

**Vargas, María Micaela**

---

**Creek: sistema de gestión  
para Camping**

**Tesis para la obtención del título de  
grado de Ingeniera de Sistemas**

Directores:

Porrini, Federico Eduardo

Carreño, Ignacio Luciano

Documento disponible para su consulta y descarga en Biblioteca Digital - Producción Académica, repositorio institucional de la Universidad Católica de Córdoba, gestionado por el Sistema de Bibliotecas de la UCC.



[Esta obra está bajo una licencia de Creative Commons Reconocimiento- No Comercial 4.0 Internacional.](#)

Universidad Católica de Córdoba  
Facultad de Ingeniería



Documento de Trabajo Final

# **Creek**

Sistema de gestión para Camping

**Autor:**

- Maria Micaela Vargas

**Directores:**

- Federico Eduardo Porrini
- Ignacio Luciano Carreño

Córdoba - Argentina

Año 2025

# Índice

|  |           |
|--|-----------|
| <b>RESUMEN.....</b>                                    | <b>4</b>  |
| <b>ABSTRACT.....</b>                                   | <b>5</b>  |
| <b>PRESENTACIÓN DEL TEMA.....</b>                      | <b>6</b>  |
| <b>GLOSARIO.....</b>                                   | <b>8</b>  |
| <b>DIAGNÓSTICO (PROBLEMÁTICA).....</b>                 | <b>9</b>  |
| Estado del Arte (Contexto Actual).....                 | 10        |
| Impacto.....   | 12        |
| Oportunidades.....                                     | 15        |
| <b>OBJETIVOS.....</b>                                  | <b>19</b> |
| <b>MARCO TEÓRICO.....</b>                              | <b>20</b> |
| Contexto General del Problema.....                     | 20        |
| Procedimientos actuales en la gestión de campings..... | 23        |
| Análisis de Campo.....                                 | 28        |
| Entrevistas y encuestas:.....                          | 29        |
| Recopilación de opiniones de los clientes:.....        | 32        |
| Observación directa.....                               | 34        |
| Opciones Similares en el Mercado:.....                 | 36        |
| Tecnologías Investigadas.....                          | 37        |
| Justificación de la elección del MERN Stack.....       | 44        |
| <b>PROPUESTA DE SOLUCIÓN.....</b>                      | <b>46</b> |
| Alcance Funcional.....                                 | 46        |
| Historias de usuario.....                              | 47        |
| Diagrama de casos de uso.....                          | 55        |
| Roles.....   | 57        |
| Casos de uso.....                                      | 59        |
| Diseño.....  | 63        |
| Diagramas.....   | 63        |
| Diagramas de Clase.....                                | 63        |
| Diagramas de Estado:.....                              | 68        |
| Arquitectura.....                                      | 72        |
| Capa de Presentación (Frontend).....                   | 73        |
| Capa de Lógica de Negocio (Backend).....               | 76        |
| Capa de Datos (Base de Datos).....                     | 78        |
| Otros módulos importados utilizados:.....              | 79        |
| Patrones de diseño.....                                | 80        |
| Modelo-Vista-Controlador (MVC).....                    | 80        |
| Singleton.....   | 84        |
| Observer.....  | 85        |
| <b>Implementación.....</b>                             | <b>88</b> |
| Estructura modular del proyecto.....                   | 88        |
| Modo de trabajo.....                                   | 93        |

|  |            |
|--|------------|
| Solución generada.....                     | 104        |
| Página web del camping.....                | 104        |
| Sistema Creek.....                         | 106        |
| <b>BENEFICIOS POST-IMPLEMENTACIÓN.....</b> | <b>120</b> |
| <b>IMPACTO SOCIAL.....</b>                 | <b>123</b> |
| <b>CONCLUSIÓN.....</b>                     | <b>126</b> |
| <b>ANEXO I.....</b>                        | <b>128</b> |
| <b>BIBLIOGRAFÍA.....</b>                   | <b>143</b> |

# RESUMEN

La industria del turismo enfrenta desafíos cada vez más complejos, especialmente para pequeños negocios familiares como el Camping Arroyito, ubicado en San Juan, Argentina. La falta de herramientas tecnológicas adecuadas para la gestión eficiente de sus operaciones ha generado problemas recurrentes, como errores en las reservas, desorganización en la administración de cabañas y carpas, y dificultades para ofrecer una experiencia satisfactoria a los clientes. Este proyecto busca abordar estas problemáticas mediante el diseño e implementación de un sistema de gestión web que modernice y optimice las operaciones del camping.

El objetivo principal es proporcionar una solución tecnológica que permita automatizar procesos clave, como la gestión de reservas, el seguimiento del estado de las cabañas y la administración contable, mejorando así la eficiencia interna y la satisfacción del cliente. Para lograrlo, se empleó una metodología de desarrollo basada en una arquitectura multicapa (frontend, backend y base de datos) utilizando React, Node.js y MongoDB. Además, se realizaron entrevistas, encuestas y observación directa para comprender las necesidades de los usuarios finales y garantizar que el sistema sea intuitivo y funcional.

Se espera que los resultados de implementar este sistema muestran una reducción significativa en los errores administrativos, una mejora en la experiencia de usuario para los clientes, y una mayor eficiencia en la gestión operativa del camping. Además, la implementación del sistema tiene la posibilidad de crear un impacto positivo en la economía local al aumentar la cantidad de visitantes y fomentar el crecimiento de negocios circundantes.

En este proyecto integrador se explicará cómo la adopción de tecnologías de la información no solo resuelve problemas operativos inmediatos, sino que también posiciona al Camping Arroyito como un modelo de negocio moderno y sostenible en el competitivo mercado turístico. También se detallará la forma en la que el sistema Creek junto con una página web fueron creados e implementados y como estos ayudaran a cumplir los objetivos propuestos.

# ABSTRACT

The tourism industry faces increasingly complex challenges, especially for small family businesses like Camping Arroyito, located in San Juan, Argentina. The lack of adequate technological tools for efficient operations management has led to recurring issues such as booking errors, disorganization in the management of cabins and tent's spots, and difficulties in providing a satisfactory customer experience. This project aims to address these problems through the design and implementation of a web management system that modernizes and optimizes the campground's operations.

The main objective is to provide a technological solution that automates key processes, such as booking management, tracking the status of cabins, and accounting administration, thereby improving internal efficiency and customer satisfaction. To achieve this, a development methodology based on a multi-layer architecture (frontend, backend, and database) was employed, using modern technologies such as React, Node.js, and MongoDB. Additionally, interviews, surveys, and direct observation were conducted to understand the needs of end-users and ensure the system is intuitive and functional.

The implementation of this system is expected to result in a significant reduction in administrative errors, an improved user experience for customers, and greater operational efficiency for the campground. Furthermore, the system has the potential to positively impact the local economy by increasing the number of visitors and fostering the growth of surrounding businesses.

This thesis explains how the adoption of information technologies not only addresses immediate operational challenges but also positions Camping Arroyito as a modern and sustainable business model in the competitive tourism market. It also details the creation and implementation of the "Creek system" and a complementary website, demonstrating how these tools help achieve the proposed objectives.

# PRESENTACIÓN DEL TEMA

En el panorama actual, la industria del turismo se encuentra en constante transformación, impulsada por el cambio en las preferencias de los consumidores y el auge de nuevas tecnologías. Dentro de este sector, los campings y complejos turísticos han experimentado un crecimiento notable, consolidándose como destinos preferidos por los viajeros que buscan una experiencia en contacto con la naturaleza y una alternativa al ritmo acelerado de la vida urbana. Sin embargo, estos establecimientos enfrentan desafíos importantes, especialmente en lo que respecta a la gestión eficiente de sus operaciones y la satisfacción de las crecientes expectativas de los clientes.

A medida que aumenta la demanda en este segmento turístico, la competencia se intensifica, obligando a los campings a buscar soluciones innovadoras que les permitan destacar y mantenerse relevantes en el mercado. En este contexto, Camping Arroyito, un negocio familiar ubicado en la provincia de San Juan, se enfrenta a una serie de retos relacionados con la gestión manual de sus procesos operativos, lo que limita su capacidad para optimizar recursos, garantizar un servicio ágil y adaptarse a las tendencias tecnológicas actuales. Esta situación no solo afecta la operación interna del camping, sino que también representa una oportunidad para explorar la implementación de herramientas tecnológicas que mejoren su desempeño y posicionamiento en el mercado.

La problemática identificada en Camping Arroyito no es un caso aislado, sino un reflejo de una necesidad más amplia en la industria de los campings: la incorporación de tecnología como medio para mejorar la experiencia del cliente y aumentar la eficiencia operativa. Actualmente, muchos campings dependen de métodos tradicionales para la gestión de reservas, la administración de recursos y la organización interna, lo que puede llevar a errores, pérdidas de tiempo y una limitada capacidad para analizar datos y tomar decisiones estratégicas.

En este marco, el propósito del proyecto es desarrollar una solución tecnológica a medida: un sistema de gestión basado en una página web que permita a Camping Arroyito modernizar sus procesos, automatizar tareas repetitivas y ofrecer una

experiencia más eficiente tanto para los administradores como para los clientes. Este sistema no sólo atenderá las necesidades específicas del camping, sino que también tendrá el potencial de convertirse en un modelo replicable para otros negocios similares dentro del sector turístico.

El proyecto busca, además, posicionar a Camping Arroyito como un referente en la adopción de tecnología en su segmento, demostrando cómo una herramienta de gestión puede ser clave para enfrentar los desafíos de la industria, mejorar la calidad del servicio y responder a las demandas de un mercado cada vez más competitivo.



# GLOSARIO

**Parcela:** Espacio físico en el camping destinado a que los clientes puedan alquilarlos y colocar carpas, motorhomes o gazebos

**Check-In:** Fecha en la que el cliente ingresa al camping.

**Check-Out:** Fecha hasta la que el cliente se hospeda en el camping, es decir, es el día en el que abandona el camping.

## DIAGNÓSTICO (PROBLEMÁTICA)

Camping Arroyito, un negocio familiar exitoso ubicado en San Juan, ha logrado consolidarse como un destino turístico atractivo gracias a su propuesta de contacto con la naturaleza y su enfoque en la hospitalidad. Este establecimiento se distingue por ofrecer un espacio tranquilo y acogedor para turistas que buscan escapar del bullicio de la ciudad y disfrutar de la belleza natural de la región. Con áreas verdes amplias, cabañas cómodas y servicios diseñados para el descanso, Camping Arroyito ha logrado captar la atención de un público diverso, desde familias hasta aventureros solitarios.

No obstante, a pesar de su éxito, enfrenta una serie de desafíos que amenazan con limitar su crecimiento y su capacidad de brindar una experiencia de calidad a sus clientes. Estos desafíos están relacionados principalmente con la gestión de sus operaciones diarias, que actualmente dependen en gran medida de procesos manuales y métodos de gestión desactualizados e insuficientes para los tiempos actuales en donde cada vez se demanda más rapidez, precisión y certeza. En el mundo actual, donde la tecnología juega un papel fundamental en la eficiencia operativa y la satisfacción del cliente, esta situación coloca a Camping Arroyito en una posición vulnerable frente a sus competidores más modernizados.

Uno de los problemas más evidentes es la falta de organización y pérdida de información que existen debido a que no hay una forma que permita administrar de manera eficiente las reservas, gastos e ingresos, los empleados y el control de ocupación de las cabañas y parcelas del camping. Actualmente, estos procesos se llevan a cabo mediante registros manuales o herramientas digitales básicas, como hojas de cálculo. Esta metodología no solo consume tiempo, sino que también aumenta el riesgo de errores humanos, como la sobreventa de cabañas o la asignación incorrecta de recursos. Además, la falta de un sistema automatizado dificulta la obtención de datos precisos y en tiempo real sobre el estado del negocio, lo que limita la capacidad de los propietarios para tomar decisiones informadas y estratégicas.

Otro aspecto problemático es la comunicación con los clientes. En un mercado

donde los turistas esperan servicios rápidos y eficientes, la ausencia de un canal de comunicación digital integrado representa una desventaja significativa. Los clientes no solo buscan realizar reservas de manera sencilla, sino también recibir confirmaciones inmediatas, consultar información sobre disponibilidad y resolver dudas sin demoras. La falta de herramientas digitales que faciliten esta interacción puede generar frustración en los usuarios y afectar negativamente la percepción del camping.

Asimismo, el camping enfrenta dificultades para monitorear y analizar el desempeño general de sus operaciones. Sin un sistema que permita recopilar y procesar datos de manera efectiva, resulta complicado identificar patrones de uso, evaluar la rentabilidad de las cabañas y planificar estrategias de mejora a largo plazo. Esto no solo limita la capacidad de optimizar los recursos disponibles, sino que también reduce las oportunidades de innovación y crecimiento en un entorno competitivo.

En resumen, aunque Camping Arroyito ha demostrado ser un negocio con gran potencial y atractivo, su dependencia de procesos tradicionales y la falta de integración tecnológica están impidiendo que alcance un nivel superior de eficiencia y competitividad. Esta situación subraya la necesidad de implementar un sistema de gestión web que no solo modernice sus operaciones, sino que también contribuya a mejorar la experiencia del cliente y a impulsar el desarrollo sostenible del negocio en el futuro.

### **Estado del Arte (Contexto Actual)**

La gestión de Camping Arroyito se realiza actualmente de forma manual, lo que genera una serie de complicaciones operativas que impactan tanto en la eficiencia del negocio como en la experiencia del cliente. Las reservas, el control de la disponibilidad de cabañas y parcelas, y la administración de recursos humanos y financieros carecen de un sistema centralizado que facilite estas tareas. Esta metodología, basada en registros manuales y herramientas digitales básicas como hojas de cálculo, no solo incrementa la posibilidad de errores humanos, sino que también dificulta la planificación estratégica y la optimización de recursos.

Uno de los principales problemas derivados de esta situación es la falta de precisión

y rapidez en los procesos administrativos. Por ejemplo, la gestión manual de reservas puede dar lugar a errores como la sobreventa de cabañas o la asignación incorrecta de parcelas, generando insatisfacción en los clientes y, en algunos casos, pérdida de ingresos. Además, la ausencia de un sistema automatizado limita la capacidad de los propietarios para obtener información confiable y en tiempo real sobre el estado del camping, lo que dificulta la toma de decisiones informadas y oportunas.

En el ámbito de la gestión de recursos humanos, la falta de digitalización también representa un obstáculo. Las tareas administrativas relacionadas con el personal, como la asignación de horarios, tareas y vacaciones, se llevan a cabo de manera manual, lo que incrementa la carga laboral y reduce la eficiencia del equipo de trabajo. Este enfoque también dificulta la identificación de áreas de mejora y la implementación de estrategias para optimizar el rendimiento del personal.

En un mercado turístico cada vez más competitivo, donde los clientes valoran la facilidad y rapidez en los procesos, la falta de digitalización se convierte en un obstáculo significativo para el crecimiento y la competitividad del camping. Actualmente, los turistas esperan poder realizar reservas de manera sencilla y recibir confirmaciones inmediatas, así como tener acceso a información actualizada sobre la disponibilidad de servicios y cabañas. Sin embargo, Camping Arroyito no cuenta con las herramientas tecnológicas necesarias para satisfacer estas expectativas, lo que puede afectar negativamente su reputación y su capacidad para atraer nuevos clientes.

Mientras otras empresas del sector han comenzado a implementar soluciones tecnológicas para mejorar su operatividad y ofrecer una experiencia más eficiente, Camping Arroyito aún no ha dado este paso, lo que lo coloca en desventaja frente a sus competidores. Sistemas de gestión web, aplicaciones móviles y herramientas de análisis de datos son algunas de las tecnologías que ya están siendo utilizadas por otros campings y establecimientos turísticos para optimizar sus procesos y mejorar la experiencia del cliente. Estos avances no solo permiten reducir costos operativos, sino también incrementar la satisfacción de los usuarios y generar mayores ingresos a largo plazo.

Además, la falta de integración tecnológica limita la capacidad del camping para adaptarse a las tendencias del mercado y a las demandas de los consumidores modernos. Por ejemplo, la posibilidad de realizar reservas en línea o consultar disponibilidad en tiempo real son aspectos que los clientes consideran esenciales al momento de elegir un destino turístico. Sin estas funcionalidades, Camping Arroyito corre el riesgo de perder oportunidades de negocio y de quedar rezagado en un sector que evoluciona rápidamente.

En el contexto actual, donde la tecnología se ha convertido en un elemento clave para la gestión empresarial, es evidente que Camping Arroyito necesita modernizar sus operaciones para mantenerse competitivo. La implementación de un sistema de gestión web centralizado podría solucionar gran parte de los problemas mencionados, permitiendo automatizar procesos, reducir errores, y ofrecer una experiencia más satisfactoria tanto para los clientes como para el personal del camping. Este tipo de soluciones no solo mejoran la eficiencia operativa, sino que también proporcionan herramientas para el análisis de datos y la planificación estratégica, lo que resulta fundamental para garantizar el crecimiento sostenible del negocio.

La situación actual de Camping Arroyito refleja una clara necesidad de transformación digital. Aunque el camping ha logrado posicionarse como un destino atractivo gracias a su propuesta de valor, su capacidad para mantenerse competitivo y satisfacer las expectativas de los clientes dependerá en gran medida de su disposición para adoptar nuevas tecnologías y modernizar sus procesos de gestión.

## **Impacto**

La falta de implementación de soluciones digitales en Camping Arroyito genera un impacto significativo tanto en la experiencia de los clientes como en la eficiencia del personal y la administración del establecimiento. La dependencia de una gestión manual trae consigo limitaciones que afectan diversas áreas operativas y comprometen la competitividad del camping en un sector donde la tecnología se ha convertido en un elemento fundamental.

### Impacto en los clientes:

Para los clientes, la ausencia de herramientas digitales y procesos automatizados puede resultar en una experiencia menos satisfactoria, lo que influye directamente en su percepción del camping y en su disposición a regresar o recomendar el lugar a otros. Algunos de los principales problemas que enfrentan los clientes incluyen:

1. Errores en las reservas:

- La gestión manual aumenta la probabilidad de errores en las reservas, como asignaciones incorrectas de cabañas o parcelas.
- Los conflictos en las fechas de disponibilidad pueden generar inconvenientes para los clientes, quienes esperan un servicio confiable y preciso.
- Estos errores pueden derivar en situaciones incómodas, como la necesidad de reubicar a los clientes o cancelar reservas, lo que afecta negativamente la experiencia general.

2. Falta de acceso a información actualizada:

- Los clientes no tienen la posibilidad de consultar en tiempo real la disponibilidad de cabañas, parcelas u otros servicios ofrecidos por el camping.
- La falta de transparencia y rapidez en la comunicación de información puede generar frustración, especialmente para aquellos que buscan planificar sus viajes con antelación.

3. Errores en el cobro:

- La forma de cobro actual (anotar en un excel o libreta el monto a cobrar a cada persona) puede causar pérdida de información, errores a la hora de cobrar e inconsistencias en los cobros
- Al no existir una forma segura y centralizada en donde consultar el precio de las cabañas/parcelas, pueden ocurrir situaciones en donde se cobre de más o de menos a los clientes

4. Experiencia de usuario limitada:

- En un mundo donde las plataformas digitales se han convertido en la norma, la ausencia de opciones para realizar reservas en línea o acceder a información de manera digital representa una experiencia limitada para los clientes.

- Los turistas modernos valoran la conveniencia y la eficiencia en los procesos, y la falta de estas opciones puede llevar a los clientes a optar por competidores que ofrezcan servicios más modernos.

### Impacto en el personal y la administración

Para el equipo de trabajo y la administración de Camping Arroyito, la gestión manual representa una carga significativa que afecta la productividad, la organización y la capacidad de tomar decisiones informadas. Entre las principales consecuencias se encuentran:

#### 1. Procesos lentos y tediosos:

- La organización de reservas, el control de recursos y la administración general del camping requieren un esfuerzo considerable al depender de registros manuales o herramientas no integradas.
- Estas tareas consumen tiempo valioso que podría destinarse a actividades más estratégicas o enfocadas en la mejora de la experiencia del cliente.

#### 2. Falta de datos consolidados:

- La ausencia de un sistema centralizado dificulta la recopilación y el análisis de datos relevantes sobre el desempeño del camping, como tasas de ocupación, ingresos generados y patrones de comportamiento de los clientes.
- Sin esta información, es complicado identificar áreas de mejora, planificar estrategias a futuro y evaluar la efectividad de las decisiones tomadas.

#### 3. Riesgo de accesos no controlados:

- La gestión manual implica un mayor riesgo de accesos no controlados a información sensible, como datos personales de los clientes, registros financieros o detalles de reservas.
- Este problema no solo compromete la seguridad y privacidad de los datos, sino que también puede afectar la confianza de los clientes y

exponer al camping a posibles sanciones legales.

4. Pérdida de datos importantes:

- Los registros manuales están sujetos a errores humanos, extravíos o daños físicos, lo que puede resultar en la pérdida de información clave para la gestión del camping.
- La falta de un respaldo digital de los datos pone en riesgo la continuidad operativa y dificulta la reconstrucción de información en caso de imprevistos.

Impacto general en el negocio

En términos generales, estas limitaciones operativas afectan la capacidad de Camping Arroyito para competir en un mercado donde la tecnología se ha convertido en un diferenciador clave. La falta de digitalización no solo reduce la eficiencia del negocio, sino que también pone en riesgo su reputación y su capacidad para atraer y retener clientes en un entorno cada vez más exigente.

Implementar soluciones digitales no es solo una mejora operativa, sino una necesidad estratégica para garantizar la sostenibilidad y el crecimiento del camping. Al adoptar un sistema de gestión digital, Camping Arroyito podría optimizar sus procesos, reducir errores, y mejorar tanto la experiencia del cliente como la productividad de su personal, asegurando así su lugar en el mercado turístico moderno.

**Oportunidades**

A pesar de los desafíos que enfrenta Camping Arroyito en su gestión actual, la situación también representa una oportunidad única para transformar su operación mediante la implementación de una solución innovadora basada en tecnologías digitales. En un entorno donde la digitalización está redefiniendo la forma en que las empresas interactúan con sus clientes y gestionan sus recursos, adoptar un sistema de gestión moderno podría posicionar al camping como un líder en su segmento y garantizar su sostenibilidad a largo plazo. A continuación, se analizan en detalle las principales oportunidades que podrían surgir de esta transformación:



### Automatización y eficiencia operativa

La implementación de un sistema de gestión digital permitiría centralizar y automatizar los procesos clave del camping, lo que se traduciría en una operación más eficiente y precisa. Algunas de las ventajas específicas incluyen:

- Reducción de errores humanos: Al automatizar tareas como la asignación de cabañas, la gestión de reservas y el control de disponibilidad, se minimizarían los errores derivados de la intervención manual.
- Optimización de recursos: La centralización de la información permitiría una asignación más efectiva de recursos, como el personal y las instalaciones, maximizando la productividad del camping.
- Ahorro de tiempo: Los procesos automatizados reducirían el tiempo necesario para realizar tareas administrativas, permitiendo al personal enfocarse en actividades que aporten mayor valor, como la atención al cliente.

### Mejora en la experiencia del cliente

Una de las principales oportunidades radica en la capacidad de mejorar significativamente la experiencia del cliente, lo que podría incrementar tanto la satisfacción como la fidelización. Entre las mejoras más destacadas se encuentran:

- Reservas en línea: Los clientes podrían realizar sus reservas de manera rápida y sencilla a través de una plataforma digital, sin necesidad de comunicarse directamente con el personal del camping.
- Consulta de disponibilidad en tiempo real: Una solución digital permitiría a los usuarios verificar la disponibilidad de cabañas y parcelas en cualquier momento, facilitando su planificación y aumentando su confianza en el servicio.
- Acceso a información detallada: A través de un sistema digital, los clientes podrían obtener información actualizada sobre los servicios disponibles, precios, actividades, lo que enriquecería su experiencia.
- Personalización del servicio: Al recopilar y analizar datos sobre las preferencias de los clientes, el camping podría ofrecer experiencias más personalizadas, aumentando la probabilidad de repetir la visita.

### Toma de decisiones basada en datos

Otro beneficio importante de la digitalización es la posibilidad de recolectar y analizar datos de manera efectiva para respaldar la toma de decisiones estratégicas. Esto incluye:

- Generación de informes confiables: Los datos recopilados sobre reservas, ocupación, ingresos y gastos podrían transformarse en informes detallados que permitan evaluar el desempeño del camping.
- Identificación de tendencias: Mediante el análisis de datos históricos, se podrían identificar patrones y tendencias que ayuden a anticipar la demanda y ajustar las estrategias de marketing y operación.
- Planificación a futuro: Con información confiable y en tiempo real, la administración podría planificar de manera más efectiva el crecimiento del camping, incluyendo posibles expansiones, mejoras en las instalaciones y nuevas ofertas de servicios.

### Ventaja competitiva

En un mercado altamente competitivo, donde cada vez más empresas adoptan tecnología para mejorar sus operaciones, Camping Arroyito tiene la oportunidad de destacar implementando una solución digital innovadora. Las ventajas incluyen:

- Posicionamiento como líder en tecnología: Al adoptar un sistema de gestión moderno, el camping podría diferenciarse de sus competidores que aún operan de manera manual o con sistemas menos avanzados.
- Atracción de nuevos clientes: Una experiencia más conveniente y eficiente podría atraer a clientes que valoren la tecnología como un factor clave en su elección de destino turístico.
- Reputación mejorada: La implementación de tecnología también podría reflejarse positivamente en la percepción del camping como un negocio moderno y comprometido con la calidad del servicio.

Los desafíos actuales que enfrenta Camping Arroyito también representan una oportunidad valiosa para modernizar sus operaciones y posicionarse como un referente en su sector. Al implementar soluciones digitales, el camping no solo podría resolver los problemas asociados a la gestión manual, sino también

aprovechar las ventajas de la automatización, mejorar la experiencia del cliente, basar sus decisiones en datos confiables y ganar una ventaja competitiva significativa. Este enfoque estratégico no solo garantizará la sostenibilidad del negocio, sino que también contribuirá a su crecimiento y éxito a largo plazo.

# OBJETIVOS

## GLOBAL

- Digitalizar y centralizar la información y tareas de gestión del camping para que permita a los empleados y administradores del camping realizar su labor de forma más ágil, sencilla y eficiente, reduciendo tiempos de trabajo así como también errores manuales.
- Proporcionar una solución digitalizada que también facilita el acceso a la información relevante para el manejo del camping de forma rápida y fácil, lo cual permitirá una mejor toma de decisiones.
- Ofrecer un medio digital destinado a mejorar la experiencia de los clientes del camping

## ESPECÍFICOS

- Se busca ampliar tanto la presencia web del camping como el alcance a potenciales huéspedes y que también facilitar la realización de reservas por parte de los clientes mediante el uso de una página web.
- Garantizar el almacenamiento, gestión y disponibilidad de información relevante sobre las reservas, clientes, finanzas y empleados de manera segura, confidencial y permanente con el fin de ayudar con la gestión del camping.
- Propiciar y controlar el acceso a la información implementando niveles de permiso según el rol de los usuarios (empleados, administradores, clientes) de forma que únicamente determinados usuarios puedan visualizar determinados datos
- Disponer de un sistema que le permita a los empleados del camping visualizar fácilmente información relevante a través de herramientas visuales para ayudar con la toma de decisiones estratégicas y optimizar la administración de las cabañas, parcelas y reservas.

# MARCO TEÓRICO

## Contexto General del Problema

La industria del turismo es un sector que se encuentra en constante crecimiento y evolución. Según la Organización Mundial del Turismo (OMT), el turismo no solo representa una de las principales fuentes de ingresos y empleo para las economías de un país, sino que también juega un papel fundamental para áreas rurales y naturales, en donde se convierte en un catalizador para el desarrollo económico y social de estas regiones. Muchas veces, estos destinos logran mantenerse en pie gracias al flujo de turistas que atraen.

En este contexto, los asentamientos turísticos tales como hoteles, campings y otros complejos juegan un papel fundamental a la hora de atraer y retener turistas. Al momento de viajar, cada persona arma una experiencia de viaje que se ajuste a lo que desea y necesita, siempre buscando tener la mejor experiencia posible. En el caso de los campings, estos ofrecen experiencias únicas en contacto con la naturaleza, fomentando la desconexión de la vida urbana y el turismo sostenible.

Hoy en día, el auge del “turismo de naturaleza” y el interés por experiencias sostenibles han convertido a los campings en una opción atractiva para una amplia variedad de viajeros. Son cada vez más las personas que optan por alojarse en campings cuando viajan, ya que buscan una escapatoria del estrés de la vida cotidiana, de la ciudad y de sus problemas, desde familias que buscan unas vacaciones más económicas hasta viajeros con un perfil más aventurero, que buscan una experiencia más minimalista y con mayor acercamiento a la naturaleza, los campings son vistos como la solución perfecta: son económicos, flexibles y sobre todo auténticos. Además, en un mundo cada vez más enfocado en alternativas “eco-friendly”, estos establecimientos sostenibles son percibidos como una opción ecológica, que promueve el respeto, la conservación del medio ambiente y ayuda económicamente a áreas marginadas. Es por este motivo que muchos gobiernos provinciales promocionan cada vez más el turismo de naturaleza, para ayudar económicamente a sectores que muchas veces son obviados por los turistas. Tal es el caso, por ejemplo, del Gobierno de la provincia de Misiones, que gracias a sus esfuerzos promoviendo actividades en entornos naturales y experiencias

sustentables, ha logrado atraer 50 mil turistas extranjeros por año quienes buscan realizar turismo de naturaleza.

Sin embargo, este crecimiento también ha generado que se intensifique la competencia, y a su vez, los clientes esperan un nivel de servicio apto y eficiente, que les permita tener una experiencia satisfactoria y en donde no ocurran problemas inesperados. En estos tiempos modernos, hay expectativas arraigadas a la experiencia de turismo moderno las cuales los clientes esperan que se cumplan, tales como poder realizar reservas en línea, conocer los servicios que se ofrecen de manera previa, una atención al cliente eficiente entre otras experiencias personalizadas que los hagan sentir únicos y valorados, lo cual obliga a los administradores de los campings a modernizarse, innovar y optimizar sus operaciones para satisfacer estas expectativas de los clientes y ganar ventaja por sobre otros competidores.

En este escenario de evolución y competitividad constante, el Camping Arroyito se encuentra inmerso en una dinámica que lo obliga a adaptarse a las nuevas exigencias del mercado turístico. Este establecimiento, de carácter familiar, está ubicado en la localidad de Rodeo, a aproximadamente 195 kilómetros de la ciudad de San Juan, en la provincia homónima. Rodeo ha experimentado en los últimos años un notable crecimiento en términos de afluencia turística, posicionándose como uno de los destinos emergentes más atractivos del noroeste argentino.

