**INGENIERIA DEL SOFTWARE I**

**ANDRES FELIPE GIRALDO ARISTIZÁBAL**

**CAMILO QUINTERO**

**JHORSS KERRY CORTES**

**DOCENTE**

**FEIBERT ALIRIO GUZMÁN**

**TRABAJO ESCRITO ACERCA DEL PROYECTO**

**APLICACIÓN WEB “INMOCONNECT”**

# CORPORACIÓN UNIVERSITARIA LASALLISTA

**MEDELLÍN 2024-2**

**Tabla de contenido**

[**Introducción** 7](#_Toc180603282)

[**Resumen del proyecto** 8](#_Toc180603283)

[**Planteamiento del problema** 9](#_Toc180603284)

[**Objetivos** 10](#_Toc180603285)

[**Marco Teórico** 11](#_Toc180603286)

[**Marco metodológico** 14](#_Toc180603287)

[**Resultados** 16](#_Toc180603288)

[**Conclusiones** 19](#_Toc180603289)

[**Recomendaciones** 20](#_Toc180603290)

[**Bibliografía** 21](#_Toc180603291)

# 

# Lista de Tablas

Tabla 1: Glosario. ...........................................................................................................................................

# Lista de Figuras

[Figura 1 16](#_Toc180605046)

[Figura 2 18](#_Toc180605047)

[Vista 1 17](#_Toc180605012)

# Glosario

***Tabla 1: Glosario.***

|  |  |  |
| --- | --- | --- |
| **PALABRA CLAVE** | **DESCRIPCIÓN** | **CITA** |
| **Inmobiliaria Digital** | Plataforma en línea que permite la compra, venta o alquiler de propiedades, facilitando la interacción entre usuarios sin necesidad de intermediarios. | Fuente: Marketing Directo. (2023). "La digitalización del sector inmobiliario". |
| **Publicación de Inmuebles** | Proceso mediante el cual un usuario sube información sobre una propiedad a la plataforma, haciéndola visible a otros usuarios interesados. | Fuente: Idealista. (2023). "Cómo publicar un anuncio inmobiliario". |
| **Filtros Avanzados** | Herramientas que permiten a los usuarios refinar sus búsquedas en la plataforma, seleccionando criterios como precio, ubicación, tipo de propiedad, etc. | Fuente: Zillow. (2024). "Search with advanced filters to find your dream home". |
| **Conexión Directa** | Funcionalidad que permite que compradores y vendedores interactúen directamente, eliminando la necesidad de intermediarios. | Fuente: Forbes. (2023). "The evolution of direct buyer-seller interaction in real estate". |
| **Angular** | Framework de desarrollo web basado en TypeScript, utilizado para construir interfaces interactivas en aplicaciones de una sola página (SPA). | Fuente: Angular Official Documentation. (2024). "Overview of Angular framework". |
| **Node.js** | Entorno de ejecución de JavaScript en el servidor que facilita el desarrollo de aplicaciones escalables y de alto rendimiento. | Fuente: Node.js Foundation. (2024). "Introduction to Node.js". |
| **MongoDB** | Base de datos NoSQL que almacena datos en formato JSON, proporcionando flexibilidad y escalabilidad a la plataforma. | Fuente: MongoDB Official Documentation. (2024). "What is MongoDB?". |
| **Autenticación de Usuarios** | Proceso de verificación de identidad de los usuarios mediante credenciales (usuario y contraseña) antes de permitirles acceder a la plataforma. | Fuente: Okta. (2023). "Authentication and authorization explained". |
| **Gestión de Propiedades** | Funcionalidad que permite a los usuarios administrar sus inmuebles dentro de la plataforma, pudiendo modificar información, ver estadísticas, entre otros. | Fuente: Realtor. (2023). "Property management in the digital age". |
| **Transacciones** | Registro de las operaciones de compra, venta o alquiler que se realizan dentro de la plataforma, documentando el intercambio entre usuarios. | Fuente: Investopedia. (2023). "How real estate transactions work". |

