

**APLICACIÓN WEB PARA LA GESTIÓN INTEGRAL DE INFORMACIÓN EN MIPYMES DE INGENIERÍA CIVIL:
EN AMERICAN CARPAS 1 SAS**

Yerson Duvan Arias Parra, Mario Rubiel Quique Quesada.



Ingeniería civil, Facultad de ingenierías

Universidad La Gran Colombia

Bogotá

2025

Aplicación web para la gestión integral de información en MiPymes de ingeniería civil: en American

Carpas 1 Sas

Yerson Duvan Arias Parra, Mario Rubiel Quique Quesada

Trabajo de Grado presentado como requisito para optar al título de ingeniero civil.

Director: Magister Wilson Enrique Torres Sánchez



UNIVERSIDAD
La Gran Colombia

Vigilada MINEDUCACIÓN

Ingeniería civil, Facultad de ingenierías

Universidad La Gran Colombia

Bogotá

2025

Tabla de contenido

RESUMEN.....	10
ABSTRACT	11
INTRODUCCIÓN	12
PLANTEAMIENTO DEL PROBLEMA	14
JUSTIFICACIÓN	15
OBJETIVOS.....	16
OBJETIVO GENERAL	16
OBJETIVOS ESPECÍFICOS	16
1. MARCO REFERENCIAL.....	17
1.1 ANTECEDENTES	17
1.2 MARCO TEÓRICO	19
1.2.1 Base de datos	20
1.2.2 Descripción de las MyPymes	22
1.2.3 Gestión de datos.....	22
1.2.4 Beneficios de una aplicación web.....	23
1.2.5 Marco legal	24
2. METODOLOGÍA.....	25
2.1 FASE DE IDENTIFICACIÓN.....	26
2.2 FASE DE DISEÑO	26
2.3 FASE DE CONSTRUCCIÓN	27
2.4 FASE DE VALIDACIÓN Y RETROALIMENTACIÓN.....	27
3. FASE DE IDENTIFICACIÓN	28

	4
3.1 NECESIDADES.....	28
3.2 MÓDULO DE INVENTARIOS.....	30
3.3 MÓDULO DE TRABAJADORES.....	31
3.4 MÓDULO DE PROVEEDORES	32
3.5 MÓDULO DE PROYECTOS	33
4. FASE DE DISEÑO	35
4.1 DISEÑO DE BASE DE DATOS (MODELO ENTIDAD-RELACIÓN)	35
4.2 ARQUITECTURA DEL SISTEMA	38
4.3 DISEÑO DE INTERFAZ DE USUARIO (WIREFRAMES/MOCKUPS)	
5. FASE DE DESARROLLO (CONSTRUCCIÓN)	45
5.1 CONFIGURACIÓN GENERAL DEL PROYECTO	46
5.1.1 Creación del proyecto Django	46
5.1.2 Configuración de Settings.py	47
5.2 MODULO DE TRABAJADORES	49
5.2.1 Creación del módulo de trabajadores	49
5.2.2 Diseño del modelo de datos.....	50
5.2.3 Catalogo de datos (Choices).....	55
5.2.4 Migraciones de base de datos.....	56
5.2.5 Vistas (Views)	57
5.2.5 Urls (Enrutamiento)	58
5.2.5 Templates (Plantillas HTML)	58
5.2.5 Formularios (Forms)	59
5.3 MODULO DE PROVEEDORES	60
5.3.1 Creación del módulo de proveedores	60
5.3.2 Diseño del Modelo de Datos (dividido en 5 fases)	61
5.3.3 Catálogos de Datos (Choices).....	65

5.3.4 Migraciones de Base de Datos	65
5.3.5 Resumen de Arquitectura	66
5.4 MODULO DE INVENTARIOS	66
5.4.1 Creación del módulo de inventarios	66
5.4.2 Arquitectura General del Módulo.....	66
5.4.3 Submódulo: catálogos de lonas	67
5.4.4 Submódulo: catálogos de estructura.....	68
5.4.5 Submódulo: inventario de lonas estructuras	69
5.4.6 Submódulo: órdenes de producción	70
Formulario para crear orden de producción	71
5.5 MODULO DE PROYECTO.....	71
5.5.1 Fase 1: Catálogos base	72
5.5.1 Fase 2: Gestión de clientes.....	73
5.5.1 Fase 3: Proyecto principal	73
5.5.1 Fase 4: Actividades y cronograma.....	74
5.5.1 Fase 5: Asignación de recursos.....	75
5.5.1 Fase 6: Documentación.....	76
5.5.1 Fase 7: Evidencia visual.....	76
6. FASE DE VALIDACIÓN Y RETROALIMENTACIÓN	77
CONCLUSIONES.....	79
LISTA DE REFERENCIA O BIBLIOGRAFÍA.....	¡Error! Marcador no definido.
ANEXOS.....	¡Error! Marcador no definido.

Lista de Figuras

Figura 1. <i>Ciclo de fases del proyecto.</i>	25
Figura 2. <i>Modelo entidad relación del módulo de trabajadores.</i>	36
Figura 3. <i>Modelo entidad relación del modulo de inventarios.</i>	37
Figura 4. <i>Modelo entidad relación del modulo de Proyectos.</i>	38
Figura 5. <i>Diagrama de archivos principales y flujo de interacción en Django</i>	39
Figura 6. <i>Diagrama de interacción de las herramientas para la aplicación web</i>	41
Figura 7. <i>Identificación de colores en la página web. visual</i>	43
Figura 8. <i>Interacción de componentes visuales en pantalla</i>	44
Figura 9. <i>Mockup Home (Inicio) para sistema web.</i>	44
Figura 10. <i>Mockup para Login (Iniciar Sesión).</i>	45
Figura 11. <i>Mockup de interacción de módulos (Trabajadores, Inventarios, Proveedores y proyectos)</i>	45
Figura 12. <i>Estructura inicial de Django en Visual Studio Code</i>	46
Figura 13. <i>Configuración de MySQL como base de datos.</i>	47
Figura 14. <i>Configuración de archivos estáticos y multimedia</i>	48
Figura 15. <i>Instalación de aplicaciones e integración de variables Railway (código)</i>	49
Figura 16. <i>Creación y estructura de módulo de trabajadores</i>	50
Figura 17. <i>Modelo Trabajador-persona en código</i>	51
Figura 18. <i>Formulario para registro de nuevo trabajador</i>	52
Figura 19. <i>Modelo trabajador-laboral (código)</i>	53

Figura 20. <i>Modelo trabajador-afiliaciones (código)</i>	54
Figura 21. <i>Modelo trabajador-documentos (código)</i>	55
Figura 22. <i>Catálogo de datos en campos de información</i>	56
Figura 23. <i>Migración para modelo de trabajador. Personal</i>	57
Figura 24. <i>Vista basada en función de listado de trabajadores</i>	58
Figura 25. <i>Enrutamiento de módulos (código)</i>	58
Figura 26. <i>Estructura interna del archivo templates</i>	59
Figura 27. <i>Formulario datos trabajador-personal</i>	60
Figura 28. <i>Creación y estructura de módulo de trabajadores</i>	61
Figura 29. <i>Fases del módulo de proveedores</i>	62
Figura 30. <i>Código para categorizar proveedor</i>	63
Figura 31. <i>Lista proveedores</i>	63
Figura 32. <i>Campos para crear un proveedor</i>	64
Figura 33. <i>Diagrama de relaciones de proveedores</i>	66
Figura 34. <i>Codificación y datos de ingreso para características de lona</i>	68
Figura 35. <i>secciones para caracterizar tubería</i>	69
Figura 36. <i>Secciones de inventario</i>	70
Figura 37. <i>Formulario para crear orden de producción</i>	71
Figura 38. <i>Estructura del módulo de proyectos (7 fases)</i>	72
Figura 39. <i>Diagrama de Gantt en proyectos (Ejemplo)</i>	74

Figura 40. <i>Visualización general de proyecto</i>	75
Figura 41. <i>Pruebas realizadas a la aplicación</i>	77
Figura 42. <i>Imagen de listado de trabajadores</i>	78
Figura 43. <i>Diagrama de flujo de información de proveedores (Actual)</i>	85
Figura 44. <i>Diagrama de flujo de información de Proyectos (Actual)</i>	85
Figura 45. <i>Diagrama de flujo de información de Trabajadores (Actual)</i>	86
Figura 46. <i>Diagrama de flujo de información de Inventarios (Actual)</i>	86

Lista de Tablas

Tabla 1. <i>Cuadro comparativo de respuestas de la encuesta sobre el manejo de la información.....</i>	29
Tabla 2. <i>Requerimientos funcionales y no funcionales del módulo de inventarios.</i>	30
Tabla 3. <i>Requerimientos funcionales y no funcionales del módulo de trabajadores.</i>	31
Tabla 4. <i>Requerimientos funcionales y no funcionales del módulo de proveedores.</i>	33
Tabla 5. <i>Requerimientos funcionales y no funcionales del módulo de proyectos.</i>	34
Tabla 6. <i>Cuadro comparativo de sistema antigua y sistema implementado (nuevo)</i>	81

Resumen

Este proyecto aborda la gestión ineficiente de información en MiPymes del sector de ingeniería civil, tomando como caso American Carpas 1 SAS, donde la dependencia de hojas de cálculo (Excel) genera duplicidad de datos, desactualización, falta de accesibilidad y controles, lo que limita la toma de decisiones y la competitividad. Se propone el desarrollo de una aplicación web integral para centralizar y optimizar la gestión de inventarios, trabajadores, proveedores y proyectos, con el propósito de garantizar la integridad, consistencia y seguridad de la información, reducir errores y reprocesos, y facilitar la planificación estratégica mediante reportes personalizados y trazabilidad de movimientos.

El proyecto se enmarca en un estudio de caso cualitativo y adopta una metodología ágil (Scrum) estructurada en las fases de identificación, diseño, construcción y validación. Para la recolección y análisis se emplearon entrevistas, observación directa y análisis documental; en diseño se aplicaron modelado entidad-relación y wireframes/mockups; y en construcción se implementó la solución con Python y Django (ORM), base de datos MySQL, frontend con HTML/Bootstrap y desarrollo en Visual Studio Code. Las pruebas incluyeron validación de formularios, integridad de datos, control de permisos y rendimiento. Se espera con ello centralizar la información, automatizar alertas (stock y vencimientos), mejorar la seguridad mediante roles y auditoría, acelerar búsquedas en tiempo real y ofrecer escalabilidad y compatibilidad móvil para acompañar el crecimiento de la empresa.

Palabras clave: *Gestión de información, MiPymes, Aplicación web, Ingeniería civil, American Carpas 1 SAS, Bases de datos.*

Abstract

This project addresses inefficient information management in MSMEs in the civil engineering sector, taking American Carpas 1 SAS as a case study, where reliance on spreadsheets (Excel) leads to data duplication, outdated information, lack of accessibility, and controls, which limits decision-making and competitiveness. The proposal is to develop a comprehensive web application to centralize and optimize the management of inventories, workers, suppliers, and projects, with the aim of ensuring the integrity, consistency, and security of information, reducing errors and rework, and facilitating strategic planning through customized reports and traceability of movements.

The project is part of a qualitative case study and adopts an agile methodology (Scrum) structured in the phases of identification, design, construction, and validation. Interviews, direct observation, and document analysis were used for data collection and analysis; entity-relationship modeling and wireframes/mockups were applied in the design phase; and in the construction phase, the solution was implemented with Python and Django (ORM), a MySQL database, HTML/Bootstrap frontend, and development in Visual Studio Code. Testing included form validation, data integrity, permission control, and performance. The aim is to centralize information, automate alerts (stock and expirations), improve security through roles and auditing, speed up real-time searches, and offer scalability and mobile compatibility to accompany the company's growth.

Keywords: *Information management, MiPymes, Web application, civil engineering, American Carpas 1 SAS, database.*

Introducción

En el escenario actual de transformación digital, las MiPymes colombianas que constituyen la mayoría del tejido empresarial y una proporción significativa del empleo afrontan obstáculos persistentes en la administración de su información. American Carpas 1 SAS, microempresa dedicada a la fabricación de carpas para eventos y para el sector de la construcción, gestiona datos clave de personal, proveedores, inventarios y proyectos a través de múltiples hojas de cálculo. Este enfoque ha propiciado duplicidad y dispersión de registros, actualización irregular, dificultades de acceso, y carencia de trazabilidad y controles, con efectos directos sobre la eficiencia operativa, el control administrativo y la toma oportuna de decisiones.

Desde el sustento teórico, la literatura y la normativa en gestión de la información enfatizan la centralización de datos mediante sistemas de gestión de bases de datos, con el fin de asegurar integridad, consistencia y seguridad. Las aplicaciones web, gracias a su arquitectura cliente-servidor y a su acceso desde distintos dispositivos, favorecen la colaboración en tiempo real, el control de versiones, la trazabilidad y la escalabilidad, elementos críticos para organizaciones con recursos limitados. Del mismo modo, la caracterización regulatoria de las MiPymes orienta el dimensionamiento de soluciones acordes a su escala, mientras los lineamientos de gestión documental electrónica resaltan la relevancia de los metadatos, el control de cambios y la conservación segura.

En este marco, el objetivo de la investigación es construir una aplicación web que optimice la gestión de datos, documentación e inventarios de la documentación interna de trabajadores y proveedores en American Carpas 1 SAS. Para lograrlo, se relevarán necesidades y requerimientos por proceso; se modelarán la base de datos (MER) y la interfaz de usuario; se desarrollarán módulos de inventarios, proveedores, proyectos y mano de obra, incorporando controles de acceso y auditoría; y se validará el sistema con usuarios finales para refinar funcionalidades y usabilidad.

La metodología adoptará un enfoque cualitativo bajo estudio de caso y desarrollo ágil (Scrum). Se prevén cuatro fases: identificación (entrevistas, observación y análisis documental para mapear flujos e incidencias), diseño (definición de arquitectura, prototipos y modelo de datos), construcción (selección tecnológica y desarrollo incremental) y validación (pruebas con usuarios y mejoras). Se espera obtener una plataforma funcional que disminuya reprocesos y errores por duplicidad, eleve la oportunidad y confiabilidad de la información, fortalezca la trazabilidad y la seguridad, y aporte indicadores útiles para la planeación. En el corto plazo, la herramienta apunta a incrementar la eficiencia y respaldar decisiones basadas en datos; adicionalmente, el modelo podrá replicarse en otras MiPymes del sector, contribuyendo a su competitividad y a procesos sostenibles de transformación digital.

Planteamiento del problema

En Colombia, las micro, pequeñas y medianas empresas (MiPymes) ocupan un lugar protagónico en la estructura económica, representando el 90% de las nuevas empresas (Confecámaras, 2024, p. 5), evidenciando su importancia en la dinámica empresarial. De ahí radica su impacto en el mercado laboral, tomando datos de la Gran Encuesta Integrada de Hogares del DANE, se estima que generan alrededor del 79% del empleo nacional, tal como se cita en (González & Llanes, 2024). Aun con este rol macroeconómico, estas organizaciones enfrentan obstáculos sustanciales para consolidarse y escalar, entre los cuales sobresale la gestión de la información (Puentes, 2013). En pleno proceso de transformación digital, muchas MiPymes continúan apoyándose en herramientas ofimáticas básicas, por ejemplo, hojas de cálculo de Excel para manejar procesos críticos. Aunque Excel ofrece versatilidad y fácil adopción, su empleo como plataforma principal de gestión de datos acarrea limitaciones que merman la competitividad. Autores como Laudon y Laudon citados por Lapiedra et al (2021, p. 36) señalan que las hojas de cálculo carecen de las herramientas de una base de datos especializada, lo que conduce a problemas de integridad de los datos, redundancia, dificultades para el acceso concurrente y una deficiente estandarización de la información, afectando directamente la capacidad de la empresa para tomar decisiones ágiles y basadas en datos confiables.

En ese escenario se encuentra American Carpas 1 SAS, que es una microempresa dedicada a la fabricación de carpas para eventos y construcción, es un claro ejemplo, ya que gestiona la totalidad de sus datos de trabajadores, proyectos, inventario de materiales y proveedores a través de múltiples archivos de Excel. Esta metodología ha generado una serie de ineficiencias operativas como la duplicidad de registros, ya que diferentes áreas tienen su propia versión de un mismo dato, haciendo que la información este descentralizada en distintos equipos lo que dificulta tener una visión integral y actualizada de los proyectos, la desactualización es una constante, pues los cambios no se reflejan en

tiempo real para todos los usuarios; y se evidencia una notable falta de accesibilidad y control , limitando el acceso a la información crítica y careciendo de mecanismos de seguridad y trazabilidad. Estas deficiencias no solo generan reprocesos y aumentan la probabilidad de errores costosos, sino que también obstaculizan la planificación estratégica. Ante esta situación, se formula la siguiente pregunta:

¿Cómo desarrollar una aplicación web que mejore la gestión de datos, documentación e inventarios en la empresa American Carpas 1 SAS, resolviendo los problemas de duplicidad, desactualización, falta de accesibilidad y control?

Justificación

La importancia de este proyecto radica en la necesidad urgente de resolver una problemática real y tangible que afecta a la microempresa American Carpas 1 SAS. La dependencia actual de la empresa sobre las hojas de cálculo dispersas no es un simple inconveniente administrativo; es una fuente constante de riesgo y fricción operativa. Esta metodología genera ineficiencias documentadas, como la duplicidad de datos , que puede llevar a costosos errores en la compra de materiales o en la facturación, y la desactualización crónica de la información, que impide a los gerentes conocer el estado real de un proyecto o del inventario. Además, la ausencia de controles de seguridad deja datos sensibles a personas vulnerables.

Este proyecto pretende ofrecer una solución directa y de acuerdo a las necesidades de la empresa, en este caso American Carpas 1 SAS, no se pretende proponer una herramienta que se pueda encontrar en el mercado, sino crear una que se adapte a las necesidades de la empresa. Los aportes prácticos serán concretos y medibles: se busca implementar un sistema funcional que impone la integridad de los datos a través de una base única, garantizando la accesibilidad controlada en tiempo

real y proporcionando una visión integral de los proyectos. Esta transformación no sólo optimizará los procesos y reducirá los errores, sino que fortalecerá la capacidad de la empresa para realizar planificaciones estratégicas basadas en datos confiables, mejorando así su agilidad y fortaleciendo su competitividad en un mercado exigente

Objetivos.

Objetivo General

Desarrollar una aplicación web que mejore la gestión de datos, documentación e inventarios del archivo interno de trabajadores y proveedores de la empresa American Carpas 1 SAS.

Objetivos Específicos

Identificar las principales necesidades y requerimientos documentales en el área administrativa de la empresa American Carpas 1 SAS, enfocándose en las áreas de inventarios, mano de obra, proveedores y proyectos.

Diseñar la estructura de la aplicación Web teniendo en cuenta la identificación de las necesidades y requerimientos documentales en el área administrativa de la empresa American Carpas 1 SAS.

Construir una aplicación Web teniendo en cuenta el lenguaje de programación más adecuado que permita atender las necesidades y requerimientos planteados.

Validar la aplicación web mediante una prueba de usuario, buscando ajustes y mejoras.

1. Marco referencial

Se establece los estudios previos y los fundamentos teóricos que sirven de base para esta investigación. Por un lado, los antecedentes evidencian que en el sector de la construcción las deficiencias en la gestión documental y de inventarios son recurrentes, principalmente por la ausencia de sistemas centralizados de información. Por otro, el marco teórico desarrolla los conceptos esenciales sobre bases de datos, gestión de información, MiPymes y aplicaciones web, que permiten comprender las implicaciones técnicas y administrativas de la solución propuesta. Así, la unión de ambos enfoques establece lo necesario para orientar el desarrollo de una aplicación web que optimice los procesos de gestión en American Carpas 1 SAS.

