
5. RESULTADOS.....	63
5.1. INGRESO DE SESIÓN.....	63
5.2. MÓDULOS PARA EL ADMINISTRADOR	64
5.3. MÓDULOS PARA EL RESIDENTE	67
6. CONCLUSIONES Y RECOMENDACIONES.....	73
7. BIBLIOGRAFÍA.....	75
ANEXO A. HISTORIA DE USUARIOS	77
ANEXO B. DIAGRAMAS UML.....	88
ANEXO C. CASOS DE PRUEBA	104

LISTA DE ANEXOS

	pág.
ANEXO A. HISTORIA DE USUARIOS	91
ANEXO B. DIAGRAMAS UML.....	107
ANEXO C. CASOS DE PRUEBA	133

LISTA DE TABLAS

Tabla 1. Software para la administración de propiedad horizontal.....	31
Tabla 2. Cargar el archivo de Excel con datos de los residentes.....	36
Tabla 3. Caso de prueba relacionado con cargar el archivo Excel	45

LISTA DE FIGURAS

	pág.
Figura 1. Metodología incremental implementada en el proyecto	23
Figura 2. Arquitectura de la aplicación	37
Figura 3. Arquitectura de la aplicación en la nube AWS	38
Figura 4. Diagrama de componentes.....	40
Figura 5. Diagrama de clases	41
Figura 6. Diagrama de secuencia, notificación por mensaje de texto	42
Figura 7. Diagrama de historias de usuario	42
Figura 8. Diagrama de actividades, registro residente	43
Figura 9. Modelo Entidad Relación de la aplicación web – primer incremento	44
Figura 10. Ingresar a la consola	46
Figura 11. Creación de un tema en el servicio SNS.....	46
Figura 12. Temas creados en SNS.....	47
Figura 13. Diagrama de clases	49
Figura 14. Diagrama de historias de uso	50
Figura 15. Modelo Entidad Relación de la aplicación web – segundo incremento	51
Figura 16. Certificado de seguridad SSL	52
Figura 17. Creación de almacenamiento en S3	53
Figura 18. Archivos cargados en S3.....	53
Figura 19. Instancia de aplicación en CloudFront	54
Figura 20. Diagrama de clases	56
Figura 21. Diagrama de historias de uso	57
Figura 22. Base de datos creada en el servicio RDS	58
Figura 23. Imagen docker creada en DockerHub	59
Figura 24. Creación del clúster en ECS.....	60
Figura 25. Tarea que corre la aplicación backend creada en el cluster de ECS... ..	60
Figura 26. Registro de la aplicación.....	61

Figura 27. Dirección IP, donde está expuesta la aplicación backend	61
Figura 28. Recursos creados en API Gateway.....	62
Figura 29. Ingreso de sesión a la aplicación	63
Figura 30. Publicación de noticias	64
Figura 31. Administración de la información de los residentes	65
Figura 32. Recordar a morosos por SMS	65
Figura 33. Solicitudes de los residentes	66
Figura 34. Actualización de datos.....	66
Figura 35. Publicaciones de noticias que realiza el administrador	67
Figura 36. Estado de pagos.....	68
Figura 37. Envío de solicitudes al administrador	68
Figura 38. Solicitudes que envían los residentes	69
Figura 39. Actualización de datos de residente	69
Figura 40. Pantallas para celular con funcionalidades de administrador	70
Figura 41. Pantallas para celular con funcionalidades para residentes	71

RESUMEN

El conjunto residencial Hacienda El Rosal, se encuentra ubicado en Funza Cundinamarca, es un conjunto cerrado en donde habitan aproximadamente 1400 residentes. Cuentan con zona de juegos, gimnasio, zonas verdes, sistema de seguridad con token para la entrada, parqueadero especial para visitantes, entre otros. En este orden de ideas, resulta ser un conjunto bastante cómodo en donde habitar.

Sin embargo, en temas de administración, el conjunto se enfrentaba con dos grandes problemas que no sabían cómo solucionar. Existía una dificultad para tener una buena comunicación entre los administradores y los residentes, así como, una mala gestión de cartera. Se planteó dar solución a estos problemas, con la implementación de una aplicación web que facilita la comunicación entre la administración y los residentes y permite el control, seguimiento y recuperación de la cartera.

El presente documento, se realizó para dar a conocer cómo se llevó a cabo el proyecto, indicando el objetivo, el alcance, las limitaciones, la metodología de desarrollo que se implementó y el desarrollo de cada una de las etapas.

INTRODUCCIÓN

El conjunto Residencial Hacienda el Rosal, presentaba varios problemas de administración. La falta de organización de la información de los residentes, como el estado de cuenta del pago de la administración e informes sobre el uso de estos dineros, llevaba al no pago de estas obligaciones y a un aumento en la cartera por deudores morosos. Por otro lado, los residentes tenían dificultades para acceder a cierta información como, por ejemplo, la programación y realización de actividades y reuniones, notificaciones sobre los eventos que se llevaban a cabo dentro del conjunto, entre otros. Como solución a esta problemática se desarrolló una aplicación web con el fin de mejorar la gestión de la cartera y facilitar la comunicación entre la administración y los residentes.

La aplicación que se implementó, le permite al administrador publicar de manera más fácil información relevante y de interés a los residentes. Ellos a su vez, pueden interactuar con dicha información para tener una retroalimentación sobre la gestión que se realiza en el conjunto. La aplicación también, da la posibilidad a los administradores de conocer la cartera de los deudores, permitiendo enviar notificaciones indicando la fecha límite de pago, el saldo de la deuda, los intereses generados o cualquier otra información que sea de interés para un residente moroso.

Para la realización de este proyecto, se implementó la metodología de desarrollo incremental, lo que permitió elaborar la aplicación en corto tiempo y solucionar los problemas presentados en el conjunto. Se realizaron tres incrementos, trabajando para cada uno de ellos las cinco etapas de la metodología: análisis de requerimientos, diseño, codificación y pruebas.

Este documento presenta la siguiente estructura: en el capítulo 1, se planteó el problema a resolver, justificando y resolviendo los objetivos, el alcance y limitaciones del proyecto; el capítulo 2, contiene la explicación de la metodología incremental que se utilizó para el desarrollo del software (aplicación web); el capítulo 3, corresponde al marco teórico, los antecedentes y el marco legal; el capítulo 4, muestra el desarrollo de la metodología planteada; el capítulo 5, presenta los resultados del proyecto; el capítulo 6, las conclusiones y recomendaciones; por último, se listan las referencias bibliográficas citadas en el documento.

1. PLANTEAMIENTO DEL PROBLEMA

1.1. DESCRIPCIÓN DEL PROBLEMA

El conjunto residencial Hacienda El Rosal, ubicado en Funza Cundinamarca, cuenta con 1400 residentes aproximadamente. Es un conjunto bastante grande y la administración se enfrenta a dos grandes problemas, en primer lugar, la comunicación entre el administrador y los residentes se realiza por medio de carteleras que pegan en la entrada del conjunto especificando los anuncios o noticias importantes que son de interés para los residentes, como: la programación y realización de actividades y reuniones, notificaciones sobre los eventos que se llevan a cabo dentro del conjunto, entre otros. Los residentes manifiestan no estar enterados de lo que sucede, en la medida en que no son notificados a tiempo, sino hasta que pasan por la cartelera y ven el anuncio.

En segundo lugar, el conjunto presenta residentes que deben el pago de administración. Los deudores, mencionan que el no pago se debe a la falta de conocimiento sobre las fechas límites de pago y del manejo dado al dinero recaudado, lo que conlleva a que algunos se retrasen o no hagan los pagos y otros, se molesten por no saber con claridad el destino final de los recursos que están aportando y prefieren dejar de pagar.

Como consecuencia de lo anteriormente mencionado, la administración ha notado un aumento en la cartera por los deudores morosos, afectando económicamente al conjunto, por no contar con los recursos necesarios para cubrir sus necesidades. Estos procesos se realizan manualmente haciendo que todo sea más demorado y no tan accesible y confiable de cara a los residentes.

Es un hecho que los residentes de este conjunto se quejaban e incluso realizaban demandas a la administración por el mal servicio que prestaban al momento de

informar, para los residentes era difícil estar enterados sobre la gestión de su cartera y lo que ocurría dentro del conjunto, puesto que no se contaba con los medios o una fuente de información disponible y de fácil acceso para realizar los comunicados.

1.2. FORMULACIÓN DEL PROBLEMA

El conjunto residencial Hacienda el Rosal tiene problemas de comunicación entre los residentes y el administrador. La administración publica los comunicados por medio de carteles ubicados en la entrada del conjunto y las torres, esto hace que los eventos que se realizan no sean conocidos oportunamente por todos los residentes. De igual forma, el administrador no siempre está disponible para atender las diferentes solicitudes de los residentes, causando inconformidad al no obtener una respuesta rápida.

Por otro lado, el control y seguimiento del recaudo y recuperación de la cartera de los deudores morosos carece de estrategias de comunicación para informar a tiempo sobre la deuda y el manejo dado al dinero recaudado, ocasionando el retraso o no pago de la administración por parte de los residentes.

1.3. JUSTIFICACIÓN

Para dar solución a los problemas de comunicación del conjunto residencial, desde la ingeniería, se propuso desarrollar una aplicación web que logre una mejor gestión de cartera y comunicación entre los residentes y el administrador, por medio de la publicación de comunicados, avisos, quejas, sugerencias y notificación personalizada al celular de cada residente, dando a conocer el estado de cuenta de los pagos de administración, la fecha límite de pago, los intereses generados o cualquier otra información que sea de interés para un residente moroso.

La implementación de una aplicación Web en la administración del Conjunto Residencial Hacienda el Rosal de Funza Cundinamarca es relevante por cuanto se convierte en una estrategia para compartir comunicación entre el administrador y los residentes. A través de un módulo de interacción facilita los avisos, notificaciones, quejas, sugerencias, entre otros.

A nivel económico, la implementación de la aplicación en el conjunto residencial, facilita al administrador dar avisos sobre el estado actual de la cartera, de manera personalizada a cada residente, por medio de notificaciones al celular. Con esta acción, el conjunto puede disminuir a mediano plazo el número de deudores morosos de administración, dado que los residentes son informados a tiempo de sus obligaciones.

Respecto a lo tecnológico, esta aplicación es totalmente web, no necesita ninguna instalación, se accede desde cualquier parte y en cualquier navegador, dado que toda la implementación de la aplicación se realizó en la nube, la base de datos, el componente back-end y front-end, utilizando los servicios de AWS.

En el ámbito profesional, con la realización de este proyecto se reforzaron los conocimientos vistos en el transcurso de la carrera. Además de aplicar lo aprendido, también, se trabajaron tecnologías actuales que a nivel laboral tienen alta demanda. Es importante la utilización de estas tecnologías, dado que aumenta las capacidades y perfil laboral.

1.4. OBJETIVOS

1.4.1. Objetivo general

Implementar una aplicación web que facilite una mejor gestión de cartera y comunicación entre los residentes y el administrador del conjunto residencial Hacienda El Rosal de Funza Cundinamarca, por medio de la publicación de

comunicados, avisos, quejas, sugerencias y notificación personalizada al celular de cada residente.

1.4.2. Objetivos específicos

- Establecer los requerimientos para realizar la aplicación, con el aporte de los residentes y la administración en las asambleas que se programen.
- Diseñar la diagramación y la arquitectura para producir un modelo o representación de la aplicación web, utilizando la notación UML.
- Desarrollar la aplicación web teniendo en cuenta los diseños y requerimientos establecidos, para dar solución a los problemas más frecuentes del conjunto residencial, utilizando servicios de Amazon Web Services (AWS).
- Realizar pruebas funcionales que permitan verificar que la aplicación web cumple con las necesidades del conjunto residencial Hacienda El Rosal, a través de los residentes y el administrador.
- Desplegar la aplicación web en la nube de manera que el administrador y los residentes puedan visualizar la aplicación desde cualquier dispositivo, a través de cualquier sitio y navegador.

1.5. ALCANCE Y LIMITACIONES DEL PROYECTO

1.5.1. Alcance

Desarrollo de una aplicación web que permita la comunicación entre los residentes y la administración, así como, el control y seguimiento del pago de administración de conjuntos residenciales, basados en propiedad horizontal. Este software se

aplicó al conjunto residencial Hacienda El Rosal, lo cual contempló la implementación de las siguientes funcionalidades:

- Módulo de seguridad y control de acceso de usuarios al aplicativo. Se manejan diferentes perfiles de acuerdo con la necesidad de cada usuario. Los usuarios ingresan con un nombre de usuario y contraseña, en este caso, para la primera autenticación la cédula en cada campo, luego el usuario puede cambiar la contraseña cuando lo desee. Según el perfil, aparecen las opciones a las que tiene acceso.
- Módulo de interacción entre residentes y administración. Los residentes pueden compartir quejas y/o sugerencias, que se le notifican al administrador, por medio de mensajes de texto, para que ingrese a la aplicación a verlas y así, tener retroalimentación sobre la gestión realizada. También, se permite a los residentes interactuar con el contenido que publica el administrador, sobre información de publicidad, notificaciones y anuncios.
- Módulo de administración de cartera. El administrador puede hacer seguimiento y gestión de la cartera de los propietarios, adjuntando un archivo Excel con datos de los pagos de administración. La aplicación le permite filtrar la información necesaria para emitir notificaciones, por medio de mensajes de texto, sobre el estado de cuenta de los residentes.
- Módulo de administración de publicación de noticias. El administrador publica a través de la aplicación, los diferentes procesos y procedimientos necesarios para el funcionamiento del conjunto. Los residentes consultan estos procedimientos, en el evento que sean de su interés.
- Notificaciones al celular de los pagos y novedades a los residentes. Esta funcionalidad permite a los residentes estar más informados sobre los pagos y novedades del conjunto. Algo muy particular de esta funcionalidad es que, los residentes no necesitan ingresar a la aplicación web, por el contrario, estas notificaciones llegan como mensaje de texto, desde la aplicación. Para

lograr esta funcionalidad, se utilizaron servicios que ofrece AWS (Amazon Web Services) para el manejo de notificaciones.

Para el entregable de esta aplicación web, fue necesario tener en cuenta los siguientes aspectos:

Respecto a la solución tecnológica, se planteó que para la parte visual y de cara al usuario (front-end), el uso de Angular 8 y para la parte del lado del servidor (back-end), se trabajó en el lenguaje de programación Java 8.0, utilizando como marco de trabajo, Spring boot, para la realización de servicios web. Con respecto a la persistencia de los datos, se implementó una base de datos en MySQL. Toda la solución se desplegó en la nube, por medio de los servicios de AWS. Por otro lado, la aplicación web se puede visualizar en escritorio, tableta y celular, es decir, la aplicación es responsiva.

1.5.2. Limitaciones

Las limitaciones de este proyecto son las siguientes:

- Con relación a las tecnologías utilizadas, durante el desarrollo de la aplicación se trabajó en la capa gratuita de AWS que dura un año. Para ello, se trabajó con pocos datos de prueba, es decir, que no se sobrepasó el límite de uso de los servicios para no generar cobro. Cuando se hizo entrega del software al conjunto, se pasó a una capa de pago, por lo tanto, el conjunto debe asumir los pagos del consumo de los servicios.
- Acerca de la información de los pagos de los residentes, la aplicación no está conectada a ningún sistema que otorgue dicha información. El administrador debe cargar un archivo Excel a la aplicación, lo que le ofrece la aplicación es realizar una serie de acciones con cada residente que no ha efectuado el

pago de la administración; una de ellas, es poder enviar notificaciones de mora y fecha límite de pago.

- En cuanto a la accesibilidad de la aplicación web, es necesario tener internet.
- El tiempo estipulado para el desarrollo es de 4 meses.

2. ASPECTOS METODOLÓGICOS

Para la aplicación web, se implementó la 0, con el fin de ejecutar un plan de incrementos que permita entregar valor al software, en corto plazo.

2.1. DESCRIPCIÓN DE LA METODOLOGÍA INCREMENTAL

La metodología incremental se basa en dividir el desarrollo en partes llamadas incrementos. En cada incremento se entrega un producto mínimo viable en el que se implemente alguno de los requerimientos más críticos o prioritarios. A medida que se realizan las entregas se van agregando funcionalidades al software, hasta terminarlo.

Generalmente, el primer programa es un producto fundamental, en donde se trabajan los requerimientos principales, sin proveer muchas características suplementarias. Se desarrolla un plan para el incremento que sigue, que incluya la modificación del producto fundamental, así como, la entrega de características adicionales y más funcionalidad. Este proceso, se repite después de entregar cada incremento, hasta terminar el producto.

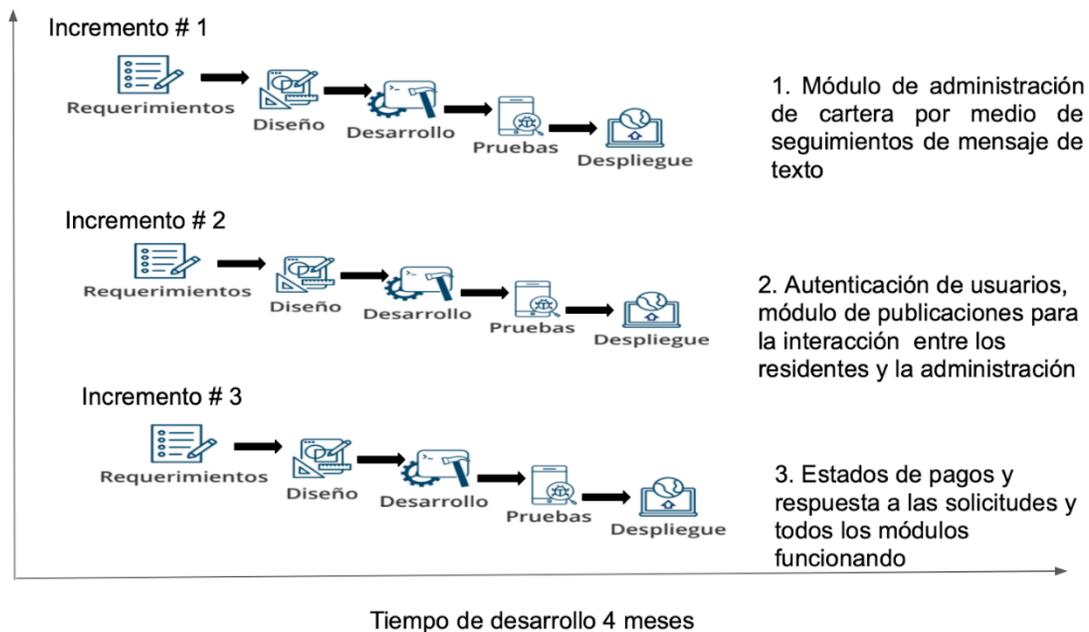
2.2. INCREMENTOS IMPLEMENTADOS EN EL PROYECTO

La **¡Error! No se encuentra el origen de la referencia.**, ilustra los 3 incrementos desarrollados en el proyecto. En cada incremento se entregó un producto funcional, hasta llegar a la aplicación web final.

En el primer incremento, se entregó el módulo de administración de cartera, haciendo el seguimiento de cartera por medio de notificaciones, a través de mensajes de texto, también la publicación de noticias el cual los residentes pueden verlas.

En el segundo, se entregó el módulo de autenticación de los usuarios de la aplicación, el módulo de interacción entre los residentes y la administración recibiendo las solicitudes enviadas, conociendo las quejas, sugerencias y peticiones.

Figura 1. Metodología incremental implementada en el proyecto



Fuente: elaboración propia

En el último incremento, el residente ya puede ver las respuestas a las solicitudes enviadas al administrador y la información de sus pagos, además se entregaron todos los módulos funcionando, con los requerimientos funcionales y no funcionales que se establecieron.

2.3. FASES DE LA METODOLOGÍA INCREMENTAL

Las fases ejecutadas en cada incremento para el desarrollo de la aplicación web, se describen a continuación. Todas las definiciones dadas se basan en la referencia

del autor (Partners Académicos, OBS Business School). Cada fase se validó con el cliente, para asegurar la continuidad de la siguiente.

2.3.1. Análisis de requerimientos

En esta etapa se analizaron los problemas y necesidades específicos del conjunto residencial Hacienda el Rosal, para determinar las características del software a desarrollar, recopilando y documentando los requerimientos funcionales y no funcionales.

Entregables:

- Historias de usuario.

2.3.2. Diseño

En esta etapa, se elaboró el correspondiente diseño de diagramas UML, base de datos y pantallas mediante mockups. De igual forma, se analizaron las herramientas usadas en la etapa de codificación.

Entregables:

- Diagramas estáticos UML:
 - Diagrama de clases.
 - Diagrama de componentes.
 - Diagrama de arquitectura de la aplicación en AWS.
 - Diagrama de arquitectura de la aplicación.
- Diagramas dinámicos UML.
 - Diagrama de secuencia.
 - Diagrama de casos de uso.
 - Diagrama de actividades.
- Diseño de la base de datos.

- Diagrama entidad relación.

2.3.3. Desarrollo

En esta etapa se realizó el desarrollo de la aplicación web, tomando en cuenta todos los requerimientos y diseños que se establecieron en la anterior etapa.

Actividades a realizar:

- Realización del front-end de la aplicación.
- Implementación de la base de datos en MySQL.
- Realización del microservicio back-end y los recursos a exponer.
- Pruebas de conexión entre back-end y front-end.

2.3.4. Pruebas

Al realizar las pruebas de la aplicación web, se verificó el funcionamiento y cumplimiento de todos los requerimientos establecidos por el conjunto residencial. Igualmente, se entregó el diseño, implementación y ejecución de las pruebas con su debida documentación.

Entregables:

- Casos de prueba.
- Ejecución de pruebas funcionales y no funcionales.
- Solución de bugs encontrados.

2.3.5. Despliegue

Luego de tener todos los casos de prueba ejecutados y pasados, en esta etapa se realizó el despliegue de la aplicación en la nube, utilizando los servicios de AWS.

De tal manera, que la aplicación pudo ser utilizada por el administrador encargado y los residentes del conjunto residencial Hacienda El Rosal.

3. MARCO DE REFERENCIA

3.1. MARCO TEÓRICO

Para poder entender el contexto en el que se enmarcó este proyecto, es necesario conocer algunos términos y conceptos importantes mencionados durante la elaboración de la aplicación. Aspectos, referentes al conjunto residencial y al desarrollo de software.

3.1.1. CONJUNTO RESIDENCIAL HACIENDA EL ROSAL

Descripción: el conjunto residencial Hacienda el Rosal se encuentra ubicado en la calle 12 # 5 – 35, del municipio de Funza, en Cundinamarca. Es un conjunto cerrado, en donde habitan aproximadamente 1400 residentes, constituido por 17 torres con 5 pisos, en donde hay 4 apartamentos por piso. Cuenta con portería, salón comunal, parque infantil, zonas verdes, zona BBQ, gimnasio, canchas deportivas, parqueadero especial para visitantes, sistema de seguridad con token para la entrada, entre otros.

Administración: la administración está a cargo de la administradora, elegida en la asamblea de propietarios, que lleva el control y seguimiento de todos los procesos que se realizan dentro del conjunto, así como, de hacer cumplir las reglas establecidas. Se maneja un horario de atención, que se da a conocer en las carteleras de la entrada al conjunto.

La administración enfoca el trabajo en dos temas: la comunicación con los residentes y la gestión de cartera.

La comunicación entre los residentes y la administración se realiza por medio de carteleras que pegan en la entrada, especificando los anuncios o noticias importantes que son de interés para los residentes, como: la programación y realización de actividades y reuniones, notificaciones sobre los eventos que se llevan a cabo dentro del conjunto, entre otros. Los residentes ingresan, pasan por la cartelera y ven el anuncio.

En cuanto al pago de la administración, el recaudo se realiza por medio de una cuenta bancaria del banco AV Villas. Los residentes consignan o hacen una transferencia por el valor de la cuota. Los pagos de las zonas comunes como: parqueadero, alquiler de salón comunal, salón de juegos y gimnasio, se cancela directamente en la administración del conjunto.

Para el seguimiento del recaudo de cartera, la administradora revisa el estado de cuenta de cada uno de los residentes, aquellos que se encuentran con deuda son informados por medio del póster de notificaciones, donde se publica el número de los apartamentos que se encuentran en estado de mora.

3.1.2. CONCEPTOS DE DESARROLLO DE SOFTWARE

3.1.2.1. Aplicación web: es un software que se construye en un lenguaje de programación que permite que un servidor interprete el código y despliegue la información en un navegador web a través de Internet. Las aplicaciones web son dinámicas, no un simple conjunto de páginas estáticas, que solo se pueden consultar o descargar. (Mateu, 2004)

La herramienta que se desarrolló para el conjunto residencial Hacienda El Rosal, en donde la administración y los residentes interactúan con una cierta funcionalidad, es una aplicación web.

3.1.2.2. Aplicación web responsiva: es un nuevo enfoque para que las aplicaciones se adapten de forma dinámica a diferentes resoluciones y dispositivos. Este enfoque se centra en el contenido y en el cliente, específicamente, en su experiencia de usuario. (Vega, s.f)

Este concepto se tuvo en cuenta en el proyecto, dado que esta aplicación web se visualiza en escritorio, tableta y celular, permitiendo a los usuarios ver el diseño de forma dinámica y fluida.

3.1.2.3. Front-end y back-end: front-end está del lado del cliente, es la parte visual de la aplicación web que interactúa con los usuarios. Back-end corre del lado del servidor, es la parte que se conecta con la base de datos y el servidor, en donde se encuentra toda la lógica del negocio. Hacen referencia a la separación entre la capa de presentación y la de acceso a datos, respectivamente. (Chapaval, 2017)

El software implementado contó con dos componentes, uno para el back-end y otro para el front-end, que se comunican por medio de peticiones HTTP.

3.1.2.4. Angular: es una plataforma y marco de trabajo para desarrollar aplicaciones web SPA (Single Page Application) o aplicaciones de una sola página utilizando HTML, TypeScript o JavaScript. (Angular, s.f)

La parte front-end de la aplicación se realizó utilizando este marco de trabajo de JavaScript, por ser una tecnología muy fácil de implementar, brinda ciertas características a la aplicación como: ser SPA, buen manejo de rutas, gestión de dependencias, realización de llamadas al back-end, para hacer pruebas, entre otras.

3.1.2.5. Aplicaciones SPA: Single Page Application o aplicación de una sola página, en la que la navegación entre secciones y páginas y la carga de datos, se realiza

de forma dinámica y asíncrona haciendo peticiones al servidor (back-end con un API REST), sin refrescar la página. (Robles, 2017).

La aplicación a desarrollar, contemplará componentes visuales re utilizables, por lo tanto, al hacer uso de Angular para la parte front-end, automáticamente la aplicación será SPA.

3.1.2.6. Java: es un lenguaje de programación y una plataforma informática muy utilizada para el desarrollo de aplicaciones que corren sobre diferentes dispositivos desde portátiles, consolas para juegos, súper computadoras, teléfonos móviles hasta Internet. Java es un lenguaje fuertemente tipado, funciona independiente de la plataforma, es rápido, seguro y fiable. (Oracle CorporationJava, s.f)

Para el proyecto a desarrollar, se utilizó este lenguaje de programación dado que se ha trabajado durante toda la carrera, por lo tanto, se tiene más conocimiento y manejo. En cuanto a la parte técnica, java permite la utilización de marcos de trabajo muy ligeros, lo cual, es necesario en esta aplicación, para la creación de servicios web.

3.1.2.7. Spring Boot: es un marco de trabajo de código abierto basado en Java que facilita la creación y funcionamiento de los micro servicios. Se utiliza para construir aplicaciones pequeñas, independientes y listas para ejecutarse de una manera fácil y casi automática. (Spring, s.f)

Spring Boot, fue el marco de trabajo central para el desarrollo back-end de la aplicación, spring boot ofrece una fácil implementación de servicios web, la configuración de inicio a fin del proyecto y la creación del micro servicio desplegado en la nube.

3.1.2.8. Arquitectura de micro servicios: es una arquitectura que permite construir micro servicios, cuyo enfoque es que el código de la aplicación se implementa en piezas pequeñas y manejables, independientes unas de otras. (Spring, s.f)

En el presente proyecto, se trabajó con una arquitectura de software de micro servicios. Esta arquitectura permite realizar componentes diferentes de acuerdo a cómo se estructuren los temas de negocio, esto ofrece tener una mejor mantenibilidad al software.

3.1.2.9. AWS (Amazon Web Services): es una plataforma que cuenta con herramientas y servicios para realizar actividades en la nube como: almacenamiento, tratamiento de datos, desarrollo de aplicaciones web, inteligencia de negocios, seguridad y control de acceso, entre otros. (AWS, 2019)

AWS, se usó en el desarrollo del proyecto para poder crear e implementar el hospedaje de la aplicación web en la nube, el almacenamiento de la base de datos, la utilización de servicios para notificaciones, entre otros, administrando de manera más flexible toda la implementación de las necesidades específicas de la aplicación.

3.2. ESTADO DEL ARTE

Actualmente, existe una gran variedad de aplicaciones web para la administración de conjuntos residenciales en Bogotá, algunas de estas y sus características se listan en la Tabla 1.

Tabla 1. Software para la administración de propiedad horizontal

APLICACIONES WEB	SOFTWARE WEB PARA LA ADMINISTRACIÓN Y GERENCIA DE PROPIEDAD HORIZONTAL	PHI-HORIZONTAL	COPROPIEDAD
DESCRIPCIÓN	Es una aplicación totalmente web, que se puede comprar por mercado libre. Cuenta con funciones administrativas, se pueden manejar, encuestas, foros, comunicados, casos, proyectos y registro de proveedores, entre otras funcionalidades. (MercadoLibre, 2013)	Es una aplicación web que agiliza la gestión y la comunicación de uno o varios conjuntos residenciales en un solo sitio. Con Phi Horizontal, como administrador puede enviar constantemente información a los celulares o móviles de los residentes. (PHI-Horizontal, s.f)	Es un software para administradores de propiedades horizontales en Colombia. Garantiza una comunicación fluida y transparente con los residentes, propietarios y consejeros de la copropiedad. El administrador cuenta con información del conjunto o edificio y soluciones tecnológicas que lo apoyan en el manejo eficiente de sus gestiones. (Copropiedad, 2019)
ACCESIBILIDAD	Esta aplicación se accede por internet. Maneja diferentes niveles de usuarios (propietarios, consejo, administrador, proveedores y porteros), cada uno con diferentes funciones dentro de la aplicación web.	Esta aplicación se accede por internet. Puede acceder solo el administrador y los residentes del conjunto.	El ingreso a esta aplicación es totalmente web. Puede ingresar el administrador, residentes y copropietarios.
LICENCIA	Se adquiere una licencia por un año, sin importar la cantidad de propietarios.	Se puede probar por 30 días gratis. Para uso de más tiempo con todas las funcionalidades es necesario comprar una licencia.	Es necesario adquirir la herramienta por medio de la compra de una licencia.

... continuación tabla1

APLICACIONES WEB	SOFTWARE WEB PARA LA ADMINISTRACIÓN Y GERENCIA DE PROPIEDAD HORIZONTAL	PHI-HORIZONTAL	COPROPIEDAD
COMUNICACIÓN	Tiene los módulos de novedades, encuestas, foros, comunicados, estos se visualizan en la plataforma, la cual da acceso a los residentes, permitiendo a los usuarios la interacción con la aplicación.	Permite al administrador enviar información a los residentes por medio de email. Además, la publicación de avisos clasificados, para los grupos a los cuales van dirigidos.	El administrador publica información de interés en la página y todos los residentes son notificados vía correo.
GESTIÓN DE CARTERA	El módulo de cobro, envía las cuentas por correo electrónico a los propietarios.	Tiene un módulo donde se administran las zonas comunes y parqueaderos. Envía archivos a los residentes.	El administrador informa a cada uno de los residentes el monto de la deuda con la administración por medio del correo electrónico. El administrador puede publicar en el portal de la copropiedad el presupuesto de gastos y de ingresos del conjunto.

Fuente: elaboración propia.

De acuerdo con la revisión realizada de las aplicaciones y sus funcionalidades, se puede deducir que todas tienen en cuenta la comunicación entre el administrador y los residentes, ya sea por medio de correo electrónico, publicación de información en la plataforma o envío de mensaje al teléfono celular. De igual forma, contemplan la gestión de cartera por medio del seguimiento a los pagos por concepto de administración realizados por los propietarios. Sin embargo, todas ellas requieren la compra de una licencia para poder tener acceso a las funcionalidades, lo que no está previsto por la administración ni por los residentes del conjunto Hacienda el Rosal.

Por lo tanto, se requirió implementar una aplicación web para la administración de propiedad horizontal a la medida, con el fin de cubrir todas las necesidades del conjunto; permitiendo a los residentes y administrador una interacción constante, facilitando y aumentando la transparencia en la gestión de la administración, mejorando la comunicación y la gestión de cartera, sin tener que incurrir en gastos y teniendo en cuenta las funcionalidades de algunas aplicaciones presentes en el mercado.

Esta aplicación cuenta con la mejor tecnología para gestionar la información administrativa de forma rápida, segura y eficaz, de manera que facilita los procesos para mejorar la gestión requerida en el conjunto residencial.

3.3. MARCO LEGAL

A continuación, se especifican algunas leyes que aplican al desarrollo de la aplicación Web desarrollada.

Ley 675 de 2001: reglamento de propiedad horizontal. Esta ley es la que regula los derechos de copropiedades de los terrenos, de los bienes comunes para así garantizar la seguridad y la convivencia en las propiedades horizontales (Ley 675, 2001).

Ley 1581 de 2012: constituye el marco general de la protección de los datos personales en Colombia, se deben reglamentar con los aspectos relacionados, junto con la autoridad del titular de la información, para que sea manejados los datos personales (Ley 1581, 2012).

Ley 23 de 1982: es la ley de derechos de autor que contempla los derechos morales y patrimoniales del software (Ley 23, 1982).