Este crecimiento se debe, en gran parte, a la cercanía del Dique Cuesta del Viento, un espejo de agua reconocido internacionalmente por ser sede de eventos de windsurf y kitesurf. El más destacado de estos es el Kitefest, considerado el evento de kitesurf más grande de América, el cual atrae cada año a miles de turistas, tanto competidores como espectadores, durante la semana en que se desarrolla.

Además del deporte, Rodeo ofrece una amplia variedad de actividades vinculadas al turismo de naturaleza, como caminatas guiadas, cabalgatas y deportes náuticos, lo que refuerza su perfil como destino eco-friendly. Durante el mes de enero, la localidad también celebra el Festival Provincial de la Semilla y la Manzana, que convoca a visitantes con propuestas culturales, espectáculos artísticos, comidas regionales y artesanías locales.

Este conjunto de eventos y atractivos naturales ha impulsado significativamente la infraestructura turística de la región, provocando una expansión en la oferta de hospedaje. En este contexto, el Camping Arroyito se enfrenta al desafío de destacarse dentro de un entorno cada vez más competitivo. La necesidad de modernizar su gestión, optimizar su operatividad y brindar una experiencia acorde a las expectativas del turista contemporáneo se vuelve crucial para sostener su posicionamiento en el mercado local.

### **Trabajos relacionados**

Uno de los antecedentes más significativos para este proyecto es el trabajo final de Ingeniería Industrial “Plan de Negocios y Mejora de Procesos para Complejo Turístico de Cabañas y Camping ‘Arroyito’”, creado por los autores Marcos Gho Silva y Chloe Vidale, presentado en el XVI Congreso Internacional de Ingeniería Industrial (COINI) en 2023 y publicado en 2024 por la Universidad Tecnológica Nacional. En dicho trabajo se realiza un diagnóstico profundo del estado organizativo y productivo del Camping Arroyito, identificando diversas oportunidades de mejora y crecimiento a futuro.

Entre los principales aspectos señalados, se destaca la ausencia de planificación estratégica, la falta de estandarización de procesos, la carencia de métricas e indicadores de gestión y la inexistencia de un sistema centralizado de atención al cliente. A partir de este diagnóstico, los autores proponen una serie de estrategias de mejora, dentro de las cuales se incluye la necesidad de implementar un sistema integrado de gestión, capaz de optimizar la operación del establecimiento y permitir un control más riguroso sobre aspectos clave como las reservas, los empleados, el mantenimiento, el stock y el servicio al cliente.

Asimismo, el trabajo sugiere la importancia de incorporar herramientas tecnológicas que permitan acceder a métricas e indicadores de desempeño para tomar decisiones basadas en datos. También se destaca la necesidad de modernizar la comunicación y comercialización del camping, por ejemplo, a través del desarrollo de una página web institucional, que funcione como canal de promoción y de gestión de reservas en línea.

Este proyecto integrador se alinea directamente con dichas recomendaciones, proponiendo el diseño e implementación de un sistema de gestión digital que responda a esas necesidades detectadas, contribuyendo así a la transformación digital del Camping

Arroyito y a su posicionamiento competitivo en el mercado turístico regional.

### **Procedimientos actuales en la gestión de campings**

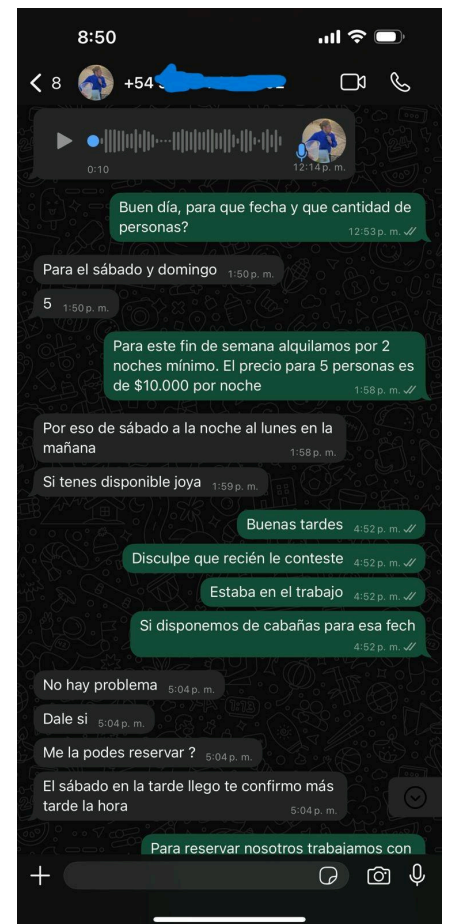
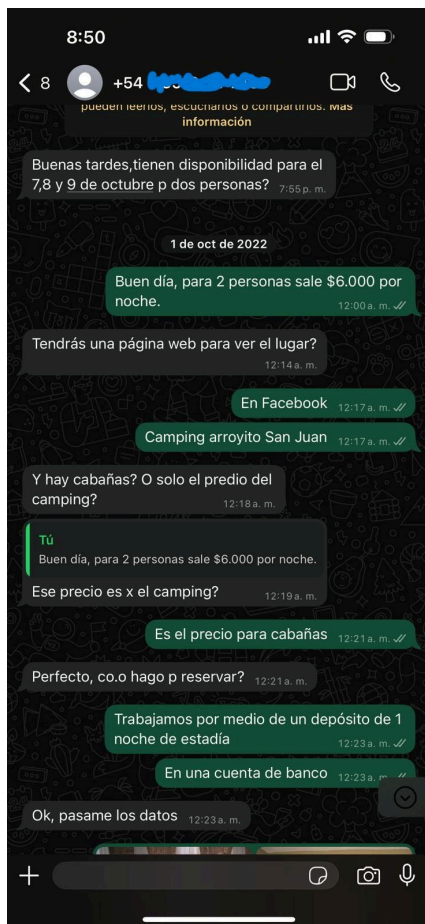
A pesar de su creciente popularidad, muchos campings continúan gestionando sus operaciones de manera manual. Esto incluye la reserva de parcelas y cabañas, el seguimiento de la ocupación, la administración de finanzas y recursos humanos, así como la atención al cliente.

El Camping Arroyito no es ajeno a estas problemáticas. Actualmente, la totalidad de sus procesos de gestión se realizan de forma manual. En lo que respecta a las reservas, los clientes sólo pueden comunicarse a través de dos medios: llamadas telefónicas o mensajes vía WhatsApp. En el pasado, también se utilizaba la mensajería de Facebook como canal de contacto, sin embargo, fue discontinuada debido a la dificultad que implicaba mantener activa la atención en múltiples plataformas y porque el acceso a esta red era exclusivo de los administradores.

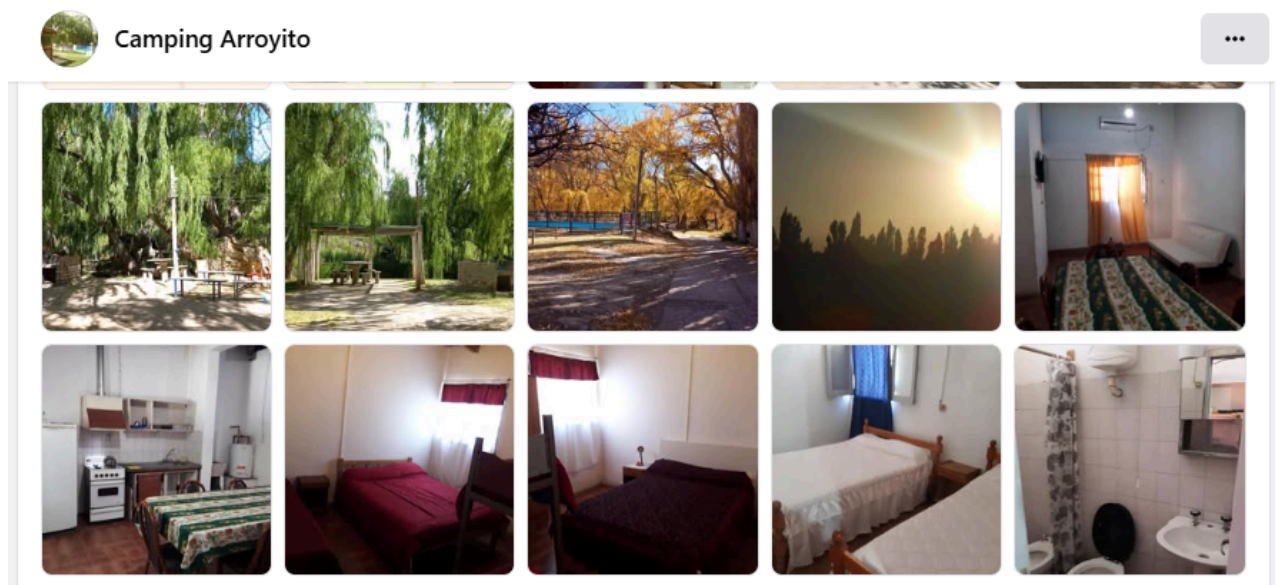
A través de estos canales, los clientes solicitan información sobre la disponibilidad de cabañas en fechas específicas, los precios correspondientes para hospedarse tanto en cabañas como en carpas, y remiten los comprobantes de pago. Si bien este método resulta directo y cercano, es altamente ineficiente y propenso a errores. En la actualidad, los usuarios esperan contar con información clara e inmediata, sin tener que depender de intermediarios que, por cuestiones operativas o de disponibilidad, pueden demorar varios minutos u horas en responder.

Asimismo, la falta de una plataforma digital centralizada obliga a los empleados a enviar manualmente fotografías e información adicional sobre las cabañas y servicios a través de los mismos canales de comunicación. En muchos casos, se deriva al cliente hacia la página de Facebook del camping, la cual no cuenta con una actualización periódica ni con material gráfico suficiente que refleje la totalidad de los servicios ofrecidos, lo que puede generar desconfianza o falta de interés por parte de potenciales huéspedes.





*chats con potenciales clientes del camping*



*Página de facebook con imágenes de los servicios y comodidades del hospedaje*

Otro de los procesos que actualmente se realiza de forma manual en el Camping Arroyito es la reserva y asignación de cabañas. Para ello, se utiliza un archivo de Excel

que contiene un calendario de ocupación y tiene como fin organizar la información. El procedimiento comienza en la hoja denominada "Disponibilidad", donde el personal verifica qué cabañas se encuentran libres en la fecha solicitada por el cliente. Cada cabaña está identificada con un número único que facilita su reconocimiento y está clasificada según su capacidad y tipo: Estándar o Deluxe. Las cabañas de tipo Deluxe poseen comodidades adicionales, como cochera, televisión y aire acondicionado, lo que las distingue de las estándar.

Una vez que se confirma la disponibilidad y el cliente decide avanzar con la reserva, se procede a recolectar sus datos personales, tales como nombre completo, número de documento, teléfono y dirección. Esta información es registrada manualmente en la pestaña "Reservas" del mismo archivo, junto con las fechas de check-in y check-out, y la cantidad de adultos y niños que se hospedarán.

Posteriormente, se calcula el monto total a abonar por la estadía, y si el cliente realiza una seña o pago anticipado, dicho importe se consigna también en la hoja. El archivo de Excel cuenta con una fórmula que permite calcular automáticamente el saldo restante, el cual debe ser abonado por el cliente al momento de su ingreso al camping.

Si bien este sistema ha permitido llevar un cierto control organizativo, su carácter manual lo vuelve susceptible a errores humanos, como superposiciones de reservas, omisión de datos o errores de cálculo. Además, la dependencia exclusiva de una planilla local limita el acceso a la información en tiempo real por parte de otros empleados o administradores, lo cual dificulta la coordinación eficiente del personal y la atención al cliente. Esta metodología, aunque funcional en escenarios de baja demanda, se vuelve cada vez más insostenible a medida que aumenta la cantidad de huéspedes, generando retrasos, confusión y pérdida de oportunidades comerciales.

| Reservas del camping |                 |               |           |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     | 01-abr-25 | 30-abr-25 |
|----------------------|-----------------|---------------|-----------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----------|-----------|
| Nº                   | Tipo habitación | Número Cabaña | Capacidad | mar | mié | jue | vie | sáb | dom | lun | mar | mié | jue | vie | sáb | dom | lun | mar | mié       | jue       |
| 1                    | Standard        | 1             | 4         |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |           |           |
| 2                    | Standard        | 2             | 4         |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |           |           |
| 3                    | Standard        | 3             | 5         |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |           |           |
| 4                    | Deluxe          | 4             | 2         |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |           |           |
| 5                    | Deluxe          | 5             | 6         |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |           |           |
| 6                    | Standard        | 6             | 5         |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |           |           |
| 7                    | Standard        | 7             | 6         |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |           |           |
| 8                    | Deluxe          | 8             | 6         |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |           |           |
| 9                    | Standard        | 9             | 2         |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |           |           |
| 10                   | Standard        | 10            | 2         |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |           |           |
| 11                   | Deluxe          | 11            | 4         |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |           |           |
| 12                   | Deluxe          | 12            | 4         |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |           |           |

■ Disponible  
■ Ocupado

| Telefono Hués | Procedencia                       | Check In  | Check Out | N° Adulto | N° Niños | Tipo de Habitación | N° de Cabañ | Total a Pagar | Depósito     | Saldo remanente |
|---------------|-----------------------------------|-----------|-----------|-----------|----------|--------------------|-------------|---------------|--------------|-----------------|
| 2645623737    | Azuénaga 6814                     | 2/4/2025  | 4/4/2025  | 6         |          | Deluxe             | 5           | \$500.000,00  | \$500.000,00 | \$0,00          |
| 2644844748    | Agustin Gomez y fias s/n Rawson   | 3/4/2025  | 5/4/2025  | 2         |          | Deluxe             | 4           | \$80.000,00   | \$18.000,00  | -\$62.000,00    |
| 2645148521    | Mitre 1963 San Juan               | 6/4/2025  | 9/4/2025  | 5         |          | Deluxe             | 5           | \$225.000,00  | \$67.500,00  | -\$157.500,00   |
| 2644714229    | Loreto 76 Rawson                  | 10/4/2025 | 14/4/2025 | 2         |          | Standard           | 3           | \$120.000,00  | \$36.000,00  | -\$84.000,00    |
| 2645545567    | Gral Paz 2488 (E) Sta Lucía       | 15/4/2025 | 20/4/2025 | 2         | 2        | Standard           | 1           | \$225.000,00  | \$67.500,00  | -\$157.500,00   |
| 2644416653    | Independencia 33 Albardon         | 17/4/2025 | 20/4/2025 | 5         |          | Standard           | 3           | \$225.000,00  | \$67.500,00  | -\$157.500,00   |
| 2644744322    | Barrio CGT Mzna N Csa 3           | 17/4/2025 | 20/4/2025 | 2         |          | Deluxe             | 4           | \$90.000,00   | \$27.000,00  | -\$63.000,00    |
| 2615315540    | Pellegrini 287 Mendoza            | 17/4/2025 | 20/4/2025 | 2         | 1        | Standard           | 7           | \$112.500,00  | \$33.750,00  | -\$78.750,00    |
|               | San Luis 954 (E)                  | 17/4/2025 | 20/4/2025 | 2         |          | Standard           | 10          | \$90.000,00   | \$27.000,00  | -\$63.000,00    |
| 2804010977    | Conesa 347 Esquel Chubut          | 17/4/2025 | 20/4/2025 | 3         |          | Deluxe             | 11          | \$135.000,00  | \$40.500,00  | -\$94.500,00    |
| 2645545567    | Gral Paz 2488 (E) Sta Lucía       | 23/4/2025 | 29/4/2025 | 5         |          | Deluxe             | 5           | \$450.000,00  | \$135.000,00 | -\$315.000,00   |
| 1165530598    | Aristóbulo del Valle 2417 Florida | 29/4/2025 | 5/5/2025  | 3         |          | Deluxe             | 11          | \$270.000,00  | \$81.000,00  | -\$189.000,00   |

Tal como se puede ver, en la pestaña de reservas también suelen anotarse otros ingresos complementarios del camping como el del alquiler mensual de una de las cabañas, de forma que el documento ocasionalmente suele funcionar como un registro general de ingresos, lo cual no es lo más óptimo.

De forma similar se lleva también el registro de los clientes que se hospedan en las parcelas del camping. Cuando un cliente que desea alquilar una parcela llega al establecimiento, el personal solicita sus datos personales (nombre, número de documento, entre otros), los cuales son asentados en el mismo archivo. También se registra el monto a cobrar por la estadía, el valor abonado (el cual se puede hacer en 2 partes) y, al igual que con las reservas de cabañas, el Excel realiza un cálculo automático para determinar el saldo remanente. Los registros generados se agrupan de manera manual y luego se filtran por mes para facilitar su revisión.

| Nombre Húesped   | N° de Documento | Telefono Hu | Procedencia                           | Check In   | Check Out  | N° Adultos | N° Niños | Total a Pagar | Depósito  | Saldo remante |
|------------------|-----------------|-------------|---------------------------------------|------------|------------|------------|----------|---------------|-----------|---------------|
| Mariano Sosa     | 36.554.332      | 2644987654  | B° América chimbas                    | 16/02/2025 | 18/02/2025 | 1          | 0        | \$ 15.000     | \$ 10.000 | \$ 5.000      |
| Lucas Ávila      | 39.987.654      | 3876543210  | B° FOECYT Rawson                      | 06/02/2025 | 08/02/2025 | 1          | 1        | \$ 25.000     | \$ 10.000 | \$ 15.000     |
| Martina Gómez    | 42.001.223      | 2614563210  | Calle General Acha 2300 Rawson        | 13/02/2025 | 15/02/2025 | 2          | 0        | \$ 45.000     | \$ 25.000 | \$ 20.000     |
| Javier Romero    | 33.112.345      | 2644301234  | Mitre 2200 San Juan                   | 12/02/2025 | 14/02/2025 | 2          | 0        | \$ 45.000     | \$ 45.000 | \$ 0          |
| Carla Ponce      | 34.789.123      | 2619876543  | Mendoza 1325, San Juan                | 10/02/2025 | 12/02/2025 | 2          | 1        | \$ 52.500     | \$ 25.000 | \$ 27.500     |
| Valeria Torres   | 40.987.654      | 3814567890  | B° Villa Lourdes Rawson               | 07/02/2025 | 10/02/2025 | 1          | 2        | \$ 60.000     | \$ 60.000 | \$ 0          |
| Nicolás Martínez | 35.234.567      | 2644123456  | Av. Libertador 1015 San Juan          | 02/02/2025 | 05/02/2025 | 2          | 1        | \$ 77.500     | \$ 40.000 | \$ 37.500     |
| Facundo Herrera  | 41.223.456      | 2644321987  | Calle 5, Pocito                       | 20/02/2025 | 22/02/2025 | 3          | 0        | \$ 67.500     | \$ 30.000 | \$ 37.500     |
| Laura Fernández  | 38.765.432      | 3517894561  | Calle Tucumán 890 SJ                  | 01/02/2025 | 03/02/2025 | 2          | 2        | \$ 90.000     | \$ 50.000 | \$ 40.000     |
| Brenda Quiroga   | 37.654.321      | 2644765432  | Lateral Norte Circunv. 1450 Rivadavia | 04/02/2025 | 07/02/2025 | 2          | 2        | \$ 90.000     | \$ 50.000 | \$ 40.000     |
| Romina Villalba  | 36.789.321      | 2644761890  | B° UDAP III Chimbas                   | 09/02/2025 | 11/02/2025 | 1          | 2        | \$ 60.000     | \$ 20.000 | \$ 40.000     |
| Hugo Suárez      | 32.000.111      | 2644954321  | Calle Lemos 760 Sta. Lucía            | 23/02/2025 | 25/02/2025 | 2          | 2        | \$ 90.000     | \$ 90.000 | \$ 0          |
| Esteban Ledesma  | 31.556.789      | 2644908765  | Av. Rioja 1120 San Juan               | 18/02/2025 | 21/02/2025 | 2          | 1        | \$ 77.500     | \$ 35.000 | \$ 42.500     |

Este enfoque manual carece de protocolos formales de organización, control y seguridad, lo que genera una serie de riesgos. En primer lugar, la información sensible de los clientes, como sus datos personales y registros de pagos, no se encuentra protegida adecuadamente, ya que el archivo es accesible para cualquier persona con acceso al dispositivo o carpeta donde se encuentra almacenado. Esto representa una vulnerabilidad significativa, tanto en términos de privacidad de los datos como de seguridad financiera.

Además, la ausencia de restricciones de edición en la planilla permite que cualquier usuario pueda modificar o eliminar información accidental o intencionalmente, lo cual aumenta la probabilidad de errores en los cobros o en el control de ingresos. La falta de trazabilidad impide identificar fácilmente quién realizó una modificación y en qué momento, lo que entorpece la posibilidad de auditar los movimientos financieros del camping.

El Camping Arroyito carece de un sistema integral y eficiente para la gestión contable. Los registros de ingresos se encuentran fragmentados en distintos archivos de Excel, lo cual dificulta una visión consolidada de la situación financiera del establecimiento. En particular, los ingresos provenientes de las reservas de cabañas se gestionan en una planilla independiente, mientras que los ingresos generados de parcelas para carpas se registran en un archivo distinto. Esta separación obliga a realizar comparaciones y análisis de forma manual, lo que aumenta la carga administrativa y genera margen para errores o duplicación de datos.

En lo que respecta a los gastos operativos, estos son registrados en un tercer archivo Excel, administrado exclusivamente por los dueños del camping. En dicho documento se

anotan todos los egresos históricos del camping, incluyendo su fecha, detalle y monto, lo que permite realizar filtrados por período y obtener el total de gastos mensuales mediante fórmulas automatizadas. Si bien esta herramienta ofrece cierta capacidad de control, sigue siendo una solución limitada y expuesta a riesgos



También las tareas y los horarios de trabajo de los empleados se realizan de forma manual de la siguiente manera: en el espacio de descanso de los empleados hay un pizarrón donde se anotan avisos o las tareas “extra” del día junto con sus horarios de realización. Los empleados por lo general saben cuales son sus tareas asignadas, cada uno tiene un rol que cumplir pero en caso de que surja algún cambio el encargado del camping suele informar sus tareas al empezar el día o es frecuente que se contacte a los empleados fuera de sus horarios de trabajo para comunicarles de tareas extra.

Hasta el día de hoy no hay una forma para controlar los empleados del camping, si bien por el momento no se cree necesario llevar este control debido a que se cuenta con solo 4 empleados (2 administradores, un personal de cocina y otro personal de limpieza), podemos considerar que a futuro, si surge la necesidad de contratar a más personas puede ser factible llevar este seguimiento.

**Análisis de Campo**

El análisis de campo es un paso fundamental en el desarrollo de proyectos que buscan ofrecer soluciones efectivas y alineadas con las necesidades reales de los usuarios o del entorno en el que se implementarán. Este proceso permite recopilar información directa y contextualizada, proporcionando una base sólida para diseñar estrategias, tomar decisiones informadas y garantizar que los resultados finales sean funcionales y

satisfactorios para todas las partes involucradas.

A continuación, se detallará cómo se llevó a cabo el análisis de campo para este proyecto, describiendo las metodologías empleadas, los instrumentos utilizados y los objetivos que guiaron esta investigación.

### Entrevistas y encuestas:

Se comenzó por realizar entrevistas y encuestas con los diversos actores que tienen lugar en el día a día del camping los cuales son los empleados, administradores y por supuesto los clientes.

En primer lugar se entrevistó a uno de los administradores del camping para conocer más a profundidad cuáles es su flujo de trabajo habitual, los problemas más comunes que surgen en el camping y qué cosas considera que se deberían mejorar:

*Entrevista con Marcos Gho, Administrador del Camping Arroyito*

**Entrevistador:** *Para comenzar, ¿podrías describirme cómo es el flujo de trabajo habitual en el camping? ¿Qué actividades realizas diariamente?*

**Marcos (Administrador):** *Claro, nuestras actividades diarias giran principalmente en torno a la atención al cliente y el mantenimiento del camping. Por la mañana, revisamos las reservas, asignamos las parcelas o cabañas a los huéspedes que llegan ese día, y actualizamos manualmente un registro que llevamos en papel. Luego, coordinamos las tareas de limpieza y mantenimiento, como el estado de las cabañas, áreas comunes y el suministro de agua o electricidad. También atendemos consultas o problemas de los huéspedes durante el día, y al final del día hacemos un balance de ingresos y egresos.*

**Entrevistador:** *¿Qué herramientas o sistemas utilizas para gestionar estas tareas?*

**Marcos (Administrador):** *Todo lo hacemos de forma manual. Usamos una agenda y un calendario para registrar las reservas, y los pagos se anotan en una planilla de Excel básica. No tenemos un sistema digitalizado, así que mucha de la información se guarda en físico o en archivos dispersos en las computadoras.*

**Entrevistador:** *¿Cómo se realizan las reservas de cabañas?*



**Marcos (Administrador):** Tenemos 3 canales para que los clientes puedan comunicarse para realizar reservas, el más usado es Whatsapp, los clientes realizan consultas de disponibilidad y servicios a través de ese medio y nosotros les contestamos con las opciones disponibles para la fecha en la que quieren venir. La otra opción es la llamada telefónica y también mediante chat de Facebook, pero a esta última no solemos prestarle tanta atención.

**Entrevistador:** ¿Cuáles son los problemas más comunes que suelen surgir en el camping debido a este método de gestión?

**Marcos (Administrador):** Lo principal son los errores en las reservas. A veces tenemos malentendidos sobre la disponibilidad de cabañas o parcelas, lo que puede generar conflictos con los huéspedes. También es un desafío mantener el control de las tareas de mantenimiento; si no se registran adecuadamente, puede pasar que algo importante quede sin atender. Otro problema es la demora en responder a consultas de los clientes, porque tenemos que buscar la información manualmente.

**Entrevistador:** Desde tu punto de vista, ¿qué cosas crees que se podrían mejorar para optimizar el funcionamiento del camping?

**Marcos (Administrador):** Definitivamente, tener un sistema que centralice toda la información en un solo lugar sería de gran ayuda. Algo que nos permita ver las reservas, el estado de las cabañas y las tareas pendientes en tiempo real. También sería ideal ofrecer a los clientes la opción de reservar online, para reducir la carga de trabajo en recepción y mejorar la experiencia de los huéspedes.

**Entrevistador:** ¿Cómo describirías la experiencia actual de los huéspedes en el camping? ¿Qué opinan ellos del proceso de reserva y la atención que reciben?

**Marcos (Administrador):** Creo que la mayoría de los huéspedes están contentos, pero recibimos quejas ocasionales sobre la falta de claridad en las reservas o demoras en la atención.

**Entrevistador:** Muchas gracias por tu tiempo

Luego se encuestó a los 4 empleados del camping en donde se les pidió que

respondan de forma anónima a las siguientes preguntas:

1. *¿Consideras que sería beneficioso automatizar algunas tareas del camping, como por ejemplo, la gestión de las reservas?*

Los 4 empleados respondieron que sí

2. *¿Te resulta difícil mantener un registro actualizado de las reservas y asignaciones manualmente?*

2 de los empleados respondieron “A menudo” y 2 respondieron “Ocasionalmente”

3. *¿Cuánto tiempo consideras que se pierde diariamente debido a procesos manuales?*

Los 4 empleados respondieron “entre 1 y 2 horas”

4. *¿Sientes que hay problemas frecuentes de comunicación y/o coordinación cuando se trata de la gestión del Camping?*

Los 4 empleados respondieron “A veces”

5. *¿Qué tan importante crees que sería poder acceder a la información del estado de las cabañas o parcelas en tiempo real?*

Los 4 empleados respondieron “Muy importante”

6. *¿Cuánto impacto crees que tendría una plataforma digital en la experiencia de los huéspedes (por ejemplo, reservas en línea, consultas rápidas)?*

2 empleados respondieron “Muy Positivo” y otros 2 “Algo Positivo”

7. *¿Qué opinas sobre la carga de trabajo actual?*

3 empleados respondieron que “es manejable” y 1 que es “Un poco alta”

8. *En tu opinión, ¿cuál es el principal desafío que enfrentan los empleados para realizar su trabajo eficientemente?*

- *Empleado 1: Considero que lo más difícil es la cantidad de llamados que se reciben en época de vacaciones, donde hay más concurrencia en el camping, es difícil atender llamados, responder mensajes y atender a los clientes todo al mismo tiempo.*
- *Empleado 2: Lo mas complicado me parece que es saber cuando una cabaña acaba de desocuparse, generalmente dejamos anotado las cabañas que hay que limpiar pero se han dado situaciones en donde eso no se dejó anotado y hay que demorar el ingreso de un nuevo huésped porque se está limpiando la cabaña*



- *Empleado 3: Los llamados y mensajes de whatsapp, me encontré con situaciones en donde los clientes suelen maltratarnos por demorar mucho en contestar*
- *Empleado 4: El registro de parcelas para las carpas, a veces suele pasar que asignamos una parcela que ya se encuentra ocupada por otra familia.*

#### Recopilación de opiniones de los clientes:

La decisión de realizar una búsqueda online a través de "Google Reviews" para recopilar opiniones de los usuarios que visitaron y se hospedaron en el Camping Arroyito se basó en la necesidad de obtener una perspectiva directa y honesta de quienes experimentaron los servicios del camping. Las reseñas online son una herramienta poderosa para evaluar la satisfacción del cliente, identificar fortalezas y detectar áreas de mejora desde el punto de vista de los usuarios finales.

En el ámbito del turismo y la hospitalidad, las experiencias y percepciones de los clientes tienen un impacto significativo en la reputación y el éxito de un negocio. Las opiniones reflejan no solo el nivel de satisfacción general, sino también aspectos específicos que los usuarios consideran positivos o negativos. Este tipo de retroalimentación es invaluable, ya que: Identifica qué elementos del servicio o infraestructura generan valor para los clientes y deben ser mantenidos o incluso potenciados. También, permite conocer problemas recurrentes o aspectos que generan insatisfacción, ofreciendo puntos de partida claros para la optimización de procesos. Al recopilar y analizar reseñas reales, se asegura que las decisiones de mejora estén alineadas con las expectativas y necesidades reales de los clientes.

En el caso del Camping Arroyito, las opiniones reflejan la experiencia directa de quienes interactuaron con el personal, utilizaron las instalaciones y participaron en los procesos operativos, como las reservas o la entrega de cabañas.

