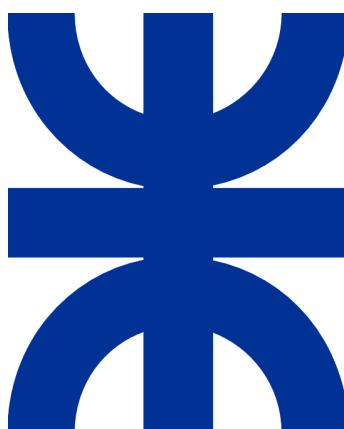




## **Proyecto Final 2026**

### **Domus**

Tema: Plataforma de contratación de servicios para el hogar

Comisión: 5K3

Organización/Cliente: No aplica

Profesor adjunto: Ing. Quinteros, Sergio

Tutor: Ing. Torres Hansen, Tomás

Profesores de la comisión:

Ing. Santos, Virginia

Ing. Trettel, Marta Cecilia

Ing. Destefanis, María Laura

Ing. Martínez, Gabriel

Ing. Parisi, Germán Nicolás

Integrantes:

- 95457 - Moreno, Matías Emanuel
- 79542 - Nieto Poklepovic, Federico Alejandro
- 66456 - Revol, Lisandro
- 79956 - Valdemarca Fragueiro, Dante



DOMUS CONECTA A CLIENTES CON PROFESIONALES CALIFICADOS DE FORMA RÁPIDA Y CONFIABLE. AYUDA A LOS TRABAJADORES A GESTIONAR Y HACER CRECER SU NEGOCIO. Y A LOS CLIENTES A CONTRATAR SIN VUELTAS.



PROFESIONALES MATRICULADOS



ÁREA DE COBERTURA PERSONALIZABLE



DIAGNÓSTICO CON FOTO/AUDIO MEDIANTE IA



NEGOCIACIÓN ENTRE PROFESIONAL Y CLIENTE



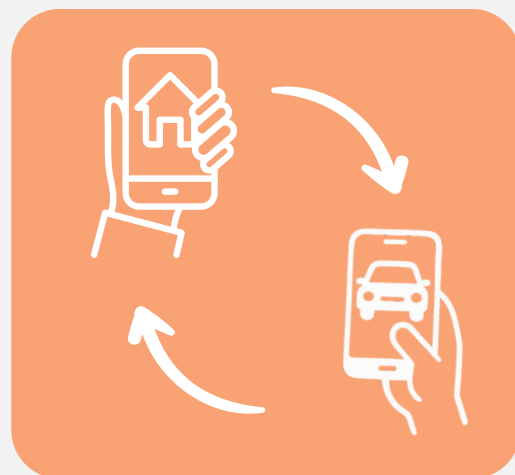
AGENDA DE TURNOS DEL PROFESIONAL



SEGUIMIENTO DEL TRABAJO DE INICIO A FIN



REPUTACIÓN DEL CLIENTE Y DEL PROFESIONAL



### RESULTADOS Y BENEFICIOS

- CLASIFICACIÓN AUTOMÁTICA DE FALLAS SIN LENGUAJE TÉCNICO.
- BÚSQUEDA RÁPIDA DE ESPECIALISTAS POR GEOLOCALIZACIÓN.
- PERFILES AUDITADOS CON DNI, MATRÍCULA Y ANTECEDENTES.
- GARANTÍA DE IDONEIDAD Y SATISFACCIÓN EN LA CONTRATACIÓN.
- GESTIÓN DE TURNOS Y DISPONIBILIDAD EN TIEMPO REAL.

### DESPLIEGUE

- CLOUD EN AWS (TERRAFORM/DOCKER) CON BACKEND SPRING BOOT/PYTHON, POSTGIS, COGNITO Y FRONTEND REACT NATIVE.

### FUTURO

- INTEGRACIÓN DE PASARELAS DE PAGO CON SISTEMA ESCROW.
- REENTRENAMIENTO DEL MODELO IA (FEEDBACK LOOP) CON PROFESIONALES.