4. DESARROLLO DEL PROYECTO

Para el desarrollo de la aplicación web, se implementó la metodología de desarrollo incremental. El desarrollo del proyecto se dividió en tres incrementos de la siguiente manera:

- **Incremento 1:** módulo de administración de residentes y SMS (Servicio de mensajes cortos), haciendo el seguimiento de cartera por medio de notificaciones, a través de mensajes de texto, módulo de Recordar morosos por SMS y módulo de publicaciones en donde el administrador publica noticias del conjunto.
- **Incremento 2:** autenticación de usuarios en la aplicación, módulo de interacción entre los residentes y la administración, permitiendo dar notificación de quejas, sugerencias y peticiones. A su vez los residentes ya pueden comentar las publicaciones que realiza el administrador.
- **Incremento 3:** ver respuesta de las solicitudes enviadas al administrador, y la información de pago por parte de los residentes, además todos los módulos funcionando, con los requerimientos funcionales y no funcionales que se establecieron.

A continuación, se describe cada uno de los incrementos.

4.1. INCREMENTO 1

En el primer incremento, se implementó el módulo de administración de residentes y SMS, haciendo el seguimiento de cartera por medio de notificaciones, a través de mensajes de texto y módulo de publicación de noticias. Para ello se ejecutaron las siguientes etapas:

4.1.1. ANÁLISIS DE REQUERIMIENTOS

Para la entrega del primer producto mínimo viable al conjunto residencial, se trabajó en la notificación de noticias importantes acerca del estado de la cartera a los residentes, por medio de publicaciones y mensajes de texto, para que la comunicación fuera instantánea. En primera instancia, se definieron los requisitos funcionales, no funcionales y los usuarios. Con base en los requisitos funcionales se elaboraron las historias de usuario.

4.1.1.1. Historias de usuario: para este incremento se elaboraron las siguientes historias de usuario:

- Relacionadas con la notificación por mensaje de texto a los residentes del conjunto:
 - Cargar el archivo de excel con datos de los residentes.
 - Guardar los datos de los residentes en la base de datos.
 - Enviar notificaciones por mensajes de texto.
 - Enviar notificaciones por mensajes de texto de manera personalizada.
 - Agregar un residente nuevo en la aplicación web.
 - Editar datos de cada residente registrado en la aplicación.
 - Filtrar por torre y apartamento los residentes del conjunto.
- Relacionadas con el seguimiento de cartera para recordar a residentes morosos a través de notificaciones por mensajes de texto:
 - Filtrar residentes en estado mora por pago de administración.
 - Notificar por mensajes de texto a todos los residentes en estado de mora a la vez.
 - Notificar por mensajes de texto a los residentes morosos de manera personalizada.
- Relacionadas con la publicación de anuncios:
 - Realizar publicaciones de novedades y anuncios importantes en la aplicación web acerca de lo que ocurre dentro del conjunto.

En la Tabla 2, se muestra la historia de usuario cargar el archivo de excel con datos de los residentes. Las demás historias de usuario de este incremento se encuentran en el Anexo A.

Tabla 2. Cargar el archivo de Excel con datos de los residentes

Número: 1		Usuario: Administrador					
Nombre de la historia de usuario: cargar el archivo de Excel con los datos de los residentes.							
Prioridad en negocio: Alta				Riesgo en desarrollo: Baja			
Programador responsable: Leidy Yohana Rodríguez Fuentes							
Descripción: yo como administrador deseo cargar un archivo Excel con los datos básicos de los residentes, para poderlos visualizar y registrar en la aplicación.							
Criterios de aceptación:							
1. La aplicación debe ser capaz de adjuntar archivos solamente Excel.							
2. La estructura del archivo Excel debe ser la siguiente:							
torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda	
1	101	1234567890	Katherine Aguirre Tique	3124922928	1	70000	
1	102	1234567896	Leidy Rodriguez Fuentes	3227525187	2	140000	
2	101	3456787654	Pepito Perez	3227623418	0	0	
3	404	3456543456	Laura Sarmiento	3125830084	4	280000	
Cada nombre de cada columna debe ser igual a la estructura definida anteriormente.							
3. En caso de no haber meses en deuda, deben colocarse 0 en meses deuda y total deuda.							
4. El administrador debe poder visualizar los registros cargados en la aplicación web.							
5. Los datos que deben aparecer en la aplicación web de cada residente son: torre apartamento, documento, nombre celular.							

Fuente: elaboración propia.

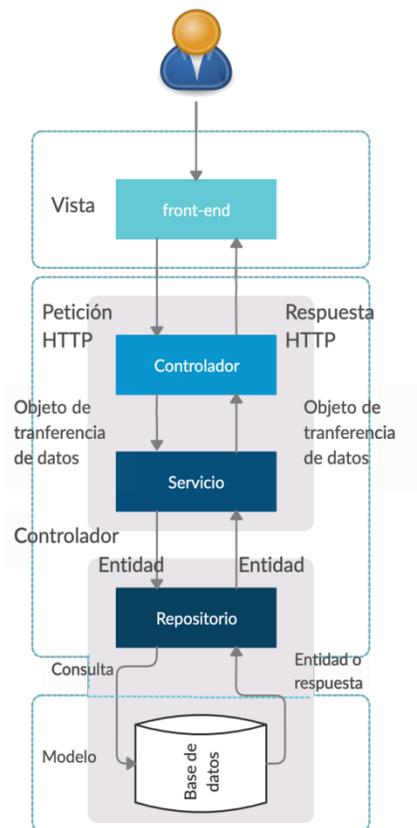
4.1.1.2. Requerimientos no funcionales: se definieron los requisitos no funcionales, que representan características generales y restricciones del sistema.

- La aplicación debe ser accesible en cualquier dispositivo ya sea Tablet, celular y computador de escritorio.
- La aplicación debe contener la combinación de colores que identifican al conjunto residencial que son: morado, amarillo y azul.
- El envío de los mensajes de texto no debe superar el tiempo de 5 segundos.
- La aplicación debe ser accesible las 24 horas y los 7 días a la semana.

4.1.2. DISEÑO

4.1.2.1. Arquitectura de la aplicación: esta aplicación web se desarrolló utilizando el patrón MVC (Modelo Vista Controlador), dado que se tiene una aplicación front-end, otra back-end y se maneja una base de datos. Como se muestra en la **¡Error! No se encuentra el origen de la referencia.**, la vista se comunica con el controlador por medio de peticiones HTTP. El controlador llama a los servicios enviando los datos que llegan de la vista; el servicio se encarga de manipular estos datos y realizar la lógica. Luego los datos son enviados al repositorio, que se encarga de ejecutar las consultas en la base de datos.

Figura 2. Arquitectura de la aplicación

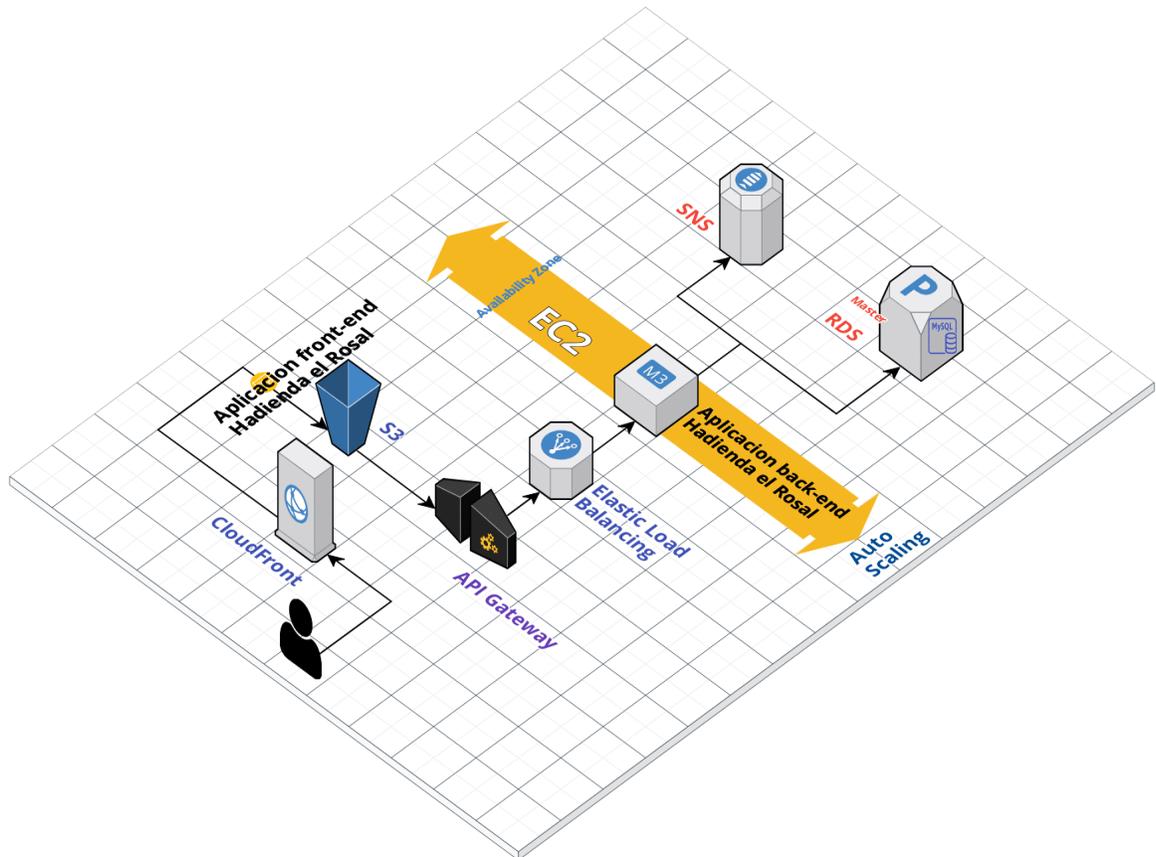


Fuente: elaboración propia

4.1.2.2. Arquitectura de la aplicación en la nube AWS: la

4.1.2.3. , hace referencia al modelo de construcción de la aplicación web en la nube, utilizando los servicios de Amazon Web Services.

Figura 3. Arquitectura de la aplicación en la nube AWS



Fuente: elaboración propia

El cliente, ingresa una URL proporcionada por el servicio CloudFront, el cual es un servicio web que acelera la distribución de contenido web estático y dinámico, como archivos .html, .css, .js y de imágenes, a los usuarios. CloudFront redirige al servicio S3 (Amazon Simple Storage Service), que es un servicio de almacenamiento para Internet. En este servicio S3, se encuentra almacenada la aplicación web front-end, hecha en Angular.

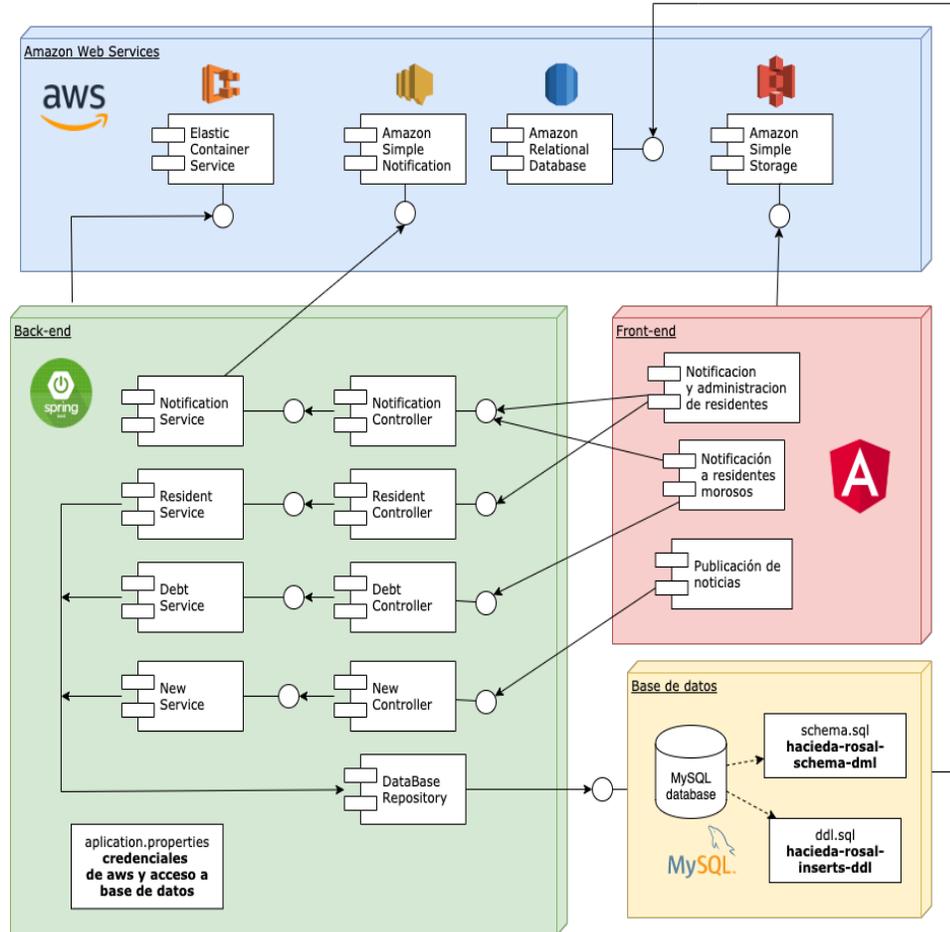
Para realizar las peticiones HTTP por REST, el front-end tiene apuntamientos a recursos expuestos por el servicio API Gateway. Este servicio es para la creación, publicación, mantenimiento, monitorización y protección de la API de REST y WebSocket, a cualquier escala. Los recursos expuestos del API Gateway, apuntan al servicio Elastic Load Balancing (balanceador de carga), que distribuye aplicaciones entrantes o tráfico de red, a través de múltiples destinos; en este caso el destino es el microservicio back-end, que corre en una máquina virtual del servicio EC2 (Amazon Elastic Compute Cloud), que sirve para lanzar servidores virtuales sin la necesidad de invertir en hardware. Este microservicio desarrollado en Spring Boot Java, tiene conexión directa con el servicio RDS (Relational Data Service), en donde se aloja la base de datos MySQL de la aplicación y con el servicio SNS (Simple Notification Service), en donde se administran todas las notificaciones por mensaje de texto que solicita la aplicación. La conexión con SNS es posible a través del SDK (kit de desarrollo de software) de Amazon Web Services, expuesto para el lenguaje de programación Java.

Para el diseño de la aplicación web se realizaron los siguientes diagramas UML.

4.1.2.4. Diagrama de componentes: en la ¡Error! No se encuentra el origen de la referencia., se observa cada uno de los artefactos con sus correspondientes tecnologías y componentes.

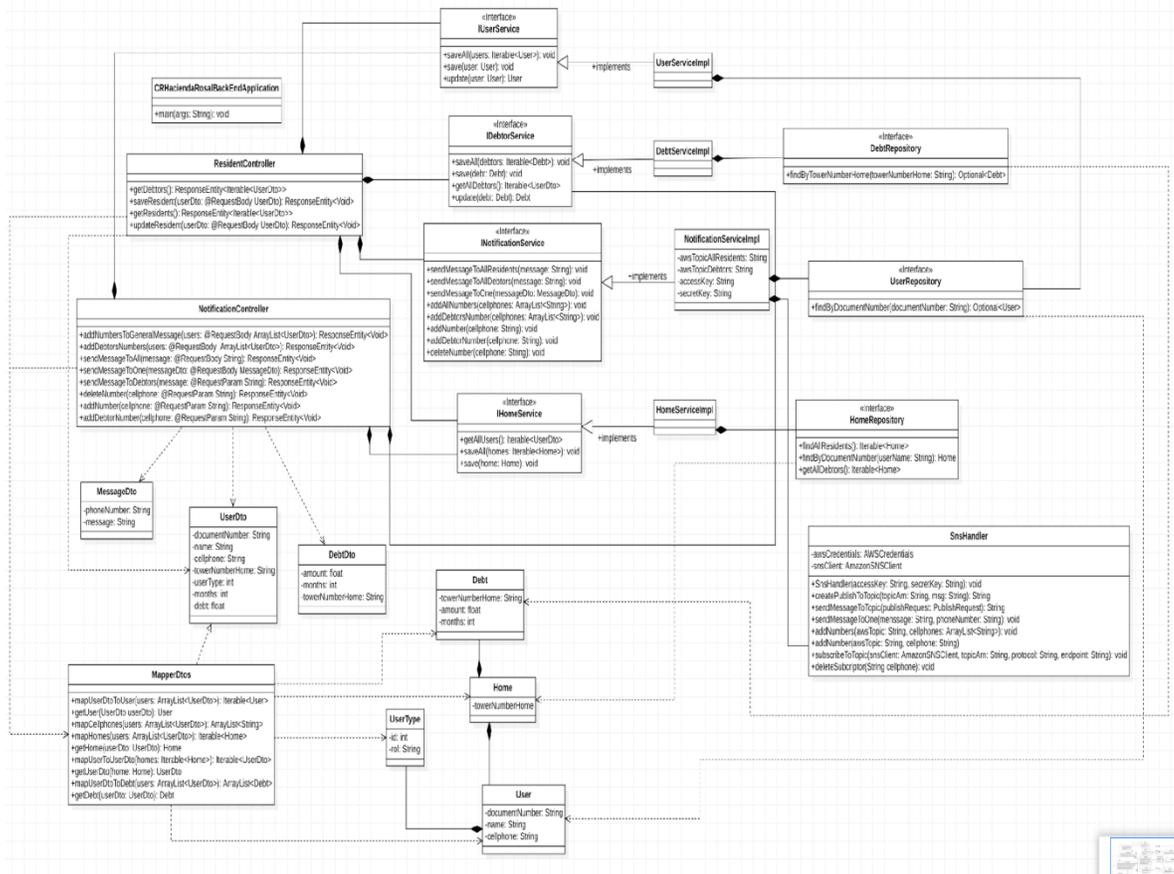
4.1.2.5. Diagrama de clases: el diagrama de clases que se muestra en la ¡Error! No se encuentra el origen de la referencia., se realizó pensando en el lenguaje de programación Java, en donde se tienen controladores, servicios, entidades, DTOs (Objetos de transferencia de datos) y repositorios.

Figura 4. Diagrama de componentes



Fuente: elaboración propia

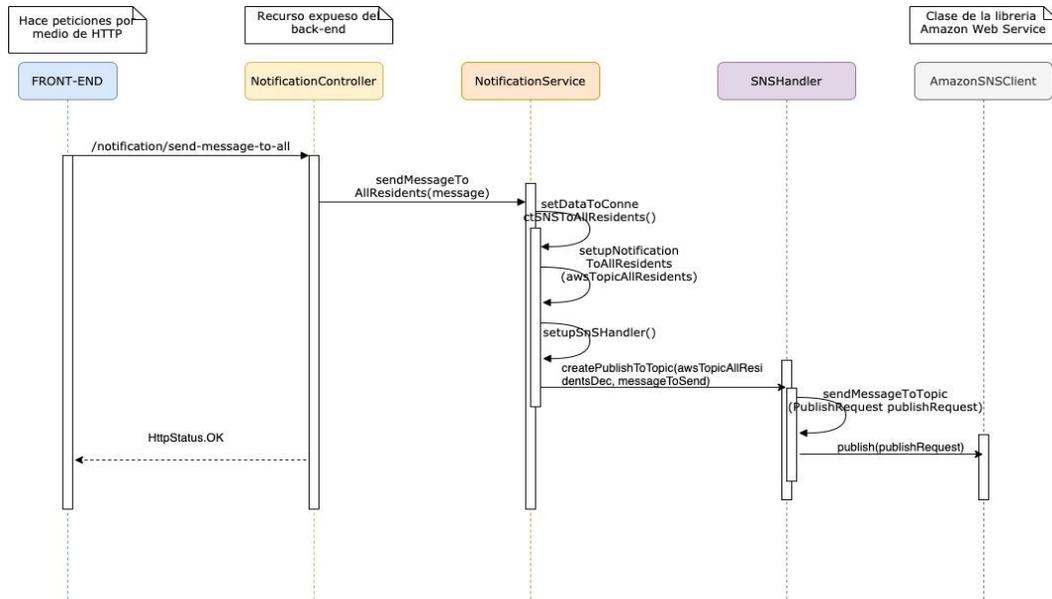
Figura 5. Diagrama de clases



Fuente: elaboración propia

4.1.2.6. Diagrama de secuencia: modela los procesos y comportamientos más importantes de la aplicación web. En la **¡Error! No se encuentra el origen de la referencia,** se muestra el diagrama de secuencia de la Notificación por mensaje de texto a todos los residentes del conjunto. Los demás se encuentran en el Anexo B.

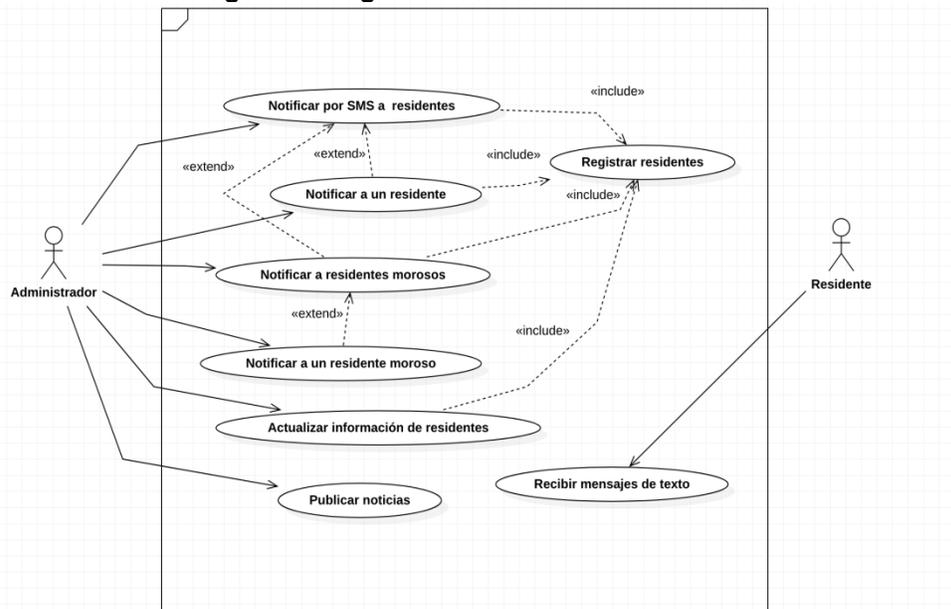
Figura 6. Diagrama de secuencia, notificación por mensaje de texto



Fuente: elaboración propia

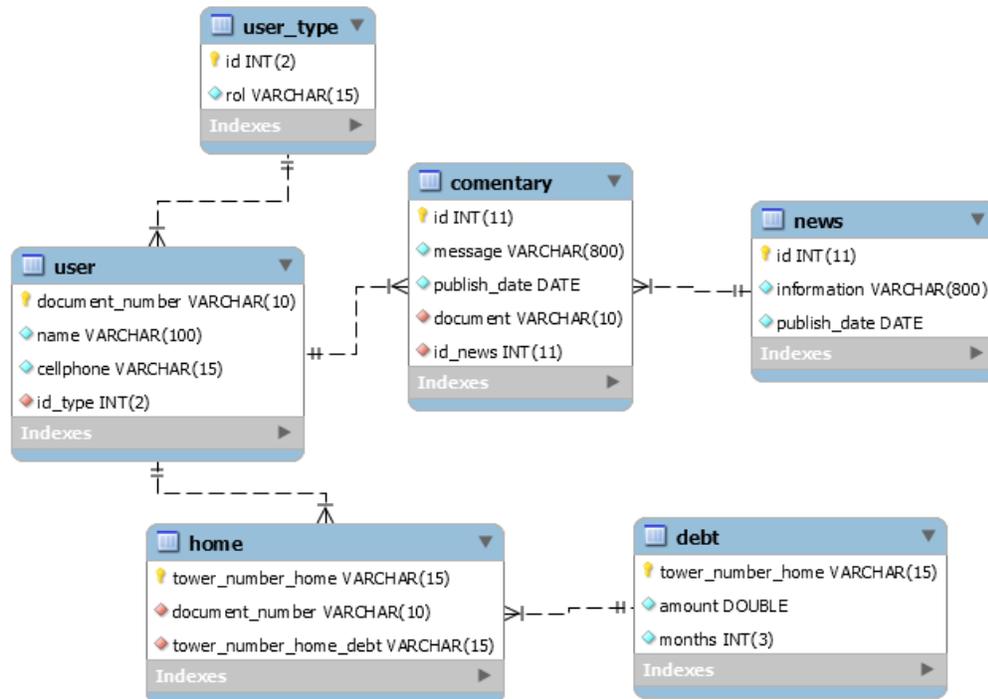
4.1.2.7. Diagrama de historias de usuario: el siguiente diagrama de historias de usuario modela las acciones que realiza cada rol.

Figura 7. Diagrama de historias de usuario



Fuente: elaboración propia

Figura 9. Modelo Entidad Relación de la aplicación web – primer incremento



Fuente: elaboración propia

4.1.3. DESARROLLO

Para el desarrollo de esta aplicación se utilizó el lenguaje de programación Java, tomando como marco de trabajo Spring Boot, para la realización del microservicio backend. Para la aplicación front-end, se utilizó el marco de trabajo de Javascript Angular. Para este primer incremento, se desarrollaron las funcionalidades descritas en las historias de usuario.

El código fuente de la aplicación web se encuentra en el GitHub. Existen dos proyectos: uno para el frontend <https://github.com/Joanne86/c-r-hacienda-rosal-front-end> y otro para el backend <https://github.com/Joanne86/c-r-hacienda-rosal-back-end>

4.1.4. PRUEBAS

Para la validación de la correcta funcionalidad de la aplicación web, se realizaron y validaron los casos de prueba para cada una de las historias de usuario desarrolladas. En la Tabla 3, se muestra el caso de prueba relacionado con la carga del archivo de excel con los datos de los residentes. Los demás casos de prueba, se consignaron en el Anexo C.

Tabla 3. Caso de prueba relacionado con cargar el archivo Excel

Número de caso de prueba: 1		Prioridad de la prueba: Alta																													
Historia de usuario: Cargar el archivo de Excel con los datos de los residentes.																															
Descripción del caso de prueba: Cargar correctamente un archivo Excel en la aplicación.																															
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido):																													
Pre-requisitos: 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo de Administración de residentes y SMS. 3. No tener ningún residente registrado en la aplicación.		Datos de prueba: 1. Documento Excel con los siguientes campos: <table border="1" data-bbox="933 940 1469 1008"> <thead> <tr> <th>torre</th> <th>apartamento</th> <th>documento</th> <th>nombre</th> <th>celular</th> <th>meses_deuda</th> <th>total_deuda</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>102</td> <td>123456789</td> <td>David Rodríguez Fuentes</td> <td>3227623418</td> <td>1</td> <td>70000</td> </tr> <tr> <td>2</td> <td>101</td> <td>543234567</td> <td>Katherine Aguirre Tique</td> <td>3124922928</td> <td>2</td> <td>140000</td> </tr> <tr> <td>3</td> <td>404</td> <td>432345432</td> <td>Pepito Perez</td> <td>3125256290</td> <td>0</td> <td>0</td> </tr> </tbody> </table> (no importa el nombre del archivo Excel)		torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda	1	102	123456789	David Rodríguez Fuentes	3227623418	1	70000	2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000	3	404	432345432	Pepito Perez	3125256290	0	0
torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda																									
1	102	123456789	David Rodríguez Fuentes	3227623418	1	70000																									
2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000																									
3	404	432345432	Pepito Perez	3125256290	0	0																									
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)																												
Seleccionar la opción de adjuntar archivo	Observar una pantalla para elegir el archivo Excel.	Se observa la pantalla para adjuntar archivo Excel.																													
Seleccionar el archivo Excel de prueba	Observar el nombre del archivo en la aplicación	Se observó nombre del archivo en la aplicación																													
Dar clic en la opción cargar	Observar una barra de espera cuando se esté cargado el archivo en la aplicación y observar una tabla con los datos cargados.	Se observó una barra de espera cuando se cargó el archivo en la aplicación y luego se observó una tabla con los datos cargados																													

Fuente: elaboración propia

4.1.5. DESPLIEGUE

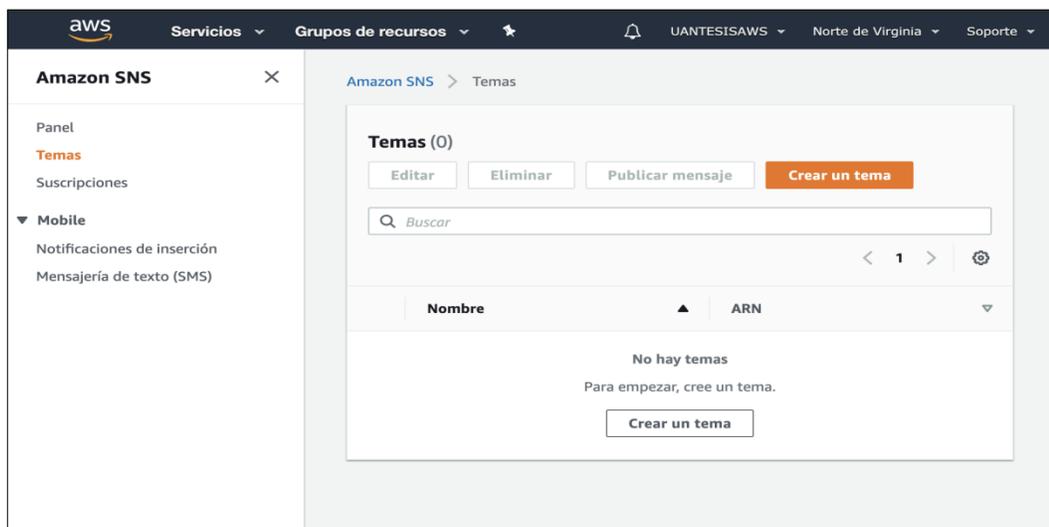
En esta etapa se utilizaron los servicios de Amazon Web Services para desplegar la aplicación web. Uno de los servicios usados para este primer incremento es el

Simple Notification Service, Servicio de Notificación que proporciona una infraestructura de bajo costo para la entrega masiva de mensajes, principalmente a usuarios móviles.

Para esta aplicación fue necesario implementar dos temas, un tema en este servicio se refiere a un grupo de comunidad, en este caso, números de teléfonos celulares que se relacionan con un mismo interés. Se creó un tema llamado “Residentes”, en el que se guardaron todos los números de teléfono celular de todos los residentes del conjunto y otro llamado “Deudores”, para todos los de los residentes que deben los pagos de administración.

La creación de los temas se realizó de la siguiente manera: ingresar a la consola de administración y luego entrar al servicio SNS. Por ahora no hay ningún tema creado, para crearlo simplemente se da clic en la opción “crear tema”, como se observa en la **¡Error! No se encuentra el origen de la referencia..**

Figura 10. Ingresar a la consola



Fuente: elaboración propia

Luego, se establece el nombre del tema “Residentes” y por último se hace clic en la opción “crear”, como se muestra en la **¡Error! No se encuentra el origen de la referencia..**

Figura 11. Creación de un tema en el servicio SNS

aws Servicios Grupos de recursos

Nombre
Residentes
Máximo de 256 caracteres. Puede incluir caracteres alfanuméricos, guiones (-) y guiones bajos (_).

Nombre para visualización - *opcional*
Para utilizar este tema con suscripciones a SMS, escriba un nombre para visualización. Solo se muestran los primeros 10 caracteres en un mensaje SMS. [Información](#)
Mi Tema
Máximo de 100 caracteres, incluidos guiones (-) y guiones bajos (_).

► **Cifrado - *opcional***
Amazon SNS ofrece el cifrado en tránsito de forma predeterminada. Habilitar el cifrado del lado del servidor añade el cifrado en reposo a su tema.

► **Política de acceso - *opcional***
Esta política define quién puede obtener acceso a su tema. De forma predeterminada, solo el propietario del tema puede publicar contenido o suscribirse al tema. [Información](#)

► **Política de reintentos de entrega (HTTP/S) - *opcional***
La política define cómo Amazon SNS reintenta las entregas fallidas en los puntos de enlace HTTP/S. Para modificar la configuración predeterminada, amplíe esta sección. [Información](#)

► **Registro del estado de entrega - *opcional***
Esta configuración establece el registro del estado de entrega del mensaje en CloudWatch Logs. [Información](#)

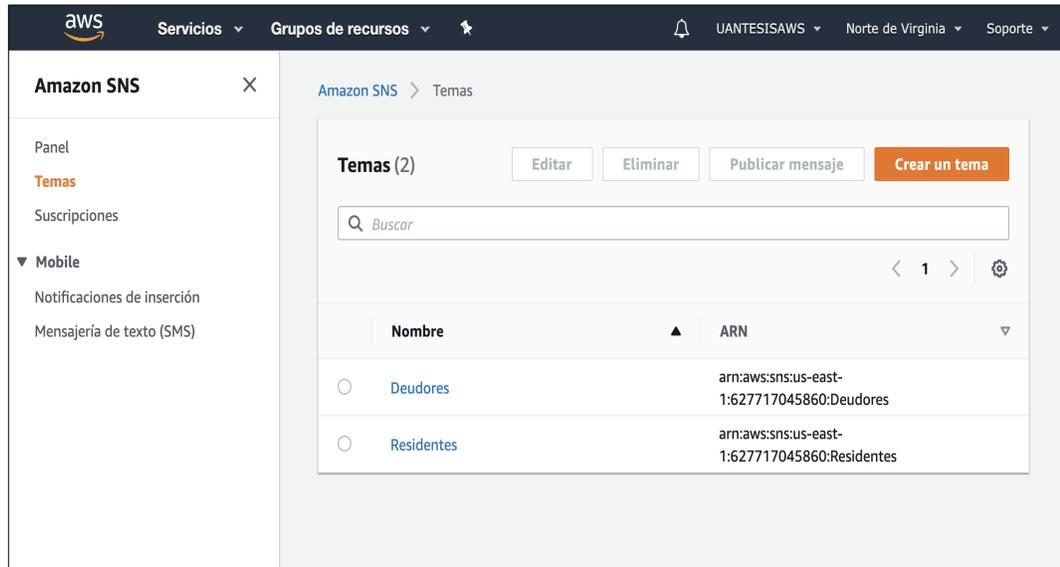
► **Etiquetas - *opcional***
Una etiqueta es una marca de metadatos que puede asignarse a un tema de Amazon SNS. Cada etiqueta consta de una clave y un valor opcional. Las etiquetas pueden utilizarse para buscar y filtrar temas y realizar un seguimiento de los costos. [Más información](#)

Cancelar **Crear un tema**

Fuente: elaboración propia

Aparece el tema creado dentro de la lista de temas, como se observa en la **¡Error!**
No se encuentra el origen de la referencia.. El mismo procedimiento se realizó para el tema “Deudores”.