**\* Fuente:** *Elaboración propia con base en las citas suministradas.*

1**Introducción**

InmoConnect es una plataforma web diseñada para facilitar la gestión y comercialización de propiedades inmobiliarias, abarcando desde terrenos hasta casas y apartamentos. A través de su interfaz intuitiva, los usuarios pueden crear perfiles personales y realizar publicaciones de inmuebles, permitiendo una conexión eficiente entre compradores y vendedores. Esta solución digital no solo centraliza el proceso de compra y venta de propiedades, sino que también optimiza la experiencia mediante el uso de filtros avanzados que permiten al usuario encontrar propiedades según sus preferencias.

Desarrollada con tecnologías modernas como Angular, Node.js y MongoDB, InmoConnect ofrece un ecosistema robusto y seguro para el mercado inmobiliario, destacándose por su capacidad de escalar y su alto rendimiento, gracias a la arquitectura basada en una base de datos no relacional.

**2** **Resumen del proyecto**

El proyecto InmoConnect se centra en el desarrollo de una plataforma digital que funciona como un mercado inmobiliario interactivo. Los usuarios pueden registrarse, crear publicaciones de inmuebles y visualizar aquellas de otros usuarios. Adicionalmente, la plataforma permite a los usuarios aplicar filtros detallados para encontrar inmuebles que se ajusten a sus necesidades. InmoConnect tiene como objetivo simplificar la compra y venta de propiedades mediante la conexión directa entre compradores y vendedores, ofreciendo herramientas integradas de contacto y gestión de publicaciones. La plataforma está desarrollada utilizando tecnologías de última generación como Angular para el frontend, Node.js para el backend, y MongoDB como base de datos, garantizando un rendimiento eficiente y escalabilidad.

### Palabras clave:

Inmobiliaria digital, Compra y venta de inmuebles, Plataforma web, Publicación de inmuebles, Filtros avanzados, Conexión directa, Angular, Node.js, MongoDB, Gestión de propiedades

**3** **Planteamiento del problema**

El mercado inmobiliario tradicional enfrenta múltiples desafíos en términos de eficiencia y accesibilidad. Los compradores y vendedores a menudo dependen de intermediarios que ralentizan el proceso de negociación, lo cual limita el acceso directo a la información y a las oportunidades en tiempo real. Además, los métodos convencionales de búsqueda de propiedades suelen ser ineficaces, ya que no permiten una personalización adecuada basada en las necesidades de los usuarios. Este proyecto busca abordar estos problemas creando una plataforma digital que centralice y agilice la compra y venta de inmuebles, permitiendo que los usuarios interactúen sin intermediarios y filtren propiedades de acuerdo con criterios específicos, mejorando así la experiencia de usuario.

### 3.1 Pregunta problematizadora

¿Cómo se puede optimizar la compra y venta de inmuebles mediante una plataforma digital que elimine intermediarios y permita a los usuarios realizar búsquedas personalizadas y contactar directamente con vendedores en tiempo real?

**4** **Objetivos**

### 4.1 Objetivo general

Desarrollar una plataforma web eficiente y escalable que permita a los usuarios realizar publicaciones de inmuebles, conectarse directamente con compradores o vendedores, y gestionar todas las interacciones dentro del mercado inmobiliario de manera autónoma.

### 4.2 Objetivos específicos

**4.2.1** Crear un sistema de registro y autenticación de usuarios que permita la creación de perfiles personalizados.

**4.2.2** Implementar funcionalidades de publicación y visualización de inmuebles, con filtros avanzados de búsqueda.

**4.2.3** Desarrollar una interfaz intuitiva que facilite la navegación y gestión de publicaciones.

**4.2.4** Utilizar tecnologías modernas como Angular, Node.js y MongoDB para garantizar un desarrollo ágil y una experiencia de usuario fluida.

**4.2.5** Incorporar un sistema de contacto que permita a los usuarios comunicarse directamente, sin la necesidad de intermediarios.