1.1 Antecedentes

Los análisis críticos e interrelacionados de ineficiencia operativa como la gestión documental en proyectos y el control de inventarios en el sector de la construcción, se reconoce una conclusión central que surge de diversas fuentes, entre ellas es la identificación de sistemas de gestión de la información inadecuados, carentes de una base de datos centralizada y estructurada, como una causa fundamental de deficiencias operativas, sobrecostos y retrasos. En consecuencia, los siguientes estudios no solo validan la necesidad de una solución tecnológica, sino que también delinear los componentes esenciales que esta debe poseer para generar un impacto estratégico:

En primera instancia, Barajas (2023), dio una “PROPUESTA DE MEJORA PARA LA GESTIÓN DOCUMENTAL DEL ÁREA DE PROYECTOS DE LA EMPRESA MIRS LATINOAMERICA S.A.S”, incluyendo así opciones como adopción del software especializado ACONEX de ORACLE, que permite la gestión centralizada de información, facilitando el almacenamiento y gestión de documentos, planos y otros archivos relacionados con proyectos. Del mismo modo presenta que para ello debe existir la elaboración

de manuales o directrices que guíen a los empleados sobre cómo gestionar correctamente los documentos y establece así una base de conocimiento común en la organización. Cabe resaltar que Edwin comenzó con la identificación de los obstáculos que encontraba en el control de los proyectos, luego definió y consultó las áreas que comprendían el análisis de un proyecto, localizando así los vacíos para conformar una buena comunicación.

Del mismo modo, en la empresa Realidad Colombia SAS encargada de la contratación y ejecución de obras civiles contaba con problemáticas operativas que derivan en demoras y pérdidas críticas. Es allí donde Parra y Fuentes (2023), en su artículo de investigación, identificaron la falta de un sistema de gestión de inventarios como una de las causas principales en estas ineficiencias. Para ello, definieron un diagnóstico conformado por medio de entrevistas, del cual obtuvieron las falencias más comunes y de allí partieron al desarrollo de estrategias como la implementación del árbol de problemas y la matriz DOFA. Siendo así, permitieron utilizar una estrategia potencial definida como análisis ABC, del cual obtuvieron resultados más precisos por la cantidad de materia prima, herramientas y productos que se llevan en la empresa Realidad Colombia SAS. Esto les permitió validar fórmulas para el inventario con el fin de desarrollar revisiones periódicas, teniendo como resultados índices de rotación de inventario, duración de mercancía y exactitud del inventario.

Por otro lado, Gómez & Guzmán (2016) ejecutaron un sistema similar al de Realidad Colombia Sas en la empresa Ingeniería Solida Ltda. estableciendo un análisis de identificación de la problemática, en este caso en la eficiencia de 5 proyectos de construcción con la gestión de inventarios y sistemas de stock, implementaron observación directa y trabajo en campo, verificando deficiencias como la falta de formatos de ingreso y salida de herramientas y materia prima, la organización espacial en el área de almacén, la asignación de responsabilidades para cada puesto, almacenistas, ingenieros y operarios. Un

factor relevante de esta investigación es el uso de indicadores de la gestión de los inventarios, dando a conocer la eficiencia de la metodología que utilizaron para el mejoramiento del sistema de gestión para la empresa Ingeniería Solida Ltda.

En ese orden de ideas, el Archivo General de la Nación (2023) establece una guía técnica y normativa sobre la implementación de Sistema de Gestión Documental Electrónica de Archivos (SGDEA), ofrece un marco metodológico para la transición de archivos físicos a digitales, lo cual está intrínsecamente ligado al diseño de un sistema de gestión de la información. La guía subraya que un SGDEA no es solo un repositorio de archivos, sino un sistema complejo que depende de una base de datos robusta y bien estructurada para gestionar metadatos, controlar versiones, asegurar la trazabilidad, mejorar y garantizar la seguridad de la información.

Asimismo, en la investigación titulada “Sistemas de gestión documental: Una implementación adecuada en las empresas públicas y privadas de nuestro país” (Montes et al 2023). Demuestra el beneficio de adoptar estas tecnologías, El estudio analiza cómo la correcta implementación de un sistema de gestión documental, soportado por una base de datos centralizada, se convierte en un pilar para el sistema de gestión de la información de una empresa. Los autores concluyen que estas herramientas optimizan la toma de decisiones, mejoran la eficiencia operativa y fortalecen la competitividad en cada área.

1.2 Marco teórico

Para el desarrollo de una aplicación tecnológica como gestora de la información en el contexto de las micro, pequeñas y medianas empresas (MiPymes) en Colombia, es importante identificar cuatro pilares interconectados: en primer lugar, se encuentra el concepto de bases de datos relacionales que

cumple el papel de núcleo central en la centralización y gestión de la información. En segundo lugar, se delimitará el concepto de MiPymes dentro de la definición dada por el Ministerio de Comercio. En tercer lugar, se presentará la gestión de datos como medio clave para la mejora de los procesos de producción. Finalmente, se expondrán las ventajas que aporta una aplicación web, con respecto a otras soluciones tecnológicas.

1.2.1 Base de datos

Una base de datos puede entenderse como una colección de datos organizada para atender de manera eficiente diversas necesidades, reduciendo la redundancia y facilitando su control. Esto contrasta con los esquemas tradicionales basados en archivos, por ejemplo, múltiples hojas de cálculo independientes en los que cada usuario o aplicación conserva sus propios registros, lo que fomenta silos e inconsistencias. El componente que hace posible la centralización de la información es el Sistema de Gestión de Bases de Datos (SGBD), un software que se coloca entre las aplicaciones y el almacenamiento físico y permite administrar los datos de forma eficiente, sin que el usuario deba conocer los pormenores de su gestión. En las organizaciones, adoptar este sistema suele ser una respuesta a problemas habituales como la inconsistencia, la dependencia entre datos y las brechas de seguridad, con el propósito de establecer una única fuente de verdad, íntegra y controlada. Su papel central consiste en gestionar la recuperación, actualización y eliminación de grandes volúmenes de información con eficiencia, preservando la integridad, la consistencia y la seguridad de los datos (Silberschatz et al, 2020). En el ámbito empresarial, estos sistemas resultan cruciales porque permiten estructurar conjuntos de datos de manera sistemática y organizarlos por áreas o módulos (Raffino, 2024). En este marco, las bases de datos relacionales dominan tanto en la academia como en el sector corporativo: organizan la información en tablas conectadas por claves primarias y foráneas, conforme al modelo

planteado por Codd en 1970, lo que habilita relaciones lógicas entre datos, consultas complejas y el aseguramiento de la integridad referencial (Codd, 1970).

La selección del motor de base de datos para una aplicación web depende de la naturaleza y el tamaño de los datos, de las exigencias de escalabilidad y de la experiencia del equipo técnico. En una microempresa que maneja información estructurada de empleados, inventario y proveedores, una base de datos relacional como PostgreSQL o MySQL suele ser la alternativa más conveniente por su robustez, sus garantías de seguridad y su buena integración con frameworks web (Elmasri & Navathe, 2017). Este enfoque no se limita a guardar datos: cubre todo su ciclo de vida, desde la captura y organización hasta el mantenimiento, el acceso, el uso y el archivado seguro. La meta de una gestión de datos bien diseñada es asegurar información precisa, confiable, consistente y disponible para respaldar la toma de decisiones (Laudon & Laudon, 2012).

En mercados competitivos, una gestión deficiente convierte la información en un recurso pasivo en una comunicación intermitente, mientras que una gestión formal soportada por sistemas centralizados la transforma en una ventaja competitiva. Así, desde el área de administración puede realizar análisis, identificar tendencias y decidir con base en evidencia, logrando que los sistemas de información evolucionen de ser un coste añadido para el negocio a un servicio fundamental para el crecimiento continuo del mismo (Lapiedra et al., 2021). En muchos casos, la gestión de la información se hace de forma manual o con herramientas básicas como hojas de cálculo, lo que incrementa la probabilidad de errores, duplicidades y pérdidas de datos relevantes. Este escenario limita la capacidad de las microempresas para decidir, optimizar recursos y competir en mercados cada vez más competitivos.

1.2.2 Descripción de las MiPymes

Las MiPymes son la columna vertebral de la economía colombiana en términos de generación de empleo y estas empresas se clasifican según sus ingresos por actividades ordinarias, de acuerdo con el Decreto 957 de 2019. Las empresas del sector manufacturero se dividen en tres categorías: las microempresas tienen menos de 10 trabajadores y perciben ingresos inferiores a 23,563 UVT; las pequeñas empresas perciben entre 23,563 y 204,995 UVT; y las medianas empresas perciben entre 204,995 y 1,736,565 UVT. Es importante comprender la clasificación de las MiPymes para identificar sus desafíos y oportunidades y desarrollar una solución tecnológica que se adecúe a su tamaño y capacidad de inversión. Así, las microempresas constituyen más del 90% de las empresas del país y emplean a un número significativo de personas (Ministerio del Comercio, Industria y Turismo, 2022).

No solo son importantes porque ayudan a generar la economía local, sino que también son muy flexibles y se adaptan al entorno en el que se encuentran. Sin embargo, enfrentan grandes obstáculos como el acceso limitado a la tecnología de la información, una administración informal y la falta de estandarización de procesos (Sarmiento et al., 2024). Por lo tanto, la transformación digital es una oportunidad para superar estas barreras mediante la implementación de herramientas tecnológicas que ayuden a mejorar la eficiencia de las operaciones, así como la gestión de la información. En este contexto, la implementación de bases de datos especializadas y aplicaciones web es necesaria para fortalecer y asegurar la sostenibilidad de las microempresas.

1.2.3 Gestión de datos

La relación entre las microempresas, las bases de datos, la gestión de datos y las aplicaciones web es crucial para comprender el impacto de la digitalización en el sector empresarial. La implementación de soluciones tecnológicas especializadas basadas en web y bases de datos permitirá a las microempresas superar las limitaciones asociadas con la gestión manual de la información, logrando

una mayor eficiencia, productividad y adaptabilidad en un entorno competitivo. Una gestión de datos adecuada, respaldada por bases de datos robustas y aplicaciones web intuitivas, permitirá la centralización de la información, la reducción de errores, la mejora de la seguridad y el acceso a herramientas de análisis que facilitarán la toma de decisiones estratégicas. Así, la tecnología se convertirá en un aliado esencial en el crecimiento y la sostenibilidad de las microempresas, no solo en Colombia sino también a nivel mundial (Sarmiento et al., 2024).

Una aplicación web, como software cliente-servidor accesible a través de un navegador, ofrece grandes ventajas sobre el software de escritorio tradicional. Su accesibilidad universal permitirá a los usuarios conectarse desde cualquier dispositivo con acceso a internet, eliminando las barreras geográficas. Para las microempresas, una aplicación web puede incorporar módulos para la gestión de empleados, inventarios, proveedores y proyectos, lo que centraliza toda la información en una única plataforma, mejorando así la eficiencia operativa y la toma de decisiones con datos actualizados y fiables (Benvenuto Vera, 2006). Además, la centralización significa que todos los usuarios trabajan con una única versión de la aplicación y la base de datos, evitando problemas de información desactualizada, y es más fácil de mantener, ya que las actualizaciones se realizan directamente en el servidor y se reflejan inmediatamente en todos los usuarios. Por lo tanto, su arquitectura permite escalar la capacidad de la aplicación según el crecimiento de la empresa, y todas estas características ayudan a mejorar la coordinación y el flujo de información, reducir los costos de transacción y aumentar la agilidad organizacional (Laudon & Laudon, 2012).

1.2.4 Beneficios de una aplicación web

Las aplicaciones web ofrecen ventajas estratégicas para las MiPymes al centralizar la información en una plataforma accesible desde cualquier dispositivo con acceso a internet, lo que permite operaciones en tiempo real y trabajo colaborativo. Su arquitectura en la nube reduce los costos

de infraestructura y mantenimiento, facilita la escalabilidad según las necesidades y permite la integración de módulos o servicios externos a través de API estandarizadas. Esto se traduce en una mayor productividad, un mayor alcance y cobertura comercial, mejores procesos de servicio al cliente y decisiones más informadas. Además, su implementación puede ser progresiva, por procesos o áreas, lo que contribuye a la solución de algunas de las barreras identificadas, como la falta de personal especializado. Sin embargo, estudios recientes muestran que la digitalización y el uso de plataformas en línea por parte de las MiPymes tienen efectos positivos en los indicadores operativos y de sostenibilidad, especialmente en términos de comunicación con el cliente y la posibilidad de vender y prestar servicios en línea, pero persisten barreras relacionadas con las capacidades digitales internas y la formación del personal (Loyola et al., 2024).

1.2.5 Marco legal

En el marco normativo el desarrollo e implementación del sistema de gestión de información propuesto se enmarca dentro de la legislación colombiana que regula tanto el fomento empresarial como las obligaciones contables y tributarias. La Ley 905 de 2004 , que modifica la Ley 590 de 2000, establece el marco para la promoción y el desarrollo de las MiPymes, subrayando la importancia de su modernización y competitividad. Por su parte, el Decreto 2649 de 1993 reglamenta los principios de contabilidad generalmente aceptados, siendo relevante para el proyecto ya que un sistema de gestión de inventarios y proyectos debe garantizar la integridad de la información que servirá de base para los registros contables. Finalmente, el Decreto 1333 de 1996 , que reglamenta el Estatuto Tributario, es pertinente en la medida en que la correcta valoración y control de inventarios y costos de proyectos tiene implicaciones fiscales directas para la empresa. El cumplimiento de estas normativas asegura que la solución propuesta no solo sea funcional, sino también legalmente sólida y alineada con las obligaciones empresariales en Colombia.

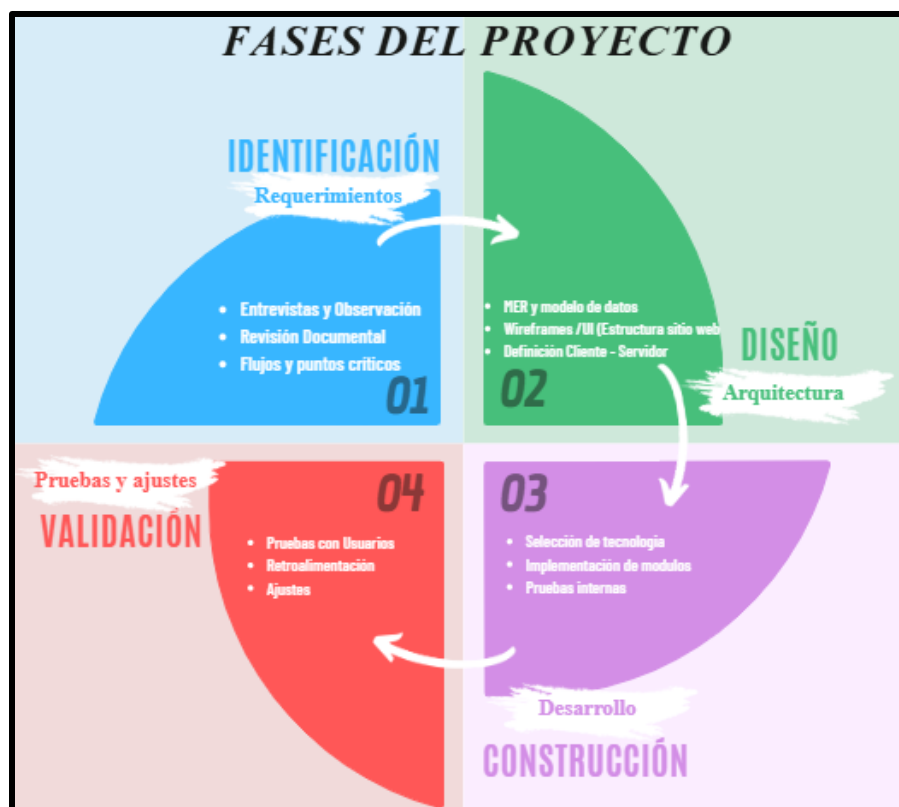
2. Metodología

Se desarrollo una investigación de enfoque cualitativo, con un tipo estudio de caso, debido a que se establecerá un modelo para una empresa en particular para los requerimientos en los módulos y una metodología ágil de desarrollo Scrum, es decir los posibles cambios inmediatos de los errores, esto para el diseño e implementación de la aplicación web. Lo cual permite un enfoque en la comprensión de las necesidades de la empresa como en la flexibilidad y eficiencia del desarrollo tecnológico. Las categorías a trabajar a lo largo de la investigación serán: la independiente que es el aplicativo web y la dependiente el mejoramiento de la gestión de datos, la documentación e inventarios.

Teniendo en cuenta los objetivos específicos que se plantean para obtener el objetivo general, se establecerán las siguientes fases: Identificación, Diseño, Construcción y Validación. Ver figura 1.

Figura 1.

Ciclo de fases del proyecto.



2.1 Fase de identificación

Técnicas utilizadas: Entrevistas, observación directa y análisis documental.

Actividades:

- I1. Reuniones con personal administrativo de las áreas de inventarios, mano de obra, proveedores y proyectos.
- I2. Revisión de documentos físicos y digitales actualmente utilizados.
- I3. Identificación de flujos de trabajo, puntos críticos en la gestión de la información y movimientos de documentación

Resultados esperados:

- I4. Documento de especificación de requerimientos por módulos o áreas.
- I5. Diagrama de flujo de procesos actuales.

2.2 Fase de Diseño

Técnicas utilizadas: Modelado de datos, wireframe (Interfaz de usuario) y arquitectura de software.

Actividades:

- D1. Diseño del modelo entidad-relación (MER) para la base de datos.
- D2. Creación de wireframes para la interfaz de usuario (UI).
- D3. Definición de la arquitectura de la aplicación: cliente-servidor, con enfoque modular.

Resultados esperados:

- D4. Documento de diseño funcional.
- D5. Prototipo navegable de la aplicación.
- D6. Diagrama de la base de datos (Inventario, colaboradores, contratos...)

2.3 Fase de construcción

Técnicas utilizadas: Análisis comparativo de tecnologías, desarrollo incremental.

Actividades:

- C1. Evaluación de lenguajes de programación y frameworks (Herramientas) adecuados (por ejemplo, Python con Django, JavaScript con React, o PHP con Laravel).
- C2. Implementación de módulos funcionales (gestión de inventarios, proveedores, proyectos y mano de obra).
- C3. Pruebas internas y validación con usuarios.

Resultados esperados:

- C4. Aplicación web funcional en entorno de pruebas.
- C5. Informe de selección tecnológica.
- C6. Manual técnico y de usuario básico.

2.4 Fase de Validación y Retroalimentación

Actividades:

- V1. Presentación del prototipo a los usuarios finales.
- V2. Recolección de observaciones y sugerencias.
- V3. Ajustes a la estructura y funcionalidades.

Resultados esperados:

- V4. Informe de retroalimentación.
- V5. Versión final ajustada de la aplicación

3. Fase de identificación

En la identificación de necesidades y dificultades en el manejo de la información en American Carpas 1 SAS, se determinó por medio de encuestas virtuales y la observación directa del proceso en las diferentes áreas (administrativa y operativa), lo que permitió reconocer vacíos y duplicidad de la información. Se evidenció una dependencia muy alta en el uso de múltiples hojas de cálculo para procesos de inventarios, producción, datos de trabajadores y proyectos, además de un manejo confuso en la obtención de documentos que se encuentran en subcarpetas, así como la gestión de proveedores mediante diferentes canales informales. Con base en estos hallazgos se establecieron diagramas de flujo y se definieron los módulos clave a intervenir: trabajadores, inventarios, proyectos/contratos y proveedores

3.1 Necesidades

Para caracterizar la información, se diseñó y distribuyó una encuesta virtual a los trabajadores de las áreas de administrativos, operativa y comercial. El cuestionario consultó sobre los tipos de datos que gestionan, la forma como lo registran, los problemas que encuentran, la información crítica que debe estar actualizada y los procesos que les gustaría que fuesen más automáticos. Del cual se obtuvieron cuatro respuestas, compuestas por puntos claves que a su vez establecen un análisis dirigido al mejoramiento del manejo de la información.

Como se puede observar en la tabla 1; las encuestas coincidieron en que la información se dispersa en varios archivos de Excel expuestos en carpetas y subcarpetas, lo que genera mayor tiempo en la búsqueda de datos y aumenta el riesgo de cometer errores en las diferentes versiones que se crean o en la digitación manual. En segundo término, se identifica como información crítica los datos de los proyectos, inventarios y del personal; donde cualquier desactualización repercute directamente en la estructura de la materia prima o en la liquidación de nómina. Por último, al preguntar qué procesos

deberían ser automatizados, dieron a conocer tres focos importantes: el control de inventario, seguimiento de los proyectos, gestión del personal y proveedores, todos expuestos en las hojas de cálculo que exigen un uso constante y de revisión manual.

