



ESPECIALIZACIÓN
INGENIERÍA DE SOFTWARE

**DESARROLLO DE UN PROTOTIPO DE RED
SOCIAL PARA FACILITAR EL ACCESO A
LA INFORMACIÓN Y GESTIÓN DE
RECURSOS EN PROPIEDAD HORIZONTAL
RESIDENCIAL**

Presentada por:
Juan Carlos Linares Garzon
Jhon Edison Nuñez Rodriguez

Índice general

I	Introduccion	8
II	Contexto	11
1.	Descripción del proyecto	12
1.1.	Estudio del Problema de Investigación	12
1.1.1.	Planteamiento del problema	12
1.1.2.	Formulación del problema	13
1.1.3.	Sistematización del problema	13
1.2.	Objetivos	13
1.2.1.	Objetivo General	13
1.2.2.	Objetivos Especificos	13
1.3.	Justificación Práctica	14
1.4.	Hipótesis	14
1.5.	Marco Metodologico	14
1.5.1.	SCRUM	14
1.5.1.1.	Roles en SCRUM	15
1.5.1.2.	Artefactos en SCRUM	16
1.5.2.	Framework web	16
1.5.3.	HTML	17
1.5.4.	Node.js	17
1.5.5.	JavaScript	18
1.5.6.	JQuery	19
1.5.7.	WebSockets	19
1.6.	Alcances, Limitaciones y Resultados	21
1.6.1.	Alcances	21
1.6.2.	Limitaciones	21
1.6.3.	Resultados Esperados	22
2.	Descripción de la Organización	23
2.1.	Conjunto Residencial Torres de Hayuelos	23
2.1.1.	Misión	23
2.1.2.	Visión	23
2.1.3.	Objetivos	23

III	Desarrollo	24
3.	Investigación	25
3.1.	Recolección de información	25
3.1.1.	Recogiendo información del proyecto	25
3.1.1.1.	Entrevista	25
3.1.1.2.	Encuesta	26
3.1.2.	Tabulación y ordenamiento de la información	26
3.2.	Análisis de la información	27
3.3.	Selección de módulos a implementar	27
3.3.1.	Modulo de Reservas	28
3.3.2.	Modulo de Chat	28
3.3.3.	Modulo de Proveedores	28
3.3.4.	Modulo de Pagos	28
3.3.5.	Modulo de Publicaciones	28
4.	Arquitectura Empresarial	29
4.1.	Diagramas de Arquitectura Capa de Negocio	29
4.1.1.	Punto de Vista de Organización	29
4.1.1.1.	Metamodelo	29
4.1.1.2.	Modelo	30
4.1.2.	Punto de Vista de Cooperación de Actor	30
4.1.2.1.	Metamodelo	30
4.1.2.2.	Modelo	31
4.1.3.	Punto de Vista de Funcion de Negocio	32
4.1.3.1.	Metamodelo	32
4.1.3.2.	Modelo	32
4.1.4.	Punto de Vista de Proceso de Negocio	33
4.1.4.1.	Metamodelo	33
4.1.4.2.	Modelo	34
4.1.5.	Punto de Vista Proceso de Cooperacion de Negocio	34
4.1.5.1.	Metamodelo	34
4.1.5.2.	Modelo	35
4.1.6.	Punto de Vista de Producto	35
4.1.6.1.	Metamodelo	35
4.1.6.2.	Modelo	36
4.2.	Diagrama de Arquitectura Capa de Aplicacion	37
4.2.1.	Punto de Vista de Comportamiento de Aplicación	37
4.2.1.1.	Metamodelo	37
4.2.1.2.	Modelo	37
4.2.2.	Punto de Vista de Cooperación de Aplicación	38
4.2.2.1.	Metamodelo	38
4.2.2.2.	Modelo	39
4.2.3.	Punto de Vista de Estructura de Aplicación	40
4.2.3.1.	Metamodelo	40
4.2.3.2.	Modelo	40

4.2.4.	Punto de Vista de Uso de Aplicación	41
4.2.4.1.	Metamodelo	41
4.2.4.2.	Modelo	41
4.3.	Diagrama de Arquitectura Capa de Infraestructura	42
4.3.1.	Infraestructura	42
4.3.1.1.	Metamodelo	42
4.3.1.2.	Modelo	43
4.3.2.	Uso Infraestructura	44
4.3.2.1.	Metamodelo	44
4.3.2.2.	Modelo	45
4.3.3.	Organizacion e Implementacion	45
4.3.3.1.	Metamodelo	45
4.3.3.2.	Modelo	46
4.4.	Diagramas de Arquitectura Capa Motivacional	47
4.4.1.	Implicados	47
4.4.1.1.	Metamodelo	47
4.4.1.2.	Modelo	47
4.4.2.	Realizacion de Objetivos	48
4.4.2.1.	Metamodelo	48
4.4.2.2.	Modelo	49
4.4.3.	Motivaciòn	49
4.4.3.1.	Metamodelo	49
4.4.3.2.	Modelo	50
4.5.	Diagrama de Arquitectura Capa de Implementaciòn y Migraciòn	51
4.5.1.	Implementaciòn y Migraciòn	51
4.5.1.1.	Metamodelo	51
4.5.1.2.	Modelo	52
5.	Metodologia	54
5.1.	Roles	54
5.2.	Sprint planing	54
5.3.	Historias de usuario	55
6.	Modelo de Datos	63
6.1.	Tecnología	63
7.	Prototipo red Social	65
7.1.	Herramientas y tecnologías	65
7.1.1.	Putty	65
7.1.2.	Brackets	65
7.1.3.	Node.js	66
7.1.4.	Express.js	67
7.1.5.	Bookshelf.js	67
7.1.6.	Bootstrap	67
7.2.	Módulos	68
7.2.1.	Registro	68

7.2.2. Modulo de Reservas	68
7.2.3. Modulo de Chat	69
7.2.4. Modulo de Proveedores	70
7.2.5. Modulo de Pagos	71
7.2.6. Modulo de Publicaciones	71
IV Cierre	73
8. Resultados y Discusión	74
9. Trabajo futuro	75
10. Conclusiones	76
Anexos	77
A. Anexo I: Encuesta	78
B. Anexo II: Tabulación de la Información	80
Referencias	83

Índice de figuras

1.1. Estructura WebSockets	20
4.1. Metamodelo Punto de Vista de Organización	29
4.2. Modelo Punto de Vista de Organización	30
4.3. Metamodelo Punto de Vista de Cooperación de Actor	30
4.4. Modelo Punto de Vista de Cooperación de Actor	31
4.5. Metamodelo Punto de Vista de Funcion de Negocio	32
4.6. Modelo Punto de Vista de Funcion de Negocio	32
4.7. Metamodelo Punto de Vista de Proceso de Negocio	33
4.8. Modelo Punto de Vista de Proceso de Negocio	34
4.9. Metamodelo Punto de Vista Proceso de Cooperacion de Negocio	34
4.10. Modelo Punto de Vista Proceso de Cooperacion de Negocio	35
4.11. Metamodelo Punto de Vista de Producto	36
4.12. Modelo Punto de Vista de Producto	36
4.13. Metamodelo Punto de Vista de Comportamiento de Aplicación	37
4.14. Modelo Punto de Vista de Comportamiento de Aplicación	38
4.15. Metamodelo Punto de Vista de Cooperación de Aplicación	39
4.16. Modelo Punto de Vista de Cooperación de Aplicación	39
4.17. Metamodelo Punto de Vista de Estructura de Aplicación	40
4.18. Modelo Punto de Vista de Estructura de Aplicación	40
4.19. Metamodelo Punto de Vista de Uso de Aplicación	41
4.20. Modelo Punto de Vista de Uso de Aplicación	42
4.21. Metamodelo Punto de Vista Infraestructura	43
4.22. Modelo Punto de Vista Infraestructura	44
4.23. Metamodelo Punto de Vista Uso Infraestructura	44
4.24. Modelo Punto de Vista Uso Infraestructura	45
4.25. Metamodelo Punto de Vista Implementacion y Organizacion	46
4.26. Modelo Punto de Vista Implementacion y Organizacion	46
4.27. Metamodelo Punto de Vista Implicados	47
4.28. Modelo Punto de Vista Implicados	48
4.29. Metamodelo Punto de Vista Realizacion de Objetivos	48
4.30. Modelo Punto de Vista Realizacion de Objetivos	49
4.31. Metamodelo Punto de Vista Motivacion	50
4.32. Modelo Punto de Vista Motivaciòn	51
4.33. Metamodelo Punto de Vista Implementaciòn y Migraciòn	52

4.34. Modelo Punto de Vista Implementaciòn y Migraciòn	53
6.1. Modelo de datos	64
7.1. Putty con conexiòn SSH	66
7.2. Editor de texto Brackets	67
7.3. Pantalla de registro	68
7.4. Pantalla de reservas	69
7.5. Pantalla Chat	69
7.6. Pantalla proveedores	70
7.7. Pantalla administraciòn de proveedores	70
7.8. Pantalla de pagos	71
7.9. Pantalla de publicaciones	72

Índice de cuadros

5.1. Sprint Planing	55
5.2. Historia 1	55
5.3. Historia 2	56
5.4. Historia 3	56
5.5. Historia 4	57
5.6. Historia 5	57
5.7. Historia 6	58
5.8. Historia 7	58
5.9. Historia 8	59
5.10. Historia 9	59
5.11. Historia 10	60
5.12. Historia 11	60
5.13. Historia 12	61
5.14. Historia 13	61
5.15. Historia 14	62

Parte I

Introduccion

Gracias al avance de las tecnologías de la comunicación y el incremento en el uso de internet se permite a los usuarios a través de aplicaciones como las redes sociales estar conectados con personas desde cualquier parte del mundo así como realizar trámites o consultas por medio de páginas web de entidades tanto públicas como privadas que facilitan y optimizan tiempo y dinero a los usuarios.

Se puede afirmar que en una propiedad horizontal residencial es importante mantener una óptima convivencia, es por esto que la administración como ente central debe aprovechar los diferentes medios de comunicación para mantener a propietarios y arrendatarios informados de los procesos que se manejan. La creación de una red social residencial surge de la necesidad de mantener a una comunidad informada sobre temas comunes relacionadas a un conjunto residencial con por ejemplo pagos realizados y pagos pendientes, solicitud de reserva de zonas sociales, disponibilidad de zonas sociales, solicitud de estacionamientos, circulares informativas sobre asambleas, normas de convivencia e información que puede ser importante compartir entre vecinos sobre temas de interés para los integrantes del conjunto desarrollando comunidades más fuertes y seguras a futuro. Así mismo para el administrador facilita procesos y tareas sin necesidad de comunicación frente a frente.

La seguridad para el ingreso a una red social debe ser un tema central para los usuarios, ya que esto les permite tener la confiabilidad de que la información no puede ser extraída por terceros, garantizando un cifrado en el almacenamiento de credenciales de acceso y de acuerdo con la normatividad sobre habeas data.

Algunos de los beneficios que aporta la creación del prototipo de red social para propiedad horizontal residencial son los siguientes:

Al propietario o arrendatario:

- Facilidad de acceso a información de eventos, asambleas, pagos y noticias.
- Solicitud de préstamo de salón de eventos, zonas comunes y parqueaderos.
- Datos de contacto de proveedores asociados.

Al administrador:

- Registrar información de eventos, asambleas, pagos y noticias, con el fin de facilitar el acceso a la información, sin necesidad de que tengan que acudir a la oficina de administración.
- Generar órdenes de reservas de salón de eventos, zonas comunes y parqueaderos.

- Mantener información actualizada de proveedores asociados e informar mediante publicaciones los temas tratados en asambleas y control de la contabilidad en cuanto a gastos e inversiones realizadas.

Parte II
Contexto

Capítulo 1

Descripción del proyecto

1.1. Estudio del Problema de Investigación

1.1.1. Planteamiento del problema

En la actualidad, en las ciudades principales del país la construcción e inversión en propiedad horizontal va en aumento, debido a que las personas tienden a adquirir vivienda en conjuntos residenciales por diferentes ventajas como la seguridad y las zonas sociales o de deporte. Los servicios mencionados anteriormente se financian a través de la cuota de administración que debe cancelar cada propietario o arrendatario. Las decisiones de cómo se va a invertir el dinero se debe tomar entre todos los propietarios y aunque es muy difícil poner de acuerdo a toda la comunidad, por medio del administrador se debe citar a sus residentes a asambleas para que a través de votaciones se distribuyan los ingresos para los gastos.

Estas asambleas que en la mayoría de casos son extensas por la misma indecisión de los propietarios termina aburriendo a las personas y generando falta de interés en las decisiones allí tomadas, es por esto que la inversión del dinero no es la correcta y se da paso a posibles gastos no justificados por la administración, los cuales no son transparentes para los propietarios.

De igual manera al momento de solicitar por parte de un propietario o arrendatario un servicio (reserva de salón social de eventos, alquiler de estacionamientos, pagos de administración, solicitudes de acceso a escenarios deportivos), la mayoría de estos trámites solo se pueden hacer con el administrador, lo cual genera incomodidad en los propietarios ya que en algunas ocasiones el administrador no se encuentra disponible o simplemente solo está en horarios laborales.

De acuerdo a lo planteado anteriormente, los procesos y solicitud de servicios por parte de la administración se vuelven dispendiosos y demorados para los residentes generando insatisfacción entre ellos, teniendo como resultado que

algunas veces las decisiones de inversión en las asambleas sean cada vez más indiferentes para los usuarios y que los ingresos de la administración no se destinen a cosas realmente importantes, sin tener en cuenta que en un futuro pueda ocurrir un accidente o alguna eventualidad por falta de inversión en un aspecto específico.

1.1.2. Formulación del problema

De acuerdo a los problemas mencionados anteriormente se plantea resolver la siguiente pregunta:

¿Qué herramienta de comunicación puede facilitar el acceso a la información y gestión de recursos en propiedad horizontal residencial, garantizando seguridad y transparencia de la información para los residentes y la administración?

1.1.3. Sistematización del problema

Las siguientes incógnitas son claves para generar un sistema centralizado que permita a los usuarios interactuar entre sí, permita participar en las decisiones, solicitar servicios y conocer la información referente y segura a las cuentas de la administración sin necesidad de acudir a la administración.

- ¿Cómo se puede facilitar el acceso a la información administrativa y recursos a través de un prototipo de red social en propiedad horizontal residencial?
- ¿De qué forma se puede garantizar la disponibilidad, confidencialidad e integridad de la información que surge en el proceso de ingreso a la red social?
- ¿Qué mecanismos de integración y parametrización debo incluir en mi Arquitectura de Software para garantizar la compatibilidad de la solución al desarrollar la red social residencial?

1.2. Objetivos

1.2.1. Objetivo General

Desarrollar un prototipo de red social para facilitar el acceso a la información y gestión de recursos en propiedad horizontal residencial.

1.2.2. Objetivos Específicos

- Realizar un diagnóstico a los procesos, servicios y solicitudes de los usuarios de una propiedad horizontal residencial por medio de clasificación de roles y técnicas de levantamiento e información para definir los módulos a implementar.

- Diseñar los artefactos que componen la arquitectura del sistema con el propósito de crear un prototipo modular con disponibilidad continua de la información.
- Validar los componentes del prototipo de red social por medio de un ambiente simulado con el fin de identificar nuevos requerimientos a implementar.

1.3. Justificación Práctica

El presente proyecto tiene como fin centralizar la información y gestión de recursos administrativos a través de un prototipo de red social que permita la interacción entre usuarios, así mismo potenciar la gestión administrativa, la seguridad, la inversión del dinero, transparencia en las decisiones y aumentar la calidad de la convivencia en propiedad horizontal residencial.

Así mismo, este prototipo se basara en un sistema modular que esté disponible para las características más generales de los conjuntos de propiedad horizontal residencial como lo son el acceso a información relevante del conjunto, interacción con otros residentes, interacción con la administración, publicación de información y reservas de zonas comunes del conjunto. Este sistema modular permitirá que los conjuntos seleccionen los módulos que desean o que pueden implementar sin causar fallos en la funcionalidad principal. Como trabajo futuro se podrá optimizar el prototipo de manera que sea genérico y se pueda implementar no solo en propiedad horizontal sino también en cualquier tipo de comunidad que cuente con una estructura similar.