A partir del análisis de las reseñas, se identificaron dos tipos principales de comentarios:

- Comentarios Positivos:
  - La mayoría de los usuarios resaltó la comodidad de las instalaciones y la calidad de la estadía. Esto sugiere que aspectos como el entorno natural,

la infraestructura y la hospitalidad en general son bien valorados por los huéspedes.

- Estos puntos positivos representan una fortaleza que diferencia al camping y contribuye a una experiencia placentera para los visitantes.
- Comentarios Negativos:
  - Se reportaron problemas relacionados con la desorganización del personal, particularmente en la atención al cliente.
  - Algunas reseñas mencionaron falta de atención en detalles operativos, como retrasos en la entrega de cabañas o falta de limpieza adecuada antes del ingreso de nuevos huéspedes.
  - Hubo quejas sobre información errónea al realizar reservas, lo que generó frustración y problemas logísticos para los clientes.

Local Guide · 66 opiniones · 204 fotos

★★★★★ Hace 4 años

Nos dieron info por Facebook que fue errónea al llegar al lugar

Local Guide · 21 opiniones

★★★★★ Hace 4 años

La atencion no es buena.. La proligidad del lugar esta ausente..

Local Guide · 233 opiniones · 3,973 fotos

★★★★★ Hace 5 años

El lugar no posee buena atención. Muy básico en temporada de marzo por ejemplo. No es recomendable. Precio alto de alquiler con respecto a lo q nos ofreció. Tuvimos q pedir q nos trajera sábanas y toallas xq no sé fijo antes de entregar las llaves de nuestra cabaña q estaba la ropa de cama usada de los anteriores inquilinos. Mucho para mejorar.

Los comentarios positivos indican que el Camping Arroyito ha logrado posicionarse como un destino atractivo para los viajeros, ofreciendo una estadía cómoda y satisfactoria en términos generales. Esto demuestra que los elementos esenciales del camping son sólidos y valorados por los usuarios, un aspecto que se debe mantener y promover.

Sin embargo, los comentarios negativos ponen en evidencia problemas específicos en la

gestión operativa y la comunicación con los clientes. Estas fallas reflejan la necesidad de:

1. Implementar procesos más eficientes: Automatizar tareas como la gestión de reservas y la asignación de cabañas podría reducir errores y mejorar la experiencia del cliente.
2. Fortalecer la comunicación interna: Una mejor coordinación entre el personal permitiría evitar problemas relacionados con la desorganización y la atención al detalle.
3. Mejorar la precisión en la información: Capacitar al personal en el uso de herramientas de gestión modernas podría garantizar que los clientes reciban información confiable y oportuna.

El análisis de las opiniones en "Google Reviews" no sólo proporcionó una visión clara de las fortalezas y debilidades del Camping Arroyito, sino que también sirvió como base para priorizar las áreas de mejora. La combinación de comentarios positivos y negativos subraya la importancia de mantener los aspectos valorados por los clientes, al mismo tiempo que se trabaja en optimizar los procesos internos para ofrecer un servicio más profesional y eficiente. Esta retroalimentación refuerza la necesidad de implementar un sistema de gestión moderno que permita resolver los problemas detectados y mejorar aún más la experiencia del cliente.

### Observación directa

Con el objetivo de comprender a fondo la dinámica operativa y los desafíos a los que se enfrentan los trabajadores del Camping Arroyito, se realizó una visita de observación directa durante el mes de febrero, un período caracterizado por alta concurrencia debido a la temporada de vacaciones. Esta experiencia permitió evaluar la carga laboral de los empleados en tiempo real, así como identificar los puntos críticos en la operación diaria del camping.

Cargas laborales identificadas:

1. Gestión de Reservas y Consultas:  
Se pudo observar que la recepción es un punto clave en la operación del camping. Durante las horas pico, los trabajadores se veían desbordados por la

cantidad de llamadas telefónicas, mensajes de WhatsApp y consultas presenciales que debían atender simultáneamente.

- Desafío principal: La falta de un sistema que centralice las reservas y la información sobre la disponibilidad de cabañas y parcelas generaba tiempos de espera prolongados para los clientes y estrés en los empleados.
- Impacto: Esto incrementa el riesgo de cometer errores en las asignaciones o de proporcionar información incorrecta a los huéspedes.

## 2. Control de Disponibilidad y Limpieza de Cabañas:

Se notaron problemas recurrentes en la coordinación para determinar qué cabañas estaban listas para ser ocupadas. En algunos casos, la información sobre la desocupación no se registraba a tiempo, lo que obligaba a los huéspedes a esperar mientras el personal limpiaba y preparaba las instalaciones.

- Desafío principal: La dependencia de un registro manual o verbal para monitorear estas tareas generaba demoras y confusión.
- Impacto: La experiencia del cliente se veía afectada negativamente, y el personal enfrentaba presión adicional para agilizar los procesos.

## 3. Atención al Cliente Presencial:

Durante el check-in y el check-out, los empleados debían realizar múltiples tareas simultáneamente, como registrar datos, cobrar tarifas, brindar información sobre las instalaciones y resolver dudas. En algunos casos, esto ocasiona filas y clientes impacientes.

- Desafío principal: La falta de un flujo de trabajo optimizado y herramientas digitales para automatizar ciertos procesos administrativos.
- Impacto: La sobrecarga de tareas limitaba la capacidad del personal para ofrecer una atención más personalizada.

## 4. Capacidades Tecnológicas Limitadas

Aunque algunos empleados utilizaban herramientas básicas como Excel para registrar información, estas soluciones no eran suficientes para gestionar la carga de trabajo durante la temporada alta. Además, no todos los trabajadores estaban familiarizados con estas herramientas.

## 5. Altas Expectativas de los Clientes

Durante la observación, fue evidente que los clientes tenían expectativas de un

servicio más rápido y organizado, especialmente aquellos que estaban acostumbrados a realizar reservas y consultas en línea en otros destinos turísticos.

La observación directa permitió confirmar que el flujo de trabajo actual, basado en procesos manuales y comunicación informal, no es sostenible durante los períodos de alta demanda. Los desafíos identificados no solo generan estrés para los empleados, sino que también impactan directamente en la experiencia de los clientes.

### **Opciones Similares en el Mercado:**

La propuesta de implementar un sistema de gestión en Camping Arroyito surgió tras observar el uso de un sistema llamado “Núcleo IT Gastronómico” en un restaurante, el cual gestiona pedidos, cobros y recursos humanos de manera eficiente. Este sistema no solo facilitaba el trabajo de los empleados y era práctico de usar, sino que también inspiraba confianza en los clientes, ya que elimina el margen de error humano en tareas sensibles como el cálculo de precios y cobros.

En la actualidad, la digitalización se ha convertido en una necesidad transversal para empresas de todo tipo. Implementar soluciones tecnológicas en comercios ha demostrado ser clave a la hora de ofrecer un mejor servicio y atención, además de ser sumamente esencial para poder mantenerse competitivo en el entorno empresarial actual. Los sistemas de gestión se encuentran implementados en cualquier tipo de sectores tales como la gastronomía, hotelería, retail, logística, entre otros, lo cual demuestra la gran evolución que han atravesado estas tecnologías, que han logrado adaptarse a cada rubro para ayudar a facilitar el manejo de los negocios, optimizar tiempos, reducir costos y sobre todo mejorar la calidad del trabajo.

Teniendo esto en cuenta, es de esperarse que existan soluciones tecnológicas que tengan como propósito y finalidad la administración de entidades turísticas tales como los campings. Uno de ellos es Lodgify, uno de los sistemas de automatización de alquileres vacacionales más populares, permite gestionar reservas y empleados, además de ofrecer la creación de páginas web personalizadas. Sin embargo, su enfoque es generalista y carece de un control detallado para entidades reservables, como cabañas o parcelas específicas.

Otro nombre muy reconocido en el área es el de Campeak, que ofrece características similares a las de Lodgify e incluye un mapa interactivo para gestionar las entidades reservables. No obstante, carece de la opción para crear una página web personalizada, lo que limita a los propietarios que buscan una integración total con su marca.

También podemos incluir otras plataformas tales como Mews, que es ampliamente usado en hoteles, se enfoca en optimizar check-ins y pagos en línea. No obstante, su complejidad y costos pueden ser excesivos para campings pequeños. Cloudbeds es otra opción que ofrece soluciones de gestión integradas, pero su interfaz no siempre resulta intuitiva para operadores sin experiencia técnica.

Estas herramientas muestran el impacto positivo de la tecnología, pero también dejan claro que no todas son adecuadas para resolver problemas específicos de empresas pequeñas o familiares. Aquí es donde el desarrollo de software a medida se posiciona como una alternativa viable. Según expertos, un software personalizado permite:

- Adaptarse completamente a los procesos de la empresa.
- Evolucionar junto al negocio sin depender de actualizaciones estándar que no siempre se alinean con las necesidades del cliente.
- Incrementar la seguridad y eliminar dependencias de licencias externas

Un software a medida permite evitar funcionalidades innecesarias que no aporten valor, optimizando así los costos de implementación y mantenimiento. Además, su diseño puede centrarse en ofrecer una interfaz intuitiva y fácil de usar, lo que garantiza que cualquier empleado, independientemente de su nivel de experiencia tecnológica, pueda manejarlo sin complicaciones. Esta solución, al ser personalizada, brinda flexibilidad, eficiencia y una base sólida para que el negocio siga creciendo y destacándose en un mercado competitivo.

## **Tecnologías Investigadas**

Para resolver las problemáticas a las que se enfrenta el Camping Arroyito, se consideraron varias tecnologías y herramientas ampliamente utilizadas en proyectos de desarrollo web. A continuación, se describen y analizan las principales opciones disponibles, evaluando sus ventajas y desventajas en función de las necesidades del proyecto.

### LAMP Stack (Linux, Apache, MySQL, PHP)

El LAMP stack es una de las tecnologías más tradicionales para el desarrollo web. Consiste en Linux como sistema operativo, Apache como servidor web, MySQL como base de datos relacional y PHP como lenguaje de programación.

#### Ventajas:

- Amplia comunidad de soporte y documentación disponible.
- Es eficaz para aplicaciones basadas en contenido, como blogs o sistemas CMS.
- Compatible con servidores web económicos.

#### Desventajas:

- Menor flexibilidad para el desarrollo de aplicaciones modernas, altamente interactivas.
- La integración con tecnologías modernas (como SPAs o aplicaciones móviles) es más compleja.
- PHP, aunque robusto, puede resultar más limitado en comparación con tecnologías más recientes.

### Django con Python y PostgreSQL

Django es un framework de alto nivel para el desarrollo rápido de aplicaciones web en Python. PostgreSQL se utiliza como la base de datos relacional.

#### Ventajas

1. Lenguaje versátil y ampliamente utilizado: Python es un lenguaje de programación multipropósito que se usa en diversas áreas, como desarrollo web, análisis de datos, inteligencia artificial, y aprendizaje automático. Esta versatilidad asegura que cualquier funcionalidad adicional o integración futura pueda ser desarrollada fácilmente dentro del mismo ecosistema.
2. Características avanzadas incorporadas: Django contiene módulos y herramientas para gestionar autenticación de usuarios, administración de contenido, generación de formularios, validación de datos, y mucho más. Estas

herramientas permiten un desarrollo más rápido y reducen la necesidad de configurar manualmente funcionalidades comunes.

3. Ideal para manejo complejo de datos: Gracias a su ORM (Object-Relational Mapping), Django permite interactuar con bases de datos de manera eficiente y estructurada.
4. Comunidad activa y documentación extensa:  
Django cuenta con una comunidad global que contribuye con plugins, paquetes adicionales y soporte técnico. Su documentación oficial es exhaustiva y amigable para desarrolladores de todos los niveles.

## Desventajas

1. Curva de aprendizaje más pronunciada: Aunque Python es considerado uno de los lenguajes más fáciles de aprender, Django puede ser desafiante para desarrolladores novatos debido a su enfoque en convenciones y estructura rígida. La configuración inicial del framework y su modelo de desarrollo basado en ORM pueden requerir un tiempo considerable de aprendizaje.
2. No ideal para aplicaciones web altamente interactivas:  
Django es excelente para renderizar páginas web completas en el servidor (SSR, Server-Side Rendering), pero no es tan eficiente para crear aplicaciones web de una sola página (SPA, Single-Page Applications) o con interactividad avanzada. Para lograr esto, es necesario integrar frameworks frontend modernos como React o Angular, lo que puede aumentar la complejidad del desarrollo.
3. Mayor consumo de recursos:  
Django, por ser un framework completo, puede consumir más recursos que frameworks más ligeros, lo que podría ser un inconveniente para aplicaciones pequeñas con presupuesto limitado en infraestructura.
4. Menos enfoque en el frontend:  
Django no incluye herramientas específicas para el diseño de interfaces de usuario modernas y dinámicas, lo que obliga a los desarrolladores a buscar soluciones externas para este aspecto.
5. Ecosistema menos enfocado en tiempo real:  
Aunque Django tiene soporte para websockets y aplicaciones en tiempo real, como chats o actualizaciones instantáneas, estas características no están



integradas de manera nativa y requieren librerías adicionales, como Django Channels, lo que complica el desarrollo en este tipo de proyectos.

### Ruby on Rails (RoR)

Ruby on Rails es un framework de desarrollo web que opera sobre el lenguaje de programación Ruby. Diseñado con el principio de "Convención sobre Configuración" (Convention over Configuration) y el paradigma DRY ("Don't Repeat Yourself"), facilita a los desarrolladores la creación de aplicaciones web con un enfoque en productividad y simplicidad. Desde su lanzamiento, ha sido popular en startups y proyectos que buscan un desarrollo ágil.

### Ventajas

1. Desarrollo rápido y simplificado:

Ruby on Rails permite construir aplicaciones rápidamente gracias a su estructura basada en convenciones predefinidas. Esto elimina la necesidad de configurar manualmente aspectos comunes del desarrollo, como rutas, controladores y vistas.

2. Ideal para prototipos y MVPs:

RoR es perfecto para startups que desean validar ideas rápidamente. Se utiliza mayormente en las fases iniciales de un proyecto para construir sus primeros prototipos.

3. Amplia biblioteca de terceros

Las bibliotecas de terceros, es decir, bibliotecas públicas permiten agregar funcionalidades de manera rápida y sencilla, como autenticación, administración de archivos o integración con APIs externas. Esto reduce el tiempo necesario para implementar características complejas.

### Desventajas

1. Menor popularidad en comparación con otras opciones:

Últimamente, Ruby on Rails ha sido reemplazado por otros frameworks más populares como Django y Node.js debido a su menor adaptabilidad en sistemas modernos. Según el informe Stack Overflow Developer Survey 2023, Ruby representa solo el 6.6% del uso de lenguajes de programación, comparado con

Python (49.1%).

2. Problemas de escalabilidad:

A medida que las aplicaciones crecen en complejidad y volumen de usuarios, Ruby on Rails puede enfrentar desafíos de escalabilidad debido a su naturaleza monolítica. Aunque existen herramientas y técnicas para mitigar estos problemas, como el uso de microservicios, el esfuerzo requerido puede ser mayor que en otras tecnologías como Node.js o Golang.

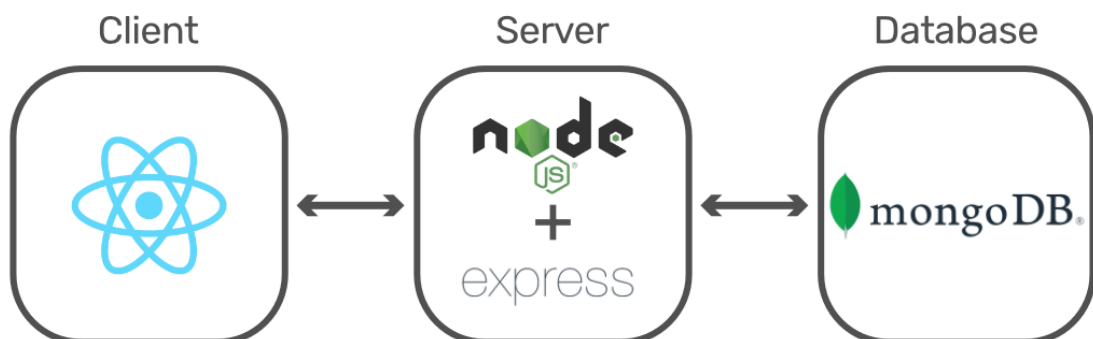
3. Desempeño:

En términos de rendimiento, Ruby on Rails es generalmente más lento que frameworks basados en lenguajes como JavaScript (Node.js) o Golang, especialmente en aplicaciones con alta demanda en tiempo real.

4. MERN Stack (MongoDB, Express.js, React.js, Node.js)

El MERN Stack es una tecnología full-stack moderna diseñada para el desarrollo de aplicaciones web dinámicas y altamente interactivas. Se compone de MongoDB (base de datos NoSQL), Express.js (framework backend), React.js (biblioteca frontend) y Node.js (entorno de ejecución backend). Este conjunto permite construir aplicaciones utilizando un único lenguaje, JavaScript, tanto en el cliente como en el servidor, lo que simplifica el desarrollo.

# The MERN stack



## Ventajas

### 1. Desarrollo Full-Stack en JavaScript

Utilizar JavaScript en todo el stack elimina la necesidad de aprender múltiples lenguajes para diferentes partes del proyecto. Esto facilita el desarrollo del proyecto y reduce la curva de aprendizaje.

### 2. Interfaces de usuario dinámicas y rápidas

React.js permite construir interfaces de usuario rápidas, modulares y altamente interactivas gracias a su enfoque basado en componentes y su Virtual DOM.

### 3. Escalabilidad y rendimiento

Node.js es ideal para aplicaciones en tiempo real, como chats o transmisiones en directo. Su arquitectura basada en eventos lo hace eficiente para manejar múltiples solicitudes simultáneamente.

### 4. Base de datos flexible y escalable

MongoDB, una base de datos NoSQL, permite manejar datos no estructurados o semi-estructurados, como JSON. Esto es útil para sistemas que pueden necesitar almacenar datos heterogéneos o evolucionar con nuevas funcionalidades. Es muy utilizado en el desarrollo de sistemas que buscan flexibilidad y que crecerán con el paso del tiempo.

### 5. Amplia comunidad y recursos

El MERN stack cuenta con una comunidad sumamente activa, dispuesta a resolver dudas y compartir conocimientos a través de foros. También hay una abundante cantidad de documentación y una gran variedad de tutoriales, lo que facilita a personas sin experiencia aprender más sobre MERN, como implementarlo y también ayuda a la solución de problemas y la implementación de mejores prácticas.

### 6. Facilidad para creaciones de Aplicaciones Móviles

Usar React Native permite compartir lógica con React.js, facilitando la creación de aplicaciones móviles con menor esfuerzo.

## Desventajas

### 1. Consultas complejas

MongoDB, al ser una base de datos NoSQL, no es ideal para proyectos que

requieren transacciones complejas o consultas SQL avanzadas.

## 2. Curva de aprendizaje para integrar componentes

Aunque cada tecnología en el stack es robusta por sí misma, integrarlas correctamente puede ser un desafío para desarrolladores sin experiencia previa.

## 3. Requerimientos de Hosting y Escalabilidad Manual

Mientras que soluciones como Django o Ruby on Rails ofrecen configuraciones más estandarizadas, en el MERN stack, los desarrolladores deben manejar manualmente aspectos como el despliegue, la escalabilidad y la optimización de rendimiento en el servidor.

## 5. MEAN Stack (MongoDB, Express.js, Angular, Node.js)

El MEAN Stack es similar al MERN Stack, pero reemplaza React con Angular como la tecnología frontend, es decir que se ve compuesto por una base de datos NoSQL MongoDB, el framework web minimalista para Node.js que es Express.js, Angular como framework frontend y Node.js como entorno de ejecución para JavaScript. Angular es un framework desarrollado por Google que ofrece una solución completa para crear aplicaciones web.

Ventajas:

- **Arquitectura completa:** Angular ofrece una solución integral con herramientas integradas como enrutamiento, inyección de dependencias y gestión de formularios, lo que facilita la construcción de aplicaciones escalables y mantenibles.
- **TypeScript:** Uso de TypeScript: Angular está construido sobre TypeScript, un superconjunto de JavaScript que introduce tipado estático y otras características avanzadas, mejorando la calidad del código y facilitando la detección temprana de errores.
- **Comunicación fácil con la API:** Angular facilita la integración con el backend a través de servicios como HttpClient, que facilitan la realización de solicitudes HTTP y el manejo de respuestas de manera eficiente

## Desventajas:

- Curva de aprendizaje pronunciada: Debido a su complejidad y a la cantidad de conceptos que maneja (como módulos, componentes, servicios y directivas), Angular puede resultar desafiante para desarrolladores novatos.
- Mayor consumo de recursos: Las aplicaciones desarrolladas con Angular tienden a ser más pesadas en términos de tamaño y pueden requerir más recursos, lo que podría afectar el rendimiento en comparación con soluciones más ligeras.

## Justificación de la elección del MERN Stack

Después de analizar todas las opciones mencionadas, se terminó optando por utilizar la tecnología de MERN stack por las siguientes razones:

1. Familiaridad: Al contrario de los otros lenguajes de programación requeridos para implementar los otros enfoques, yo ya contaba con conocimientos intermedios de programación en JavaScript y también había utilizado Node.js en otros proyectos por lo cual se intuyó que sería más sencillo y rapido familiarizarme con MERN stack en vez de tener que aprender en nuevo lenguaje y framework desde cero. También había trabajado con la base de datos MongoDB en ocasiones anteriores así que eso también era un punto a favor para MERN.
2. Modernidad: MERN está diseñada principalmente para crear aplicaciones modernas, tales como sistemas de gestión, que requieren una interfaz interactiva, atractiva y actualizaciones en tiempo real.
3. Escalabilidad: MongoDB permite una fácil adaptación a nuevos requerimientos sin necesidad de remodelar la base de datos. Esto es esencial para poder ofrecer una solución al Camping Arroyito que no se quede estancada en el tiempo, si no que podrá crecer a la par del camping al permitir que se puedan sumar nuevas funcionalidades sin poner en riesgo la integridad de las funcionalidades ya existentes.
4. Abundante documentación e información: Mientras investigaba esta tecnología, encontré una gran cantidad de documentación y tutoriales que pensé resultan de gran ayuda a la hora de crear una solución para el Camping Arroyito. Además, hay muchos foros y proyectos en Github que resultan útiles a la hora de buscar ejemplos, consejos y conseguir información.

5. Compatibilidad con móviles: MERN puede integrarse fácilmente con React Native, permitiendo una expansión futura hacia aplicaciones móviles. Este punto resulta clave ya que sería bueno tener la opción disponible de poder expandir la solución de Camping Arroyito a teléfonos móviles de ser así necesario.
6. Eficiencia en el desarrollo: React y Node.js, en conjunto, permiten un flujo de trabajo rápido y eficiente, ideal para entregar resultados en tiempos razonables.

El MERN stack no solo satisface las necesidades actuales del Camping Arroyito, sino que también proporciona una base sólida para futuras expansiones y adaptaciones. Se trata de una tecnología que permitirá realizar un trabajo de forma eficiente y que permitirá el desarrollo de funcionalidades adicionales sin necesidad de cambiar la arquitectura base, lo que facilita la evolución del sistema a medida que crecen las demandas del negocio. La naturaleza escalable de Node.js y MongoDB asegura que el sistema pueda manejar un aumento en la cantidad de usuarios, reservas o datos sin comprometer el rendimiento. React.js, por su parte, permite la creación de interfaces modernas y dinámicas que pueden adaptarse a nuevas tendencias de diseño y usabilidad, garantizando una experiencia de usuario atractiva y funcional.

En términos operativos, el MERN stack ofrece eficiencia no solo en el desarrollo inicial, sino también en el mantenimiento continuo del sistema. Esto lo convierte en una tecnología no solo adecuada para las necesidades inmediatas del Camping Arroyito, sino también en una inversión tecnológica a largo plazo, preparada para sostener y potenciar su crecimiento en un mercado competitivo.

# PROPUESTA DE SOLUCIÓN

Para abordar los desafíos operativos y de gestión identificados en Camping Arroyito, se propone una solución tecnológica integral basada en los principios y herramientas analizados en el marco teórico. Esta propuesta busca proporcionar una respuesta eficiente y personalizada a las necesidades específicas del negocio, utilizando las mejores prácticas y tecnologías disponibles en el mercado.

El objetivo principal de esta solución es automatizar y optimizar los procesos actuales del camping, mejorando tanto la experiencia de los empleados como la de los clientes. A través del desarrollo de un software a medida, la propuesta se centra en superar las limitaciones de los sistemas genéricos existentes, adaptándose a las particularidades del camping, su estructura organizativa y su visión de futuro.

Esta sección detalla cómo se estructurará y desarrollará el sistema, abordando aspectos técnicos, funcionales y operativos. También se explicará cómo esta solución contribuirá a resolver las problemáticas identificadas, generando valor tanto para los administradores como para los usuarios finales.

## **Alcance Funcional**

El alcance funcional del sistema define las capacidades y comportamientos que este deberá tener para satisfacer las necesidades específicas de los usuarios del Camping Arroyito. Estas necesidades fueron identificadas a través de entrevistas realizadas con los propietarios y empleados del camping, permitiendo recopilar información clave sobre las expectativas de los distintos roles involucrados. El objetivo es garantizar que el sistema propuesto no solo resuelva los problemas actuales, sino que también se convierta en una herramienta eficiente para la gestión y operación del camping.

Definir un alcance funcional claro es esencial para garantizar que el sistema cumpla con las expectativas de los usuarios y aporte valor al camping. Este enfoque permite priorizar las funcionalidades más importantes, identificar posibles limitaciones y establecer un marco sólido para el desarrollo. Además, asegura que cada aspecto del sistema esté alineado con los objetivos operativos y estratégicos del Camping Arroyito, transformándolo en una herramienta indispensable para su crecimiento y modernización.

Con este alcance funcional, el sistema propuesto no solo resolverá los problemas actuales, sino que también establecerá las bases para un modelo de gestión más eficiente, adaptado a las exigencias del mercado y a las necesidades de sus usuarios.

### Historias de usuario

Las historias de usuario son una herramienta clave en el desarrollo de sistemas, ya que permiten comprender las necesidades y expectativas de los usuarios finales de una manera directa y centrada en su experiencia. Estas historias nos permiten conocer los detalles y funcionalidades necesarios para el sistema desde el punto de vista de los diferentes roles que interactúan con él, qué funcionalidades necesitan, por qué las requieren y qué criterios deben cumplirse para que esas funcionalidades se consideren satisfactorias.

En el caso de este proyecto, se recopilaron 15 historias de usuario que abarcan las expectativas de los diferentes roles involucrados (administradores, empleados y clientes). Estas historias reflejan tanto las necesidades operativas actuales como las oportunidades de mejora que el sistema debe abordar. Entre las funcionalidades solicitadas, destacan aspectos clave como la gestión de reservas, la asignación de tareas y la mejora de la experiencia del cliente.