Tabla 1.

Cuadro comparativo de respuestas de la encuesta sobre el manejo de la información.

Categoría de análisis	Respuestas individuales (ejemplos)	Coincidencias / Necesidades identificadas
Problemas frecuentes en el manejo de información	• Uso de varios archivos en carpetas • Demoras para encontrar datos • Errores por digitación manual • Información no actualizada	Duplicidad y dispersión de archivos en Excel, alto riesgo de errores y pérdida de tiempo en la búsqueda.
Información crítica que debe estar actualizada	• Datos de proyectos • Inventarios • Información de trabajadores • Datos de proveedores	Necesidad de mantener al día datos de: proyectos, inventario, personal y proveedores.
Procesos que se desean automatizar	• Control de inventario • Gestión de proyectos • Registro de personal y horas extras • Consultas a proveedores	Automatización prioritaria en inventarios, proyectos y gestión de personal.
Errores o reprocesos principales	• Errores en digitación • Demoras en tiempos de búsqueda • Información duplicada • Dificultad en verificación de datos	Necesidad de digitalización confiable, centralización de información y trazabilidad.

Estas coincidencias, relacionándolos con la observación directa en el área comercial y administrativa, donde se reconoció el proceso de órdenes de producción impresas en papel, los inventarios divididos en área de lonas y área de estructura, más los documentos repartidos en múltiples carpetas, reforzaron la necesidad de consolidar los cuatro módulos a intervenir para mejorar el manejo de la información (Inventarios, trabajadores, proveedores y proyectos). En consecuencia, se construyeron los requerimientos necesarios para cada módulo.

3.2 Módulo de Inventarios

Se establecieron requerimientos orientados a garantizar las limitaciones detectadas en la gestión actual de insumos, especialmente en lo referente a lonas y estructuras. A nivel funcional, se consideró indispensable garantizar el registro, actualización y consulta de los movimientos de inventario, la asociación de salidas a órdenes de producción, la definición de niveles mínimos con alertas automáticas, así como la generación de reportes y el mantenimiento de un historial (kárdex) que asegure la trazabilidad de materiales. Por su parte, los requerimientos no funcionales priorizan aspectos de seguridad mediante perfiles de usuario, usabilidad con consultas rápidas y compatibilidad móvil que facilite la verificación desde diferentes lugares. La tabla 2 presenta la síntesis de los requerimientos más relevantes, mientras que el detalle completo se encuentra en los anexos.

Tabla 2.

Requerimientos funcionales y no funcionales del módulo de inventarios.

Tipo de requerimiento	Aspecto	Descripción sintetizada
Funcional	Gestión de inventario	Registrar, consultar, actualizar y eliminar insumos en lonas y estructuras.
Funcional	Asociación a órdenes	Cada salida debe estar vinculada a una orden de producción registrada.
Funcional	Control de stock	Definir niveles mínimos y generar alertas automáticas cuando se superen umbrales.
Funcional	Kardex	Mantener un registro histórico de entradas/salidas con fecha, cantidad, responsable y proyecto.
Funcional	Consultas específicas	Ver disponibilidad en tiempo real por atributos (tipo, color, ancho, calibre, medida).
Funcional	Reportes	Emitir reportes personalizados de stock, mínimos, histórico y consumo por proyecto.
No Funcional	Seguridad	Acceso gestionado por roles y perfiles (ej. solo bodega da salida de material).
No Funcional	Auditoría	Transacciones registradas de forma inalterable con usuario, fecha y hora.
No Funcional	Usabilidad	Consultas de disponibilidad en máximo dos pasos (búsqueda + resultado).
No Funcional	Compatibilidad	Acceso desde dispositivos móviles con interfaz responsiva.

No Funcional	Escalabilidad	Soporte hasta 50.000 movimientos por año sin afectar rendimiento.
No Funcional	Respaldo e integración	Respaldo automático diario e integración futura con compras y producción.

3.3 Módulo de Trabajadores

Los requisitos identificados tienen como objetivo optimizar la gestión integral del personal de la empresa, mejorando las limitaciones actuales de archivos dispersos y control manual. A nivel funcional, se estableció la necesidad de registrar, consultar y actualizar la información de los trabajadores en secciones específicas (datos personales, laborales, afiliaciones y roles dentro del sistema), con soporte para adjuntar certificados, controlar asistencia según proyectos, gestionar el estado del empleado (activo, retirado o en licencia) y mantener un historial de cambios que garantice trazabilidad. Igualmente, se contemplan alertas automáticas sobre el vencimiento de certificados y la capacidad de generar reportes o exportaciones de información en formatos digitales.

En cuanto a los requerimientos no funcionales, se resaltan la seguridad y confidencialidad de los datos mediante cifrado y controles de acceso, la usabilidad de la interfaz con operaciones rápidas, la rapidez de respuesta en las consultas, así como la compatibilidad multiplataforma y móvil, también se destaca la escalabilidad hasta miles de registros, la auditoría de todas las acciones relevantes y la integridad de la información para evitar modificaciones accidentales. La tabla 3 demuestra los requerimientos mencionados.

Tabla 3.

Requerimientos funcionales y no funcionales del módulo de trabajadores.

Tipo de requerimiento	Aspecto	Descripción sintetizada
Funcional	Gestión de empleados	Registro, consulta y actualización de datos personales, laborales, afiliaciones y roles.

Funcional	Certificados	Alertas automáticas sobre vencimiento próximo de certificados del personal.
Funcional	Documentos y reportes	Generación de reportes y exportación de datos con archivos adjuntos por empleado o grupos.
Funcional	Asistencia	Control de asistencia de empleados según los proyectos asignados.
Funcional	Estados y trazabilidad	Gestión de estados (activo, licencia, retirado) y bitácora de modificaciones en la información.
No Funcional	Seguridad	Cifrado de datos y accesos restringidos por roles de usuario.
No Funcional	Usabilidad	Interfaz intuitiva que permita registrar, consultar o editar datos en máximo 3 clics.
No Funcional	Rendimiento	Consultas rápidas de información con tiempos de respuesta menores a 2 segundos en promedio.
No Funcional	Compatibilidad y movilidad	Acceso desde navegadores principales y dispositivos móviles con diseño responsivo.
No Funcional	Escalabilidad y auditoría	Soporte para miles de registros y registro inalterable de acciones de usuario.

3.4 Módulo de proveedores

Se busca centralizar la gestión de los aliados estratégicos de la empresa, superando la actual dispersión de información que dificulta el control de insumos y la trazabilidad de compras. A nivel funcional, permitirá registrar, consultar y actualizar las fichas de proveedores, incluyendo múltiples contactos y documentos asociados, así como mantener el historial de pedidos vinculados. En cuanto a los requerimientos no funcionales, se priorizan la seguridad y control de accesos, la auditoría de modificaciones en la ficha del proveedor, y la usabilidad de filtros intuitivos que agilicen búsquedas en

tiempo real. Se garantiza también rendimiento en las consultas, escalabilidad para miles de registros, y compatibilidad móvil. La Tabla 4 presenta estos requerimientos.

Tabla 4.

Requerimientos funcionales y no funcionales del módulo de proveedores.

Tipo de requerimiento	Aspecto	Descripción sintetizada
Funcional	Gestión de proveedores	Registrar, consultar, actualizar y eliminar proveedores (CRUD).
Funcional	Datos básicos	Capturar razón social, NIT, contacto principal, teléfonos, correo, dirección y sitio web.
Funcional	Contactos múltiples	Asociar varios asesores por proveedor diferenciando roles (ventas, soporte).
Funcional	Historial de pedidos	Vincular a cada proveedor pedidos con cantidad, fecha, valor y estado.
Funcional	Búsquedas y filtros	Búsqueda rápida por nombre, material, estado o ubicación.
Funcional	Estado del proveedor	Gestionar estados: activo, suspendido o retirado.
Funcional	Documentos y certificaciones	Adjuntar RUT, certificaciones, contratos u otros soportes.
Funcional	Integración con inventario/pedidos	Mostrar insumos de un proveedor en stock y pedidos asociados.
No Funcional	Seguridad	Controles de acceso por roles para gestionar modificaciones.
No Funcional	Auditoría	Registro de altas, bajas y cambios con usuario, fecha y hora.
No Funcional	Usabilidad	Filtrado en máximo 2 pasos con formularios intuitivos.
No Funcional	Rendimiento	Búsquedas por nombre o material ≤ 2 segundos.
No Funcional	Escalabilidad	Soporte de al menos 5.000 proveedores sin degradación de tiempos.
No Funcional	Compatibilidad	Acceso responsivo en móviles y navegadores principales.
No Funcional	Exportación de reportes	Reportes exportables a formatos estándar (PDF y Excel).

3.5 Módulo de proyectos

Se establece como el eje central de la aplicación, al articular la relación entre clientes, pedidos y consumo de inventarios. Los requerimientos funcionales incluyen el registro y gestión completa de

proyectos con datos básicos, responsables y fecha, la trazabilidad cliente - proyecto - pedido, y la integración de cronogramas y documentos clave. De igual manera, el sistema permitirá el seguimiento automático del avance porcentual del proyecto, la asignación de responsables de producción y la generación de reportes personalizados, todo ello garantizando una trazabilidad integral. En cuanto a los requerimientos no funcionales, se destacan la seguridad en la creación y modificación de proyectos, la usabilidad mediante vistas unificadas cliente–proyecto–pedido, así como el rendimiento de consultas rápidas, la compatibilidad multiplataforma, y el soporte de reportes exportables y notificaciones automáticas ante hitos críticos. La tabla 5 presenta la síntesis de los requerimientos más relevantes, mientras que el detalle completo se encuentra en los anexos.

Tabla 5.

Requerimientos funcionales y no funcionales del módulo de proyectos.

Tipo de requerimiento	Aspecto	Descripción sintetizada
Funcional	Gestión de proyectos	Registrar, consultar y actualizar proyectos asociados a clientes.
Funcional	Trazabilidad	Mantener relación cliente → proyecto → pedido.
Funcional	Datos y estatus	Incluir nombre, ubicación, responsables, fechas y estados.
Funcional	Gestión documental	Vincular documentos y cronogramas de proyectos.
Funcional	Seguimiento automático	Actualizar estatus y calcular avance porcentual según pedidos.
Funcional	Reportes	Reportes personalizados por cliente, proyecto o pedido.
No Funcional	Seguridad	Acceso restringido a usuarios autorizados.
No Funcional	Usabilidad	Vista unificada cliente–proyecto–pedido
No Funcional	Rendimiento	Consultas rápidas ≤ 3 segundos.
No Funcional	Disponibilidad	Disponibilidad mínima del 99% del tiempo hábil.
No Funcional	Escalabilidad	Soportar al menos 10.000 proyectos activos sin degradación.
No Funcional	Compatibilidad	Acceso desde navegadores y móviles (responsive).
No Funcional	Notificaciones	Alertas en fechas críticas o atrasos de pedidos.
No Funcional	Reportes	Exportar reportes a PDF y Excel.

De lo anterior se establece una identificación en los procedimientos actuales que presenta cada uno de los módulos (Inventarios, Trabajadores, Proveedores y Proyectos), generando un diagrama de

flujo para cada uno y es allí donde se identifican los requerimientos y necesidades para llevar a cabo un mecanismo más automático en los pasos que compone cada sección, ver el Anexo A compuesta por las figuras 43, 44, 45 y 46.

4. Fase de diseño

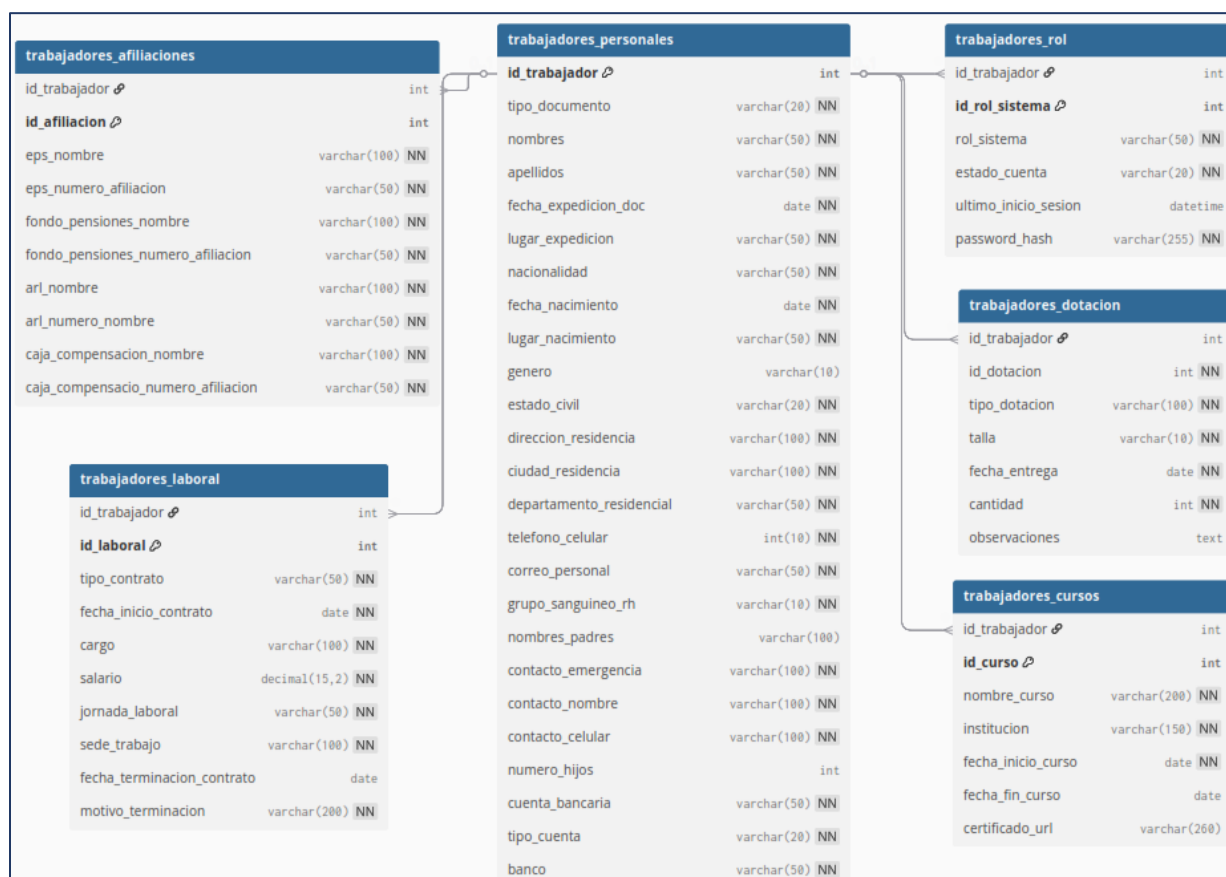
En el diseño de la aplicación de web para la gestión de base de datos de American Carpas, se establece la estructura de modelos Entidad – Relación en los módulos identificados por las necesidades de la empresa (Trabajadores, Proveedores, inventarios, Proyectos). Esto facilita el diseño lógico de las bases de datos para generar una configuración en la arquitectura, las funciones específicas y sus conexiones para integrarlas, con el objetivo primordial de facilitar el acceso, la manipulación y el desarrollo eficiente de los datos empresariales. Este sistema está diseñado para ser una herramienta de apoyo para los administrativos y jefes de área, permitiéndoles interactuar con la base de datos de manera intuitiva y adaptada a sus actividades específicas.

4.1 Diseño de Base de Datos (Modelo Entidad-Relación)

Como primer módulo se diseña la base de datos centrada en la información de los “Trabajadores”, el cual se observa en la figura 2. Conformado por seis entidades principales y relacionadas entre sí, que almacenan aspectos laborales y personales que se utilizan en la empresa. Se definen factores primarios como el nombre y número de identificación de los trabajadores, esto con el fin de relacionar información personal, de afiliaciones, cursos, accesos y rol en el sistema dotaciones y capacitaciones.

Figura 2.

Modelo entidad relación del módulo de trabajadores.



Así mismo, se establece el diseño para la entidad relación del módulo de información de “Inventarios”, donde se identifica información de las dos subáreas (Ornamentación y carpas) que hacen parte el área de producción. Se identifica en la figura 3 relaciones del grado de información para los elementos en cada subárea, el tipo, medida, calibre, color y ancho de la materia prima. Se representan claves primarias y foráneas relacionándose de acuerdo a la congruencia de los datos. El módulo permite una gestión detallada y separada de los dos tipos principales de materiales: las estructuras metálicas y las lonas, permitiendo un seguimiento específico de sus existencias, ubicaciones, costos y proveedores.

Figura 3.

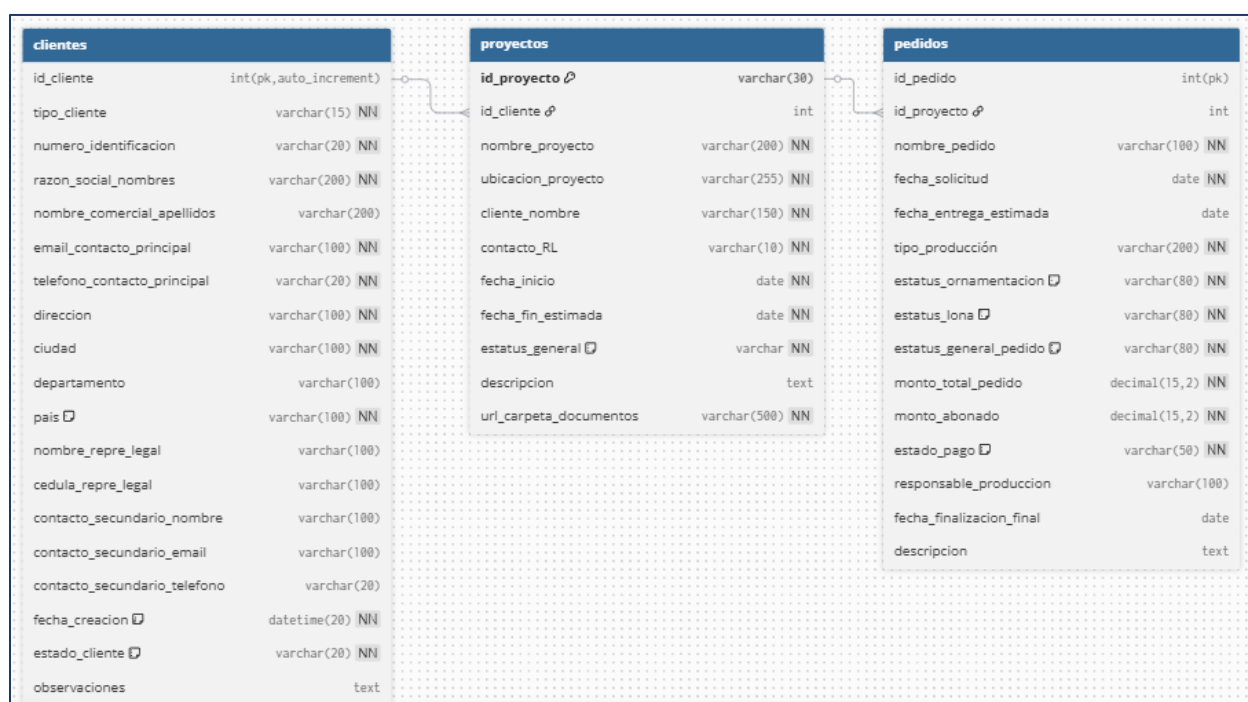
Modelo entidad relación del modulo de inventarios.



Del mismo modo, se elabora el reconocimiento de los requerimientos y las necesidades del sistema que se usa, para reorganizar los proceso para los proyectos o clientes de la empresa, el cual se observa en la figura con los datos relacionados, que caracterizan los diferentes proyectos que se estan desarrollando Figura 4.

Figura 4.

Modelo entidad relación del modulo de Proyectos.