Figura 12. Temas creados en SNS



Fuente: elaboración propia

4.2. INCREMENTO 2

En el segundo incremento, se implementó la autenticación de usuarios, el módulo de interacción entre los residentes del conjunto y la administración, el cual permite dar notificaciones de quejas y sugerencias. A su vez, los residentes pueden ver las publicaciones que realizan la administración y comentar sobre ellas. Para esto, se ejecutaron las siguientes etapas:

4.2.1. ANÁLISIS DE REQUERIMIENTOS

Para la entrega del segundo producto mínimo viable al conjunto residencial, se trabajó en la autenticación de los residentes y del administrador, módulo de noticias o publicaciones que realiza el administrador, módulo de realización de solicitudes al administrador ya sean quejas, peticiones o sugerencias y módulo de novedades de residentes, en donde el administrador puede ver las solicitudes realizadas.

4.2.1.1. Historias de usuario: para este segundo incremento se elaboraron las siguientes historias de usuario:

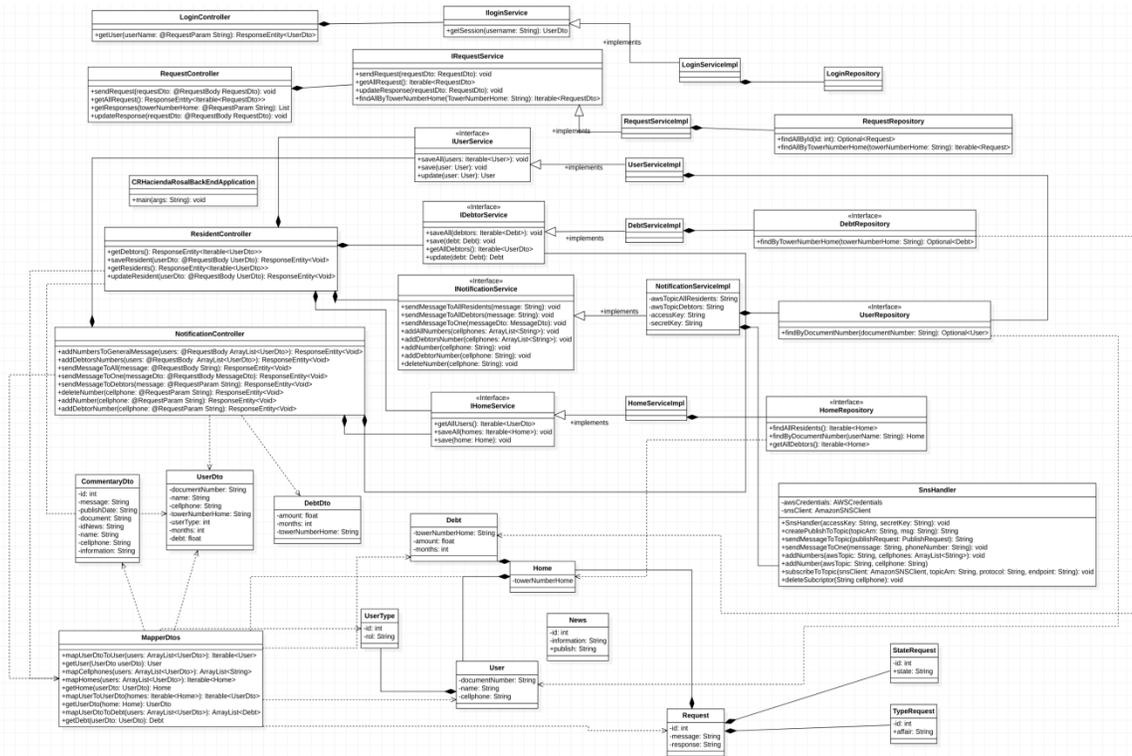
- Relacionadas con la autenticación de usuarios.
 - Autenticar diferentes roles en la aplicación web.
- Relacionadas con el envío de solicitudes al administrador.
 - Enviar solicitud al administrador.
 - Recibir solicitudes de los residentes.
- Relacionadas con las publicaciones que realiza el administrador.
 - Ver publicaciones que realiza el administrador.
 - Comentar publicaciones que realiza el administrador.
 - Ver comentarios que realizan los residentes sobre las publicaciones.

Las historias de usuario de este segundo incremento se encuentran en el Anexo A.

4.2.2. DISEÑO

4.2.2.1. Diagrama de clases: en la Figura 13, se observa el diagrama de clases, donde se anexan las clases implementadas para este incremento.

Figura 13. Diagrama de clases

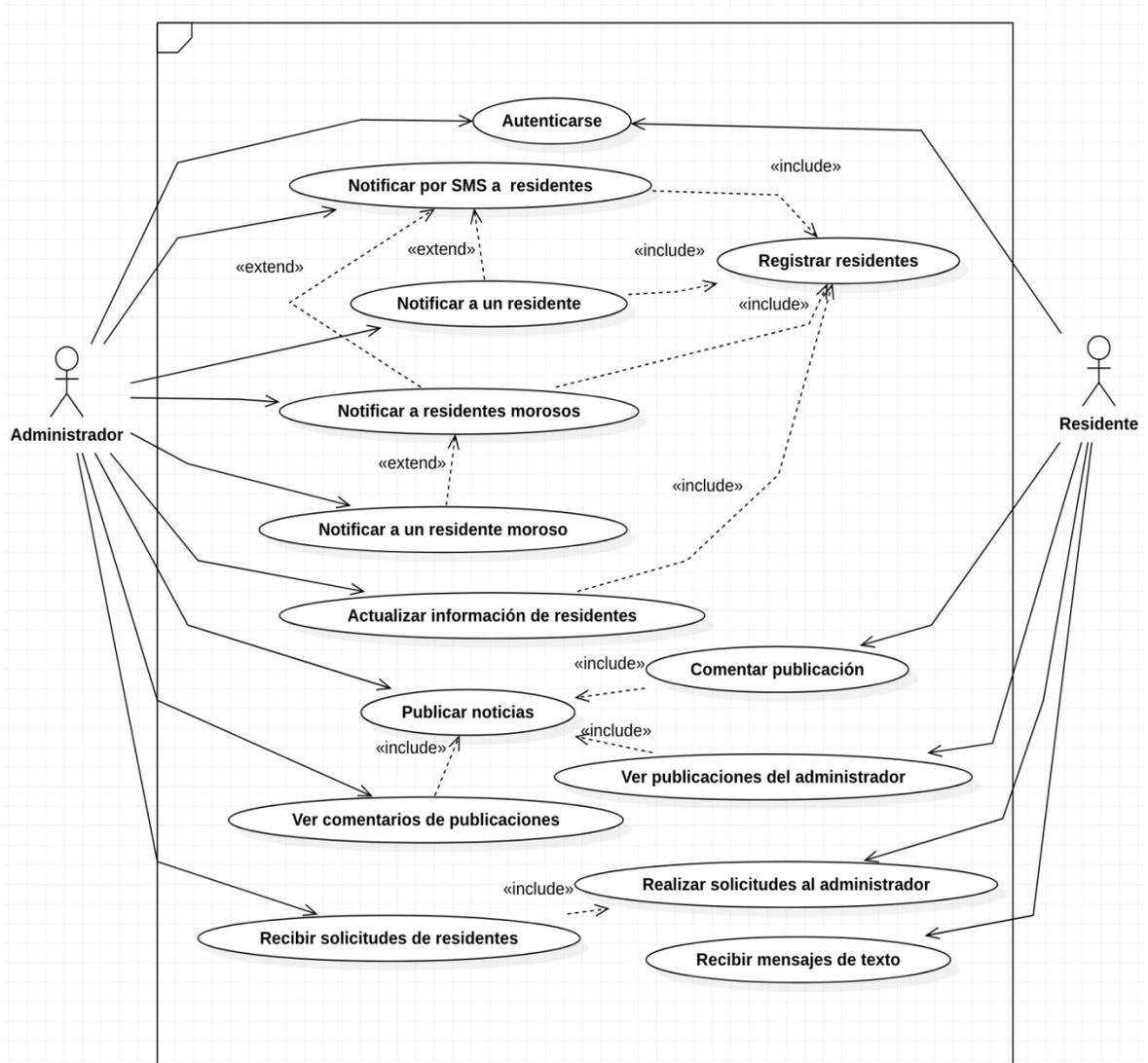


Fuente: elaboración propia

4.2.2.2. Diagramas de secuencia: modela los procesos y comportamientos más importantes de la aplicación web, implementado en este incremento. Los diagramas de secuencia de este incremento se encuentran en el Anexo B.

4.2.2.3. Diagrama de historias de usuario: en la Figura 14, se observan las funcionalidades de la aplicación hasta este segundo incremento.

Figura 14. Diagrama de historias de uso

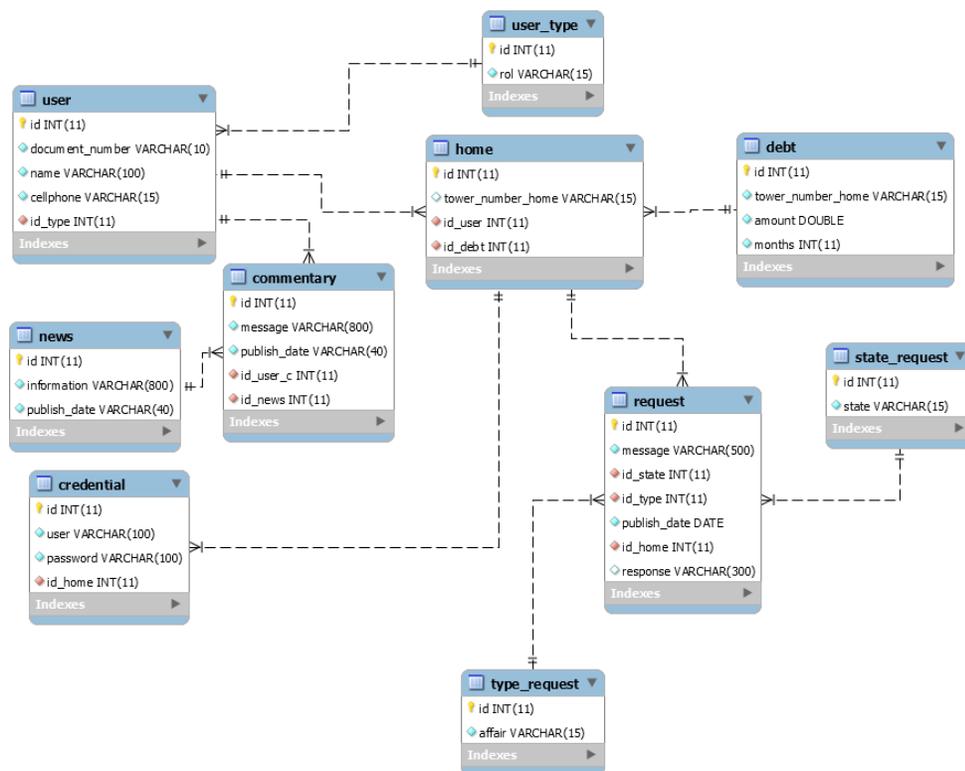


Fuente: elaboración propia

4.2.2.4. Diagrama de actividades: los diagramas de actividades de este incremento se pueden consultar en el Anexo B.

4.2.2.5. Diseño de la base de datos: en la Figura 15, se observa el diseño final del modelo entidad relación de la base de datos de la aplicación web, en donde se muestran las diferentes entidades y relaciones entre ellas.

Figura 15. Modelo Entidad Relación de la aplicación web – segundo incremento



Fuente: elaboración propia

4.2.3. DESARROLLO

La versión del desarrollo de este incremento se encuentra en GitHub:

Aplicación frontend: <https://github.com/Joanne86/c-r-hacienda-rosal-frontend/commit/f8686d600095b4fa691377d13e8534e3f351c53e>

Aplicación backend: <https://github.com/Joanne86/c-r-hacienda-rosal-backend/commit/9d8d3f6c0189a18b4e454cb8873da6f9235bc509>

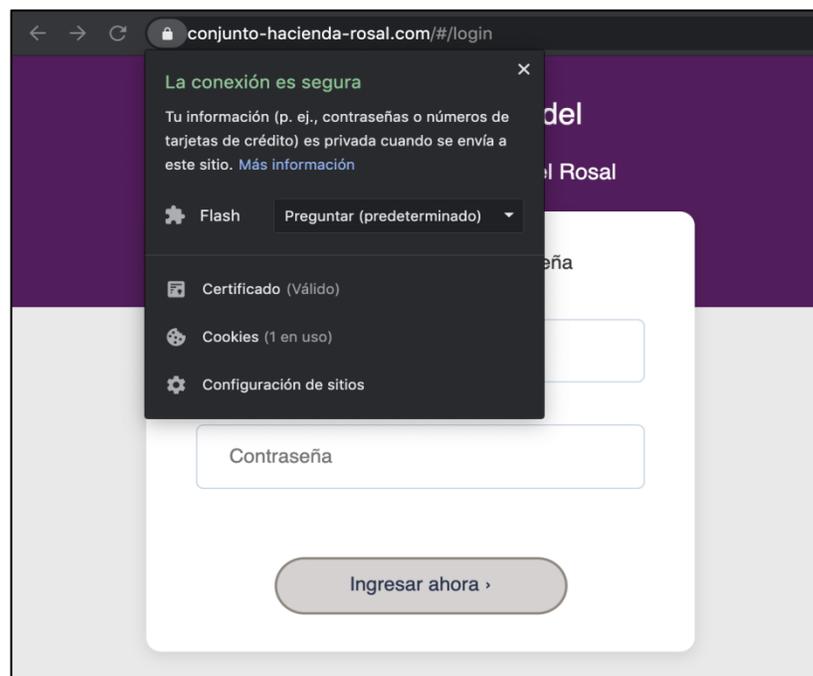
4.2.4. PRUEBAS

Para la validación de la correcta funcionalidad de este segundo incremento se realizaron y validaron los casos de prueba para cada una de las historias de usuario desarrolladas. Estos se consignaron en el Anexo C.

4.2.5. DESPLIEGUE

Para el despliegue de la aplicación frontend fue importante adquirir un certificado de seguridad SSL (Secure Sockets Layer) del sitio en Amazon Web Services que permite realizar comunicaciones, comercio electrónico e interacciones de forma segura en sitios Web. Esto se realizó gracias al servicio route 53; en este servicio se compra el dominio de la aplicación con un certificado de seguridad. En la Figura 16, se puede ver que la aplicación quedó como una conexión segura.

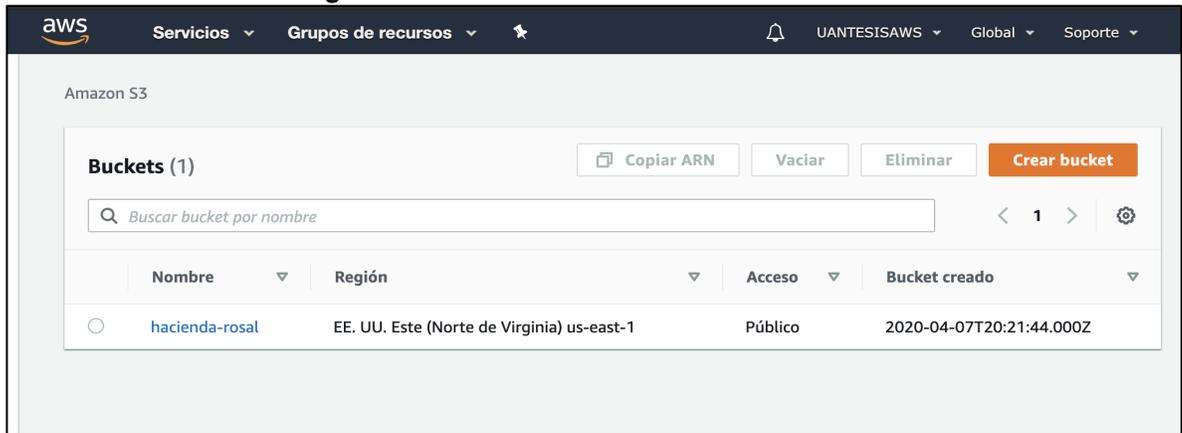
Figura 16. Certificado de seguridad SSL



Fuente: elaboración propia

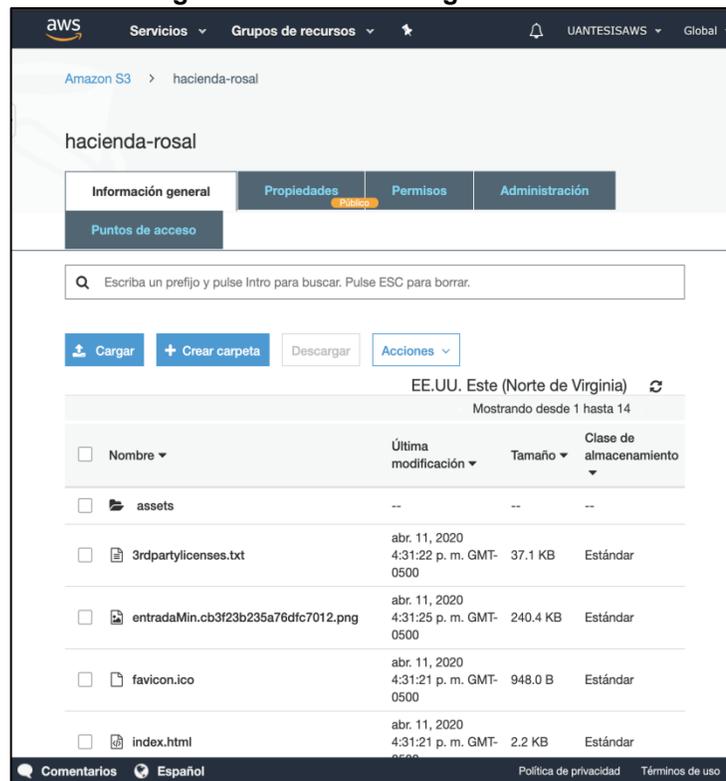
Luego se creó un espacio de almacenamiento en el servicio S3 de Amazon Web Services, en donde se subieron todos los archivos compilados para producción del proyecto frontend, ver Figura 17 y Figura 18.

Figura 17. Creación de almacenamiento en S3



Fuente: elaboración propia

Figura 18. Archivos cargados en S3



Fuente: elaboración propia

Posteriormente, se creó una instancia en CloudFront, como se observa en la Figura 19. Esta instancia accede directamente de forma segura a los archivos cargados en

el S3. El nombre del dominio de la aplicación quedó de la siguiente manera:
https://conjunto-hacienda-rosal.com/

Figura 19. Instancia de aplicación en CloudFront

The screenshot shows the AWS CloudFront console interface. At the top, there are navigation tabs for 'Servicios', 'Grupos de recursos', and a search bar. Below this, there is a notification banner about best practices for Lambda@Edge. The main heading is 'CloudFront Distributions'. There are buttons for 'Create Distribution', 'Distribution Settings', 'Delete', 'Enable', and 'Disable'. Below the buttons, there are filters for 'Viewing: Any Delivery Method' and 'Any State'. A table lists the distribution details:

Delivery Method	ID	Domain Name	C	Orig	CNAMEs	Status	State	Last Modified
Web	EHYCSHVX8S3PU	dwv15i7najiyy.cloudfront.	-	haci	conjunto-hacienda-rosal.com	Deployed	Enabled	2020-04-08 13:47 UTC-5

Fuente: elaboración propia

4.3. INCREMENTO 3

En el tercer incremento, se implementó el módulo de ver solicitudes realizadas por los residentes al administrador. En este el residente puede consultar las respuestas que ha dado el administrador, a cada una de las solicitudes enviadas. El módulo de estado de pagos, en donde los residentes pueden estar informados sobre sus pagos, indicando si está en estado de mora o al día. Por último, la funcionalidad en donde el administrador puede dar respuesta a las peticiones que realizan los residentes. Para ello se ejecutaron las siguientes etapas:

4.3.1. ANÁLISIS DE REQUERIMIENTOS

Para la entrega de este último producto mínimo viable al conjunto residencial, se trabajaron los siguientes módulos: para el residente el módulo de ver sus solicitudes y el módulo de estado de pagos. Para el administrador la funcionalidad de responder a las solicitudes de los residentes en el módulo de novedades de residentes.

4.3.1.1. Historias de usuario: para este tercer incremento se elaboraron las siguientes historias de usuario:

- Relacionadas con ver las solicitudes enviadas al administrador.
 - Ver solicitudes enviadas al administrador.
 - Ver respuesta de las solicitudes enviadas al administrador.
- Relacionadas con el módulo de estado de pagos.
 - Ver estado de pagos.
- Relacionadas con dar respuesta a las solicitudes que envían los residentes.
 - Responder a las solicitudes que envían los residentes.
 - Filtrar por estado (queja, sugerencia o petición) las solicitudes de los residentes.

Las historias de usuario de este incremento se encuentran en el Anexo A.

4.3.2. DISEÑO

4.3.2.1. Diagrama de clases: en la Figura 20, se observa parte del diagrama de clases de toda la aplicación en donde se encuentran las clases implementadas para este incremento. El diagrama completo se encuentra en el Anexo B.

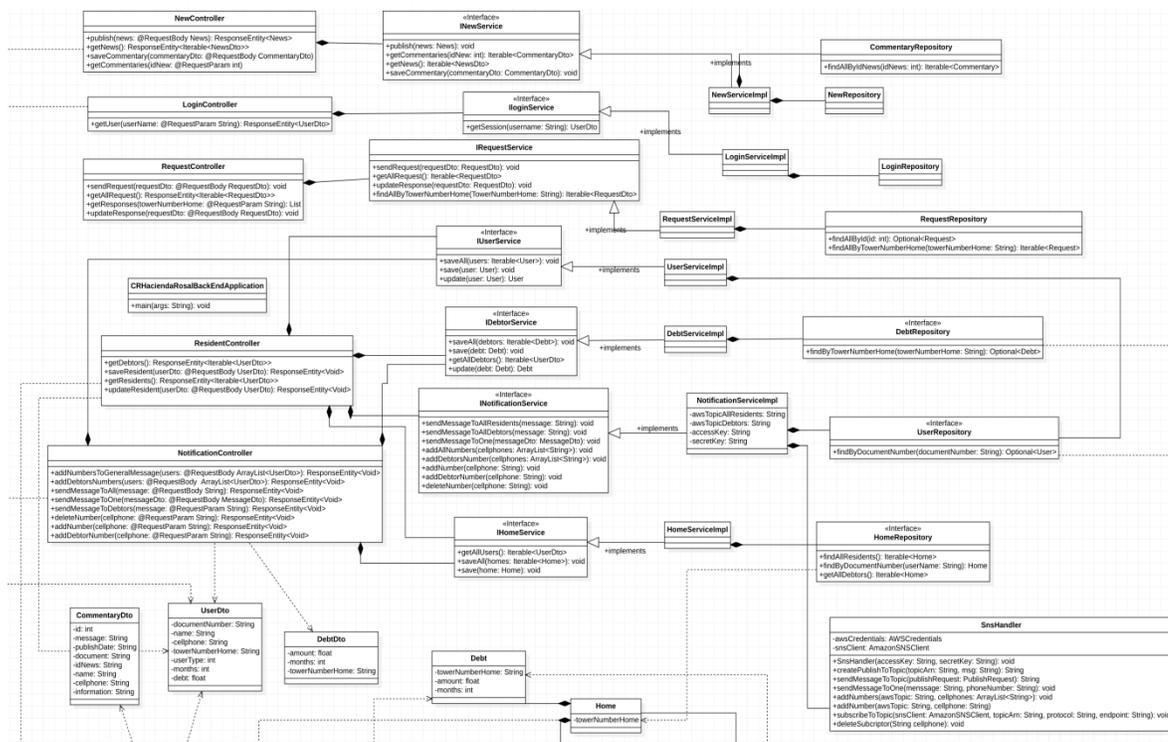
Para la construcción de todo este diseño, se utilizó como base el patrón MVC (Modelo, Vista, Controlador). En los controladores se reciben las peticiones HTTP por Rest que efectúa el front. Dentro de ellas, se llama a los servicios que realizan la lógica con los datos que llegan del controlador.

Posteriormente, los datos son enviados a los repositorios, que es donde se encuentran alojadas todas las consultas HQL (Hibernate Query Language) o Lenguaje de Consulta de Hibernate. Estas consultas mapean las clases de java a entidades que puedan ser entendidas por MySQL, para ello, se utilizó JPA (Java Persistence API) o framework del lenguaje

de programación Java, que maneja datos relacionales en aplicaciones usando la Plataforma Java.

Para devolver los datos al frontend, se utilizaron DTOs (Data Transfer Object) u Objeto de Transferencia de Datos. Para hacer mapeo de varios datos de varias entidades en una sola clase de Java plana, en donde resulta más fácil manejar DTOs, que las propias entidades en java.

Figura 20. Diagrama de clases

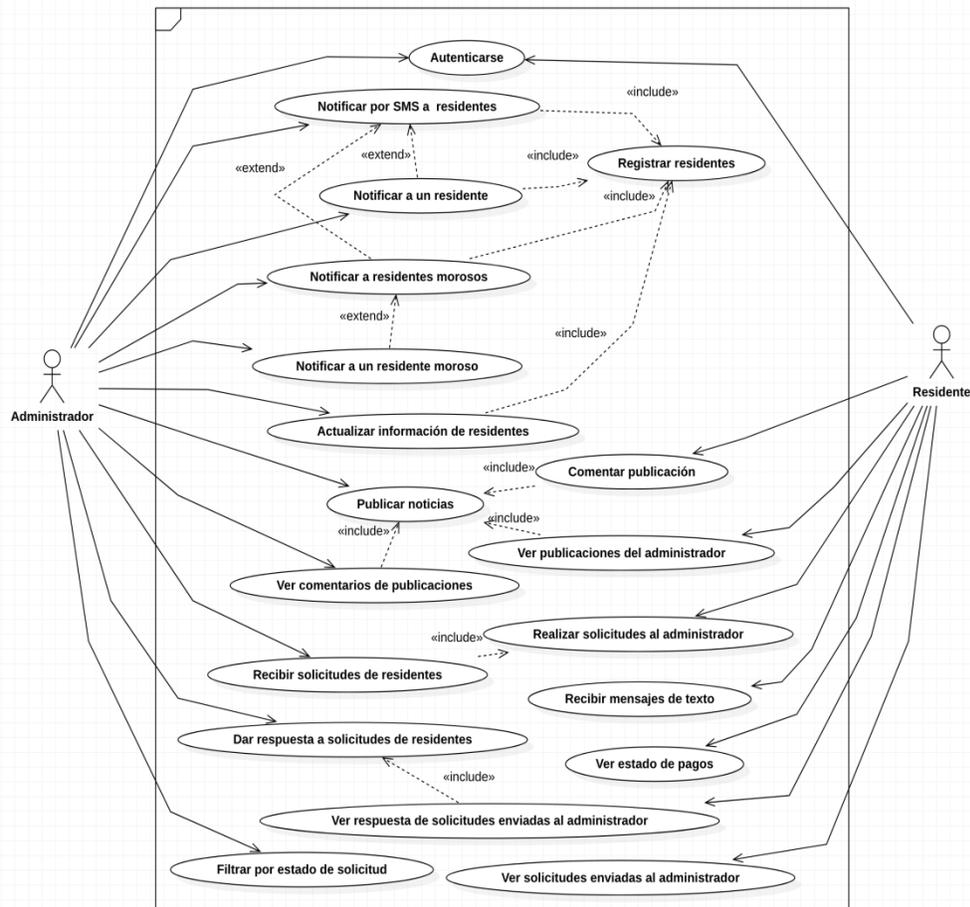


Fuente: elaboración propia

4.3.2.2. Diagramas de secuencia: modela los procesos y comportamientos más importantes de la aplicación en el incremento 3. Los diagramas de secuencia elaborados en este incremento se encuentran en el Anexo B.

4.3.2.3. Diagramas de historias de usuario: la Figura 21, contienen las funcionalidades elaboradas para la aplicación.

Figura 21. Diagrama de historias de uso



Fuente: elaboración propia

4.3.2.4. Diagrama de actividades: los diagramas de actividades se pueden consultar en el Anexo B.

4.3.2.5. Diseño de la base de datos: en la Figura 15, se observa el diseño final del modelo entidad relación de la base de datos de la aplicación web.

4.3.3. DESARROLLO

La versión del desarrollo de este último incremento de encuentra en GitHub:

Aplicación frontend: <https://github.com/Joanne86/c-r-hacienda-rosal-front-end>

Aplicación backend: <https://github.com/Joanne86/c-r-hacienda-rosal-back-end>

4.3.4. PRUEBAS

Para la validación de la correcta funcionalidad de este tercer y último incremento se realizaron los casos de prueba para cada una de las historias de usuario desarrolladas. Los casos de prueba se consignaron en el Anexo C.

4.3.5. DESPLIEGUE

Para este incremento se creó la base de datos en AWS con el Servicio de base de Datos Relacional RDS, como se muestra en la Figura 22. Con este servicio se puede tener la base de datos corriendo en la nube 24 horas, 7 días a la semana.

Figura 22. Base de datos creada en el servicio RDS

The screenshot shows the AWS RDS console interface for a database instance named 'rds-hacienda-rosal'. The breadcrumb navigation is 'RDS > Databases > rds-hacienda-rosal'. The instance name 'rds-hacienda-rosal' is displayed at the top, along with 'Modify' and 'Actions' buttons. Below this is a 'Summary' section with a table of key metrics:

DB identifier	CPU	Info	Class
rds-hacienda-rosal	1.64%	Available	db.t2.micro
Role	Current activity	Engine	Region & AZ
Instance	13 Connections	MySQL Community	us-east-1f

Below the summary are tabs for 'Connectivity & security' (selected), 'Monitoring', 'Logs & events', 'Configuration', 'Maintenance & backups', and 'Tags'. The 'Connectivity & security' section is expanded, showing a table with three columns: 'Endpoint & port', 'Networking', and 'Security'.

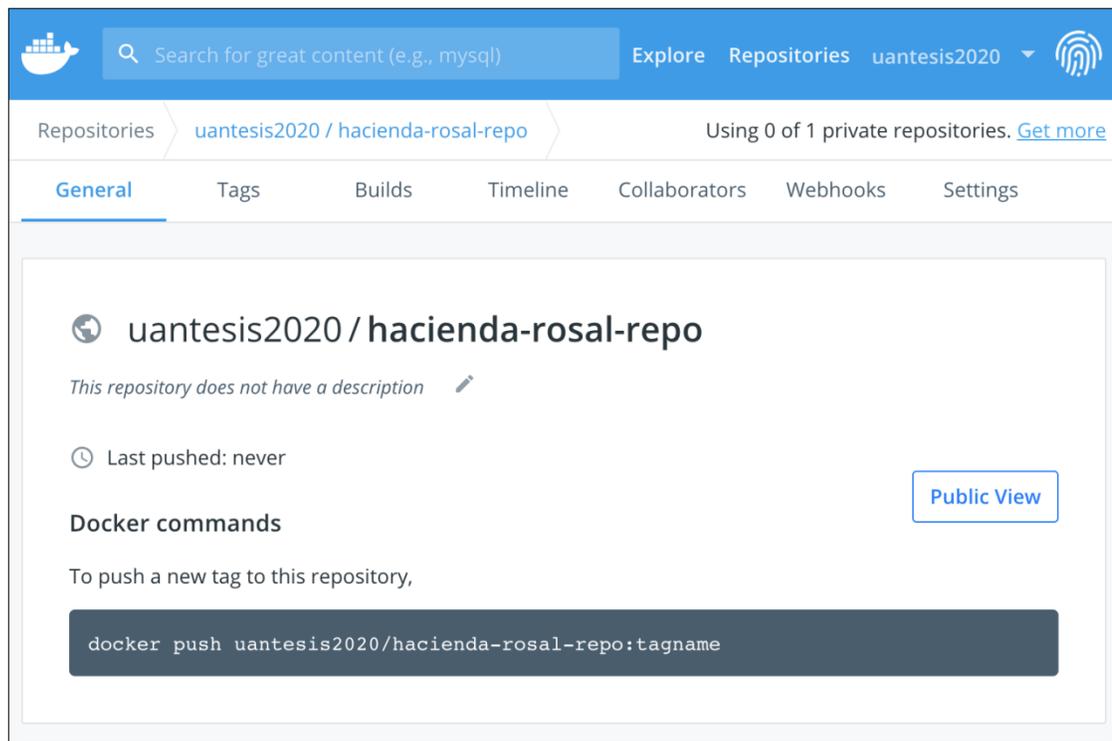
Endpoint & port	Networking	Security
Endpoint rds-hacienda-rosal.chg16m6djwnx.us-east-1.rds.amazonaws.com	Availability zone us-east-1f	VPC security groups hacienda-rosal-sg-db (sg-09119185bcfbe112a) (active)
Port 3306	VPC vpc-0a92dc50adc99008c	Public accessibility Yes
	Subnet group default-vpc-0a92dc50adc99008c	Certificate authority

Fuente: elaboración propia

En este incremento se desplegó la aplicación backend en la nube. Para esto se utilizó el Servicio de Contenedores Elásticos ECS, creando una instancia en una máquina virtual que permite tener una tarea corriendo las 24 horas y los 7 días a la semana la aplicación.

Para realizar esto, se creó una imagen de docker de la aplicación java, que se alojó en DockerHub, como se observa en la Figura 23.