De las 15 historias recopiladas, se destacan las siguientes por su relevancia para el éxito del sistema:

- 1. Como usuario del sistema Creek, quiero que únicamente puedan ingresar al sistema usuarios administradores o empleados para resguardar información sensible que se maneja dentro del sistema que es crítica para el funcionamiento del camping:** Es fundamental que sólo los usuarios autorizados, como administradores y empleados del camping, puedan acceder al sistema, ya que este contiene información sensible tanto del negocio como de los clientes. Para garantizar la seguridad, se deben implementar medidas como autenticación mediante credenciales únicas, cifrado de datos y control de acceso basado en roles. Esto asegura que cada usuario solo acceda a las funcionalidades necesarias según sus responsabilidades, protegiendo la confidencialidad de la información y previniendo accesos no autorizados que



puedan comprometer la operación del camping.

| HISTORIA: LOGUEO DE USUARIOS AUTORIZADOS |                                 |   |  |
|--|---------------------------------|---|--|
| ID                                       | Descripción Historia de Usuario |   | Criterios de validación  |
| 7  | Como                            | usuario del sistema Creek   | 1. El usuario deberá ingresar su mail y contraseña en el Login del sistema   |
|  | Quiero                          | que unicamente puedan ingresar al sistema usuarios admin o empleados  | 2. Unicamente usuarios identificados como admin o employee podrán loguearse  |
|  | Para                            | resguardar la información sensible que se maneja dentro del sistema que es critica para el funcionamiento del camping | 3. Usuarios que quieran loguearse y no sean admin o employee visualizaran un mensaje de "no tienes los permisos necesarios"  |
|  |                                 |   | 4. El elemento login se asegurará que el hash de la contraseña ingresada corresponda con el hash de la contraseña guardada en la base de datos<br>5. Se mostrarán mensajes de error si el email ingresado no tiene los permisos necesarios o si la contraseña ingresada es erronea<br>6. El login no debe tardar más de 0.5 segundos |

2. Como usuario administrador del sistema Creek, quiero poder visualizar todas las reservas activas del Camping arroyito para saber la información de los clientes que llegan al camping, la fecha de llegada y salida, cuanto hay que cobrarles y en qué cabaña se hospedarán: Una de las principales razones para desarrollar este sistema es facilitar a los empleados del Camping Arroyito una mejor administración de las reservas. El sistema permitirá centralizar toda la información relacionada con las reservas activas, brindando acceso rápido y organizado a los datos, lo que reducirá errores y mejorará la eficiencia en la gestión. Esta funcionalidad será clave para optimizar el flujo de trabajo y ofrecer un mejor servicio a los clientes.

| HISTORIA: VISUALIZAR RESERVAS ACTIVAS |                                 |   |   |
|---------------------------------------|---------------------------------|---|---|
| ID                                    | Descripción Historia de Usuario |   | Criterios de validación   |
| 8                                     | Como                            | usuario admin del sistema Creek   | 1. En la sección reservas se podrá visualizar una lista de todas las reservas activas que fueron realizadas mediante la pagina web del Camping Arroyito o añadidas de forma manual por un empleado del camping  |
|                                       | Quiero                          | poder visualizar todas las reservas activas del Camping Arroyito  | 2. Cada elemento de la lista contará con el nombre de la cabaña reservada, nombre de la persona que realizó la reserva, fecha de check-in y check-out, cantidad de personas a hospedarse, precio a cobrar y estado de la reserva<br>3. El administrador tambien podrá hacer click en el botón "view" para conocer mas detalles de la reserva, en donde saldrán los datos personales del usuario que realizó la reserva y los datos de la cabaña reservada.<br>4. El administrador tambien podrá cancelar reservas.  |
|                                       | Para                            | saber la información de los clientes que llegaran al camping, la fecha en la que llegarán, cuando se iran, cuanto hay que cobrarles y en que cabaña se hospedarán | 1. Se tomarán los datos de las reservas guardados en la base de datos MongoDB y se las presentará en forma de lista<br>2. Cada reserva tendrá un numero de identificación unico para asegurar que no hay repeticiones<br>3. La lista de reservas activas debe mantenerse actualizada en tiempo real. Ante una nueva reserva o una cancelacion se deben reflejar los cambios inmediatamente.<br>4. El sistema debe garantizar la privacidad y seguridad de los datos de las reservas. Solo los usuarios autorizados pueden tener acceso a esta información |
|                                       |                                 |   |   |

3. Como usuario del sistema Creek, quiero visualizar las cabañas ocupadas por huéspedes que hay en el camping para conocer que cabañas están disponibles para ser alquiladas: El seguimiento del estado de las cabañas permitirá a los empleados del camping identificar cuáles están disponibles, cuáles necesitan mantenimiento y qué clientes están ocupando cada una. Esta funcionalidad no sólo agilizará la toma de decisiones, sino que también reducirá errores humanos, optimizando la gestión operativa y mejorando la experiencia tanto para los empleados como para los huéspedes.

| HISTORIA: VISUALIZAR ESTADO DE CABAÑAS |                                 |   |  |   |
|--|---------------------------------|---|--|---|
| ID                                     | Descripción Historia de Usuario |   | Criterios de Aceptación  | Criterios de validacion   |
| 9                                      | Como                            | usuario admin del sistema Creek                                     | 1. En la pestaña "Cabañas" se mostrará una lista de todas las cabañas del camping<br>2. Cada elemento de la lista contará con el nombre de la cabaña, su capacidad de ocupacion, su estado (ocupado, libre, en mantenimiento) y un comentario de ser necesario | 1. Se tomarán los datos de las cabañas guardados en la base de datos MongoDB y se las presentará en forma de lista  |
|  | Quiero                          | visualizar las cabañas ocupadas por huéspedes que hay en el camping | 3. El usuario podrá hacer click en el botón "view" para visualizar datos adicionales de la cabaña así como tambien las reservas anexadas a esa cabaña  | 2. Cada elemento cabaña tiene un atributo "estado" que puede variar entre "ocupado, libre o en mantenimiento"<br>3. El estado de las cabañas debe mantenerse actualizado en tiempo real. Ante una nueva reserva o una cancelacion se deben reflejar los cambios inmediatamente. |
|  | Para                            | conocer que cabañas estan disponibles para ser alquiladas           | 4. El usuario puede cambiar el estado de la cabaña a libre, ocupada o en mantenimiento<br>5. El usuario puede ver la transición de estados por los que paso una cabaña   | 4. En caso de que la cabaña este en estado "ocupado" se deben mostrar tambien los datos de la reserva   |
|  |                                 |   |  | 5. Solo los usuarios autorizados pueden tener acceso a esta información   |

4. Como usuario admin del Sistema Creek, quiero ingresar en el sistema que ingresó una nueva carpa al camping para saber qué parcelas están ocupadas y saber cuantas personas se encuentran hospedandose en carpas: El registro de nuevos clientes que se hospedan en carpas es una funcionalidad esencial, dado que este tipo de alojamiento es uno de los más solicitados en el camping. El sistema permitirá ingresar los datos pertinentes de cada cliente y asignarles una parcela específica, que quedará marcada como ocupada hasta que finalice su estadía. Esta herramienta garantizará un control eficiente y organizado de las parcelas disponibles, optimizando la experiencia tanto para los clientes como para el personal.

| HISTORIA: INGRESAR UNA NUEVA CARPA |                                 |   |   |   |
|------------------------------------|---------------------------------|---|---|---|
| ID                                 | Descripción Historia de Usuario |   | Criterios de Aceptación   | Criterios de validacion   |
| 10                                 | Como                            | usuario admin del sistema Creek   | 1. En la pestaña "Carpas" se podran visualizar una lista de usuarios que se encuentran hospedandose en carpas   | 1. Se tomarán los datos de los husespedes hospedados en carpas guardados en la base de datos MongoDB y se las presentará en forma de lista  |
|                                    | Quiero                          | ingresar en el sistema que ingresó una nueva carpa al camping                                   | 2. Cada elemento de la lista tendrá un id, el nombre de uno de los clientes, su DNI, celular, email, cantidad de husespedes, fecha de ingreso y fecha de egreso | 2. Cuando se quiera añadir un nuevo husesped se deberá abrir un formulario para que el administrador/empleadillo llene.<br>3. Los datos ingresados al formulario se deberán guardar en la base de datos MongoDB<br>4. El proceso de guardar los datos no debe demorar más de 0.3 segundos |
|                                    | Para                            | saber que parcelas estan ocupadas y saber cuantas personas se encuentran hospedandose en carpas | 3. El admin/employee podra agregar una nueva carpa llenando una plantilla con los datos requeridos.<br>4. Todos los datos de la plantilla son obligatorios.     | 4. La parcela asignada a la carpa debe marcarse como "ocupada"  |

5. Como usuario administrador del sistema Creek, quiero poder visualizar información de ingresos y egresos de dinero al camping para llevar un registro de los cobros que se efectuaron por cabañas/carpas/otros y de los gastos del camping: Contar con un registro detallado de los ingresos y egresos del camping es fundamental para mantener una gestión financiera clara y precisa. El sistema permitirá a los administradores visualizar la información contable de forma organizada, incluyendo las ganancias generadas por reservas, los gastos operativos y el balance general. Esta funcionalidad facilitará el análisis financiero del negocio, permitiendo tomar decisiones informadas

sobre inversiones, ajustes de precios y control de gastos. Además, al centralizar estos datos en un solo lugar, se mejora la transparencia y se optimiza el seguimiento del rendimiento económico del camping.

| HISTORIA: MOSTRAR INFORMACIÓN DE LOS INGRESOS/EGRESOS |                                 |   |  |  |
|---|---------------------------------|---|--|--|
| ID  | Descripción Historia de Usuario |   | Criterios de Aceptación  | Criterios de validación  |
| 12  | Como                            | usuario admin del sistema Creek   | 1. El usuario admin podrá ingresar a la pestaña "contabilidad" que mostrará una lista con el monto, monto, tipo (ingreso o egreso), estado (parcial, pagada o seña), fecha de realización y en caso de ser un cobro por estadia, de que cliente proviene.<br>2. El administrador podra visualizar diversos graficos para conocer como evoluciona economicamente el camping | 1. Se tomarán los datos de ingresos y egresos guardados en la base de datos MongoDB y se los presentará en forma de lista  |
|   | Quiero                          | poder visualizar información de los ingresos y egresos de dinero al camping                             | 3. Si el usuario hace click en el botón "info" podrá visualizar información extra del ingreso como por ejemplo, informacion sobre la reserva de la que proviene y los diferentes pagos que se hicieron para completar el monto total de la reserva.  | 2. Cuando se quiera añadir una nuevo registro de ingreso/egreso se deberá abrir un formulario para que el administrador llene.<br>3. Los datos ingresados al formulario se deberán guardar en la base de datos MongoDB |
|   | Para                            | llevar un registro de los cobros que se efectuaron por cabañas/carpas/otros y de los gastos del camping | 4. El administrador podra completar los pagos que esten en "parcial" haciendo click en el boton "completar pago"<br>5. El administrador tambien podrá eliminar un registro de ingreso/egreso o añadir uno nuevo.   | 4. Solo los usuarios autorizados pueden tener acceso a esta información  |
|   |                                 |   |  |  |

6. **Como usuario administrador del sistema Creek, quiero poder visualizar la información relevante de los empleados del camping para llevar un registro de los empleados que trabajan en el camping y facilitar la toma de decisiones:** El seguimiento de los empleados es un requisito clave para los administradores, ya que les permitirá tener un control más eficiente sobre quiénes trabajan en el camping, hace cuanto tiempo que son empleados del camping, cuál es su salario y otras métricas relevantes. Esta funcionalidad no solo facilitará la gestión administrativa, sino que también apoyará la toma de decisiones relacionadas con el personal de manera informada y organizada.

| HISTORIA: MOSTRAR INFORMACIÓN DE LOS EMPLEADOS |                                 |  |  |   |
|--|---------------------------------|--|--|---|
| ID   | Descripción Historia de Usuario |  | Criterios de Aceptación  | Criterios de validación   |
| 11   | Como                            | usuario admin del sistema Creek  | 1. El usuario admin podrá ingresar a la pestaña "empleados" mostrará una lista con el id, foto, nombre, apellido y DNI de cada empleado.   | 1. Se tomarán los datos de los empleados guardados en la base de datos MongoDB y se los presentará en forma de lista  |
|  | Quiero                          | poder visualizar información relevante de los empleados del camping                              | 2. Si el usuario hace click en el botón "info" podrá visualizar información extra del empleado en donde aparecan ademas de los datos previamente mencionados su cargo, fecha de nacimiento, email, dirección, fecha de ingreso y salario | 2. Cuando se quiera añadir un nuevo empleado se deberá abrir un formulario para que el administrador llene.<br>3. Los datos ingresados al formulario se deberán guardar en la base de datos MongoDB<br>4. El valor del atributo isEmployee se seteará a True  |
|  | Para                            | llevar un registro de los empleados que trabajan en el campina y facilitar la toma de decisiones | 3. El administrador será capaz de modificar cada uno de los datos del empleado<br>4. El administrador tambien podrá eliminar un empleado o añadir uno nuevo.   | 4. La información del empleado debe mantenerse actualizada en tiempo real. Ante una modificación o una eliminación se deben reflejar los cambios inmediatamente.<br>5. El sistema debe garantizar la privacidad y seguridad de los datos de los empleados. Solo los usuarios autorizados pueden tener acceso a esta información |

7. **Como usuario administrador del sistema Creek, quiero poder asignar tareas a los empleados del camping desde la plataforma, para mejorar la organización interna y evitar el uso de canales informales como WhatsApp donde puede perderse información o generar confusiones:** La asignación de tareas actualmente no se hace de la forma más eficiente y responsable posible,

para evitar canales de comunicación externos o formas que puedan causar la pérdida de información se decidió la creación de un calendario en el cual se podrán crear tareas que quedan asignadas a cada empleado, esto permitirá mejorar la organización dentro del camping y también que cada empleado conozca cuales son las tareas que deberá realizar durante un determinado día.

| HISTORIA: ASIGNACIÓN DE TAREAS |                                 |  |   |  |
|--------------------------------|---------------------------------|--|---|--|
| ID                             | Descripción Historia de Usuario |  | Criterios de Aceptación   | Criterios de validacion  |
| 13                             |                                 | usuario administrador del sistema Creek  | 1. El usuario admin podrá ingresar a la pestaña "empleados" mostrará una lista con foto, nombre, apellido y estado (activo o inactivo) de cada empleado.<br>2. El usuario administrador podra hacer click sobre el botón "info" para ver la ficha completa del empleado | 1. Las tareas creadas se guardaran en la base de datos MongoDB y se presentaran como un item en el calendario del empleado en la fecha correspondiente |
|                                | Como                            |  |   |  |
|                                | Quiero                          | poder asignar tareas a los empleados del camping desde la plataforma   | 3. En la ficha del empleado hay un calendario que muestra las tareas asignadas segun fecha y hora para el mismo   | 2. Cuando se quiera añadir una nueva tarea a un empleado se deberá abrir un widget con campos que el administrador deberá rellenar                     |
|                                | Para                            | mejorar la organización interna y evitar el uso de canales informales como WhatsApp donde puede perderse información o generar confusiones | 4. El administrador podra agregar tareas haciendo click en el botón "Agregar tarea" donde detallará: nombre de la tarea, fecha de inicio y fin<br>5. La tarea quedará asignada unicamente al empleado elegido   | 3. Las tareas solo pueden ser visualizadas por el empleado al que le fueron asignadas y por el administrador   |

- 8. Como usuario administrador del sistema Creek, quiero poder visualizar gráficos de la situación actual del camping para mejorar la toma de decisiones:** La visualización de estadísticas a través de gráficos es una funcionalidad clave para la gestión eficiente del camping. Esta herramienta permitirá al administrador acceder a indicadores relevantes como la cantidad de reservas realizadas, la duración promedio de las estadías, el nivel de ocupación, y otros datos útiles para el análisis operativo. La información podrá ser filtrada por fecha lo cual facilitará la identificación de patrones, tendencias y períodos de alta o baja demanda. Gracias a esta funcionalidad, se promoverá una toma de decisiones más informada y estratégica

| HISTORIA: VISUALIZACIÓN DE GRÁFICOS |                                 |   |  |  |
|-------------------------------------|---------------------------------|---|--|--|
| ID                                  | Descripción Historia de Usuario |   | Criterios de Aceptación  | Criterios de validacion  |
| 14                                  |                                 | usuario administrador del sistema Creek                             | 1. El usuario admin podrá ingresar a la pestaña "estadísticas" diversos graficos, listas e informes de la situacion actual del camping tal como cantidad de reservas, promedio de dia de estadía, cabañas con mayor ocupación, entre otros | 1. Se tomarán los datos de los cabañas, carpas y reservas directamente de base de datos MongoDB y se realizarán los calculos necesarios para la creacion de los graficos   |
|                                     | Como                            |   |  |  |
|                                     | Quiero                          | quiero poder visualizar gráficos de la situación actual del camping | 2. El administrador podrá filtrar la informacion segun fecha especifica, mes o año, segun le sea mas conveniente   | 2. Los gráficos deberan ser facilmente entendibles y legibles para la facil compension de aquellos que los visualicen<br>3. Los filtros aplicados (por fecha, mes o año) deberán reflejarse correctamente en los gráficos, actualizando los datos de manera dinámica y precisa.  |
|                                     | Para                            | para mejorar la toma de decisiones                                  |  | 4. Los títulos, leyendas y etiquetas de los gráficos deben estar correctamente nombrados y contextualizados con los datos que representan.<br>5. Todos los cálculos realizados para generar los gráficos deberán coincidir con los datos reales almacenados en la base de datos. |

- 9. Como usuario del sitio web Camping Arroyito, quiero poder crear una cuenta en el sitio para poder realizar reservas:** La página web del camping permitirá a los clientes acceder de forma rápida y sencilla a los servicios principales, mejorando su experiencia desde el primer contacto. Los usuarios

podrán registrarse con sus datos personales para crear una cuenta, lo que les permitirá gestionar sus reservas de forma segura y personalizada.

| HISTORIA: REGISTRO DE USUARIO |                                 |                                    |  |   |
|-------------------------------|---------------------------------|------------------------------------|--|---|
| ID                            | Descripción Historia de Usuario |                                    | Criterios de Aceptación  | Criterios de validación   |
| 1                             | Como                            | usuario del sitio Camping Arroyito | 1. La pagina debe permitirle al usuario ingresar nombre, apellido, email, contraseña, DNI, telefono, fecha de nacimiento y ocupación<br>2. El email no debe ser un email ya registrado | 1. Al terminar el registro se guardará un nuevo perfil de usuario en la base de datos con todos los datos proporcionados por el usuario.<br>2. Los datos deberán guardarse con una demora no mayor a 0,5 segundos |
|                               | Quiero                          | poder crear una cuenta en el sitio | 3. Todos los campos son obligatorios y no pueden estar vacios<br>4. Se crea la cuenta del usuario permitiendole realizar reservas de cabañas   | 3. La contraseña del usuario se guardará como un hash   |
|                               | Para                            | poder realizar reservas            |  |   |

**10. Como usuario del sitio Camping Arroyito, quiero ingresar el mail con el que me registre y mi contraseña para poder loguearme en el sitio:** A través de la página web, el usuario podrá iniciar sesión utilizando su correo electrónico y contraseña registrados previamente. Esta funcionalidad permitirá al usuario acceder a su perfil, visualizar y gestionar sus reservas y modificar su contraseña. La implementación de un sistema de login confiable también permite al camping proteger los datos de los usuarios, evitar accesos no autorizados y asegurar la integridad de las operaciones realizadas en el sitio.

| HISTORIA: REGISTRO DE USUARIO |                                 |   |   |  |
|-------------------------------|---------------------------------|---|---|--|
| ID                            | Descripción Historia de Usuario |   | Criterios de Aceptación   | Criterios de validacion  |
| 2                             | Como                            | usuario del sitio Camping Arroyito                      | 1. La pagina debe validar que el email y contraseña ingresados sean correctos y pertenezcan a un usuario registrado<br>2. Todos los campos son obligatorios y no pueden estar vacios            | 1. Se buscará el email ingresado dentro de la base de datos "users" de MongoDB para verificar que exista                                 |
|                               | Quiero                          | ingresar el mail con el que me registre y mi contraseña | 3. No se permitira loguearse a usuarios que ingresen un email no registrado<br>4. No se permitira loguearse a usuarios que ingresen una contraseña incorrecta                                   | 2. Se hará una comparación entre el hash de la contraseña ingresada y el hash de la contraseña guardada para determinar que son la misma |
|                               | Para                            | poder realizar loguearme en el sitio                    | 3. Al loguearse el usuario puede visualizar su perfil el cual cuenta con su información, con las reservas que realizó y tambien le permite cambiar su contraseña y cerrar su sesión en el sitio | 3. La demora de todo el proceso no puede ser mayor a 0,5 segundos  |

**11. Como usuario del sitio web Camping Arroyito, quiero poder hacer una búsqueda según las fechas en las que realizaré mi viaje y la cantidad de personas con las que viajare para poder ver las cabañas disponibles en esa fecha y para ese número de personas:** A través de la página, podrán buscar cabañas disponibles según sus fechas deseadas y la cantidad de personas que se hospedaran, visualizar la información detallada de cada cabaña (como capacidad, servicios incluidos, precio y fotos) y realizar una reserva directamente desde la web. Estas funcionalidades no sólo brindan comodidad al cliente, sino que también automatizan el proceso de reserva, reducen la carga operativa del personal y optimizan la ocupación del camping.

| HISTORIA: BUSQUEDA DE CABAÑAS DISPONIBLES |                                 |   |  |   |
|---|---------------------------------|---|--|---|
| ID  | Descripción Historia de Usuario |   | Criterios de Aceptación  | Criterios de Validación   |
| 3   | Como                            | usuario del sitio Camping Arroyito  | 1.La pagina debe permitir al usuario ingresar la fecha de arribo y la fecha de egreso así como tambien la cantidad de personas a hospedarse en el camping (adultos y niños)  | 1. El sistema debe filtrar las cabañas y mostrar únicamente las que coincidan con lo que el usuario busca.                            |
|   |                                 |   |  | 2. El ingreso de las fechas de check-in y check-out se hará mediante el componente Calendar que tendra un tamaño de no más de 350x350 |
|   | Quiero                          | poder hacer una búsqueda según la fechas en la que realizaré mi viaje y la cantidad de personas con las que viajaré | 2. No se podrán realizar reservas en fechas anteriores a la actual<br>3. El resultado de la búsqueda serán las cabañas disponibles en ese rango de fechas y que tengan la capacidad de hospedar la cantidad de personas ingresadas | 3. La busqueda no debe llevar más de 0.2 segundos   |
|   | Para                            | para poder ver las cabañas disponibles en esa fecha y para ese numero de personas                                   | 4. La busqueda podrá modificarse en cualquier momento actualizando automaticamente las cabañas que cumplan con los nuevos requisitos.  | 4. La resolusion de las imagenes de las cabañas debe ser de aproximadamente 1080 px   |



**12. Como usuario del sitio Camping Arroyito, quiero poder ver fotos y conocer las características de cada cabaña para validar que la cabaña tiene las comodidades que requieres antes de decidir si hacer la reserva o no:** Brindar información detallada y visual sobre las cabañas disponibles es fundamental para generar confianza en los potenciales huéspedes y facilitar el proceso de toma de decisiones. A través del sistema, el usuario podrá visualizar fotografías reales de cada cabaña, así como también consultar sus características principales. Esta funcionalidad permitirá a los usuarios comparar opciones de manera clara y transparente, asegurando que el alojamiento elegido se ajuste a sus necesidades antes de efectuar una reserva. De esta manera, se mejora la experiencia del usuario y se reduce la posibilidad de cancelaciones o disconformidades posteriores.

| HISTORIA: MOSTRAR INFORMACIÓN DE LA CABAÑA |                                 |   |   |
|--|---------------------------------|---|---|
| ID   | Descripción Historia de Usuario |   | Criterios de validacion   |
| 4  | Como                            | usuario del sitio Camping Arroyito  | 1. Se mostrará toda la información anexada a la cabaña  |
|  | Quiero                          | poder ver fotos y conocer las características de cada cabaña  | 2. El usuario debe poder visualizar toda la información clara utilizando una letra fuente Arial resaltando el nombre la cabaña y títulos en negrita   |
|  | Para                            | validar que la cabaña tiene las comodidades que requiero antes de decidir si hacer la reserva o no. | 3. La resolución de las imagenes debe ser de 1080 px  |
|  |                                 |   | 1. Cuando el usuario busca una cabaña le aparecerá una lista de todas las cabañas disponibles con una imagen de la cabaña, capacidad y una descripción<br>2. Cuando el usuario hace click sobre el boton "reservar" lo llevará a la pagina de información de la cabaña la cual le mostrará mas fotos, la capacidad, descripción, información del check-in y check-out y servicios disponibles |

**13. Como usuario del sitio Camping Arroyito, quiero poder elegir una cabaña, una fecha de ingreso, una fecha de salida y la cantidad de acompañantes para poder realizar una reserva de una cabaña:** El proceso de reserva online es una funcionalidad central para brindar autonomía y comodidad a los usuarios que desean hospedarse en el camping. Esta funcionalidad facilita la planificación del viaje por parte del usuario y optimiza la gestión de disponibilidad en el sistema, contribuyendo a una operación más ágil y eficiente del camping.

| HISTORIA: RESERVAR UNA CABAÑA |                                 |   |   |
|-------------------------------|---------------------------------|---|---|
| ID                            | Descripción Historia de Usuario |   | Criterios de validacion   |
| 5                             | Como                            | usuario del sitio Camping Arroyito  | 1. Se guardaran los datos de la reserva en la colección "bookings" la base de datos en MongoDB                          |
|                               | Quiero                          | poder elegir una cabaña, una fecha de ingreso, una fecha de salida y la cantidad de acompañantes  | 2. El proceso de guardar los datos no puede tardar mas de 0,5 segundos.   |
|                               | Para                            | poder realizar una reserva de una cabaña  | 3. Las fechas durante las cuales la cabaña estará reservada se marcarán como no disponibles                             |
|                               |                                 |   | 4. Debe abrirse una ventana pop up que le haga entender perfectamente al usuario que su reserva fue realizada con exito |
|                               |                                 | 2. El usuario podrá elegir entre las cabañas disponibles la que mas se adecue a sus necesidades.<br>3. El usuario deberá hacer click en el botón Reservar la cual lo dirigirá a la página de informacion de la cabaña<br>4. El usuario podrá modificar los datos de arribo y salida si es que así lo desea y si es que la fecha esta disponible.<br>5. El usuario deberá hacer click en el boton Reservar ahora<br>6. Se le mostrará al usuario un cartel de "Reserva exitosa" y se lo redirigirá a la pagina de "mis reservas" |   |

**14. Como usuario del sitio web Camping Arroyito, quiero tener siempre conocimiento del estado de mi reserva para tener claridad y seguridad sobre mi plan de visita al camping:** Contar con información clara y actualizada

sobre el estado de una reserva es fundamental para brindar una experiencia transparente y confiable a los usuarios. Esta funcionalidad permitirá que cada usuario pueda acceder a su perfil dentro del sistema y consultar el estado actual de sus reservas. De este modo, el visitante podrá realizar un seguimiento en tiempo real de su reserva, conocer si requiere alguna acción adicional (como el pago de una señal), y sentirse seguro respecto a la planificación de su estadía. Esta herramienta reduce la necesidad de consultas manuales al personal y fortalece la comunicación entre el camping y sus clientes.

| HISTORIA: CONOCER ESTADO DE LA RESERVA |                                 |  |  |
|--|---------------------------------|--|--|
| ID                                     | Descripción Historia de Usuario |  | Criterios de Validación  |
| 6                                      | <b>Como</b>                     | usuario del sitio Camping Arroyito                                 | 1. El usuario registrado podrá acceder a su perfil en donde se puede visualizar una lista con todas las reservas activas y el estado actual en el que se encuentra tal como: "Pendiente", "Activa" o "Cancelada" |
|  | <b>Quiero</b>                   | tener siempre conocimiento del estado de mi reserva                | 2. Ante el cambio de un estado en la reserva el sistema deberá actualizar automáticamente el estado en la lista de reservas que el usuario puede visualizar  |
|  | <b>Para</b>                     | para tener claridad y seguridad sobre mi plan de visita al camping | 3. Ante el cambio del estado en la reserva el sistema automáticamente enviará un mail al correo del usuario probeyendo toda la informacion necesaria   |

**15. Como usuario del sitio web Camping Arroyito, quiero poder cancelar mi reserva para informar que no visitaré el camping y que me devuelvan el dinero de mi señal:** Brindar al usuario la posibilidad de cancelar su reserva de forma sencilla y autónoma es esencial para mejorar la experiencia del cliente y mantener una gestión ordenada de la disponibilidad del camping. A través del sitio web, el usuario podrá acceder a sus reservas activas y, si lo desea, cancelarlas dentro del plazo permitido. Esta acción actualizará automáticamente el estado de la reserva, liberará la cabaña asignada, y generará un mensaje de confirmación con la información necesaria para solicitar la devolución de la señal.

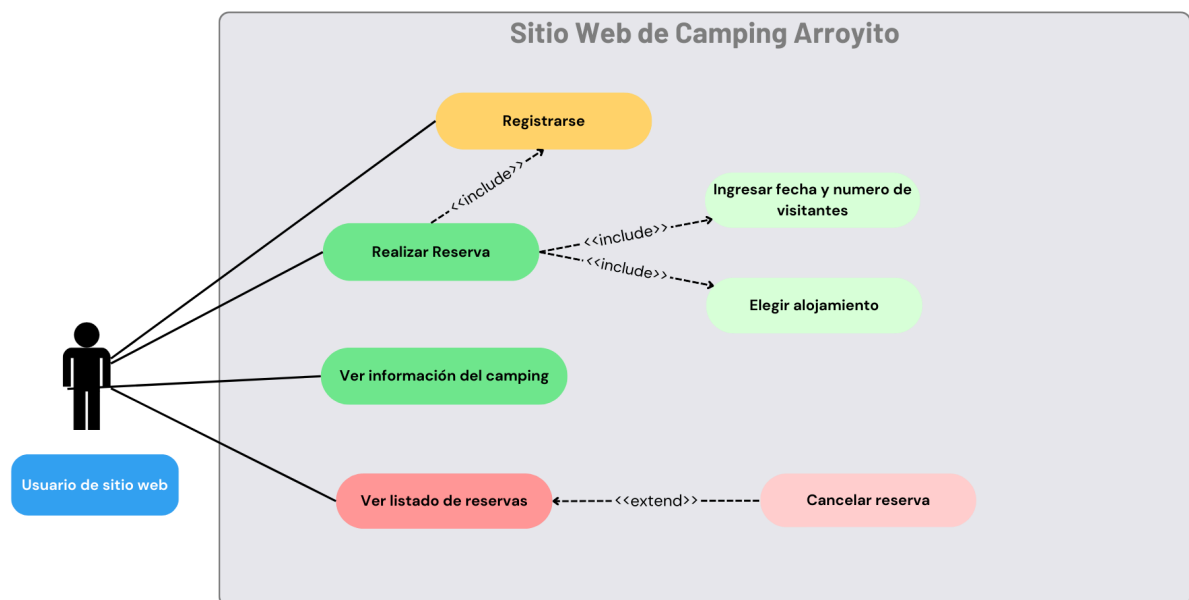
| HISTORIA: CANCELAR RESERVA |                                 |  |   |
|----------------------------|---------------------------------|--|---|
| ID                         | Descripción Historia de Usuario |  | Criterios de Validación   |
| 2                          | <b>Como</b>                     | usuario del sitio Camping Arroyito   | 1. El sistema accederá a los registros de reservas almacenados en la base de datos MongoDB y devolverá una lista con las reservas correspondientes al usuario loggeado  |
|                            | <b>Quiero</b>                   | poder cancelar mi reserva  | 2. La funcionalidad de cancelación debe estar disponible únicamente para usuarios autenticados con reservas activas.  |
|                            | <b>Para</b>                     | informar que no visitaré el camping y que me devuelvan el dinero de mi señal | 3. El botón "Cancelar" no debe ser visible ni interactivo si faltan menos de 48 horas para la fecha de Check-In.<br>4. Al cancelar una reserva, el estado de la reserva debe cambiar a "Cancelada" en la base de datos y la cabaña asignada debe quedar disponible nuevamente.<br>5. Todos los cambios generados por la cancelación (estado de reserva, disponibilidad de la cabaña, envío de email) deben reflejarse correctamente y en tiempo real en el sistema. |

## Diagrama de casos de uso

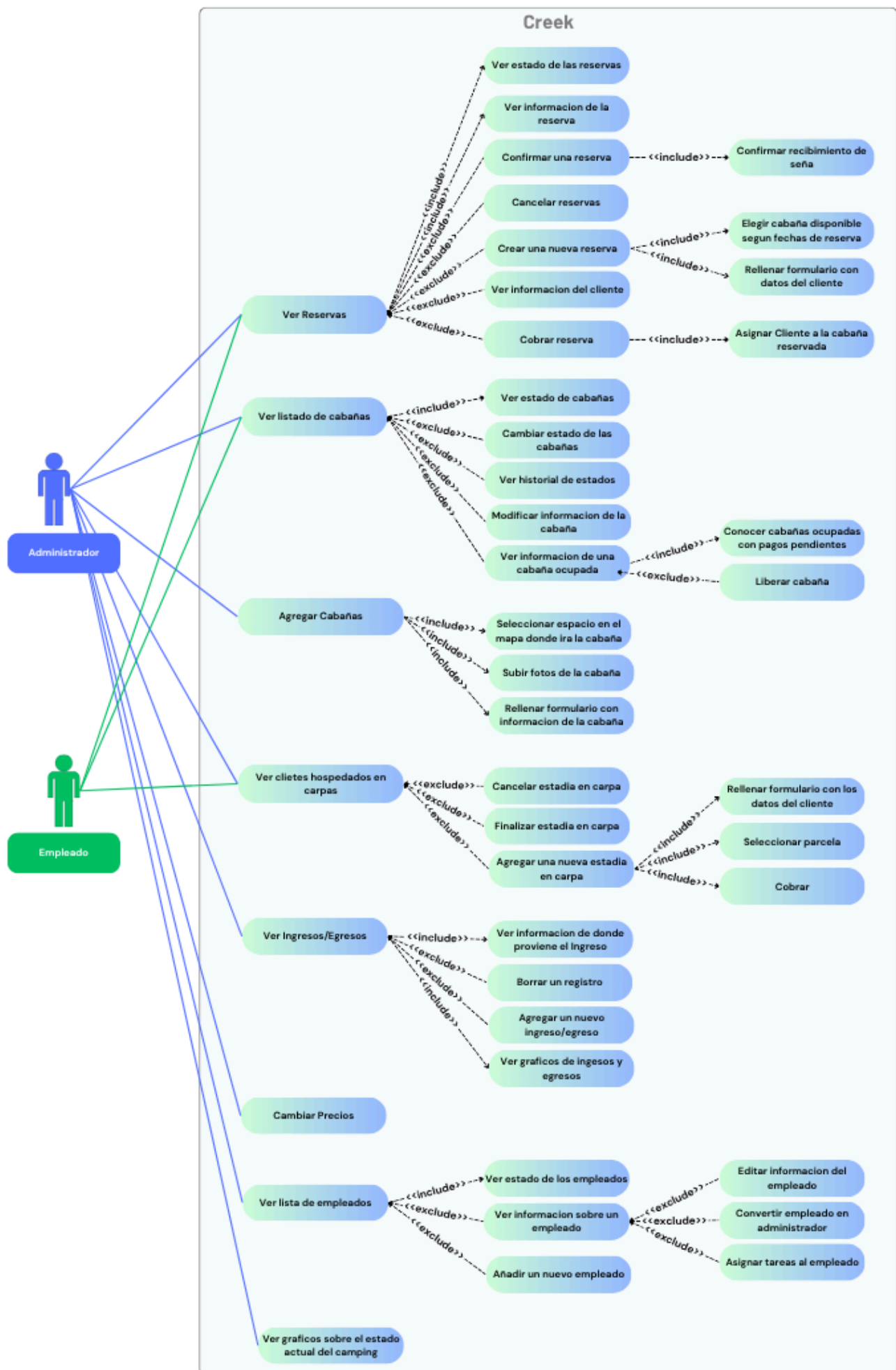
El siguiente paso en el desarrollo del sistema es realizar la esquematización de los diferentes actores involucrados en el día a día del camping, identificando a los usuarios principales que interactúan con el sistema. Para esto, se utilizan diagramas de caso de uso, una herramienta clave en el análisis y diseño de sistemas que permite visualizar de manera clara las interacciones entre los usuarios y las funcionalidades del sistema.