4.2 Arquitectura del sistema

Se definen las herramientas y aplicaciones que permiten la interacción de archivos creando así la página web y su interacción entre cliente – servidor, entre ello se define lo siguiente:

- **Visual Studio Code:** Es el entorno de software que se utiliza para escribir, depurar y gestionar el código de tu proyecto. No ejecuta la aplicación por sí solo, sino que te proporciona el entorno de trabajo con herramientas. Visual Studio Code (s.f)

- **Django:** Es una colección de bibliotecas y herramientas, un marco de trabajo completo que sigue el patrón de MTV Models – Templates – Views (Modelos - Plantillas – Vistas) y ORM (Mapeo Objeto Relacional) escrito en Python, que te permite construir la aplicación web de manera rápida organizada y eficiente con una serie de archivos principales, ver en la figura 5 y 6. Django Software Foundation. (2024)

Figura 5.

Diagrama de archivos principales y flujo de interacción en Django



- **MySQL:** Sistema gestor de bases de datos relacional que almacena y gestiona los datos de forma estructurada mediante tablas. Utiliza SQL para consultas y operaciones. Oracle Corporation. (2024)
- **Python:** Lenguaje de marcado estándar para estructurar el contenido de páginas web (títulos, párrafos, formularios, enlaces). En Django se usa dentro de templates para generar páginas dinámicas. DuBois, P. (2013). MySQL (5th ed.). Addison-Wesley Professional.
- **Html:** Conjunto de herramientas predefinidas de CSS y JavaScript para crear interfaces web responsivas. Proporciona componentes prediseñados (botones, menús, formularios) que se adaptan a diferentes tamaños de pantalla. Duckett, J. (2011). HTML and CSS: Design and Build Websites. Wiley.
- **Bootstrap:** Es el conjunto de bibliotecas para crear interfaces web responsivas. Proporciona componentes prediseñados (botones, menús, formularios, grids) que se adaptan a diferentes tamaños de pantalla. Bootstrap Team. (2024).
- **Css:** Lenguaje para definir el estilo visual de documentos HTML: colores, tipografías, márgenes, espaciados y animaciones. Personaliza la apariencia de las páginas web. Mozilla Developer Network (MDN). (2024)
- **Mockups:** Prototipos visuales estáticos de la interfaz de usuario. Se crean en la fase de diseño para definir cómo se verán las pantallas antes de implementarlas en código. Initium Software (2024).

Figura 6.

Diagrama de interacción de las herramientas para la aplicación web



4.3 Diseño de Interfaz de Usuario (Wireframes/Mockups)

Se desarrolla el diseño para la interacción entre usuario y el sistema, en esta etapa se centra en la creación de prototipos visuales que representan la estructura, disposición y comportamiento de cada pantalla de la aplicación, antes de proceder con la implementación en código. Siendo así, se establecen los componentes para visualización de módulos a trabajar, el diseño de la experiencia del usuario, establecer la entidad visual (colores institucionales), facilitar la intercomunicación del sistema y jefes de área, reducir tiempos de ejecución de actividades, entre otros.

Para la creación de los mockups se implementa un sistema que actualmente se utiliza, la herramienta de la IA en la cual se interpreta pantallas para diferentes espacios en el sistema, entre los

cuales se encuentran los siguientes: visualización para home (Inicio), para ingreso de usuario (Login), espacio de módulos a trabajar y sus respectivos enrutamientos para cada uno según sus funciones. De lo anterior se mantiene una personalización de estructura que contiene lo siguiente:

- **Disposición de elementos:** ubicación de menús, botones, formularios y contenido.
- **Jerarquía visual:** tamaños, colores y contrastes que guían la atención del usuario.
- **Componentes interactivos:** botones, enlaces, campos de entrada, etc.

Del mismo modo se establece el flujo de navegación para describir el recorrido que realiza el usuario a través de las diferentes pantallas del sistema para completar una tarea específica. **Ver anexo**

B.

1. **Acceso inicial:** El usuario ingresa a la URL del sistema.
2. **Autenticación:** Si no está autenticado, debe iniciar sesión con credenciales válidas.
3. **Dashboard:** Una vez autenticado, accede al panel principal con los 4 módulos.
4. **Selección de módulo:** El usuario elige entre Trabajadores, Proveedores, Inventarios o Proyectos.
5. **Operaciones CRUD:** Realiza operaciones de Crear, Leer, Actualizar o Eliminar registros.
6. **Persistencia:** Los datos se almacenan en la base de datos MySQL mediante el ORM de Django.
7. **Respuesta:** El sistema renderiza la vista correspondiente con confirmaciones o mensajes.
8. **Ciclo:** El usuario puede continuar operando o cerrar sesión.

En consecuencia, en la Figura 7 se define una identificación visual que está anclada a un diseño específico y característico que tiene la página, allí se seleccionan los colores institucionales que representan la identidad de la Universidad La Gran Colombia y American Carpas, se tiene lo siguiente:

✓ **Verde principal** (#059669)



✓ **Azul oscuro principal** (#1e40af)



✓ **Blanco** (ffffff) para fondo principal



Figura 7.

Identificación de colores en la página web. visual

```

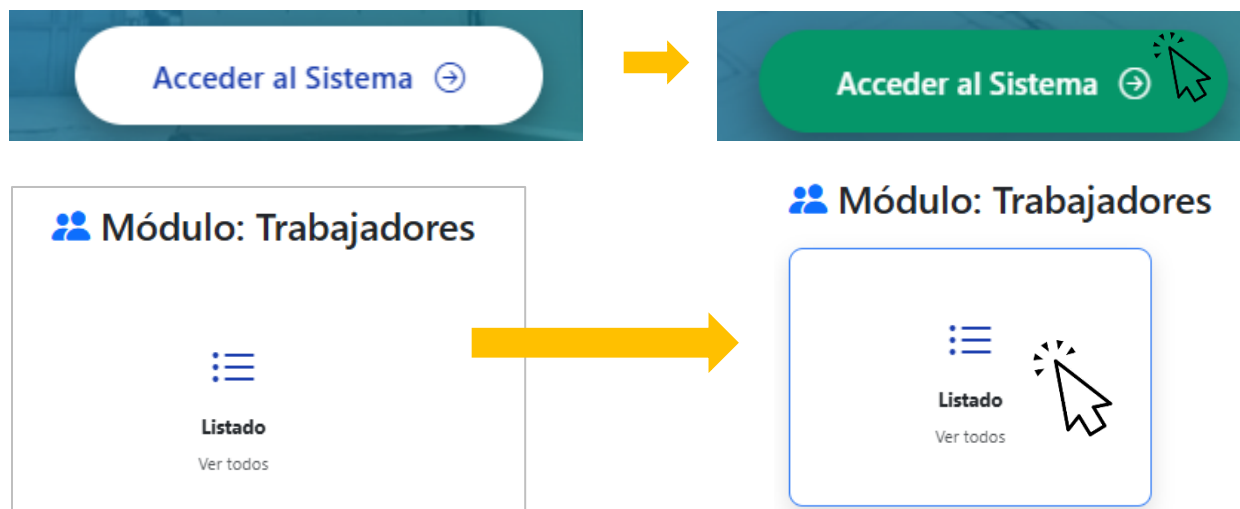
1  /* =====
2  ESTILOS PERSONALIZADOS - DISEÑO HÍBRIDO
3  American Carpas 1 SAS + Universidad La Gran Colombia
4  ===== */
5
6  /* --- Variables de Colores --- */
7  :root {
8      /* Colores American Carpas (Azul) */
9      --color-azul-primary: #1e40af; /* Azul oscuro principal */
10     --color-azul-secondary: #3b82f6; /* Azul medio */
11     --color-azul-light: #60a5fa; /* Azul claro */
12
13     /* Colores Universidad (Verde) */
14     --color-verde-primary: #059669; /* Verde principal */
15     --color-verde-secondary: #10b981; /* Verde medio */
16     --color-verde-light: #34d399; /* Verde claro */
17
18     /* Colores neutros */
19     --color-blanco: #ffffff;
20     --color-gris-claro: #f3f4f6;
21     --color-gris-medio: #6b7280;
22     --color-gris-oscuro: #1f2937;
23
24     /* Colores de estado */
25     --color-exito: var(--color-verde-primary);
26     --color-advertencia: #f59e0b;
27     --color-peligro: #dc2626;
28     --color-info: var(--color-azul-secondary);
29 }

```

Por consiguiente, se define la tipografía y los componentes visuales que proporciona la legibilidad de las diferentes pantallas y la compatibilidad con navegadores web Figura 8. De manera que se especifican elementos como los Botones con diseño redondeado y efecto hover (aplica estilos a un elemento cuando el puntero del ratón pasa sobre él, permitiendo cambios de color, tamaño, o animaciones para indicar interactividad), los formularios con campos con bordes definidos y etiquetas claras, las tarjetas para agrupar información relacionada y la iconografía (iconos) para mejorar la comprensión visual.

Figura 8.

Interacción de componentes visuales en pantalla



De acuerdo a los diseños detallados para el sistema general se implementa el Mockups con los espacios adecuados para la interacción con el usuario y la función que requieren siendo así, se comprende los siguientes Screenshots (Figura 9, Figura 10 y Figura 11)

Figura 9.

Mockup Home (Inicio) para sistema web.



Figura 10.

Mockup para Login (Iniciar Sesión).

The mockup shows a login interface for 'American Carpas 1 SAS'. It features a central white card with the title 'Iniciar Sesión'. Below the title are two input fields: 'Usuario' with the placeholder 'Tu correo o usuario' and 'Contraseña' with a masked password '*****'. A blue 'Ingresar' button is positioned below the password field. At the bottom of the card, there is a link: '¿Olvidaste tu contraseña? Recupérala aquí'.

Figura 11.

Mockup de interacción de módulos (Trabajadores, Inventarios, Proveedores y proyectos)

The mockup displays a dashboard for 'American Carpas 1 SAS' with the subtitle 'Sistema de Gestión de Datos'. The top right corner shows the user 'j.perez' as 'Administrador' with a 'Cerrar sesión' button. The main content area is titled 'Bienvenido' and includes the instruction 'Selecciona un módulo para continuar. La visibilidad depende de tus permisos.' Below this, there are four module cards, each with a blue header and a white body containing an icon, the module name, and a brief description:

- Trabajadores**: Gestión de personal y novedades.
- Inventarios**: Entradas, salidas y existencias.
- Proveedores**: Altas, contratos y contactos.
- Proyectos**: Seguimiento y planificación.

5. Fase de desarrollo (construcción)

Se desarrolla el proceso técnico de implementación de la página web, la cual está estructurada por módulos funcionales para la empresa American Carpas 1 SAS. Cada uno de los módulos se organiza

siguiendo una estructura consistente que permite entender tanto las decisiones técnicas como la implementación práctica para el usuario en la empresa.

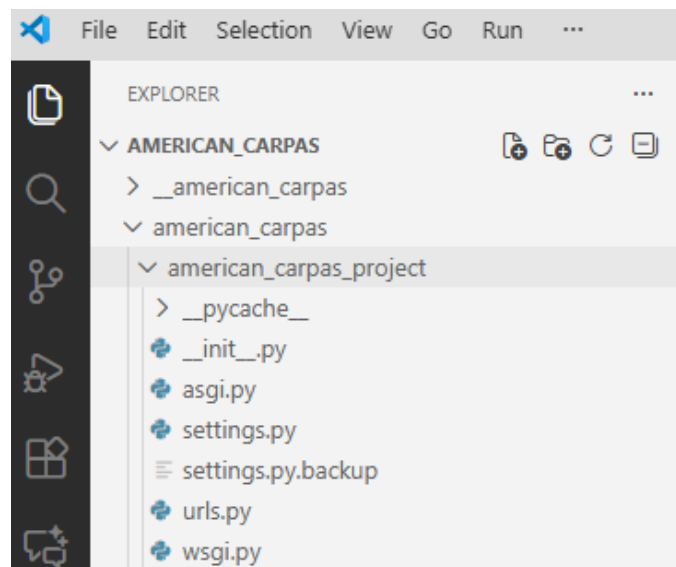
5.1 Configuración general del proyecto

5.1.1 Creación del proyecto Django

El proyecto se inició utilizando el programa de Django, quien permite la creación de la página web por medio de un conjunto de herramientas, bibliotecas y código predefinido de un alto nivel escrito en Python que impulsa el desarrollo rápido y diseño práctico en el entorno de Visual Studio Code, que funciona como aplicativo de codificación (Ver Figura 12). Su instalación y configuración se desarrolló con lo siguiente:

Figura 12.

Estructura inicial de Django en Visual Studio Code



5.1.2 Configuración de *Settings.py*

Se realiza en el archivo *Settings.py* la configuración general de la aplicación con aspectos relevantes como a conexión de base de datos (MySQL), las llaves secretas, instalación de apps, el enrutamiento de archivos, las constantes del proyecto, lo que representa las configuraciones críticas:

a) Base de datos MySQL

MySQL fue seleccionada por la robustez para simplificar el trabajo, por su capacidad de flujo en datos y compatibilidad con Railway como aplicación de despliegue para que los usuarios puedan usarla.

(Ver Figura 13)

Figura 13.

Configuración de MySQL como base de datos

```
13  from pathlib import Path
14  import os
158  # CONFIGURACIÓN ROBUSTA PARA RAILWAY
159  # Si estamos en Railway (producción), usar volume persistente
160  # Si estamos en desarrollo local, usar carpeta media normal
161  if os.environ.get('RAILWAY_ENVIRONMENT'):
162      # En Railway: usar volume montado en /data
163      MEDIA_ROOT = '/data/media'
164
```

b) Archivos estáticos y multimedia

En Django se utiliza una biblioteca (Whitenoise) que funciona como servidor interno en la codificación del aplicativo, el cual permite servir los archivos estáticos y de multimedia (Formatos de imágenes y videos), dando eficiencia en el mismo entorno, sin necesidad de crear un servidores adicional fuera de este entorno, lo cual podemos observar en la figura 14.

Figura 14.*Configuración de archivos estáticos y multimedia*

```

134 # =====
135 # STATIC FILES (CSS, JavaScript, Images)
136 # =====
137
138 STATIC_URL = '/static/'
139
140 # Directorios donde Django buscará archivos estáticos adicionales
141 STATICFILES_DIRS = [
142     BASE_DIR / "static",
143 ]
144
145 # Directorio donde se recopilarán todos los archivos estáticos en producción
146 STATIC_ROOT = BASE_DIR / "staticfiles"
147
148 # Configuración de WhiteNoise para servir archivos estáticos
149 STATICFILES_STORAGE = 'whitenoise.storage.CompressedManifestStaticFilesStorage'
150
151
152 # =====
153 # MEDIA FILES (Uploads de usuarios)
154 # =====
155
156 MEDIA_URL = '/media/'
157
158 # CONFIGURACIÓN ROBUSTA PARA RAILWAY
159 # Si estamos en Railway (producción), usar volume persistente
160 # Si estamos en desarrollo local, usar carpeta media normal
161 if os.environ.get('RAILWAY_ENVIRONMENT'):
162     # En Railway: usar volume montado en /data
163     MEDIA_ROOT = '/data/media'

```

c) Aplicaciones instaladas

Se estableció las instalaciones de las aplicaciones, las cuales permiten desde la base, la configuración automática de bases de datos, los modelos, vistas, la funcionalidad de habilitar la autenticación, administración de sesiones y la integración de las mismas. Del mismo modo, se definen las aplicaciones especiales (Módulos) del proyecto (Trabajadores, inventario, proveedores y proyectos). Por otro lado, se comprende la configuración para incorporar la plataforma que funciona como nube de despliegue de la aplicación web para la empresa, en la figura 15 se identifica las aplicaciones instaladas y configuración para asociar las variables de Railway.

Figura 15.

Instalación de aplicaciones e integración de variables Railway (código)

```

35 # Application definition
36
37 INSTALLED_APPS = [
38     'django.contrib.admin',
39     'django.contrib.auth',
40     'django.contrib.contenttypes',
41     'django.contrib.sessions',
42     'django.contrib.messages',
43     'django.contrib.staticfiles',
44     'django.contrib.humanize',
45     'trabajadores',
46     'proveedores',
47     'proyectos',
48     'inventario',
49 ]

```

```

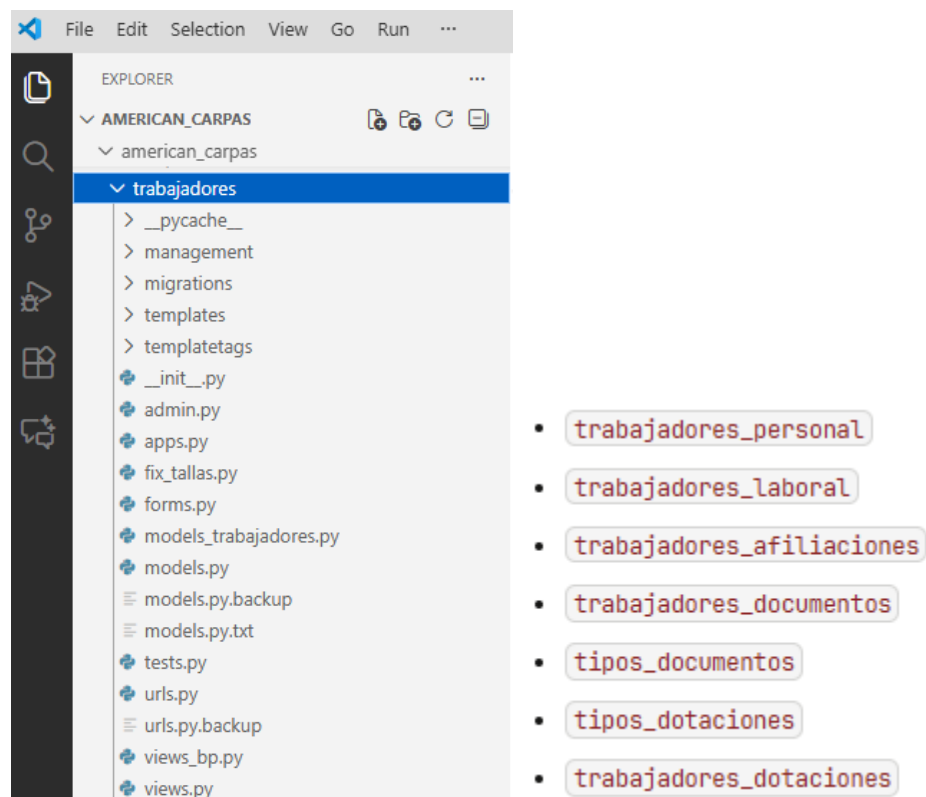
86 # Configuración que lee las variables de entorno de Railway
87 DATABASES = {
88     'default': {
89         'ENGINE': 'django.db.backends.mysql',
90         'NAME': os.environ.get('DB_NAME', 'american_carpas_db'),
91         'USER': os.environ.get('DB_USER', 'root'),
92         'PASSWORD': os.environ.get('DB_PASSWORD', '@PGym2025'),
93         'HOST': os.environ.get('DB_HOST', 'localhost'),
94         'PORT': os.environ.get('DB_PORT', '3306'),
95         'OPTIONS': {
96             'charset': 'utf8mb4',
97             'init_command': "SET sql_mode='STRICT_TRANS_TABLES'",
98         },
99     }
100 }

```

5.2 Modulo de trabajadores

5.2.1 Creación del módulo de trabajadores

El módulo de trabajadores fue el primero en desarrollarse, del cual se establece una estructura general que también funciona como patrón para los otros módulos del proyecto, en la figura 16 se evidencia la creación del módulo con su respectiva organización de archivos.

Figura 16.*Creación y estructura de módulo de trabajadores*

5.2.2 Diseño del modelo de datos

En el módulo de trabajadores se implementó una estructura base de datos del cual se separa la información en diferentes entidades relacionales, generando un sistema que se enfoca en el análisis de los requerimientos realizados en la fase de identificación.

5.2.2.1 Modelo de trabajador-personal

El propósito de este modelo define el almacenamiento de la información personal e identificación de cada trabajador, se define el campo de “Id_trabajador” como la clave primaria representada por el número de documento del trabajador, garantizando unificación numérica y limite en diligenciamiento de dígitos. Del mismo modo se comprenden los campos de datos personales

(Nombres, apellidos, fecha de nacimiento, genero, estado civil, teléfono, correo, familia, entre otros).

Ver figura 17.

Figura 17.