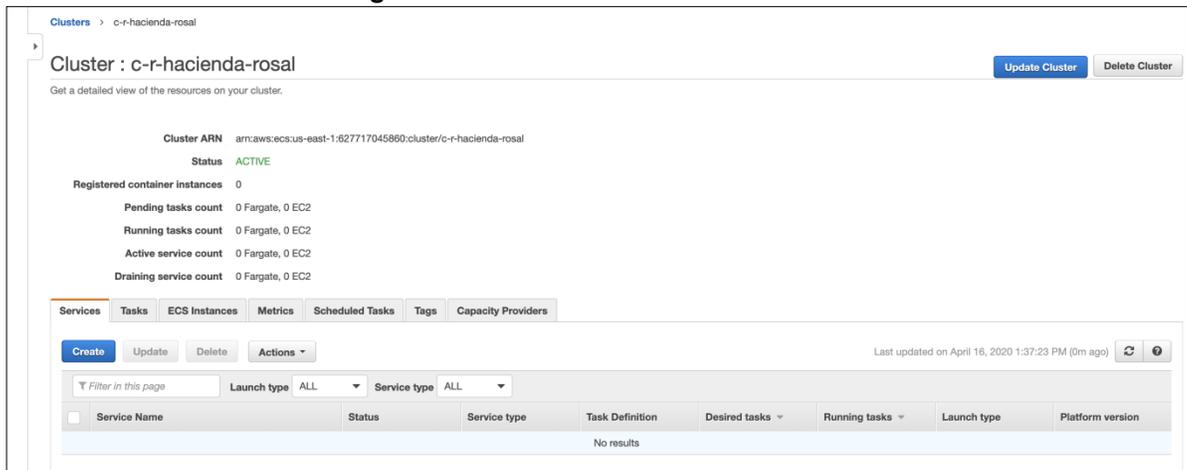
Figura 23. Imagen docker creada en DockerHub



Fuente: elaboración propia

Luego de tener lista la imagen de docker, se creó un clúster en ECS para permitir la ejecución de la tarea en una máquina virtual, como se observa en la Figura 24.

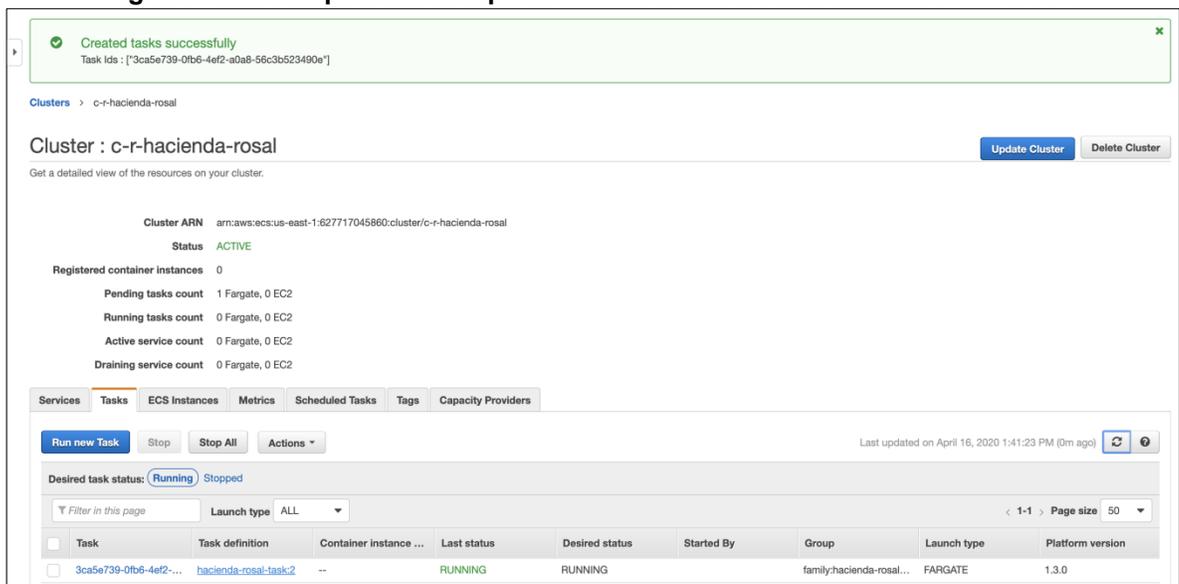
Figura 24. Creación del clúster en ECS



Fuente: elaboración propia

Luego se creó la tarea que llama el contenedor de docker, donde está la aplicación y la ejecuta, como se puede observar en la Figura 25.

Figura 25. Tarea que corre la aplicación backend creada en el cluster de ECS



Fuente: elaboración propia

Para verificar que la tarea se ejecuta correctamente en la aplicación backend, se visualizaron los registros que deja la aplicación spring boot cuando corre por el puerto 8080, como se muestra en la Figura 26.

Figura 26. Registro de la aplicación

The screenshot shows a web interface for viewing application logs. At the top, it displays the task ID: `Task : 3ca5e739-0fb6-4ef2-a0a8-56c3b523490e`. Below this, there are tabs for 'Details', 'Tags', and 'Logs', with 'Logs' being the active tab. The logs are displayed in a table with columns for 'Timestamp (UTC+00:00)' and 'Message'. The messages include information about the application starting, Tomcat starting on port 8080, JPA configuration, and Hibernate database operations.

Timestamp (UTC+00:00)	Message
2020-04-16 13:41:09	2020-04-16 18:41:09.765 INFO 1 --- [main] c.h.r.CRHaciendaRosalBackEndApplication : Started CRHaciendaRosalBackEndAppli...
2020-04-16 13:41:09	2020-04-16 18:41:09.762 INFO 1 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 8080 (http) w...
2020-04-16 13:41:07	2020-04-16 18:41:07.177 INFO 1 --- [main] o.s.s.concurrent.ThreadPoolTaskExecutor : Initializing ExecutorService 'applicationTas...
2020-04-16 13:41:06	2020-04-16 18:41:06.565 WARN 1 --- [main] JpaBaseConfiguration\$JpaWebConfiguration : spring.jpa.open-in-view is enabled by...
2020-04-16 13:41:02	2020-04-16 18:41:02.882 INFO 1 --- [main] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for ...
2020-04-16 13:41:02	2020-04-16 18:41:02.870 INFO 1 --- [main] o.h.e.t.j.p.i.JtaPlatformInitiator : HHH000490: Using JtaPlatform implementation: [org....
2020-04-16 13:41:02	Hibernate: create table resident_credentials (id integer not null auto_increment, name varchar(255), number_cellphone varchar(255...
2020-04-16 13:40:58	2020-04-16 18:40:58.891 INFO 1 --- [main] org.hibernate.dialect.Dialect : HHH000400: Using dialect: org.hibernate.dialect.MySQ...
2020-04-16 13:40:58	2020-04-16 18:40:58.463 INFO 1 --- [main] o.hibernate.annotations.common.Version : HCANN000001: Hibernate Commons Ann...
2020-04-16 13:40:57	2020-04-16 18:40:57.767 INFO 1 --- [main] org.hibernate.Version : HHH000412: Hibernate Core (5.4.8.Final)

Fuente: elaboración propia

Actualmente, la aplicación backend está configurada en una dirección IP, como se muestra en la Figura 27. Es decir, que se puede acceder a los servicios por el puerto 8080.

Figura 27. Dirección IP, donde está expuesta la aplicación backend

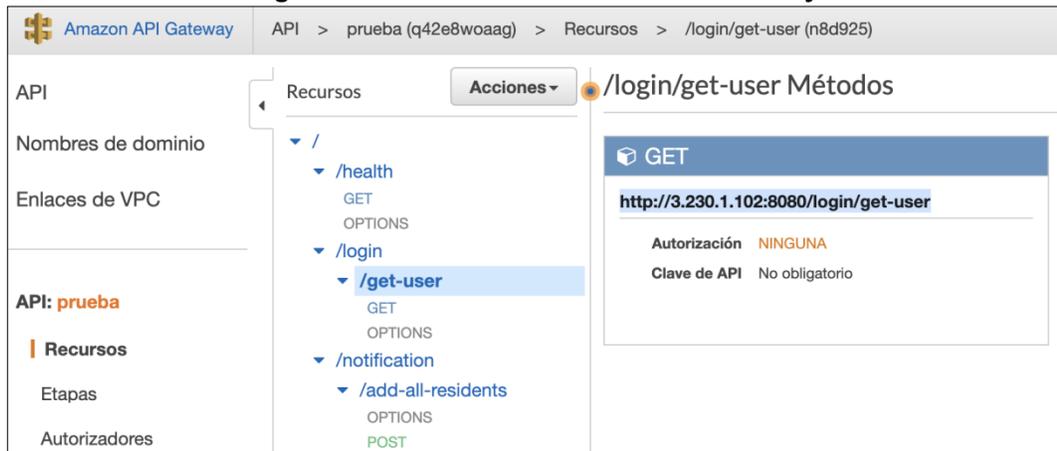
The screenshot shows the AWS console interface. The 'Network' section displays the following configuration: Network mode: `awsvpc`, ENI Id: `eni-099816157f09ba809`, Subnet Id: `subnet-0734285b21ccd5c1c`, Private IP: `10.88.1.5`, Public IP: `3.230.1.102`, and Mac address: `02:da:bc:33:b4:db`. Below this, the 'Containers' section shows a table with columns for Name, Container Runtime I..., Status, Image, and Image Digest. One container is listed with the name `h-c-hacien...`, runtime ID `095b21d3e9eeb3870...`, status `RUNNING`, and image `docker.io/uantesis2020/hacienda-r...`.

Name	Container Runtime I...	Status	Image	Image Digest
h-c-hacien...	095b21d3e9eeb3870...	RUNNING	docker.io/uantesis2020/hacienda-r...	

Fuente: elaboración propia

El acceder por medio de esta dirección IP, resultó ser inseguro para la aplicación. Por lo tanto, se usó el servicio API Gateway de AWS para crear una capa segura, que acceda a la IP que tiene expuesto el backend. Este servicio cuenta con todos los recursos HTTP de la aplicación backend, cada uno con sus métodos Rest (PUT, POST, DELETE y GET), como se observa en la Figura 28.

Figura 28. Recursos creados en API Gateway



Fuente: elaboración propia

Con esta API, no se accede a la dirección IP directamente si no al recurso que expone: <https://q42e8woaag.execute-api.us-east-1.amazonaws.com/hacienda-rosal>. Al ser pública y segura, la aplicación frontend la puede consumir debido a que está expuesto en un dominio con certificación de seguridad.

5. RESULTADOS

En este capítulo se muestran los módulos del aplicativo web creado de acuerdo con los requerimientos del conjunto residencial Hacienda el Rosal.

5.1. INGRESO DE SESIÓN

Como se observa en la Figura 29, se cuenta con una interfaz para ingresar a la aplicación. Cuando un usuario se loguea, el aplicativo valida que las credenciales se encuentren creadas y el perfil asignado. Posteriormente, puede acceder a los módulos asignados al perfil.

Figura 29. Ingreso de sesión a la aplicación

The image shows a login screen with a dark purple header. The text reads: "Bienvenido al portal del Conjunto residencial Hacienda el Rosal". Below this, there is a white login box with a purple user icon and the text "Ingresa tu usuario y contraseña". The username field contains "123456789" and the password field contains ".....". At the bottom of the box is a yellow button labeled "Ingresar ahora >".

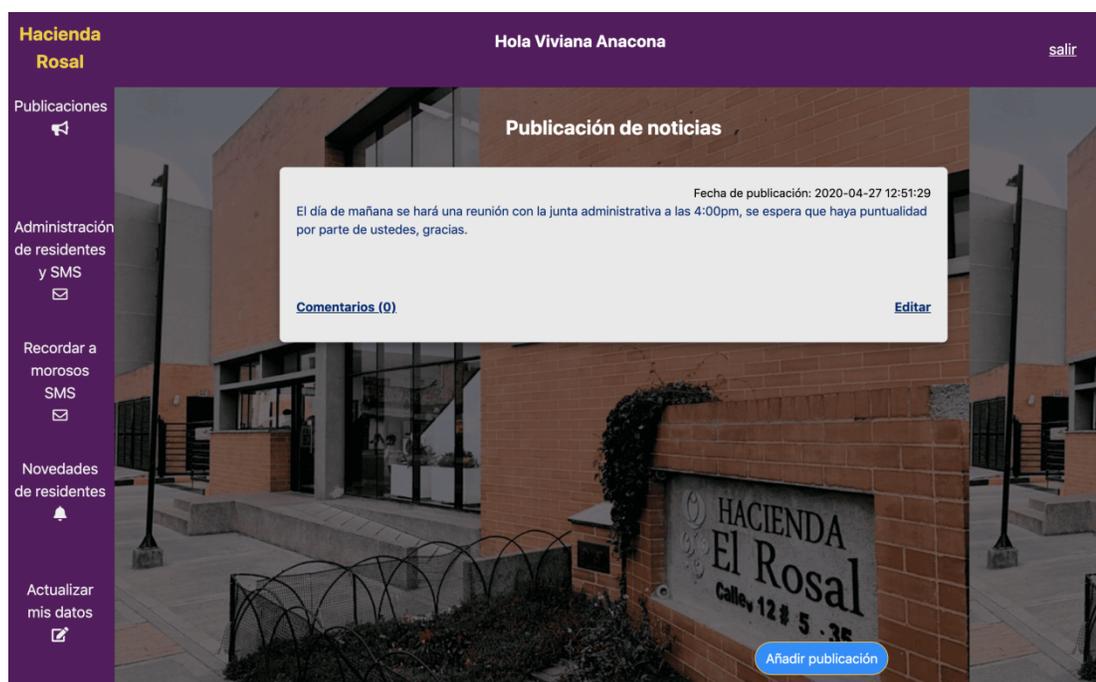
Fuente: elaboración propia

5.2. MÓDULOS PARA EL ADMINISTRADOR

El usuario administrador tiene acceso a los siguientes módulos:

5.2.1. Publicaciones: como se muestra Figura 30, con esta funcionalidad el administrador puede publicar noticias y ver los comentarios realizados por los residentes sobre las publicaciones.

Figura 30. Publicación de noticias



Fuente: elaboración propia

5.2.2. Administración de la información de los residentes y SMS: en esta sección el administrador realiza la creación, edición y consulta de los residentes del conjunto. La creación la puede hacer mediante la carga de información por medio de un archivo de excel o de manera individual, según sea requerido. También puede enviar notificaciones de forma masiva a todos los residentes del conjunto o de forma personalizada, como se puede observar en la Figura 31.

Figura 31. Administración de la información de los residentes

Información de los residentes del conjunto residencial

En esta sección usted puede registrar todos los residentes del conjunto adjuntando un archivo excel y también notificar por mensaje de texto un aviso importante para todos los residentes o para uno en específico.

Los datos que se encuentran en una caja de texto son editables.

Torre-apartamento

Torre Apartamento	Documento	Nombre	Celular	Meses en deuda	Deuda total	Enviar mensaje a todos
01-103	345678912	Leidy Yohana Rodriguez	3227525187	1	\$ 70000	<input type="button" value="Notificar"/>
02-101	543234567	Katherine Aguirre Tique	3227623418	2	\$ 140000	<input type="button" value="Notificar"/>
03-404	432345432	Maria Nury Escobar	3125256290	0	\$ 0	<input type="button" value="Notificar"/>

Total de residentes registrados (3)

Fuente: elaboración propia

5.2.3. Recordar a morosos SMS: como se observa en la Figura 32, en esta sección el administrador puede tener una lista de los residentes morosos por concepto de pago de administración y puede notificarlos por mensajes de texto.

Figura 32. Recordar a morosos por SMS

Deudores morosos de pago de administración

En esta sección usted podrá notificar por mensaje de texto un aviso importante para todos los deudores de administración o para uno en específico.

Torre-apartamento

Torre Apartamento	Nombre	Cédula	Meses en mora	Deuda total	Enviar mensaje a todos
01-103	Leidy Yohana Rodriguez	345678912	3	\$140,000.00	<input type="button" value="Notificar"/>
02-101	Katherine Aguirre Tique	543234567	2	\$140,000.00	<input type="button" value="Notificar"/>
01-105	Daniel Castiblanco	5412454323	1	\$70,000.00	<input type="button" value="Notificar"/>

Total de deudores (3)

Fuente: elaboración propia

5.2.4. Novedades de residentes: con esta funcionalidad el administrador puede ver las diferentes solicitudes de los residentes (queja, sugerencia o petición) para darles respuesta, como se observa en la Figura 33.

Figura 33. Solicitudes de los residentes

Solicitudes de los residentes del conjunto
En esta sección usted puede ver y responder las solicitudes que realizan los residentes del conjunto residencial Hacienda el Rosal.

Filtrar por tipo de solicitud: Todos | Filtrar por estado de solicitud: Todos

Torre	Apartamento	Fecha	Nombre	Petición	Tipo	Estado
	1-102	2020-04-18	David Torres Castellanos	Los vecinos hacen demasiado escandalo todas las noches, necesito que por favor	Petición	Atendido
	1-102	2020-04-18	David Torres Castellanos	Necesito por favor que me habiliten el parqueadero porque mañana me llegan	Sugerencia	Sin respuesta

Total solicitudes (2) - Con respuesta (1) - Sin respuesta (1)

Fuente: elaboración propia

5.2.5. Actualización de datos: con esta funcionalidad se pueden actualizar los datos del administrador, ver Figura 34.

Figura 34. Actualización de datos

Actualizar mis datos

Nombre completo: Viviana Anacona | Contraseña actual: *****

Nombre de usuario: admin | Contraseña nueva: *****

Cédula: 123456789 | Confirmación de contraseña nueva: *****

Actualizar >

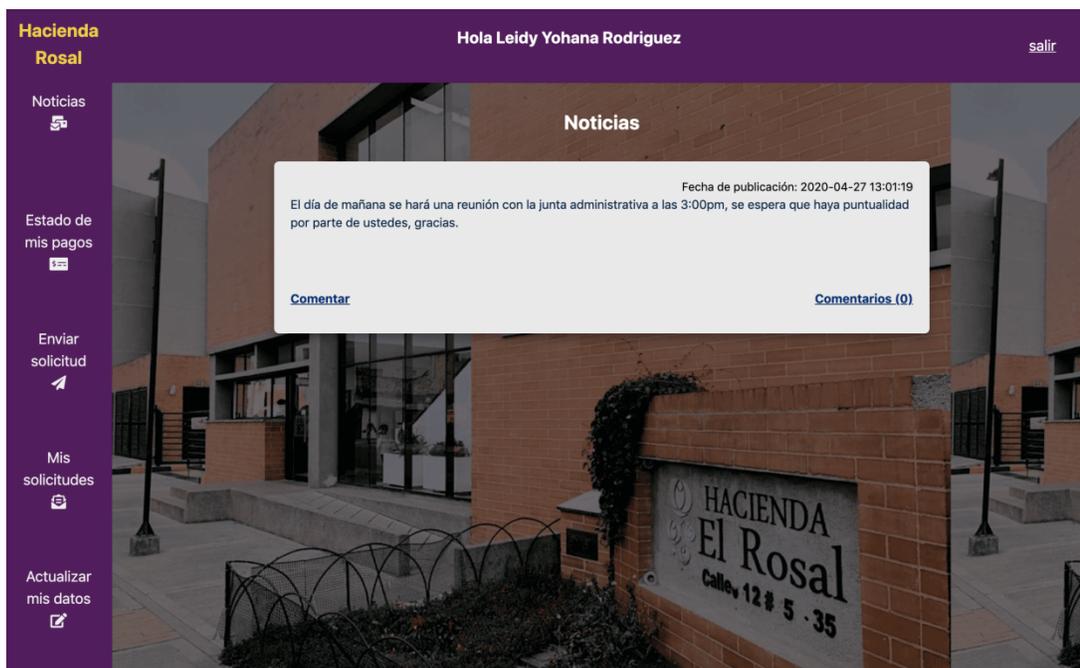
Fuente: elaboración propia

5.3. MÓDULOS PARA EL RESIDENTE

El usuario residente tiene acceso a los siguientes módulos:

5.3.1. Noticias: en esta sección los residentes pueden ver todas las publicaciones que realiza el administrador, las pueden comentar y pueden ver los comentarios realizados, como se muestra en la Figura 35.

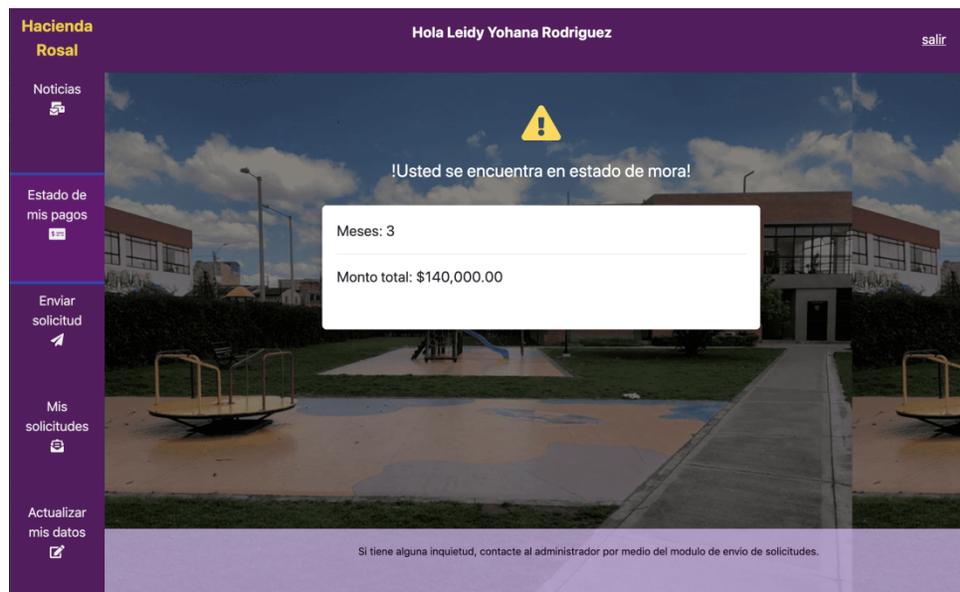
Figura 35. Publicaciones de noticias que realiza el administrador



Fuente: elaboración propia

5.3.2. Estado de mis pagos: en esta sección los residentes pueden ver su estado de pago, si está al día o en mora, como se observa en la Figura 36.

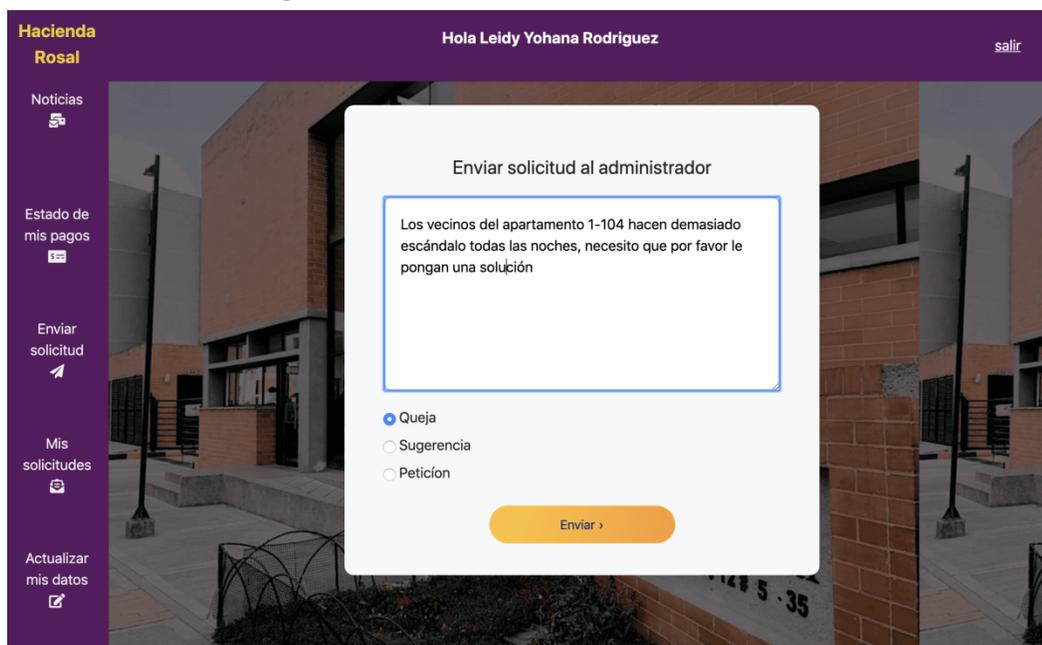
Figura 36. Estado de pagos



Fuente: elaboración propia

5.3.3. Enviar solicitud: en esta sección los residentes pueden enviar diferentes tipos de solicitud (queja, sugerencia o petición) al administrador, de acuerdo con sus necesidades, como se observa en la Figura 37.

Figura 37. Envío de solicitudes al administrador



Fuente: elaboración propia

5.3.4. Mis solicitudes: en esta sección los residentes pueden ver todas las solicitudes que han realizado al administrador, con el estado en que se encuentran y la respuesta, ver Figura 38.

Figura 38. Solicitudes que envían los residentes

Hacienda Rosal
Hola Leidy Yohana Rodriguez salir

Mis solicitudes

En esta sección usted podrá ver las solicitudes que ha enviado al administrador del conjunto y ver las respuestas del administrador.

Filtrar por tipo de solicitud: Todos
Filtrar por estado de solicitud: Todos

Fecha	Tipo	Petición	Estado	Ver
2020-04-27	Queja	Los vecinos del apartamento 1-104 hacen demasiado escándalo todas las noches,	Sin respuesta	--
2020-04-27	Sugerencia	Buenos días para sugerir que les recuerden a los residentes que levanten los residuos de	Atendido	

Total solicitudes (2) - Con respuesta (1) - Sin respuesta (1)

Fuente: elaboración propia

5.3.4. Actualización de datos: en esta sección el residente podrá actualizar sus datos personales, Figura 39.

Figura 39. Actualización de datos del residente

Hacienda Rosal
Hola Leidy Yohana Rodriguez salir

Actualizar mis datos

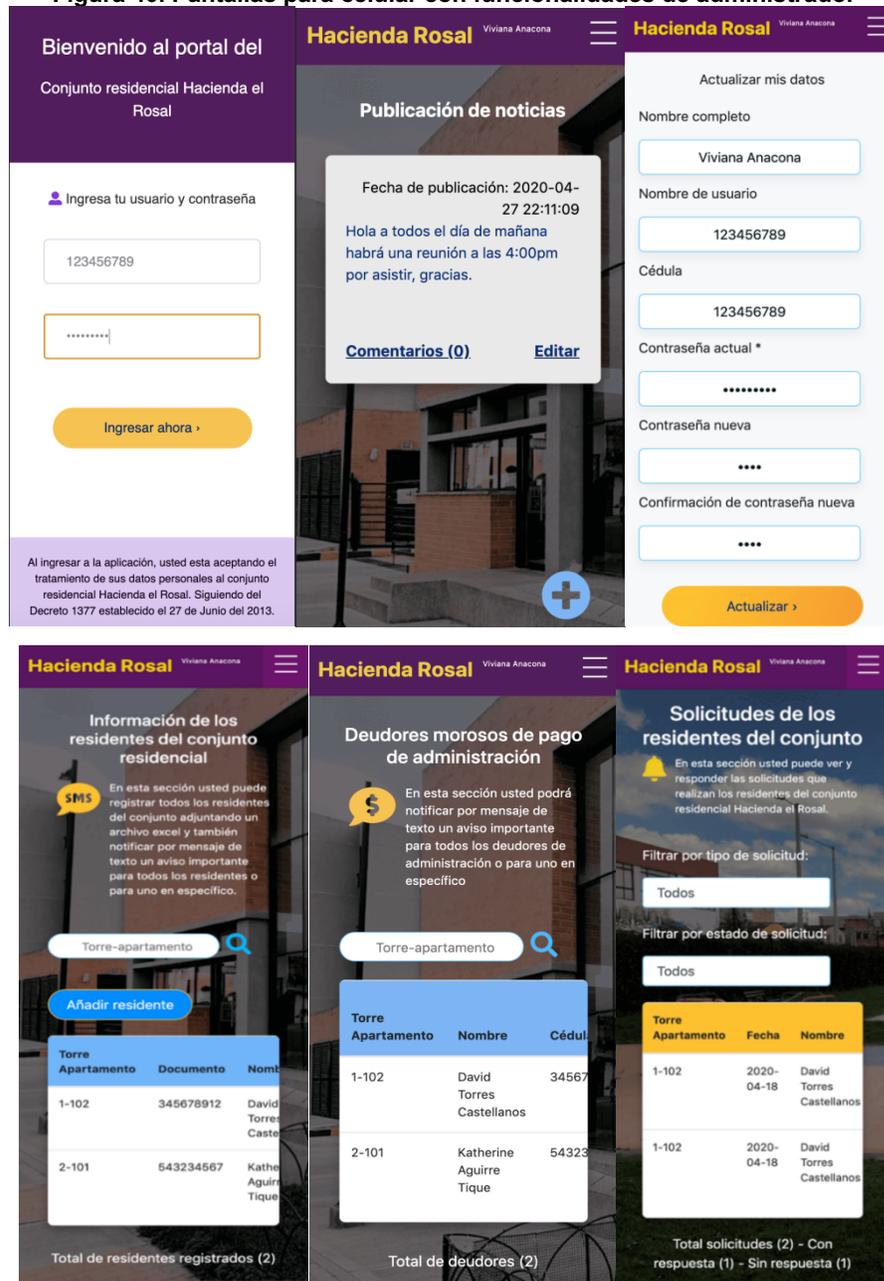
Nombre completo: Leidy Yohana Rodriguez
Contraseña actual *: *****
Nombre de usuario: 345678912
Contraseña nueva: *****
Cédula: 345678912
Confirmación de contraseña nueva: *****

Actualizar >

Fuente: elaboración propia

Adicionalmente, uno de los requerimientos de esta aplicación web, es que fuera responsiva, es decir, que se pueda usar para diferentes tamaños de pantalla. En la Figura 40, se puede observar el login y las funcionalidades anteriormente descritas para el administrador vistas en celular.

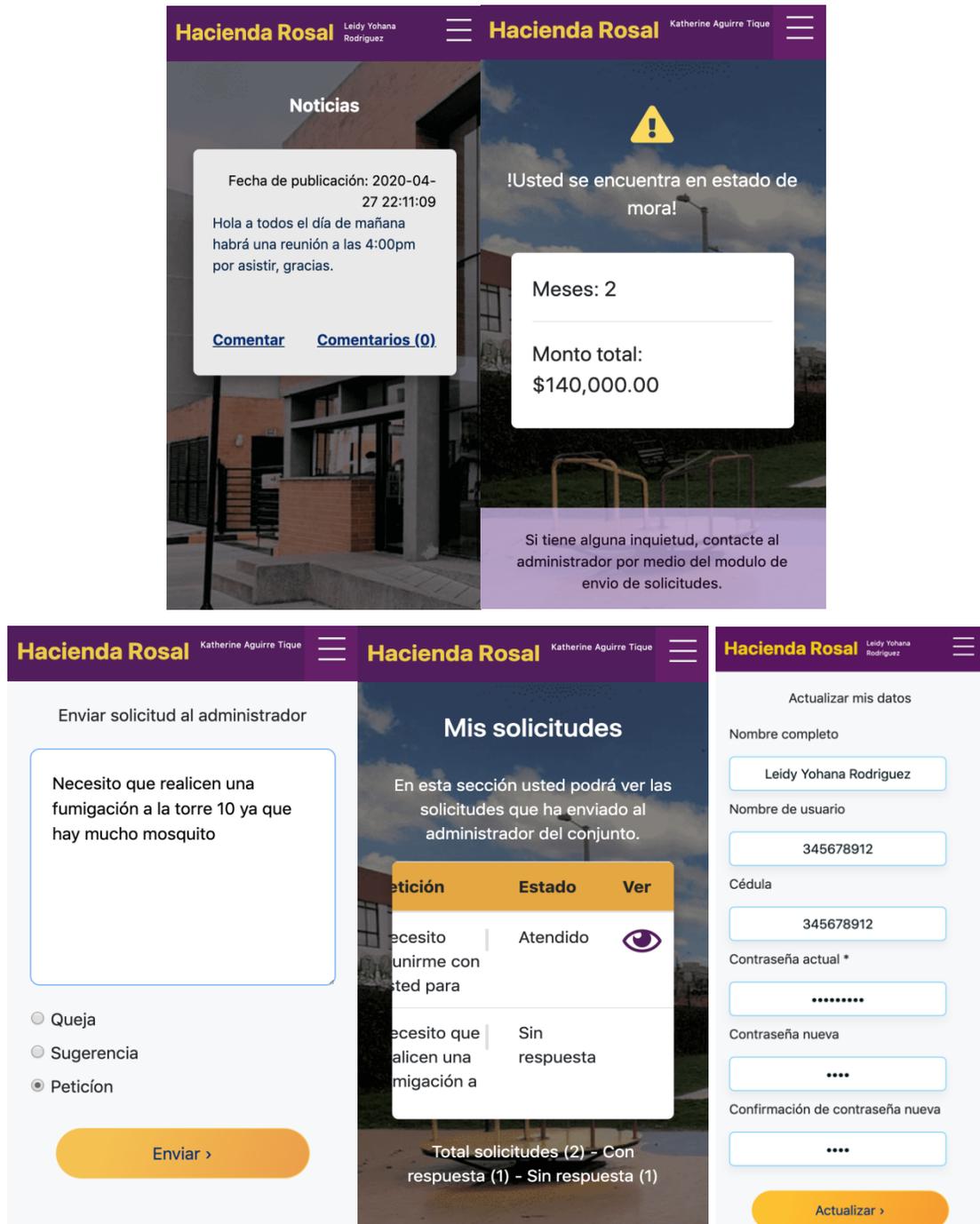
Figura 40. Pantallas para celular con funcionalidades de administrador



Fuente: elaboración propia

En la Figura 41 se pueden ver las pantallas para celular con las funcionalidades para los residentes, estas pantallas equivalen a las anteriormente descritas.