En el caso del Camping Arroyito, el diagrama de caso de uso identifica tres roles principales: administradores, empleados y clientes, cada uno con funcionalidades específicas dependiendo de su rol en el sistema. Esta segmentación asegura que cada usuario tenga acceso solo a las herramientas necesarias para sus tareas, lo que no solo mejora la eficiencia del sistema, sino que también garantiza la seguridad de la información.

Ahora se describirán los casos de uso correspondientes a cada actor identificado, detallando las funcionalidades disponibles para cada uno:







## **Roles**

Para comprender cómo el sistema Creek organiza y distribuye las funcionalidades, es esencial entender las diferencias entre los roles que interactúan directa o indirectamente con él. Cada rol representa un grupo de usuarios con necesidades y niveles de acceso específicos, diseñados para garantizar un uso eficiente, seguro y organizado del sistema

Cliente: Los clientes son el corazón del negocio, ya que representan a las personas que visitan el camping con la intención de hospedarse en una cabaña o rentar una parcela para carpas. Aunque los clientes no interactúan directamente con el sistema Creek (es decir, no tienen credenciales ni acceso interno al sistema), su experiencia depende en gran medida de las funcionalidades que este ofrece a los otros roles, sus solicitudes ingresan automáticamente al sistema, donde los administradores y empleados las gestionan

Desde la perspectiva del cliente, su interacción con el camping se realiza principalmente a través de la página web del Camping Arroyito. A través de este portal, los clientes pueden:

- Realizar reservas de cabañas o parcelas.
- Consultar la disponibilidad en tiempo real.
- Recibir confirmaciones y consultar detalles sobre sus reservas.
- Cancelar reservas
- Conocer más información sobre el camping (ubicación, servicios disponibles, actividades de la zona, entre otros)

Empleado: El rol de empleado está diseñado para los trabajadores operativos del camping, quienes se encargan de ejecutar las tareas diarias que garantizan el buen funcionamiento del lugar. Aunque los empleados tienen acceso al sistema Creek, su nivel de permisos es más limitado que el de los administradores, lo cual asegura que solo puedan interactuar con las funcionalidades necesarias para su trabajo.

Entre las secciones disponibles para los empleados, se incluyen:

- Los empleados pueden actualizar el estado de las cabañas (por ejemplo, marcarlas como disponibles o en mantenimiento).
- Pueden hacer un control de reservas, en donde gestionan el flujo normal de cuando un huésped llega al camping a través del sistema, lo cual implica asegurarse que el pago de la seña se haya efectuado lo cual confirma la reserva, cobrarle al cliente el resto de la reserva una vez que se presente en el camping, seleccionar si el cliente hace un pago completo o parcial y luego de esto se le asigna a la cabaña que había reservado y esta se marcará como ocupada.
- También pueden controlar que las estadías se encuentren pagadas ya que visualizarán una alerta de “Reserva impaga” cuando las personas hospedadas en cabañas o carpas aun no hayan completado el pago. El empleado podrá completar el pago así como también visualizar los datos de la reserva.
- El empleado también podrá registrar reservas de cabañas de forma manual, solo deberá completar un formulario con los datos del cliente, poner la cantidad de personas que desean hospedarse en el camping y en qué fechas, en base a esto el sistema le devolverá las cabañas disponibles para esa reserva al usuario y cual es el precio a cobrar.
- También podrá registrar estadías de clientes en parcelas, en donde deberá completar un formulario con los datos del cliente, seleccionar la parcela que será ocupada y durante qué fechas y proceder con el cobro.

Administrador: Los administradores son los usuarios con el mayor nivel de acceso dentro del sistema Creek. Este rol está diseñado para los propietarios o gerentes del Camping Arroyito, quienes tienen la responsabilidad de supervisar todas las operaciones del camping y tomar decisiones estratégicas.

Además de contar con las funcionalidades disponibles para los empleados, los administradores también disponen de las siguientes funcionalidades:

- La gestión de reservas, donde ellos pueden visualizar y gestionar todas las reservas realizadas por los clientes.
- La supervisión de operaciones, los administradores tienen la capacidad de acceder y modificar la información relacionada a las cabañas y parcelas.

- El control contable, debido a que tienen acceso completo a la sección de contabilidad, donde pueden realizar seguimientos financieros, revisar ingresos y egresos, cambiar el precio de las cabañas y carpas y visualizar gráficos que los ayuden a mantener un mejor control y en la toma de decisiones.
- Los administradores tienen acceso a los gráficos que detallan la situación actual del camping, lo cual les permitirá tomar mejores decisiones que ayuden con la mejora y el manejo del camping.
- Gestión de Personal: Los administradores también pueden supervisar y agregar tareas a los empleados, acceder a información personal y de contacto, y realizar ajustes organizativos cuando sea necesario. También podrán otorgar permisos de administrador a usuarios que aún no lo sean.

Este nivel de acceso les permite no solo gestionar el día a día del camping, sino también analizar su rendimiento general y tomar decisiones informadas para optimizar su operación.

Es importante destacar que los empleados no tienen acceso a las secciones de Contabilidad, Gestión de Personal y Admin Portal, lo que garantiza que la información financiera y administrativa del camping quede restringida únicamente a los administradores.

Esta estructura de roles asegura que cada usuario tenga acceso solo a las herramientas y datos necesarios para realizar su trabajo, minimizando riesgos de errores o accesos no autorizados, y optimizando la operación del Camping Arroyito.

## **Casos de uso**

### Página Web (Usuario)

1. Registrarse: El usuario deberá crear una cuenta en la página web del camping con la cual podrá realizar las reservas. La cuenta se crea agregando un mail, contraseña y los datos personales requeridos.
2. Realizar reserva: Permite a los usuarios seleccionar un alojamiento, ingresar la cantidad de personas que se hospedan, las fechas deseadas y conocer el precio que deberán abonar una vez asistan al camping.

3. Ver información del camping: El usuario puede conocer detalles específicos del camping tal como su ubicación, los servicios que provee, ver fotos del camping, entre otros datos.
4. Ver Listado de Reservas: Permite al usuario consultar sus reservas activas y se le permitirá cancelar la reserva hasta con 3 días de anticipación.

## Sistema Creek

### Empleado y Administrador

1. Ver Reservas: Visualiza todas las reservas realizadas, incluyendo información detallada sobre las fechas, los usuarios, las cabañas elegidas para la reserva y el estado de cada reserva. Tendrá la opción de poder ver información más detallada sobre el cliente, también podrá cobrar las reservas una vez que el usuario se presente en el camping e incluso podrá crear las reservas de aquellos clientes que decidan efectuarlas mediante otros medios tales como llamada telefónica o whatsapp.

|  |   |   |   |
|--|---|---|---|
| Nombre del caso de uso   | Crear una reserva   | Nro   | 1 |
| Actores  | Administrador o Empleado  |   |   |
| Objetivo   | Crear una reserva de forma manual a través del sistema Creek  |   |   |
| Precondiciones   | 1. El usuario administrador o empleado debe estar loggeado en el sistema  |   |   |
| Flujo Principal  |   | Flujo Alternativo   |   |
| 1. El usuario accede a la pestaña "Reservas"   |   |   |   |
| 2. El usuario hace click en el botón "Crear Reserva"   |   |   |   |
| 3. El usuario ingresa la cantidad de adultos y niños que se hospedarán   |   |   |   |
| 4. El usuario elige en el calendario las fechas durante las cuales se desea hacer la reserva                                       |   |   |   |
| 5. El sistema devuelve la lista de cabañas con capacidad para esta cantidad de personas y disponible durante las fechas ingresadas |   | 1. Si no hay cabañas disponibles para esa cantidad de personas y durante esas fechas, se mostrara un mensaje de error y permitirá modificar los criterios de búsqueda |   |
| 6. El usuario elige una cabaña   |   |   |   |
| 7. El usuario ingresa el mail de la persona para la cual se hace la reserva  |   | 2. Si el mail esta ya registrado, automaticamente se autocompletan los campos con la información del cliente  |   |
| 8. El usuario ingresa los datos personales del cliente tales como número de telefono, DNI y ocupación                              |   |   |   |
| 9. El sistema devuelve el precio que deberá pagar el cliente por la reserva  |   |   |   |
| 10. El usuario hace click en el botón "Crear reserva"  |   |   |   |
| 11. La reserva se crea   |   |   |   |
| 12. Se redirige al usuario a la pestaña de "Reservas" donde puede visualizar la nueva reserva                                      |   |   |   |
|  |   |   |   |
|  |   |   |   |
| Postcondiciones  | 1. La reserva queda creada bajo el estado "Pendiente" a la espera de que el cliente mande comprobante de pago de la seña solicitada<br>2. La cabaña elegida queda queda bloqueada en el sistema para las fechas indicadas, impidiendo nuevas reservas sobre ese mismo periodo.<br>3. Si el cliente no estaba registrado previamente, se crea un nuevo usuario vinculado al correo electrónico ingresado, permitiéndole acceder al sistema y consultar su reserva.<br>4. Se envia un mail al cliente con los datos de la reserva, contraseña temporal que deberá cambiar al ingresar por primera vez a la pagina web y detalles de donde depositar la seña y a donde mandar el comprobante de pago |   |   |

*Trazo grueso del caso de uso "Crear una reserva"*

2. Ver listado de las cabañas: Muestra el estado actual de las cabañas (disponibles, ocupadas, en mantenimiento), también se puede cambiar el estado de la cabaña a Mantenimiento o si la cabaña está ocupada se puede liberar lo cual implica marcarla como “En Mantenimiento”
  - Mostrar historial de estados: Se trata de una lista que muestra todos los estados por los que transiciona la cabaña.
  - Modificar información de la cabaña: Aquí el usuario puede cambiar los datos de la cabaña que luego se muestran en la página web como por ejemplo cambiar la descripción de la cabaña, sacar o agregar fotos, cambiar la capacidad, etc.
  - Ver información de una cabaña ocupada: cuando una cabaña se encuentre en estado “ocupado” el usuario puede conocer los datos del cliente que se está hospedando allí, lo que debía pagar, si se debe algún pago, etc.

|  |  |  |   |
|--|--|--|---|
| <b>Nombre del caso de uso</b>  | Liberar una cabaña   | <b>Nro</b>   | 3 |
| <b>Actores</b>   | Administrador o empleado   |  |   |
| <b>Objetivo</b>  | Liberar una cabaña cuando se vaya un cliente así puede ser acondicionada y preparada para el siguiente cliente   |  |   |
| <b>Precondiciones</b>  | 1. El usuario administrador o empleado debe estar loggeado en el sistema<br>2. Debe haber existido una reserva previa de un cliente sobre la cabaña<br>3. Un huesped debe estar ocupando la cabaña, es decir, la cabaña esta en estado "Ocupada" |  |   |
| <b>Flujo Principal</b>   |  | <b>Flujo Alternativo</b>   |   |
| 1. El usuario accede a la pestaña "Cabañas"  |  |  |   |
| 2. Busca la cabaña ocupada por el usuario que se esta retirando del camping                                    |  |  |   |
| 3. El usuario hace click en el botón "Liberar"   |  |  |   |
|  |  | 1. Si la reserva aun tiene un pago pendiente, el botón liberar se muestra desactivado  |   |
|  |  | 2. Hay un icono que indica que la reserva que ocupa esa cabaña aún tiene pagos pendientes  |   |
|  |  | 3. El usuario hace click sobre el icono de pagos pendientes lo cual lo redirige a la plantilla con el registro de pagos                              |   |
|  |  | 4. El usuario revisa la tabla de historial de pagos y revisa cual es el m  |   |
|  |  | 5. El usuario le cobra el monto faltante al cliente  |   |
|  |  | 6. El usurio hace click sobre "Completar Pago"   |   |
|  |  | 7. El pago queda registrado en la tabla  |   |
|  |  | 8. El usuario vuelve a la pestaña "Cabañas"  |   |
|  |  | 9. El botón liberar ahora esta disponible al lado de la cabaña que est   |   |
|  |  | ocupando el cliente  |   |
| 4. El usuario hace click sobre "Aceptar" en el pop-up preguntandole si esta seguro de querer liberar la cabaña |  |  |   |
| 5. Se libera la cabaña (se desliga la reserva que la estaba ocupando)  |  |  |   |
| 7. La cabaña se pone en estado "Mantenimiento" con el comentario "Limpiar cabaña"                              |  |  |   |
|  |  |  |   |
|  |  |  |   |
| <b>Postcondiciones</b>   |  | 1. Se guarda el registro de la reserva en el historial de estados de la cabaña<br>2. Se guarda el registro de la reserva en el historial de reservas |   |

### *Trazo fino del caso de uso “Liberar una cabaña”*

3. Ver clientes hospedados en carpas: Visualiza la lista de clientes que actualmente están alojados en carpas. Aquí el usuario puede cancelar la estadía en carpa (el cliente decide no hospedarse o se registró erróneamente la estadía entonces se cancela) y también finalizar las estadías (se completó la estadía y el cliente se retira del camping).

- Agregar nueva estadía en carpa: El usuario deberá registrar la estadía de todos aquellos clientes que quieren hospedarse en una carpa, para ellos deberá elegir la parcela en donde se hospeda el cliente, rellenar un formulario con los datos del cliente y por último cobrarles el monto debido (la estadía se puede pagar con un pago total o parcial)

## Administrador

1. Agregar Cabañas: Añade nuevas cabañas al sistema, especificando detalles como capacidad, ubicación, fotos, servicios disponibles, etc.
2. Cambiar Precios: Permite modificar los precios de alquiler de cabañas o carpas.
3. Ver Ingresos/Egresos: Permite visualizar una lista que cuenta con todos los ingresos y egresos financieros del camping, aquí se detalla el monto ingresado/gastado, de donde proviene, el estado (pagada, parcial o seña) y la fecha y hora en la que se concretó. También hay gráficos que muestran la situación actual del camping en términos económicos.
  - Ver información de donde proviene el ingreso: En el caso de los pagos, el usuario puede ver información específica del cliente y la reserva de la que proviene el pago
  - Borrar un registro: Los registros pueden ser borrados de forma permanente en caso de que se trate de un registro erróneo.
  - Añadir gastos o ingresos: Se podrán añadir manualmente registros de ingresos al camping que estén relacionados o no con el hospedaje de clientes en cabañas/carpas y también registros de pagos
4. Ver lista de empleados: muestra todos el personal del camping y sus estados.
  - Añadir nuevo empleado: El usuario puede agregar nuevos empleados. Cada empleado recibe una contraseña temporal la cual le es enviada por email y que se le pedirá cambiar con su primer ingreso al sistema.
  - Ver información de un empleado: Se puede elegir un empleado en específico para poder visualizar toda su información personal tal como nombre, email, salario, edad, puesto, antigüedad, etc. También el usuario puede asignar tareas al usuario y modificar sus datos y nivel de permisos.

|  |  |  |   |
|--|--|--|---|
| Nombre del caso de uso   | Asignar una tarea a un empleado  | Nro  | 2 |
| Actores  | Administrador  |  |   |
| Objetivo   | Agregar una terea al empleado para que le salga en el calendario cada vez que entra al sistema y este enterado de sus tareas del dia |  |   |
| Precondiciones   | 1. El usuario administrador o empleado debe estar loggeado en el sistema   |  |   |
| Flujo Principal  |  | Flujo Alternativo  |   |
| 1. El usuario accede a la pestaña "Empleados"  |  |  |   |
| 2. El usuario hace click en el botón "Ver" al lado del nombre el empleado al cual asignarle la tarea |  |  |   |
| 3. El usuario baja hasta la seccion "Calendario de tareas"   |  |  |   |
| 4. El usuario hace click en "Agregar tarea"  |  |  |   |
| 5. El sistema abre un widget donde se pueden ingresar los datos de la tarea                          |  |  |   |
| 6. El usuario ingresa el nombre de la tarea  |  |  |   |
| 7. El usuario ingresa la fecha de incio y fin  |  |  |   |
| 8. El usuario ingrssa la hora de inicio y fin  |  |  |   |
| 9. El usuario hace click en "Crear tarea"  |  |  |   |
|  |  | 1. Si al final no se desea crear la tarea el usuario puede hacer click en "Cancelar" |   |
| 10. La tarea queda creada y se la puede visualizar en el calendario                                  |  | 2. Si se hace click sobre la tarea en el calendario se puede borrar la tarea creada  |   |
|  |  |  |   |
|  |  |  |   |
|  |  |  |   |
|  |  |  |   |
| Postcondiciones  | 1. El empleado podrá ver la tarea en el calendario que aparece en la pagina principal cuado ingresa al sistema                       |  |   |

### *Trazo grueso del caso de uso "Agregar tarea a un empleado"*

- Ver gráficos del estado actual del camping: Los administradores tienen acceso a el portal de estadísticas el cual muestra gráficos sobre la situación actual del camping, es decir, nivel de ocupación, las cabañas con más reservas, la cantidad de reservas por día, mes, año, entre otros datos útiles para la toma de decisiones.

## Diseño

El diseño es una etapa crucial en el desarrollo de cualquier sistema, ya que establece cómo se organizará y estructurará la solución antes de proceder con su implementación. En esta sección, se busca traducir los requerimientos funcionales previamente identificados en una representación visual y técnica que guíe el desarrollo del sistema. El objetivo es garantizar que todos los componentes del sistema trabajen de manera integrada y eficiente, cubriendo las necesidades de los usuarios y resolviendo los problemas detectados en el análisis previo.

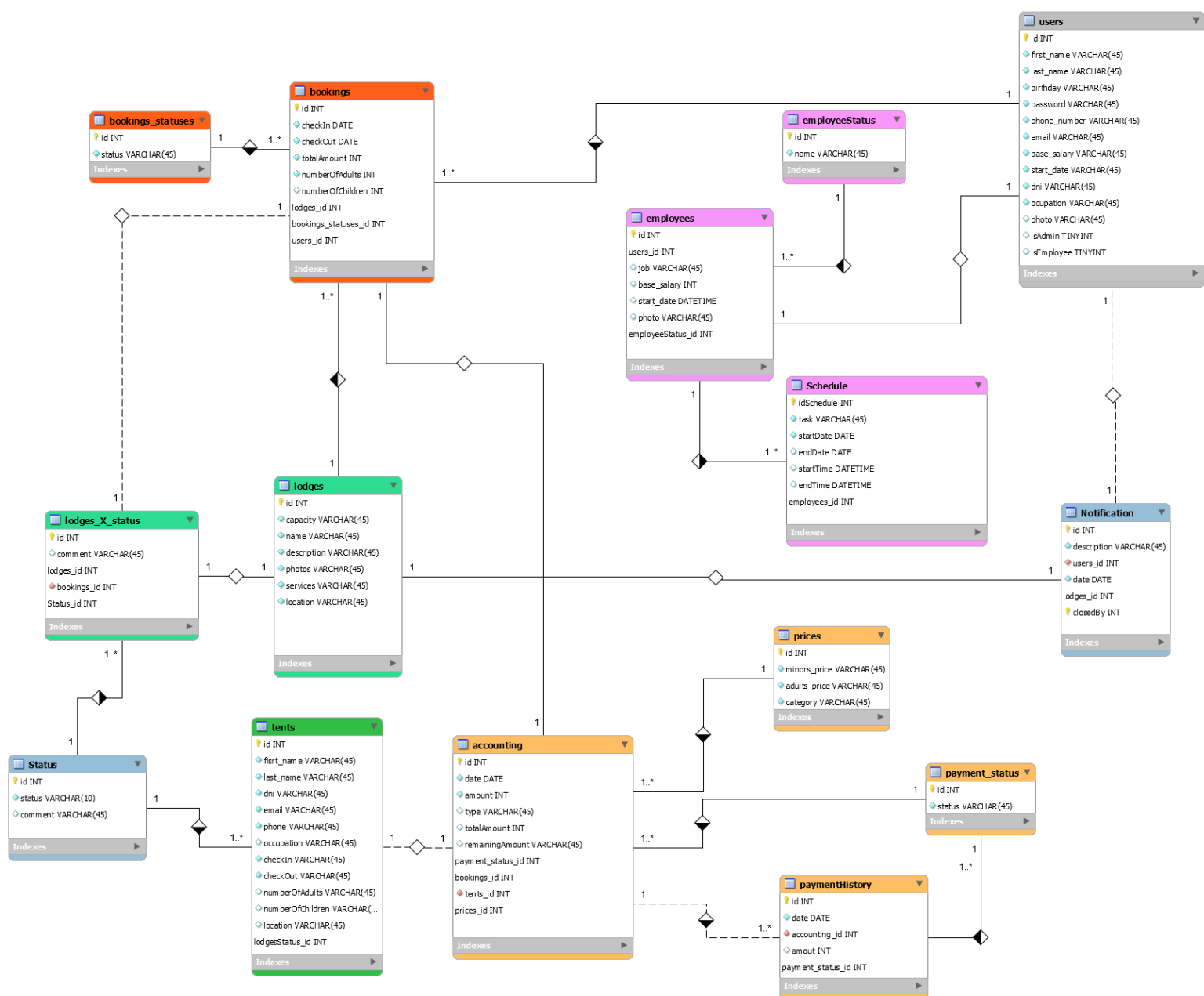
## Diagramas

### Diagramas de Clase

Los diagramas de clase son una herramienta fundamental en el diseño de sistemas orientados a objetos. Representan de manera visual la estructura del sistema,



mostrando las clases, sus atributos y las relaciones entre ellas. Este tipo de diagrama, es clave para comprender cómo se organizan y conectan los diferentes elementos que componen un software. En este trabajo integrador, los diagramas de clase cumplen un rol crucial para justificar y validar el diseño del sistema Creek. Proporcionan una base sólida para la implementación al detallar los elementos fundamentales del sistema y cómo estos están interconectados. Además, demuestran un enfoque metódico en el desarrollo, asegurando que el sistema propuesto esté bien diseñado desde su conceptualización.



users:

- Representa a los usuarios del sistema, como clientes, administradores y empleados.
- Contiene todos los atributos necesarios para poder identificar inequívocamente al usuario como nombre, apellido, DNI, correo electrónico, fecha de nacimiento, número de teléfono, etc.
- Tiene 2 atributos que permitirán diferenciar si el usuario es Administrador o Empleado, en caso de ser alguno de estos dos, el atributo correspondiente se marcará como True.
- Se relaciona con las entidades employees, bookings y notification.

employees:

- Detalla información específica sobre los empleados, como puesto de trabajo, salario base y fecha de inicio.
- Se relaciona con employeeStatus para definir el estado del empleado.
- Se relaciona con users la cual contendrá información personal del empleado.
- Se relaciona con Schedule ya que cada tarea agregada le corresponde a un empleado.

Schedule:

- Son las tareas creadas que se agregaran al calendario de tareas de los empleados.
- Cada tarea tiene un nombre, una fecha de inicio y fecha de fin. Se relaciona con employees en donde cada tarea está asignada a un empleado.

employeeStatus:

- Define los diferentes estados que un empleado puede tener, como "Activo" o "Inactivo".
- Se relaciona con employees.

lodges:

- Representa las cabañas del camping. Contiene información como capacidad,

nombre, descripción, fotos, y precios.

- Se relaciona con bookings para saber que usuarios realizan una reserva de esa cabaña
- Se relaciona con lodges\_x\_status para poder mantener un registro de los diferentes estados por los que atraviesa la cabaña a lo largo del tiempo.

status:

- Define los estados posibles de las cabañas y carpas, como "Disponible" o "Ocupada".
- Se relaciona con lodges\_x\_status para registrar el estado histórico de las cabañas.
- También se relaciona con Tents para registrar los estados por los que atraviesa una determinada estadia en Carpa

lodges\_x\_status:

- Registra el historial de estados de cada cabaña.
- Relaciona lodges con status.
- Guarda el registro de Booking siempre que el estado de la cabaña sea "Ocupada"

bookings:

- Maneja las reservas realizadas por los clientes. Contiene datos como fechas de entrada y salida, número de adultos y niños que se hospedaran (debido a que dependiendo de eso varia el precio)
- Se relaciona con lodges para poder saber qué cabaña reservó el usuario
- Se relaciona con bookings\_statuses para poder determinar en que estado se encuentra la reserva (Activa, cancelada o finalizada)
- Se relaciona con accounting para permitir visualizar a que reserva pertenece el nuevo registro de ingresos
- Se relaciona con users para poder acceder a los datos personales del cliente.

bookings\_statuses:

- Define los diferentes estados de las reservas, como "Confirmada", "Pendiente" o

"Cancelada".

- Se relaciona con bookings.

tents:

- Representa a los clientes que se hospedan en carpas. Contiene información como nombre, apellido, contacto y detalles de la reserva.
- Se relaciona con status ya que puede atravesar los estados "ocupada" y "disponible"
- Se relaciona con accounting para poder saber a que estadia en carpa pertenece el nuevo ingreso de dinero

accounting:

- Gestiona las operaciones contables del camping, como ingresos y gastos
- Tiene como atributos el monto de dinero ingresado actualmente (amount), el monto total a pagar (totalAmount), y la cantidad de dinero que se debe en caso de que se haya hecho un pago incompleto (remainingAmount), el tipo (ingreso o egreso) y la fecha en la que se realizó el registro.
- Se relaciona con bookings para saber a que reserva corresponde el pago
- Se relaciona con payment\_status para poder saber en qué estado se encuentra el pago (Pagada, Parcial, Señal o Cancelado)
- Se relaciona con paymentHistory la cual lleva un registro de todos los pagos correspondientes a un registro de ingresos.
- Se relaciona con prices para saber que valor de precios deberá tomar para poder calcular el monto a pagar de la reserva

Prices:

- Son los registros de los precios que las cabañas y carpas fueron tomando a lo largo del tiempo
- Tienen como atributos el precio para niños y el precio para adultos. También cuenta con el atributo "categoría" que permite saber si estos precios corresponden a las cabañas o a las carpas.
- Se relacionan con Accounting

payment\_status:

- Define los estados de los pagos, como "Pagado", "Parcial", "Cancelado" o "Seña"
- Se relaciona con accounting y paymentHistory.

paymentHistory:

- Registra el historial de pagos relacionados con reservas u operaciones contables.
- Se relaciona con accounting para saber a qué pago corresponde el historial
- Se relaciona con payment\_status para poder hacer un seguimiento de los distintos estados por los que pasa el pago

notification:

- Gestiona las notificaciones enviadas a los usuarios del sistema.
- Se relaciona con users para saber a que usuario hace referencia la notificación
- Tiene un atributo llamado "closedBy" para poder saber que usuario (administrador o empleado) cerró la notificación y evitar que le vuelva a aparecer

El sistema organiza las relaciones entre clientes, reservas, empleados y operaciones contables de manera centralizada. Las tablas users, lodges, tents, bookings y accounting son ejes principales, conectándose con otras entidades para proporcionar funcionalidad como gestión de pagos, historial de estados y administración de usuarios. Esto permite un manejo integral y eficiente del camping.

### **Diagramas de Estado:**

Los diagramas de estado son una herramienta visual utilizada durante el desarrollo de sistemas para representar cómo un objeto o entidad cambia su estado a lo largo del tiempo en respuesta a eventos o condiciones específicas.