**5** **Marco Teórico**

El marco teórico de InmoConnect proporciona la base conceptual y las referencias necesarias para entender la importancia de la digitalización del mercado inmobiliario, la gestión eficiente de propiedades, y la implementación de tecnologías web modernas. InmoConnect es una plataforma diseñada para optimizar los procesos de compra, venta y alquiler de bienes inmuebles, ofreciendo a los usuarios una experiencia interactiva y segura.

Este marco teórico abarca tanto los fundamentos del comercio electrónico en el sector inmobiliario como los principios de diseño y desarrollo web. Además, se analizan las tecnologías utilizadas en el desarrollo del proyecto, como Angular, Node.js y MongoDB, y su rol en la creación de una plataforma escalable y flexible. También se exploran las metodologías ágiles, como Scrum, que permiten una gestión eficiente del equipo de desarrollo, asegurando la entrega continua de un producto funcional.

###### 5.1 Comercio Electrónico en el Sector Inmobiliario

El comercio electrónico ha revolucionado diversos sectores, incluido el inmobiliario, permitiendo que las operaciones de compra, venta y alquiler de propiedades se realicen de manera más eficiente y accesible. Las plataformas inmobiliarias online permiten a los usuarios explorar, comparar y tomar decisiones informadas desde cualquier lugar, optimizando el proceso para vendedores y compradores.

###### 5.2 Desarrollo Web: Tecnologías Utilizadas

El desarrollo de una página web inmobiliaria requiere de un enfoque multidisciplinario que cubra tanto el frontend como el backend, garantizando una experiencia de usuario eficiente y una gestión sólida de los datos.

5.2.1 Frontend

El frontend se encarga de la interfaz de usuario (UI) y de la experiencia de usuario (UX), es decir, lo que los usuarios ven y cómo interactúan con la plataforma. Para este proyecto se han utilizado las siguientes tecnologías:

**5.2.1.1 HTML (Hypertext Markup Language)**: Es el lenguaje estándar para la estructura de las páginas web. Define la organización de los elementos visuales como textos, imágenes, botones, etc.

**5.2.1.2 CSS (Cascading Style Sheets)**: Permite dar estilo a las páginas web, controlando la apariencia visual, como colores, fuentes y distribución de los elementos en la pantalla. Se utiliza para crear diseños responsivos y atractivos.

**5.2.1.3 TypeScript**: Es un superconjunto de JavaScript que añade tipadoestático. Aporta seguridad al desarrollo de aplicaciones al detectar errores en tiempo de compilación, facilitando la escalabilidad del proyecto.

**5.2.1.4 Angular**: Es un framework basado en TypeScript, ideal para el desarrollo de aplicaciones web de una sola página (SPA). Angular facilita la creación de interfaces de usuario interactivas, al proporcionar un patrón de arquitectura MVC (Modelo-VistaControlador) y funcionalidades como el data**-**binding (enlace de datos), lo que mejora la eficiencia en la actualización de vistas y datos.

5.2.2 Backend

El backend se encarga de la lógica del servidor, el manejo de las bases de datos y la autenticación de usuarios, asegurando que las solicitudes del cliente se procesen adecuadamente y que la información esté protegida.

**5.2.2.1 Node.js**: Un entorno de ejecución de JavaScript en el servidor que permite manejar grandes volúmenes de solicitudes de manera eficiente, gracias a su arquitectura basada en eventos y su naturaleza asincrónica.

**5.2.2.2 Express**: Un framework ligero para Node.js, que facilita la creación de aplicaciones web y APIs mediante un conjunto de herramientas que simplifican el manejo de rutas y middleware.

**5.2.2.3 Body-parser**: Middleware de Express utilizado para procesar datos entrantes en formato JSON y URL encoded, lo cual es fundamental para manejar solicitudes HTTP POST.