Modelo Trabajador-persona en código

```

71 class TrabajadorPersonal(models.Model):
72     id_trabajador = models.CharField(
73         primary_key=True,
74         max_length=20,
75         validators=[doc_validator],
76         verbose_name="Número de Documento"
77     )
78     tipo_documento = models.CharField(max_length=20, choices=TIPO_DOCUMENTO_CHOICES)
79     nombres = models.CharField(max_length=50)
80     apellidos = models.CharField(max_length=50)
81     fecha_expedicion_doc = models.DateField()
82     lugar_expedicion = models.CharField(max_length=50)
83     nacionalidad = models.CharField(max_length=50)
84     fecha_nacimiento = models.DateField()
85     lugar_nacimiento = models.CharField(max_length=50)
86     genero = models.CharField(max_length=10, choices=GENERO_CHOICES, blank=True, null=True)
87     estado_civil = models.CharField(max_length=20, choices=ESTADO_CIVIL_CHOICES)
88     direccion_residencia = models.CharField(max_length=100)
89     ciudad_residencia = models.CharField(max_length=100)
90     departamento_residencia = models.CharField(max_length=50)
91     telefono_celular = models.CharField(max_length=20)
92     correo_personal = models.CharField(max_length=50)
93     grupo_sanguineo_rh = models.CharField(max_length=10)
94     nombres_padre = models.CharField(max_length=100, blank=True, null=True)
95     nombres_madre = models.CharField(max_length=100, blank=True, null=True)
96     nombre_contacto_emergencia = models.CharField(max_length=100)
97     numero_contacto_emergencia = models.CharField(max_length=40)
98     numero_hijos = models.IntegerField(blank=True, null=True, validators=[MinValueValidator(0)])
99     cuenta_bancaria = models.CharField(max_length=50)
100    tipo_cuenta = models.CharField(max_length=20, choices=TIPO_CUENTA_CHOICES)
101    banco = models.CharField(max_length=50)

```

Se resalta en la figura 18 el proceso de diligenciamiento de datos, donde se encuentran campos obligatorios, campos opcionales por privacidad (genero), así como la validación de números que no puede ser negativos (Hijos), la confirmación de información crítica (contactos de emergencia) e información necesaria de nómina. En consecuencia, se elaboró una personalización de modelo, en el cual se especifican las validaciones en los datos personales como los cursos-capacitaciones y documentación completa, generando alertas preventivas para que el usuario en la aplicación pueda evidenciar dichas insuficiencias e implementarlas.

Figura 18.*Formulario para registro de nuevo trabajador*

Nuevo Trabajador

Campos marcados con * son obligatorios (según restricciones NOT NULL).

Datos personales

Número de documento *	Departamento residencia *
Este será el identificador único del trabajador (5-20 dígitos)	
Tipo de documento *	Teléfono celular *
Nombres *	Correo personal *
Apellidos *	Grupo sanguíneo y RH *
Ej: O+, A-, B+, AB-	
Fecha expedición documento *	Nombres padre
dd/mm/aaaa	

5.2.2.2 Modelo de trabajador-laboral

Este modelo permite gestionar la información de historial de contratos, registros laborales de cada trabajador, permitiendo comprimir la información de varios contratos según su tipo, estado y experiencias. Así pues, se existe una relación de muchos a uno, lo que quiere decir que un trabajador puede tener varios contratos, permitiendo registrar terminaciones de contrato con motivo y validaciones según tipo de contrato, igualmente la renovación de contrato, los cambios de cargo e historial de campos de salario a un tiempo prologado de un trabajador de confianza. Ver figura 19.

Figura 19.*Modelo trabajador-laboral (código)*

```

126 class TrabajadorLaboral(models.Model):
127     id_laboral = models.AutoField(primary_key=True)
128     id_trabajador = models.ForeignKey(
129         TrabajadorPersonal,
130         on_delete=models.CASCADE,
131         db_column='id_trabajador',
132         related_name='registros_laborales'
133     )
134     tipo_contrato = models.CharField(max_length=50, choices=TIPO_CONTRATO_CHOICES)
135     fecha_inicio_contrato = models.DateField()
136     cargo = models.CharField(max_length=100)
137     salario = models.DecimalField(max_digits=15, decimal_places=2)
138     jornada_laboral = models.CharField(max_length=50, choices=JORNADA_CHOICES)
139     sede_trabajo = models.CharField(max_length=100)
140     fecha_terminacion_contrato = models.DateField(blank=True, null=True)
141     motivo_terminacion = models.CharField(blank=True, null=True, max_length=200)
142
143     class Meta:
144         db_table = 'trabajadores_laboral'
145         verbose_name = 'Registro Laboral'
146         verbose_name_plural = 'Registros Laborales'
147
148     def __str__(self):
149         return f"{self.id_trabajador.nombres} {self.id_trabajador.apellidos} - {self.cargo}"
150

```

5.2.2.3 Modelo de trabajador-afiliaciones

En dicho modelo (Figura 20) se busca integrar la información relacionada con los procesos de afiliación de seguridad de los trabajadores en Colombia, lo que incluye afiliación de ARL, de salud EPS, fondo de pensiones y compensación familiar. Esta documentación debe estar actualizada según la normativa laboral de la nación, además porque es la seguridad y salud del trabajador.

Figura 20.*Modelo trabajador-afiliaciones (código)*

```

152 class TrabajadorAfiliaciones(models.Model):
153     id_afiliacion = models.AutoField(primary_key=True)
154     id_trabajador = models.ForeignKey(
155         TrabajadorPersonal,
156         on_delete=models.CASCADE,
157         db_column='id_trabajador',
158         related_name='afiliaciones'
159     )
160     eps_nombre = models.CharField(max_length=100)
161     eps_numero_afiliacion = models.CharField(max_length=50)
162     fondo_pensiones_nombre = models.CharField(max_length=100)
163     fondo_pensiones_numero_afiliacion = models.CharField(max_length=50)
164     arl_nombre = models.CharField(max_length=100)
165     arl_numero_nombre = models.CharField(max_length=50)
166     caja_compensacion_nombre = models.CharField(max_length=100)
167     caja_compensacion_numero_afiliacion = models.CharField(max_length=50)
168
169     class Meta:
170         db_table = 'trabajadores_afiliaciones'
171         verbose_name = 'Afiliación'
172         verbose_name_plural = 'Afiliaciones'
173
174     def __str__(self):
175         return f"Afiliaciones de {self.id_trabajador}"

```

5.2.2.4 Modelo de trabajador-documentos

Con relación al modelo de documentación, que permite gestionar los documentos digitales asociados a cada trabajador, como los certificados, cursos, contratos, exámenes, cédulas, etc. (Ver figura 21). Allí se establece la organización de datos por años, el control de vigencia por ejemplo de los curso y categorización del tipo de documento. Al mismo tiempo, se ejecutan procesos de verificación de datos, vigencia del tiempo restante, tipos de archivos (Jpg, pdf, png, gif) y especifica en la sección de alerta para comunicarle al usuario los datos faltantes o de pronta fecha de vencimiento en los documentos.

Figura 21.*Modelo trabajador-documentos (código)*

```

553 class TrabajadorDocumento(models.Model):
554     """
555     Documentos adjuntos a cada trabajador (PDFs, imágenes, etc.)
556     """
557     id_documento = models.AutoField(primary_key=True)
558     id_trabajador = models.ForeignKey(
559         TrabajadorPersonal,
560         on_delete=models.CASCADE,
561         db_column='id_trabajador',
562         related_name='documentos',
563         verbose_name="Trabajador"
564     )
565     tipo_documento = models.ForeignKey(
566         TipoDocumento,
567         on_delete=models.PROTECT,
568         related_name='documentos_trabajadores',
569         verbose_name="Tipo de Documento",
570         help_text="Seleccione el tipo de documento del catálogo"
571     )

```

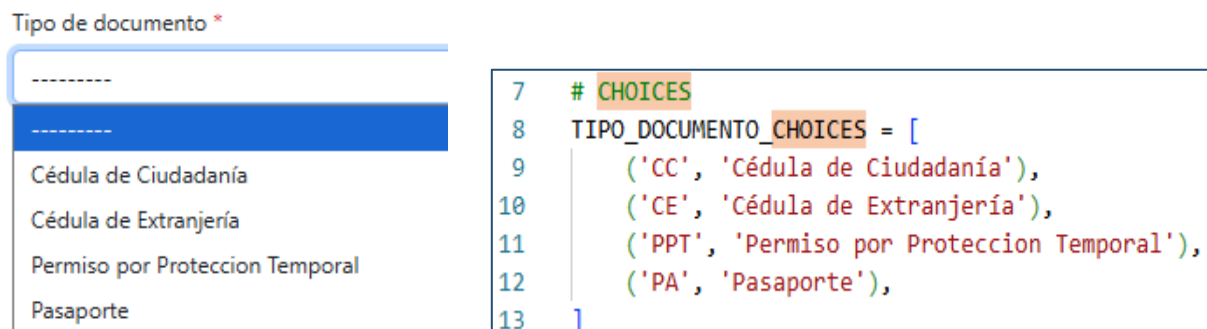


5.2.3 Catálogo de datos (Choices)

De acuerdo al diligenciamiento de datos se requieren y es de mayor facilidad de uso, los procesos de catálogos en cada campo, que permite al usuario interactuar para seleccionar el campo correspondiente a la información, en ello encontramos lo que se ilustra en la figura 22. Es decir, los catálogos generan automáticamente selectores de formularios en cada campo y valida los datos.

Figura 22.

Catálogo de datos en campos de información



5.2.4 Migraciones de base de datos

Una vez definidos los modelos, se establecen las migraciones que al mismo tiempo sincronizan la base de datos con el código de manera que permite aplicar y revertir cambios en la estructura de la base de datos y crear las tablas. Ver figura 23. Allí se genera el archivo de migración con *“python manage.py makemigrations trabajadores”* y confirmar con el comando *“python manage.py migrate trabajadores”* para las migraciones.

Figura 23.

Migración para modelo de trabajador. Personal

```

3  import django.db.models.deletion
4  from django.db import migrations, models
5
6
7  class Migration(migrations.Migration):
8
9      initial = True
10
11      dependencies = [
12      ]
13
14      operations = [
15          migrations.CreateModel(
16              name='TrabajadorPersonal',
17              fields=[
18                  ('id_trabajador', models.AutoField(primary_key=True, serialize=False)),
19                  ('tipo_documento', models.CharField(choices=[('CC', 'Cédula de Ciudadanía'),
20                  ('nombres', models.CharField(max_length=50)),
21                  ('apellidos', models.CharField(max_length=50)),
22                  ('fecha_expedicion_doc', models.DateField()),
23                  ('lugar_expedicion', models.CharField(max_length=50)),
24                  ('nacionalidad', models.CharField(max_length=50)),
25                  ('fecha_nacimiento', models.DateField()),
26                  ('lugar_nacimiento', models.CharField(max_length=50)),

```

5.2.5 Vistas (Views)

Las vistas ponen en funcionamiento la lógica del entorno de la página web para las operaciones, en este módulo da a conocer el CRUD, por medio de las funciones de Creas, Leer, actualizar y eliminar datos de trabajadores, representado en la figura 24.

Figura 24.

Vista basada en función de listado de trabajadores

```

397 # ====
398 # VISTAS CRUD - TrabajadorPersonal
399 # ====
400
401 def trabajador_list(request):
402     """
403     Vista de listado de trabajadores con búsqueda, filtros y paginación.
404
405     Parámetros GET:
406     - q: Búsqueda por nombre, apellido o documento
407     - tipo_documento: Filtro por tipo de documento
408     - estado_civil: Filtro por estado civil
409     - genero: Filtro por género
410     - ordenar: Campo por el cual ordenar (apellidos, nombres, id_trabajador)
411     """
412
413     # Obtener todos los trabajadores
414     trabajadores = TrabajadorPersonal.objects.all()

```

5.2.6 Urls (Enrutamiento)

Los urls ejecutan el mapeo que asigna las direcciones web (urls) a las funciones de vista correspondientes a la aplicación (ver figura 25), enrutando al usuario según las funciones que se van desarrollando, es decir, va desplazando al cliente de vista en vista según las acciones que toma en la página web.

Figura 25.

Enrutamiento de módulos (código)

```

74
75     # CRUD: TrabajadorPersonal (detalle, editar, eliminar)
76     path('<str:id_trabajador>', views.trabajador_detail, name='trabajador_detail'),
77     path('<str:id_trabajador>/editar/', views.trabajador_update, name='trabajador_update'),
78     path('<str:id_trabajador>/eliminar/', views.trabajador_delete, name='trabajador_delete'),
79

```

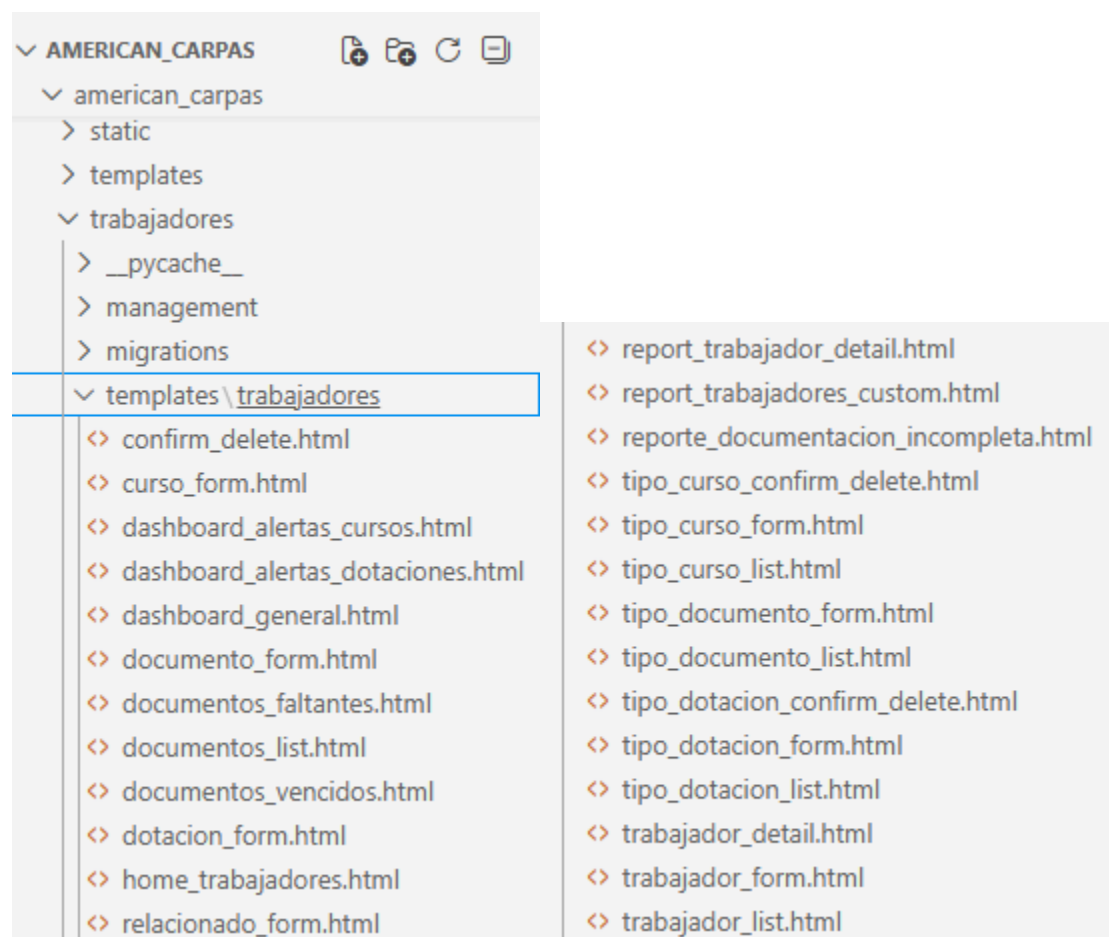
5.2.7 Templates (Plantillas HTML)

De acuerdo a este archivo concede definir el diseño y la estructura para el módulo de trabajadores, donde existen los archivos HTML o plantillas donde demuestra marcadores de posición y

lógica para definir un contenido dinámico. A continuación, en la figura 26 se observa los archivos HTML compuestos en dicha sección.

Figura 26.

Estructura interna del archivo templates



5.2.8 Formularios (Forms)

Este archivo permite crear los formularios de acuerdo a los modelos definidos para la sección de trabajadores, recopilando los datos de información del personal para las secciones de dotación, cursos, datos personales, capacitaciones, entre otros datos que compone la información del trabajador. Ver figura 27.

Figura 27.*Formulario datos trabajador-personal*

```

443 class TrabajadorPersonalForm(forms.ModelForm):
444     class Meta:
445         model = TrabajadorPersonal
446         fields = [
447             'tipo_documento', 'id_trabajador', 'nombres', 'apellidos', 'fecha_expedicion_doc', 'lugar_expedicion',
448             'nacionalidad', 'fecha_nacimiento', 'lugar_nacimiento', 'genero', 'estado_civil',
449             'direccion_residencia', 'ciudad_residencia', 'departamento_residencia',
450             'telefono_celular', 'correo_personal', 'grupo_sanguineo_rh', 'nombres_padre', 'nombres_madre',
451             'nombre_contacto_emergencia', 'numero_contacto_emergencia',
452             'numero_hijos', 'cuenta_bancaria', 'tipo_cuenta', 'banco'
453         ]

```

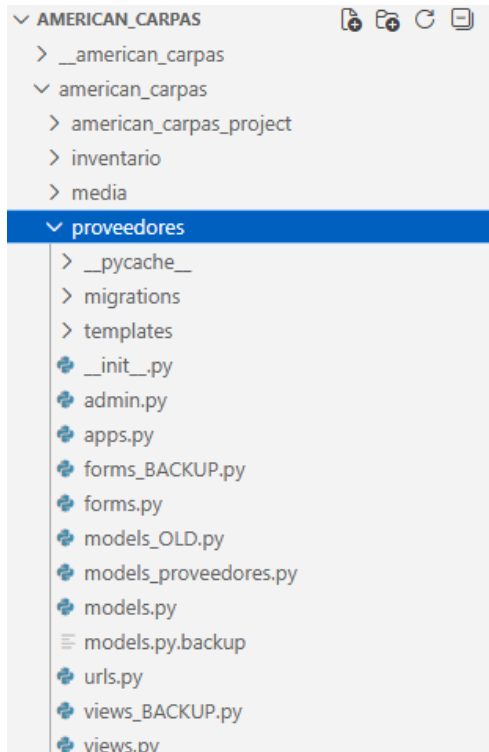
5.3 Modulo de proveedores

5.3.1 Creación del módulo de proveedores

El módulo de proveedores es el segundo modulo funcional de la aplicación en desarrollar, el cual está diseñado para agrupar y optimizar la información de los proveedores, sus productos, productos, y documentación respectivamente. Con ello se busca obtener dicha información de manera más rápida, sencilla y completa, por tanto, se establece un entorno dinámico en la interacción de los subgrupos que abarcan la información para cada proveedor (Ver figura 28). Del mismo modo, este módulo va estar relacionado con el módulo de inventarios e indirectamente con el módulo de proyectos, ya que es la base de suministro de la materia prima en producción y proyectos.

Figura 28.