1.4. Hipótesis

El desarrollo de un prototipo de red social permitirá integrar a los diferentes participantes del proceso a través de una plataforma accesible desde la web, mejorando el acceso a la información, garantizando la transparencia de los procesos y otorgando calidad en el servicio prestado a la comunidad.

1.5. Marco Metodologico

El presente marco metodológico está compuesto inicialmente por una breve descripción de la metodología SCRUM

1.5.1. SCRUM

SCRUM es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto

1.5.1.1. Roles en SCRUM

A modo diferenciador de la mayoría de metodologías, en SCRUM no existe un gerente del proyecto. Las responsabilidades de este se encuentran divididas en los tres roles que conforman SCRUM

Product Owner

Quien gestiona el producto y la inversión. Sus responsabilidades son(Pete Deemer2009, Pete Deemer2009):

- Obtención de una visión compartida.
- Recolección de requerimientos.
- Administración y priorización del Product Backlog.
- Aceptación de software al final de cada iteración.
- Administración del plan de release.
- Maximización del retorno de inversión del proyecto.

SCRUM Master

Quien gestiona el proceso. Sus responsabilidades son(Pete Deemer2009, Pete Deemer2009):

- Aseguramiento de un entorno de trabajo para el equipo, protegido de interferencias y directivas.
- Remoción de impedimentos.
- Fomento del uso y respeto al proceso.
- Extensión del uso de SCRUM a lo largo de la organización.

Equipo

Quien se gestiona así mismo. Sus responsabilidades son(Pete Deemer2009, Pete Deemer2009):

- Estimación del tamaño de los ítems del Product Backlog.(Narkwilai et al.2015, Narkwilai et al.2015)
- Compromiso de entregar incrementos de software con calidad de producción.
- Seguimiento de su propio avance.
- Auto-organización, aunque con la responsabilidad ante el
- Product Owner de entregar el producto.

1.5.1.2. Artefactos en SCRUM

Los artefactos o herramientas esenciales de SCRUM están conformadas principalmente por el Product Backlog, Sprint Backlog, SCRUM Taskboard y el Burndown chart.

Product Backlog

También conocido como la lista de objetivos/requisitos, el Product Backlog expone la visión e intereses del cliente en cuanto a los objetivos y entrega del producto o desarrollo del proyecto.

Sprint Backlog

Consta de la lista de tareas que el equipo elabora en la reunión de planificación del Sprint con el objetivo de acordar las labores para cumplir los objetivos de la iteración. Esta lista permite ver las tareas donde el equipo está teniendo problemas y no avanza, con lo que le permite tomar decisiones al respecto.

SCRUM Taskboard

También conocido como tablero de tareas, el SCRUM Taskboard permite gestionar la lista de objetivos a completar en la iteración (Product Backlog items). Para cada objetivo, se asignan al lado derecho las tareas necesarias para completarlo y su estado actual (pendiente de iniciar, en progreso, hecha).

Burndown chart

Consta de un gráfico el cual refleja el avance que ha tenido el equipo de trabajo, la velocidad a la que está completando objetivos/requisitos y define finalmente si se podrá completar el trabajo en el tiempo estimado.

1.5.2. Framework web

Se define un framework como un conjunto de librerías y componentes junto con una documentación y metodología de uso, que permite diseñar, construir e implementar aplicaciones con mayor calidad y agilidad de programación. Se puntualiza que un framework provee de:

- Librerías: Un conjunto de subprogramas utilizados para el desarrollo de aplicaciones, estos contienen código y datos que proporcionan servicios a programas independientes.
- Componentes: Un componente de software es un elemento de un sistema que ofrece un servicio predefinido, y es capaz de comunicarse con otros

componentes, de una manera más simple; un componente es un objeto escrito de acuerdo a unas especificaciones.

- Documentación: Un framework debe ofrecer una cantidad de documentos, foros y tutoriales para el aprendizaje y elaboración de software sobre el mismo.
- Metodología de uso: Cada framework tienen una metodología de uso; es decir, una arquitectura a la cual apegarse para el desarrollo de aplicaciones. Esta arquitectura muestra la separación entre la lógica del negocio y la de la capa de presentación.

Un framework web cumple con las características ya mencionadas y la diferencia sustancial con los frameworks para aplicaciones desktop se centra simplemente en el ambiente para al cual están dirigidos.(Cordero2013, Cordero2013)

1.5.3. HTML

Significa Lenguaje de Marcado para Hipertextos (HyperText Markup Language) es el elemento de construcción más básico de una página web y se usa para crear y representar visualmente una página web. Determina el contenido de la página web, pero no su funcionalidad.

HTML le añade "marcado." a un texto estándar en español. "Hiper Texto" se refiere a enlaces que conectan una página Web con otra, haciendo de la Telaraña Mundial (World Wide Web) lo que es hoy. Al crear y subir páginas Web a Internet, usted se hace un participante activo de esta Telaraña Mundial una vez su sitio está en línea. HTML soporta imágenes y también otro tipo de elementos multimedia. Con la ayuda de HTML todos pueden hacer sitios web estáticos y dinámicos. HTML es el lenguaje que describe la estructura y el contenido semántico de un documento web. El contenido dentro de una página web es etiquetado con elementos HTML como ``, `<title>`, `<p>`, `<div>`, y así sucesivamente. Estos elementos conforman los bloques de construcción de un sitio web.(Network2016a, Network2016a)

1.5.4. Node.js

Es un entorno JavaScript de lado de servidor que utiliza un modelo asíncrono y dirigido por eventos

Node.js usa el motor de JavaScript V8 de Google: una VM tremendamente rápida y de gran calidad escrita por gente como Lars Bak, uno de los mejores ingenieros del mundo especializados en VMs. No olvidemos que V8 es actualizado constantemente y es uno de los intérpretes más rápidos que puedan existir en la actualidad para cualquier lenguaje dinámico. Además las capacidades de Node para I/O (Entrada/Salida) son realmente ligeras y potentes, dando al desarrollador la posibilidad de utilizar a tope la I/O del sistema.

Node.js soporta protocolos TCP, DNS y HTTP.

Node.js está diseñado para construir aplicaciones de red escalables, es similar en diseño y está influenciado por sistemas como el de Ruby o Python, Node toma el modelo de eventos un poco más lejos, se presenta un ciclo de eventos como una construcción en tiempo de ejecución en lugar de como una biblioteca. Node se caracteriza por simplemente definir un bucle de eventos después de ejecutar el script de entrada, y sale del bucle de eventos cuando no hay más llamadas. Este comportamiento es como JavaScript en el navegador donde todos los eventos son invisibles para el usuario.

Node está diseñado sin hilos, no significa que no se pueda tomar ventaja de múltiples núcleos en su entorno. Los procesos secundarios pueden ser generados mediante el uso de subprocesos como ramas e hijos de la API, y están diseñados para ser fáciles de comunicar.(rmunoz2011, rmunoz2011)

1.5.5. JavaScript

Es un lenguaje de scripting multiplataforma y orientado a objetos. Es un lenguaje pequeño y liviano. Dentro de un ambiente de host, JavaScript puede conectarse a los objetos de su ambiente y proporcionar control programático sobre ellos.

JavaScript contiene una librería estándar de objetos, tales como Array, Date, y Math, y un conjunto central de elementos del lenguaje, tales como operadores, estructuras de control, y sentencias. El núcleo de JavaScript puede extenderse para varios propósitos, complementándolo con objetos adicionales, por ejemplo:

Client-side JavaScript extiende el núcleo del lenguaje proporcionando objetos para controlar un navegador y su modelo de objetos (o DOM, por las iniciales de Document Object Model). Por ejemplo, las extensiones del lado del cliente permiten que una aplicación coloque elementos en un formulario HTML y responda a eventos del usuario, tales como clicks del ratón, ingreso de datos al formulario y navegación de páginas.

Server-side JavaScript extiende el núcleo del lenguaje proporcionando objetos relevantes a la ejecución de JavaScript en un servidor. Por ejemplo, las extensiones del lado del servidor permiten que una aplicación se comunique con una base de datos, proporcionar continuidad de la información de una invocación de la aplicación a otra, o efectuar manipulación de archivos en un servidor.(Network2016b, Network2016b)

1.5.6. JQuery

JQuery es una biblioteca de JavaScript, creada inicialmente por John Resig, que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web.

Fue presentada el 14 de enero de 2006 en el BarCamp NYC. jQuery es software libre y de código abierto, posee un doble licenciamiento bajo la Licencia MIT y la Licencia Pública General de GNU v2, permitiendo su uso en proyectos libres y privativos. jQuery, al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerirían de mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio.

Las empresas Microsoft y Nokia anunciaron que incluirán la biblioteca en sus plataformas. Microsoft la añadirá en su IDE Visual Studio y la usará junto con los frameworks ASP.NET AJAX y ASP.NET MVC, mientras que Nokia los integrará con su plataforma Web Run-Time.

Sus características más importantes son:

- Selección de elementos DOM.
- Interactividad y modificaciones del árbol DOM, incluyendo soporte para CSS 1-3 y un plugin básico de XPath.
- Eventos, manipulación de la hoja de estilos CSS.
- Efectos y animaciones. Animaciones personalizadas.
- AJAX. Soporta extensiones.
- Utilidades varias como obtener información del navegador, operar con objetos y vectores, funciones para rutinas comunes, etc.

(Q-Success2016, Q-Success2016)

1.5.7. WebSockets

WebSockets es una tecnología avanzada que hace posible abrir una sesión de comunicación interactiva entre el navegador del usuario y un servidor. Con esta API, puede enviar mensajes a un servidor y recibir respuestas controladas por eventos sin tener que consultar al servidor para una respuesta.

HTML5 como estándar incorpora Websockets entre sus nuevas características, para comprender esta tecnología en primer lugar vamos a exponer que intenta

resolver Websockets y como se implementa la comunicación entre cliente y servidor hoy en día.

Explicaremos el concepto y las diferencias entre técnicas de polling y Websockets pero no las desarrollaremos específicamente de manera teórica o práctica.

Comunicación Cliente - Servidor

La problemática que intenta resolver websockets, generalizando, es comunicar los procesos del servidor con los procesos del cliente (navegador) en una plataforma web, mejorando las técnicas actuales.

Para los que no conocen la plataforma de desarrollo web básica esta se compone generalmente de un lado cliente y un lado servidor, que interactúan mediante peticiones.

Websockets permite comunicar el cliente y el servidor a través de un canal Full Duplex bidireccional y sin tener que hacer polling por parte del cliente.

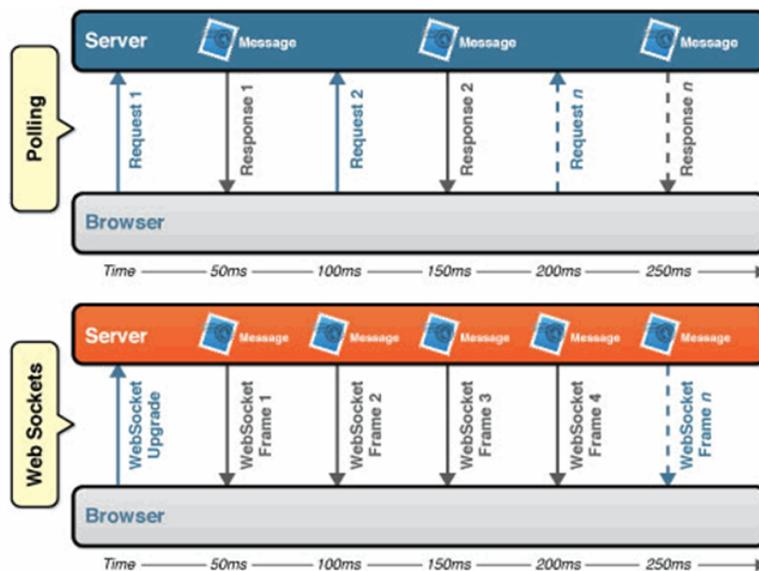


Figura 1.1: Estructura WebSockets

(DocPlayer.es2016, DocPlayer.es2016)

Aquí podemos ver que esta tecnología nos permite intercambiar información entre el cliente y el servidor cuando cualquiera de las dos partes así lo requiera, sin necesidad de que el cliente esté sondeando al servidor, y con la ventaja extra

de que las partes pueden enviar información al mismo tiempo por el mismo canal ya que al ser una conexión Full Duplex los mensajes no “chocaran”.
(Network2016c, Network2016c)

1.6. Alcances, Limitaciones y Resultados

1.6.1. Alcances

El alcance de la investigación, es el desarrollo de los siguientes módulos administrativos a través del prototipo de red social:

- **Perfil de usuario:** Permitirá el acceso a la información personal registrada por el mismo usuario.
- **Módulo de consulta de pagos:** Permitirá a los usuarios tener la posibilidad de consultar el histórico de pagos y pagos pendientes.
- **Módulo de servicios:** Permitirá realizar solicitudes como reserva de salón social de eventos, zonas comunes y reserva de estacionamientos.
- **Zona de noticias y eventos:** Publicación mensual de resumen de control de ingresos y gastos de la administración. Así mismo publicaciones individuales por parte de los usuarios registrados.
- **Módulo de terceros:** Solo información de contacto de proveedores.
- **Módulo de reuniones:** Permitirá que los residentes que se registren puedan interactuar y compartir ideas en pro de mejorar su espacio habitacional.
- **Chat:** Permitirá crear conversaciones personales entre usuarios registrados.

1.6.2. Limitaciones

- El prototipo no contiene un módulo contable para generar los estados financieros.
- El módulo de consulta de pagos permitirá verificar pagos realizados y pendientes, más no realizar pagos en línea.
- Solo podrá existir un usuario registrado por vivienda para el ingreso a la red social, esto con el fin de garantizar la seguridad en la información.
- La autorización de solicitudes de servicios solo las podrá aprobar el administrador, así como el registro de acceso a la red social.
- No se pueden crear conversaciones grupales (más de 2 usuarios).
- El prototipo no se implementará en un ambiente real, las pruebas se realizarán en un entorno simulado.

1.6.3. Resultados Esperados

Se espera contar con un prototipo de red social modular que sea funcional y de cobertura a los procesos mas importantes de la administración del conjunto residencial así como a los tramites y servicios que necesitan los usuarios. Se espera la participación de los usuarios de un conjunto residencial para validar la viabilidad de la aplicación así como los módulos que pueden llegar a ser implementados en este proyecto, se entiende que bajo el tiempo establecido y por las limitaciones no se pueden desarrollar todos módulos que surgen de la investigación, pero si se documentaran para que estén pendientes en próximas versiones de la aplicación.

Se pretende validar como una metodología de desarrollo ágil ayuda en el desarrollo del proyecto, que aspectos negativos y positivos se pueden evidencias y así mismo generar un feedback para futuros desarrollos bajo esta guía.

Se entregara una documentación que describe todo el desarrollo de la aplicación, como fue la metodología, las tecnologías que se utilizaron , cada uno de los componentes y las configuraciones que se utilizaron para estas tecnologías y una validación de cada una, también se mostrara un informe en el que se describan los resultados obtenidos. Este informe será complementado por anexos que contengan la documentación generada a lo largo del proyecto.

Capítulo 2

Descripción de la Organización

2.1. Conjunto Residencial Torres de Hayuelos

se selecciono el conjunto residencial torres de Hayuelos para simular el prototipo de red social, este conjunto residencial facilita el desarrollo del proyecto gracias a que el administrador actual del conjunto residencial esta dispuesto a colaborar aportando la información referente a la administración y los diferentes tramites que este realiza día a día.

2.1.1. Misión

El Conjunto Residencial Torres de Hayuelos tiene como misión acatar las normas de convivencia en la propiedad horizontal con respeto y responsabilidad.

2.1.2. Visión

El Conjunto Residencial Torres de Hayuelos será el mejor conjunto de propiedad horizontal, en cuanto a convivencia de su propia realidad y la de su entorno, con excelentes desempeños en todas las dimensiones de su vida en un contexto de interacción armónica donde el quehacer diario se sustente en los valores, la cultura y el respeto a los demás.

2.1.3. Objetivos

La presente ley regula la forma especial de dominio, denominada propiedad horizontal, en la que concurren derechos de propiedad exclusiva sobre bienes privados y derechos de copropiedad sobre el terreno y los demás bienes comunes, con el fin de garantizar la seguridad y la convivencia pacífica en los inmuebles sometidos a ella, así como la función social de la propiedad.