Figura 41. Pantallas para celular con funcionalidades para residentes



Fuente: elaboración propia

Con los resultados obtenidos en este proyecto, se facilita la gestión de cartera y comunicación entre los residentes y el administrador del conjunto residencial Hacienda El Rosal de Funza Cundinamarca, por medio de la publicación de comunicados, avisos, quejas, sugerencias y notificación personalizada al celular de cada residente.

6. CONCLUSIONES Y RECOMENDACIONES

El proyecto que se realizó contribuyó a identificar y resaltar los lineamientos a tener en cuenta al momento de implementar de manera exitosa un sistema de información. En este caso, una aplicación web en donde se maneja información del conjunto residencial Hacienda el Rosal.

La aplicación web que se desarrolló, satisface todas las necesidades requeridas por el conjunto residencial, para los procesos de comunicación y estado de la cartera de pagos de administración. Ahora con la aplicación web, el administrador puede realizar estos procesos de manera más fácil y ahorrando recursos.

Dentro de los puntos que se considera de más importancia en este proyecto, es haber podido detectar todas las necesidades de los residentes y el administrador. El llevar a cabo un buen análisis y diseño de la aplicación, sentó la base para obtener un buen resultado al momento de desarrollarla y ponerla en marcha.

Las tecnologías que se usaron para el desarrollo de esta aplicación web, hoy en día son bastante demandadas por su buena capacidad para desarrollos a gran escala. El entender cómo construir microservicios con Java Spring Boot, interfaces gráficas con Angular, aplicación de los principios de arquitectura REST, uso de los servicios de Amazon Web Services para implementar la aplicación en la nube y tener en cuenta las mejores prácticas de desarrollo y calidad de software, hizo que las estudiantes aumentaran la capacidad y experiencia para ser mejores desarrolladoras de software y así facilitar el ingreso al mundo laboral.

Esta aplicación tiene una gran oportunidad de seguir creciendo, debido a que se realizó con un buen diseño y mejores prácticas de desarrollo, una de estas, es tener

un análisis de los datos que se recopilan diariamente. Se recomienda implementar técnicas de analítica de datos, como el uso de un Dashboard (panel de control o informe de datos recopilados) que proporcione información, como por ejemplo: ¿Cuál es el residente que más seguido debe dinero a la administración?, ¿Cuál es el residente que siempre ha estado al día con sus pagos?, ¿Cuántos residentes al día envían solicitudes a la administración?, ¿Cuál es el residente que más se queja?, ¿Cuál es el residente que más usa la aplicación?, ¿Qué tanto se usa la aplicación?, entre otras. El tener respuesta a estas preguntas, permitirá que la administración del conjunto residencial, tome mejores decisiones para su gestión.

7. BIBLIOGRAFÍA

- Angular (s.f) Introduction to Angular concepts. Recuperado de <https://angular.io/guide/architecture>
- AWS (2019). Informática en la nube con AWS. Recuperado de <https://aws.amazon.com/es/what-is-aws/>
- Copropiedad (2019) Copropiedad, software para administradores de propiedades horizontales, Recuperado de <https://www.copropiedad.co/>
- Decreto 1317. (27 Junio 2013). Ministerio de comercio, industria y turismo. Recuperado de https://www.mintic.gov.co/portal/604/articles-4274_documento.pdf
- Decreto 1360 de 1989. (Junio 23 del 2007). Por el cual se reglamenta la inscripción del soporte lógico (software) en el Registro Nacional del Derecho de Autor. Recuperado de <http://derechodeautor.gov.co/documents/10181/187670/DECRETO+1360+DE+1989.pdf/db076bd8-f8b8-41d2-8f39-76faa2e212db>
- Diario Oficial. (4 de Agosto de 2001). Ley 675 de 2001 Nivel Nacional. Recuperado de http://www.sic.gov.co/sites/default/files/normatividad/Ley_675_2001.pdf
- Homify (2015) Conjunto residencial: ideas, arquitectura e imágenes. Recuperado de <https://www.homify.com.co/espacios/conjunto-residencial>
- Ing. Software del ITSON. (2009). Definición de modelo de cascada. Recuperado de <http://definicionmodelocascada.blogspot.com>
- Mercado libre (2013). Software web para la administración y gerencia de propiedad horizontal. Recuperado de https://articulo.mercadolibre.com.co/MCO-460666311-software-web-administracion-y-gerencia-propiedad-horizontal-_JM?quantity=1#position=9&type=item&tracking_id=005a1dbb-bfa7-4562-8de0-6be6ced4ecce
- Mateu, C., (2004). Software libre. Desarrollo de aplicaciones web. Barcelona, España: Fundació per a la Universitat Oberta de Catalunya
- Nicole Chapaval (2017). Front-end y Back-end. Recuperado de www.platzi.com/blog/que-es-frontend-y-backend/

- Oracle Corporation (s.f) Definición de java. Recuperado de https://www.java.com/es/download/faq/whatis_java.xml
- Oscar, B. (2018). Experiencia en el desarrollo de software y en la industria de las tecnologías de la información. Recuperado de <https://www.oscarblancartebl og.com/2018/07/17/spring-boot-relacion-los-microservicios/>
- PHI-Horizontal, (s.f) PHI-horizontal, información de la aplicación web. Recuperado de <https://phi-horizontal.webnode.es/#>
- Robles, V. (2017). Desarrollador web - Formador online - Blogger. Recuperado de <https://victorroblesweb.es/2017/08/05/que-es-angular-y-para-que-sirve/>
- Rodríguez, A. (2010) Importancia de la Ingeniería de Software para lograr mayor calidad en el desarrollo de videojuegos. Propuesta de una metodología de desarrollo para videojuegos. Recuperado de https://www.ecured.cu/Metodolog%C3%ADa_de_cascada
- Softwaretestinghelp (2019) Que es el modelo en cascada. Recuperado de <https://www.softwaretestinghelp.com/what-is-sdlc-waterfall-model/>
- Spring (s.f) Overview. Recuperado de <https://spring.io/why-spring>
- TIC Portal (2019). Waterfall: metodología para el desarrollo secuencial de tareas. Recuperado de <https://www.ticportal.es/glosario-tic/waterfall-metodologia-desarrollo-secuencial>
- Tutorialspoint (s.f) Sprint Boot Introducción. Recuperado de https://www.tutorialspoint.com/spring_boot/spring_boot_introduction.htm#
- Vega, A. (s.f) Responsive Web Design: Interfaces Web Adaptables al dispositivo empleando HTML5 y CSS3 (trabajo de grado, Universidad Politécnica superior). Recuperado de <https://ebuah.uah.es/dspace/bitstream/handle/10017/19972/Memoria.pdf?sequence=1&isAllowed=y>
- WIBOO (2017). Desarrollo de aplicaciones. Madrid. Recuperado de <https://wiboomeia.com/que-son-las-aplicaciones-web-ventajas-y-tipos-de-desarrollo-web/>
- Wikipedia (2019) Front-end y Back-end. Recuperado de https://es.wikipedia.org/wiki/Front-end_y_back-end

ANEXO A. HISTORIAS DE USUARIOS

1. INCREMENTO 1:

De la Tabla 2 hasta la Tabla 14, se muestran las historias de usuario del incremento 1.

Tabla 4. Cargar el archivo de Excel con datos de los residentes

Número: 1	Usuario: Administrador																																			
Nombre de la historia de usuario: cargar el archivo de Excel con los datos de los residentes.																																				
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja																																			
Programador responsable: Leidy Yohana Rodríguez Fuentes																																				
Descripción: yo como administrador deseo cargar un archivo Excel con los datos básicos de los residentes, para poderlos visualizar y registrar en la aplicación.																																				
Criterios de aceptación:																																				
6. La aplicación debe ser capaz de adjuntar archivos solamente Excel.																																				
7. La estructura del archivo Excel debe ser la siguiente:																																				
<table border="1"> <thead> <tr> <th>torre</th> <th>apartamento</th> <th>documento</th> <th>nombre</th> <th>celular</th> <th>meses_deuda</th> <th>total_deuda</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>101</td> <td>1234567890</td> <td>Katherine Aguirre Tique</td> <td>3124922928</td> <td>1</td> <td>70000</td> </tr> <tr> <td>1</td> <td>102</td> <td>1234567896</td> <td>Leidy Rodriguez Fuentes</td> <td>3227525187</td> <td>2</td> <td>140000</td> </tr> <tr> <td>2</td> <td>101</td> <td>3456787654</td> <td>Pepito Perez</td> <td>3227623418</td> <td>0</td> <td>0</td> </tr> <tr> <td>3</td> <td>404</td> <td>3456543456</td> <td>Laura Sarmiento</td> <td>3125830084</td> <td>4</td> <td>280000</td> </tr> </tbody> </table>		torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda	1	101	1234567890	Katherine Aguirre Tique	3124922928	1	70000	1	102	1234567896	Leidy Rodriguez Fuentes	3227525187	2	140000	2	101	3456787654	Pepito Perez	3227623418	0	0	3	404	3456543456	Laura Sarmiento	3125830084	4	280000
torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda																														
1	101	1234567890	Katherine Aguirre Tique	3124922928	1	70000																														
1	102	1234567896	Leidy Rodriguez Fuentes	3227525187	2	140000																														
2	101	3456787654	Pepito Perez	3227623418	0	0																														
3	404	3456543456	Laura Sarmiento	3125830084	4	280000																														
Cada nombre de cada columna debe ser igual a la estructura definida anteriormente.																																				
8. En caso de no haber meses en deuda, deben colocarse 0 en meses deuda y total deuda.																																				
9. El administrador debe poder visualizar los registros cargados en la aplicación web.																																				
10. Los datos que deben aparecer en la aplicación web de cada residente son: torre apartamento, documento, nombre celular.																																				

Fuente: elaboración propia.

Tabla 5. Guardar los datos de los residentes en la base de datos

Número: 2	Usuario: Administrador
Nombre de la historia de usuario: guardar en base de datos y mostrar en la aplicación todos los datos que son recibidos del archivo Excel.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Programador responsable: Leidy Yohana Rodríguez Fuentes	
Descripción: yo como administrador requiero que al volver a la aplicación todos los datos que se cargaron en un principio del Excel aparezcan en una lista para interactuar con cada uno de ellos.	
Criterios de aceptación:	
1. Una vez se adjunte el archivo Excel, los datos deben ser guardados en base de datos.	
2. La aplicación debe mostrar la lista de todos los residentes que se cargaron anteriormente con el archivo Excel.	
3. En caso de fallar el guardado de los datos de los residentes en base de datos, se debe observar un mensaje de error y poder adjuntar el archivo Excel de nuevo.	
4. En caso de fallar el servicio que trae los residentes del conjunto registrados, debe aparecer un mensaje de error.	

Fuente: elaboración propia.

Tabla 6. Enviar notificaciones por mensajes de texto

Número: 3	Usuario: Administrador
Nombre de la historia de usuario: enviar notificaciones por mensajes de texto a todos los residentes a la vez.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Programador responsable: Leidy Yohana Rodríguez Fuentes	
Descripción: yo como administrador deseo enviar notificaciones por mensaje de texto a todos los residentes del conjunto a la vez para dar comunicados al instante.	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. La aplicación debe mostrar un mensaje con una entrada de texto al momento que el administrador le dé clic en “notificar a todos” en donde podrá ingresar el mensaje que desea enviar a todos los residentes del conjunto. 2. Al momento que el mensaje sea enviado con éxito, la aplicación debe mostrar un mensaje indicando que el envío fue exitoso. 3. Si el mensaje no fue enviado por algún error, la aplicación debe mostrar un mensaje indicando que hubo un error al enviar la notificación. 4. No se debe poder enviar un mensaje que supere los 238 caracteres. 	

Fuente: elaboración propia.

Tabla 7. Enviar notificaciones por mensajes de texto de manera personalizada

Número: 4	Usuario: Administrador
Nombre de la historia de usuario: enviar notificaciones por mensajes de texto de manera personalizada.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Programador responsable: Leidy Yohana Rodríguez Fuentes	
Descripción: yo como administrador requiero enviar notificaciones por mensaje de texto de manera personalizada para dar un comunicado importante a la persona deseada.	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. La aplicación debe mostrar una lista de todos los residentes que están registrados en la aplicación con un botón al lado de cada registro “Notificar” 2. La aplicación debe mostrar un mensaje con una entrada de texto al momento que el administrador de clic en “notificar” en donde podrá ingresar el mensaje que desea enviar a un residente específico del conjunto. 3. Al momento que el mensaje sea enviado con éxito, la aplicación debe mostrar un mensaje indicando que el envío de la notificación fue exitoso. 4. Si el mensaje no fue enviado por algún error, la aplicación debe mostrar un mensaje indicando que hubo un error al enviar la notificación. 5. No se debe poder enviar un mensaje que supere los 238 caracteres. 	

Fuente: elaboración propia.

Tabla 8. Agregar un residente nuevo en la aplicación web

Número: 5	Usuario: Administrador
Nombre de la historia de usuario: agregar un residente nuevo en la aplicación web	
Prioridad en negocio: Media	Riesgo en desarrollo: Baja
Programador responsable: Leidy Yohana Rodríguez Fuentes	
Descripción: yo como administrador deseo agregar un residente nuevo en caso de no haberse adjuntado en el archivo Excel.	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. La aplicación debe mostrar un botón con la opción de agregar un residente nuevo en la aplicación. 2. Se deben tener los mismos campos que están en la estructura del Excel y llenarlos de la misma manera, por lo tanto, la aplicación debe poder validar los campos. 3. Cuando se guarde el nuevo residente, la aplicación debe: guardar en base de datos dicha información, aparecer un mensaje de guardado con éxito y aparecer en la lista existente en la aplicación con la información ingresada. 4. Al momento de que falle el guardado de un nuevo residente, la aplicación debe mostrar un modal indicando que hubo un error. 	

Fuente: elaboración propia.

Tabla 9. Editar datos de cada residente registrado en la aplicación

Número: 6	Usuario: Administrador
Nombre de la historia de usuario: editar datos de cada residente registrado en la aplicación.	
Prioridad en negocio: Media	Riesgo en desarrollo: Baja
Programador responsable: Leidy Yohana Rodríguez Fuentes	
Descripción: yo como administrador deseo editar los datos cada residente para mantener la información actualizada.	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. Al momento de editar algún dato de algún residente, debe aparecer un botón con el ícono de guardado y se debe poder editar los campos: teléfono celular, número de meses en deuda y deuda total. 2. En caso de fallar la actualización de los datos de los residentes en base de datos, se debe observar un modal de error. 	

Fuente: elaboración propia.

Tabla 10. Filtrar por torre y apartamento los residentes del conjunto

Número: 7	Usuario: Administrador
Nombre de la historia de usuario: filtrar por torre y apartamento los residentes del conjunto.	
Prioridad en negocio: Baja	Riesgo en desarrollo: Baja
Programador responsable: Leidy Yohana Rodríguez Fuentes	
Descripción: yo como administrador deseo poder filtrar por torre y apartamento los residentes registrados en la aplicación para acceder a su información de manera más rápida.	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. La aplicación debe mostrar un campo para ingresar el número de torre y apartamento a filtrar con un botón al lado de búsqueda. 2. La aplicación debe mostrar en la lista la información del residente que se filtró por número de torre y apartamento en el formato 12-101, por ejemplo. 3. Se debe validar el número de torre y apartamento con el siguiente formato: número de torre- número de apartamento por ejemplo 12-101. 4. En caso de que el formato de entrada en el filtro no sea válido, el botón de búsqueda no debe activarse. 	

Fuente: elaboración propia.

Tabla 11. Filtrar residentes en estado mora por pago de administración.

Número: 8	Usuario: Administrador
Nombre de la historia de usuario: filtrar residentes en estado mora por pago de administración.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Programador responsable: Leidy Yohana Rodríguez Fuentes	
Descripción: yo como administrador deseo ver filtrados a todos los residentes que se encuentren en estado de mora por pagos de administración para tener un control y seguimiento sobre ellos.	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. La aplicación debe mostrar en el módulo de administrar morosos, a todos los residentes en estado de mora con los siguientes datos: torre-apartamento, nombre, cédula, meses en mora, deuda total y un botón "Enviar mensaje a todos". 2. En cada uno de los registros debe aparecer un botón "Notificar". 	

Fuente: elaboración propia.

Tabla 12. Notificar por mensajes de texto a todos los residentes en estado de mora a la vez.

Número: 9	Usuario: Administrador
Nombre de la historia de usuario: notificar por mensajes de texto a todos los residentes en estado de mora a la vez.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Programador responsable: Leidy Yohana Rodríguez Fuentes	
Descripción: yo como administrador deseo notificar por mensaje de texto a todos los residentes en estado de mora al tiempo para poder darles un comunicado en general.	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. Cuando el administrador de clic en “Enviar mensaje a todos”, la aplicación debe mostrar un modal en donde se ingresa el mensaje que desea que les lleguen a todos los residentes morosos del conjunto. 2. Los mensajes de texto que envíe el administrador deben llegar al instante a todos los residentes morosos del conjunto, no debe tardar más de 5 segundos. 3. No debe poder ingresar un mensaje de más de 238 caracteres. 4. Al momento que el mensaje sea enviado con éxito, la aplicación debe mostrar un mensaje indicando que el envío fue exitoso. 5. Si el mensaje no fue enviado por algún error, la aplicación debe mostrar un mensaje indicando que hubo un error al enviar la notificación. 	

Fuente: elaboración propia.

Tabla 13. Notificar por mensajes de texto a los residentes morosos de manera personalizada.

Número: 10	Usuario: Administrador
Nombre de la historia de usuario: notificar por mensajes de texto a los residentes morosos de manera personalizada.	
Prioridad en negocio: Alta	Riesgo en desarrollo: Baja
Programador responsable: Leidy Yohana Rodríguez Fuentes	
Descripción: yo como administrador deseo notificar por mensaje de texto a los residentes morosos de forma personalizada, para dar un comunicado de acuerdo con la situación en que se encuentre.	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. Cuando el administrador de clic en “Notificar”, la aplicación debe mostrar un modal en donde se ingresará el mensaje que desea que le llegue al residente moroso de pago de administración. 2. Los mensajes de texto que envíe el administrador deben llegar al instante al residente moroso del conjunto, no debe tardar más de 5 segundos. 3. No debe poder ingresar un mensaje de más de 238 caracteres. 4. Al momento que el mensaje sea enviado con éxito, la aplicación debe mostrar un mensaje indicando que el envío fue exitoso. 5. Si el mensaje no fue enviado por algún error, la aplicación debe mostrar un mensaje indicando que hubo un error al enviar la notificación. 	

Fuente: elaboración propia.

Tabla 14. Realizar publicaciones de novedades y anuncios importantes en la aplicación web acerca de lo que ocurre dentro del conjunto

Número: 11	Usuario: Administrador
Nombre de la historia de usuario: realizar publicaciones de novedades y anuncios importantes en la aplicación web acerca de lo que ocurre dentro del conjunto.	
Prioridad en negocio: Media	Riesgo en desarrollo: Baja
Programador responsable: Leidy Yohana Rodríguez Fuentes	
Descripción: yo como administrador deseo realizar publicaciones sobre novedades y anuncios importantes en la aplicación web para que los residentes estén enterados de todo lo que pasa dentro del conjunto.	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. La aplicación debe tener un botón para añadir una publicación, este botón debe ser con el ícono del signo más. 2. Cuando el administrador de le clic en el botón de añadir publicación, deberá aparecer un modal en donde podrá ingresar la información de interés que desea publicar. 3. Una vez el administrador ingrese la información que desea comunicar, deberá poder dar clic en el botón publicar y esta se guardará en la sección de “publicaciones”. 4. En caso de que falle la publicación del anuncio, deberá aparecer un modal indicando que hubo un error al realizar la publicación. 	

Fuente: elaboración propia.

2. INCREMENTO 2:

De la Tabla 15 hasta la Tabla 20, se muestran las historias de usuario del incremento 2.

Tabla 15. Autenticar diferentes roles en la aplicación web

Número: 12	Usuario: Administrador, residente
Nombre de la historia de usuario: autenticar diferentes roles en la aplicación web	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Programador responsable: Leidy Yohana Rodríguez Fuentes	
Descripción: yo como usuario de la aplicación web deseo autenticarme por medio de un login para poder utilizar las funcionalidades que me ofrece la aplicación de acuerdo con mi rol.	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. La aplicación debe tener una pantalla de login, en donde se ingrese la cédula en los campos usuario y contraseña. 2. En caso de no existir el usuario debe aparecer un mensaje de error. 3. En caso de fallar la autenticación del usuario debe aparecer un mensaje de error. 	

Fuente: elaboración propia

Tabla 16. Recibir solicitudes de los residentes

Número: 13	Usuario: Administrador
Nombre de la historia de usuario: recibir solicitudes de los residentes	
Prioridad en negocio: Alta	Riesgo en desarrollo: Alta
Programador responsable: Leidy Yohana Rodríguez Fuentes	
Descripción: yo como administrador deseo recibir las diferentes solicitudes que realizan los residentes con el fin de estar mejor comunicado acerca de las situaciones de ellos.	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. La aplicación debe mostrar una lista de todas las solicitudes de los residentes. 2. En caso de no haber solicitudes, se debe mostrar un mensaje indicando que no hay solicitudes aún. 3. La aplicación debe tener un filtro de tipo de solicitud. 	

Fuente: elaboración propia.

Tabla 17. Enviar solicitudes al administrador

Número: 14	Usuario: Residente
Nombre de la historia de usuario: enviar solicitud al administrador	
Prioridad en negocio: Media	Riesgo en desarrollo: Baja
Programador responsable: Leidy Yohana Rodríguez Fuentes	
Descripción: yo como residente, deseo enviar una solicitud al administrador de acuerdo con mi interés, con el fin de que el administrador esté enterado de mi situación.	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. La aplicación debe mostrar diferentes opciones de tipo de solicitud (queja, sugerencia, petición). 2. En caso de fallar el envío de la solicitud, debe aparecer un mensaje de error. 3. En caso de enviar el mensaje correctamente, debe aparecer un mensaje de éxito. 	

Fuente: elaboración propia.

Tabla 18. Ver publicaciones que realiza el administrador

Número: 15	Usuario: Residente
Nombre de la historia de usuario: ver publicaciones que realiza el administrador	
Prioridad en negocio: Media	Riesgo en desarrollo: Baja
Programador responsable: Leidy Yohana Rodríguez Fuentes	
Descripción: yo como residente, deseo ver las publicaciones que realiza el administrador para poder estar enterado de todo lo que nos quiere comunicar.	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. La aplicación debe mostrar las publicaciones con fecha de publicación. 	

Fuente: elaboración propia.

Tabla 19. Comentar publicaciones que realiza el administrador

Número: 16	Usuario: Residente
Nombre de la historia de usuario: comentar publicaciones que realiza el administrador	
Prioridad en negocio: Media	Riesgo en desarrollo: Baja
Programador responsable: Leidy Yohana Rodríguez Fuentes	
Descripción: yo como residente, deseo comentar las publicaciones que realiza el administrador para que el administrador tenga una retroalimentación acerca de lo que está comunicando.	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. La aplicación debe mostrar las publicaciones, en ellas la opción de comentar. 2. Cuando el residente decida comentar una publicación, debe aparecer una pantalla donde el residente podrá ingresar su comentario y enviarlo. 3. En caso de falla el envío del comentario, debe aparecer un mensaje de error. 	

Fuente: elaboración propia.

Tabla 20. Ver comentarios que realizan los residentes sobre las publicaciones

Número: 17	Usuario: Administrador
Nombre de la historia de usuario: ver comentarios que realizan los residentes sobre las publicaciones	
Prioridad en negocio: Media	Riesgo en desarrollo: Baja
Programador responsable: Leidy Yohana Rodríguez Fuentes	
Descripción: yo como administrador deseo ver los comentarios que hacen los residentes acerca de las publicaciones que realizó para tenerlas en cuenta en mi gestión.	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. La aplicación debe mostrar en cada publicación el número de comentarios que se han hecho sobre ella. 2. Al dar clic sobre el comentario debe aparecer una pantalla mostrando la lista de comentarios que se han hecho sobre esa publicación. 	

Fuente: elaboración propia.

3. INCREMENTO 3:

De la Tabla 21 hasta la Tabla 25, se muestran las historias de usuario del incremento 3.

Tabla 21. Ver solicitudes enviadas al administrador

Número: 18	Usuario: Residente
Nombre de la historia de usuario: ver solicitudes enviadas al administrador	
Prioridad en negocio: Media	Riesgo en desarrollo: Baja
Programador responsable: Leidy Yohana Rodríguez Fuentes	
Descripción: yo como residente quiero ver las solicitudes que he enviado al administrador, con el fin de llevar un seguimiento de ellas.	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. La aplicación debe mostrar en una lista todas las solicitudes que cada residente a enviado al administrador. 2. En caso de que el residente no haya hecho solicitudes aún, debe aparecer un mensaje indicándolo. 	

Fuente: elaboración propia

Tabla 22. Ver respuesta de las solicitudes enviadas al administrador

Número: 19	Usuario: Residente
Nombre de la historia de usuario: ver respuesta de las solicitudes enviadas al administrador	
Prioridad en negocio: Media	Riesgo en desarrollo: Baja
Programador responsable: Leidy Yohana Rodríguez Fuentes	
Descripción: yo como residente quiero tener la opción de ver la respuesta que hizo el administrador a mis solicitudes.	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. La aplicación debe mostrar una opción de ver la respuesta que realizó el administrador sobre la solicitud. 2. Cuando el residente desee ver la respuesta, debe aparecer una pantalla con su respuesta dada. 3. En caso de que no se le haya dado respuesta a la solicitud, no deberá aparecer la opción de ver respuesta. 	

Fuente: elaboración propia.

Tabla 23. Ver estado de pagos de los residentes

Número: 20	Usuario: Residente
Nombre de la historia de usuario: ver estado de pagos de los residentes	
Prioridad en negocio: Media	Riesgo en desarrollo: Baja
Programador responsable: Leidy Yohana Rodríguez Fuentes	
Descripción: yo como residente quiero ver mi estado de pago para saber si estoy al día o tengo que ponerme al día con mis obligaciones.	
Criterios de aceptación:	
<ol style="list-style-type: none"> 1. La aplicación debe mostrar el estado de pago de cada residente indicando el monto total y la cantidad de meses que debe. 2. Si el residente se encuentra al día, en monto total y cantidad de meses deberá aparecer 0. 	

Fuente: elaboración propia.

Tabla 24. Responder a las solicitudes que envían los residentes

Número: 21	Usuario: Administrador
Nombre de la historia de usuario: responder a las solicitudes que envían los residentes	
Prioridad en negocio: Media	Riesgo en desarrollo: Baja
Programador responsable: Leidy Yohana Rodríguez Fuentes	
Descripción: yo como administrador deseo responder a las solicitudes que envían los residentes para tener una mejor comunicación.	
Criterios de aceptación: <ol style="list-style-type: none"> 1. La aplicación debe mostrar la opción de dar respuesta a cada solicitud. 2. Cuando el administrador decida responder a una solicitud, deberá aparecer una pantalla en donde ingrese la respuesta. 3. Deberá aparecer un mensaje de éxito al momento de enviar la respuesta con éxito. 4. Deberá aparecer un mensaje de error al momento de fallar el envío de la respuesta a esa solicitud. 	

Fuente: elaboración propia.

Tabla 25. Filtrar por estado (queja, sugerencia o petición) las solicitudes de los residentes

Número: 22	Usuario: Administrador
Nombre de la historia de usuario: filtrar por estado (queja, sugerencia o petición) las solicitudes de los residentes	
Prioridad en negocio: Media	Riesgo en desarrollo: Baja
Programador responsable: Leidy Yohana Rodríguez Fuentes	
Descripción: yo como administrador deseo filtrar por estado de las solicitudes (atendido o sin respuesta) para poder tener una mejor visualización de las solicitudes.	
Criterios de aceptación: <ol style="list-style-type: none"> 1. La aplicación debe mostrar la opción de filtrar por estado de solicitud (atendido o sin respuesta) 2. Se deben filtrar todas las solicitudes de acuerdo a la opción elegida. 	

Fuente: elaboración propia.

ANEXO B. DIAGRAMAS UML

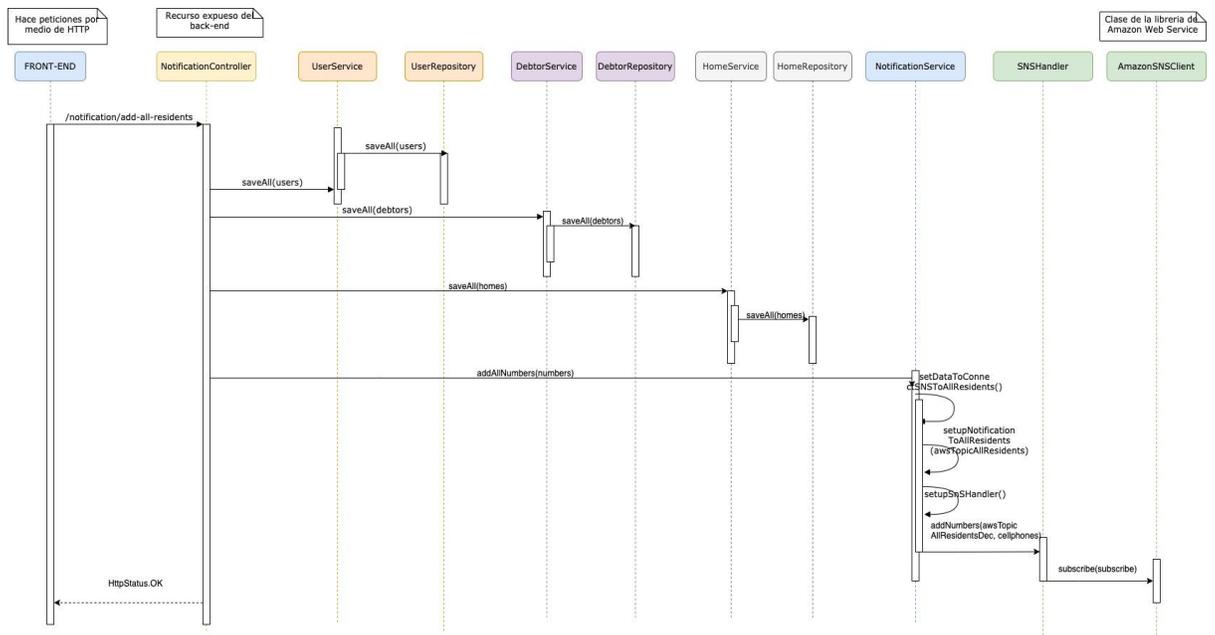
1. DIAGRAMAS DE SECUENCIA

1.1. INCREMENTO 1

De la Figura 42 a la Figura 46, se muestran los diagramas de secuencia del incremento 1.

- Registro de los residentes en la aplicación cargados del archivo Excel.