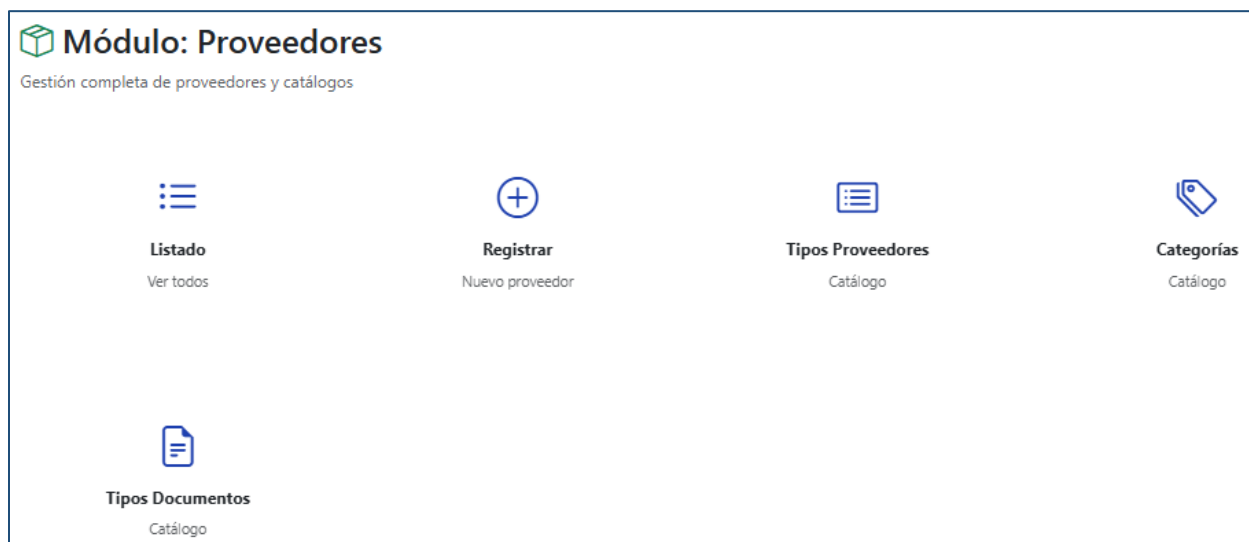
Creación y estructura de módulo de trabajadores



5.3.2 Diseño del Modelo de Datos (dividido en 5 fases)

Este módulo implementa 5 fases de composición, que separan la responsabilidad de la información del proveedor, generando un espacio controlado en lo que el proveedor ofrece: la materia prima, servicios, productos, entre otros. Este sistema permite una escalabilidad y mantenimiento eficiente de la aplicación. Las etapas están compuestas de la siguiente manera (Ver figura 29):

- **Fase 1:** Catalogos de clasificación (Tipos de proveedor, categorías, tipos de documento)
- **Fase 2:** Proveedor principal
- **Fase 3:** Contactos del proveedor
- **Fase 4:** Documentos del proveedor
- **Fase 5:** Productos y servicios del proveedor

Figura 29.*Fases del módulo de proveedores*

5.3.2.1 Catálogos de clasificación

En los catálogos se definen las tablas maestras en las cuales muestra las opciones de la clasificación del proveedor, es decir confirmar el tipo de proveedor que está vinculado y puede estar categorizado como fabricante, importador, mayorista, minorista, distribuidor, etc. El modelo se describe como *"TipoProveedor"* y así se clasifica el proveedor. Cabe resaltar que dicho listado es configurable desde el área de administración "Panel administrativo" y permite asociar iconos para la representación visual de los mismos. Además, se tiene la opción de definir el proveedor si se encuentra activo para que el usuario se informe si está habilitada dicho abastecedor sin necesidad de eliminar la información de los mismos. Así pues, la categoría también establece las opciones para los subproductos que comprende cada agente. En la figura 30 se observa el código para categorizar los proveedores y en la figura 31 se observan los catálogos.

Figura 30.

Código para categorizar proveedor

```

84 # =====
85 # MODELOS DE CATÁLOGOS - FASE 1
86 # =====
87
88 class TipoProveedor(models.Model):
89     """
90     Catálogo de tipos de proveedores
91     Ejemplos: Fabricante, Distribuidor, Importador, etc.
92     """
93     id_tipo_proveedor = models.AutoField(primary_key=True)
94     nombre_tipo = models.CharField(max_length=100, unique=True, verbose_name="Nombre del Tipo")
95     descripcion = models.TextField(blank=True, null=True, verbose_name="Descripción")
96     icono = models.CharField(max_length=50, blank=True, null=True, verbose_name="Icono")
97     activo = models.BooleanField(default=True, verbose_name="Activo")
98     fecha_creacion = models.DateTimeField(auto_now_add=True, verbose_name="Fecha de Creación")
99
15 ESTADO_CHOICES = [
16     ('ACTIVO', 'Activo'),
17     ('INACTIVO', 'Inactivo'),
18 ]

```

Figura 31.

Lista proveedores

Tipos de Proveedores				
Proveedores / Tipos de Proveedores				
Buscar por nombre... Buscar				
Icono	Nombre	Descripción	Estado	Acciones
	Constructora	Proveedor servicios de ingeniería civil	Activo	
	Fabricante	Fabricante de lonas	Activo	

5.3.2.2 Proveedor principal

Con relación a esta fase se establece un modelo central que almacena toda la información del proveedor, incluyendo los datos de identificación, contacto, información bancaria, fiscal y tributaria, evaluación de desempeño y condiciones comerciales. En la figura 32 se evidencia los procesos de diligenciamiento de datos para la creación de un proveedor, lo cual fue ejecutado en el archivo de “models”. Por otro lado, se ingresan procesos de validación como las calificaciones del proveedor, la

numeración de crédito no negativo, los plazos de pago negativo y el email en formato valido. En consecuencia, la información se determina por campos como el id_proveedor, digito_verificación, calificación, plazo_pago_días, entre otros campos que se solicita al usuario para completar la información.

Figura 32.

Campos para crear un proveedor

Registrar Nuevo Proveedor

[Proveedores](#) / [Listado](#) / [Crear](#)

Identificación Básica

Tipo de Persona: Persona Jurídica ▼ Razón Social: Nombre legal completo del proveedor

Nombre Comercial: Nombre comercial (opcional)

Tipo de Documento: NIT ▼ Número de Documento: NIT, CC, CE o Pasaporte Dígito de Verificación: DV (solo para NIT)

Tipo de Proveedor: -- Seleccione un tipo -- ▼ Categoría Principal: -- Seleccione una categoría -- ▼ Estado: Activo ▼

Información Legal y Tributaria

Régimen Tributario: Ej: Régimen Común, Simplificado Responsabilidad Fiscal: No Responsable de IVA ▼ Selección de responsabilidad fiscal País de Origen: Colombia

Actividad Económica: Descripción de la actividad económica Código CIIU: Ej: 4711

Información de Contacto

5.3.2.3 Contactos del proveedor

El modelo de contacto de proveedor tiene como propósito gestionar de manera eficiente los múltiples contactos que existen en un proveedor, lo cual permite la organizar y categorizar los contactos al área responsable, como el área comercial, el área de técnica o financiera, área ejecutiva o

representantes y área de producción. Esto incluye una relación de muchos a uno donde se almacena la información de teléfono, correo electrónico y cargo en cada una de las áreas.

5.3.2.4 Documentos del proveedor.

Este modelo está diseñado con el fin de organizar los documentos digitales de los proveedores, en el cual tiene un control de vigencia con alertas, detección del tamaño de archivos, almacenamiento organizado por año o mes y categorización por medio del Tipo_documento_Proveedor definiendo los diferentes tipos de archivos (ver figura 33), entre los cuales encontramos, el rut, cámara de comercio, certificaciones, entre otros. Esto permite revisar el estado de los proveedores a nivel general

5.3.2.5 Productos y servicios del proveedor.

En el prototipo de los productos se gestiona el catálogo de productos y servicios que ofrece cada proveedor, incluyendo proceso de costos, disponibilidad, condiciones comerciales y descuentos.

Se especifican factores de múltiples unidades de medida, soporte de multimoneda y controles de stock opcional según los productos.

5.3.3 Catálogos de Datos (Choices)

Establece un conjunto de las constantes globales de código para asegurar la consistencia de información, es decir opciones estandarizadas para campos claves, como las responsabilidades fiscales, los tipos de documentos de identidad, las unidades de medida para productos y monedas soportadas.

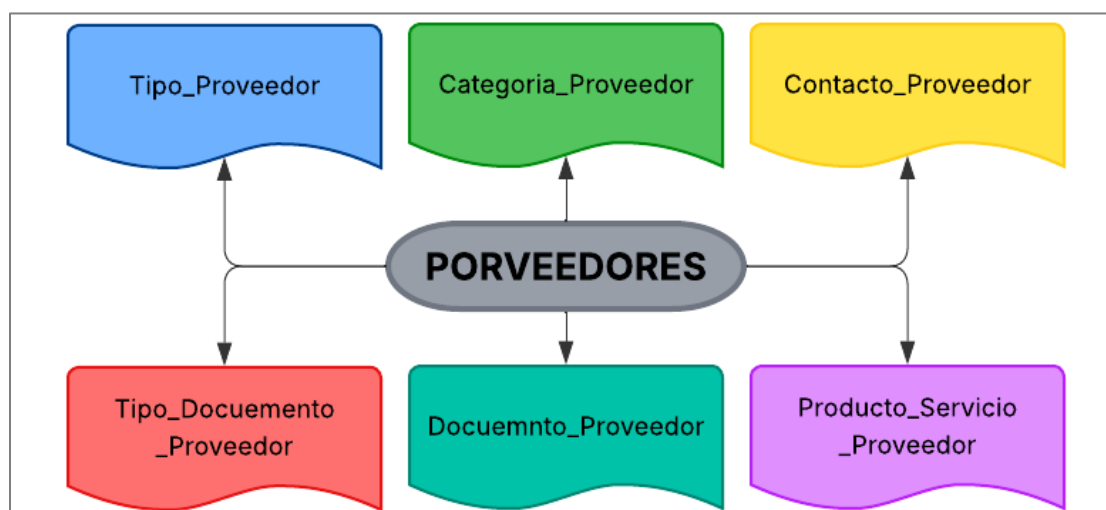
5.3.4 Migraciones de Base de Datos.

En la etapa de migraciones de bases de datos se detalla el proceso para crear y aplicar la estructura de base de datos con comandos (*makemigrations* y *migrate*), las cuales generan las tablas para la aplicación de MySQL, entre las cuales se encuentra tipos de proveedores, categorías de proveedor, tipos de documentos, contactos, productos y documentos en general.

5.3.5 Resumen de Arquitectura

Figura 33.

Diagrama de relaciones de proveedores



5.4 Modulo de inventarios

5.4.1 Creación del módulo de inventarios

El módulo de inventarios es uno de los sistemas más críticos en la aplicación, ya que se maneja información altamente dinámica con múltiples funciones y estados, movimientos en cantidades y trazabilidad de las operaciones. Es así, que se lleva un registro de cantidades de material (tubería, lona y herramienta), control de ordenes de producción, información de costos y la generación de alertas sobre los informes de inventario como la poca disponibilidad de metros en lona y tubería. Esto representa la composición robusta de dicho modulo y del cual se definen los siguientes aspectos.

5.4.2 Arquitectura General del Módulo

Para el módulo de inventarios en general se diseñaron 3 capas funcionales, siguiendo una responsabilidad especificada para cada material o herramienta utilizada allí en American Carpas 1 sas.

- **Catálogos de configuración:** comprende la configuración de las características del material, tipo, color ancho, medida, calibre, ubicación, entre otras.
- **Catálogos principales:** se gestiona los tres modelos principales del inventario, en el cual se almacena información para la lona, la tubería y los accesorios; cada uno implementa la lógica de caracterización del material o en su defecto del accesorio.
- **Operaciones y trazabilidad:** se gestiona las transacciones sobre el inventario, es decir las ordenes de producción que se registra del material utilizado se genera un historial para proceso de descuentos

5.4.3 Submódulo: catálogos de lonas

Se construye este sistema más completo ya que constituye las características más específicas de la lona, como es el ancho de lona, los diferentes colores que se manejan por códigos de referencias, tipo de lona (calibre de lona), tratamientos espaciales en la lona; esto permite conocer un catálogo independiente de combinación de estas características para la base de datos, lo cual evita la explosión combinatoria en caso de crear catálogos únicos para cada característica (Ver figura 34)

Figura 34.

Codificación y datos de ingreso para características de lona

```

1284 # Características
1285 tipo_lona = models.ForeignKey(
1286     TipoLona,
1287     on_delete=models.PROTECT,
1288     related_name='lotes',
1289     verbose_name="Tipo de Lona"
1290 )
1291 ancho_lona = models.ForeignKey(
1292     AnchoLona,
1293     on_delete=models.PROTECT,
1294     related_name='lotes',
1295     verbose_name="Ancho de Lona"
1296 )
1297 color_lona = models.ForeignKey(
1298     ColorLona,
1299     on_delete=models.PROTECT,
1300     related_name='lotes',
1301     verbose_name="Color de Lona"
1302 )

```

+ Nuevo Rollo de Lona

Características de la Lona

Tipo de Lona *

Ancho *

Color *

Tratamiento

Gramaje (g/m²)

Peso en gramos por metro cuadrado

Control de Metraje

Metros Iniciales *

Metros totales del rollo

Metros Disponibles *

Metros actuales disponibles

Metros Reservados

Para órdenes pendientes

Mínimo Alerta

Stock mínimo

\$ Costos y Ubicación

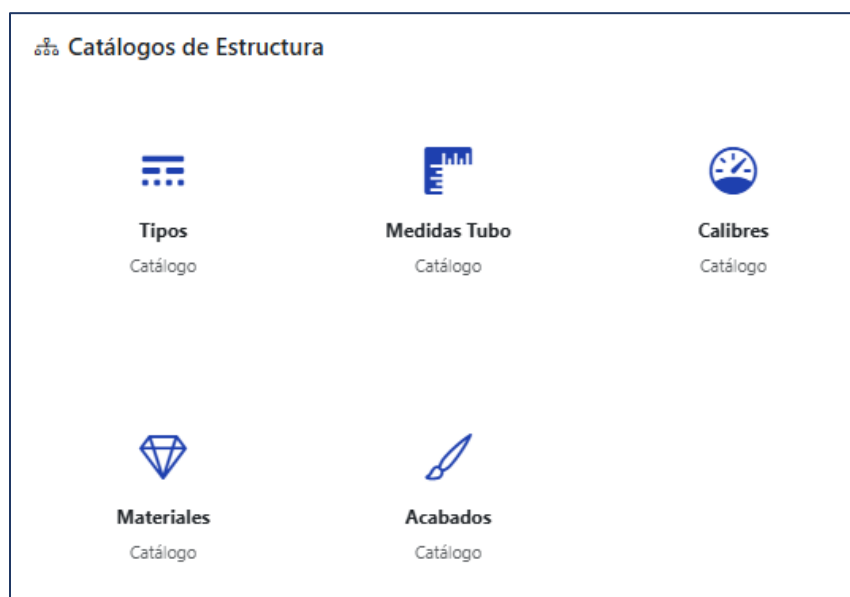
De lo anterior se resalta procesos identificación única para los rollos de lona (Número de lote), como las conversiones en unidades de medida si fuese necesario, el control de ordenamiento de anchos de lona, el cálculo de cortes que se pueden realizar en un rollo para una operación y codificación de colores.

5.4.4 Submódulo: catálogos de estructura

Este catálogo representa una diversidad de características, donde se implementa el tipo de tubería (Cuadrada, redonda, perfil en C, varillas, platinas, ángulos), cada uno de estos tipos se representan en diferentes medidas (1", 2", 3", etc.) que a su vez también se encuentran en diferentes metros lineales; lo cual hace que sea un módulo más complejo en crear. Sin embargo, se establecen identificaciones como tipo_estructura, medida_tubo, valor_pulgadas y espesor para completar el registro de dicha sección. Es así que en la figura 35 se observa su registro y verificación de medidas para categorizar y conocer dicho material.

Figura 35.

secciones para caracterizar tubería



5.4.5 Submódulo: inventario de lonas estructuradas

El inventario de lona y estructuras es el núcleo que representa las cantidades de material en almacén, este modelo mantiene la identidad individual en cada, lo que permite una trazabilidad de saber que rollo de lona y tubería se utiliza, al mismo tiempo el control de calidad (defectos de fabrica) y ubicación del material en la empresa (saber dónde se encuentra). Así encontramos la lógica de estados en metraje para diligenciar metros iniciales, disponibles y reservados correspondientemente. Del mismo modo se crea una configuración de alertas en las cantidades mínimas para cada material, así el usuario queda notificado para proceder a comprar y consultar con el proveedor. En la figura 36 se identifica las secciones para verificar cantidades existentes en cada material y que optimiza en la busque para procesos de logística y operaciones de producción en las respectivas áreas.

Figura 36.

Secciones de inventario

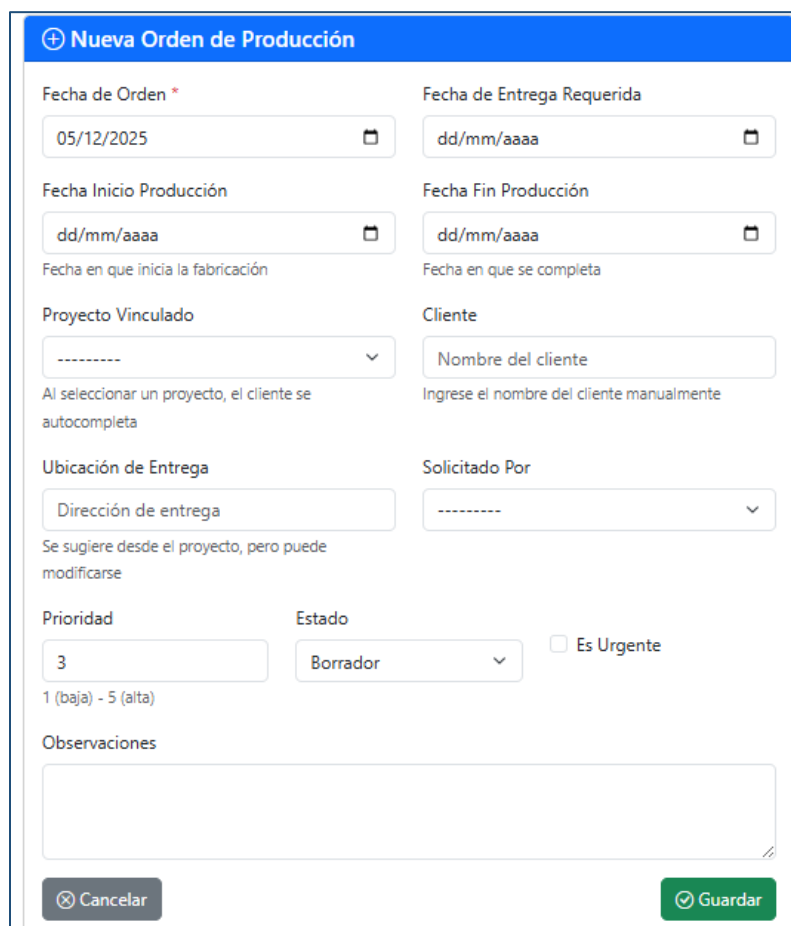


5.4.6 Submódulo: órdenes de producción

Complementando la logística de inventario general, se establece la generación de órdenes de producción como el corazón operativo de dicha sección, que representa el consumo de materiales, la asignación de operarios, a trazabilidad, el costo y planificación de los proyectos en las actividades para la fabricación. En la figura 37 se define el formulario para crear las órdenes de producción. A causa de ello se genera un número de orden, un ciclo de fabricación (programación, estado proceso, sección completa y cancelada o terminada). En efecto se implementa el módulo de proyectos, llamando así al cliente al cual se le está fabricando el producto y del cual, por medio de la verificación de completar las actividades, completa al mismo tiempo el proyecto en general.

Figura 37.