Parte III
Desarrollo

Capítulo 3

Investigación

3.1. Recolección de información

3.1.1. Recogiendo información del proyecto

Cuando se propuso el proyecto de red social ya se tenía una vaga idea de los procesos que se llevan a cabo en un conjunto residencial por experiencias propias y por observación informal de las situaciones, mas sin embargo para crear una solución mas completa era necesario realizar una investigación y documentar otros puntos de vista con el fin de crear un sistema mas genérico y a la vez completo, para esta solución se plantearon 2 metodologías par levantamiento de información, la encuesta y la entrevista.(EIRL2015, EIRL2015)

Solo se toma en cuenta la opinión de 2 roles dentro de la investigación, el administrador del conjunto residencial y los residentes que son propietarios, basados en estos roles se pretendía determinar si se debía incluir algún rol adicional. para el caso de la administración se facilitaba utilizar la entrevista debido a que es una sola persona, para los residentes se realizo una encuesta con preguntas referentes a la información suministrada por el administrador y a la solución que se pretendía entregar.

3.1.1.1. Entrevista

La estructura de la entrevista se dividió en 3 universos:

- Uso de las TIC en la administración de un conjunto residencial
- Procesos y funciones mas importantes en la administración de conjuntos residenciales
- Problemas comunes dentro de la administración de un conjunto residencial

La administración de un conjunto residencial abarca bastantes obligaciones, para el desarrollo de este proyecto nos enfocamos en los procesos que involucren

tanto al administrador como a los residentes. como resultado de la entrevista se crearon las siguientes conclusiones:

- El uso de las TIC en los conjuntos residenciales no es fuerte, depende de las metodologías del administrador hace uso de herramientas en la nube para almacenar la información pero no se utiliza para compartir información con los residentes.
- Se hace uso de herramientas como el Whatsapp y el correo electrónico para casos muy puntuales.
- Los procesos mas desgastan la administración están dentro del manejo financiero, hacer cobros de administración, realizar inversión del dinero.
- Los residentes prefieren que el administrador haga parte del conjunto residencial para tener un canal de comunicación mas asequible y mayor compromiso
- Las asambleas son una tarea complicada debido a que es difícil poner de acuerdo a todos los residentes
- Existe mucho desinterés por parte de los propietarios.
- Los residentes se quejan de la administración pero en la mayoría de los casos son residentes que no conocen bien los procesos.
- La administración es responsable del cuidado de los recursos físicos y de hacer seguimiento a los proveedores de servicios.(UNAD2015a, UNAD2015a)

3.1.1.2. Encuesta

Para diseñar la encuesta se utilizo la información que se recolecto por medio de la entrevista con la administración, la encuesta consta de 10 preguntas basadas en lo siguiente:

- Uso de redes sociales y portales de información.
- Relación con la administración del conjunto residencial y nivel de participación.
- Procesos y servicios de los conjuntos residenciales.(de Cordoba2015, de Cordoba2015)

3.1.2. Tabulación y ordenamiento de la información

La información recogida por medio de la entrevista nos ayuda a enfocar las encuestas que son la fuente real de información para definir los módulos que se van a desarrollar en el proyecto, la encuesta y los resultados se pueden observar en el Anexo 01 y Anexo 02.(UNAD2015b, UNAD2015b)

3.2. Análisis de la información

El objetivo principal de la entrevista y las encuestas era validar la viabilidad del desarrollo del proyecto, se validó si los usuarios si hacen uso de redes sociales, ¿Si es necesaria una red social donde solo se maneje información correspondiente al conjunto residencial?. Bajo el análisis de los resultados se generan las siguientes conclusiones:

- El 90 por ciento de las personas entrevistadas hacen uso de las redes sociales y portales de internet
- El 100 por ciento de las personas considera que la información correspondiente al conjunto residencial no es transparente
- Un 50 por ciento de los encuestados tiene un nivel de participación muy bajo en las decisiones de la administración por la metodología que se implementa.
- El 100 por ciento de los encuestados considera que debería existir una pagina web para consultar la información del conjunto residencial.

Uno objetivo principal de levantar la información de los conjuntos residenciales era validar la necesidad de los módulos que se habían planteado inicialmente e identificar que módulos deberían ser incluidos o cuales no son útiles, entre los planteados inicialmente encontramos:

- Modulo de reservas o alquiler de recursos físicos.
- Modulo de publicaciones y eventos
- Modulo de interacción entre usuarios
- Modulo de consulta de información del conjunto residencial

Como resultado de las encuestas se encontraron otros servicios que los residentes proponen lo cual deja abierto el proyecto para las mejoras futuras. (Kumar et al.2016, Kumar et al.2016)

3.3. Selección de módulos a implementar

Segun un preanálisis de los conjuntos residenciales se plantearon unos modulos iniciales para el desarrollo de la red social modular, los modulos planteados inicialmente tienen buena acogida por los usuarios, por este motivo se mantuvieron y decidieron implementar los siguientes modulos (Tilkov and Vinoski2010, Tilkov and Vinoski2010):

3.3.1. Modulo de Reservas

El modulo de reservas se plantea para facilitar el alquiler de recursos físicos del conjunto residencial, entre los espacios que se plantearon inicialmente están los salones de recepciones de los conjuntos residenciales, las terrazas BBQ y los parqueaderos de visitante. Debido a que los conjuntos residenciales son todos diferentes se plantea que sea un modulo parametrizable por el administrador, existen conjuntos residenciales que cuentan con mas zonas sociales o cuentan con zonas deportivas como canchas de fútbol o piscinas.

3.3.2. Modulo de Chat

El chat es un complemento para que los usuarios puedan interactuar entre si sin necesidad de compartir datos personales como el correo electrónico o el numero telefónico, se propone como un espacio para encontrar personas dentro del conjunto residencial y entablar nueva relaciones.

3.3.3. Modulo de Proveedores

Este modulo es requerido por los usuarios residentes de los conjuntos residenciales, se plantea como un modulo para compartir información referente a la administración del conjunto residencial, los usuarios por medio de este modulo pueden consultar la información de los proveedores de servicios de los conjuntos residenciales en cualquier momento, como por ejemplo la empresa que presta servicios de seguridad, aseo o mantenimiento del conjunto residencial.

3.3.4. Modulo de Pagos

Este modulo seria uno de los módulos mas importantes para los usuarios residentes, el 100 por ciento de los residentes están de acuerdo con que los pagos se puedan realizar por medio electrónico y se pueda consultar los pagos pendientes y los pagos realizados. Por limitaciones de tiempo y adquisición de servicios adicionales este modulo se desarrolla como un modulo de consulta de histórico de pagos y listado de pagos pendientes, dejando para un desarrollo futuro la inclusión del medio de pago electrónico.

3.3.5. Modulo de Publicaciones

Este modulo se plantea como un modulo para que los usuarios residentes puedan compartir información con todos los usuarios residentes del conjunto residencial, este modulo permite publicar noticias, eventos o clasificados que puedan ser de interés para el conjunto residencial, este modulo también es una oportunidad para comunicar novedades de la administración.

Capítulo 4

Arquitectura Empresarial

4.1. Diagramas de Arquitectura Capa de Negocio

4.1.1. Punto de Vista de Organización

4.1.1.1. Metamodelo

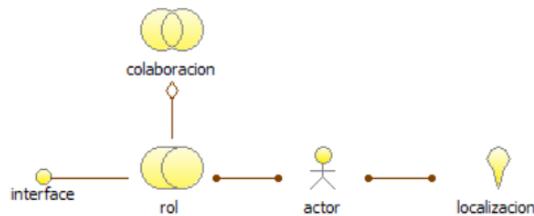


Figura 4.1: Metamodelo Punto de Vista de Organización

Se centra en la organización(interna) de una compañía, un departamento, una red de compañías o de otra entidad organizativa. Es posible presentar modelos de esta punto de vista como diagramas de bloques anidados, sino también de una manera más tradicional, tales como organigramas. El punto de vista Organización es muy útil en la identificación de las competencias, la autoridad y las responsabilidades dentro de una organización, el metamodelo propuesto por archimate se puede observar en la figura 4.1.(Sauer and Willcocks2004, Sauer and Willcocks2004)

4.1.1.2. Modelo

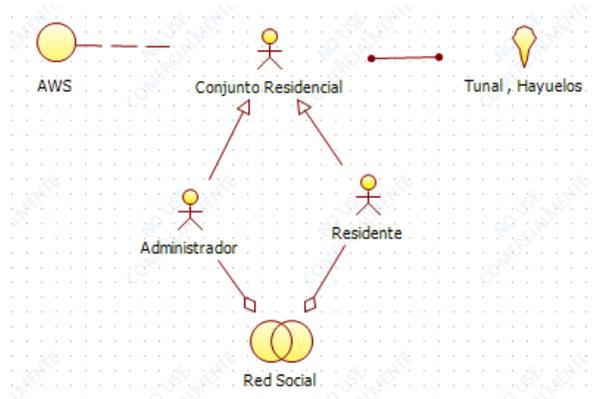


Figura 4.2: Modelo Punto de Vista de Organización

Se tiene en cuenta para este modelo los dos actores que interactúan en la red social, el administrador y el usuario residente, estos 2 conforman el conjunto residencial sobre el cual actual la aplicación de la red social.

4.1.2. Punto de Vista de Cooperación de Actor

4.1.2.1. Metamodelo

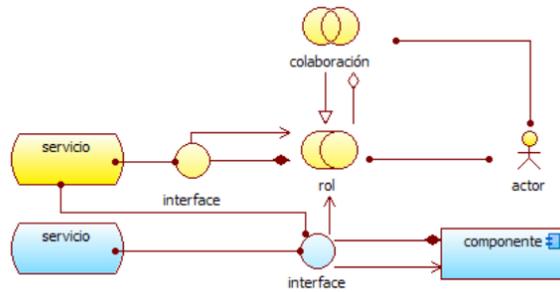


Figura 4.3: Metamodelo Punto de Vista de Cooperación de Actor

El punto de vista Actor Cooperación se centra en las relaciones de los actores entre sí y con su entorno. Un ejemplo común de esto es el "diagrama de contexto", lo que pone una organización en su entorno, que consiste en las partes

externas, tales como clientes, proveedores y otros socios comerciales. Es muy útil en la determinación de dependencias externas y colaboraciones y muestra la cadena de valor o de la red en el que opera el actor como se observa en la figura 4.3 el metamodelo propuesto por el framework archimate.

4.1.2.2. Modelo

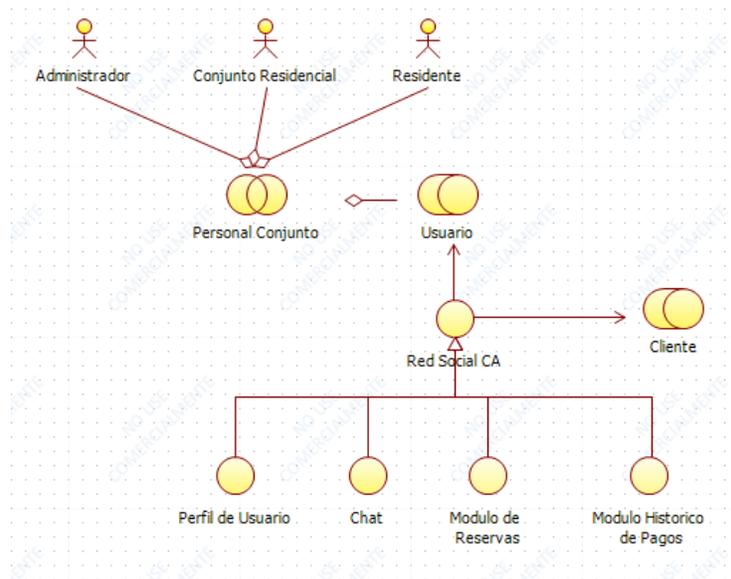


Figura 4.4: Modelo Punto de Vista de Cooperación de Actor

Por medio de una colaboración entre el administrador, residentes y la plataforma física del conjunto residencial se da forma a los definidos integrantes del conjunto residencial (figura 4.4) estos integrantes interactúan entre sí haciendo uso de los módulos definidos como el caso del chat, el perfil de cada usuario, los módulos de reservas y las consultas de trámites como pagos y servicios. Estos módulos junto a los integrantes dan vida a la red social, la cual con cada uno de sus componentes puede ser escalable a cualquier conjunto residencial haciendo las tareas más genéricas facilitando la administración del conjunto residencial.

4.1.3. Punto de Vista de Funcion de Negocio

4.1.3.1. Metamodelo

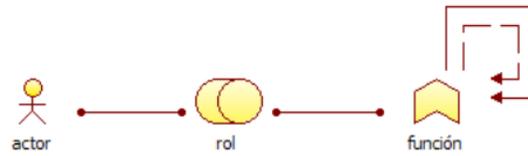


Figura 4.5: Metamodelo Punto de Vista de Funcion de Negocio

El punto de vista de funcion de negocio muestra las principales funciones de negocio de una organización y sus relaciones en términos de los flujos de información, el valor, y cosas entre ellos, en la figura 4.5 se puede observar el metamodelo propuesto por archimate.

4.1.3.2. Modelo

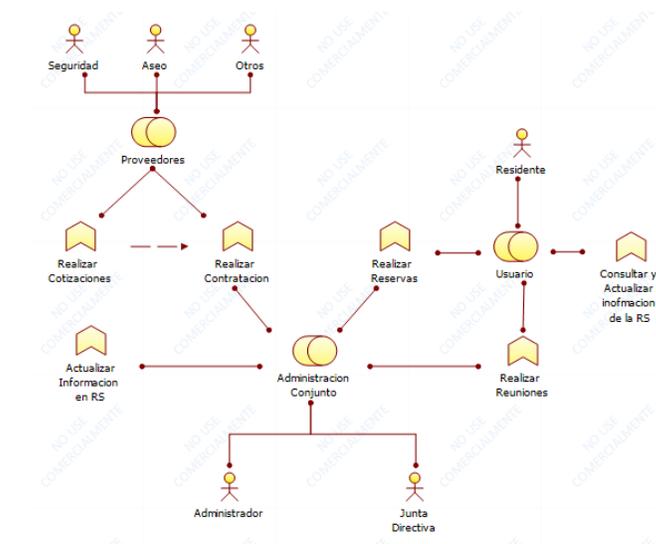


Figura 4.6: Modelo Punto de Vista de Funcion de Negocio

Dentro de las funciones planteadas podemos ver como el administrador esta presente en todas las funciones, este es el eje principal de la red social y por medio de cada una de las funciones los residentes o usuarios retornan valor por medio del administrador, la figura 4.6 da una perspectiva mas completa de como cada una de las funciones que debe tener el conjunto interactúa con los usuarios.(Heyl2014, Heyl2014)

4.1.4. Punto de Vista de Proceso de Negocio

4.1.4.1. Metamodelo

El metamodelo propuesto por archimate en la figura 4.7 de punto de vista de procesos de negocio se utiliza para mostrar la estructura de alto nivel y la composición de uno o más procesos de negocio. Al lado de los procesos mismos, este punto de vista contiene otros conceptos directamente relacionados, tales como:

- Los servicios que ofrece un proceso de negocio con el mundo exterior, que muestra cómo un proceso contribuye a la realización de producto de la compañía.
- La asignación de los procesos de negocio a los roles, lo que da una idea de las responsabilidades de los actores asociados.
- La información utilizada por el proceso de negocio.

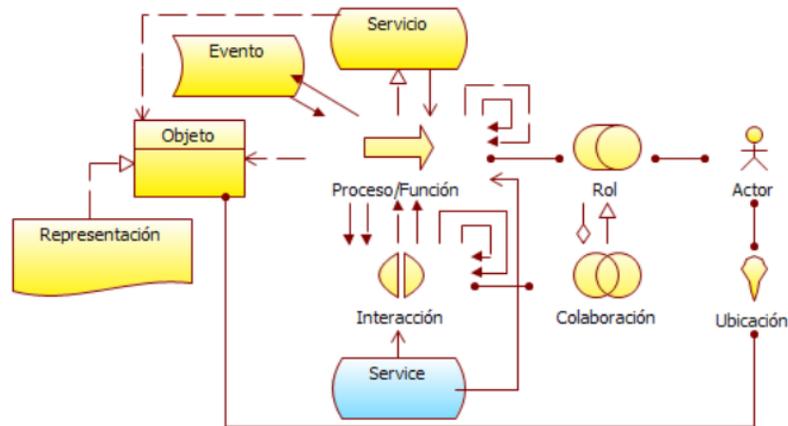


Figura 4.7: Metamodelo Punto de Vista de Proceso de Negocio

4.1.4.2. Modelo

En el modelo de la figura 4.8 se expresa el proceso de reserva de espacios dentro del conjunto residencial, inicia con una validación de disponibilidad de espacios para realizar una reserva, después de validar se hace el procedimiento para radicar la reserva del espacio en el tiempo seleccionado. Por parte de la administración se debe aprobar la solicitud de reserva, esta solicitud puede ser aprobada o rechazada según las condiciones de la misma. una vez que se ha realizado la reserva y se ha aprobado por parte de la administración, se procede a entregar una respuesta al usuario.