UNIVERSIDAD TECNOLÓGICA NACIONAL  
FACULTAD REGIONAL CÓRDOBA  
INGENIERÍA EN SISTEMAS DE INFORMACIÓN

PROYECTO FINAL 5K3 2026

DOCENTES  
ING. QUINTEROS, SERGIO  
ING. TORRES HANSEN,  
TOMÁS

EQUIPO  
MORENO. MATÍAS  
NIETO, FEDERICO  
REVOL, LISANDRO  
VALDEMARCA, DANTE



# Domus, la solución para reparar tu hogar

**Moreno, Matías Emanuel – Nieto Poklepovic, Federico – Revol, Lisandro – Valdemarca Fragueiro, Dante**

*Universidad Tecnológica Nacional, Facultad Regional Córdoba*

## **Abstract**

Domus surgió para satisfacer la necesidad de centralizar y formalizar la búsqueda y contratación de servicios profesionales e independientes en un entorno digital confiable. Ante la fragmentación del mercado actual y la falta de mecanismos eficientes para verificar antecedentes, reputación y disponibilidad de los prestadores, se diseñó e implementó una plataforma integral de gestión. El sistema permite a los usuarios buscar profesionales por especialidad, evaluar perfiles mediante un esquema transparente de calificaciones y coordinar solicitudes de servicio de manera directa. Como elemento diferencial, la plataforma integra un módulo de inteligencia artificial para el diagnóstico preliminar de fallas mediante el análisis de imágenes y notas de audio expresadas en lenguaje natural. Por su parte, los prestadores disponen de un canal estructurado para autogestionar su oferta laboral y validar sus credenciales. La evaluación del sistema confirmó la viabilidad operativa de la solución, garantizando flujos de interacción fluidos y una alta precisión en la categorización automática de incidentes.

## **Palabras Clave**

Producto, Contratación de Servicios, Intermediación Digital, Formalización Laboral, Gestión de Perfiles, Confianza Digital.

## **Introducción**

La plataforma Domus surge ante la dificultad de los usuarios para contratar servicios técnicos calificados y la necesidad de los trabajadores autónomos de ordenar su gestión y presencia digital. En el ámbito regional, el rubro de oficios (electricidad, plomería, gas, refrigeración) padece una marcada desarticulación: falta de mecanismos para validar la idoneidad, presupuestos ambiguos, incapacidad del usuario para describir fallas técnicas y ausencia de antecedentes laborales verificables.

La mayoría de estas contrataciones ocurren de forma informal por mensajería o referencias verbales. Aunque resuelven el nexo inicial, carecen de trazabilidad, no validan identidades ni antecedentes y obligan a reiterar explicaciones complejas. Frente a esto, Domus se concibe como un ecosistema digital distribuido que simplifica la vinculación entre oferta y demanda. El proyecto integra una app móvil multiplataforma para clientes y prestadores junto a un backend escalable, capaz de procesar consultas geoespaciales en tiempo real, clasificar incidentes con inteligencia artificial multimodal y auditar perfiles profesionales.

## **Elementos del Trabajo y Metodología**

Para la ejecución integral del proyecto se implementó el marco de trabajo ágil Scrum, subdividiendo el desarrollo en ciclos incrementales (*sprints*) de dos semanas. La planificación del *backlog*, el seguimiento de historias de usuario y la trazabilidad de tareas se centralizaron en la plataforma Jira mediante su sistema de asignación de *issues* y tableros Kanban, manteniendo la comunicación síncrona del equipo a través de Discord y WhatsApp.

La arquitectura de software se diseñó bajo un esquema distribuido de microservicios y capas *decoupled* fundamentado en servicios API RESTful. En el extremo del servidor (backend) se utilizó Spring Boot (Java) para la lógica de negocio central y el procesamiento transaccional, complementado con modelos y componentes desarrollados en Python para el procesamiento de lenguaje natural, la transcripción de audio y la inferencia sobre imágenes para el diagnóstico preliminar de

incidentes. La interfaz de usuario (frontend móvil) se construyó utilizando React Native, permitiendo capturar fotografías, grabar notas de voz y desplegar una experiencia nativa y reactiva en dispositivos iOS y Android a partir de un código base unificado.

El almacenamiento de información se estructuró sobre una base de datos relacional PostgreSQL, expandida mediante el motor PostGIS para el soporte de tipos de datos geométricos y geográficos. Esta extensión posibilitó la ejecución de consultas espaciales optimizadas (como el cálculo de radio de cobertura y distancia euclidiana entre cliente y prestador). La evolución del esquema de la base de datos y el control de versiones de los DDL se administraron mediante Liquibase, garantizando entornos de desarrollo y producción sincronizados.