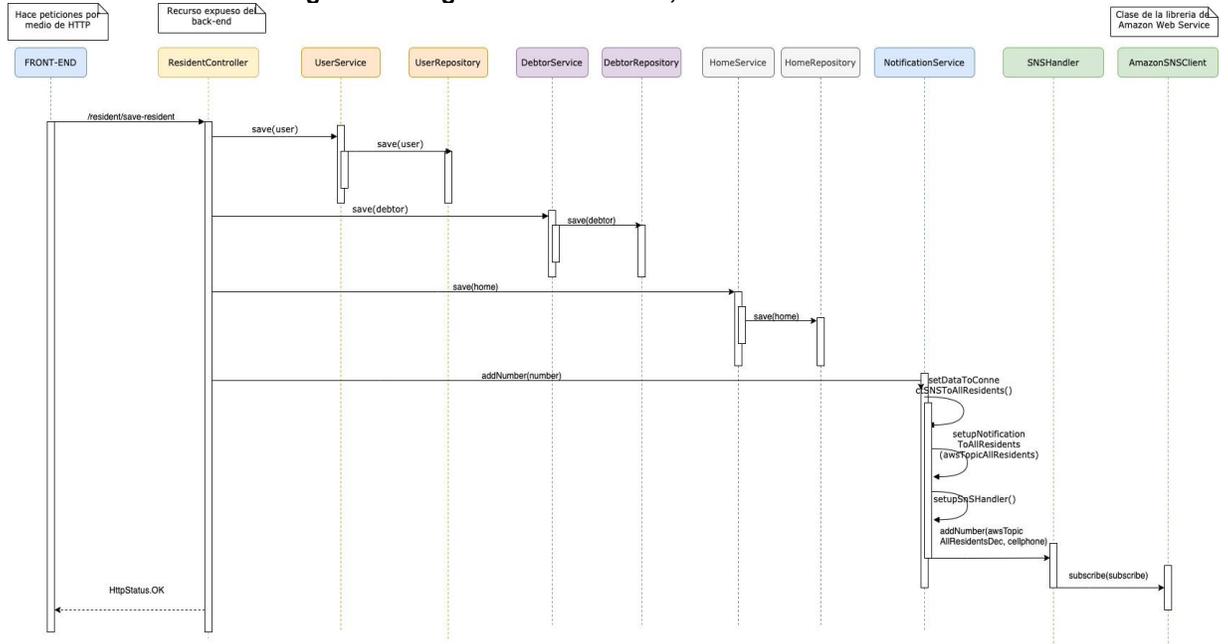
Figura 42. Diagrama de secuencia, registro de los residentes



Fuente: elaboración propia

- Añadir a un residente en la aplicación

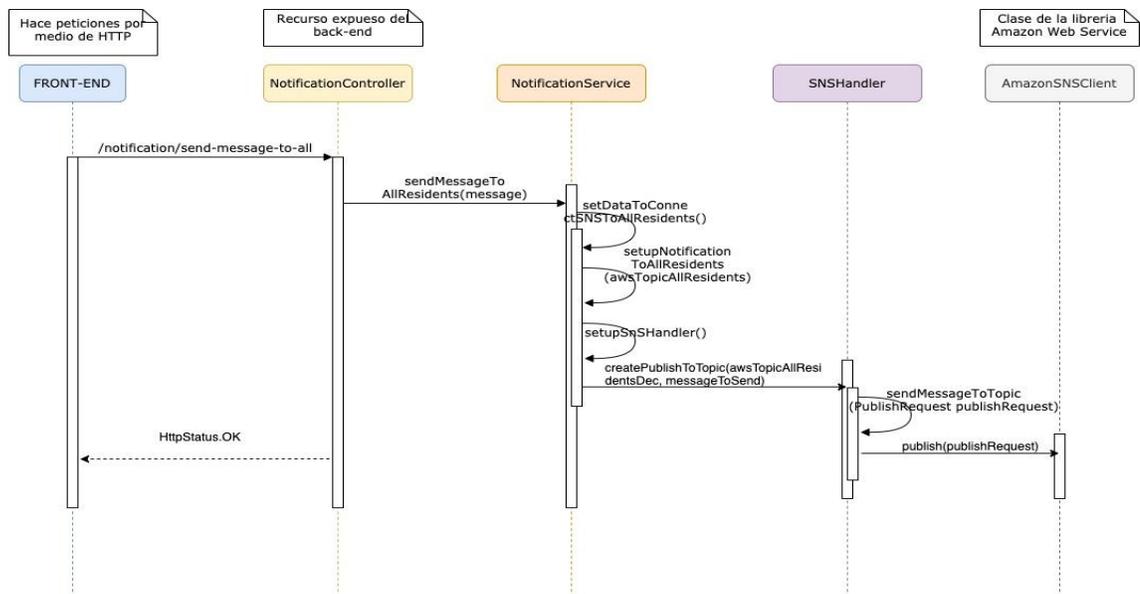
Figura 43. Diagrama de secuencia, añadir un residente



Fuente: elaboración propia

- Notificación por mensaje de texto a todos los residentes del conjunto

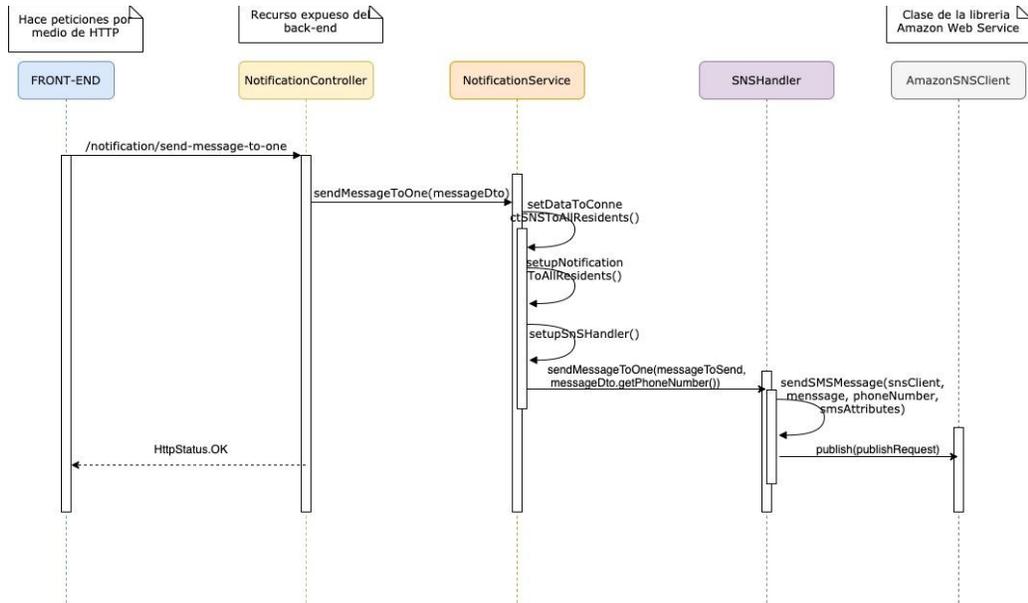
Figura 44. Diagrama de secuencia, notificación por mensaje a todos los residentes



Fuente: elaboración propia

- Notificación por mensaje de texto a un solo residente del conjunto.

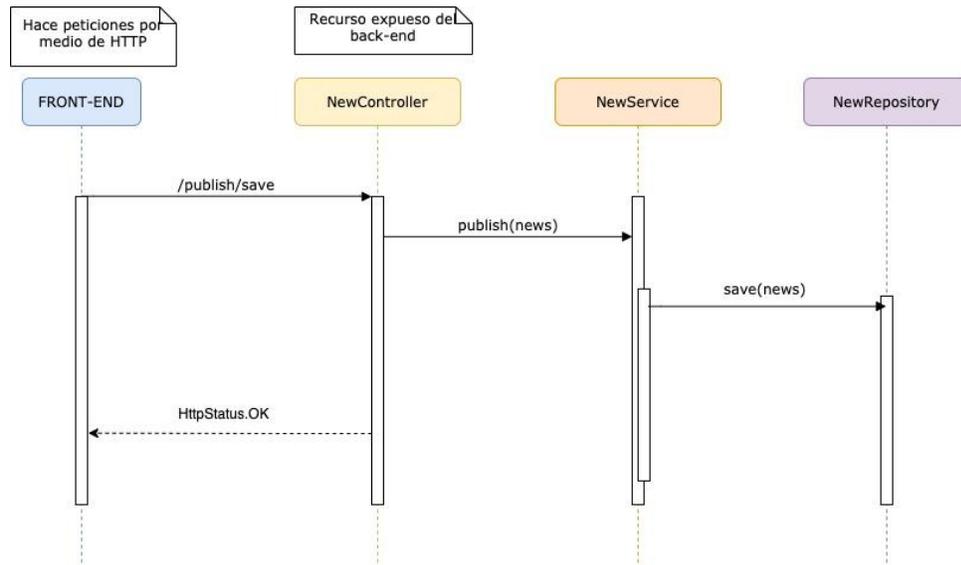
Figura 45. Diagrama de secuencia, notificación por mensaje a solo un residente



Fuente: elaboración propia

- Publicación de un anuncio o noticia en la aplicación

Figura 46. Diagrama de secuencia, publicación de un anuncio



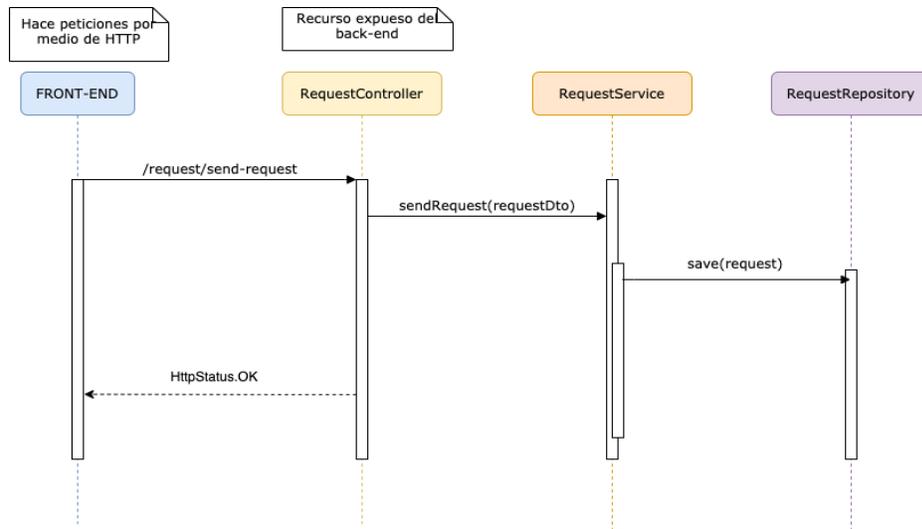
Fuente: elaboración propia

1.2. INCREMENTO 2

De la Figura 42, se muestran los diagramas de secuencia del incremento 2.

- Envío de solicitudes al administrador.

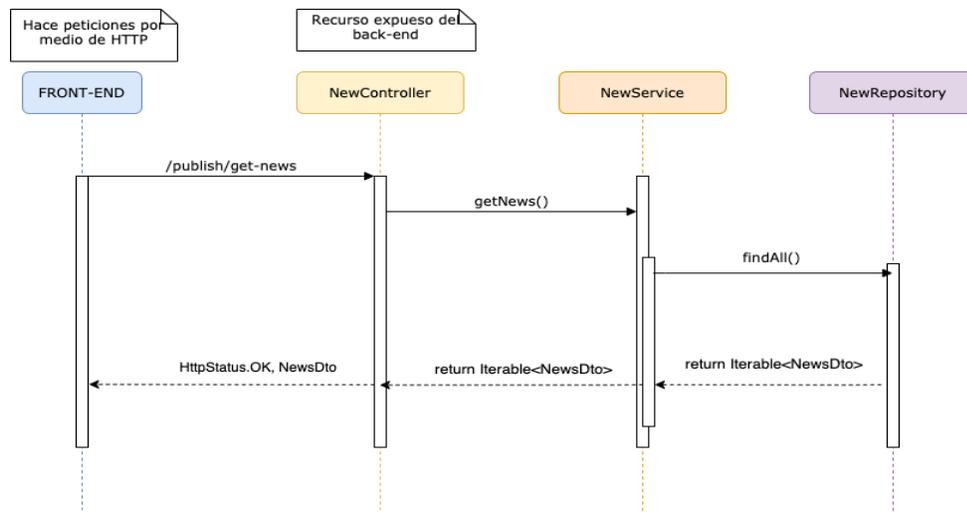
Figura 47. Diagrama de secuencia, envío de solicitud al administrador



Fuente: elaboración propia

- Ver las publicaciones que realiza el administrador.

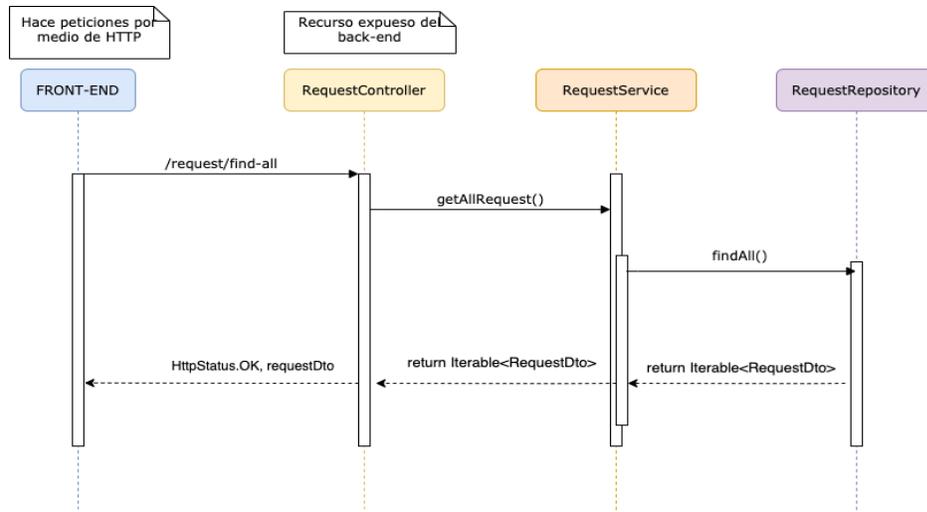
Figura 48. Diagrama de secuencia, publicaciones de la administración



Fuente: elaboración propia

- Recibir solicitudes de los residentes

Figura 49. Diagrama de secuencia, solicitudes de los residentes



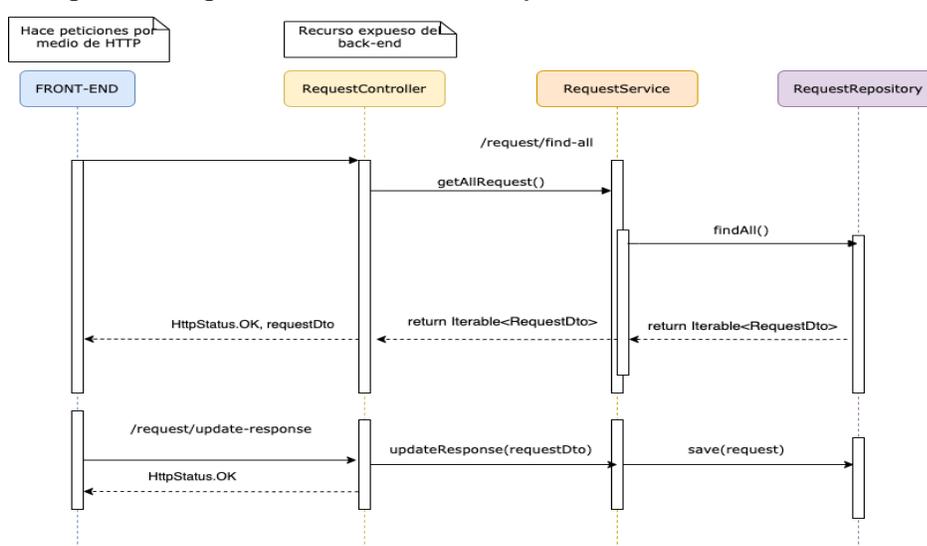
Fuente: elaboración propia

1.3. INCREMENTO 3

De la Figura 42Figura 52, se muestran los diagramas de secuencia del incremento 3.

- Dar respuesta a las solicitudes de los residentes.

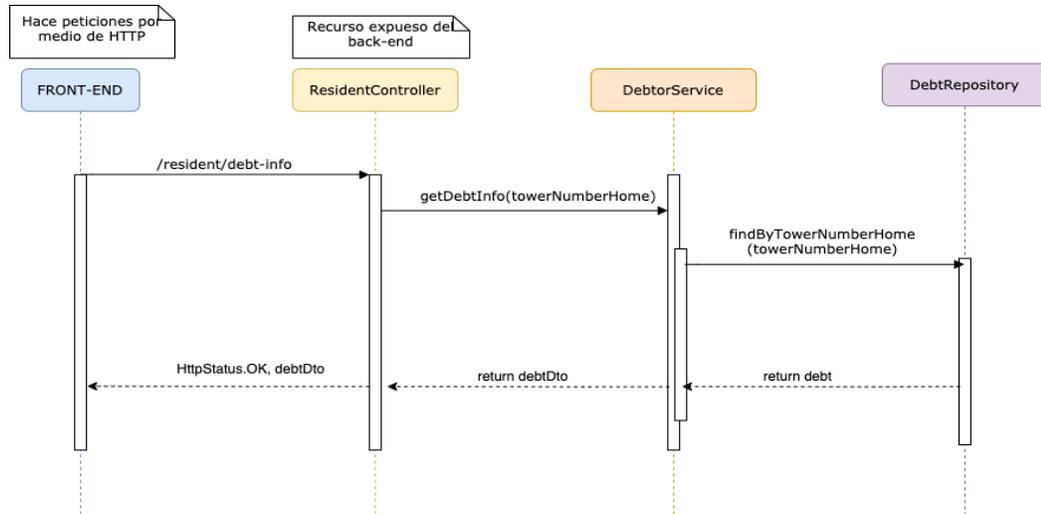
Figura 50. Diagrama de secuencia, dar respuesta a las solicitudes de residentes



Fuente: elaboración propia

- Ver estado de pagos por parte de los residentes.

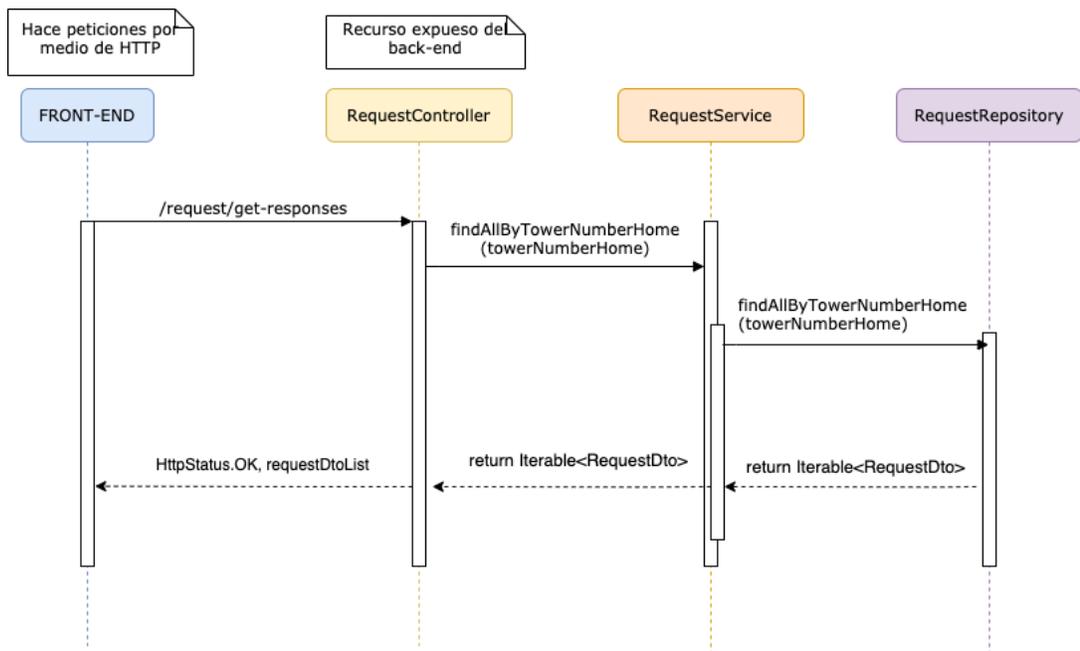
Figura 51. Diagrama de secuencia, ver estados de pago



Fuente: elaboración propia

- Ver solicitudes enviadas al administrador

Figura 52. Diagrama de secuencia, solicitudes al administrador



Fuente: elaboración propia

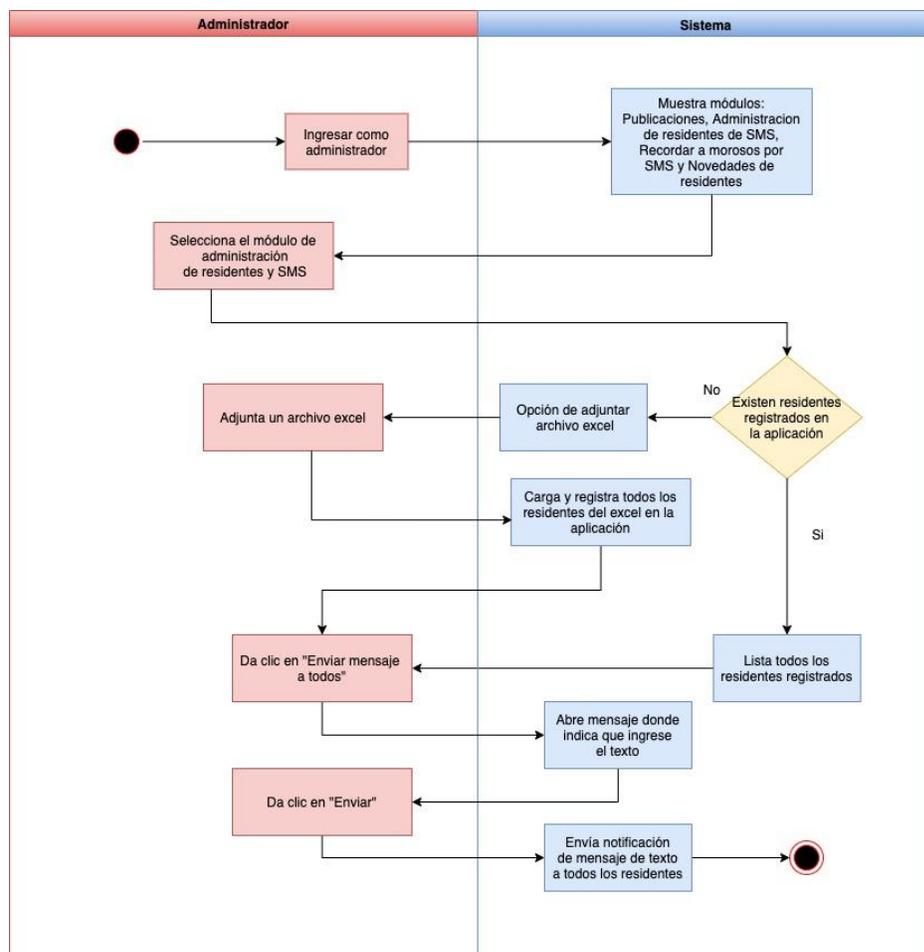
2. DIAGRAMAS DE ACTIVIDADES

2.1. INCREMENTO 1

En las Figura 53 y Figura 54, se muestran los diagramas de actividades del incremento 1.

- Envío de mensaje de texto a todos los residentes del conjunto residencial

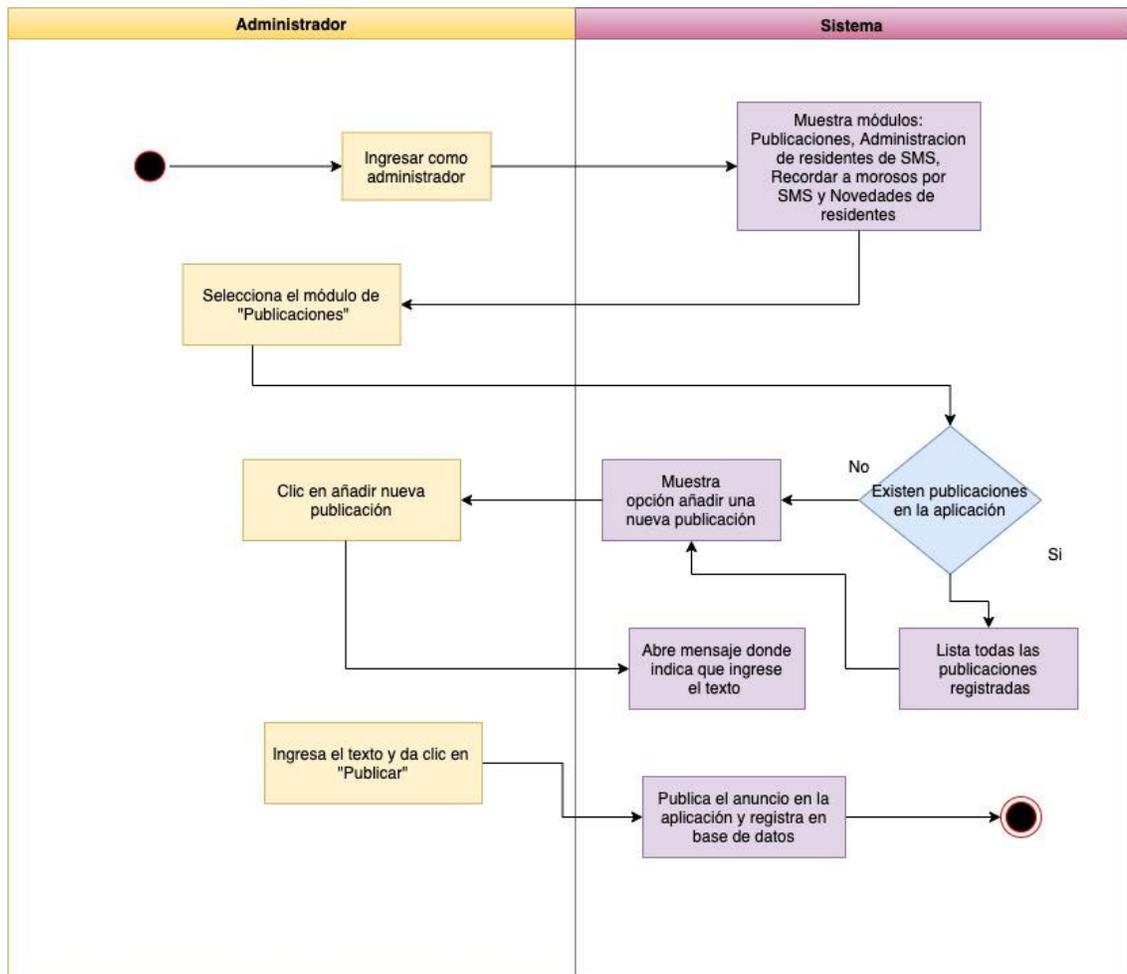
Figura 53. Diagrama de actividades, envío mensaje



Fuente: elaboración propia

- Publicar una noticia en la aplicación.

Figura 54. Diagrama de actividades, publicar una notificación

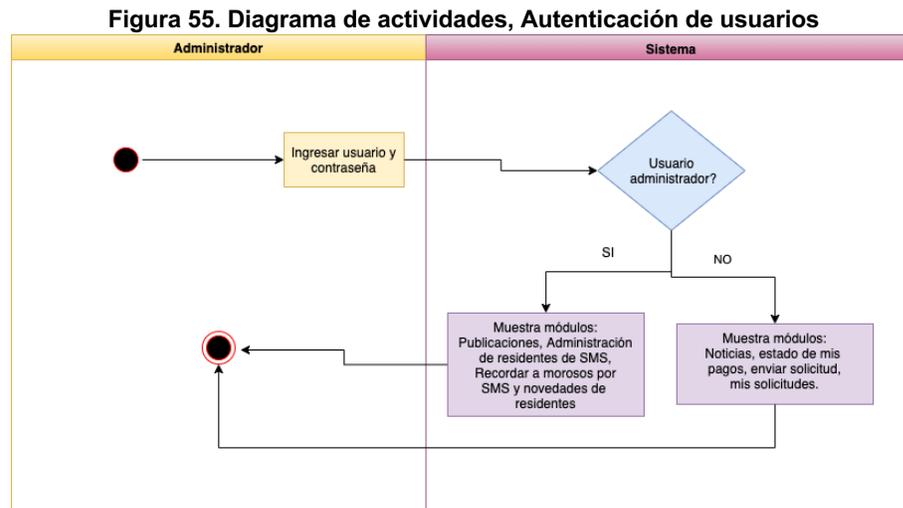


Fuente: elaboración propia

2.2. INCREMENTO 2:

De la Figura 55, se muestran los diagramas de actividades del incremento 2.

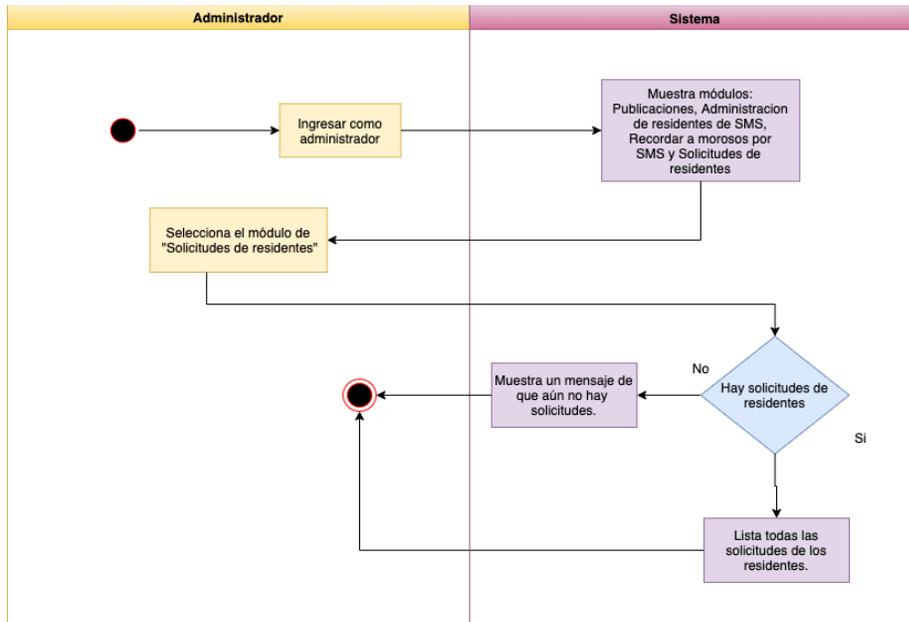
- Autenticación de diferentes roles en la aplicación.



Fuente: elaboración propia

- Recibir solicitudes de los residentes

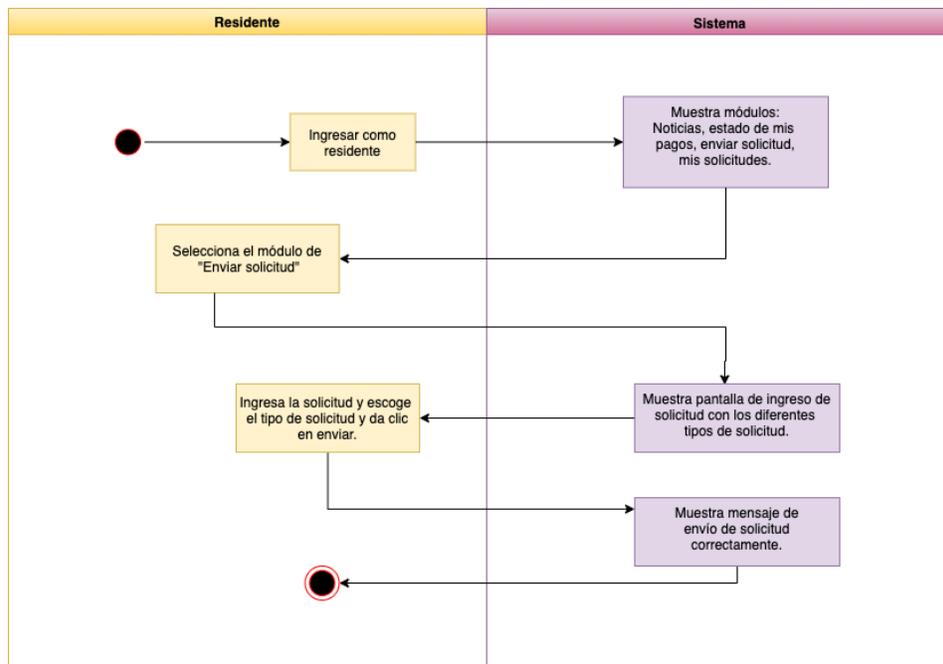
Figura 56. Diagrama de actividades, solicitud de los residentes



Fuente: elaboración propia

- Enviar solicitud al administrador

Figura 57. Diagrama de actividades, solicitud de la administración



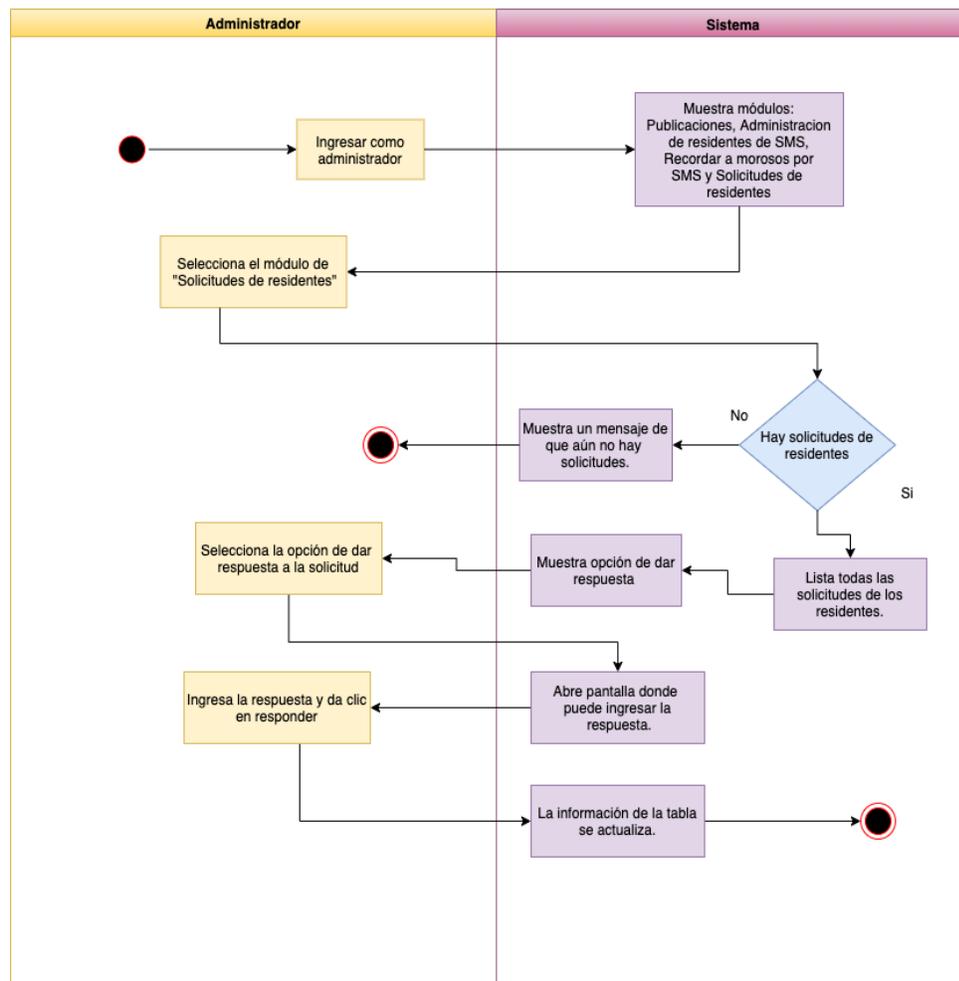
Fuente: elaboración propia

2.3. INCREMENTO 3:

De la Figura 58, se muestran los diagramas de actividades del incremento 2.

- Dar respuesta de las solicitudes de los residentes.