Figura 4.8: Modelo Punto de Vista de Proceso de Negocio

4.1.5. Punto de Vista Proceso de Cooperacion de Negocio

4.1.5.1. Metamodelo

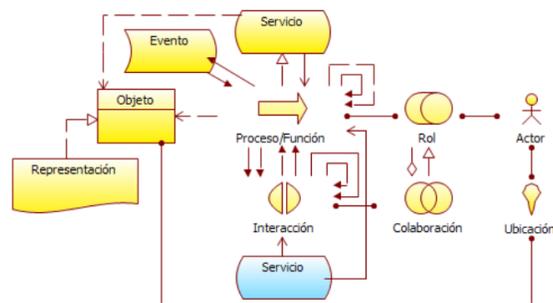


Figura 4.9: Metamodelo Punto de Vista Proceso de Cooperacion de Negocio

La Cooperación punto de vista de procesos de negocio se utiliza para mostrar las relaciones de uno o más procesos de negocio entre sí y / o con su entorno como se observa en la figura 4.9. Puede ser utilizado tanto para crear un diseño de alto nivel de los procesos de negocio dentro de su contexto y para proporcionar un gestor operativo responsable de uno o más de dichos procesos con penetración en sus dependencias.

4.1.5.2. Modelo

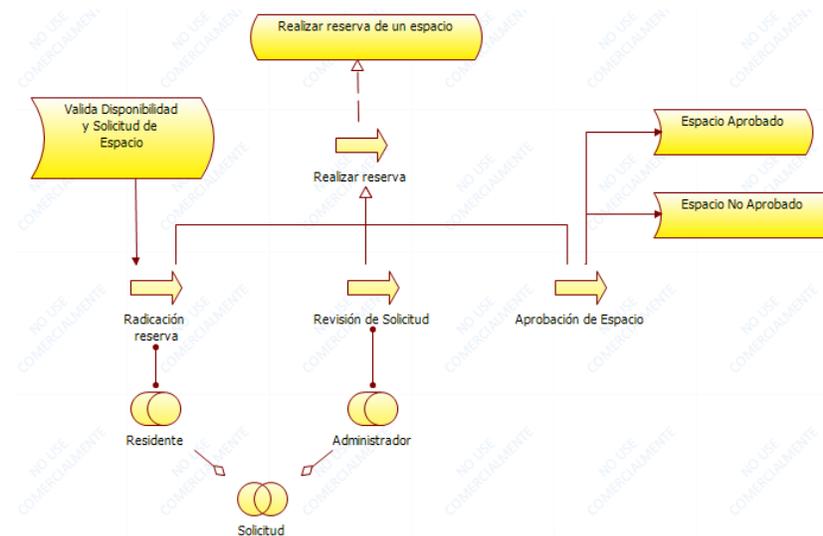


Figura 4.10: Modelo Punto de Vista Proceso de Cooperacion de Negocio

En el proceso de reserva de espacios estará disponible todo el tiempo para los usuarios, este proceso se encuentra dentro del modulo de reserva que facilita los tramites entre los residentes y la administración del conjunto residencial, este proceso va desde radicar la solicitud hasta la aprobación y préstamo del espacio como se observa en la figura 4.10.

4.1.6. Punto de Vista de Producto

4.1.6.1. Metamodelo

El punto de vista del producto (figura 4.11) representa el valor que estos productos ofrecen a los clientes u otras partes externas involucradas y se muestra la composición de uno o más productos en términos de la Constitución (de negocios o aplicación) los servicios, y el contrato (s) asociado u otros acuerdos.

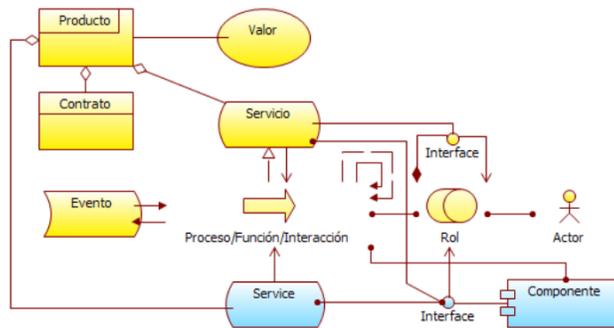


Figura 4.11: Metamodelo Punto de Vista de Producto

4.1.6.2. Modelo

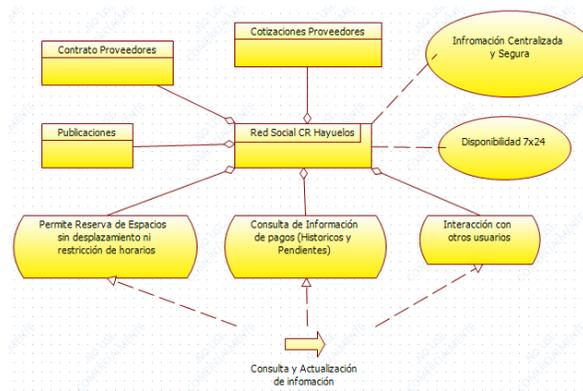


Figura 4.12: Modelo Punto de Vista de Producto

El producto se define Red Social CR Hayuelos, en el se enmarcan cada una de las características que lo componen así como los objetos que resaltan del producto, todo está enfocado a la consulta y actualización de la información. En la figura 4.12 se puede ver el valor que se obtiene del producto.

4.2. Diagrama de Arquitectura Capa de Aplicación

4.2.1. Punto de Vista de Comportamiento de Aplicación

4.2.1.1. Metamodelo

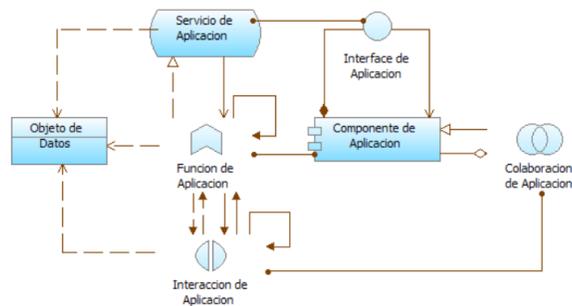


Figura 4.13: Metamodelo Punto de Vista de Comportamiento de Aplicación

El punto de vista del comportamiento de aplicación describe el comportamiento interno de una aplicación. Este punto de vista es útil en el diseño del comportamiento principal de aplicaciones, o en la identificación de solapamiento funcional entre diferentes aplicaciones (figura 4.13).

4.2.1.2. Modelo

En este modelo (figura 4.14) se observa cómo cada función hace parte de un componente de aplicación y cómo se conectan entre ellos, en primera instancia encontramos el componente del administrador, este tiene las funciones de:

- Administrar Usuarios: Agregar quitar y editar usuarios dentro de la red social Administrar Recursos: Gestionar y validar solicitudes de recursos físicos del conjunto residencial, el administrador puede seleccionar qué recursos están disponibles para reserva.
- Administrar Información de los usuarios: Los usuarios comparten cierta información y el administrador es capaz de escoger qué información es necesario quitar o publicar.
- Publicar Información de Proveedores: La información de los proveedores es importante para la comunidad y el administrador tendrá la opción de agregar o quitar esta información.

El componente de aplicación de Usuario tiene las siguientes funciones:

- Realizar publicaciones de usuarios: Los usuarios realizan ciertas publicaciones que pueden ser afirmativas o pueden ser solicitudes de permisos o reservas de espacios.
- Consultar información: así como se comparten publicaciones e información de los proveedores por parte del administrador, los usuarios necesitan un componente que les permite visualizar esta información.

Todas las funciones se integran con el componente general de red social.

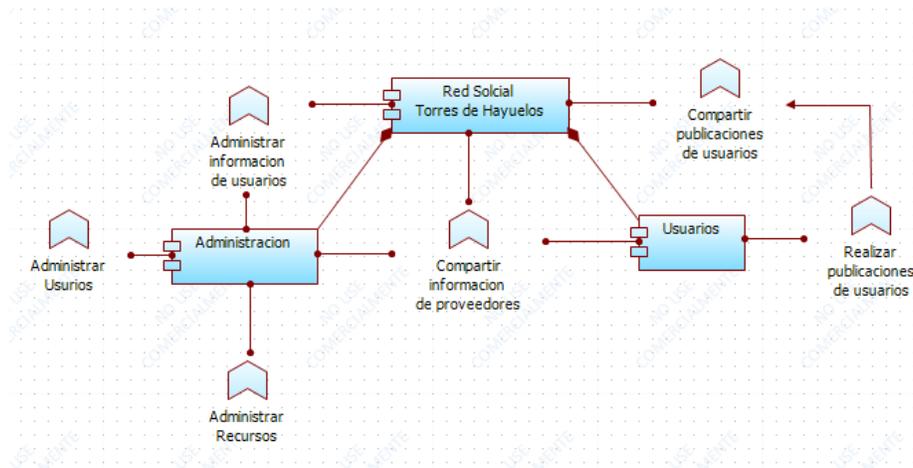


Figura 4.14: Modelo Punto de Vista de Comportamiento de Aplicación

4.2.2. Punto de Vista de Cooperación de Aplicación

4.2.2.1. Metamodelo

Este punto de vista describe las relaciones entre los componentes de aplicación en términos de flujo de información. Utilizado principalmente para mostrar a la organización un mapa acerca de los componentes, así como una idea de dirección en la ejecución de los mismos. En la figura 4.15 se puede observar el metamodelo propuesto por Archimate.

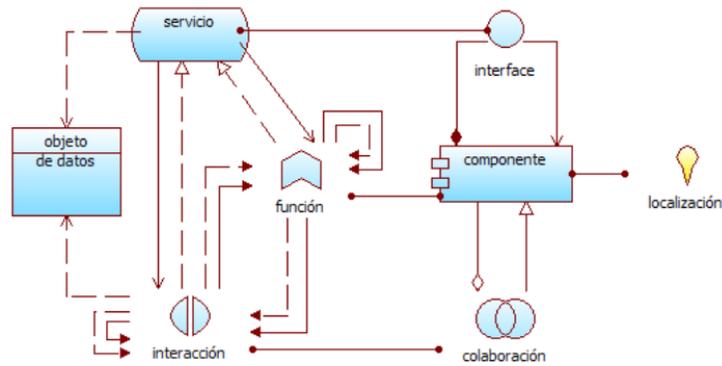


Figura 4.15: Metamodelo Punto de Vista de Cooperación de Aplicación

4.2.2.2. Modelo

En este modelo (figura 4.16) se visualiza como se distribuyen las diferentes aplicaciones en las capas del back y el front, El componente de software principal (Red social Torres de Hayuelos) tiene componentes en ambas capas pero por el lado del software que utiliza el administrador y los usuarios si se puede especificar en detalle su participación en cada capa. El componente de administración de usuarios por parte del administrador se expresa dentro de la capa Back ya que este procesos se realiza por medio de aplicaciones que no tienen componentes visuales. Compartir aplicaciones, realizar publicaciones y compartir información de los proveedores es algo que tiene componentes en la capa front para poder ser visualizada por todos los usuarios.

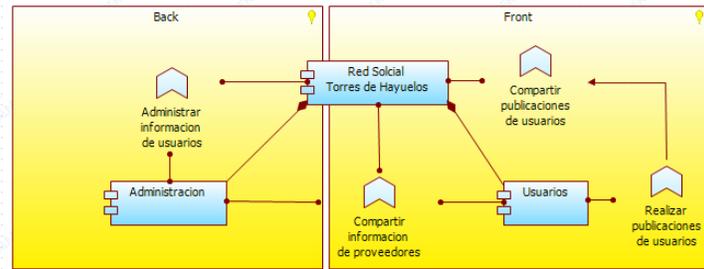


Figura 4.16: Modelo Punto de Vista de Cooperación de Aplicación

4.2.3. Punto de Vista de Estructura de Aplicación

4.2.3.1. Metamodelo

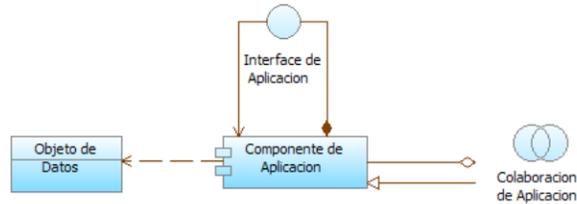


Figura 4.17: Metamodelo Punto de Vista de Estructura de Aplicación

Este punto de vista describe la estructura de una o más aplicaciones o componentes (figura 4.17). Es muy útil en diseño y entendimiento de la estructura principal de la aplicación o componentes que interactúan con el fin de identificar a modo componentes heredados que son utilizables en labores de integración /migración.

4.2.3.2. Modelo

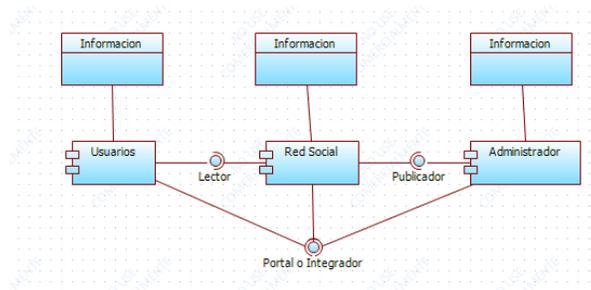


Figura 4.18: Modelo Punto de Vista de Estructura de Aplicación

El objetivo principal de nuestro software es compartir la información de la administración y el conjunto residencial así como los residentes como el fin de hacer que los procesos sean más transparentes, por este motivo se establece que el objeto principal de cada uno de los componentes es la información. Desde el punto de vista de la estructura de la aplicación se establece como flujo primario que el administrador publica información en la red social y los usuarios son lectores de esta, el caso en el cual el usuario es el encargado de publicar información pasa a un segundo plano solo para dar valor agregado al sistema

de software en la figura 4.18 se observa el modelo descrito con cada una de las interacciones descritas anteriormente.

4.2.4. Punto de Vista de Uso de Aplicación

4.2.4.1. Metamodelo

Este punto de vista describe el cómo las aplicaciones son usadas para soportes uno o más procesos de negocio y como ellos son usados por otras aplicaciones. Es muy útil en diseño y aplicación para identificar los servicios necesarios por los procesos de negocio y otras aplicaciones para describir los servicios que están disponibles. En la figura 4.19 se puede observar el metamodelo propuesto por el framework de arquitectura Archimate que se describió anteriormente.

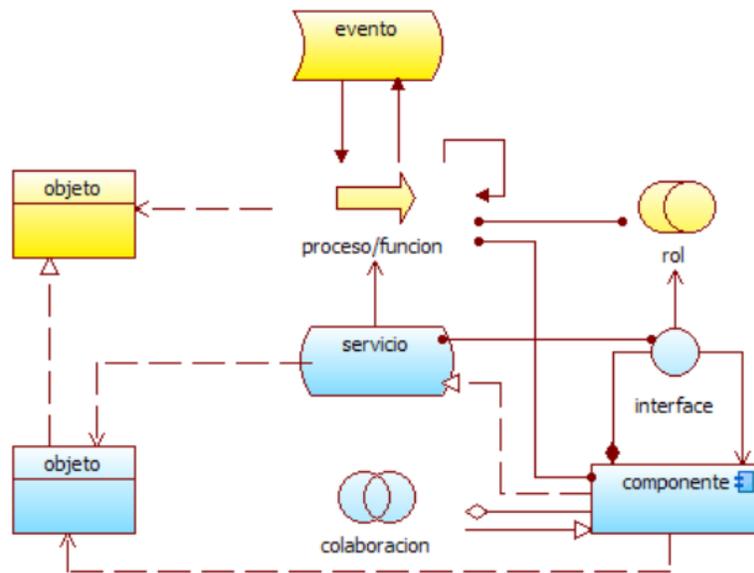


Figura 4.19: Metamodelo Punto de Vista de Uso de Aplicación

4.2.4.2. Modelo

En este punto de vista más alto nivel extraemos la Red Social Torres de Hayuelos como el componente de software principal, su función se resalta como recibir información tanto del administrador y de los usuarios para ser organizada, el proceso de negocio que se realiza con este componente de software es administrar la información del conjunto residencial, se establece qué información se va a compartir y se presenta en diferentes roles y vistas figura 4.20. la función

de este proceso de negocio es compartir la información del conjunto residencial, para que esto se cumpla debe estar bien administrada.