La infraestructura cloud se desplegó sobre la plataforma Amazon Web Services (AWS). El aprovisionamiento, la modificación y el mantenimiento de los recursos en la nube se realizaron mediante el enfoque de Infraestructura como Código (IaC) utilizando Terraform, lo que aseguró la replicabilidad del entorno. Para la automatización de eventos, notificaciones y flujos de integración entre servicios de terceros se configuró un motor de orquestación n8n. La capa de seguridad, autenticación, autorización y resguardo de credenciales se configuró a través de AWS Cognito, implementando el control de accesos basado en *JSON Web Tokens* (JWT) y esquemas de refresco de sesión (*refresh tokens*).

El ciclo de diseño y calidad acompañó todo el proceso: los esquemas de interfaz, componentes de UI y flujos de navegación se diseñaron en Figma, realizando pruebas de usabilidad previas a la programación. La documentación técnica, manuales de arquitectura y diagramas UML se centralizaron en Google Drive. Por último, la verificación funcional, el contraste de

esquemas JSON y la calidad de los endpoints de la API se auditaron de forma continua mediante la herramienta de pruebas Bruno.

## Resultados

El desarrollo de Domus permitió consolidar un ecosistema funcional que resuelve integralmente el flujo de contratación de servicios técnicos. La implementación del módulo de inteligencia artificial para el diagnóstico previo demostró ser un diferencial clave en la experiencia del usuario, permitiendo clasificar solicitudes a partir de fotografías y notas de audio en lenguaje coloquial. Esto simplifica significativamente la carga de incidentes y le otorga al usuario una comprensión clara del desperfecto antes de solicitar una cotización.

En cuanto al filtrado y localización geoespacial de prestadores, el algoritmo basado en el índice R-Tree de PostGIS y variables ponderadas de calificación y verificación demostró un desempeño óptimo en la sugerencia de profesionales cercanos, optimizando la asignación de solicitudes frente a los métodos de búsqueda informales.

En términos de seguridad y gestión de identidades, la integración con AWS Cognito y el módulo de validación documental permitió verificar la autenticidad de los medios de contacto y de las credenciales adjuntas (DNI, certificados de antecedentes y matrículas), elevando la percepción de confianza en las interacciones. Asimismo, las pruebas funcionales sobre la arquitectura desacoplada en Spring Boot y Python confirmaron la estabilidad de los servicios, garantizando baja latencia en el procesamiento de datos y una alta disponibilidad para el despliegue del sistema.

## Discusión

La propuesta resuelve la falta de transparencia en el trabajo independiente. El diagnóstico asistido por IA elimina el desconocimiento técnico del usuario al describir fallas, proveyendo al prestador una referencia clara antes de cotizar.

Durante el desarrollo se optimizaron los tiempos de inferencia de la IA y el consumo de las API de visión y audio para reducir latencias en dispositivos móviles. La combinación con Terraform permitió escalar eficientemente los recursos cloud en AWS, garantizando un entorno predecible que beneficia tanto al cliente como a la formalización del trabajo autónomo.

## Conclusión

Domus formaliza el contacto entre la oferta independiente y los usuarios mediante un marco operativo ágil, inteligente y confiable. La integración de IA multimodal y geolocalización permite resolver el ciclo completo de atención, desde la identificación del problema hasta la contratación.

Como trabajo futuro se contempla la integración de pasarelas de pago con custodia de fondos (escrow), un módulo de agenda en tiempo real y el reentrenamiento continuo del modelo de IA (feedback loop) con la validación de los profesionales. La solución constituye una alternativa robusta dentro de la ingeniería de software aplicada.

## Agradecimientos

Queremos manifestar nuestro sincero reconocimiento al Ing. Sergio Quinteros, profesor titular de la cátedra, por la formación brindada y la coordinación de las clases a lo largo del curso, así como al docente Tomás Torres Hansen, cuyas orientaciones, correcciones y constante acompañamiento resultaron claves durante las etapas de ejecución y validación de este trabajo. De igual manera, agradecemos al cuerpo académico de la Universidad Tecnológica Nacional – Facultad Regional Córdoba por el marco institucional provisto y a nuestros compañeros por el intercambio compartido durante el desarrollo de la carrera.