Formulario para crear orden de producción



+ Nueva Orden de Producción

Fecha de Orden * 05/12/2025	Fecha de Entrega Requerida dd/mm/aaaa
Fecha Inicio Producción dd/mm/aaaa	Fecha Fin Producción dd/mm/aaaa
Fecha en que inicia la fabricación	Fecha en que se completa
Proyecto Vinculado -----	Cliente Nombre del cliente
Al seleccionar un proyecto, el cliente se autocompleta	Ingrese el nombre del cliente manualmente
Ubicación de Entrega Dirección de entrega	Solicitado Por -----
Se sugiere desde el proyecto, pero puede modificarse	
Prioridad 3	Estado Borrador
1 (baja) - 5 (alta)	☐ Es Urgente
Observaciones 	
Cancelar Guardar	

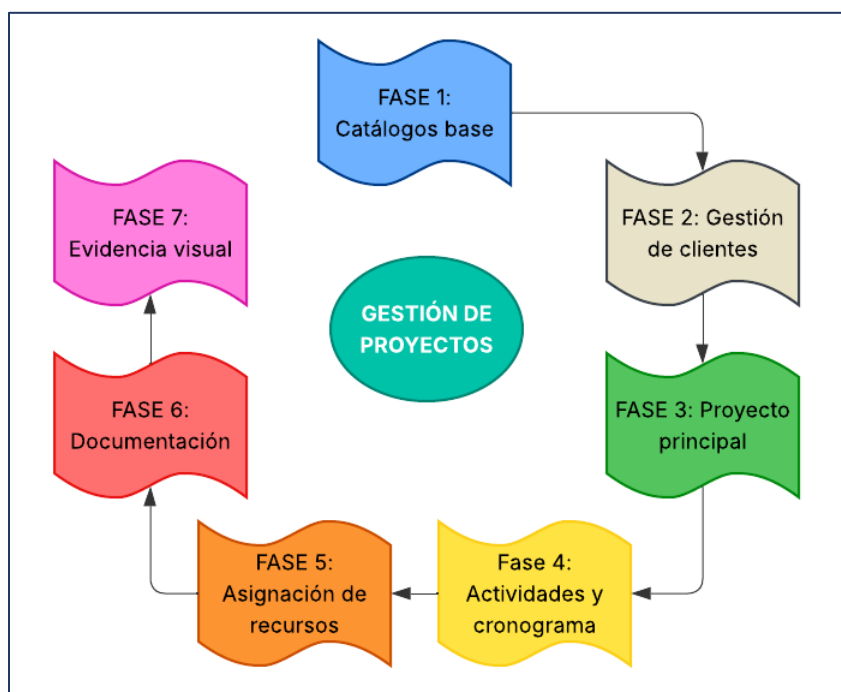
5.5 Modulo de proyecto

Es el módulo de integración de todos los módulos de la página web, actuando como el punto de convergencia donde se consolida toda la información de trabajadores, proveedores, inventarios y clientes de proyectos de American Carpas. Del cual a comparación de los otros módulos que gestionan procesos individuales, en este módulo se ejecuta los procesos completos desde que se vende al cliente hasta que se le entrega. Este módulo comprende los siguientes objetivos para establecer la mejor especialidad en los datos de información y se crea un modelo de fases que comprende. (Ver figura 38)

- Gestionar múltiples proyectos al mismo tiempo
- Mantener cronograma de actividades para organizar trabajo
- Asignación de actividades en las áreas correspondientes (Lonas – estructuras)
- Almacenamiento de documentación del proyecto
- Registro de evidencias fotográficas
- Calcular avance del proyecto por cliente
- Generar reporte de avance y procesos de actividades

Figura 38.

Estructura del módulo de proyectos (7 fases)



5.5.1 Fase 1: Catálogos base

El propósito de esta fase es definir las opciones configurables del proyecto, que clasifica y controla el comportamiento del mismo. Se inicia su configuración por línea de código, nombre_tipo, color_identificador, icono y activo. El modelo controla el ciclo del proyecto mediante los estados

configurables desde el inicio hasta el final, generando de esta manera diferentes estados sobre las actividades. Por otro parte, el modelo de tipo documento, exige os tipos de documentos que pueden adjuntarse a proyectos con un sistema sofisticado de control de vigencia, donde documentos como licencias de construcción y pólizas de seguros requieren control de vencimiento con alertas 60 y 30 días antes respectivamente. Este sistema genera automáticamente alertas visuales codificadas por colores donde el rojo indica documentos vencidos, el amarillo señala documentos por vencer dentro del período de alerta y el verde muestra documentos vigentes, permitiendo a la empresa mantener al día toda su documentación legal sin depender de recordatorios manuales.

5.5.2 Fase 2: Gestión de clientes

En estos modelo se centraliza toda la información del cliente/proyecto, ya se persona natural o jurídica. El cliente fue diseñado para mantener 4 tipos de documentos de identificación, los cuales son Nit, cedula ciudadanía, cedula extranjera y pasaporte. Sin embrago también se implementan datos de contacto, email, ubicación de empresa o proyecto, ciudad, país, entre otros. Los métodos empleados permiten generar reportes de los proyectos activos y su avance.

5.5.3 Fase 3: Proyecto principal

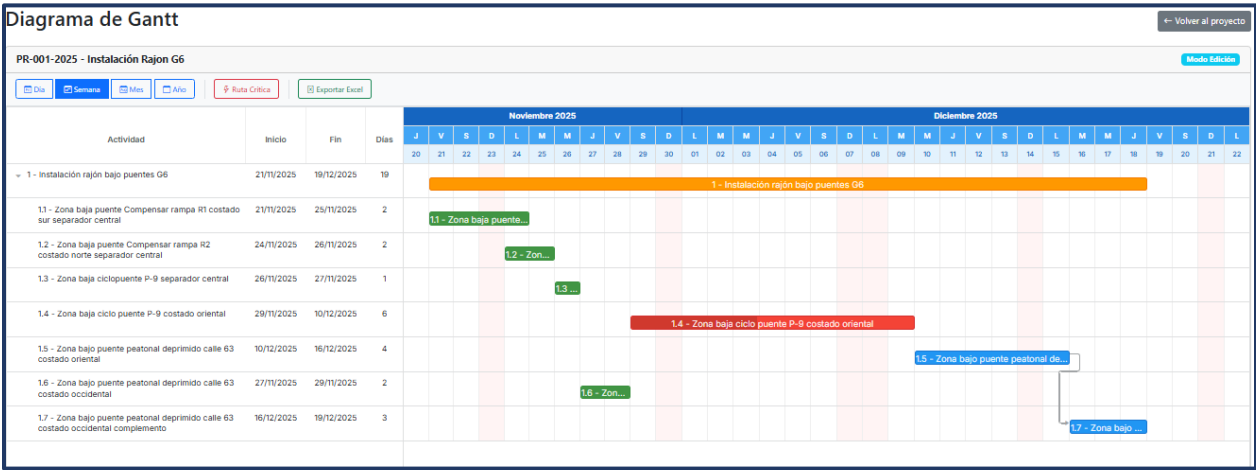
Contiene el núcleo central que integra toda la información del sistema, asegurando la integridad de los datos al impedir la eliminación de clientes o catálogos asociados. Cada proyecto se detalla con un código único, el ingeniero responsable del cliente y la ubicación física de la obra, independientemente de la sede del cliente. Su función principal es el control de tiempo mediante fechas de inicio, fin estimado y fin real, permitiendo la **detección automática de atrasos**. La característica más estratégica es el **Cálculo de Avance Ponderado**, que asigna peso a las actividades, ofreciendo un porcentaje de progreso **realista** y superior al promedio simple, crucial para generar alertas gerenciales rápidas sobre el estado y los retrasos del proyecto.

5.5.4 Fase 4: Actividades y cronograma

implementa tres modelos para gestionar el desglose de trabajo. El modelo actividad crea una jerarquía tipo WBS (padre/hija) y la funcionalidad clave del avance cuantitativo, donde el progreso se calcula con base en cantidades físicas ejecutadas (ej. metros) en lugar de estimaciones subjetivas, propagando automáticamente el avance hacia las tareas superiores y generando alertas preventivas. El modelo Avance_Actividad registra el historial de progreso parcial, recalculando automáticamente los totales para mantener la consistencia y permitir análisis de productividad temporal. Ver figura 39, donde se observa un seguimiento de las actividades que comprende un proyecto (Diagrama de Gantt) y además en la figura 40 un proceso de caracterización del proyecto en general con avance

Figura 39.

Diagrama de Gantt en proyectos (Ejemplo)



desasignación cuando el trabajador completa su participación. Lo anterior permite establecer un control histórico y de reconocimiento de un operador en los proyectos.

5.5.6 Fase 6: Documentación

Esta fase se centra en la gestión de archivos mediante el modelo Documento_Proyecto, creando un sistema inteligente de control de vigencia. Los documentos se adjuntan a su proyecto (eliminando los documentos si el proyecto es borrado), pero no se pueden eliminar sus tipos si tienen archivos asociados, garantizando la integridad. Cada archivo se almacena de forma organizada y contiene datos clave como el nombre, número de radicado y, fundamentalmente, la Fecha de Vencimiento. El sistema implementa una lógica de alertas automática que utiliza tres métodos para determinar si un documento está vigente, calcular los días para vencer y verificar si requiere renovación (basado en un período de alerta configurable, típicamente 30 o 60 días). Esto genera alertas visuales codificadas por color: verde (vigente), amarillo (próximo a vencer) y rojo (vencido), priorizando la intervención en los documentos más críticos y próximos a expirar en el *dashboard*.

5.5.7 Fase 7: Evidencia visual

En esta última fase se implementa el modelo Evidencia_Fotográfica para documentar visualmente el proyecto, clasificando las imágenes en siete tipos (Inicio, Avance, Problema, Solución, Calidad, Final y Otro) que cubren el ciclo completo de la obra. Las evidencias se vinculan al proyecto y opcionalmente a una actividad específica, almacenando datos como el título, la descripción y la ubicación. Del mismo modo se utiliza un método de extracción automática EXIF de imágenes, dando a conocer que, al subir dichas imágenes, el sistema busca la fecha y hora real de captura de fotos; esto evita errores de digitación o de ingreso manual.

6. Fase de validación y retroalimentación

Se realizaron pruebas exhaustivas para verificar el correcto funcionamiento del sistema completo, el cual dio lugar a crear las interacciones de los usuarios siendo más sencillas y completas, es así que se realiza lo siguiente (ver figura 41 y 42):

1. **Pruebas de flujo completo:** Verificar que un proceso de principio a fin funcione correctamente.
Como crear un proyecto → Asignar trabajadores → Asignar materiales → Generar reporte.
2. **Pruebas de integridad de datos:** Asegurar que las relaciones entre tablas se mantengan consistentes. Por ejemplo, al eliminar un proveedor, verificar que los productos asociados no queden huérfanos.
3. **Pruebas de validación:** Comprobar que las validaciones funcionen en todos los formularios, es decir, no permitir números negativos, fechas inválidas, documentos duplicados.

Figura 41.

Pruebas realizadas a la aplicación

Seleccionar campos para exportar
✕

Selecciona los campos que deseas incluir en la exportación:

⌵ Seleccionar todos
✕ Deseleccionar todos

Identificación
☒ Documento
 ☐ Tipo de documento
 ☒ Nombres
 ☒ Apellidos
 ☐ Fecha expedición documento
 ☐ Lugar expedición
 ☐ Nacionalidad

Datos Personales
☐ Fecha nacimiento
 ☐ Lugar nacimiento
 ☐ Género
 ☐ Estado civil
 ☐ Grupo sanguíneo
 ☐ Número de hijos
 ☐ Nombre padre
 ☐ Nombre madre

Contacto
☒ Celular
 ☒ Correo
 ☐ Dirección
 ☐ Ciudad
 ☐ Departamento
 ☐ Contacto emergencia
 ☐ Teléfono emergencia

Reporte de Trabajadores

Totale de registros: 2 | Generado el 30/11/2025 00:15

Documento	Tipo Doc.	Nombres	Apellidos	Fecha expedición doc.	Género	Celular
12345678	Cédula de Ciudadanía	Prueba Prueba	Prueba Prueba	1 de Febrero de 2022	Masculino	12345678
1098765432	Cédula de Ciudadanía	Aldemar	Quesada González	15 de Marzo de 2015	Masculino	3201234567

Información de Contacto

Dirección:
Dirección completa
• Este campo es obligatorio.

Ciudad:
Ej: Bogotá
• Este campo es obligatorio.

Departamento/Estado:
Ej: Cundinamarca

País:
Colombia

Código Postal:
050001

Sitio Web:
https://www.ejemplo.com

Teléfono Principal:
+57 300 123 4567
• Este campo es obligatorio.

Teléfono Secundario:
Opcional

Email Principal:
contacto@proveedor.com
• Este campo es obligatorio.

Email Secundario:
Opcional

Email de Facturación:
Email para facturación electrónica
Para facturación electrónica

- Pruebas de permisos:** Verificar que los controles de acceso funcionen correctamente.
- Pruebas de rendimiento:** Evaluar el tiempo de respuesta con volúmenes de datos realistas.

Figura 42.

Imagen de listado de trabajadores

Listado de Trabajadores
Sección de personal de Área con 1.200.

[Registrar nuevo trabajador](#)

Búsqueda y filtros

☐ Seleccionar todos en esta página ☐ Desseleccionar todos ☐ Agrupar seleccionados a la lista de exportación

Documento	Tipo Doc.	Nombres	Apellidos	Teléfono	Correo	Acciones
12345	Cédula de ciudadanía	Enrique	Casullo Castillo	3103456789	enr.casullo@proba.com	E A D
1034567890	Cédula de ciudadanía	Diana Carolina	García Montañez	3118765432	di.garcia@proba.com	E A D
52341678	Cédula de ciudadanía	Ismael Darío	Jiménez Chaparro	3110456789	isma.jimenez@gmail.com	E A D
1023456789	Cédula de ciudadanía	Maria Fernanda	López Martínez	3154567890	ma.lopez@gmail.com	E A D
12345634	Cédula de ciudadanía	Karen	Martínez Díaz	3142345678	ka.mart@gmail.com	E A D
43567812	Cédula de ciudadanía	José Luis	Méndez Pineda	3143567890	jo.mendez@gmail.com	E A D
1045678901	Cédula de ciudadanía	León Fernando	Morales Castro	3109012345	le.morales@gmail.com	E A D
79456123	Cédula de ciudadanía	Sofía	Navarro Álvarez	3103123456	so.navarro@gmail.com	E A D
1012379025	Cédula de ciudadanía	Roberto	Quintero	31456789	ro.quintero@gmail.com	E A D
1087654321	Cédula de ciudadanía	Juan Pablo	Ramírez Silva	3108765432	jp.ramirez@gmail.com	E A D

[«](#) [<](#) [1](#) [2](#) [3](#) [»](#) [»](#)

Página 1 de 2

Durante las pruebas de integración se identificaron y corrigieron diversos errores:

- Errores de validación en formularios
- Problemas de permisos en ciertas vistas
- Inconsistencias en el formato de fechas

- Errores de cálculo en reportes
- Problemas de rendimiento en consultas complejas

Conclusiones y Recomendaciones

El proyecto logró desarrollar una aplicación web que optimiza la gestión de datos, documentación e inventarios en la microempresa American Carpas 1 SAS, superando las limitaciones del manejo tradicional con hojas de cálculo dispersas. La solución centraliza la información en una plataforma accesible y segura, mejora la integridad y trazabilidad de los datos, y automatiza procesos clave como alertas de stock y vencimientos. La implementación tecnológica basada en Python, Django y MySQL permitió construir un sistema funcional, escalable y compatible con dispositivos móviles, que facilita la toma de decisiones y la planificación estratégica. Las pruebas realizadas confirmaron la efectividad del sistema, identificando y corrigiendo errores para garantizar su robustez y usabilidad. En conclusión, la aplicación contribuye a la transformación digital de la empresa, aumentando su eficiencia operativa y competitividad, y puede ser replicada en otras MiPymes del sector.

Del mismo modo este sistema de gestión de información integral (Base de datos) para American Carpas 1 sas completa lo establecido en diseño, y desarrollo de la aplicación, generando los cuatro módulos fundamentales interconectados: Trabajadores, Proveedores, Inventarios y Proyectos. Este representa la transformación digital que reemplaza los métodos tradicionales y manuales que caracterizan la gestión operativa y administrativa de una empresa, eliminando problemas de duplicidad de información, inconsistencias de datos, generación de alertas y trazabilidad de procesos. La arquitectura modular implementada bajo el framework Django con base de datos MySQL garantiza escalabilidad, mantenibilidad y extensibilidad del sistema, permitiendo que cada módulo evolucione independientemente sin afectar la funcionalidad de los demás. El sistema desplegado en producción

mediante Railway en la URL <https://gestociv.cloud/> ha demostrado estabilidad operacional y transformación positiva en el manejo de información en las diferentes áreas establecidas en la empresa.

En cuanto a los procesos analizados en los objetivos, se comprueba la identificación de necesidades por medio de las encuestas, observación directa y revisión de herramientas que se utilizaban, esta información permitió identificar 4 áreas críticas para el modelo de sistema de gestión (Trabajadores, proveedores, inventario y proyectos). Esto dio paso a la elaboración de una estructura modular que representa cada una de estos componentes fundamentales generados sistemas lógicos interconectados como los diagramas de Entidad-relación que permitieron definir las conexiones directas entre módulos y procesos, sirviendo como guía clara para la evolución del sistema. De manera que se implementa el proceso de codificación con las aplicaciones y plataformas (Django, Railway, MySQL, entre otras) que permitieron unificar el sistema de la página con operaciones CRUD (Crear, Leer, actualizar y Eliminar). Adicional se completan los objetivos del proyecto con la verificación y validación de diferentes usuarios que representan la empresa y que actualmente se encuentra vinculados al control y aprendizaje de la aplicación generando optimización tiempos y proceso que antes no podían gestionar de una mejor forma. Tabla 6.

Tabla 6.

Cuadro comparativo de sistema antigua y sistema implementado (nuevo)

Aspecto	Sistema Anterior	Sistema Implementado	Mejora
Centralización	20+ archivos Excel dispersos	Base de datos única integrada	✓ Completa
Acceso	Archivos locales /Google Drive	Aplicación web desde cualquier dispositivo	✓ Universal
Actualización	Manual en múltiples lugares	Automática con propagación instantánea	✓ Tiempo real
Consistencia	Información duplicada y desactualizada	Única fuente de verdad sin duplicidad	✓ Garantizada
Trazabilidad	Inexistente	Registro completo de todos los movimientos	✓ Total
Alertas	Ninguna, todo manual	Automáticas y configurables	✓ Proactivas
Cálculos	Manuales propensos a errores	Automáticos y precisos	✓ Inmediatos
Reportes	Compilación manual de horas	Generación automática en segundos	✓ Instantáneos
Búsquedas	Texto plano limitado	Búsqueda avanzada con filtros múltiples	✓ Sofisticadas
Respalos	Manuales a inconsistentes	Automáticos diarios	✓ Confiable
Estabilidad	Limitada por Excel	Ilimitada con base de datos	✓ Arquitectura

Referencias

Angulo, A. M. (2011). La Pensión de Invalidez en Colombia. *La Pensión de Invalidez en Colombia*. Bogotá

D.C., Cundinamarca., Colombia.: Universidad Libre de Colombia.

Asamblea Nacional Constituyente de Francia. (26 de agosto de 1789). Declaración Universal de los

Derechos del Hombre y del Ciudadano. [https://www.conseil-](https://www.conseil-constitutionnel.fr/sites/default/files/as/root/bank_mm/espagnol/es_ddhc.pdf)

[constitutionnel.fr/sites/default/files/as/root/bank_mm/espagnol/es_ddhc.pdf](https://www.conseil-constitutionnel.fr/sites/default/files/as/root/bank_mm/espagnol/es_ddhc.pdf)

Bootstrap Team. (2024). *Bootstrap Documentation*. <https://getbootstrap.com/docs/>

Constitución política de Colombia [Const. P.]. (1991). Colombia: Leyer, 2.da ed. 10/02/2020.

http://www.secretariasenado.gov.co/senado/basedoc/constitucion_politica_1991.html

Constitución política de Colombia [Const. P.]. (1991). Colombia: Leyer, 2.da ed. 10/02/2020.

http://www.secretariasenado.gov.co/senado/basedoc/constitucion_politica_1991.html

Corte Constitucional [C.C.], enero 22, 2004, M.P: M. Cepeda. Sentencia T-025/04. Colombia.

10/02/2020. <https://www.corteconstitucional.gov.co/relatoria/2004/t-025-04.htm>

Corte Constitucional [C.C.], enero 23, 2008, M.P: R. Escobar. Sentencia C-030/08. Colombia. 10/02/2020.