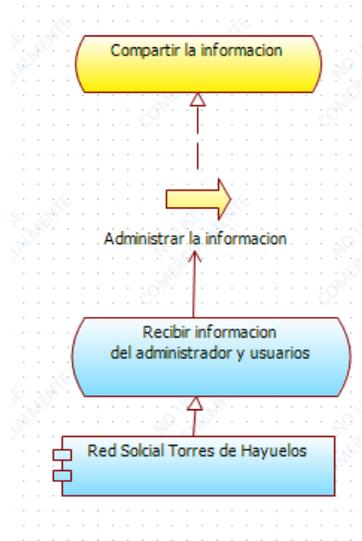


Figura 4.20: Modelo Punto de Vista de Uso de Aplicación

4.3. Diagrama de Arquitectura Capa de Infraestructura

4.3.1. Infraestructura

4.3.1.1. Metamodelo

El punto de vista de infraestructura (figura 4.21) contiene los elementos de la infraestructura de hardware y software de apoyo a la capa de aplicación, tales como dispositivos físicos, redes o software del sistema (por ejemplo, sistemas operativos, bases de datos y middleware).

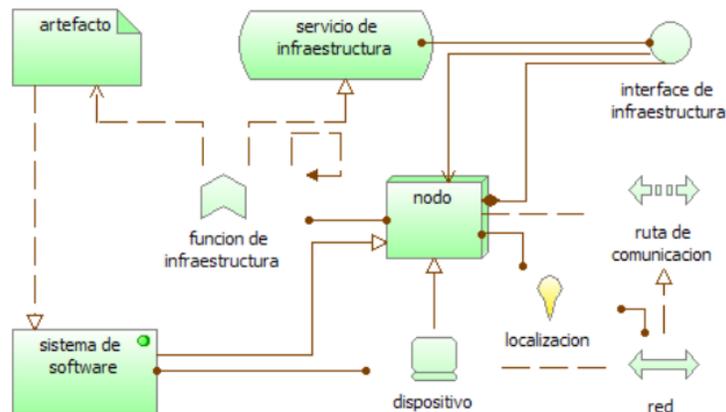


Figura 4.21: Metamodelo Punto de Vista Infraestructura

4.3.1.2. Modelo

La infraestructura se compone inicialmente de la parte del cliente que puede acceder desde cualquier dispositivo que tenga un navegador web (Celular, Tablet, Notebook). el cliente va a interactuar con una página web desarrollada con HTML, Hojas de estilo y JavaScript, para esta parte del cliente se implementa el framework de diseño Bootstrap 3, para utilizar estas tecnologías basta con usar un navegador web moderno como Google chrome o Mozilla Firefox entre otros.

Independientemente del dispositivo que se elija debe contar con una conexión a internet, ya sea por red local o celular. con una conexión a internet establecida se implementan protocolos de comunicación TCP/IP como HTTP y WebSockets para comunicarse con el servidor que alberga la aplicación.

Ya por el otro extremo podemos observar una infraestructura física que para este caso es rentada, basada en los servicios ofrecidos por AWS (Amazon Web Services), amazon nos alquila una máquina que puede ser compartida en la cual creamos 2 Instancias, una que administra la base de datos de la aplicación y otra en la cual se desarrolla toda la lógica de la aplicación y es la encargada de recibir todas las peticiones por parte del cliente, basada en una servidor de aplicaciones desarrollado con Node.js, se aclara que ambas instancias del servidor de Amazon corren sobre un sistema operativo Linux. en la figura 4.22 se3 puede observar el modelo realizado.

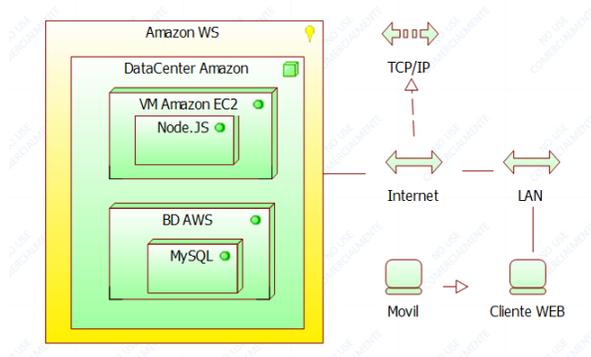


Figura 4.22: Modelo Punto de Vista Infraestructura

4.3.2. Uso Infraestructura

4.3.2.1. Metamodelo

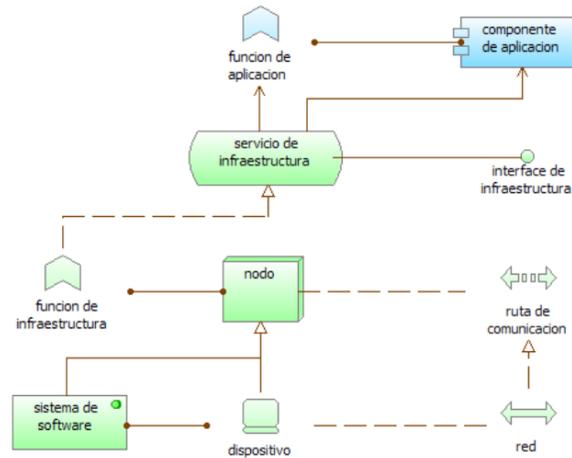


Figura 4.23: Metamodelo Punto de Vista Uso Infraestructura

El punto de vista de uso de infraestructura (figura 4.23), muestra cómo las aplicaciones son compatibles con la infraestructura de software y hardware: los servicios de infraestructura son entregados por los dispositivos; software y redes de sistema que se proporcionan a las aplicaciones. Este punto de vista desempeña un papel importante en el análisis de rendimiento y escalabilidad, puesto

que se refiere la infraestructura física para el mundo lógico de aplicaciones. Es muy útil en la determinación de los requisitos de rendimiento y calidad de la infraestructura basada en las exigencias de las diferentes aplicaciones que lo utilizan.

4.3.2.2. Modelo

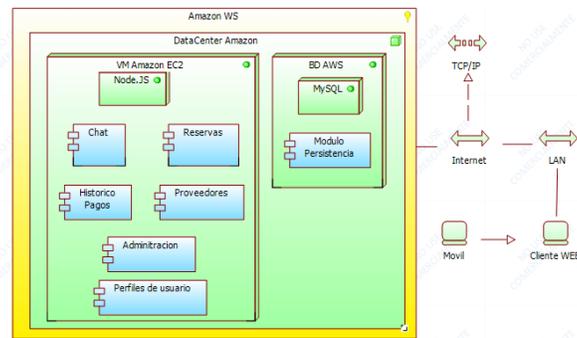


Figura 4.24: Modelo Punto de Vista Uso Infraestructura

Para este modelo simplemente agregamos los diferentes componentes de la aplicación que se implementan, e indicamos bajo que estructura física están albergadas. En la figura 4.24 se observa la que en la instancia de AWS de base de datos solamente se despliega una aplicación para realizar las transacciones a bases de datos, el resto de aplicaciones son desplegadas en la otra instancia donde está el node.js y todos los módulos que componen la aplicación de la Red Social.

4.3.3. Organización e Implementación

4.3.3.1. Metamodelo

El Punto de vista Organización e Implementación propuesto en la figura 4.25 muestra cómo se realizan una o más aplicaciones en la infraestructura. Esto comprende el mapeo de aplicaciones (lógicas) y componentes de artefactos (físicas), tales como Enterprise Java Beans, y el mapeo de la información utilizada por estas aplicaciones y componentes sobre la infraestructura de almacenamiento subyacente; por ejemplo, bases de datos o tablas de otros archivos. El Punto de vista Organización e Implementación juega un papel importante en el análisis de rendimiento y escalabilidad, puesto que se refieren la infraestructura física para el mundo lógico de aplicaciones. En seguridad y análisis de riesgos, este punto de vista se usa para identificar, por ejemplo, las dependencias y los riesgos críticos.

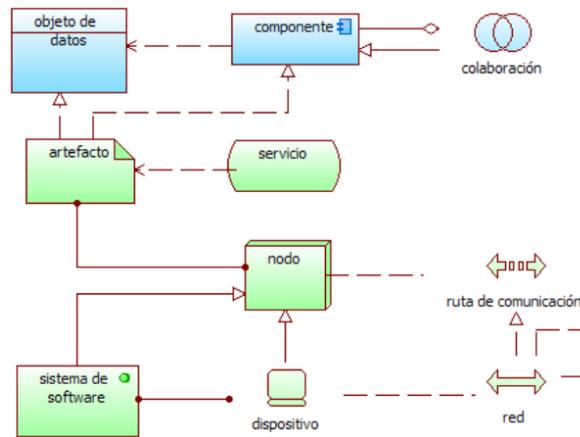


Figura 4.25: Metamodelo Punto de Vista Implementacion y Organizacion

4.3.3.2. Modelo

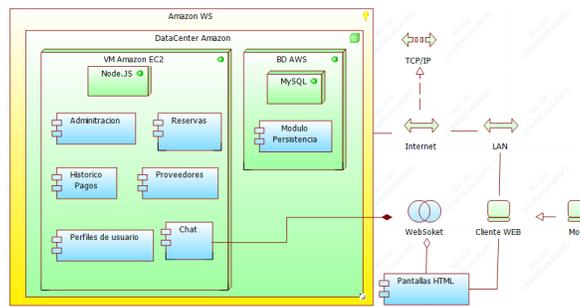


Figura 4.26: Modelo Punto de Vista Implementacion y Organizacion

En este modelo nos basamos en prácticamente el mismo modelo de infraestructura y el uso, pero agregamos un componente de software por el lado del cliente, este se basa en incluir por la parte del cliente un protocolo de comunicación que establece una un enlace con un componente de software del servidor remoto, este protocolo se basa en webSockets el cual abre un canal de comunicación entre el cliente y el componente del servidor y transmite información sin necesidad de enviar solicitudes, simplemente se envía la información por el canal que ya se estableció. este componente es útil para aplicaciones en tiempo real como el caso del chat (figura 4.26).

4.4. Diagramas de Arquitectura Capa Motivacional

4.4.1. Implicados

4.4.1.1. Metamodelo

El punto de vista de los Implicados permite analizar el modelo para las partes interesadas, los controles internos y externos para el cambio, y las evaluaciones (en términos de fortalezas, debilidades, oportunidades y amenazas) de estos controles. Además, los vínculos con los objetivos iniciales que abordan estas preocupaciones y su evaluación son descritos. Estos objetivos constituyen la base para el proceso de ingeniería de requisitos, incluyendo el refinamiento de objetivos, análisis de la contribución y el conflicto, y la derivación de los requisitos que dan creación a los objetivos (figura 4.27).

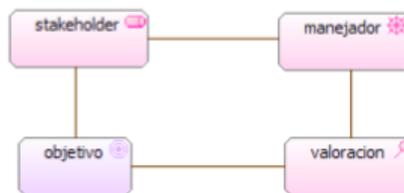


Figura 4.27: Metamodelo Punto de Vista Implicados

4.4.1.2. Modelo

Para el caso de los implicados dentro del proyecto está el rol del administrador y el rol del residente, los objetivos de estos dos roles son muy similares, su finalidad es compartir información ya sea referente a la administración del conjunto residencial por el lado del administrador, o sus opiniones, propuestas y votos por parte de los residentes, mas sin embargo el administrador tiene un poco más de responsabilidad en este caso. El administrador tiene la obligación de subir la información relevante del conjunto residencial con el objetivo de hacerla transparente para los residentes, para esto utiliza la herramienta de red social, mientras por el lado del residente tiene el deber de participar mas no está obligado a hacerlo, en este caso el objetivo de la herramienta es potenciar la participación del residente, es decir el administrador facilita el acceso a la información y el residente simplemente utiliza un medio de comunicación, en la figura 4.28 se observan los dos implicados directos de la aplicación.

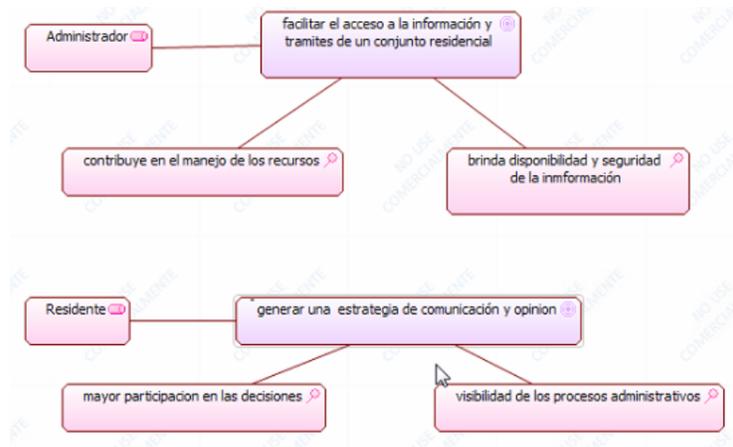


Figura 4.28: Modelo Punto de Vista Implicados

4.4.2. Realizacion de Objetivos

4.4.2.1. Metamodelo

Permite a un diseñador el refinamiento para el modelo de los objetivos (de alto nivel) en objetivos más concretos, y el refinamiento de objetivos concretos en requisitos o condiciones que describen las propiedades que son necesarias para alcanzar los objetivos. El refinamiento de las metas en sub-objetivos se modela utilizando la relación de agregación (Figura 4.29).

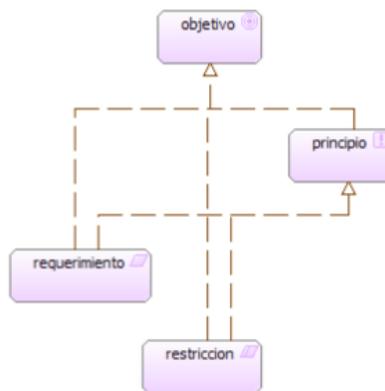


Figura 4.29: Metamodelo Punto de Vista Realizacion de Objetivos

4.4.2.2. Modelo

Este punto de vista generamos un requerimiento más detallado basados en un objetivo, para este caso generar una estrategia de comunicación y opinión, de este objetivo podemos encontrar una restricción que nos ayuda a crear más específico el requerimiento, ya que se debe crear un medio de opinión para la interacción de los usuarios, esto hace que sea más fácil plantear un requerimiento basado en implementar un grupo de discusión, esto termina siendo un componente fundamental para la red social debido a que no ayuda a centralizar las opiniones y a compartir la información del conjunto residencial (Figura 4.30).

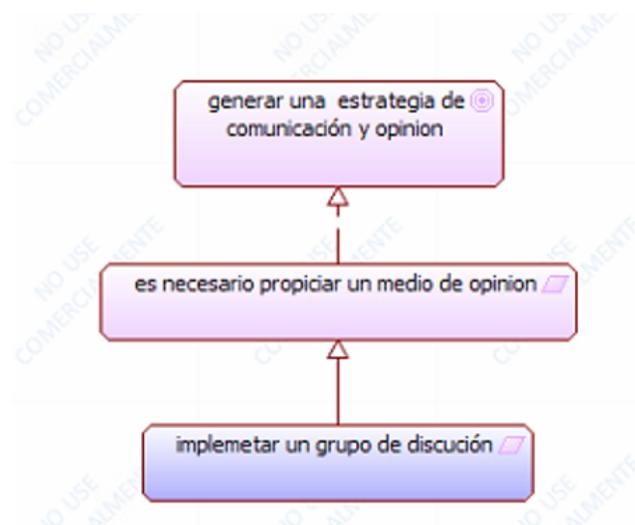


Figura 4.30: Modelo Punto de Vista Realización de Objetivos

4.4.3. Motivación

4.4.3.1. Metamodelo

El punto de vista de motivación permite al diseñador o analista modelar el aspecto de motivación, sin centrarse en determinados elementos dentro de este aspecto. Por ejemplo, este punto de vista se puede utilizar para presentar una visión completa o parcial del aspecto motivacional, relacionando las partes interesadas, de sus objetivos principales, los principios que se aplican, y los principales requisitos de servicios, procesos, aplicaciones y objetos. En la figura 4.31 se puede observar el metamodelo propuesto en el framework Archimate.