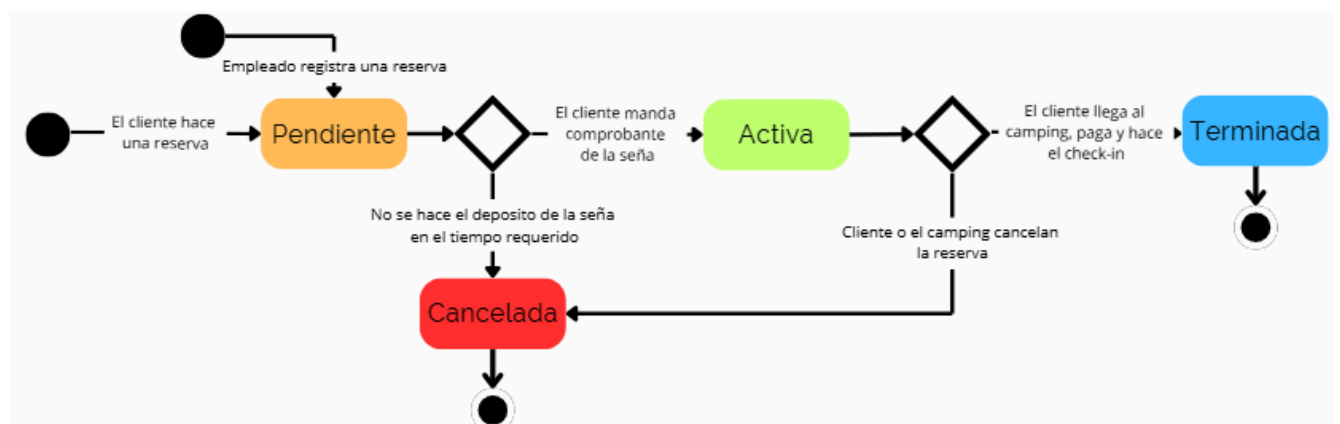
Describen los diferentes estados por los que pasa un objeto en un sistema y las transiciones entre estos estados, desencadenadas por eventos o acciones específicas. En el contexto del proyecto de implementación de un sistema de gestión para Camping Arroyito, los diagramas de estado desempeñan un papel crucial para modelar el comportamiento de los elementos claves para la funcionalidad del sistema.

### Diagrama de los estados de las reservas:

Al momento de efectuar una reserva, el cliente cuenta con dos modalidades disponibles. La primera consiste en realizar la gestión a través del sitio web oficial de Camping Arroyito. En este caso, luego de seleccionar una cabaña disponible durante las fechas especificadas y hacer click en "Reservar", la reserva queda inicialmente en estado "pendiente", a la espera de la confirmación del pago de la seña requerida. Para validar dicha seña, el cliente debe enviar una imagen del comprobante de transferencia al número de WhatsApp detallado en el email que se le enviará. Una vez recibido el comprobante, un empleado o administrador del camping verifica la operación y, de ser correcta, procede a actualizar el estado de la reserva a "activa".

El día del check-in, al momento de presentarse en el camping, el cliente es informado del monto restante correspondiente a su estadía. Una vez abonado este importe, se considera que la reserva ha sido efectivamente completada por lo tanto esta se pone en estado "Completada", y el cliente comienza su período de alojamiento en la cabaña asignada. Cabe destacar que las reservas pueden ser canceladas tanto en estado "pendiente" como "activa", en cuyo caso se notificará al cliente vía correo electrónico sobre la cancelación.

La segunda modalidad de reserva contempla los canales de atención directa, como WhatsApp, vía telefónica o de manera presencial. En estas situaciones, es un empleado quien se encarga de registrar los datos del cliente en el sistema y efectuar la reserva. El flujo de la reserva será el mismo al detallado para las reservas realizadas desde la página web.

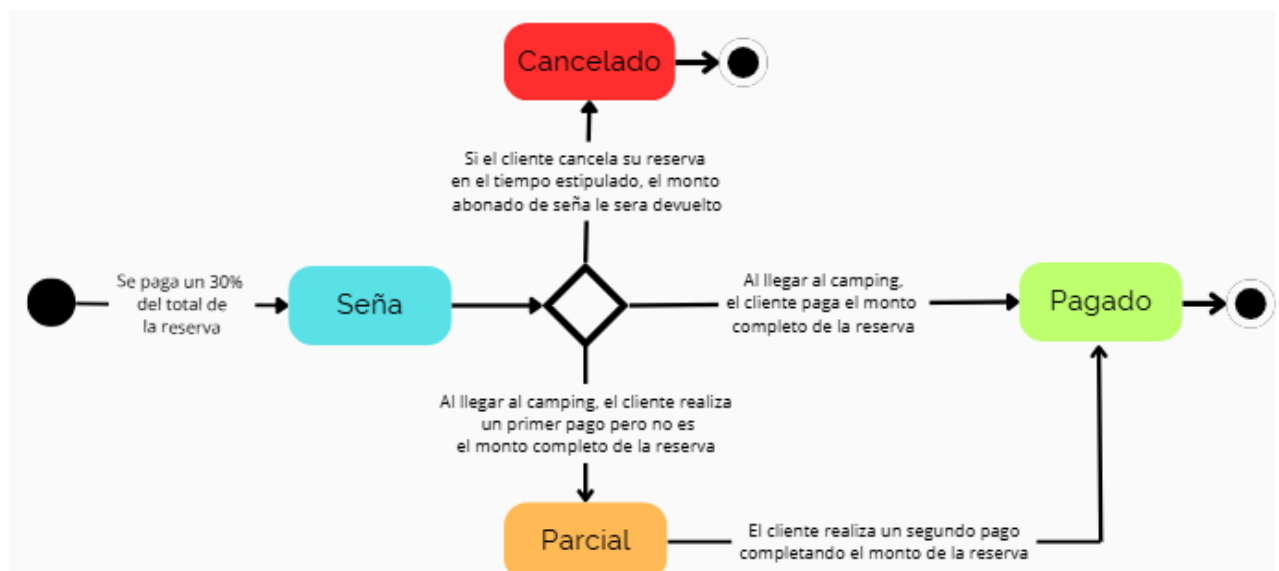


### Diagrama de los estados de los pagos:

Los pagos asociados a las reservas pueden presentar cuatro estados diferentes a lo largo del proceso. El primero de ellos es el estado "Seña", que corresponde al abono inicial del 30% del valor total de la reserva. Este pago se exige al momento de efectuar una reserva a través del sitio web y tiene como finalidad asegurar la disponibilidad del alojamiento.

Una vez que el cliente se presenta en el camping, puede optar por abonar la totalidad restante del monto, lo que hace que el estado del pago se actualice a "Pagado", indicando que la operación fue saldada en su totalidad. Sin embargo, en determinadas situaciones, se les ofrece a los huéspedes la posibilidad de realizar un pago parcial al momento del ingreso, con el compromiso de completar el pago antes de finalizar su estadía. En estos casos, el estado del pago se modifica a "Parcial", y una vez que se cancela el importe restante, el sistema actualiza automáticamente el estado a "Pagado".

El último posible estado es el de "Cancelado", utilizando en situaciones en donde el cliente haya realizado ya el pago de una seña pero días después decide cancelar la reserva, la política del camping es que el dinero de la seña le será devuelto al cliente si cancela hasta 3 días antes del día estipulado de Check-In, entonces aquellos ingresos que son devueltos al cliente serán marcados como "Cancelado"



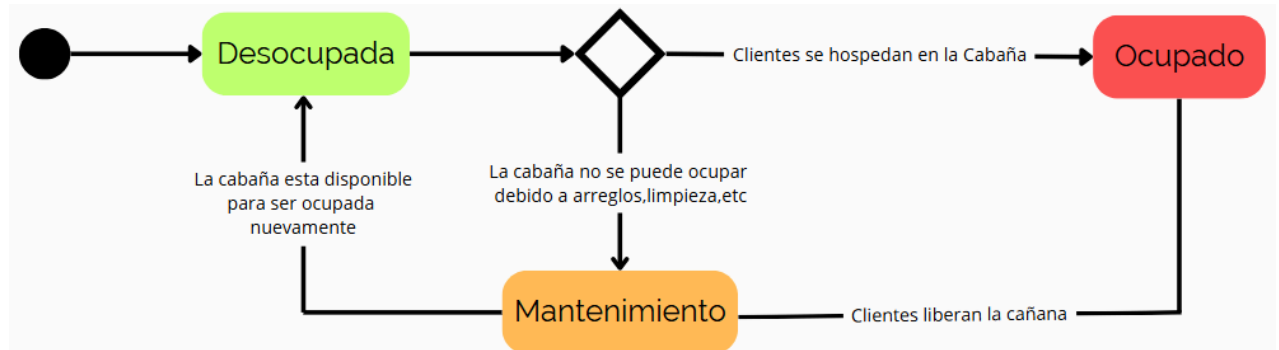
### Diagrama de estados de Cabañas:

Las cabañas pueden encontrarse en distintos estados operativos, lo que permite gestionar su disponibilidad de manera precisa. Por defecto, una cabaña inicia en estado

"Desocupada", lo que indica que no se encuentra actualmente en uso y está habilitada para recibir huéspedes. Desde este estado, puede transicionar a dos situaciones distintas según las necesidades operativas.

Una de ellas es el estado "En mantenimiento", que se utiliza cuando la cabaña no está disponible temporalmente, ya sea por tareas de limpieza, reparaciones u otras razones que impidan su uso inmediato. Una vez que las condiciones necesarias para su funcionamiento se restablecen, debe volver al estado "Desocupada", quedando nuevamente disponible para ser reservada.

La otra transición posible desde el estado "Desocupada" es hacia el estado "Ocupada", lo que indica que un cliente se encuentra actualmente alojado en la cabaña. Al finalizar el período de hospedaje, es decir, en la fecha de check-out, y una vez que el cliente haya desocupado la unidad, la cabaña pasa al estado "Mantenimiento", indicando que debe ser reacondicionada para que pueda recibir al siguiente huésped. Una vez el trabajo de limpieza haya sido terminado, la cabaña está lista para ser hospedada nuevamente.



#### Diagrama de Estado de Carpas:

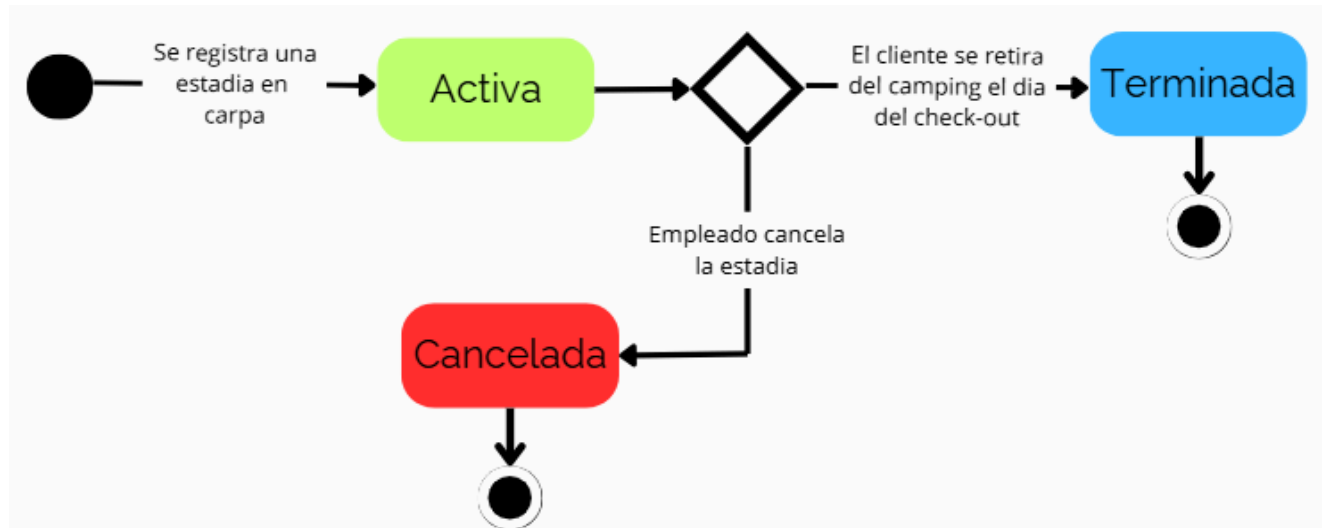
En el caso de estadías en carpas, el proceso comienza cuando un cliente recurre al camping para hospedarse en una parcela. En ese momento, el personal del establecimiento procede a registrar los datos del cliente e informar el monto correspondiente según la cantidad de personas y días de estadía solicitados. Una vez completado este procedimiento y confirmado el pago, el sistema registra la estadía en estado "Activa", lo que indica que el huésped se encuentra actualmente alojado en el camping y ocupando la parcela asignada.

Cuando se alcanza la fecha de check-out y el cliente se retira efectivamente del establecimiento, la estadía se actualiza al estado "Terminada", indicando que el período



de alojamiento ha concluido.

Cabe destacar que el sistema contempla también la posibilidad de cancelar la estadía en cualquier momento previo a su finalización, ya sea por solicitud del cliente o por razones administrativas, dejando constancia de esta acción en el historial correspondiente.



#### Estados de los empleados:

En lo que respecta a la gestión del personal, los empleados pueden encontrarse en dos estados operativos: "Activo" e "Inactivo". Un empleado se considera activo cuando se encuentra realizando tareas laborales dentro del camping durante la semana, ya sea en tareas operativas, administrativas o de mantenimiento. Este estado implica que el empleado está disponible y asignado a funciones dentro del establecimiento.

Por otro lado, un empleado pasa al estado inactivo cuando, por algún motivo, no está desempeñando actividades laborales en el camping durante un período determinado o indefinido. Este estado está contemplado para cubrir situaciones como vacaciones, licencias médicas, ausencias justificadas u otras circunstancias que impliquen una pausa temporal en su vínculo operativo.

La transición entre los estados activo e inactivo es bidireccional, lo que permite adaptar dinámicamente la situación de cada trabajador según su disponibilidad y situación personal, sin necesidad de eliminar su registro del sistema.

#### **Arquitectura**

La arquitectura utilizada para la creación del sistema de gestión de Camping Arroyito se basa en una arquitectura multicapa, un enfoque ampliamente adoptado en el

desarrollo de software debido a su capacidad para dividir responsabilidades de manera clara y ordenada. Esta arquitectura permite una mejor organización, mantenimiento y escalabilidad del sistema, ya que cada capa tiene una función bien definida y es independiente de las demás en la medida de lo posible.

En este caso, el sistema se organiza en tres capas principales: la capa de presentación (frontend), la capa de lógica de negocio (backend) y la capa de datos (base de datos). A continuación, se detallan las características y roles de cada una de estas capas.

### Capa de Presentación (Frontend)

La capa de presentación es la interfaz que conecta a los usuarios finales con el sistema. Su principal objetivo es proporcionar una experiencia de usuario intuitiva, eficiente y atractiva para que los diferentes tipos de usuarios, es decir, administradores y empleados que tengan o no experiencia utilizando este tipo de programas, puedan interactuar de manera sencilla con el sistema.

Tecnologías utilizadas:

- React: El sistema utiliza React como framework principal para construir interfaces de usuario dinámicas y modernas. React permite desarrollar componentes reutilizables y gestionar el estado de la aplicación de manera eficiente.
- Material-UI (MUI): Librería de componentes que aporta diseño profesional y adaptabilidad (responsive).
- CSS y librerías de diseño: Se emplean estilos personalizados y librerías como Material-UI o Bootstrap para garantizar que la interfaz sea visualmente atractiva y responsive, adaptándose a diferentes dispositivos, como computadoras, tablets o smartphones.

Módulos importados

- Hooks de React: Son hooks de React que permiten gestionar el estado local y manejar efectos secundarios dentro de los componentes funcionales.
  - useState: se utiliza para definir y actualizar variables de estado en componentes funcionales y para manejar datos que cambian dinámicamente

- `useEffect`: ejecuta efectos secundarios, como solicitudes HTTP, sincronización de datos o configuración de listeners, al montar o actualizar un componente.
- `useContext` y `createContext`: Se usan principalmente para pasar datos y funciones a través de los componentes sin necesidad de props. Esto es útil para compartir el estado global entre componentes sin tener que elevar el estado a un componente superior.
- `useReducer`:
- `react-router-dom`: Proporcionados por la librería React Router, son esenciales para manejar la navegación entre páginas y rutas dinámicas.
  - `useNavigate`: permite programáticamente redirigir al usuario a diferentes rutas de la aplicación, es muy útil para flujos basados en acciones, como redirigir después de enviar un formulario.
  - `useParams`: recupera parámetros dinámicos de la URL, como IDs, lo cual facilita la creación de páginas específicas, como detalles de una cabaña basada en su ID.
  - `Route`: Se utiliza para definir las rutas dentro de la aplicación y asociarlas con componentes específicos que se deben renderizar cuando la URL coincida. Este componente se utiliza directamente en la interfaz de usuario para estructurar la navegación entre vistas.
  - `Routes`: Es el contenedor de rutas. Se utiliza para agrupar varias rutas dentro de un componente `Route` y es la forma en que se configuran las rutas en una aplicación.
  - `Outlet`: Es un componente utilizado para renderizar los elementos secundarios dentro de una ruta anidada. Permite que las rutas anidadas se muestren dentro de un contenedor. Es parte del sistema de enrutamiento y, al igual que `Route` y `Routes`, se usa para gestionar cómo se organiza la navegación entre vistas, lo que lo coloca en la capa de presentación.
  - `Link`: Este componente se utiliza para crear enlaces de navegación entre diferentes rutas. Permite a los usuarios navegar a otras vistas de la aplicación sin recargar la página.

- DataGrid: Componente proporcionado por la librería Material-UI (MUI) para mostrar datos tabulares interactivos y personalizables. Tiene funcionalidades avanzadas como filtración de datos, paginación y ordenamiento.
- react-date-range: es una librería de React que proporciona componentes para seleccionar rangos de fechas en la interfaz de usuario. Permite crear calendarios interactivos con múltiples opciones de personalización
- FullCalendar: Es una librería de React que permite la visualización de calendarios interactivos. Se usa en la interfaz de usuario para mostrar eventos, como reservas de cabañas o disponibilidad. También se importaron dos complementos de FullCalendar tales como dayGridPlugin que agrega la vista de cuadrícula al calendario y InteractionPlus que permite la interacción del usuario con el calendario (utilizada para que el usuario pueda elegir los días de estadia haciendo click en el calendario)
- Recharts: biblioteca de gráficos componibles basada en componentes de React y D3.js. Contiene API que permiten añadir fácilmente 11 tipos de gráficos diferentes y altamente configurables a aplicaciones React.
- @mui/x-charts: es parte del ecosistema de Material-UI (ahora MUI) y proporciona componentes listos para usar para crear gráficos de líneas, barras, áreas, entre otros
- @mui/material: Proporciona una amplia variedad de componentes UI, como botones, cajas de texto, tarjetas, tablas, menús, y más, todos diseñados siguiendo las pautas de Material Design.
- @mui/icons-material: parte de Material-UI (MUI). Esta librería proporciona una amplia variedad de iconos que cumplen con las pautas de diseño de Google Material Design.

Responsabilidades de la capa de presentación:

1. Interacción con el usuario: Ofrecer formularios, gráficos, tablas y paneles interactivos que permitan a los usuarios realizar diversas acciones, como:
  - Poder administrar cabañas o carpas de forma sencilla mediante tablas que muestran toda la información pertinente.
  - Registrar y gestionar empleados o recursos.
  - Visualizar y analizar datos en tiempo real mediante gráficos, mapas y

tablas dinámicas.

2. Validación de datos: Realizar validaciones iniciales de los datos ingresados por los usuarios (como formatos de fecha o campos obligatorios) antes de enviarlos al backend.
3. Comunicación con el backend: Enviar y recibir datos a través de solicitudes HTTP a una API RESTful, asegurándose de que las interacciones entre el cliente y el servidor sean fluidas y rápidas.

Ventajas de esta capa:

- Mejora la experiencia del usuario al proporcionar una interfaz moderna y fácil de usar.
- Permite la actualización de contenidos en tiempo real, lo que es esencial para funcionalidades como la disponibilidad de cabañas o carpas.
- Desacopla la lógica de presentación de la lógica de negocio, facilitando futuras actualizaciones o rediseños de la interfaz.

### Capa de Lógica de Negocio (Backend)

La capa de lógica de negocio es el núcleo del sistema, ya que gestiona todas las operaciones y procesos que garantizan el correcto funcionamiento del camping. Esta capa se encarga de implementar las reglas de negocio, validar datos y coordinar las interacciones entre la capa de presentación y la capa de datos.

Tecnologías utilizadas:

- Node.js: Es el entorno de ejecución elegido por su alta eficiencia y escalabilidad. Node.js permite manejar múltiples solicitudes concurrentes, lo cual es crucial para sistemas que deben procesar datos en tiempo real.
- Express: Este framework se utiliza para crear y gestionar la API RESTful del sistema. Express simplifica la creación de rutas y middleware necesarios para manejar solicitudes HTTP.

## Módulos Importados:

- **axios:** Biblioteca de JavaScript que se utiliza para hacer solicitudes HTTP desde el navegador o Node.js útil para consumir APIs RESTful.
- **useFetch:** Hook personalizado para simplificar las llamadas a APIs en componentes React.
- **cookie-parser:** Permite analizar cookies en las solicitudes entrantes. Se utiliza para manejar autenticación y sesiones, es decir que está relacionado con la gestión de usuarios y acceso al sistema.
- **bcryptjs:** Este módulo se utiliza para encriptar contraseñas y verificar contraseñas en procesos de autenticación. En el contexto de la lógica de negocio, su función es garantizar que las contraseñas de los usuarios se almacenen de forma segura (cifradas) y que las contraseñas ingresadas por los usuarios sean comparadas con las almacenadas de manera segura.
- **jsonwebtoken:** Este módulo se utiliza para generar y verificar tokens JWT (JSON Web Tokens) en el proceso de autenticación y autorización. Los tokens son utilizados para mantener la sesión del usuario, asegurando que las solicitudes a ciertas rutas solo sean accesibles por usuarios autenticados y autorizados.
- **nodemailer:** Es un módulo de Node.js que permite el envío de correos electrónicos. Se utiliza para enviar confirmaciones de reservas o avisos de cancelación de reservas.
- **cron:** Módulo utilizado para ejecutar tareas programadas automáticamente en intervalos definidos (por ejemplo, cada día o cada hora). En este sistema, se utiliza para tareas como verificar el vencimiento de reservas o enviar recordatorios por correo electrónico sin necesidad de intervención manual.

## Responsabilidades de la capa de lógica de negocio:

1. **Gestión de reservas:** Verifica la disponibilidad de cabañas y carpas, registra nuevas reservas y calcula los costos asociados.
2. **Gestión de estados:** Actualiza los estados de las cabañas, carpas y reservas (por ejemplo, "disponible", "ocupada", "mantenimiento") y asegura la sincronización con los datos almacenados.
3. **Procesamiento de pagos:** Implementa procesos de validación de pagos y genera

registros contables, asegurando la integridad financiera del sistema.

4. Historial y reportes: Genera reportes de historial, como estados de cabañas y movimientos contables, para facilitar la toma de decisiones por parte de los administradores.
5. Seguridad y autenticación: Gestiona el acceso al sistema mediante autenticación de usuarios y control de roles, asegurando que cada usuario solo pueda realizar acciones permitidas.

Ventajas de esta capa:

- Centraliza y organiza la lógica del negocio, garantizando consistencia en las operaciones.
- Permite un fácil mantenimiento, ya que las reglas de negocio están separadas de la presentación y la base de datos.
- Escalable, lo que permite la integración futura con nuevos servicios.

#### Capa de Datos (Base de Datos)

La capa de datos se encarga de almacenar, organizar y gestionar la información del sistema. Es una capa crítica, ya que asegura la persistencia de los datos y permite su recuperación eficiente.

Tecnologías utilizadas:

- MongoDB: Se ha elegido MongoDB como el sistema de gestión de bases de datos debido a su flexibilidad y escalabilidad. Al ser una base de datos NoSQL, MongoDB permite almacenar datos en documentos JSON, lo cual es ideal para manejar estructuras de datos complejas y dinámicas.

Módulos importados:

- mongoose: Proporciona una conexión con MongoDB y permite trabajar con un ORM para manejar esquemas y datos.

Ventajas de esta capa:

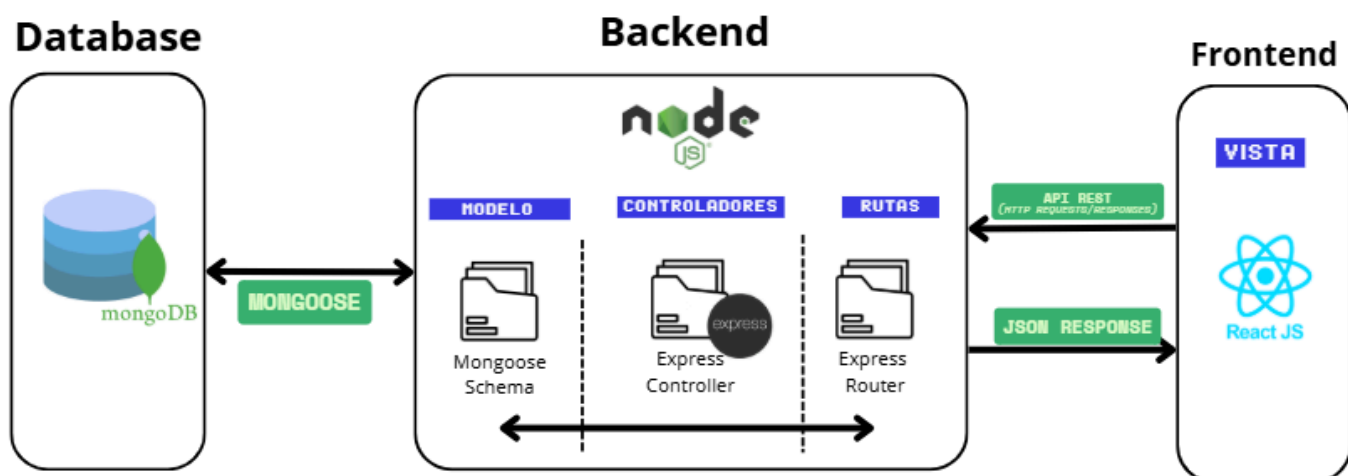
- Flexibilidad en el manejo de datos estructurados y no estructurados.

- Altamente escalable, lo que permite manejar grandes volúmenes de datos a medida que crece el sistema.
- Integración eficiente con Node.js mediante bibliotecas como Mongoose, que facilita la validación y consulta de datos.

#### Otros módulos importados utilizados:

- dotenv: Carga variables de entorno desde un archivo .env, facilitando la configuración del sistema de manera segura. Garantiza que datos sensibles como la URI de la base de datos o claves secretas no estén en el código fuente.
- cors: Habilita la comunicación entre el frontend y el backend, permitiendo solicitudes desde orígenes diferentes.
- multer: Maneja la carga de archivos, en el caso de este proyecto, la carga de imágenes

Esta arquitectura multicapa implementada en el sistema de Camping Arroyito proporciona una solución robusta, escalable y eficiente. Este diseño permite la integración de tecnologías modernas, optimizando tanto la experiencia del usuario como la administración del negocio. La separación de responsabilidades entre las capas asegura un mantenimiento más sencillo y la posibilidad de futuras expansiones sin alterar significativamente la estructura del sistema.





## Patrones de diseño

En el desarrollo del sistema de gestión web para Camping Arroyito, se han aplicado patrones de diseño reconocidos que permiten estructurar de manera eficiente el código, garantizar la mantenibilidad del sistema y facilitar futuras expansiones. Entre los patrones utilizados destacan los siguientes:

### Modelo-Vista-Controlador (MVC)

El patrón MVC es uno de los pilares en la organización de la arquitectura del sistema, especialmente en la separación de responsabilidades entre el frontend y el backend. Este patrón divide la aplicación en tres componentes principales:

- **Modelo:** Es la representación de la lógica de datos y las reglas de negocio del sistema. En este proyecto, el modelo se implementa principalmente en el backend utilizando la herramienta Mongoose, una biblioteca de modelado de datos para la MongoDB en Node.js.

En el modelo se declaran todos los atributos, tipos de datos, relaciones entre entidades y validaciones requeridas para cada colección de la base de datos. Como ejemplo de la implementación de un modelo en este proyecto podemos utilizar el creado para el registro de empleados dentro del sistema:

```
import mongoose from 'mongoose';
const { Schema } = mongoose;

const EmployeeSchema = new Schema({
  user: {
    type: Schema.Types.ObjectId,
    ref: 'User',
    required: true
  },
  job: {
    type: String,
    required: true
  },
  base_salary: {
    type: Number,
    required: true
  },
  start_date: {
    type: Date,
```

```

    required: true
  },
  photo: {
    type: String,
  },
  status: {
    type: mongoose.Schema.Types.ObjectId,
    ref: 'EmployeeStatus',
  }
});

export default mongoose.model('Employee', EmployeeSchema);

```

Aquí se puede ver cómo cada campo representa un atributo específico del empleado, como su puesto de trabajo, su salario, la fecha en la que inició a trabajar y una foto. También existen referencias a otros modelos como el de usuario representando la relación entre un empleado y su cuenta de usuario en el sistema o la de status que contiene los estados por los cuales puede pasar el empleado (activo o inactivo). También se detallan validaciones, en donde hay campos que no son obligatorios como por ejemplo el de cargar una foto y otros sí lo son, es decir, que si un campo obligatorio está vacío no se permitirá la creación de un nuevo registro de empleado, asegurando que no se puedan cargar registros incompletos o inválidos.

- **Controlador:** Actúa como un intermediario entre el modelo y la vista. Se encarga de manejar las solicitudes del usuario que provienen desde el frontend y las traduce en acciones que actualizan el modelo o modifican la vista.

En este proyecto los controladores se ven implementados en el backend con Node.js/Express, se encargan de procesar la información y datos recibidos, llaman a los modelos para realizar consultas, crear nuevos registros o modificar la base de datos, como por ejemplo la creación de una nueva reserva o la actualización del estado de una cabaña y luego devuelven una respuesta al cliente.