**5.2.2.4 Mongoose**: Una biblioteca de Node.js que facilita la interacción con bases de datos MongoDB, permitiendo definir esquemas de datos y realizar consultas de manera más estructurada.

**5.2.2.5 Jwt-simple**: Herramienta para gestionar tokens JWT (JSON Web Tokens), usados para autenticar usuarios de forma segura en la plataforma.

**5.2.2.6 Bcrypt**: Una biblioteca para encriptar contraseñas, asegurando que los datos sensibles de los usuarios, como las credenciales de acceso, se almacenen de forma segura.

**5.2.2.7 Moment**: Biblioteca utilizada para el manejo de fechas y tiempos en JavaScript, facilitando el procesamiento y formateo de datos temporales, cruciales en una plataforma inmobiliaria para gestionar fechas de disponibilidad, contratos, etc.

**5.2.2.8 MongoDB**: Base de datos NoSQL que se caracteriza por almacenar los datos en formato de documento JSON, lo que le otorga flexibilidad para manejar grandes volúmenes de información no estructurada o semiestructurada, como las propiedades y sus descripciones.

###### 5.3 Herramientas de Desarrollo

El uso de herramientas adecuadas puede acelerar y facilitar el proceso de desarrollo, mejorando la eficiencia del equipo.

**5.3.1 Visual Studio Code (VS Code)**: Es un entorno de desarrollo integrado (IDE) ligero y poderoso, ampliamente utilizado en el desarrollo de aplicaciones web. Ofrece soporte para TypeScript, JavaScript, Angular, Node.js y otras tecnologías a través de extensiones, lo que lo convierte en una opción flexible para desarrolladores.

**5.3.2 Cursor**: Es una herramienta de colaboración que permite a los desarrolladores escribir y revisar código en tiempo real, lo que facilita el trabajo en equipo y la solución rápida de problemas.

###### 5.4 Metodologías Ágiles: Scrum

Para gestionar el desarrollo del proyecto, se ha utilizado la metodología ágil Scrum, la cual está diseñada para trabajar en equipos pequeños, incrementando la productividad y entregando productos funcionales en ciclos cortos llamados sprints. Se basa en la iteración continua, permitiendo realizar entregas parciales del producto y adaptarse a los cambios en los requisitos. Esto es especialmente útil en proyectos de desarrollo web, donde los objetivos pueden ajustarse rápidamente según las necesidades del cliente o del mercado.

Para la gestión del proyecto se utilizó la herramienta “Trello” donde las tareas pueden ser organizadas visualmente en tarjetas. Trello permite una gestión clara y sencilla de los sprints, tareas y prioridades del equipo, facilitando la colaboración y el seguimiento del progreso.

**6** **Marco metodológico**

El desarrollo de una base de datos eficiente es crucial para la gestión de información dentro de una organización. En este contexto, el proceso metodológico adoptado para el proyecto de *InmoConnect* sigue un enfoque estructurado que garantiza la correcta identificación de requisitos, diseño del modelo, implementación y verificación. Las fases se desarrollan de la siguiente manera:

**6.1 Análisis de Requisitos:** En esta fase, se realizó una identificación detallada de las necesidades de la empresa (InmoConnect) para asegurar que la base de datos cubra todos los aspectos necesarios para la gestión inmobiliaria. El objetivo principal es permitir la publicación, búsqueda y gestión de inmuebles por parte de los usuarios, integrando funcionalidades clave como filtros avanzados y autenticación. Durante este análisis, se identifican también los actores involucrados, los flujos de datos y las operaciones que deben ser soportadas por el sistema, como la gestión de perfiles de usuario, inmuebles y transacciones.

**6.2 Diseño del Modelo ER:** A diferencia de las bases de datos relacionales tradicionales (como MySQL), en este proyecto se utiliza MongoDB, una base de datos no relacional (NoSQL). MongoDB almacena los datos en formato JSON, lo que permite una mayor flexibilidad en la estructura de las colecciones.