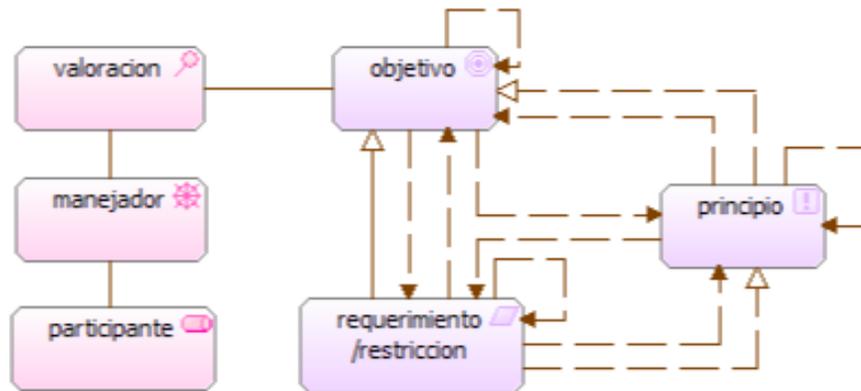


Figura 4.31: Metamodelo Punto de Vista Motivacion

4.4.3.2. Modelo

En este punto de vista planteamos como el rol interesado al residente debido a que es el que encuentra más valor en la implementación del sistema o simplemente porque son varios residentes mientras que administrador o área administrativa son de 2 a 3 personas, el residente es quien ve los principios motivacionales (Confianza, Responsabilidad, compromiso) tanto por el lado del administrador como de los otros residentes con el fin de cuidar su propiedad, adicional a esto el residente completa ciertos objetivos que tal vez no se perciben pero son importantes para el funcionamiento correcto en la estructura del conjunto residencial como por ejemplo la mayor participación en las decisiones y mayor transparencia en los procesos que es algo que debería ser importante para todos los residentes, por esto surgen ciertos requerimientos que crean la necesidad de desarrollar el sistema (Red Social) demostrando así la viabilidad del proyecto. El modelo se puede observar en Figura 4.32 de la pagina 50 con lo descrito anteriormente.

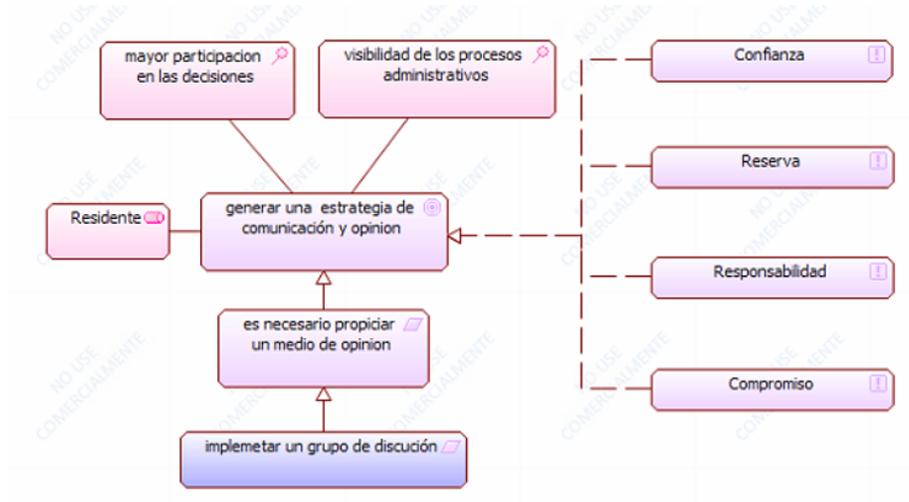


Figura 4.32: Modelo Punto de Vista Motivación

4.5. Diagrama de Arquitectura Capa de Implementación y Migración

4.5.1. Implementación y Migración

4.5.1.1. Metamodelo

El punto de vista de la implementación y la migración se utiliza para relacionar los programas y proyectos de las partes de la arquitectura que se implementan. Esta vista permite el modelado del alcance de los programas, proyectos, actividades del proyecto en términos de los elementos de la arquitectura individuales que se ven afectados. Además, la forma en que se ven afectados los elementos puede ser indicado por la anotación de las relaciones. Por otra parte, este punto de vista se puede utilizar en combinación con el punto de vista de los programas y proyectos para apoyar la gestión de la cartera, la unión de otros modelos y puntos de vista da vida a este modelo mas completo que se puede observar en la figura 4.33:

- El punto de vista de los programas y proyectos es adecuado para relacionar los objetivos de negocio a los programas y proyectos. Por ejemplo, esto hace posible el análisis a un nivel alto si todos los objetivos de negocio se cubren de manera suficiente.
- El punto de vista de la implementación y la migración es adecuado para relacionar los objetivos de negocio (y requisitos) a través de programas

y proyectos de la arquitectura. Por ejemplo, esto hace que sea posible analizar la posible superposición entre las actividades del proyecto o para analizar la coherencia entre las dependencias y las dependencias del proyecto entre los elementos de la arquitectura.

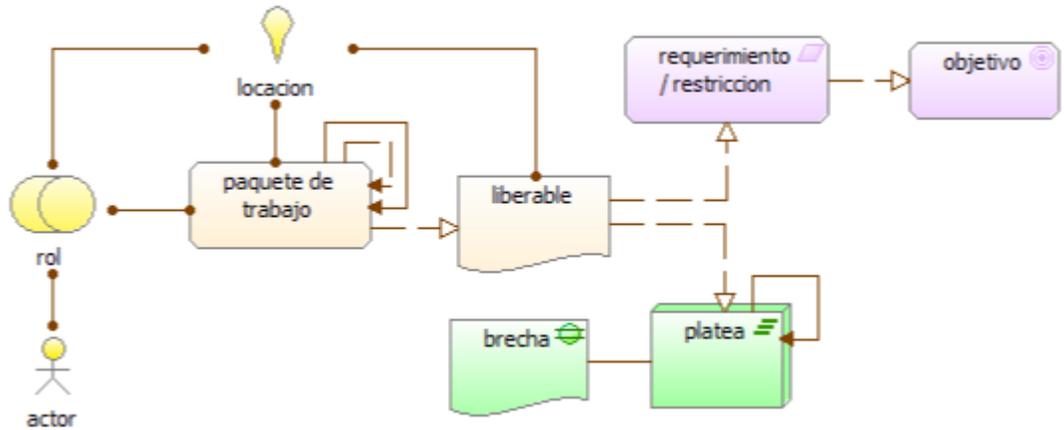


Figura 4.33: Metamodelo Punto de Vista Implementación y Migración

4.5.1.2. Modelo

En este punto de vista vemos la integración de los puntos de vista de proyecto y migración. Obtenemos un resultado de un prototipo y agregamos un esfuerzo adicional para llegar a una versión posterior de la red social, quitamos la palabra prototipo e incluyendo unos módulos específicos que hagan que el proyecto sea más completo e interesante. El modelo de proyecto se puede observar en la parte superior del modelo de la Figura 4.34 y el modelo de migración en la parte inferior de la misma, estos modelos son los componentes para el modelo de Implementación y migración.

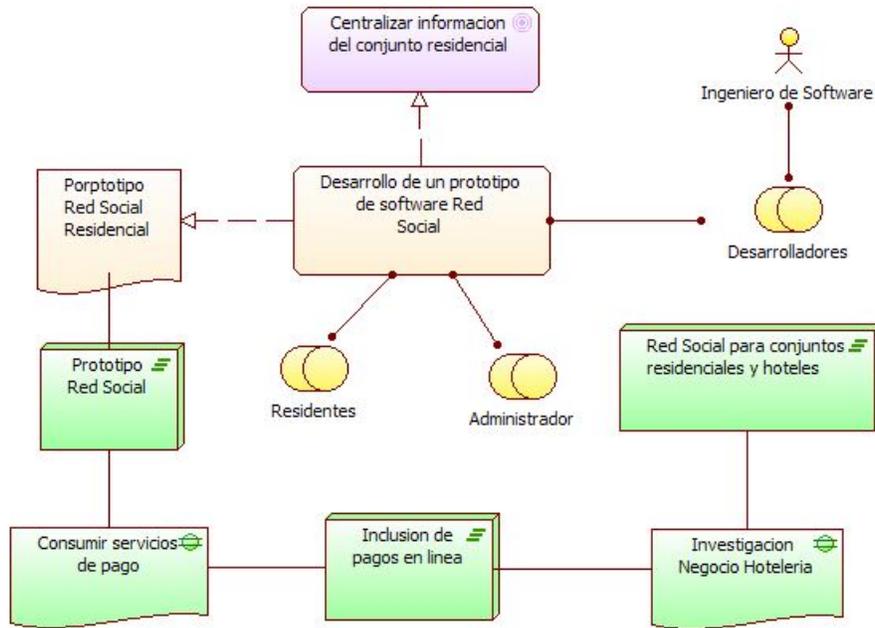


Figura 4.34: Modelo Punto de Vista Implementación y Migración

Se describe la arquitectura por medio de 17 modelos en las 5 capas propuestas por Archimate, se omitieron algunos modelos que son componente de modelos mas grandes que si se tuvieron en cuenta para el presente documento.

Capítulo 5

Metodologia

Para el desarrollo de este proyecto se planteo la metodología basada en el manifiesto ágil SCRUM. Se utiliza la metodología SCRUM como una guía y se toman algunas practicas de la metodología, pero en efecto no se implementa en todo su esquema.

5.1. Roles

Como primer limitación es un equipo de 2 personas, pero con la ventaja de tener una experiencia previa en la metodología, esto fue fundamental la seleccion de la misma dentro del proyecto, con la experiencia previa se omitió el rol de Scrum master dentro del desarrollo, los roles se plantearon de la siguiente forma

- Juan Carlos Linares: Product Owner y desarrollador Junior

En este integrante se encargo de la organización de las entregas parciales del proyecto, fue el encargado priorizar las entregas, implementar las herramientas de gestión del proyecto, definir contenido y orden de cada Sprint. Este integrante también participa en los desarrollos propuestos en cada sprint.

- Jhon Edison Nuñez: Desarrollador Senior

Este integrante era la cabeza del equipo de desarrollo, es quien se encargaba de completar las entregas que daban valor al proyecto en cada Sprint.

5.2. Sprint planing

Para plantear los Sprint se crearon historias de usuario con cada uno de los requerimientos ya priorizados por el Product Owner, Se definen 10 Sprintn y se plantean en iteraciones de 2 Semanas cada uno, en la siguiente tabla se puede observar el plan de Sprint que se propuso.

Definicion de Sprint			
ID	Historia	Sprint	Estado
1	NODE JS y MYSQL	1	Completo
2	PLANTILLA RED SOCIAL	2	Completo
3	MÓDULO LOGIN y SIGN IN	3	Completo
4	MÓDULO ADMINISTRACIÓN DE USUARIOS	4	Completo
5	MÓDULO PERFIL DE USUARIO	5	Completo
6	MÓDULO PUBLICACIONES	6	Completo
7	MÓDULO ADMINISTRACIÓN DE PROVEEDORES	7	Completo
8	MÓDULO HISTÓRICO DE PAGOS y PAGOS PENDIENTES	8	Completo
9	MÓDULO PROVEEDORES	9	Completo
10	MÓDULO CHAT y RESERVAS	10	Completo

Cuadro 5.1: Sprint Planing

5.3. Historias de usuario

Historia			
Numero	001		
Nombre	NODE JS		
Prioridad	Media	Iteracion	1
Responsable	Jhon Nuñez		
Descripcion			
<ul style="list-style-type: none"> • Instalación Node Js • Configuración Servidor • Instalación e Integración de Componentes adicionales 			

Cuadro 5.2: Historia 1

Historia			
Numero	002		
Nombre	MYSQL		
Prioridad	Media	Iteracion	1
Responsable	Juan Linares		
Descripcion			
<ul style="list-style-type: none"> • Instalación MySql. • Diseño modelo de datos. • Implementación conexión servidor Node Js – MySql. 			

Cuadro 5.3: Historia 2

Historia			
Numero	003		
Nombre	PLANTILLA RED SOCIAL		
Prioridad	Media	Iteracion	2
Responsable	Jhon Nuñez		
Descripcion			
<ul style="list-style-type: none"> • Definición Colores. • Diseño estructura menú. • Diseño componentes generales. 			

Cuadro 5.4: Historia 3

Historia			
Numero	004		
Nombre	Módulo Login		
Prioridad	Media	Iteracion	3
Responsable	Jhon Nuñez		
Descripcion			
<ul style="list-style-type: none"> • Diseño pantalla de logueo. • Implementación validaciones de login. • Desarrollo funcionalidad de acceso. • Desarrollo funcionalidad recuperación de acceso. 			

Cuadro 5.5: Historia 4

Historia			
Numero	005		
Nombre	Módulo Sign in		
Prioridad	Media	Iteracion	3
Responsable	Juan Linares		
Descripcion			
<ul style="list-style-type: none"> • Diseño pantalla de Registro de usuario. • Implementación validaciones de Registro. • Desarrollo funcionalidad de registro de usuarios. 			

Cuadro 5.6: Historia 5

Historia			
Numero	006		
Nombre	Módulo administración de usuarios		
Prioridad	Media	Iteracion	4
Responsable	Jhon Nuñez		
Descripcion			
<ul style="list-style-type: none"> • Diseño pantalla de administración. • Implementación funcionalidad de autorización de usuarios. 			

Cuadro 5.7: Historia 6

Historia			
Numero	007		
Nombre	Módulo perfil de usuario		
Prioridad	Media	Iteracion	5
Responsable	Jhon Nuñez		
Descripcion			
<ul style="list-style-type: none"> • Diseño pantalla de perfil de usuario. • Implementación opción de edición de datos de usuario. • Desarrollo publicación de texto. • Implementación cambio de imagen. 			

Cuadro 5.8: Historia 7

Historia			
Numero	008		
Nombre	Módulo publicaciones		
Prioridad	Media	Iteracion	6
Responsable	Jhon Nuñez		
Descripcion			
<ul style="list-style-type: none"> • Implementación publicación por usuario. • Desarrollo eliminación publicación por usuario. • Implementación histórico de publicaciones. 			

Cuadro 5.9: Historia 8

Historia			
Numero	009		
Nombre	Módulo administración de proveedores		
Prioridad	Media	Iteracion	7
Responsable	Jhon Nuñez		
Descripcion			
<ul style="list-style-type: none"> • Diseño pantalla administración proveedores. • Implementación funcionalidad de activación de proveedores. 			

Cuadro 5.10: Historia 9

Historia			
Numero	010		
Nombre	Módulo historio de pagos		
Prioridad	Media	Iteracion	8
Responsable	Jhon Nuñez		
Descripcion			
<ul style="list-style-type: none"> • Diseño pantalla de histórico de pagos. • Implementación consulta pagos realizados. 			

Cuadro 5.11: Historia 10

Historia			
Numero	011		
Nombre	Módulo pagos pendientes		
Prioridad	Media	Iteracion	8
Responsable	Juan Linares		
Descripcion			
<ul style="list-style-type: none"> • Diseño pantalla pagos pendientes. • Implementación consulta pagos pendientes. 			

Cuadro 5.12: Historia 11

Historia			
Numero		012	
Nombre	Módulo de proveedores		
Prioridad	Media	Iteracion	9
Responsable	Jhon Nuñez		
Descripcion			
<ul style="list-style-type: none"> • Diseño pantalla lista proveedores. • Implementación consulta proveedores visibles. 			

Cuadro 5.13: Historia 12

Historia			
Numero		013	
Nombre	Módulo de Chat		
Prioridad	Media	Iteracion	10
Responsable	Jhon Nuñez		
Descripcion			
<ul style="list-style-type: none"> • Diseño ventana lista usuarios registrados/activos. • Implementación identificación usuarios conectados/desconectados. • Diseño ventana de chat usuario. • Implementación chat usuario-usuario. • Implementación vista de histórico de conversaciones. 			