<https://www.corteconstitucional.gov.co/relatoria/2008/c-030-08.htm>

Dec. 2374 / 93, noviembre 30, 1993. Ministerio de Educación Nacional. (Colombia). 10/02/2020.

https://www.mineducacion.gov.co/1621/articles-104283_archivo_pdf.pdf

Dec. 2613 / 13, noviembre 20, 2013. Ministerio del Interior. (Colombia). 10/02/2020.

https://www.mininterior.gov.co/sites/default/files/11_decreto_2613_de_2013.pdf

Django Software Foundation. (2024). *Django Documentation*. <https://docs.djangoproject.com/>

Django Software Foundation. (2024). *Django FAQ: General*. Encontrado en:

<https://docs.djangoproject.com/en/stable/faq/general/>

Django Software Foundation. (2024). *Django Documentation - Tutorial*. Definido en lo siguiente:

<https://docs.djangoproject.com/en/stable/intro/tutorial01/>

Django Software Foundation. (2024). *Models and databases*. Tomado de:

<https://docs.djangoproject.com/en/stable/topics/db/>

Django Software Foundation. (2024). *Django FAQ: General*.

<https://docs.djangoproject.com/en/stable/faq/general/>

Garrett, J. J. (2010). *The Elements of User Experience: User-Centered Design for the Web and Beyond* (2nd ed.). New Riders.

Ley 70 / 93, agosto 27, 1993. Diario Oficial. [D.O.]: 41.013. (Colombia).10/02/2020.

<https://www.acnur.org/fileadmin/Documentos/BDL/2006/4404.pdf?file=fileadmin/Documentos/BDL/2006/4404>

Ley 89 / 90, noviembre 25, 1890. Ministerio de Interior. [OIPI]. (Colombia).10/02/2020.

<https://www.mininterior.gov.co/la-institucion/normatividad/ley-89-de-1890>

Naciones Unidas Derechos humanos. (diciembre, 1965). Convención Internacional sobre la Eliminación de todas las Formas de Discriminación Racial. Trabajo presentado en la Asamblea General en su resolución 2106 A (XX). Colombia.

https://www.ohchr.org/Documents/ProfessionalInterest/cerd_SP.pdf

Oracle Corporation. (2024). *MySQL 8.0 Reference Manual*. <https://dev.mysql.com/doc/refman/8.0/en/>

Organización de los Estados Americanos. (1994). Convención Interamericana para prevenir, sancionar y erradicar la violencia contra la mujer. Trabajo presentado en la Organización de los Estados Americanos Departamento de Derecho Internacional, Belém.

<https://www.oas.org/es/mesecvi/convencion.asp>

Organización de los Estados Americanos. (noviembre, 1969). Convención americana sobre derechos humanos suscrita en la conferencia especializada interamericana sobre derechos humanos (B-32). Trabajo presentado en la Organización de los Estados Americanos Departamento de

Derecho Internacional, Costa Rica. https://www.oas.org/dil/esp/tratados_b-32_convencion_americana_sobre_derechos_humanos.htm

Presidente de la Republica. (2010). Garantía del derecho fundamental a la consulta previa de los grupos étnicos nacionales. Directiva Presidencial no. 1 (p.1–7). Colombia. Recuperado de https://www.mininterior.gov.co/sites/default/files/13_directiva_presidencial_01_de_2010.pdf

Presidente de la Republica. (2013). *Guía para la realización de consulta previa*. Directiva Presidencial no. 10 (p.1–26). Colombia. https://www.mininterior.gov.co/sites/default/files/12_directiva_presidencial_ndeg_10_del_07_de_noviembre_2013_4.pdf

Python Software Foundation. (2024). *Python Documentation*. <https://docs.python.org/3/>

Rodríguez Mesa, R. (2019). *Tratado Sobre Seguridad Social*. Universidad Del Norte. <https://Ugc.Elogim.Com:3107/Es/Ereader/Ugc/122380?Page=64>

Velásquez, M, (2007). *El Sistema Pensional Colombiano*. Señal Editora. Medellín Colombia.

World Wide Web Consortium (W3C). (2024). *HTML Living Standard*. <https://html.spec.whatwg.org/>

World Wide Web Consortium (W3C). (2024). *CSS Specifications*. <https://www.w3.org/Style/CSS/>

ANEXO A

Diagramas de flujo de procesos actuales de los módulos

Figura 43. Diagrama de flujo de información de proveedores (Actual)

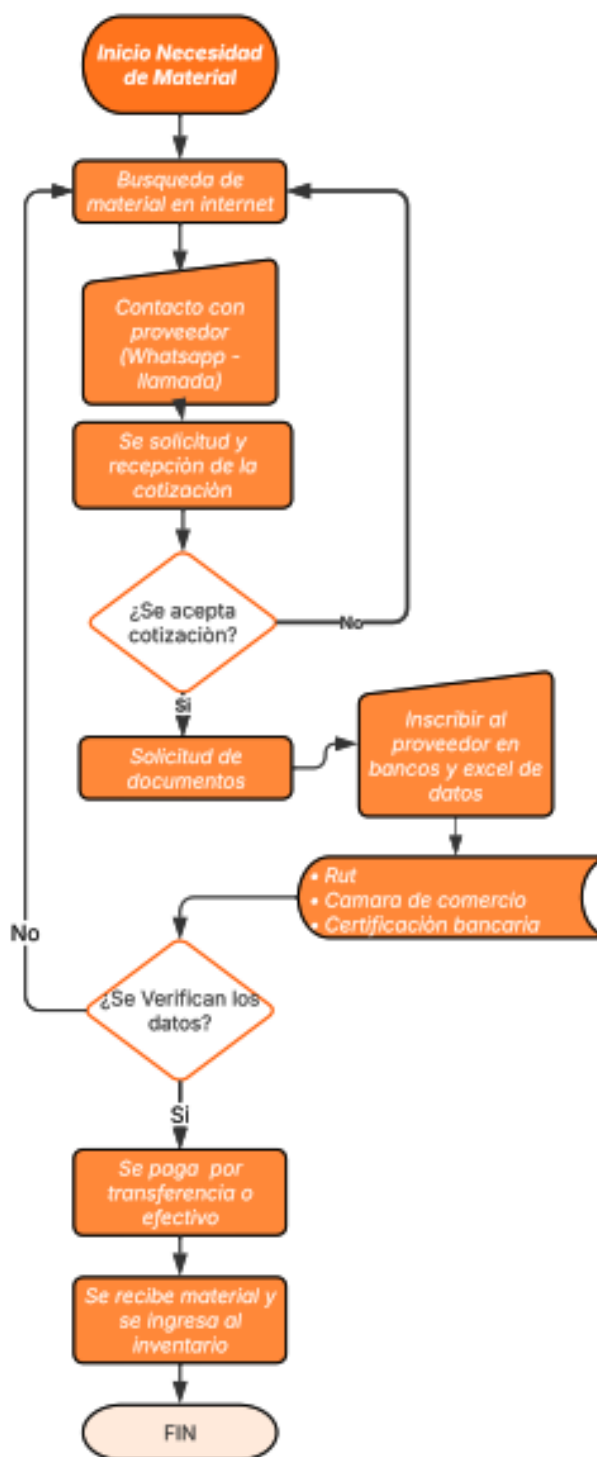


Figura 44. Diagrama de flujo de información de Proyectos (Actual)

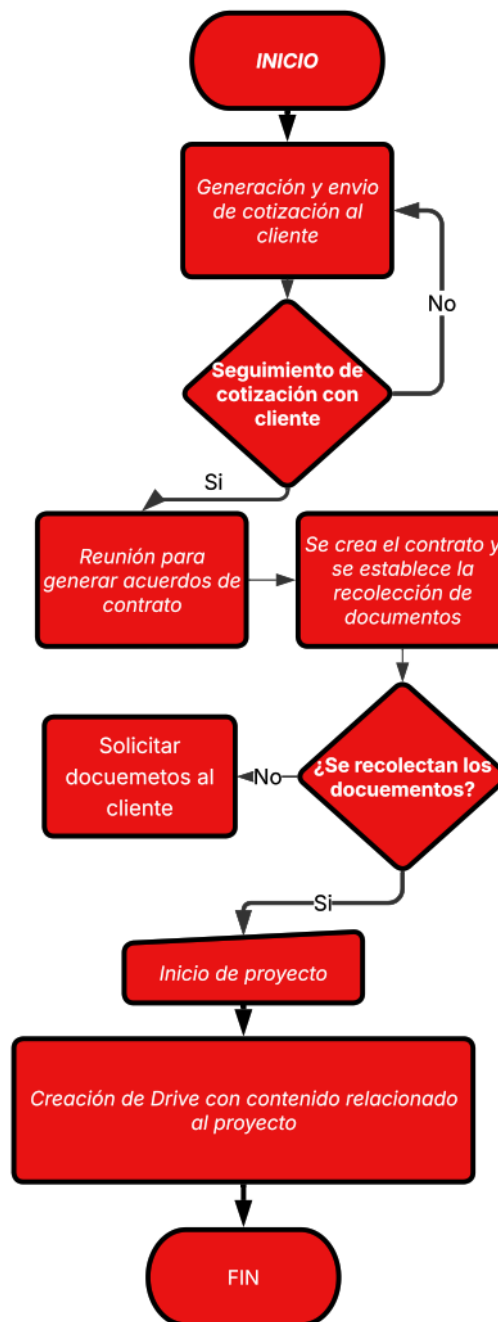


Figura 45. Diagrama de flujo de información de Trabajadores (Actual)

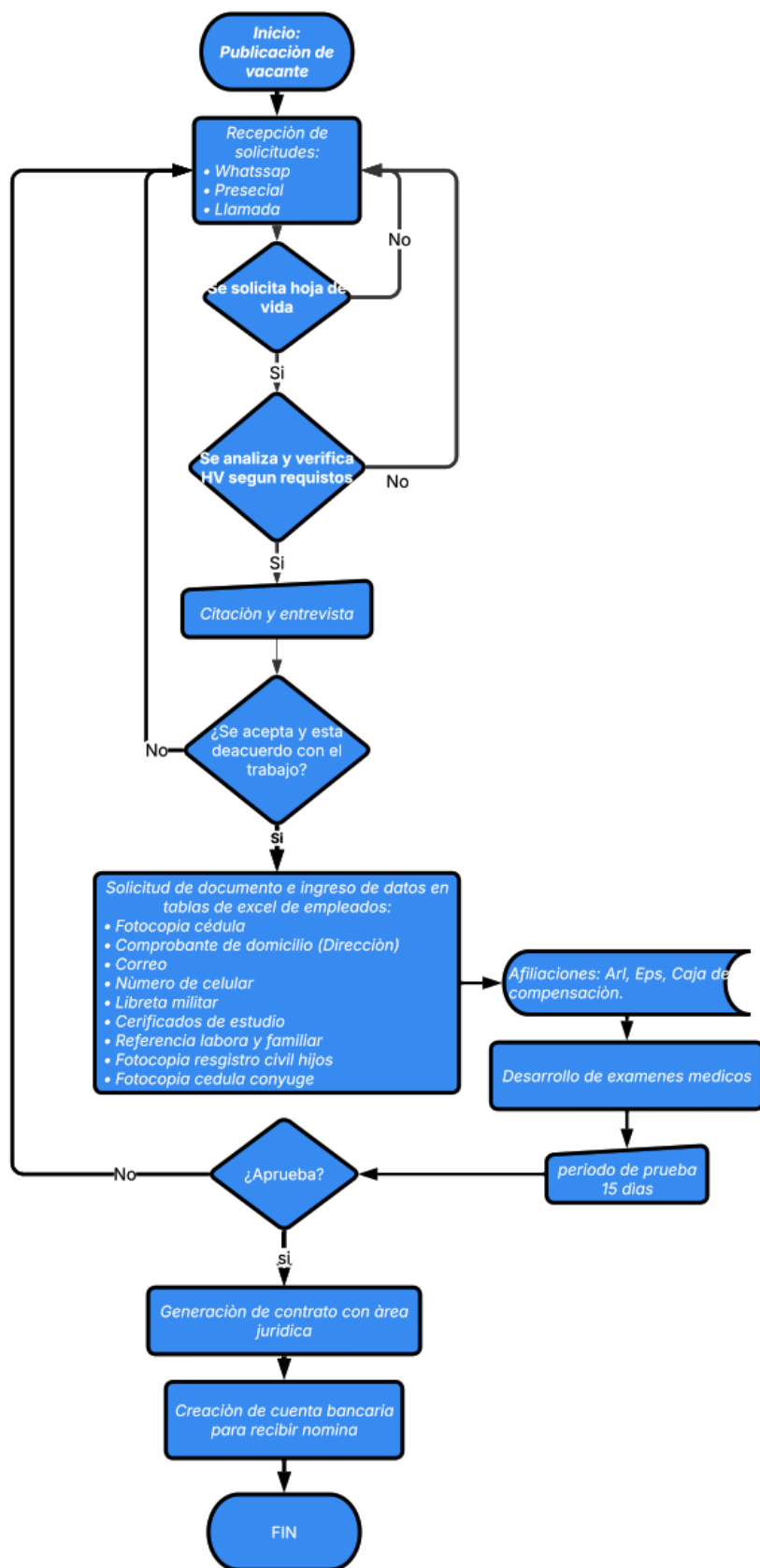
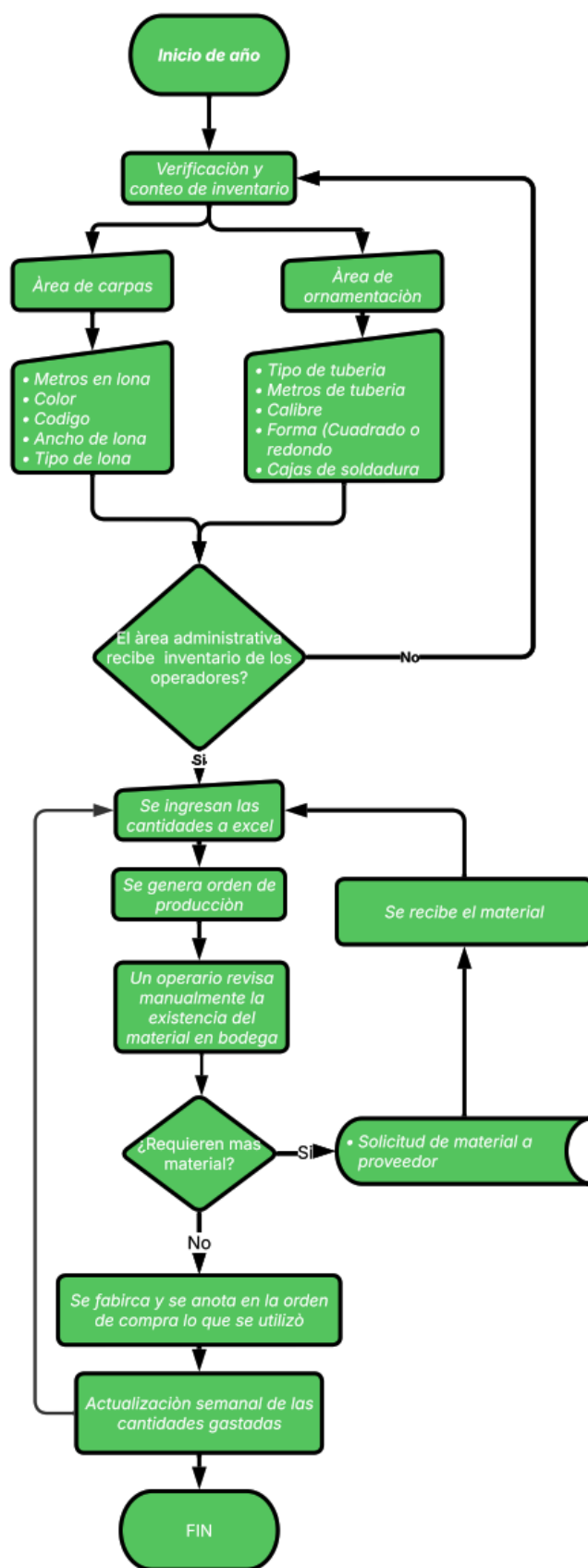
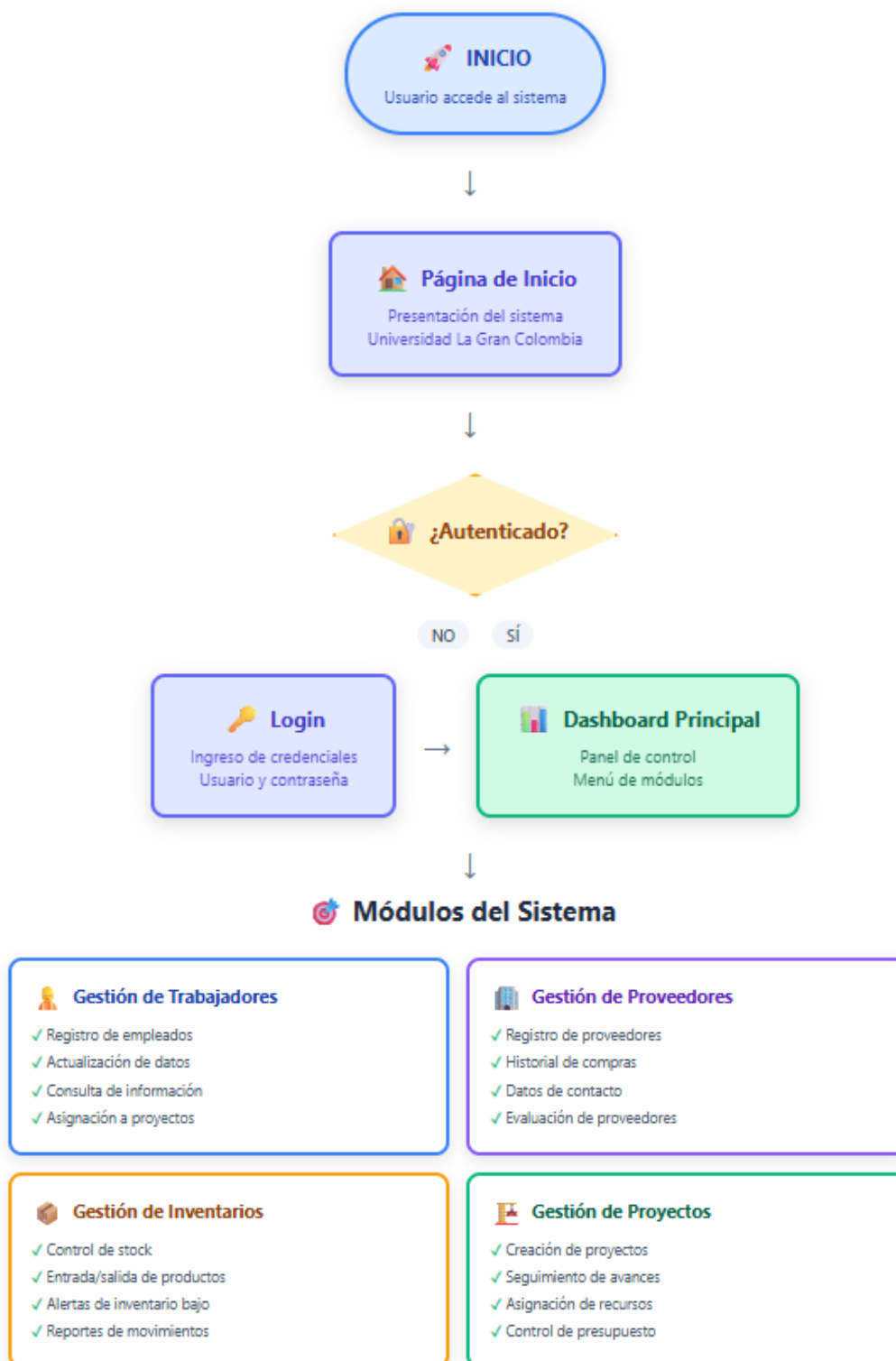


Figura 46. Diagrama de flujo de información de Inventarios (Actual)



ANEXO B

Diagrama de flujo de navegación del sistema





8% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography
- Quoted Text
- Cited Text
- Small Matches (less than 10 words)

Top Sources

- 3%  Internet sources
- 0%  Publications
- 7%  Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

C
V
i

Mario Y Yerson Quique Arias

Monografía de Grado

-  Quick Submit
-  Quick Submit
-  Universidad La Gran Colombia

Document Details

Submission ID

trn:oid:::1:3442796557

Submission Date

Dec 12, 2025, 8:59 AM GMT-5

Download Date

Dec 12, 2025, 9:22 AM GMT-5

File Name

Plantilla_Norma_APA_UGC_2025_Trabajo_Grado.pdf

File Size

2.4 MB

88 Pages

14,997 Words

97,407 Characters