El diseño del modelo de datos se basa en la definición de las principales colecciones y sus relaciones. Algunas de las colecciones clave incluyen:

6.2.1 **Usuarios:** Esta colección almacena la información de los usuarios registrados, incluyendo su perfil personal, credenciales de acceso, y sus preferencias.

6.2.2 **Inmuebles:** Esta colección contiene los datos de las propiedades listadas en la plataforma, tales como la descripción, ubicación, precio, y características del inmueble.

6.2.3 **Transacciones:** Almacena el historial de transacciones entre usuarios, como ventas, alquileres o contactos realizados a través de la plataforma.

**6.3 Implementación en MongoDB:** La implementación de la base de datos en

MongoDB sigue el modelo entidad-relación diseñado en la fase anterior. Se utiliza Mongoose, una biblioteca para Node.js, que permite la creación de esquemas y la interacción con MongoDB de manera eficiente. Las colecciones y los documentos se crean basados en los esquemas definidos previamente.

Tecnologías clave utilizadas:

6.3.1 **MongoDB:** Para el almacenamiento de datos no relacional.

6.3.2 **Mongoose:** Para la interacción entre la aplicación y MongoDB.

6.3.3 **Express:** Para el manejo de las rutas y lógica del servidor.

6.3.4 **bcrypt:** Para la encriptación de contraseñas de los usuarios.

6.3.5 **jwt-simple:** Para la autenticación y manejo de sesiones a través de tokens JWT.

**6.4 Pruebas y Validación:** Con la base de datos implementada, se procede a la fase de pruebas. Aquí se ejecutan diferentes casos de prueba para verificar que la base de datos funcione de manera correcta y eficiente. Se verifican aspectos como la integridad de los datos, la ejecución adecuada de las consultas, y el rendimiento del sistema bajo diferentes escenarios. Además, se evalúa la capacidad de la base de datos para soportar el crecimiento de usuarios y transacciones, asegurando su escalabilidad.

**6.5 Despliegue y Documentación:** En la última fase, la base de datos se despliega en el entorno de producción, permitiendo su uso por parte de los usuarios de

InmoConnect. Además, se elabora la documentación correspondiente, que incluye tanto la descripción del diseño como las guías de uso y mantenimiento de la base de datos. Esta documentación es crucial para garantizar que el sistema pueda ser mantenido y actualizado en el futuro.

**7** **Resultados**

La presente Figura 1 muestra las carpetas necesarias para la creación del backend donde se clasificarán sus archivos de acuerdo con su funcionalidad.

***Figura 1: Carpetas necesarias para el proyecto***

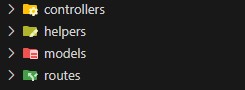
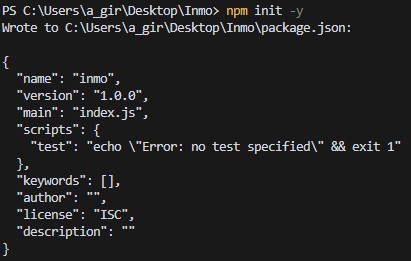


Figura 1

**Fuente:** Elaboración propia

A continuación, debemos iniciar el proyecto con NodeJs para contar con un entorno de desarrollo.

***Fase 1:*** Iniciar proyecto de NodeJs



Vista 1

**Vista 1:** Permite observar las características del proyecto iniciado mediante NodeJs

##### Figura 2: Agregar dependencias dentro de package.json



Figura 2

**Fuente:** Elaboración propia en Cursor con NodeJs

En la **Figura 2** se evidencia las dependencias anexadas en el proyecto, de las cuales haremos uso para lograr los resultados esperados.