Podemos tomar como ejemplo algunos de los controladores creados para manejar las solicitudes de Cabañas:

```

export const createLodge = async (req, res, next) => {
  try {
    const status = await Status.findOne({ status: "desocupada" });

```

```

    const newLodge = new Lodges({
      ...req.body,
      state: status._id,
      photos: req.file ? req.file.path : req.body.photos
    });
    const savedLodge = await newLodge.save();
    res.status(200).json(savedLodge);
  } catch (err) {
    console.error("Error al crear lodge:", err);
    next(err);
  }
};

export const updateLodge = async (req, res, next) => {
  try {
    const updatedLodge = await Lodges.findByIdAndUpdate(req.params.id, { $set: req.body }, { new: true });
    res.status(200).json(updatedLodge);
  } catch (err) {
    next(err);
  }
};

export const deleteLodge = async (req, res, next) => {
  try {
    await Lodges.findByIdAndDelete(req.params.id);
    res.status(200).json("Cabaña eliminada");
  } catch (err) {
    next(err);
  }
};

export const getLodge = async (req, res, next) => {
  try {
    const lodge = await Lodges.findById(req.params.id).populate('state');
    res.status(200).json(lodge);
  } catch (err) {
    next(err);
  }
};

```

Para poder utilizar estos controladores, se definieron rutas.. Las rutas asocian un método HTTP (GET, POST, PUT, DELETE) y un endpoint de la API con el controlador correspondiente, indicando así qué acción debe ejecutarse ante cada solicitud.

```

import express from "express";
import { createLodge, deleteLodge, getLodge, uploadPhotos, getLodges,
  getLodgeAvailability, getOccupiedPositions, updateLodge,
  setMantenimientoWithComment, eliminarComment } from "../controllers/lodge.js";

```

```

const router = express.Router();

//CREATE
router.post("/", createLodge);
router.put("/:id", updateLodge);
router.put("/comment/:id", setMantenimientoWithComment);

//DELETE
router.delete("/:id", deleteLodge);
//GET
router.get("/:id", getLodge);
router.get("/occupiedPositions", getOccupiedPositions);
router.get("/occupied-dates/:id", getLodgeAvailability);

//GET ALL
router.get("/", getLodges);

export default router

```

Como se puede ver, cada uno de estos controladores se encarga de manejar un pedido específico que provenga desde los usuarios a través de las vistas. Estos pedidos pueden venir en forma de post que se encarga de crear nuevas entradas en la base de datos, de put que permiten modificar los registros ya existentes, de delete que eliminan registros de la base de datos y de get que devuelve información específica de los registros.

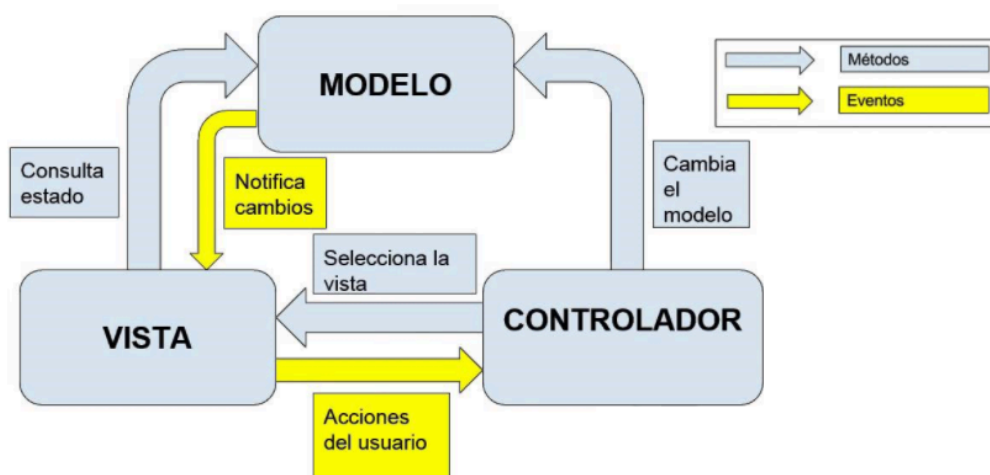
- Vista: Es la interfaz gráfica del sistema que permite la interacción de los usuarios finales y la aplicación. La vista se encarga de presentar los datos proporcionados por el backend y recolectar los datos, información y acciones que el usuario realice en el sistema para enviarlas al servidor.

En este proyecto, la vista está desarrollada utilizando React.js, la cual es una biblioteca de JavaScript que facilita la creación de interfaces de usuario de forma modularizada, reactiva y eficiente.

En el sistema, las vistas se ven implementadas de diferentes formas tales como: Formularios, Dashboards, Listas y Tablas, Calendarios interactivos y Mapas de ocupación.

La interfaz en este proyecto tiene la ventaja de que se actualiza automáticamente

cada vez que cambian los datos gracias al sistema de gestión del estado de React en donde se utiliza el `useState` y el `useEffect`. También se utilizan llamadas HTTP mediante `Axios` para obtener, enviar o actualizar datos de forma asíncrona asegurando que la vista siempre refleje el estado más actualizado de la base de datos. Otra ventaja de esta implementación es que hay diversos componentes que pudieron ser extraídos y convertidos en componentes reutilizables, es decir, aquellos componentes que se utilizarán en diversas páginas tales como la sidebar, calendario, el mapa, entre otros se crean una sola vez y luego se los llama en la vista en la que se desee usar, permitiendo que sea más fácil escalar y mantener el proyecto.



### Singleton

El patrón Singleton se utiliza para garantizar que ciertas clases o componentes tengan solo una instancia en toda la aplicación. Esto es particularmente útil para gestionar recursos compartidos o configuraciones globales.

En el sistema, la conexión con la base de datos se gestiona mediante una clase Singleton que asegura que solo exista una instancia de conexión activa, reduciendo el consumo innecesario de recursos.

Esta instancia única de `Axios` se exporta como default y se utiliza en todos los módulos que necesiten comunicarse con el backend. Así cada vez que se quiera hacer una solicitud HTTP se utilizará siempre la misma instancia:

```
import axios from "axios";
const axiosInstance = axios.create({
  baseURL: process.env.REACT_APP_API_URL ||
  "https://creek-project.onrender.com",
  withCredentials: true, //para las cookies
});
export default axiosInstance;
```

Luego cuando un módulo necesite hacer una petición al backend podrá hacerlo simplemente utilizando el `axiosInstance` y declarando cuál es la ruta a la que quiere acceder, tal como se hace aquí:

```
const fetchAccountingRecords = async () => {
  try {
    const response = await axiosInstance.get("/accounting"); //axiosIntance
    setAccountingData(response.data);
  } catch (error) {
    console.error("Error al obtener registros de pago:", error);
  }
};
```

Este patrón se implementó para evitar tener que declarar siempre la URL completa del backend cada vez que se deseaba hacer una solicitud a la base de datos, también permite que si en el futuro se necesite cambiar la URL del servidor o si se quieren agregar tokens de autenticación se deberá modificar esta única instancia en vez todos los archivos que usen `axios`.

## Observer

Observer es un patrón de diseño de comportamiento que permite definir un mecanismo de suscripción para notificar a varios objetos sobre cualquier evento que le suceda al objeto que están observando.

El objeto que tiene un estado interesante suele denominarse sujeto. El resto de los objetos que quieren conocer los cambios en el estado del sujeto, se denominan suscriptores.

Para lograr implementar este patrón en el sistema lo que se hizo fue implementar la clase `EventObserver` que contará con los métodos “`subscribe`”, “`unsubscribe`” y “`notify`” los cuales permitirán gestionar la lista de suscriptores y garantizar que estos sean notificados de forma automática cuando haya un cambio relevante en el objeto sujeto.

```

class EventObserver {
  constructor() {
    this.events = {};
  }
  subscribe(eventName, fn) {
    if (!this.events[eventName]) {
      this.events[eventName] = [];
    }
    this.events[eventName].push(fn);
  }
  unsubscribe(eventName, fn) {
    if (!this.events[eventName]) return;
    this.events[eventName] = this.events[eventName].filter((f) => f !== fn);
  }

  notify(eventName, data) {
    if (!this.events[eventName]) return;
    this.events[eventName].forEach((fn) => fn(data));
  }
}
const globalObserver = new EventObserver();
export default globalObserver;

```

Luego se creó la clase Subscriptions, la cual cuenta con las acciones que realizarán los suscriptores de un determinado objeto luego de que reciban la notificación mediante el notify()

```

import globalObserver from "../observer";
import axiosInstance from "../axios/axiosInstance";

//actualizar el estado de la cabaña a completada
globalObserver.subscribe("reserva-completada", async ({ booking }) => {
  const lodgeId = booking.lodge._id;
  const bookingId = booking._id;
  try {
    // Actualizar estado actual
    await axiosInstance.put(`/lodges/${lodgeId}`, {
      state: "668f303c70711974d54762cf",
    });
    // Crear nuevo estado en historial
    await axiosInstance.post(`/lodge_x_status/`, {
      lodge: lodgeId,
      status: "668f303c70711974d54762cf",
      booking: bookingId,
    });
    console.log("Estado de la cabaña actualizado exitosamente.");
  } catch (error) {
    console.error("Error actualizando estado de cabaña:", error);
  }
});

```

```

    }
  });

  // Suscripción para actualizar los gráficos
  globalObserver.subscribe("reserva-completada", ({ booking }) => {
    const event = new CustomEvent("updateCharts", { detail: { booking } });
    window.dispatchEvent(event);
  });

```

Como ejemplo del uso de este patrón en el sistema podemos utilizar la acción de actualización del estado de las cabañas, en donde cuando una reserva cambia su estado a "terminada", el módulo de reservas notifica automáticamente al módulo de cabañas para actualizar su estado a "disponible". Esto también actualiza los gráficos estadísticos que muestran la ocupación y reservas del camping el cual es usado por los administradores y el mapa que muestra el estado actual de las entidades del camping. Esta notificación indirecta entre módulos representa una aplicación del patrón Observer a nivel de backend, donde el módulo de reservas actúa como sujeto observado y el módulo de estado de cabañas como suscriptor, manteniendo así la sincronización del sistema sin acoplamientos rígidos

También se utilizó el patrón observer para las siguientes funcionalidades:

| Acción   | Suscriptor                                   | Sujeto                              |
|--|--|-------------------------------------|
| Cambiar la vista de los gráficos según día/mes/año | Gráficos estadísticos                        | Vista del usuario                   |
| Agregar una nueva tarea para los empleados         | Calendario interactivo                       | Empleado                            |
| Cambiar el estado de reservas                      | Datatable de reservas y gráfico de ocupación | Datos de reservas (bookings, tents) |

La combinación de patrones de diseño como MVC, Singleton y Observer aseguran que el sistema de gestión de Camping Arroyito sea modular, escalable y fácil de mantener. Estos patrones no solo optimizan el desarrollo actual, sino que también preparan el sistema para futuras expansiones, como la incorporación de nuevas funcionalidades o la integración con otros servicios.



# Implementación

La implementación de un sistema de software eficaz requiere dividir su desarrollo en módulos o componentes lógicos. Los módulos son bloques funcionales autónomos que, al integrarse, conforman el sistema completo. Cada módulo se diseña para cubrir una funcionalidad específica, como la autenticación de usuarios, la gestión de inventarios o la generación de reportes. Esta estructura modular es esencial para garantizar que el sistema sea organizado, escalable y fácil de mantener.

La utilización de módulos es una práctica estándar en el desarrollo de software porque mejora significativamente la organización del proyecto. Un sistema grande y complejo puede volverse inmanejable si no se divide en partes más pequeñas y manejables. Los módulos permiten que el desarrollo sea más ágil, fácil y promueve un enfoque iterativo donde las funcionalidades se construyen y ensamblan gradualmente.

## Estructura modular del proyecto

En el caso del sistema de gestión web para Camping Arroyito, los módulos principales son los siguientes:

### 1. Módulo de Autenticación

El módulo de autenticación es fundamental para garantizar la seguridad y el control de acceso al sistema. Este módulo gestiona el inicio de sesión, el registro de usuarios y la asignación de permisos según el rol asignado (administrador, empleado o cliente).

Responsabilidades principales:

- Inicio de sesión: Verificar las credenciales del usuario para garantizar que solo las personas autorizadas puedan acceder al sistema.
- Registro de usuarios: Permitir que nuevos usuarios, como usuarios, empleados o administradores, se registren en el sistema con un rol definido.
- Gestión de roles: Asignar permisos específicos según el rol del usuario:
  - Administrador: Acceso completo al sistema, incluyendo módulos de reservas, cabañas, contabilidad y gestión de empleados.

- Empleado: Acceso limitado a funcionalidades relacionadas con tareas operativas y gestión de reservas.
- Cliente: Acceso indirecto mediante la plataforma de reservas en línea.
- Autenticación segura: Implementar medidas como el cifrado de contraseñas y autenticación basada en tokens (JWT) para proteger las sesiones de los usuarios.

Interacción con otros módulos:

- Se comunica con el módulo de gestión de reservas para permitir el acceso a los datos privados de los clientes.
- Se integra con el módulo de gestión de empleados para registrar nuevos usuarios y asignarles roles.
- Colabora con el módulo de contabilidad al controlar quién tiene acceso a la información financiera.

La implementación de este módulo mejora la seguridad del sistema al restringir el acceso según los roles y también facilita la gestión de usuarios y garantiza que cada uno tenga acceso solo a las funcionalidades que necesita.

## 2. Módulo de Gestión de Reservas

El módulo de gestión de reservas es uno de los componentes centrales del sistema, ya que gestiona todas las operaciones relacionadas con la creación, modificación y cancelación de reservas, tanto para cabañas como para parcelas destinadas a carpas.

Responsabilidades principales:

- Creación de reservas: Permitir a los usuarios registrar nuevas reservas especificando la fecha, la cantidad de huéspedes que se hospedarán en el camping, la cabaña a reservar y los datos del cliente.
- Verificación de disponibilidad: Consultar el estado actual de las cabañas/parcelas para evitar conflictos en las reservas.
- Cancelación de reservas: Registrar y gestionar cancelaciones, liberando las unidades asociadas para nuevas reservas.
- Gestión de estados: Actualizar el estado de las reservas (pendiente, confirmada, cancelada) y sincronizar esta información con otros módulos.

Interacción con otros módulos:

- Trabaja con el módulo de gestión de cabañas para asegurarse de que las reservas estén asociadas correctamente a las unidades disponibles.
- Envía datos al módulo de contabilidad para registrar los ingresos generados por las reservas.
- Verifica los permisos de los usuarios a través del módulo de autenticación antes de permitir la creación de reservas.

Este módulo optimiza la experiencia del cliente al garantizar un proceso de reservas fluido y sin errores y también permite a los empleados visualizar y gestionar fácilmente la ocupación del camping.

### 3. Módulo de Gestión de Cabañas

Este módulo administra todo lo relacionado con el estado y la disponibilidad de las cabañas del camping. Es crucial garantizar que las unidades estén en las condiciones adecuadas para recibir a los clientes.

Responsabilidades principales:

- Gestión de estados: Actualizar el estado de las cabañas como disponible, ocupada o en mantenimiento.
- Asociación con reservas: Vincular cada cabaña con una reserva específica y actualizar su estado en función de la ocupación.
- Historial de estados: Mantener un registro del historial de cambios en el estado de cada cabaña para fines de auditoría y mantenimiento.
- Control de mantenimiento: Permitir que los empleados marquen una cabaña como en mantenimiento y registren las tareas realizadas.
- Creación de nuevas cabañas: Los administradores podrán cambiar un espacio asignado a una parcela por uno asignado a una cabaña, por si el camping decide agregar nuevas cabañas en un futuro.

Interacción con otros módulos:

- Recibe información del módulo de gestión de reservas para actualizar el estado de las cabañas.

Este módulo ayuda a reducir los errores en la asignación de cabañas y mejora la experiencia del cliente. También permite llevar un mejor control y seguimiento de los estados de las cabañas.

#### 4. Módulo de Contabilidad

El módulo de contabilidad centraliza todas las operaciones financieras del camping, asegurando un control eficiente de los ingresos, egresos y costos asociados a las reservas.

Responsabilidades principales:

- Registro de ingresos: Documentar los pagos realizados por las reservas y otros servicios adicionales.
- Registro de egresos: Controlar gastos relacionados con mantenimiento, personal, suministros y todo otro gasto que los administradores creen pertinentes y estén relacionados a la gestión del camping.
- Cálculo de costos: Automatizar el cálculo de tarifas que deberán pagar los clientes que vayan a hospedarse al camping.
- Visualización de datos: Muestra gráficos interactivos e intuitivos para facilitar la toma de decisiones en el camping

Interacción con otros módulos:

- Recibe datos del módulo de gestión de reservas para registrar ingresos por cada reserva.

Los beneficios que trae implementar este módulo son que facilita la gestión financiera del camping y ofrece datos claros y precisos para la toma de decisiones estratégicas.

## 5. Módulo de Gestión de Empleados

Este módulo se encarga de administrar toda la información relacionada con los empleados del camping, desde su contratación hasta la asignación de tareas.

Responsabilidades principales:

- Registro de empleados: Almacenar datos personales, laborales y de contacto.
- Gestión de roles: Asignar permisos y roles específicos según las responsabilidades de cada empleado.
- Asignación de tareas: Asignar tareas operativas como limpieza de cabañas o reparaciones.

Interacción con otros módulos:

- Se comunica con el módulo de autenticación para crear y gestionar cuentas de empleados.

Beneficios:

- Mejora la organización interna del camping.
- Aumenta la eficiencia al asignar tareas de manera clara y precisa.

La integración de estos módulos asegura que el sistema sea robusto, eficiente y capaz de satisfacer las necesidades operativas y estratégicas del Camping Arroyito. Cada módulo está diseñado para trabajar de manera autónoma, pero su interacción garantiza una experiencia fluida tanto para los empleados como para los clientes.

## Modo de trabajo

El proceso de desarrollo y despliegue del sistema de gestión web para Camping Arroyito se basó en un flujo de trabajo organizado y estructurado, que permitió gestionar de manera eficiente los cambios y actualizaciones del proyecto. A continuación, se detalla el paso a paso del flujo de trabajo utilizado:

### Paso 1: Desarrollo

Para iniciar con el desarrollo se comenzó por instalar Yarn como gestor de paquetes debido a su rapidez y eficiencia en comparación con npm. En el directorio raíz del proyecto, se ejecutó `yarn init` para inicializar un archivo `package.json` que gestiona las dependencias del proyecto. Luego se procedió con la instalación de Express como framework backend ejecutando `yarn add express`.

Otras dependencias importantes incluyeron:

- `mongoose`: Para manejar la conexión con MongoDB y la creación de modelos de datos (*`yarn add mongoose`*).
- `dotenv`: Para gestionar variables de entorno de forma segura (*`yarn add dotenv`*).
- `cors`: Para permitir la comunicación entre el frontend y backend (*`yarn add cors`*).
- `jsonwebtoken` y `bcryptjs`: Para manejar la autenticación y el cifrado de contraseñas (*`yarn add jsonwebtoken bcryptjs`*).
- `axios`: Para realizar solicitudes HTTP desde el frontend (*`yarn add axios`*).

Lo siguiente que se hizo fue la configuración de la base de datos, lo primero fue crear la conexión con MongoDB Atlas y se creó una base de datos llamada *`camping_arroyito`* donde se garantizará la persistencia de datos en un entorno remoto.

Se creó un archivo principal *`index.js`* que inicializa el servidor utilizando Express y conecta la base de datos, allí también se configuraron las rutas iniciales para manejar endpoints de usuarios, reservas, cabañas, carpas, etc.

Luego el servidor se ejecutó con `yarn start` durante el desarrollo, utilizando nodemon para reiniciar automáticamente tras cambios en el código.

Para comenzar con el desarrollo del frontend se configuró un proyecto React para el frontend ejecutando *`yarn create react-app client`*. Se instalaron múltiples librerías útiles

como Material-UI para el diseño de interfaces o *react-router-dom* para manejar la navegación entre páginas.

Cada vez que se trabajaba en el frontend, se ejecutaba yarn start para visualizar los cambios en tiempo real.

## Paso 2: Pruebas.

Cada vez que se agregaba una nueva funcionalidad esta se probaba de forma local primero, donde se hacían pruebas manuales para verificar que los resultados o eventos obtenidos sean los esperados, también se implementaron pruebas automáticas sobre funcionalidades básicas del camping para asegurar que estas funcionen correctamente. Las pruebas se ejecutaron tanto para el frontend como para el backend, asegurando que la comunicación entre ambos fuera fluida y libre de errores.

Para asegurar que los datos de prueba no “infecten” la base de datos con los datos reales del negocio se procedió a crear una base de datos de prueba la cual contendrá todos los datos que se utilizan durante esta etapa, lo cual permitirá realizar inserciones, modificaciones y eliminaciones de datos sin afectar la información de producción.

Las pruebas que se realizaron fueron las siguientes:

### Pruebas manuales:

Estas pruebas consistían en la validación del flujo de trabajo esperado en distintas secciones del sistema, tales como:

- Registro y autenticación de usuarios.
- Creación, edición y eliminación de reservas.
- Gestión de estados de las cabañas y carpas.
- Cálculo automático de montos a pagar según la cantidad de días y personas.
- Pruebas de interfaz de usuario para asegurar que la experiencia fuera intuitiva y sin errores visuales.

Pruebas automatizadas:

Para mejorar la calidad y estabilidad del código, se implementaron pruebas automatizadas sobre funcionalidades clave del sistema. Se utilizaron herramientas como Jest y Supertest para realizar pruebas en el backend, enfocándose en:

- Verificación de los endpoints de la API (GET, POST, PUT, DELETE).
- Autenticación y validación de credenciales con jsonwebtoken.
- Pruebas de integridad en la base de datos utilizando MongoDB en un entorno de prueba.

Pruebas de integración:

Se verificó que la comunicación entre el frontend y el backend fuera fluida y libre de errores, realizando pruebas con herramientas como Insomnia para simular peticiones HTTP y validar las respuestas del servidor.

A continuación se colocaran algunos ejemplos de las peticiones HTTP realizadas en Insomnia:

The screenshot shows an Insomnia API client interface. The top bar indicates a POST request to `http://localhost:3001/register` with a status of **200 OK**, a response time of 211 ms, and a body size of 382 B. The left pane shows the JSON body of the request, and the right pane shows the JSON body of the response.

```
1 {
2   "_id": "67da2ceb07058620d91e3cre",
3   "email": "pruebaemail@example.com",
4   "password": "securepassword",
5   "first_name": "Test",
6   "last_name": "User",
7   "phone": "123456789",
8   "dni": "24880200",
9   "birthday": "01-01-1999",
10  "occupation": "Tester"
11 }
12
```

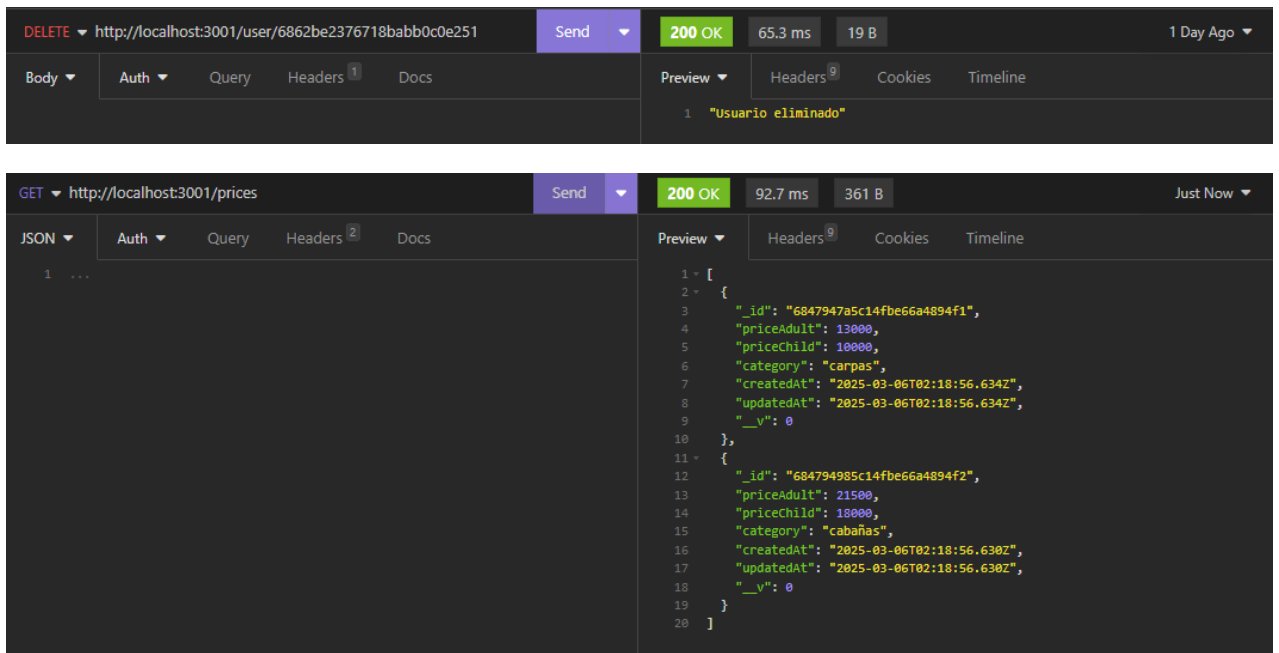
```
1 {
2   "email": "pruebaemail@example.com",
3   "password":
4     "$2a$10$w6VGL6Uv80y51KEHdwJJg.limi0zpsd5s3cV6jub2XGA0eIwc6ys5",
5   "first_name": "Test",
6   "last_name": "User",
7   "dni": "24880200",
8   "phone": "123456789",
9   "birthday": "01-01-1999",
10  "occupation": "Tester",
11  "isAdmin": false,
12  "isEmployee": false,
13  "_id": "6862be2376718babb0e251",
14  "createdAt": "2025-06-30T16:41:07.005Z",
15  "updatedAt": "2025-06-30T16:41:07.005Z",
16  "__v": 0
17 }
```

The screenshot shows an Insomnia API client interface. The top bar indicates a PUT request to `http://localhost:3001/user/682be492427c566d515638c9` with a status of **200 OK**, a response time of 62.8 ms, and a body size of 325 B. The left pane shows the JSON body of the request, and the right pane shows the JSON body of the response.

```
1 {
2   "email": "m@gmail.com"
3 }
```

```
1 {
2   "_id": "682be492427c566d515638c9",
3   "email": "m@gmail.com",
4   "password":
5     "$2a$10$fg5dskjo/cgoq4w9FznJeeNv9jzfPLYIhEPOKuaEKehsBH.uvmK3q",
6   "first_name": "Mica",
7   "last_name": "Prueba",
8   "dni": "42444190",
9   "phone": "380532845",
10  "birthday": "2000-03-01",
11  "occupation": "Tester",
12  "isAdmin": false,
13  "isEmployee": true,
14  "updatedAt": "2025-07-01T00:13:10.658Z"
15 }
```

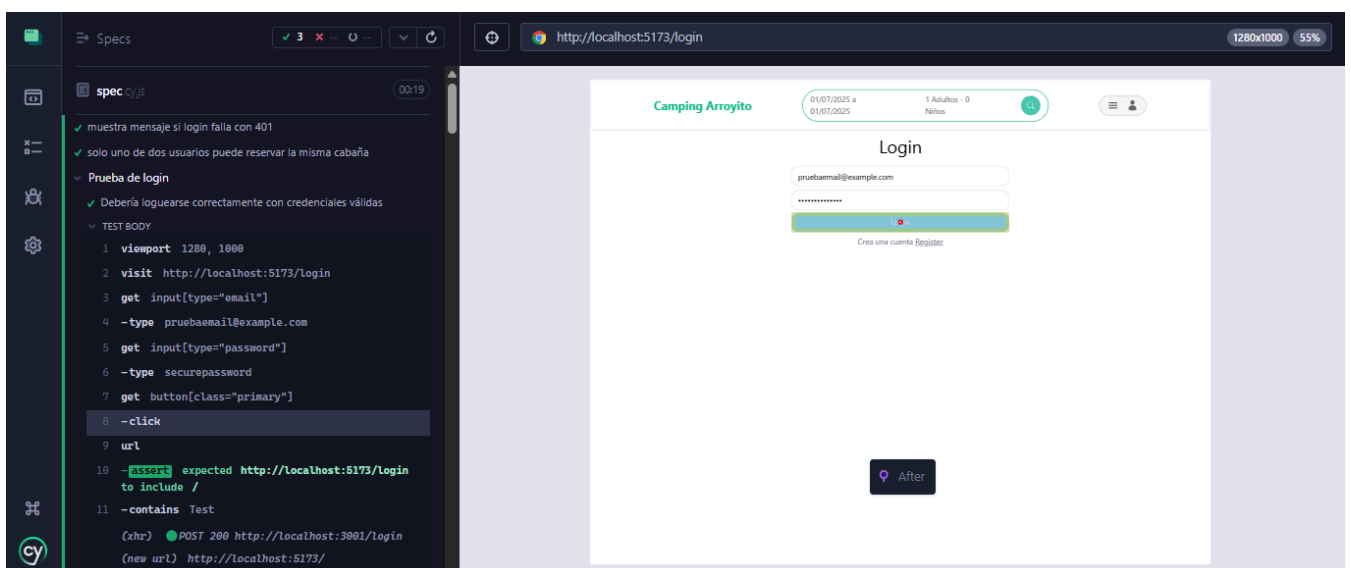




En cuanto al frontend, se llevaron a cabo pruebas a través de Cypress que permitieron validar el correcto funcionamiento de los componentes visuales y su interacción con el usuario. Se comprobó que los botones, formularios y elementos dinámicos respondieran adecuadamente a las acciones realizadas, garantizando así una experiencia de usuario clara, consistente y sin fallos de comportamiento

Además, se probaron escenarios como qué sucede si hay fallos en la autenticación o simular múltiples usuarios accediendo e interactuando con la página web.

Estas pruebas se hicieron mayormente sobre la página web de Camping Arroyito ya que se prevé que allí es donde haya mayor tráfico de usuarios



### Paso 3: Control de Versiones en GitHub

Una vez que las pruebas locales fueron satisfactorias, el código actualizado se confirmó (commit) y se envió (push) al repositorio remoto de GitHub.

Se utilizaron ramas específicas para organizar el desarrollo, como:

- master: Rama principal con las versiones estables del sistema.
- production: Rama para integrar nuevas funcionalidades que están listas para ser testeadas.

### Paso 4: Despliegue

Cuando llegó el momento de decidir sobre la plataforma para el despliegue en la nube, se analizaron dos alternativas principales: Heroku y Vercel. Ambas eran opciones viables que ofrecían características atractivas, como:

- Conexión y agilidad: Ambas plataformas permiten manejar aplicaciones modernas con tiempos de respuesta rápidos.
- Escalabilidad: Las dos opciones cuentan con planes flexibles que permiten escalar la aplicación en caso de aumento de usuarios o de requerimientos.
- Facilidad de integración: Tanto Heroku como Vercel ofrecen herramientas y documentación amplia para facilitar el proceso de despliegue.

Inicialmente, se optó por Heroku, dado que es una plataforma consolidada y cuenta con una comunidad activa que proporciona soporte, tutoriales y recursos útiles. Además, su popularidad lo convierte en una opción confiable para proyectos pequeños y medianos. Sin embargo, durante la configuración del despliegue, surgió un obstáculo significativo: Heroku requiere un pago inicial de \$25 para activar su servicio en modo de producción y mantener la aplicación funcionando de manera continua. Este costo, aunque razonable en muchos casos, motivó la búsqueda de alternativas más económicas o gratuitas, especialmente para una etapa inicial del proyecto.