Cuadro 5.14: Historia 13

Historia			
Numero	014		
Nombre	Módulo reservas		
Prioridad	Media	Iteracion	10
Responsable	Juan Linares		
Descripcion			
<ul style="list-style-type: none"> • Diseño pantalla calendario. • Implementación registro de reservas. • Implementación modificación de registros. • Implementación opción de autorización de evento. • Implementación opción de rechazo de evento. • Implementación opción de eliminación de evento. 			

Cuadro 5.15: Historia 14

Se describen las 14 historias de usuario que se plantearon para el desarrollo del proyecto, existen otras tareas adicionales como las de definición de la interfaz de la aplicación y de toma de algunas decisiones e investigaciones previas que se hicieron antes del plan de los Sprint para evitar que no cambiaran en el futuro.

Capítulo 6

Modelo de Datos

6.1. Tecnología

Para desarrollar este proyecto se implemento un servidor EC2 de Amazon Web Services, en esta instancia se instalan las diferentes tecnologías par el funcionamiento del proyecto

Para la parte de datos del proyecto se implemento el sistema de gestión de bases de datos MySQL, es un sistema de bases de datos muy popular para desarrollos web, es utilizado en plataformas como Joomla, wordpress y Drupal, además es una herramienta Open Source con una licencia Community para desarrollar, implementar y distribuir; es fácil de instalar y fácil de administrar.

Se realizo la instalación de MySQL en el servidor de AWS, se creo un esquema para la aplicación y se procedió a implementar el modelo de datos como se puede observar en la figura 6.1.

Para establecer la conexión con la base de datos y administrar el contenido del desarrollo se utilizo la herramienta MySQL Workbench en la versión 6.3, esta aplicación nos permite administrar la base de datos y mediante el desarrollo poder validar la persistencia de la aplicación, también fue la herramienta que nos ayudo a generar el modelo de la imagen 6.1 para estructurar la aplicación y tener la información mas organizada para el desarrollo.

Si se desea observar el modelo relacional completo con los atributos de cada tabla, se puede observar por medio del siguiente link:

<https://drive.google.com/open?id=0BwZtdnnHBTuWZEhPa2I5SEZFdm8>

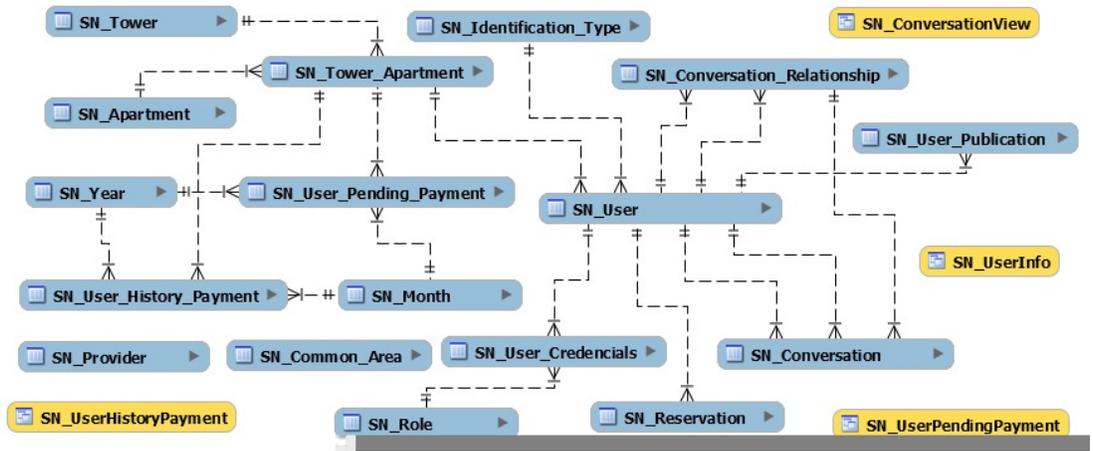


Figura 6.1: Modelo de datos

Capítulo 7

Prototipo red Social

En la instancia de AWS se realizo la instalacion de MySQL para la base de datos de la aplicacion, adicionala esto se implemento el FrameWork de Node.js para el desarrollo de la aplicacion, implementando un servidor web con Express sobre Node.js se utiliza el patron MVC para separar los conceptos y para facilitar el desarrollo y mantenimiento del mismo.

7.1. Herramientas y tecnologías

En esta sección se hace una breve descripción de las herramientas y tecnologías que se utilizaron para el desarrollo del proyecto, es importante aclarar que las herramientas descritas no eran indispensables para el desarrollo del proyecto, simplemente se acordaron por experiencias previas o por algún valor agregado al desarrollo.

Al ser un proyecto que implica desarrollo web, se entiende que se implementan tecnologías como HTML, CSS, JavaScript, al igual que el gestor de bases de datos MySQL descrito anteriormente, en esta etapa enfatizaremos en algunos frameworks y herramientas no tan generales describiendo su mas grandes ventajas en el desarrollo.

7.1.1. Putty

Para realizar la conexión con la instancia de Amazon Web Service se utilizo la herramienta esta herramienta, por medio de un protocolo SSH se establece una conexión segura con el servidor y de este modo podemos hacer la instalación del gestor de base de datos, el servidor de aplicaciones y una aplicación FTP para subir los archivos de despliegue en los directorios específicos.

7.1.2. Brackets

Es un editor de texto de código abierto desarrollado por Adobe que se encuentra disponible para múltiples plataformas como OS X, Linux y Windows.

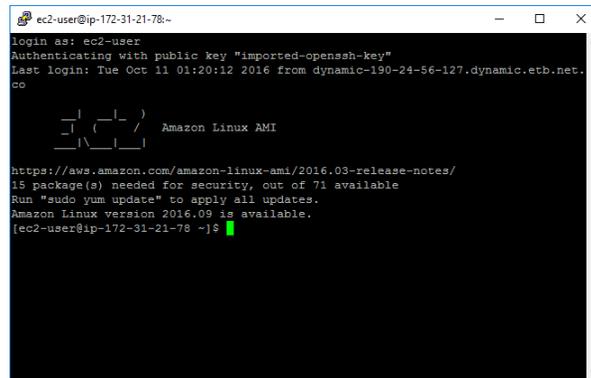


Figura 7.1: Putty con conexión SSH

Está desarrollada utilizando tecnologías web, pudiéndose ampliar sus funcionalidades a través de plugins que pueden ser instalados fácilmente a través de su gestor de paquetes. Esta posibilidad hace que se convierta en un editor de texto muy personal, ya que cada desarrollador elige las características que desea tener en su IDE sin afectar su rendimiento. Una de las características principales de este IDE es que está diseñado para desarrollo web, se encuentran varios plugins enfocados a desarrollo web y esto ayuda a agilizar el proceso de desarrollo de la plataforma de red social.

7.1.3. Node.js

Para la parte del servidor se implementó Node.js, este entorno es adecuado cuando se desea realizar varias operaciones al mismo tiempo, es utilizado en la mayoría de casos para aplicaciones en tiempo real, para nuestro caso era la mejor opción para realizar el módulo de chat y las publicaciones de los usuarios. Otra de las grandes ventajas que tiene Node.js es su gestor de paquetes NPM; existe una gran comunidad de desarrolladores que crean librerías y paquetes y las publican en el NPM, de esta forma optimizamos los tiempos de desarrollo reutilizando código desarrollado por otras personas.

La experiencia de utilizar Node.js para el desarrollo del proyecto es bastante buena, es un entorno bastante rápido, con una flexibilidad al momento de desarrollar, utilizar el lenguaje de JavaScript de el lado del servidor y del lado del cliente al mismo tiempo, cuando se tiene una experiencia en el lenguaje agiliza el desarrollo, en el proceso de desarrollo es muy rápido realizar cambios e ir probando gracias a la forma en la que se despliega la aplicación.

Esta tecnología junto a el IDE Brackets agilizaron bastante el desarrollo del proyecto, al manejar el lenguaje de JavaScript por el lado de la vista y por el lado del controlador.

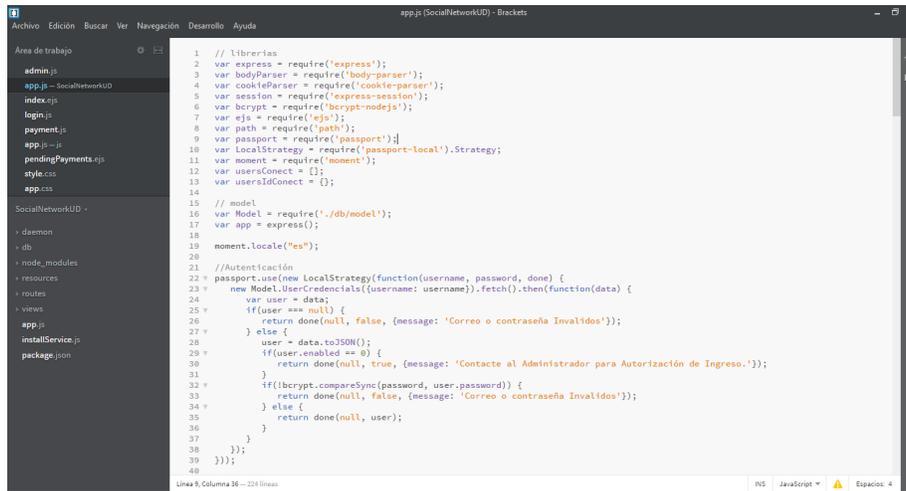


Figura 7.2: Editor de texto Brackets

7.1.4. Express.js

Express es un framework Web básico y muy flexible para Node.js (conocido como el estándar para servidores web de Node.js) que proporciona un conjunto robusto de características para aplicaciones web y móviles. Express proporciona una capa delgada de características fundamentales de aplicaciones web. Facilita la creación de APIs gracias a la gran variedad de métodos HTTP y middleware que proporciona. Utilizar express para crear el servidor web es bastante sencillo, basta con instalar el paquete y dar el comando `Express init` y él genera unos archivos necesarios para correr el servidor, solo hace falta administrar los archivos locales como estilos, páginas HTML y Scripts de Js.

7.1.5. Bookshelf.js

Es un framework que instalamos desde el gestor de paquetes de node, este framework facilita todas las operaciones CRUD utilizando el patrón de arquitectura data mapper, de esta forma podemos hacer un Mapeo de Objetos Relacionales desde la base de datos de MySQL y almacenarlos en memoria RAM.

7.1.6. Bootstrap

Es un framework que contiene plantillas de diseño con tipografías, formularios, botones, cuadros, menús de navegación y otros elementos de diseño basados en HTML y CSS. Bootstrap está orientado a facilitar el diseño y elaboración de aplicaciones web utilizando un sistema de cuadrículas, para el caso de la red social fue muy fácil crear los menús de navegación así como editar las pantallas

según los roles, este framework nos ayudó a crear las pantallas de forma rápida y al mismo tiempo a conseguir 'responsives', haciendo que las aplicaciones se vean con diferentes diseños en diferentes dispositivos, (ordenadores personales, tabletas, móviles, etc.), por lo que solo se utilizaron las clases CSS correctamente.

7.2. Módulos

7.2.1. Registro

Registro Conjunto Residencial

Tipo	# Documento	Correo Electrónico
CC ▾	<input type="text" value="Ingrese # Documento"/>	<input type="text" value="Ingrese Correo Electrónico"/>
Nombres Completos		
<input type="text" value="Ingrese Nombres"/>		<input type="text" value="Ingrese Apellidos"/>
Password		
<input type="text" value="Ingrese Contraseña"/>		<input type="text" value="Confirme Contraseña"/>
Torre-Apto	Teléfono	Perfil
1 - 102 ▾	<input type="text" value="Ingrese Teléfono"/>	Propietario ▾

[Registrarse](#)

[Ya tiene una cuenta? Ingresar!](#)

Figura 7.3: Pantalla de registro

7.2.2. Modulo de Reservas

El modulo de reservas se desarrollo basado en un calendario similar a google Calendar, este modulo permite al rol de usuario residente reservar los espacios que se encuentran habilitados, estas solicitudes quedan en un estado pendiente esperando a que el administrador los aprueba, en este calendario se puede observar cuando algunos de los espacio esta reservado y por quien esta reservado.

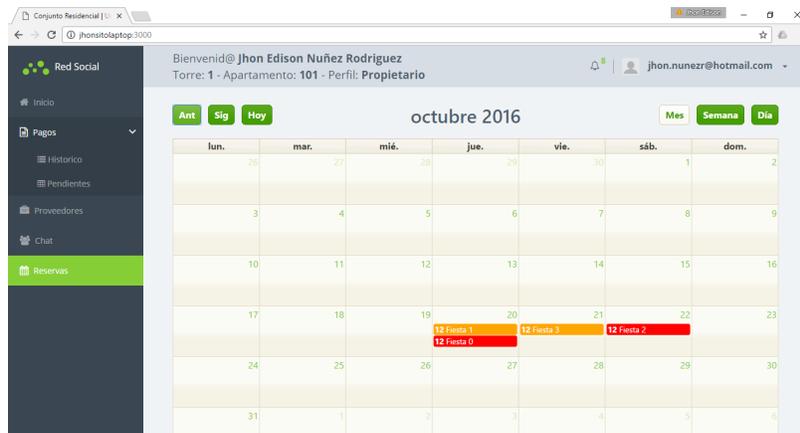


Figura 7.4: Pantalla de reservas

7.2.3. Modulo de Chat

El modulo de chat se desarrollo con sockets web en Node.js, este modulo permite a los usuarios ver que otros usuarios están conectados y establecer comunicación con ellos, el modulo de chat se desarrollo únicamente para establecer conversaciones entre 2 personas, no es posible realizar grupos de discusión

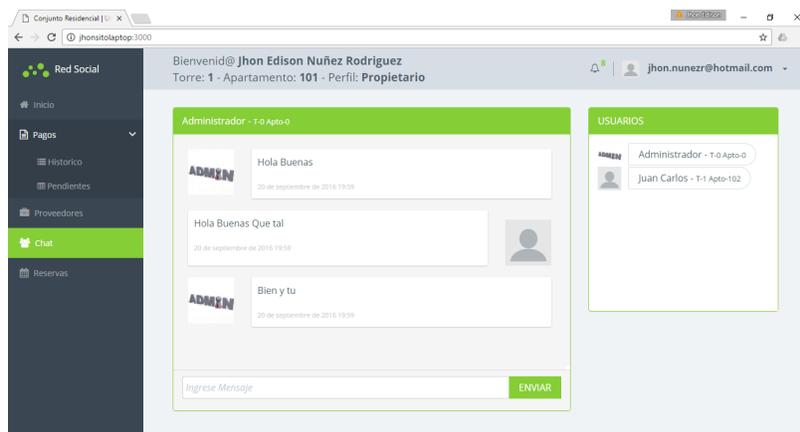


Figura 7.5: Pantalla Chat

7.2.4. Modulo de Proveedores

En este modulo solo es visualiza la información referente a los proveedores del conjunto residencial, el administrador podrá observar los proveedores que se encuentran activos así como los proveedores que en el pasado prestaron algún servicio al conjunto residencial. Por el lado de los residentes solo se podar observar los proveedores que se encuentran activos.

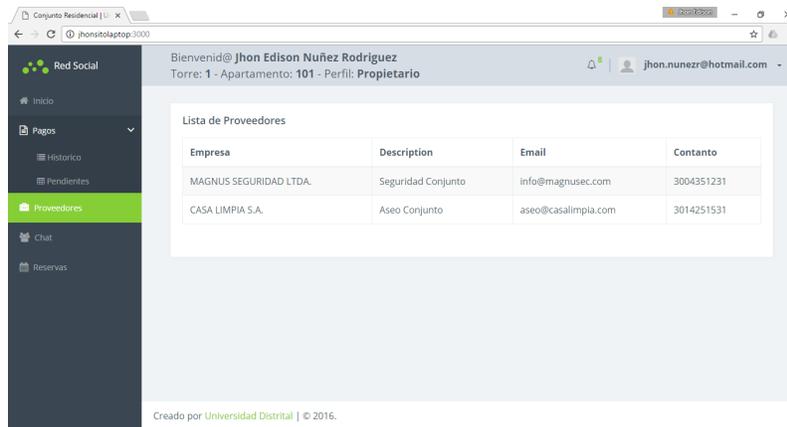


Figura 7.6: Pantalla proveedores

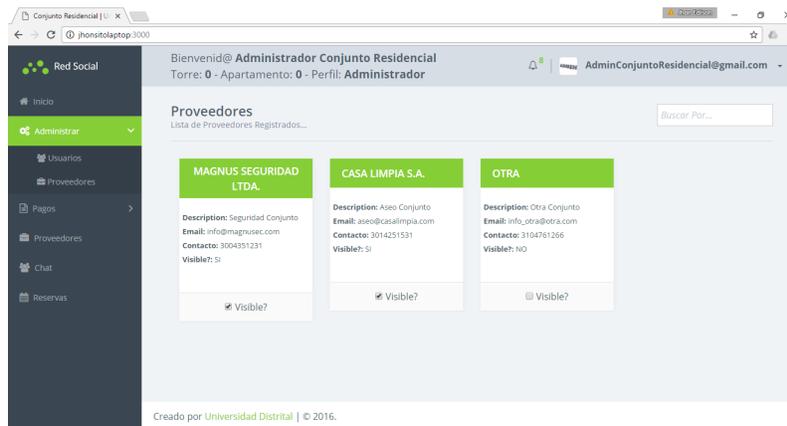


Figura 7.7: Pantalla administración de proveedores

7.2.5. Modulo de Pagos

El modulo de pagos es un sencillo modulo donde se pueden ver los pagos realizados por los residentes referente a la administración o cuotas que se presenten por parte de la administración al igual que el histórico de los mismos, este modulo es solo informativo no tiene interacción alguna.