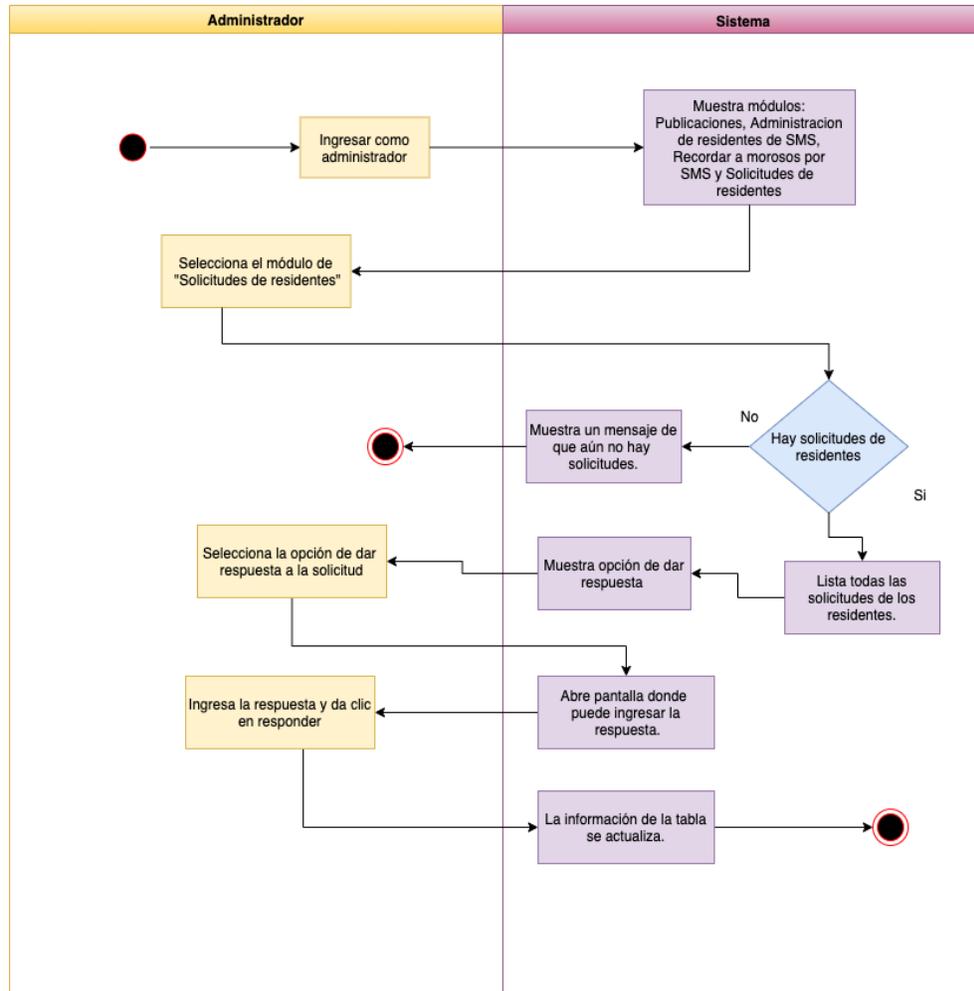
Figura 58. Diagrama de actividades, solicitudes de los residentes



Fuente: elaboración propia

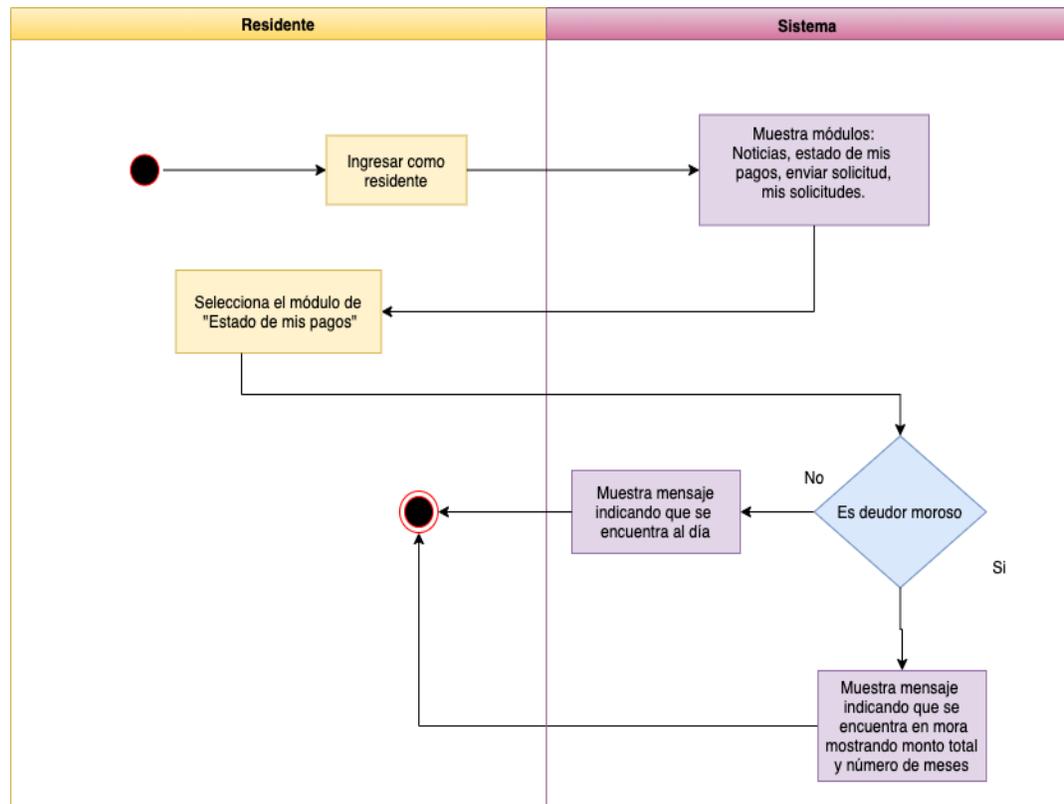
- Ver estado de pagos de los residentes.

Figura 59. Diagrama de actividades, ver estado de pagos de los residentes



Fuente: elaboración propia

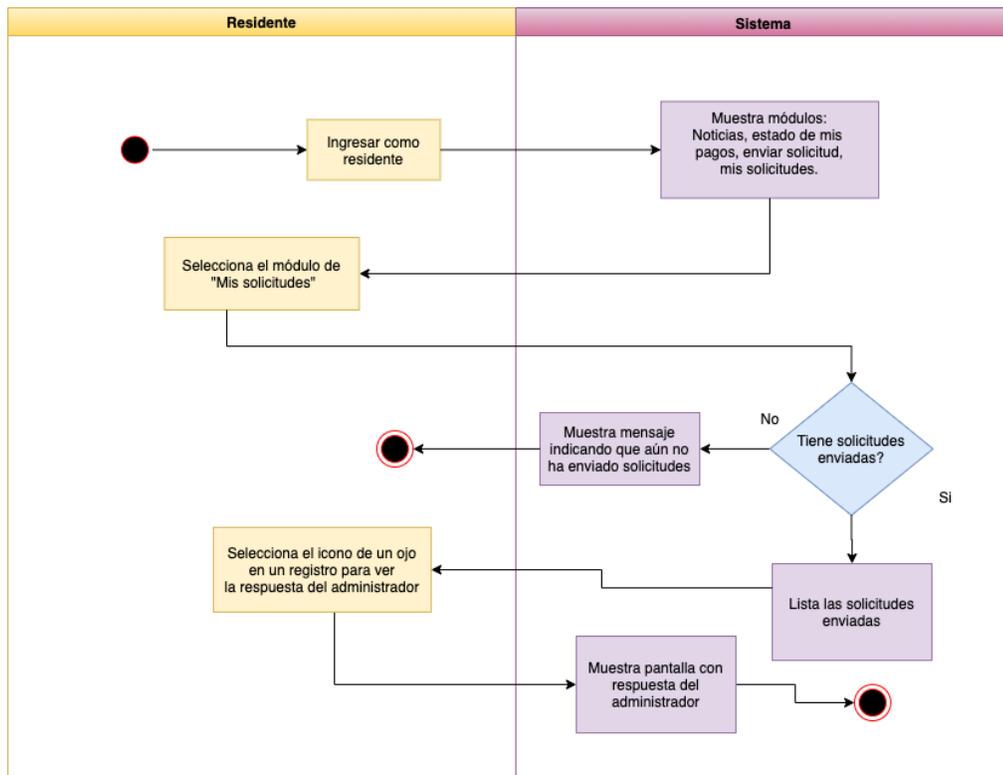
Figura 60. Diagrama de actividades, ver estados de pagos



Fuente: elaboración propia

- Ver respuesta de las solicitudes enviadas al administrador por los residentes del conjunto.

Figura 61. Diagrama de actividades, solicitudes enviadas a administración

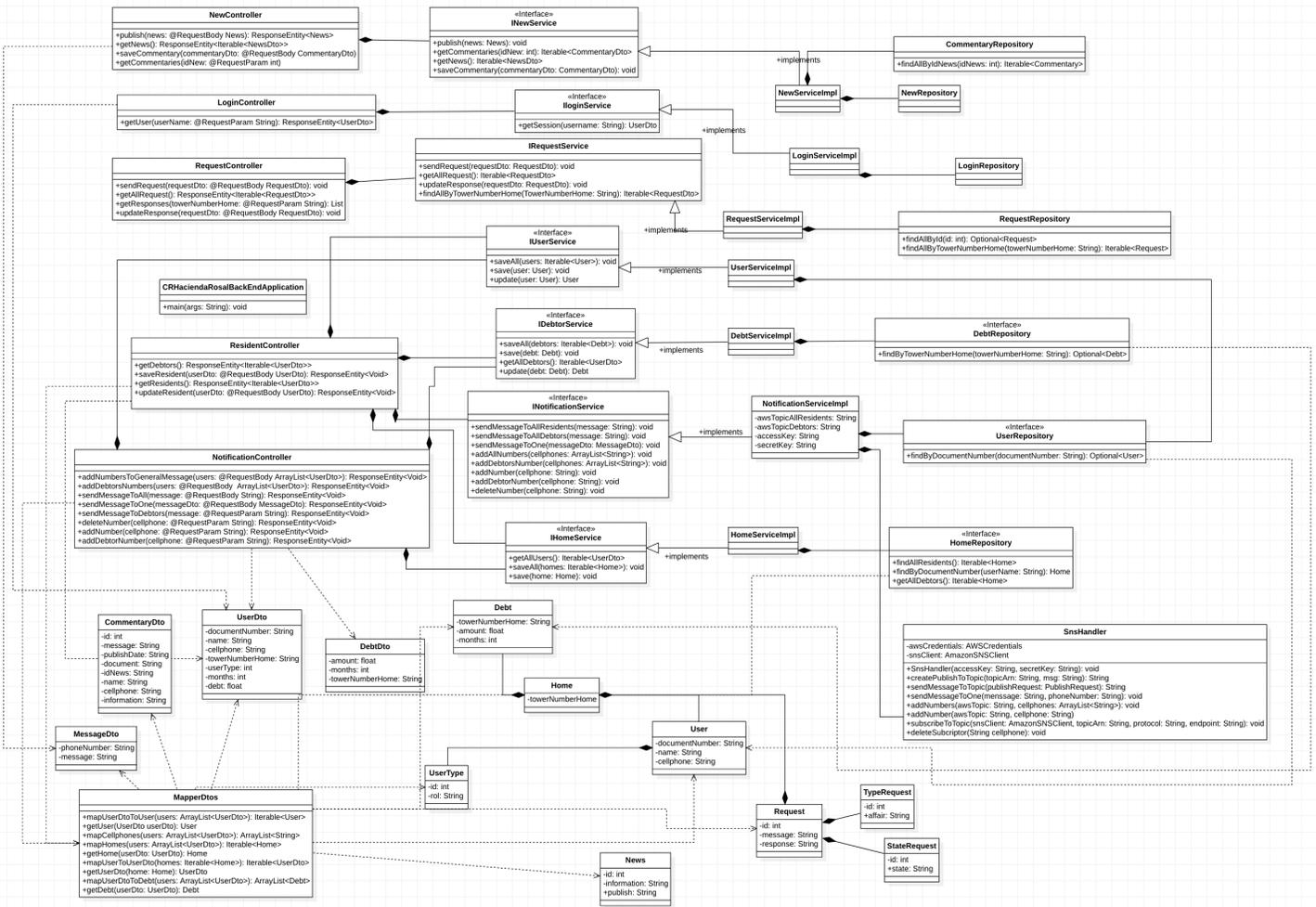


Fuente: elaboración propia

3. DIAGRAMA DE CLASES

En la Figura 62, se observa el diagrama de clases de la aplicación.

Figura 62. Diagrama de clase



Fuente: elaboración propia

ANEXO C. CASOS DE PRUEBA

1. INCREMENTO 1

De la Tabla 26 a la Tabla 48, se encuentran los casos de prueba del incremento 1.

Tabla 26. Caso de prueba relacionado con cargar el archivo Excel

Número de caso de prueba: 1		Prioridad de la prueba: Alta																													
Historia de usuario: cargar el archivo de Excel con los datos de los residentes.																															
Descripción del caso de prueba: cargar correctamente un archivo Excel en la aplicación.																															
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado																													
Pre-requisitos: 4. Ingresar a la aplicación como administrador. 5. Ingresar al módulo de Administración de residentes y SMS. 6. No tener ningún residente registrado en la aplicación.		Datos de prueba: 2. Documento Excel con los siguientes campos: <table border="1" data-bbox="954 829 1474 892"> <thead> <tr> <th>torre</th> <th>apartamento</th> <th>documento</th> <th>nombre</th> <th>celular</th> <th>meses_deuda</th> <th>total_deuda</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>102</td> <td>123456789</td> <td>David Rodriguez Fuentes</td> <td>3227623418</td> <td>1</td> <td>70000</td> </tr> <tr> <td>2</td> <td>101</td> <td>543234567</td> <td>Katherine Aguirre Tique</td> <td>3124922928</td> <td>2</td> <td>140000</td> </tr> <tr> <td>3</td> <td>404</td> <td>432345432</td> <td>Pepito Perez</td> <td>3125256290</td> <td>0</td> <td>0</td> </tr> </tbody> </table> (no importa el nombre del archivo Excel)		torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda	1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000	2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000	3	404	432345432	Pepito Perez	3125256290	0	0
torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda																									
1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000																									
2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000																									
3	404	432345432	Pepito Perez	3125256290	0	0																									
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)																												
Seleccionar la opción de adjuntar archivo	Observar una pantalla para elegir el archivo Excel.	Se observa la pantalla para adjuntar archivo Excel.																													
Seleccionar el archivo Excel de prueba	Observar el nombre del archivo en la aplicación	Se observó nombre del archivo en la aplicación																													
Dar clic en la opción cargar	Observar una barra de espera cuando se esté cargado el archivo en la aplicación y observar una tabla con los datos cargados.	Se observó una barra de espera cuando se cargó el archivo en la aplicación y luego se observó una tabla con los datos cargados																													

Fuente: elaboración propia

Tabla 27. Caso de prueba relacionado con cargar el archivo Excel y validación de registros

Número de caso de prueba: 2		Prioridad de la prueba: Alta																													
Historia de usuario: cargar el archivo de Excel con los datos de los residentes.																															
Descripción del caso de prueba: verificar que en cada registro de la tabla con los datos del Excel cargados este el botón con la opción "Notificar" y un botón general "Notificar a todos".																															
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado																													
Pre-requisitos: 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo de Administración de residentes y SMS. 3. Tener al menos un usuario registrado en la aplicación.		Datos de prueba: Documento Excel con los siguientes campos: <table border="1"> <thead> <tr> <th>torre</th> <th>apartamento</th> <th>documento</th> <th>nombre</th> <th>celular</th> <th>meses_deuda</th> <th>total_deuda</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>102</td> <td>123456789</td> <td>David Rodriguez Fuentes</td> <td>3227623418</td> <td>1</td> <td>70000</td> </tr> <tr> <td>2</td> <td>101</td> <td>543234567</td> <td>Katherine Aguirre Tique</td> <td>3124922928</td> <td>2</td> <td>140000</td> </tr> <tr> <td>3</td> <td>404</td> <td>432345432</td> <td>Pepto Perez</td> <td>3125256290</td> <td>0</td> <td>0</td> </tr> </tbody> </table> (no importa el nombre del archivo Excel)		torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda	1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000	2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000	3	404	432345432	Pepto Perez	3125256290	0	0
torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda																									
1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000																									
2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000																									
3	404	432345432	Pepto Perez	3125256290	0	0																									
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)																												
Verificar si están los botones en cada registro.	Observar un botón en cada registro de la tabla con la opción "Notificar".	Se observó un botón en cada registro de la tabla con la opción "Notificar".																													
Verificar si el botón con la opción "Notificar a todos" está encima de la tabla.	Observar si el botón con la opción "Notificar a todos" está encima de la tabla.	Se observó que el botón con la opción "Notificar a todos" está encima de la tabla.																													

Fuente: elaboración propia

Tabla 28. Caso de prueba relacionado con guardar en base de datos y mostrar los datos en la aplicación.

Número de caso de prueba: 3		Prioridad de la prueba: Alta																													
Historia de usuario: guardar en base de datos y mostrar en la aplicación todos los datos que son recibidos del archivo Excel.																															
Descripción del caso de prueba: verificar en base de datos que los datos que fueron adjuntados del archivo Excel estén guardados en base de datos.																															
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado																													
Pre-requisitos: 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo de Administración de residentes y SMS. 3. Haber adjuntado anteriormente el archivo Excel con los datos de los residentes del conjunto.		Datos de prueba: Documento Excel con los siguientes campos: <table border="1"> <thead> <tr> <th>torre</th> <th>apartamento</th> <th>documento</th> <th>nombre</th> <th>celular</th> <th>meses_deuda</th> <th>total_deuda</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>102</td> <td>123456789</td> <td>David Rodriguez Fuentes</td> <td>3227623418</td> <td>1</td> <td>70000</td> </tr> <tr> <td>2</td> <td>101</td> <td>543234567</td> <td>Katherine Aguirre Tique</td> <td>3124922928</td> <td>2</td> <td>140000</td> </tr> <tr> <td>3</td> <td>404</td> <td>432345432</td> <td>Pepto Perez</td> <td>3125256290</td> <td>0</td> <td>0</td> </tr> </tbody> </table> (no importa el nombre del archivo Excel)		torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda	1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000	2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000	3	404	432345432	Pepto Perez	3125256290	0	0
torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda																									
1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000																									
2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000																									
3	404	432345432	Pepto Perez	3125256290	0	0																									
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)																												
Ingresar a la base de datos.	Validar que los datos estén allí almacenados.	Se observó que los datos de todos los residentes cargados han sido almacenados en base de datos.																													

Fuente: elaboración propia

Tabla 29. Caso de prueba relacionado con verificación de mensajaje de error al guardar los datos.

Número de caso de prueba: 4		Prioridad de la prueba: Alta																													
Historia de usuario: guardar en base de datos y mostrar en la aplicación todos los datos que son recibidos del archivo Excel.																															
Descripción del caso de prueba: verificar que se muestre un mensajaje de error si ocurre un fallo interno al momento de guardar los datos en base de datos.																															
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado																													
Pre-requisitos: 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo de administración de residentes y SMS. 3. No tener ningún usuario registrado en la aplicación.		Datos de prueba: Documento Excel con los siguientes campos: <table border="1" data-bbox="954 772 1474 829"> <thead> <tr> <th>torre</th> <th>apartamento</th> <th>documento</th> <th>edad</th> <th>telefonos</th> <th>meses</th> <th>total_deuda</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>102</td> <td>123456789</td> <td>David Rodriguez Fuentes</td> <td>31227623418</td> <td>1</td> <td>20000</td> </tr> <tr> <td>2</td> <td>101</td> <td>543234567</td> <td>Katherine Aguirre Tique</td> <td>3124922928</td> <td>2</td> <td>140000</td> </tr> <tr> <td>3</td> <td>404</td> <td>432345432</td> <td>Pepito Perez</td> <td>3125256290</td> <td>0</td> <td>0</td> </tr> </tbody> </table> (no importa el nombre del archivo Excel)		torre	apartamento	documento	edad	telefonos	meses	total_deuda	1	102	123456789	David Rodriguez Fuentes	31227623418	1	20000	2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000	3	404	432345432	Pepito Perez	3125256290	0	0
torre	apartamento	documento	edad	telefonos	meses	total_deuda																									
1	102	123456789	David Rodriguez Fuentes	31227623418	1	20000																									
2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000																									
3	404	432345432	Pepito Perez	3125256290	0	0																									
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)																												
Adjuntar el archivo Excel con datos inválidos en la aplicación para que falle el guardado en base de datos.	Mensaje de error.	Se observó un mensaje de error.																													

Fuente: elaboración propia

Tabla 30. Caso de prueba relacionado con enviar notificaciones a todos los residentes

Número de caso de prueba: 5		Prioridad de la prueba: Alta																													
Historia de usuario: enviar notificaciones por mensajes de texto a todos los residentes a la vez.																															
Descripción del caso de prueba: verificar que los residentes que estén registrados en la aplicación les haya llegado el mensaje de prueba al tiempo y en menos de 5 segundos.																															
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado																													
Pre-requisitos: 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo de administración de residentes y SMS. 3. Tener al menos un residente registrado en la aplicación.		Datos de prueba: Documento Excel con los siguientes campos: <table border="1"> <thead> <tr> <th>torre</th> <th>apartamento</th> <th>documento</th> <th>nombre</th> <th>celular</th> <th>meses_deuda</th> <th>total_deuda</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>102</td> <td>123456789</td> <td>David Rodriguez Fuentes</td> <td>3227623415</td> <td>1</td> <td>70000</td> </tr> <tr> <td>2</td> <td>101</td> <td>543234567</td> <td>Katherine Aguirre Tique</td> <td>3124922928</td> <td>2</td> <td>140000</td> </tr> <tr> <td>3</td> <td>404</td> <td>432345432</td> <td>Pepito Perez</td> <td>3125256290</td> <td>0</td> <td>0</td> </tr> </tbody> </table> (no importa el nombre del archivo Excel) Mensaje: Hola esto es una prueba.		torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda	1	102	123456789	David Rodriguez Fuentes	3227623415	1	70000	2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000	3	404	432345432	Pepito Perez	3125256290	0	0
torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda																									
1	102	123456789	David Rodriguez Fuentes	3227623415	1	70000																									
2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000																									
3	404	432345432	Pepito Perez	3125256290	0	0																									
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)																												
Dar clic en el botón "Enviar mensaje a todos"	Debe aparecer un mensaje donde permita ingresar el texto a enviar con un máximo de caracteres no supera de los 238.	Se observó el mensaje.																													
Ingresar el mensaje: Hola esto es una prueba	Debe permitir ingresar el mensaje "Hola esto es una prueba".	Se observó el mensaje esperado																													
Dar clic en el botón "enviar"	Debe llegar el mensaje a todos los residentes notificados de la siguiente manera: "Hacienda el Rosal te informa que, Hola esto es una prueba"	El mensaje llega con éxito a los celulares de los residentes.																													

Fuente: elaboración propia

Tabla 31. Caso de prueba relacionado con validar mensaje de error al enviar notificaciones

Número de caso de prueba: 6		Prioridad de la prueba: Alta																													
Historia de usuario: enviar notificaciones por mensajes de texto a todos los residentes a la vez.																															
Descripción del caso de prueba: validar que exista un mensaje de error cuando falle el envío de la notificación a todos los residentes.																															
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado																													
Pre-requisitos: 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo de administración de residentes y SMS. 3. Tener al menos un residente registrado en la aplicación.		Datos de prueba: Documento Excel con los siguientes campos: <table border="1" style="font-size: small;"> <thead> <tr> <th>torre</th> <th>apartamento</th> <th>documento</th> <th>nombre</th> <th>celular</th> <th>meses_deuda</th> <th>total_deuda</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>102</td> <td>123456789</td> <td>David Rodriguez Fuentes</td> <td>3227623418</td> <td>1</td> <td>70000</td> </tr> <tr> <td>2</td> <td>101</td> <td>543234567</td> <td>Katherine Aguirre Tique</td> <td>3124922928</td> <td>2</td> <td>140000</td> </tr> <tr> <td>3</td> <td>404</td> <td>432345432</td> <td>Pepto Perez</td> <td>3125256290</td> <td>0</td> <td>0</td> </tr> </tbody> </table> (no importa el nombre del archivo Excel)		torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda	1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000	2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000	3	404	432345432	Pepto Perez	3125256290	0	0
torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda																									
1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000																									
2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000																									
3	404	432345432	Pepto Perez	3125256290	0	0																									
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)																												
Dar clic en el botón "Enviar mensaje a todos"	Debe aparecer un mensaje donde permita ingresar el texto a enviar con un máximo de caracteres no superior a los 238.	Se observó el mensaje.																													
Ingresar un mensaje	Debe permitir ingresar el mensaje, el cual superó los 238 caracteres.	Se observó el mensaje.																													
Dar clic en el botón "enviar"	Se observar el mensaje de error.	Observar el mensaje de error.																													

Fuente: elaboración propia

Tabla 32. Caso de prueba relacionado con validar el envío de notificación a un residente

Número de caso de prueba: 7		Prioridad de la prueba: Alta																													
Historia de usuario: enviar notificaciones por mensajes de texto de manera personalizada.																															
Descripción del caso de prueba: validar que al enviar un mensaje de texto a un residente deseado le haya llegado con éxito.																															
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado																													
Pre-requisitos: 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo de administración de residentes y SMS. 3. Tener al menos un residente registrado en la aplicación.		Datos de prueba: Documento Excel con los siguientes campos: <table border="1" data-bbox="954 667 1474 730"> <thead> <tr> <th>torre</th> <th>apartamento</th> <th>documento</th> <th>nombre</th> <th>celular</th> <th>meses_deuda</th> <th>total_deuda</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>102</td> <td>123456789</td> <td>David Rodriguez Fuentes</td> <td>3227623418</td> <td>1</td> <td>70000</td> </tr> <tr> <td>2</td> <td>101</td> <td>543234567</td> <td>Katherine Aguirre Tique</td> <td>3124922928</td> <td>2</td> <td>140000</td> </tr> <tr> <td>3</td> <td>404</td> <td>432345432</td> <td>Pepto Perez</td> <td>3125256290</td> <td>0</td> <td>0</td> </tr> </tbody> </table> (no importa el nombre del archivo Excel)		torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda	1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000	2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000	3	404	432345432	Pepto Perez	3125256290	0	0
torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda																									
1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000																									
2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000																									
3	404	432345432	Pepto Perez	3125256290	0	0																									
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)																												
Dar clic en la opción "Notificar" de algún registro.	Debe aparecer un mensaje donde le pedirá que ingrese el texto a enviar y que el máximo de caracteres no supera de los 238.	Se observó el mensaje esperado.																													
Ingresar el mensaje "Hola esto es una prueba"	Debe llegar el mensaje al residente notificado de la siguiente manera: "Hacienda el Rosal te informa que, Hola esto es una prueba".	El mensaje llega con éxito.																													
Dar clic en el botón "enviar"	Debe llegar el mensaje al residente notificado de la siguiente manera: "Hacienda el Rosal te informa que, Hola esto es una prueba".	El mensaje llega con éxito.																													

Fuente: elaboración propia

Tabla 33. Caso de prueba relacionado con validar mensaje de error al enviar una notificación a un solo residente

Número de caso de prueba: 8		Prioridad de la prueba: Alta																													
Historia de usuario: enviar notificaciones por mensajes de texto de manera personalizada.																															
Descripción del caso de prueba: validar que exista un mensaje de error cuando falle el envío de la notificación a un residente.																															
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado																													
Pre-requisitos: <ol style="list-style-type: none"> 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo de administración de residentes y SMS. 3. Tener al menos un residente registrado en la aplicación. 4. Bajar el microservicio para que falle el envío de las notificaciones. 		Datos de prueba: Documento Excel con los siguientes campos: <table border="1"> <thead> <tr> <th>torre</th> <th>apartamento</th> <th>documento</th> <th>nombre</th> <th>celular</th> <th>meses_deuda</th> <th>total_deuda</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>102</td> <td>123456789</td> <td>David Rodriguez Fuentes</td> <td>3227623418</td> <td>1</td> <td>70000</td> </tr> <tr> <td>2</td> <td>101</td> <td>543234567</td> <td>Katherine Aguirre Tique</td> <td>3124922928</td> <td>2</td> <td>140000</td> </tr> <tr> <td>3</td> <td>404</td> <td>432345432</td> <td>Pepto Perez</td> <td>3125256290</td> <td>0</td> <td>0</td> </tr> </tbody> </table> (no importa el nombre del archivo Excel)		torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda	1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000	2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000	3	404	432345432	Pepto Perez	3125256290	0	0
torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda																									
1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000																									
2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000																									
3	404	432345432	Pepto Perez	3125256290	0	0																									
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)																												
Dar clic en la opción "Notificar" de algún registro.	Debe aparecer un mensaje donde le pedirá que ingrese el texto a enviar y que el máximo de caracteres no supera de los 238.	Se observó el mensaje esperado.																													
Ingresar el mensaje "Hola esto es una prueba"	Debe permitir ingresar el mensaje, que no supere los 238 caracteres y ver que el botón se active.	Se ingresó el mensaje con éxito y se activó el botón.																													
Dar clic en el botón "enviar".	Se debe observar el mensaje de error.	se observó el mensaje de error.																													

Fuente: elaboración propia

Tabla 34. Caso de prueba relacionado con agregar un nuevo residente

Número de caso de prueba: 9		Prioridad de la prueba: Alta																													
Historia de usuario: agregar un residente nuevo en la aplicación web.																															
Descripción del caso de prueba: registrar un residente exitosamente, validando campos con formatos correctos y vacíos.																															
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado																													
Pre-requisitos: 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo de administración de residentes y SMS. 3. Tener al menos un residente registrado en la aplicación.		Datos de prueba: Documento Excel con los siguientes campos: <table border="1" data-bbox="954 590 1474 653"> <thead> <tr> <th>torre</th> <th>apartamento</th> <th>documento</th> <th>nombre</th> <th>celular</th> <th>meses_deuda</th> <th>total_deuda</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>102</td> <td>123456789</td> <td>David Rodriguez Fuentes</td> <td>3227623418</td> <td>1</td> <td>70000</td> </tr> <tr> <td>2</td> <td>101</td> <td>543234567</td> <td>Katherine Aguirre Tique</td> <td>3124922928</td> <td>2</td> <td>140000</td> </tr> <tr> <td>3</td> <td>404</td> <td>432345432</td> <td>Pepto Perez</td> <td>3125256290</td> <td>0</td> <td>0</td> </tr> </tbody> </table> (no importa el nombre del archivo Excel)		torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda	1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000	2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000	3	404	432345432	Pepto Perez	3125256290	0	0
torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda																									
1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000																									
2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000																									
3	404	432345432	Pepto Perez	3125256290	0	0																									
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)																												
Dar clic en la opción registrar residente.	Observar un mensaje con los campos a llenar.	Se observó un mensaje.																													
Ingresar en los campos los datos a llenar.	Permite llenar los campos.	Se observó los campos ingresados.																													
Dar clic en el botón "guardar"	Se guardan los campos diligenciados.	Se observó los campos diligenciados exitosamente.																													

Fuente: elaboración propia

Tabla 35. Caso de prueba relacionado con agregar un residente nuevo y validación de campos

Número de caso de prueba: 10		Prioridad de la prueba: Alta	
Historia de usuario: agregar un residente nuevo en la aplicación web			
Descripción del caso de prueba: verificar que no se active el botón de guardar cuando se ingrese un campo o más campos no válidos.			
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado	
Pre-requisitos: 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo de administración de residentes y SMS. 3. Tener al menos un residente registrado en la aplicación.		Datos de prueba: 1. Campos inválidos: torre apartamento: 115-1021 celular: 472722142561	
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)
Dar clic en la opción registrar residente.	Observar un mensaje con los campos a llenar.	Se observó un mensaje.	
Ingresar campos inválidos.	Permite llenar los campos.	Se observó los campos ingresados.	
Dar clic en el botón "guardar"	Observa que no se activó el botón de guardar.	No se activa el botón guardar.	

Fuente: elaboración propia

Tabla 36. Caso de prueba relacionado con agregar un residente nuevo y validación de registros en base de datos

Número de caso de prueba: 11		Prioridad de la prueba: Alta	
Historia de usuario: agregar un residente nuevo en la aplicación web			
Descripción del caso de prueba: verificar que en base de datos se guarden los registros nuevos y que aparezca un mensaje de éxito cuando se guarde el nuevo residente.			
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado	
Pre-requisitos: 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo de Administración de residentes y SMS. 3. Tener al menos un residente registrado en la aplicación.		Datos de prueba: 2. Campos inválidos: Torre - apartamento: 10-404 Número de celular: 3125830084 Nombre completo: Sara González Número de documento: 695432910 Número de meses en deuda de administración: 0 Deuda total: 0	
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)
Ingresar a la base de datos.	Observar los datos del residente que se acaba de guardar.	Se observan los datos de los residentes en base de datos.	

Fuente: elaboración propia

Tabla 37. Caso de prueba relacionado con agregar un residente nuevo y validación de mensaje de error

Número de caso de prueba: 12		Prioridad de la prueba: Alta	
Historia de usuario: agregar un residente nuevo en la aplicación web.			
Descripción del caso de prueba: verificar que aparezca un mensaje de error cuando ocurra un fallo en el guardado de un residente en la aplicación.			
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado	
Pre-requisitos: 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo de administración de residentes y SMS. 3. Tener al menos un residente registrado en la aplicación. 4. Bajar el microservicio para que el guardado del residente falle.		Datos de prueba: 3. Campos inválidos: Torre - apartamento: 110-404 Número de celular: 3125830084 Nombre completo: Sara González Número de documento: 695432910 Número de meses en deuda de administración: 0 Deuda total: 0	
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)
Dar clic en la opción registrar residente.	Observar un mensaje con los campos a llenar.	Se observó un mensaje.	
Ingresar los campos.	Debe permitir llenar los campos y visualizar el botón "Guardar" activado.	Se observó los campos ingresados y el botón activo.	
Dar clic en el botón "guardar".	Observa un mensaje de error.	Se observó un mensaje de error.	