## Referencias

- [1] **Scrum Guide:** *The Definitive Guide to Scrum: The Rules of the Game*. Disponible en: [scrumguides.org/scrum-guide.html](https://scrumguides.org/scrum-guide.html)
- [2] **Spring Boot Documentation:** *Spring Framework Reference Documentation*. VMware, Inc. Disponible en: [spring.io/projects/spring-boot](https://spring.io/projects/spring-boot)
- [3] **React Native Documentation:** *A framework for building native apps using React*. Meta Platforms, Inc. Disponible en: [reactnative.dev](https://reactnative.dev)
- [4] **PostgreSQL & PostGIS:** *Spatial and Geographic Objects for PostgreSQL*. PostGIS Project. Disponible en: [postgis.net](https://postgis.net)
- [5] **AWS Cognito:** *Developer Guide for Amazon Cognito User Pools*. Amazon Web Services, Inc. Disponible en: <https://aws.amazon.com/cognito>
- [6] **Terraform by HashiCorp:** *Infrastructure as Code Documentation*. HashiCorp. Disponible en: [terraform.io](https://terraform.io)
- [7] **Liquibase:** *Database Schema Change Management Official Docs*. Liquibase Inc. Disponible en: [liquibase.org](https://liquibase.org)
- [8] **Bruno API Client:** *Fast, Open-source API Client for Git*. Disponible en: [docs.usebruno.com](https://docs.usebruno.com)
- [9] **GitLab Docs:** *DevOps platform and Repository Management*. GitLab Inc. Disponible en: [gitlab.com](https://gitlab.com)
- [10] **Figma:** *Collaborative Interface Design Tool*. Figma Inc. Disponible en: [help.figma.com](https://help.figma.com)

## Datos de Contacto

Moreno, Matías Emanuel. Universidad Tecnológica Nacional. Email: [morenomatias847@gmail.com](mailto:morenomatias847@gmail.com)

Nieto Poklepovic, Federico. Universidad Tecnológica Nacional. Email: [federiconieto04@gmail.com](mailto:federiconieto04@gmail.com)

Revol, Lisandro. Universidad Tecnológica Nacional. Email: [revol.lisandro@gmail.com](mailto:revol.lisandro@gmail.com)

Valdemarca Fragueiro, Dante. Universidad Tecnológica Nacional. Email: [dante.valdemarca@gmail.com](mailto:dante.valdemarca@gmail.com)

## PLANILLA PARA CATALOGAR EL PROYECTO FINAL

|                                                                         |      |                                                          |           |
|-------------------------------------------------------------------------|------|----------------------------------------------------------|-----------|
| AÑO                                                                     | 2026 | CURSO Y NRO. DE GRUPO                                    | 5K3 - G04 |
| NOMBRE DEL SISTEMA / PROYECTO                                           |      |                                                          |           |
| Domus                                                                   |      |                                                          |           |
| CATEGORÍA (Solución De Negocio / Producto / Proyecto De Impacto Social) |      |                                                          |           |
| Producto                                                                |      |                                                          |           |
| HERRAMIENTAS Y TECNOLOGÍAS UTILIZADAS                                   |      |                                                          |           |
| ÁMBITO DE APLICACIÓN                                                    |      | NOMBRE Y VERSIÓN                                         |           |
| ENTORNO DE DESARROLLO                                                   |      | VS Code                                                  |           |
| REPOSITORIOS Y VERSIONADO                                               |      | Gitlab                                                   |           |
| PROGRAMACIÓN                                                            |      | Spring Boot - React Native - Python                      |           |
| BASE DE DATOS                                                           |      | PostgreSQL + PostGIS - Liquibase - AWS - Terraform - n8n |           |
| COMUNICACIÓN INTERNA                                                    |      | Whatsapp - Discord                                       |           |
| CAPACITACIÓN                                                            |      | Documentación técnica oficial - Cursos de internet       |           |
| PRUEBAS DE SISTEMA                                                      |      | Bruno - Swagger                                          |           |
| GESTIÓN DEL PROYECTO                                                    |      | Jira                                                     |           |
| DOCUMENTACIÓN                                                           |      | Google Docs - Google Drive                               |           |
| MODELOS                                                                 |      | Figma                                                    |           |