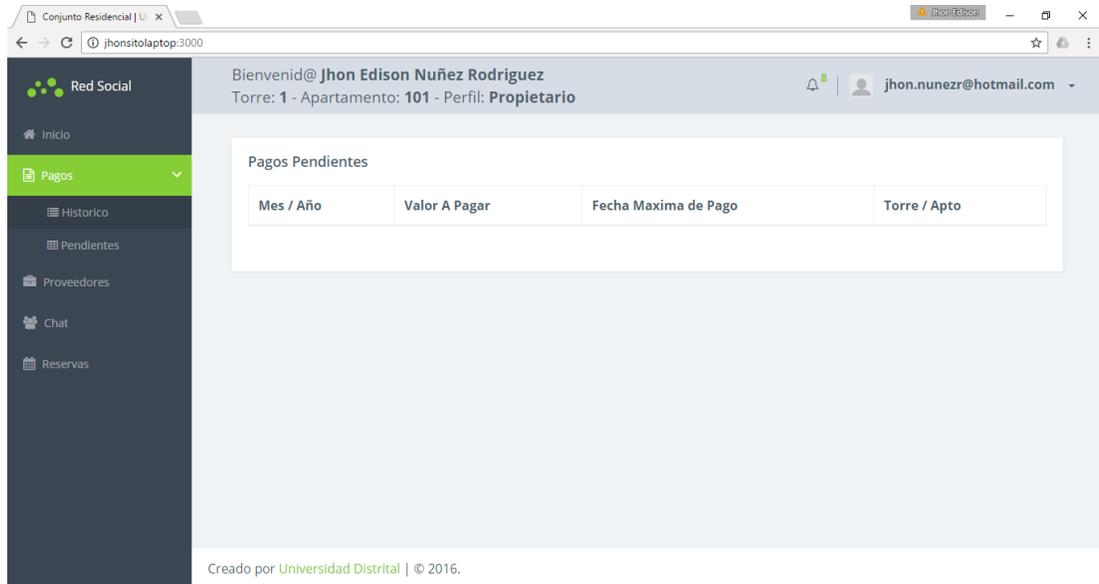


Figura 7.8: Pantalla de pagos

7.2.6. Modulo de Publicaciones

Este modulo es la pantalla principal de la red social, se basa en la pantalla donde salen todas as publicaciones que han realizado los usuarios de la red, es posible realizar publicaciones de clasificados o noticias relevantes por parte de la administración.

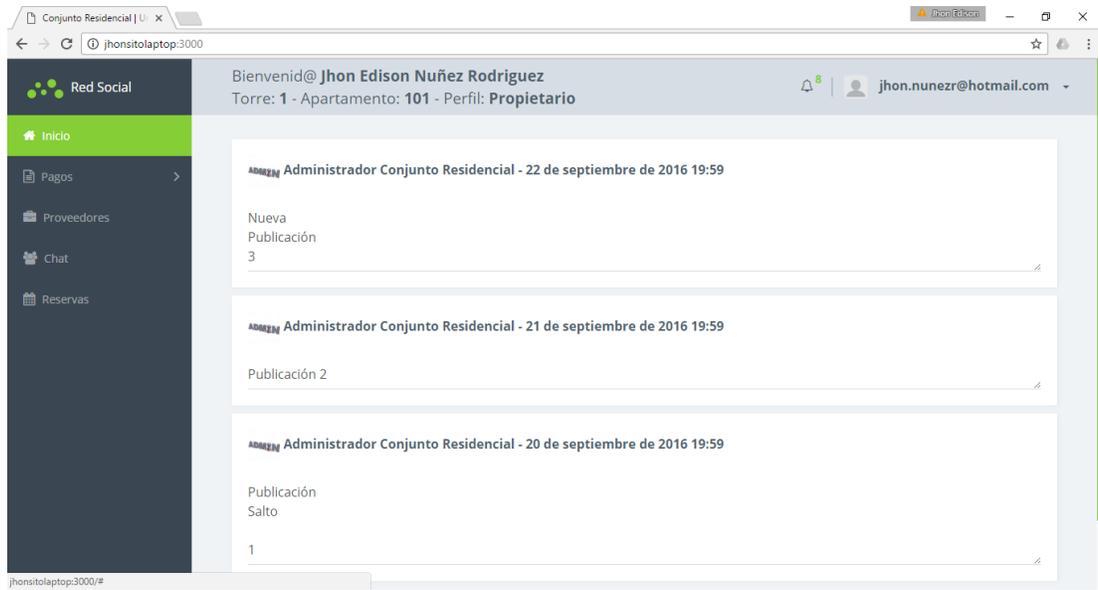


Figura 7.9: Pantalla de publicaciones

Parte IV

Cierre

Capítulo 8

Resultados y Discusión

Como resultado de la investigación y el desarrollo del proyecto se observa una aplicación web con las principales características de una red social con los módulos propuestos desde el inicio de la investigación que fueron desarrollados dentro del tiempo establecido para el proyecto guiados por medio de los principios de la metodología de desarrollo ágil SCRUM.

La solución de red social para conjuntos residenciales crea una propuesta bastante amplia para trabajos futuros basados experiencias y en las encuestas de la investigación.

La aplicación es totalmente accesible desde varios dispositivos como Smartphones, tablets, computadores de escritorio y laptops, al igual que desde los navegadores mas famosos como Google Chrome, Mozilla Firefox, Opera entre otros

La arquitectura que se realizo permite visualizar las capas de la aplicación para de esta forma facilitar la comprensión del desarrollo y poder realizar los trabajos futuros propuestos.

En cuanto la metodología de desarrollo Scrum fue bastante útil en el desarrollo proyecto, debido al escaso tiempo sirvió la implementación de una metodología ágil que definiera pequeñas tareas o requerimientos cada 2 semanas donde se entregaban componentes de un modulo o en algunos casos el modulo completo, durante todo el desarrollo del proyecto se pudo observar una parte funcional que iba creciendo cada iteración. A pesar de no coincidir con los tiempos y cronogramas que se planificaron del proyecto, el resultado fue un exitoso y los desfases de tiempo en general no sobrepasan los contemplados dentro del plan inicial.

Capítulo 9

Trabajo futuro

El proyecto se desarrollo dentro de unas limitaciones de tiempo y por esto mismo fue necesario limitar su funcionamiento, se eliminaron algunos módulos que podían ser importantes para los usuario y estos quedaran pendientes para un trabajo futuro, aparte de esto en el proceso de investigación surgen varias propuestas por parte de residentes para implementar algunos módulos que no se habían tenido en cuenta; algunos de los trabajos que se pueden hacer a futuro:

- Implementar el sistema de pagos en linea, este modulo junto con los de histórico de pago y pagos pendientes completaría la gestión para recoger la cuota de administración.
- Modulo para pedir domicilios, varios residentes estuvieron de acuerdo con pedir domicilios dentro de la misma plataforma, ver los diferentes sitios en los cuales se puede comprar comida a los alrededores del conjunto residencial.
- Integración con seguridad del conjunto residencial, poder visualizar las cámaras del conjunto residencial por medio de la aplicación de red social.
- Desarrollar una aplicación móvil para la red social, la mayoría de las persona considera que es mas fácil acceder a este tipo de plataformas por medio de una aplicación móvil y no por medio de una pagina web

Capítulo 10

Conclusiones

El uso de aplicaciones móviles es cada vez mas fuerte, la alta de demanda de servicios hace que los usuarios busquen como primera opción la aplicación del servicio que necesitan, esto indica que las aplicaciones móviles si ayudan a los usuarios a realizar sus tareas diarias, por esto se entiende que cualquier plataforma web como la red social debe estar acompañada de su aplicación móvil.

El uso de una arquitectura de software, en este caso desarrollada con el framework Archimate, permite comprender con su propuesta los Stakeholders del negocio, la estructura de su organización a nivel de sistemas de información, el detalle de la infraestructura que se implementa y facilita además la identificación y definición de requisitos y requerimientos y la forma en que se van a abordar desde diferentes puntos de vista

ArchiMate como otros frameworks es una guía para modelar la arquitectura de la aplicación o de la empresa en general, mas sin embargo es necesario conocer el enfoque que se tiene y las tecnologías que se van a implementar y así mismo nos ayuda con la identificación de procesos, roles y otros componentes que no se perciben desde el inicio

En el caso puntual de la red social se aborda la necesidad para el procesamiento con los puntos de vista diseñados, se evidencia la viabilidad en la realización del proyecto, pues además de resolver la necesidad de centralizar la información y exponer cada proceso se puede observar que el proyecto es escalable a cualquier otra entidad que desee administrar algún recurso.

La red social es aceptada por un grupo de ingenieros dentro de la Universidad Distrital así como algunos docentes y las personas que hicieron parte de la fuente de información, de cada presentación siguieron nuevas ideas para ser implementadas a futuro y sirviendo de inicio para proyectos de mas alto nivel.

Anexos

Anexos A

Anexo I: Encuesta

- 1 ¿Cuántas personas viven en su apartamento?
 - a) 1
 - b) 2
 - c) 3
 - d) mas de 3
- 2 ¿Hacen uso de las redes sociales o portales web?
 - a) Si ¿Indique algunos?
 - b) No ¿Porque?
- 3 ¿Cómo es su relación con el administrador del conjunto residencial?
 - a) Excelente
 - b) Buena
 - c) Regular
 - d) Mala
- 4 ¿Con qué frecuencia acude a las asambleas del conjunto residencial?
 - a) Siempre
 - b) En la mayoría
 - c) En muy pocas
 - d) Nunca ¿Porque?
- 5 ¿Considera usted que la información referente al conjunto residencial es transparente para todos los residentes?
 - a) Si
 - b) No ¿Porque?

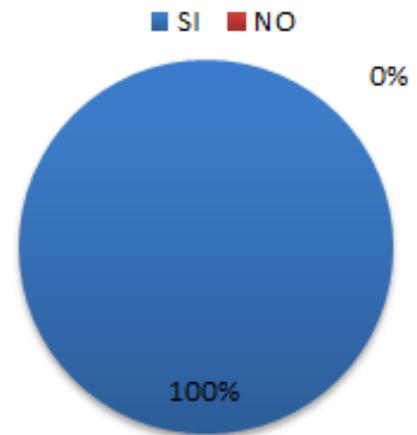
- 6 ¿Considera usted que los procesos administrativos del conjunto residencial si se están llevando de forma adecuada?
- a) Si
 - b) No
 - c) No sabe
- 7 ¿Con qué frecuencia necesita comunicarse con el administrador?
- a) Más de 1 vez al mes
 - b) 1 vez cada 3 meses
 - c) Muy rara vez
 - d) Nunca
- 8 ¿Que tipo de procesos conoce usted que se manejan dentro del conjunto residencial?
- 9 ¿Considera usted que hace falta una herramienta que centralice todos los procesos del conjunto residencial?
- a) Si
 - b) No
- 10 ¿Qué procesos adicionales le gustaría tener dentro de su conjunto residencial?

Anexos B

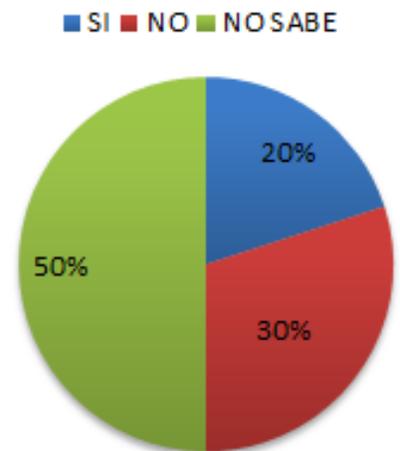
Anexo II: Tabulación de la Información



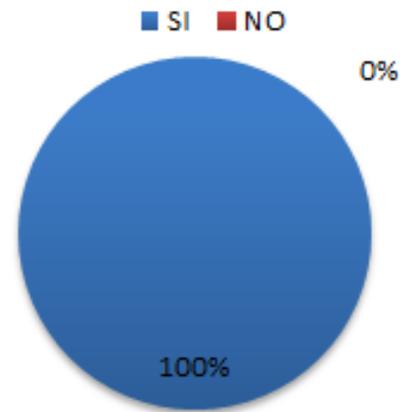
¿Considera usted que la información referente al conjunto residencial es transparente para todos los residentes?



¿Considera usted que los procesos administrativos del conjunto residencial si se están llevando



¿Considera usted que hace falta una herramienta que centralice los procesos del conjunto residencial?



Referencias

- Cordero, M. G. O. (2013). Evaluación de frameworks. pages 1–195.
- de Cordoba, U. (2015). Diseño de encuestas. pages 1–33.
- DocPlayer.es (2016). Web sockets. url<http://docplayer.es/docs-images/27/12009908/images/21-0.png>.
- EIRL, B. (2010-2015). Recolección de datos. url<http://seminariosdeinvestigacion.com/la-recoleccion-de-datos/>.
- Heyl, B. H. (2014). Literature survey on business model-based integration of business and process architecture. In *2014 33rd International Conference of the Chilean Computer Science Society (SCCC)*, pages 7–11.
- Kumar, S. S., Kumar, K. S., and Kayarvizhy, N. (2016). Analysis of information propagation in academic social networks. In *2016 International Conference on Recent Trends in Information Technology (ICRTIT)*, pages 1–4.
- Narkwilai, M., Funilkul, S., and Supasitthimethee, U. (2015). Factors influencing the thai elderly’s intention to use social network for quality of life: A case study line application. In *2015 7th International Conference on Information Technology and Electrical Engineering (ICITEE)*, pages 593–598.
- Network, M. D. (2005-2016a). Html. url <https://developer.mozilla.org/es/docs/Web/HTML>.
- Network, M. D. (2005-2016b). Javascript. url <https://developer.mozilla.org/es/docs/Web/JavaScript/Guide/Introduccion>.
- Network, M. D. (2005-2016c). Websockets. url<https://developer.mozilla.org/es/docs/WebSockets-840092-dup>.
- Pete Deemer, Gabrielle Benefield, C. L. B. V. (2009). Información básica de scrum (the scrum primer). page 20.
- Q-Success (2009-2016). JQuery. url<https://w3techs.com/technologies/details/js-jquery/all/all>.
- rmunoz (2011). Node js. url<http://www.rmuno.net/introduccion-a-node-js.html>.
- Sauer, C. and Willcocks, L. (2004). Strategic alignment revisited: connecting organizational architecture and it infrastructure. In *System Sciences, 2004. Proceedings of the 37th Annual Hawaii International Conference on*, pages 10 pp.–.
- Tilkov, S. and Vinoski, S. (2010). Node.js: Using javascript to build high-performance network programs. *IEEE Internet Computing*, 14(6):80–83.

UNAD (2010-2015a). La elaboración de preguntas para entrevistas y cuestionarios "la inteligencia consiste no sólo en el conocimiento. [urlhttp://datateca.unad.edu.co/contenidos/100104/100104EXE/leccion_9_la_elaboracion_de_preguntas_para_entrevistas_y_cuestionarios.html](http://datateca.unad.edu.co/contenidos/100104/100104EXE/leccion_9_la_elaboracion_de_preguntas_para_entrevistas_y_cuestionarios.html).

UNAD (2015b). Resultados de la investigación. pages 1–8.