***Fase 2:*** Diseño de modelos para las entidades

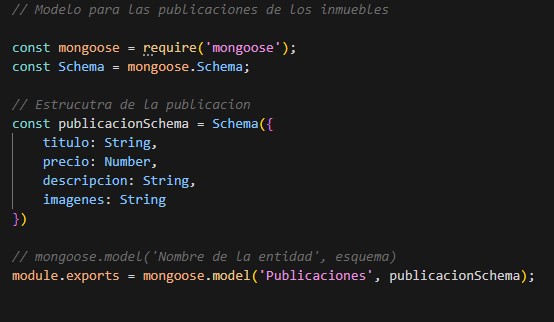


Ilustración 4

Todos los resultados se pueden visualizar y evidenciar en el GIthub.

https://github.com/DeveloperXPK/InmoConnectX

**8** **Conclusiones**

La digitalización eficiente del mercado inmobiliario fue posible gracias a la creación de una plataforma que optimiza el proceso de compra y venta de inmuebles, eliminando intermediarios y facilitando la interacción directa entre compradores y vendedores.

Se implementaron tecnologías modernas como Angular, Node.js y MongoDB, lo que permitió el desarrollo de un sistema escalable, eficiente y seguro, capaz de manejar grandes volúmenes de datos y crecer junto con la base de usuarios.

La experiencia del usuario se mejoró significativamente mediante la incorporación de filtros avanzados y una interfaz intuitiva, facilitando las búsquedas personalizadas y la navegación por la plataforma.

La automatización y la autonomía proporcionadas a los usuarios les permiten gestionar sus publicaciones e interacciones sin necesidad de intermediarios, lo que acelera los procesos y aumenta la satisfacción del usuario.

La adaptabilidad y escalabilidad del sistema aseguran que la plataforma podrá soportar un alto volumen de transacciones y usuarios, manteniendo su funcionalidad en un entorno en constante cambio.

**9** **Recomendaciones**

9.1. Implementar un sistema de alertas para notificar a los usuarios sobre propiedades que coincidan con sus criterios de búsqueda o que presenten cambios importantes en sus características, como reducciones de precio o nuevas ofertas.

9.2. Realizar auditorías periódicas en la base de datos de propiedades para asegurar la calidad y precisión de la información registrada, evitando publicaciones duplicadas o incorrectas.

9.3. Capacitar a los usuarios en el uso de la plataforma mediante tutoriales o guías interactivas, para que puedan aprovechar al máximo las funcionalidades de publicación, búsqueda y gestión de propiedades.

9.4. Incorporar herramientas de análisis y estadísticas que permitan a los usuarios visualizar el rendimiento de sus publicaciones, así como datos sobre el mercado inmobiliario local para facilitar la toma de decisiones.

9.5. Continuar mejorando la interfaz de usuario para hacerla más accesible y fácil de usar, basándose en la retroalimentación de los usuarios y las mejores prácticas de diseño de experiencia de usuario (UX).

9.6. Mejorar la usabilidad móvil: Se recomienda realizar pruebas adicionales en dispositivos móviles para asegurar una experiencia fluida en pantallas más pequeñas, mejorando la compatibilidad y el diseño responsivo.

**10** **Bibliografía**

Marketing Directo. (2023). La digitalización del sector inmobiliario. \*Marketing Directo\*. https://www.marketingdirecto.com

Idealista. (2023). Cómo publicar un anuncio inmobiliario. \*Idealista\*. https://www.idealista.com

Zillow. (2024). Search with advanced filters to find your dream home. \*Zillow\*. https://www.zillow.com

Forbes. (2023). The evolution of direct buyer-seller interaction in real estate. \*Forbes\*. https://www.forbes.com

Angular Official Documentation. (2024). Overview of Angular framework. \*Angular\*. https://angular.io

Node.js Foundation. (2024). Introduction to Node.js. \*Node.js\*. https://nodejs.org

MongoDB Official Documentation. (2024). What is MongoDB? \*MongoDB\*. https://www.mongodb.com

Okta. (2023). Authentication and authorization explained. \*Okta\*. https://www.okta.com

Realtor. (2023). Property management in the digital age. \*Realtor\*. https://www.realtor.com

Investopedia. (2023). How real estate transactions work. \*Investopedia\*. https://www.investopedia.com