Fuente: elaboración propia

Tabla 38. Caso de prueba relacionado con editar datos y validación botón de guardado

Número de caso de prueba: 13		Prioridad de la prueba: Alta																													
Historia de usuario: editar datos de cada residente registrado en la aplicación.																															
Descripción del caso de prueba: verificar que se active el botón de guardado cada vez que se edite un campo de cada registro.																															
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado																													
Pre-requisitos: 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo de administración de residentes y SMS. 3. Tener al menos un residente registrado en la aplicación.		Datos de prueba: Documento Excel con los siguientes campos: <table border="1" data-bbox="954 667 1474 730"> <thead> <tr> <th>torre</th> <th>apartamento</th> <th>documento</th> <th>nombre</th> <th>celular</th> <th>meses_deuda</th> <th>total_deuda</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>102</td> <td>123456789</td> <td>David Rodriguez Fuentes</td> <td>3227623418</td> <td>1</td> <td>70000</td> </tr> <tr> <td>2</td> <td>101</td> <td>543234567</td> <td>Katherine Aguirre Tique</td> <td>3124922928</td> <td>2</td> <td>140000</td> </tr> <tr> <td>3</td> <td>404</td> <td>432345432</td> <td>Pepito Perez</td> <td>3125256290</td> <td>0</td> <td>0</td> </tr> </tbody> </table> (no importa el nombre del archivo Excel)		torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda	1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000	2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000	3	404	432345432	Pepito Perez	3125256290	0	0
torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda																									
1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000																									
2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000																									
3	404	432345432	Pepito Perez	3125256290	0	0																									
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)																												
Dar clic en algún campo y editarlo.	Se activa el botón de guardar.	Se observó el cambio del campo.																													

Fuente: elaboración propia

Tabla 39. Caso de prueba relacionado con editar datos y validación de no actualización de datos

Número de caso de prueba: 14		Prioridad de la prueba: Alta																													
Historia de usuario: editar datos de cada residente registrado en la aplicación.																															
Descripción del caso de prueba: verificar que no se active el botón de guardado si no se edita un campo de cada registro.																															
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado																													
Pre-requisitos: 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo de administración de residentes y SMS. 3. Tener al menos un residente registrado en la aplicación.		Datos de prueba: Documento Excel con los siguientes campos: <table border="1" data-bbox="954 1392 1474 1455"> <thead> <tr> <th>torre</th> <th>apartamento</th> <th>documento</th> <th>nombre</th> <th>celular</th> <th>meses_deuda</th> <th>total_deuda</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>102</td> <td>123456789</td> <td>David Rodriguez Fuentes</td> <td>3227623418</td> <td>1</td> <td>70000</td> </tr> <tr> <td>2</td> <td>101</td> <td>543234567</td> <td>Katherine Aguirre Tique</td> <td>3124922928</td> <td>2</td> <td>140000</td> </tr> <tr> <td>3</td> <td>404</td> <td>432345432</td> <td>Pepito Perez</td> <td>3125256290</td> <td>0</td> <td>0</td> </tr> </tbody> </table> (no importa el nombre del archivo Excel)		torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda	1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000	2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000	3	404	432345432	Pepito Perez	3125256290	0	0
torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda																									
1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000																									
2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000																									
3	404	432345432	Pepito Perez	3125256290	0	0																									
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)																												
Dar clic en algún campo sin editarlo.	Observar que no se activa el botón de guardar.	Se observó que el botón no se activó.																													

Fuente: elaboración propia.

Tabla 40. Caso de prueba relacionado con editar datos de los residentes y verificación de actualizaciones en base de datos

Número de caso de prueba: 15		Prioridad de la prueba: Alta																													
Historia de usuario: editar datos de cada residente registrado en la aplicación.																															
Descripción del caso de prueba: verificar que en base de datos se actualicen los registros de cada residente.																															
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado																													
Pre-requisitos: 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo de administración de residentes y SMS. 3. Tener al menos un residente registrado en la aplicación. 4. Haber actualizado los datos de un residente.		Datos de prueba: Documento Excel con los siguientes campos: <table border="1" data-bbox="954 695 1474 758"> <thead> <tr> <th>torre</th> <th>apartamento</th> <th>documento</th> <th>nombre</th> <th>celular</th> <th>meses_deuda</th> <th>total_deuda</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>102</td> <td>123456789</td> <td>David Rodriguez Fuentes</td> <td>3227623418</td> <td>1</td> <td>70000</td> </tr> <tr> <td>2</td> <td>101</td> <td>543234567</td> <td>Katherine Aguirre Tique</td> <td>3124922928</td> <td>2</td> <td>140000</td> </tr> <tr> <td>3</td> <td>404</td> <td>432345432</td> <td>Pepito Perez</td> <td>3125256290</td> <td>0</td> <td>0</td> </tr> </tbody> </table> (no importa el nombre del archivo Excel)		torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda	1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000	2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000	3	404	432345432	Pepito Perez	3125256290	0	0
torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda																									
1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000																									
2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000																									
3	404	432345432	Pepito Perez	3125256290	0	0																									
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)																												
Ingresar a la base de datos.	Se debe ver los datos del residente actualizados en las tablas.	Se observó los datos actualizados en base de datos.																													

Fuente: elaboración propia

Tabla 41. Caso de prueba relacionado con filtrar torre y apartamento

Número de caso de prueba: 16		Prioridad de la prueba: Alta																													
Historia de usuario: filtrar por torre y apartamento los residentes del conjunto.																															
Descripción del caso de prueba: validar que, al momento de filtrar por torre y apartamento, la búsqueda sea automática.																															
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado																													
Pre-requisitos: 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo de administración de residentes y SMS. 3. Tener al menos un residente registrado en la aplicación.		Datos de prueba: Documento Excel con los siguientes campos: <table border="1" data-bbox="954 1451 1474 1514"> <thead> <tr> <th>torre</th> <th>apartamento</th> <th>documento</th> <th>nombre</th> <th>celular</th> <th>meses_deuda</th> <th>total_deuda</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>102</td> <td>123456789</td> <td>David Rodriguez Fuentes</td> <td>3227623418</td> <td>1</td> <td>70000</td> </tr> <tr> <td>2</td> <td>101</td> <td>543234567</td> <td>Katherine Aguirre Tique</td> <td>3124922928</td> <td>2</td> <td>140000</td> </tr> <tr> <td>3</td> <td>404</td> <td>432345432</td> <td>Pepito Perez</td> <td>3125256290</td> <td>0</td> <td>0</td> </tr> </tbody> </table> (no importa el nombre del archivo Excel)		torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda	1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000	2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000	3	404	432345432	Pepito Perez	3125256290	0	0
torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda																									
1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000																									
2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000																									
3	404	432345432	Pepito Perez	3125256290	0	0																									
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)																												
Dar clic en el campo de búsqueda.	Permite la búsqueda automática.	Se observa búsqueda de torre y apartamento.																													
Ingresar torre y apartamento a buscar.	Se observa la búsqueda.	Se observa búsqueda exitosamente.																													

Fuente: elaboración propia

Tabla 42. Caso de prueba relacionado con filtrar residentes morosos y envío de SMS

Número de caso de prueba: 17		Prioridad de la prueba: Alta																													
Historia de usuario: filtrar residentes en estado mora por pago de administración.																															
Descripción del caso de prueba: validar que, en el módulo de recordar a morosos SMS solo aparezcan los residentes morosos por pago de administración.																															
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado																													
Pre-requisitos: 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo de recordar a morosos SMS. 3. Tener al menos un residente registrado en la aplicación.		Datos de prueba: Documento Excel con los siguientes campos: <table border="1" data-bbox="954 667 1474 730"> <thead> <tr> <th>torre</th> <th>apartamento</th> <th>documento</th> <th>nombre</th> <th>celular</th> <th>meses_deuda</th> <th>total_deuda</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>102</td> <td>123456789</td> <td>David Rodriguez Fuentes</td> <td>3227623418</td> <td>1</td> <td>70000</td> </tr> <tr> <td>2</td> <td>101</td> <td>543234567</td> <td>Katherine Aguirre Tique</td> <td>3124922928</td> <td>2</td> <td>140000</td> </tr> <tr> <td>3</td> <td>404</td> <td>432345432</td> <td>Pepto Perez</td> <td>3125256290</td> <td>0</td> <td>0</td> </tr> </tbody> </table> (no importa el nombre del archivo Excel)		torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda	1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000	2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000	3	404	432345432	Pepto Perez	3125256290	0	0
torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda																									
1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000																									
2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000																									
3	404	432345432	Pepto Perez	3125256290	0	0																									
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)																												
Dar clic en el módulo recordar a morosos SMS.	Se observa los residentes que se encuentran en mora.	Observó campos de los residentes morosos.																													

Fuente: elaboración propia

Tabla 43. Caso de prueba relacionado con filtrar residentes morosos con validación de botones de notificación

Número de caso de prueba: 18		Prioridad de la prueba: Alta																													
Historia de usuario: filtrar residentes en estado mora por pago de administración.																															
Descripción del caso de prueba: validar que, en el módulo de recordar a morosos SMS solo aparezcan los residentes morosos por pago de administración con un botón en cada registro “Notificar” y otro “Enviar mensaje a todos”.																															
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado																													
Pre-requisitos: 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo de recordar a morosos SMS. 3. Tener al menos un residente moroso registrado en la aplicación.		Datos de prueba: Documento Excel con los siguientes campos: <table border="1" data-bbox="954 1472 1474 1535"> <thead> <tr> <th>torre</th> <th>apartamento</th> <th>documento</th> <th>nombre</th> <th>celular</th> <th>meses_deuda</th> <th>total_deuda</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>102</td> <td>123456789</td> <td>David Rodriguez Fuentes</td> <td>3227623418</td> <td>1</td> <td>70000</td> </tr> <tr> <td>2</td> <td>101</td> <td>543234567</td> <td>Katherine Aguirre Tique</td> <td>3124922928</td> <td>2</td> <td>140000</td> </tr> <tr> <td>3</td> <td>404</td> <td>432345432</td> <td>Pepto Perez</td> <td>3125256290</td> <td>0</td> <td>0</td> </tr> </tbody> </table> (no importa el nombre del archivo Excel)		torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda	1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000	2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000	3	404	432345432	Pepto Perez	3125256290	0	0
torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda																									
1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000																									
2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000																									
3	404	432345432	Pepto Perez	3125256290	0	0																									
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)																												
Ingresar al módulo de Recordar a morosos SMS.	Debe aparecer una lista con todos los residentes morosos por pago de administración con sus respectivos botones.	se observa la información deseada.																													

Fuente: elaboración propia

Tabla 44. Caso de prueba relacionado con notificación por mensaje a todos los residentes morosos

Número de caso de prueba: 19		Prioridad de la prueba: Alta																													
Historia de usuario: notificar por mensajes de texto a todos los residentes en estado de mora a la vez.																															
Descripción del caso de prueba: verificar que al notificar por mensaje de texto a todos los residentes morosos a la vez les llegue el mensaje en menos de 5 segundos.																															
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado																													
Pre-requisitos: 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo de recordar a morosos SMS. 3. Tener al menos un residente moroso registrado en la aplicación.		Datos de prueba: Documento Excel con los siguientes campos: <table border="1" data-bbox="954 720 1472 783"> <thead> <tr> <th>torre</th> <th>apartamento</th> <th>documento</th> <th>nombre</th> <th>celular</th> <th>meses_deuda</th> <th>total_deuda</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>102</td> <td>123456789</td> <td>David Rodriguez Fuentes</td> <td>3227623418</td> <td>1</td> <td>70000</td> </tr> <tr> <td>2</td> <td>101</td> <td>543234567</td> <td>Katherine Aguirre Tique</td> <td>3124922928</td> <td>2</td> <td>140000</td> </tr> <tr> <td>3</td> <td>404</td> <td>432345432</td> <td>Pepito Perez</td> <td>3125256290</td> <td>0</td> <td>0</td> </tr> </tbody> </table> (no importa el nombre del archivo Excel)		torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda	1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000	2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000	3	404	432345432	Pepito Perez	3125256290	0	0
torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda																									
1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000																									
2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000																									
3	404	432345432	Pepito Perez	3125256290	0	0																									
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)																												
Dar clic en la opción "Enviar mensaje a todos".	Debe aparecer un mensaje donde permita ingresar el texto a enviar con un máximo de caracteres no superior a 238.	Se observa mensaje.																													
Dar clic en la opción enviar.	Debe aparecer un mensaje.	Se observa mensaje enviado con éxito.																													

Fuente: elaboración propia

Tabla 45. Caso de prueba relacionado con notificación por mensaje y validación de errores cuando falla en el envío de SMS

Número de caso de prueba: 20		Prioridad de la prueba: Alta																													
Historia de usuario: notificar por mensajes de texto a todos los residentes en estado de mora a la vez.																															
Descripción del caso de prueba: validar que aparezca un mensaje de error si falla el envío de la notificación a todos los residentes morosos a la vez.																															
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado																													
Pre-requisitos: 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo de recordar a morosos SMS. 3. Tener al menos un residente moroso registrado en la aplicación. 4. Bajar el microservicio para hacer que el envío del mensaje falle.		Datos de prueba: Documento Excel con los siguientes campos: <table border="1" data-bbox="954 724 1469 787"> <thead> <tr> <th>torre</th> <th>apartamento</th> <th>documento</th> <th>nombre</th> <th>celular</th> <th>meses_deuda</th> <th>total_deuda</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>102</td> <td>123456789</td> <td>David Rodriguez Fuentes</td> <td>3227623418</td> <td>1</td> <td>70000</td> </tr> <tr> <td>2</td> <td>101</td> <td>543234567</td> <td>Katherine Aguirre Tique</td> <td>3124922928</td> <td>2</td> <td>140000</td> </tr> <tr> <td>3</td> <td>404</td> <td>432345432</td> <td>Pepito Perez</td> <td>3125256290</td> <td>0</td> <td>0</td> </tr> </tbody> </table> (no importa el nombre del archivo Excel)		torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda	1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000	2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000	3	404	432345432	Pepito Perez	3125256290	0	0
torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda																									
1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000																									
2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000																									
3	404	432345432	Pepito Perez	3125256290	0	0																									
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)																												
Dar clic en la opción "Enviar mensaje a todos".	Debe aparecer un mensaje donde permita ingresar el texto a enviar con un máximo de caracteres no superior a 238.	Se observa mensaje.																													
Ingresar mensaje a los residentes morosos inválido.	Permite ingresar mensaje y se activa el botón de enviar.	Se observa mensaje ingresado y el botón activado.																													
Dar clic en la opción enviar.	Debe aparecer un mensaje de fallo.	Se observó mensaje fallido.																													

Fuente: elaboración propia

Tabla 46. Caso de prueba relacionado con notificación por mensaje de texto a solo un residente moroso

Número de caso de prueba: 21		Prioridad de la prueba: Alta																													
Historia de usuario: notificar por mensajes de texto a los residentes morosos de manera personalizada.																															
Descripción del caso de prueba: validar que el residente con estado de mora le llegue el mensaje enviado.																															
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado																													
Pre-requisitos: 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo de recordar a morosos SMS. 3. Tener al menos un residente moroso registrado en la aplicación.		Datos de prueba: Documento Excel con los siguientes campos: <table border="1"> <thead> <tr> <th>torre</th> <th>apartamento</th> <th>documento</th> <th>nombre</th> <th>celular</th> <th>meses_deuda</th> <th>total_deuda</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>102</td> <td>123456789</td> <td>David Rodriguez Fuentes</td> <td>3227623418</td> <td>1</td> <td>70000</td> </tr> <tr> <td>2</td> <td>101</td> <td>543234567</td> <td>Katherine Aguirre Tique</td> <td>3124922928</td> <td>2</td> <td>140000</td> </tr> <tr> <td>3</td> <td>404</td> <td>432345432</td> <td>Pepito Perez</td> <td>3125256290</td> <td>0</td> <td>0</td> </tr> </tbody> </table> (no importa el nombre del archivo Excel)		torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda	1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000	2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000	3	404	432345432	Pepito Perez	3125256290	0	0
torre	apartamento	documento	nombre	celular	meses_deuda	total_deuda																									
1	102	123456789	David Rodriguez Fuentes	3227623418	1	70000																									
2	101	543234567	Katherine Aguirre Tique	3124922928	2	140000																									
3	404	432345432	Pepito Perez	3125256290	0	0																									
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)																												
Dar clic en la opción "Notificar".	Debe aparecer un mensaje donde permita ingresar el texto a enviar con un máximo de caracteres no superior a 238.	Se observa mensaje.																													
Ingresar mensaje a los residentes morosos.	Permite ingresar mensaje.	Se observa mensaje ingresado.																													
Dar clic en la opción enviar.	Se envía mensaje.	Se observa mensaje enviado exitosamente.																													

Fuente: elaboración propia

Tabla 47. Caso de prueba relacionado con publicaciones de novedades

Número de caso de prueba: 22		Prioridad de la prueba: Alta	
Historia de usuario: realizar publicaciones de novedades y anuncios importantes en la aplicación web acerca de lo que ocurre dentro del conjunto.			
Descripción del caso de prueba: publicar un anuncio exitosamente en la sección "Publicaciones".			
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado	
Pre-requisitos: 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo de "Publicaciones"		Datos de prueba: "Las nuevas reglas de convivencia del conjunto residencial son las siguientes: ..."	
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)
Dar clic en el icono más.	Se observa mensaje de publicar anuncio.	Se observa mensaje de publicar.	
Ingresar publicación.	Se activa el botón publicar.	Se observa la publicación.	
Dar clic en el botón "Publicar".	Debe aparecer un mensaje de éxito la publicación en primer plano en el módulo de Publicaciones.	Se observa la publicación.	

Fuente: elaboración propia.

Tabla 48. Caso de prueba relacionado con publicaciones de novedades y validación de errores

Número de caso de prueba: 23		Prioridad de la prueba: Alta	
Historia de usuario: realizar publicaciones de novedades y anuncios importantes en la aplicación web acerca de lo que ocurre dentro del conjunto.			
Descripción del caso de prueba: validar mensaje de error en caso de fallar la publicación del anuncio.			
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado	
Pre-requisitos: 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo de "Publicaciones" 3. Se debe bajar el microservicio para que falle el envío de la publicación.		Datos de prueba: "Las nuevas reglas de convivencia del conjunto residencial son las siguientes: ..."	
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)
Dar clic en el icono más.	Debe aparecer una pantalla donde solicite el ingreso del texto.	Se observó la pantalla esperada.	
Ingresar el mensaje que se quiere publicar.	Se debe de habilitar el botón de "Publicar" y aparecer en la sección de Publicaciones de primeras.	Se activó el botón como se esperaba.	
Dar clic en el botón "Publicar".	Debe aparecer un mensaje de error.	Se observó el mensaje de error esperado.	

Fuente: elaboración propia

2. INCREMENTO 2

Desde la Tabla 26 hasta la Tabla 59 **Tabla 48**, se encuentran los casos de prueba del incremento 2.

Tabla 49. Caso de prueba relacionado con solicitudes de los residentes

Número de caso de prueba: 24		Prioridad de la prueba: Alta	
Historia de usuario: recibir solicitudes de los residentes.			
Descripción del caso de prueba: validar registros de solicitudes de residentes.			
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado	
Pre-requisitos: 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo de novedades de residentes.		Datos de prueba: administrador Usuario: 123456789 Contraseña: 123456789	
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)
Ver un registro de solicitud de algún residente con los datos deseados	Lista con un registro	Lista con un registro	

Fuente: elaboración propia

Tabla 50. Caso de prueba relacionado con validación de mensaje cuando no hay solicitudes

Número de caso de prueba: 25		Prioridad de la prueba: Alta	
Historia de usuario: recibir solicitudes de los residentes.			
Descripción del caso de prueba: validar mensaje cuando no haya ninguna solicitud de residentes en la aplicación.			
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado	
Pre-requisitos: 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo de novedades de residentes. 3. No tener ninguna solicitud de residentes.		Datos de prueba: administrador Usuario: 123456789 Contraseña: 123456789	
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)
Ingresar al módulo de novedades de residentes.	Visualizar mensaje de que aún no hay solicitudes de los residentes.	Se visualiza el mensaje deseado.	

Fuente: elaboración propia

Tabla 51. Caso de prueba relacionado con validar filtro de tipo de solicitud de los residentes

Número de caso de prueba: 26		Prioridad de la prueba: Alta	
Historia de usuario: recibir solicitudes de los residentes.			
Descripción del caso de prueba: validar filtro por tipo de solicitud.			
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado	
Pre-requisitos: 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo de novedades de residentes. 3. Tener al menos 3 solicitudes con tipo (queja, sugerencia, petición).		Datos de prueba: administrador Usuario: 123456789 Contraseña: 123456789	
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)
Ingresar al módulo de novedades de residentes.	Visualizar lista con los tres registros de tipos diferentes.	Se visualiza la lista deseada.	
Filtrar por queja	Visualizar solo la solicitud que es una queja.	Se visualiza la solicitud deseada.	
Filtrar por sugerencia	Visualizar solo la solicitud que es una sugerencia.	Se visualiza la solicitud deseada.	
Filtrar por petición	Visualizar solo la solicitud que es una petición.	Se visualiza la solicitud deseada.	

Fuente: elaboración propia

Tabla 52. Caso de prueba relacionado con autenticación de roles

Número de caso de prueba: 27		Prioridad de la prueba: Alta	
Historia de usuario: autenticar diferentes roles en la aplicación web.			
Descripción del caso de prueba: autenticación de los diferentes roles en la aplicación			
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado	
Pre-requisitos: Ingresar a la aplicación en el siguiente enlace https://conjunto-hacienda-rosal.com/#!/login		Datos de prueba: Rol administrador Usuario: 123456789 Contraseña: 123456789 Rol residente Usuario: 345678901 Contraseña: 345678901	
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)
Ingresar usuario	Campo lleno	Campo validado	
Ingresar contraseña	Campo lleno y botón de ingresar activado	Campo validado	
Clic en "Ingresar"	Ingresar a la aplicación con el rol deseado.	Se ven las funcionalidades de cada rol autenticado.	

Fuente: elaboración propia

Tabla 53. Caso de prueba relacionado con autenticación de roles y validación de mensajes de error

Número de caso de prueba: 28		Prioridad de la prueba: Alta	
Historia de usuario: autenticar diferentes roles en la aplicación web.			
Descripción del caso de prueba: validar campos, mensajes de error por fallo de autenticación y mensaje de usuario y/o contraseña incorrectos en login.			
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado	
Pre-requisitos: Ingresar a la aplicación en el siguiente enlace https://conjunto-hacienda-rosal.com/#/login		Datos de prueba: Rol administrador Usuario: 123456789 Contraseña: 123456789 Rol residente Usuario: 345678901 Contraseña: 345678901	
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)
Ingresar usuario	Campo lleno	Campo validado	
Ingresar contraseña	Campo lleno y botón de ingresar activado	Campo validado	
Clic en "Ingresar"	Ingresar a la aplicación con el rol deseado.	Se ven las funcionalidades de cada rol autenticado.	
Ingresar usuario incorrecto	Campo lleno	Campo validado	
Ingresar contraseña incorrecta	Campo lleno	Campo validado	
Clic en "Ingresar"	Ver mensaje de usuario y/o contraseña incorrectos.	Se observó el mensaje esperado.	

Fuente: elaboración propia

Tabla 54. Caso de prueba relacionado con publicaciones de la administración

Número de caso de prueba: 29		Prioridad de la prueba: Alta	
Historia de usuario: ver publicaciones que realiza el administrador.			
Descripción del caso de prueba: validar que se vean las publicaciones del administrador			
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado	
Pre-requisitos: 1. Ingresar a la aplicación como residente. 2. Ingresar al módulo de noticias. 3. Tener al menos una noticia publicada.		Datos de prueba: Rol administrador Usuario: 123456789 Contraseña: 123456789 Rol residente Usuario: 345678901 Contraseña: 345678901	
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)
Ingresar al módulo de noticias	Ver noticia publicada con su fecha de publicación	Se observó la noticia publicada.	

Fuente: elaboración propia

Tabla 55. Caso de prueba relacionado con validación de cero publicaciones

Número de caso de prueba: 30		Prioridad de la prueba: Alta	
Historia de usuario: ver publicaciones que realiza el administrador.			
Descripción del caso de prueba: validar mensaje cuando no haya noticias publicadas.			
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado	
Pre-requisitos: 1. Ingresar a la aplicación como residente. 2. Ingresar al módulo de noticias. 3. No tener noticias publicadas.		Datos de prueba: Rol administrador Usuario: 123456789 Contraseña: 123456789 Rol residente Usuario: 345678901 Contraseña: 345678901	
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)
Ingresar al módulo de noticias.	Ver mensaje indicando que aún no hay noticias publicadas por administrador.	Se observó mensaje correctamente.	

Fuente: elaboración propia

Tabla 56. Caso de prueba relacionado con enviar diferentes tipos de solicitudes a la administración

Número de caso de prueba: 31		Prioridad de la prueba: Alta	
Historia de usuario: enviar solicitud al administrador			
Descripción del caso de prueba: enviar diferentes tipos de solicitud al administrador.			
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado	
Pre-requisitos: 1. Ingresar a la aplicación como residente. 2. Ingresar al módulo de enviar solicitud.		Datos de prueba: Rol residente Usuario: 345678901 Contraseña: 345678901	
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)
Ingresar al módulo de enviar solicitud.	Ver pantalla de ingreso de solicitud con sus diferentes opciones de tipo (queja, sugerencia, petición)	Se observó pantalla correctamente.	
Ingresar solicitud en caja de texto.	Campo lleno	Se observa el campo lleno.	
Escoger tipo de solicitud	Se debe activar el botón de enviar.	Se activa el botón.	
Clic en enviar.	Se debe de enviar correctamente y visualizar el mensaje de envió con éxito.	Se observa el mensaje de envió con éxito.	

Fuente: elaboración propia

Tabla 57. Caso de prueba relacionado con validación de error al enviar solicitud a la administración

Número de caso de prueba: 32		Prioridad de la prueba: Alta	
Historia de usuario: enviar solicitud al administrador			
Descripción del caso de prueba: validar mensajes de error de envío de solicitud.			
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado	
Pre-requisitos: 1. Ingresar a la aplicación como residente. 2. Ingresar al módulo de enviar solicitud. 3. Bajar el microservicio para que falle el envío de la solicitud.		Datos de prueba: Rol residente Usuario: 345678901 Contraseña: 345678901	
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)
Ingresar al módulo de enviar solicitud.	Ver pantalla de ingreso de solicitud con sus diferentes opciones de tipo (queja, sugerencia, petición)	Se observó pantalla correctamente.	
Ingresar solicitud en caja de texto.	Campo lleno	Se observa el campo lleno.	
Escoger tipo de solicitud	Se debe activar el botón de enviar.	Se activa el botón.	
Clic en enviar.	Se debe visualizar el mensaje de error de envío de la solicitud.	Se observa el mensaje correctamente.	

Fuente: elaboración propia

Tabla 58. Caso de prueba relacionado con comentar las publicaciones de la administración

Número de caso de prueba: 33		Prioridad de la prueba: Alta	
Historia de usuario: comentar publicaciones que realiza el administrador.			
Descripción del caso de prueba: comentar correctamente una publicación que realice el administrador.			
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado	
Pre-requisitos: 1. Ingresar a la aplicación como residente. 2. Ingresar al módulo publicaciones 3. Tener al menos una publicación en la aplicación.		Datos de prueba: Rol residente Usuario: 345678901 Contraseña: 345678901	
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)
Ingresar al módulo de publicaciones.	Ver publicaciones realizadas por el administrador.	Se observó las publicaciones correctamente.	
Dar clic en comentar a alguna publicación	Ver pantalla de ingreso de comentario.	Se observa la pantalla deseada	
Enviar comentario	Ver mensaje de envío de comentario exitoso.	Se visualiza el mensaje correctamente	

Fuente: elaboración propia

Tabla 59. Caso de prueba relacionado con ver comentarios de las publicaciones

Número de caso de prueba: 34		Prioridad de la prueba: Alta	
Historia de usuario: ver comentarios que realizan los residentes sobre las publicaciones.			
Descripción del caso de prueba: ver comentarios que hacen los residentes de cada una de las publicaciones correctamente.			
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado	
Pre-requisitos: 1. Ingresar a la aplicación como administrador. 2. Ingresar al módulo publicaciones 3. Tener al menos una publicación en la aplicación. 4. Tener al menos un comentario en alguna publicación que se haya hecho anteriormente.		Datos de prueba: Rol administrador Usuario: 1234567890 Contraseña: 1234567890	
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)
Ingresar al módulo de publicaciones.	Ver publicaciones realizadas por el administrador.	Se observó las publicaciones correctamente.	
Ver el número de comentarios que tiene cada publicación	Ver un número de comentarios.	Se observa el mensaje con número de comentarios correctamente.	
Dar clic en el número de comentarios	Ver pantalla con una lista de comentarios que se hayan hecho a esa publicación.	Se visualiza la lista de comentarios correctamente.	

Fuente: elaboración propia

3. INCREMENTO 3

De la Tabla 60 a la Tabla 65 **Tabla 48**, se encuentran los casos de prueba del incremento 3.

Tabla 60. Caso de prueba relacionado con solicitudes enviadas a la administración

Número de caso de prueba: 35		Prioridad de la prueba: Alta	
Historia de usuario: ver solicitudes enviadas al administrador.			
Descripción del caso de prueba: validar que al enviar una solicitud al administrador se vea en el módulo de "Mis solicitudes"			
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado	
Pre-requisitos: 1. Ingresar a la aplicación como residente.		Datos de prueba: residente Usuario: 345678912 Contraseña: 345678912	
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)
Enviar una solicitud al administrador.	Ver que se envió correctamente.	Se envió correctamente.	
Ingresar al módulo "Mis solicitudes"	Ver la solicitud enviada sin respuesta	Se observó la solicitud enviada.	

Fuente: elaboración propia

Tabla 61. Caso de prueba relacionado con ver mensaje de cero solicitudes enviadas al administrador

Número de caso de prueba: 36		Prioridad de la prueba: Alta	
Historia de usuario: ver respuesta de las solicitudes enviadas al administrador.			
Descripción del caso de prueba: validar mensaje cuando no haya ninguna solicitud de residentes en la aplicación.			
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado	
Pre-requisitos: 1. Ingresar a la aplicación como residente. 2. Haber enviado una solicitud al administrador. 3. Haber dado respuesta a la solicitud por parte del administrador.		Datos de prueba: residente Usuario: 345678912 Contraseña: 345678912	
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)
Ingresar al módulo de "Mis solicitudes"	Ver la solicitud con respuesta y opción de ver.	Se visualiza la opción deseada.	
Dar clic en la opción de ver.	Deberá mostrarse una pantalla con la	Se observa la pantalla con	

respuesta del administrador.	del	respuesta correctamente.	
------------------------------	-----	--------------------------	--

Fuente: elaboración propia

Tabla 62. Caso de prueba relacionado con ver estados de pago en mora

Número de caso de prueba: 37		Prioridad de la prueba: Alta	
Historia de usuario: ver estado de pagos de los residentes.			
Descripción del caso de prueba: validar que la información de un residente en estado de mora se muestre correctamente.			
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado	
Pre-requisitos: 1. Ingresar a la aplicación como residente.		Datos de prueba: residente Usuario: 345678912 Contraseña: 345678912	
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)
Ingresar al módulo de "Estado de mis pagos"	Ver información de pago.	Se observa la información de número de meses en deuda y monto total correctamente.	

Fuente: elaboración propia

Tabla 63. Caso de prueba relacionado con ver estado de pago de los residentes que se encuentran al día

Número de caso de prueba: 38		Prioridad de la prueba: Alta	
Historia de usuario: ver estado de pagos.			
Descripción del caso de prueba: validar la información de un residente que se encuentra al día con sus pagos.			
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado	
Pre-requisitos: 1. Ingresar a la aplicación como residente.		Datos de prueba: residente Usuario: 4567891234 Contraseña: 4567891234	
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)
Ingresar al módulo de "Estado de mis pagos"	Ver información de pago y ver mensaje de que se encuentra al día.	Se observa la información de número de meses en deuda y monto total correctamente en ceros.	

Fuente: elaboración propia

Tabla 64. Caso de prueba relacionado con las respuestas a las solicitudes de los residentes

Número de caso de prueba: 39		Prioridad de la prueba: Alta	
Historia de usuario: responder a las solicitudes que envían los residentes.			
Descripción del caso de prueba: responder a una solicitud de un residente correctamente			
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado	
Pre-requisitos: 1. Ingresar como administrador 2. Tener al menos una solicitud enviada por un residente.		Datos de prueba: Rol administrador Usuario: 123456789 Contraseña: 123456789	
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)
Ingresar al módulo de “Novedades de residentes”	Ver una lista de las novedades de los residentes	Se observa la lista correctamente	
Se selecciona una solicitud en estado sin respuesta	Deberá mostrar una pantalla en donde se pueda ingresar la respuesta.	Se muestra la pantalla deseada	
Ingresar la respuesta y se da clic en responder	Muestra mensaje de envío de respuesta con éxito y actualiza la información de la tabla, pasa a estar a estado “Atendido”	Se muestran los mensajes deseados y se actualiza la información de la solicitud.	

Fuente: elaboración propia

Tabla 65. Caso de prueba relacionado con filtro de solicitudes de los residentes

Número de caso de prueba: 40		Prioridad de la prueba: Alta	
Historia de usuario: filtrar por estado (queja, sugerencia o petición) las solicitudes de los residentes.			
Descripción del caso de prueba: validar que se vean las publicaciones del administrador			
Tester: Katherine Aguirre		Estado de la prueba (aprobado/fallido): Aprobado	
Pre-requisitos: 1. Ingresar a la aplicación como administrador 2. Tener al menos 2 solicitudes con estado (Sin respuesta o Atendido)		Datos de prueba: Rol administrador Usuario: 123456789 Contraseña: 123456789	
Paso	Resultados esperados	Resultados obtenidos	Estado (aprobado/fallido)
Ingresar al módulo de “Novedades de residentes”	De debe mostrar la lista con al menos dos solicitudes con diferente estado	Se muestran las solicitudes perfectamente con diferentes estados	
Filtrar por estado “Sin respuesta”	Se debe mostrar solo la solicitud con el estado “Sin respuesta”	Se filtra correctamente	

Filtrar por estado "Atendido"	Se debe mostrar solo la solicitud con el estado "Atendido"	Se filtra correctamente	
----------------------------------	--	-------------------------	--

Fuente: elaboración propia