Teniendo en cuenta el pago requerido por Heroku, se decidió intentar el despliegue en Vercel, una plataforma conocida por su enfoque en aplicaciones web modernas y su modelo de precios más flexible. En Vercel, el pago es opcional y únicamente se requiere

si el usuario desea mejorar aspectos como la velocidad, el almacenamiento o escalar el sistema a necesidades más avanzadas. Esto lo convirtió en una opción ideal para un proyecto en sus primeras etapas. Lograr el despliegue en vercel no resultó tan complicado debido a que Vercel cuenta con muchas facilidades cuando se trata de aplicaciones creadas usando React, tales como reconocimiento automático de frameworks donde Vercel detecta de forma automática el framework utilizado en el desarrollo del proyecto y también cuenta con integración con GitHub, ya que Vercel permite conectar directamente el repositorio de GitHub. Cada vez que se realiza un push a la rama principal (master), la plataforma ejecuta un nuevo despliegue automáticamente, asegurando que la aplicación siempre esté actualizada, y también ofrece muchas facilidades para manejar múltiples proyectos

Para el despliegue se crearon 2 proyectos en Vercel, el primer proyecto desplegará únicamente el página web de camping arroyito y el otro proyecto desplegará el sistema Creek

En el caso de la página web, la configuración de despliegue consiste en elegir el framework utilizando durante el desarrollo, en este caso fue vite, y esto automáticamente establece los Build Commands e Install Commands adecuados para este framework, luego se estableció el Root Directory, es decir, el lugar en donde el proyecto se encuentra alojado, indicando a Vercel que deberá sacar el código que se encuentra alojado en el repositorio de Github llamado "Creek\_Project" desplegar únicamente el código que se encuentre en la carpeta client.

Para el caso del sistema Creek, los pasos tomados fueron muy parecidos, la única diferencia es que el framework no era el de Vite, si no que se eligió la opción "Other" y se establecieron manualmente los Build e Install Commands y luego se configuró el Root Directory para apuntar a la carpeta admin dentro del repositorio, asegurando que solo se despliegue el código correspondiente al sistema administrativo.

Una vez creados los proyectos, ambos requerían conexión a la base de datos alojada en MongoDB Atlas. Afortunadamente, Vercel cuenta con un add-on oficial de MongoDB que facilita esta integración. Los pasos realizados fueron:

1. Instalación del add-on: Esto permitió que Vercel se comunicara con MongoDB Atlas.

2. Autorización: Se otorgaron permisos a MongoDB para acceder a los proyectos desplegados en Vercel.
3. Creación de variables de entorno: Al instalar el add-on, Vercel generó automáticamente las siguientes variables necesarias para la conexión:
  - MONGODB\_URI: Es la cadena de conexión principal que Vercel utiliza para conectar la aplicación a la base de datos MongoDB alojada en MongoDB Atlas. Esta URI contiene toda la información necesaria para establecer la conexión con la base de datos.
  - DATA\_API\_URL: Es la URL del endpoint de la Data API de MongoDB Atlas, que permite interactuar con la base de datos a través de HTTP en lugar de usar un cliente tradicional como Mongoose.
  - DATA\_API\_KEY: Es la clave de acceso necesaria para autenticar las solicitudes realizadas a la Data API de MongoDB Atlas. Funciona como un token de seguridad que asegura que solo las aplicaciones autorizadas puedan interactuar con la base de datos mediante el API.

Para garantizar que el backend estuviera siempre disponible sin necesidad de reiniciarlo manualmente cada vez que se quiera utilizar el sistema, se optó por desplegarlo en Render, una plataforma moderna orientada al alojamiento de aplicaciones backend

La decisión de usar Render para el backend y Vercel para el frontend se basó en una cuestión de especialización y funcionamiento: Vercel está más optimizado para aplicaciones frontend, especialmente aquellas creadas con frameworks como React, Next.js o Vite. Sin embargo, no está pensado para mantener servidores Express corriendo permanentemente, ya que su modelo serverless no ofrece soporte completo para APIs con estado o servicios que requieren ejecución continua.

Por otro lado, Render sí permite alojar servidores Express con ejecución permanente, ofrece servicios tipo "Web Service", ideales para mantener el backend corriendo en segundo plano de manera constante.

De esta manera, se garantiza una arquitectura sólida y escalable, donde cada parte del sistema se despliega en la plataforma que mejor se adapta a su naturaleza.

Pasos realizados en Render:

1. Creación del servicio: Se creó un nuevo servicio de tipo "Web Service" en Render.

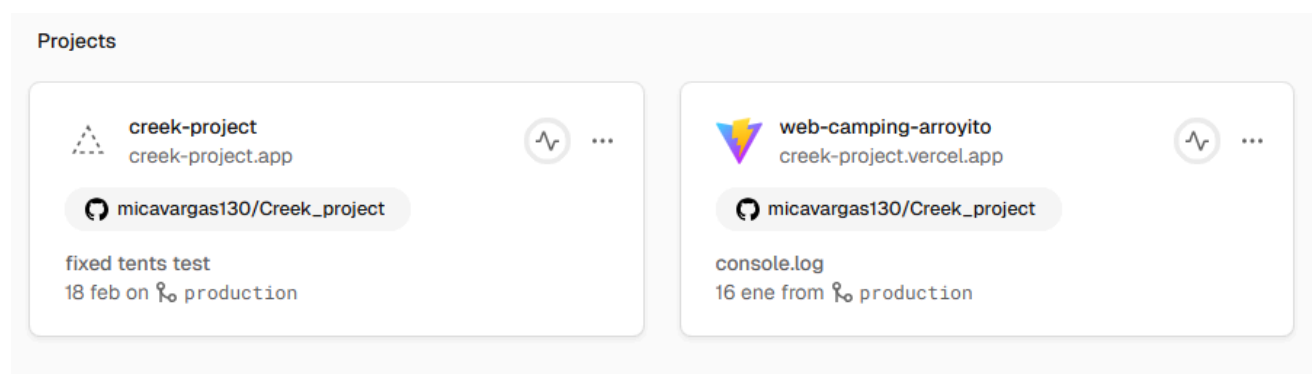
2. Comando de ejecución: Se indicó el comando `yarn start` para que Render supiera cómo iniciar el servidor Express.
3. Variables de entorno: Se configuraron las variables necesarias para la conexión a la base de datos MongoDB Atlas
4. Despliegue automático desde GitHub: Se vinculó el repositorio para que cada vez que se hiciera un push a la rama principal, Render ejecutará automáticamente un nuevo despliegue del backend.

Gracias a esta configuración, el servidor backend quedó siempre activo, funcional y disponible para responder a las solicitudes del sistema, sin interrupciones.

Configuración de CORS: Las URLs proporcionadas por Vercel para las páginas desplegadas se añadieron a la lista de orígenes permitidos en las configuraciones de CORS del backend. Esto era necesario para permitir el acceso a los datos desde estas direcciones.

Una vez completada la configuración, ambos proyectos quedaron accesibles mediante las URLs proporcionadas por Vercel. El sistema ahora cuenta con las siguientes características:

- Actualización automática: Cualquier cambio en el código que sea subido a la rama principal en GitHub desencadena un nuevo despliegue.
- Separación de proyectos: Esto permite mantener la página web y el sistema administrativo gestionados de manera independiente.
- Conexión estable con la base de datos: La integración con MongoDB Atlas asegura que los datos estén siempre disponibles.



## Paso 5: Automatización del Despliegue (CI/CD)

Se configuró un flujo de trabajo de integración continua (CI/CD) utilizando GitHub Actions.

Cada vez que se realiza un push a la rama Production, el flujo ejecuta los siguientes pasos:

1. Instalar las dependencias del proyecto.
2. Ejecuta pruebas automatizadas para garantizar la estabilidad del código.
3. Hacer el despliegue automático a Render para que persista la base de datos
4. Realiza el despliegue automático en la aplicación de Vercel configurada para pruebas, en donde el despliegue recibe el nombre de “Preview”, esto significa que es un despliegue que se hace sobre una url de prueba de corta duración, esto se hace con el motivo de probar la aplicación en un entorno remoto y asegurarse de que funcione correctamente antes de desplegarlo a la url utilizada por los usuarios.

|                      |                          |   |                          |     |
|----------------------|--------------------------|---|--------------------------|-----|
| DkiwEobXa<br>Preview | ● Ready<br>21s (13h ago) | 🔗 production<br>> _dacac98 bd routes update | 13h ago by micavargas130 | ... |
|----------------------|--------------------------|---|--------------------------|-----|

Este proceso asegura que cada nueva funcionalidad sea desplegada y probada en un entorno remoto antes de fusionarse con la rama principal.

← CI/CD Pipeline

🟢 fixed tents test #9

Summary

Jobs

🟢 Ejecutar Tests

🟢 Deploy a Vercel

Run details

🕒 Usage

📄 Workflow file

Triggered via push last month

micavargas130 pushed

 → 2901188 

production

Status

Success

Total duration

12m 26s

Artifacts

—

github-actions.yml

on: push

🟢 Ejecutar Tests

 1m 12s → 

🟢 Deploy a Vercel

 10m 54s

Summary

Jobs

Ejecutar Tests

Deploy a Vercel

Run details

Usage

Workflow file

Ejecutar Tests

succeeded last month in 1m 12s

Search logs

> Set up job1s

> Clonar Repositorio2s

> Configurar Nodejs2s

> Instalar Dependencias35s

> Configurar Variables de Entorno0s

> Ejecutar Tests31s

> Post Configurar Nodejs0s

> Post Clonar Repositorio0s

> Complete job0s

Summary

Jobs

Ejecutar Tests

Deploy a Render y Vercel

Run details

Usage

Workflow file

Deploy a Render y Vercel

succeeded 3 minutes ago in 3m 59s

Search logs

> Set up job1s

> Clonar Repositorio2s

> Deploy a Render2s

> Instalar Vercel CLI13s

> Deploy a entorno correcto en Vercel3m 39s

> Post Clonar Repositorio0s

> Complete job0s

## Paso 6: Consolidación en la Rama Principal

Una vez que la aplicación se considera correctamente desplegada y funcionando, se pasa a hacer un pull request de “production” a “master” para que los cambios se fusionen. Github Actions vuelve a ejecutar el CI/CD pipelines para asegurar que todo funcione bien:

All checks have passed

6 successful checks

✓ CI/CD Pipeline / Deploy a Vercel (pull\_request)

Successful in 13s

...

✓ CI/CD Pipeline / Deploy a Vercel (push)

Successful in 1m

...

✓ CI/CD Pipeline / Ejecutar Tests (pull\_request)

Successful in 58s

...

✓ CI/CD Pipeline / Ejecutar Tests (push)

Successful in 54s

...

✓ Vercel – web-camping-arroyito

Deployment has completed

...

✓ Vercel Preview Comments

No unresolved feedback

...

✓ No conflicts with base branch

Merging can be performed automatically.

Merge pull request

You can also merge this with the command line. [View command line instructions.](#)

La aplicación final se ve desplegada en Vercel y se puede acceder a ella a través del siguiente link: "<https://creek-project-ruby.vercel.app/>", el cual es que usaran los usuarios para acceder al sistema y los clientes que deseen acceder a la página web podrán hacerlo a través de "<https://creek-project.vercel.app/>". Además se puede acceder al código del sistema a través de

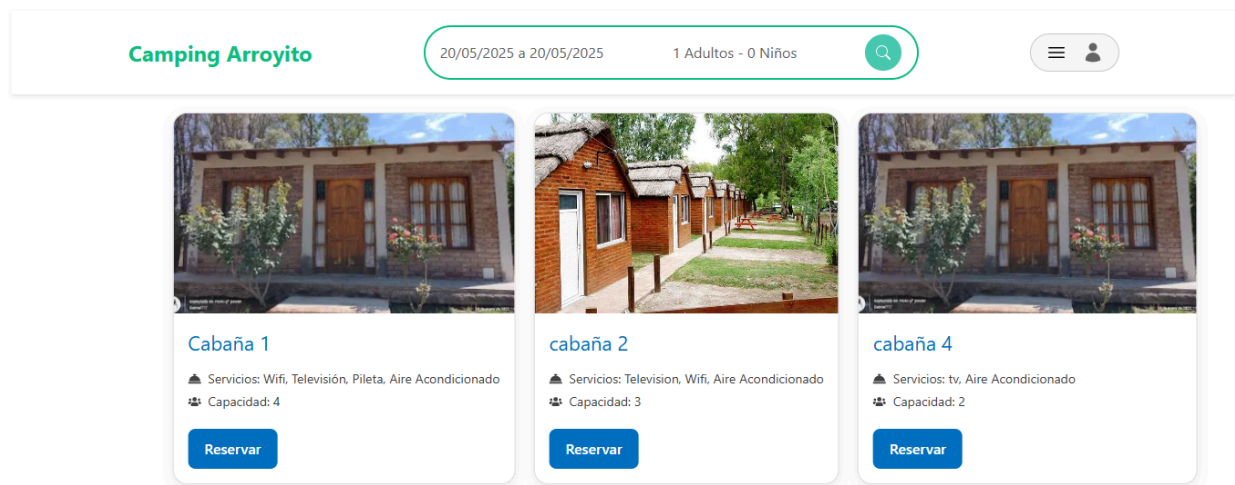


## Solución generada

### Página web del camping

Se trata de una página web sencilla y fácil de utilizar, a través de la cual los usuarios tendrán la posibilidad de realizar reservas en línea, permitiendo a los clientes seleccionar fechas y cabañas disponibles de forma sencilla. También cuenta con un apartado de información detallada sobre las cabañas, carpas y servicios adicionales, proporcionando imágenes, precios y descripciones claras.

Los usuarios deberán estar registrados para poder hacer la reserva por lo cual cuenta con un formulario que deberán llenar para registrarse y luego ingresar utilizando el login. Esto también permite que los usuarios puedan ver las reservas que realizaron y tendrán la posibilidad de cancelarla si así lo desean.



## Login

your@email.com

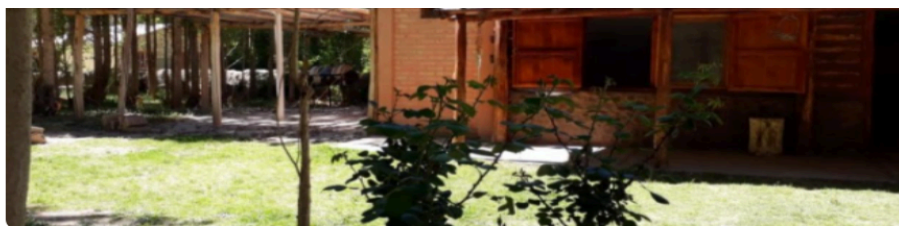
password

Login

Crea una cuenta [Register](#)

El usuario podrá realizar una reserva desde la página web, solo deberá seleccionar la cabaña deseada entre las opciones disponibles, según las fechas indicadas y la cantidad de personas detalladas.

Luego de completar la reserva, ésta quedará registrada como “pendiente”, a la espera del pago de una señal. Inmediatamente, el sistema enviará un correo electrónico a la dirección registrada por el usuario, con todos los detalles necesarios: monto de la señal, datos de la cuenta para efectuar la transferencia, whatsapp a donde enviar el comprobante del pago y una advertencia de que la reserva se cancelará automáticamente si el pago no se realiza dentro de los próximos 3 días. Además, el usuario podrá consultar en todo momento el estado de su reserva desde la sección “Mis Reservas” dentro de su perfil en la página web.



## Detalles

Cabaña para 6 personas, con cochera y asador privado. Cuenta con 2 piezas con cama matrimonial y una pieza con 2 camas de una plaza

## Servicios

📺 Tv

🌬️ aire acondicionado

📶 wifi

## Información importante

## Normas de la casa

Check-in a partir de las 10:00 AM

Se permiten mascotas

Se ofrece desayuno gratuito a partir de las 8:30 a 11:30 AM

Acceso gratuito a la pileta

## Cancelación

Cancelación gratuita hasta el día antes del check-in.

## Fechas de reserva

20/05/2025 to 20/05/2025

## Cantidad de personas

1 Adultos - 0 Niños

## Detalle de precio

Precio por adulto: \$13000

Precio por niño: \$10000

Cantidad de días: 1

Total a pagar: \$13000

Reserva ahora!

cabaña 2

\$ 43000

Check-in: 26/5/2025

Check-out: 28/5/2025

Estado: Activa

Cancelación gratuita hasta 27/5/2025

Incluye impuestos y tasas

Cancelar

cabaña 4

\$ 43000

Check-in: 27/5/2025

Check-out: 29/5/2025

Estado: Pendiente: Esperando transferencia

Incluye impuestos y tasas

Cancelar

## Sistema Creek

### Login

El sistema cuenta con un login simple cuya única función es asegurarse que únicamente los usuarios que cuentan con permisos de administrador o de empleado puedan acceder al sistema. En este inicio también se puede hacer el cambio de la clave temporaria que se le otorga a un nuevo usuario empleado.

### Home

Una vez validadas las credenciales del usuario, se lo redirigirá a la página inicial del sistema, la cual es un dashboard general con toda la información necesaria para poder tener un vistazo rápido de la situación actual del camping.

Lo primero que el usuario puede visualizar es un calendario que cuenta con sus tareas asignadas. Los administradores pueden asignar a los empleados tareas y también pueden asignarse tareas a sí mismos, esto servirá para qué cada empleado pueda conocer sus tareas del día de forma más eficiente y rápida

Creek

Bienvenido Gabriela Montes

LISTS

Reservas

Cabañas

Carpas

ADMINISTRACION

Estadísticas

Contabilidad

Empleados

Logout

Tus tareas asignadas

+ Agregar tarea

Today Back Next

June 2025

Month Week Day Agenda

| Mon            | Tue               | Wed            | Thu            | Fri            | Sat | Sun |
|----------------|-------------------|----------------|----------------|----------------|-----|-----|
| 26             | 27                | 28             | 29             | 30             | 31  | 01  |
| 02             | 03                | 04             | 05             | 06             | 07  | 08  |
| Ordenar piezas | Ordenar piezas    | Ordenar piezas | Ordenar piezas | Ordenar piezas |     |     |
| 09             | 10                | 11             | 12             | 13             | 14  | 15  |
|                | Limpiar Recepcion |                |                |                |     |     |
| 16             | 17                | 18             | 19             | 20             | 21  | 22  |
| 23             | 24                | 25             | 26             | 27             | 28  | 29  |
| 30             | 01                | 02             | 03             | 04             | 05  | 06  |

También se podrá visualizar un mapa interactivo que mostrará la ubicación general del camping y de cada una de sus instalaciones. Las cabañas o parcelas ocupadas estarán marcadas en color rojo, lo que permitirá identificar rápidamente su disponibilidad. Además, se indicarán claramente las ubicaciones de los asadores, el comedor, los estacionamientos y la pileta, brindando una mejor orientación sobre la distribución de las distintas áreas dentro del predio.

Junto al mapa se incluirán widgets informativos que mostrarán en tiempo real la cantidad de cabañas y parcelas ocupadas, así como el porcentaje total de ocupación del camping, facilitando un acceso rápido y visual a esta información clave para la gestión diaria.

Por último, se contará con un apartado de notificaciones que alertará al personal cada vez que se registre una nueva reserva, cuando llegue la fecha de check-out de una cabaña o carpa, y ,en caso de que aún no se haya enviado el comprobante de seña, se mostrará un recordatorio indicando que la reserva será cancelada automáticamente en las próximas 24 horas si no se recibe el comprobante, lo cual permitirá que el empleado o administrador pueda contactar al usuario para recordarle el envío correspondiente.



## Reservas

Esta sección permite visualizar todas las reservas registradas, tanto aquellas realizadas a través del sitio web como las ingresadas manualmente por administradores o empleados. Cada reserva cuenta con un botón “Info”, que al hacer clic despliega información detallada sobre el cliente, la cabaña reservada y los pagos asociados.

Al momento de ser creada, cada reserva comenzará con el estado “Pendiente”, hasta que se confirme la recepción del pago de la seña. Una vez verificado dicho pago, el empleado podrá hacer clic en el botón “Confirmar”, lo que cambiará automáticamente el estado de la reserva a “Activa”.

Cuando el cliente se presente en el establecimiento en la fecha indicada (check-in), el empleado deberá presionar el botón “Cobrar”. Esto abrirá un pop-up donde se podrá seleccionar el estado del pago (completo o parcial). Una vez confirmada esta información, el sistema asignará automáticamente al cliente a la cabaña correspondiente y se generará una nueva entrada en el historial de pagos, reflejando el estado elegido.

Creek

LISTS

Reservas

Cabañas

Carpas

ADMINISTRACION

Estadísticas

Contabilidad

Empleados

Logout

Reservas

Historia

Nueva Reserva

| Cliente        | Nombre de Cabaña | Check-In  | Check-Out | Precio | Total People | Status    |  |
|----------------|------------------|-----------|-----------|--------|--------------|-----------|--|
| Ana Joy        | cabaña 2         | 31/5/2025 | 3/6/2025  | 183000 | 3            | Pendiente | <div>Confirmar</div> <div>Info</div> <div>Cancelar</div> |
| Micaela Vargas | cabaña 2         | 28/5/2025 | 30/5/2025 | 86000  | 2            | Pendiente | <div>Confirmar</div> <div>Info</div> <div>Cancelar</div> |
| Felix Kjelberg | cabaña 4         | 27/5/2025 | 29/5/2025 | 43000  | 1            | Activa    | <div>Cobrar \$</div> <div>Info</div> <div>Cancelar</div> |

Información del huésped

Ana Joy

Email: 1803112@ucc.edu.ar

Phone: 2342423432

DNI: 3242344

Occupation: Profesora

Birthday: 2000-06-23

Información de la Reserva

Nombre de la cabaña:cabaña 2

Descripción de la cabaña:Cabaña para 2 personas con hermosas vistas, cerca de la pileta

Fecha de Check-In:31/5/2025

Fecha de Check-Out:3/6/2025

Cantidad de Adultos:2

Cantidad de niños:1

Monto a pagar:183000

Monto de la seña:54900

Los empleados también tienen la posibilidad de cargar reservas manualmente. Para ello, deberán ingresar la cantidad de personas que se hospedarán y en qué fechas, tras lo cual el sistema mostrará las cabañas disponibles con capacidad suficiente. Una vez completados todos los datos del cliente, se procederá a guardar la reserva. Cuando se genera una reserva de esta forma, el sistema enviará automáticamente un

## Cabañas

También es posible cambiar el estado de una cabaña a “Mantenimiento”, lo que la inhabilita temporalmente para nuevas reservas. Al seleccionar este estado, el sistema solicitará ingresar un comentario explicativo con el motivo. Mientras la cabaña esté en mantenimiento, no podrá ser asignada a ningún huésped. Para volverla a habilitar, se debe hacer clic en el botón “Liberar”, lo que cambiará su estado a “Disponible”.

Creek

LISTS

Reservas

Cabañas

Carpas

ADMINISTRACION

Estadísticas

Contabilidad

Empleados

Logout

Cabañas

Añadir Cabaña

| Cabaña   | Capacidad | Estado        | Comentario |  |
|----------|-----------|---------------|------------|--|
| Cabaña 1 | 4         | ocupado       |            | <div>Liberar</div> <div>Info</div> <div>Modificar</div> <div>Historial</div> <div></div> |
| cabaña 2 | 3         | desocupada    |            | <div>Mantenimiento</div> <div>Modificar</div> <div>Historial</div>                       |
| Cabaña 3 | 5         | mantenimiento | Limpieza   | <div>Liberar</div> <div>Modificar</div> <div>Historial</div>                             |
| cabaña 4 | 2         | desocupada    |            | <div>Mantenimiento</div> <div>Modificar</div> <div>Historial</div>                       |
| Cabaña 5 | 6         | ocupado       |            | <div>Liberar</div> <div>Info</div> <div>Modificar</div> <div>Historial</div>             |
| cabaña 6 | 3         | desocupada    |            | <div>Mantenimiento</div> <div>Modificar</div> <div>Historial</div>                       |

Finalmente, está disponible la funcionalidad de historial, que permite visualizar todos los cambios de estado por los que ha pasado cada cabaña, ofreciendo un registro completo de su uso y mantenimiento.

Otra funcionalidad disponible será la de agregar una nueva cabaña la cual reemplazará a una entidad de parcela y permitirá que el administrador cree la nueva entidad cabaña al cargar imágenes e información que se publicarán en la página web para que los clientes puedan reservar la nueva cabaña creada.

### Mapa de ocupación

|  |  |  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|--|--|
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |

Seleccionar archivo

Ningún archivo seleccionado

Nombre de la Cabaña

Descripción

Capacidad

Servicios

Guardar


## Carpas

La sección Carpas permite gestionar las estadías en parcelas del camping. Se presenta una lista que muestra los datos de al menos un integrante del grupo familiar, incluyendo información clave para su identificación y contacto, así como la fecha de check-in y check-out, y la cantidad de personas que ocupan la parcela (con un máximo permitido de 6 personas por reserva). También se indica si la reserva tiene pagos pendientes.

Cada entrada cuenta con botones que permiten gestionar la estadía:

- “Finalizar”: marca la estadía como “Terminada”, liberando la parcela para futuras reservas.
- “Cancelar”: permite al personal cancelar la estadía en caso necesario, ya sea por decisión del cliente o por situaciones imprevistas.
- “Info”: muestra todos los datos relevantes del huésped, los contactos, el monto total a abonar, y el dinero entregado hasta el momento, brindando un resumen claro de la situación de la reserva.
- “Historia”: redirige al historial de estadías en carpas, donde se pueden consultar tanto hospedajes anteriores como reservas canceladas.

### Carpas

| + Añadir Carpa |          | Historia |             |                   |         |       |           |           |   |
|----------------|----------|----------|-------------|-------------------|---------|-------|-----------|-----------|---|
| Nombre         | Apellido | DNI      | Telefono    | Email             | Adultos | Niños | Check-In  | Check-Out |   |
| Federico       | Torres   | 17225... | 38046667... | fede.torres231... | 2       | 1     | 21/5/2025 | 24/5/2025 | <div><div></div><div>Finalizar</div><div>Info</div><div>Cancelar</div></div> |
| Norma          | Sanchez  | 20325... | 351234954   | norma_s@gm...     | 1       | 1     | 22/5/2025 | 26/5/2025 | <div><div>Finalizar</div><div>Info</div><div>Cancelar</div></div>   |

Para registrar una nueva estadía, el administrador debe hacer clic en el botón “Añadir Carpa”, que abrirá un formulario donde se ingresan los datos del cliente. A continuación, se selecciona la parcela que desea ocupar y se hace clic en “Registrar”. Como paso final, se debe indicar el estado del pago:

- Pago parcial: el empleado deberá registrar el monto entregado, y el sistema marcará la reserva como parcialmente abonada.
- Pago completo: la reserva será marcada como “Pagada” y quedará confirmada.





# Contabilidad

Agregar Ingreso/Egreso

Modificar precios

| Monto  | Tipo    | Estado    | Fecha            | Comentario      | Cliente        | Entidad  |  |
|--------|---------|-----------|------------------|-----------------|----------------|----------|--|
| 24600  | Ingreso | seña      | 19/7/2025, 22:10 |                 | Jorge Newman   | cabaña 8 | <div>Info</div> <div>Cancelar</div> <div>+</div>           |
| 10000  | Egreso  | pagado    | 19/7/2025, 22:08 | Proveedores     | N/A            | N/A      | <div>Cancelar</div> <div>+</div>                           |
| 200000 | Ingreso | parcial   | 18/7/2025, 22:53 |                 | Pedro Krois    | Carpa    | <div>Cancelar</div> <div>+</div> <div>Completar Pago</div> |
| 82000  | Ingreso | cancelado | 17/7/2025, 23:43 |                 | Teodoro Doe    | Cabaña 3 | <div>Info</div> <div>+</div>                               |
| 19800  | Ingreso | cancelado | 17/7/2025, 23:29 | Cliente canceló | Felix Kjelberg | cabaña 2 | <div>Info</div> <div>+</div>                               |
| 60000  | Ingreso | pagado    | 15/7/2025, 23:18 |                 | Micaela Vargas | Cabaña 1 | <div>Info</div> <div>Cancelar</div> <div>+</div>           |

Para cada ingreso asociado a una estadía, se incluye el botón “Info”, que redirige al empleado a una pantalla con todos los datos del cliente, detalles de la reserva y una tabla de pagos realizados. En esa tabla, junto a cada pago, se puede subir un comprobante de transferencia enviado por el cliente como respaldo del pago realizado. En esta página también podrá completarse el pago total de la estadía si es que aún se adeuda dinero.

Creek

LISTS

Reservas

Cabañas

Carpas

ADMINISTRACION

Estadísticas

Contabilidad

Empleados

Logout

Información del huésped

Federico Torres

Email: fede.torres2312@gmail.com

Phone: 3804666793

DNI: 17225868

Información de la carpa

Fecha de Check-In:21/5/2025

Fecha de Check-Out:24/5/2025

Cantidad de Adultos:2

Cantidad de niños:1

Monto total a pagar:108000

Historial de Pagos

| Fecha     | Monto    | Estado  | Comprobante |
|-----------|----------|---------|-------------|
| 22/5/2025 | \$100000 | parcial |             |

Completar Pago (\$8000)

Los ingresos o egresos que no estén vinculados a una estadía (por ejemplo, compras de insumos, sueldo de los empleados, gastos del camping, etc) pueden añadirse manualmente mediante el botón “Agregar ingreso/egreso”. Al hacer clic, se accede a un formulario donde el administrador debe seleccionar: el tipo de operación (ingreso o egreso), el monto y un comentario opcional. Cabe aclarar que los ingresos ingresados manualmente no pueden vincularse a clientes específicos, ya que no provienen de una reserva.

Añadir nuevo Ingreso/Egreso

Tipo: 

Seleccionar tipo

Amount

Monto

Comment

Comment

Send

Por último, el botón “Modificar Precio” redirige a la pantalla de tarifas actuales para adultos y niños, tanto en cabañas como en carpas. Desde allí, el administrador puede actualizar los precios, y los nuevos valores se aplicarán automáticamente a todas las reservas futuras. Las reservas ya registradas mantendrán los precios originales previos a la modificación. Se podrá visualizar una tabla con todos los cambios que se hicieron de los precios según la fecha y entidad para llevar un mejor control sobre ellos.

PRECIOS DE CABAÑAS

Precio Adulto  
22000  
Precio Niño  
19000

PRECIOS DE CARPAS

Precio Adulto  
13500  
Precio Niño  
11000

Edit

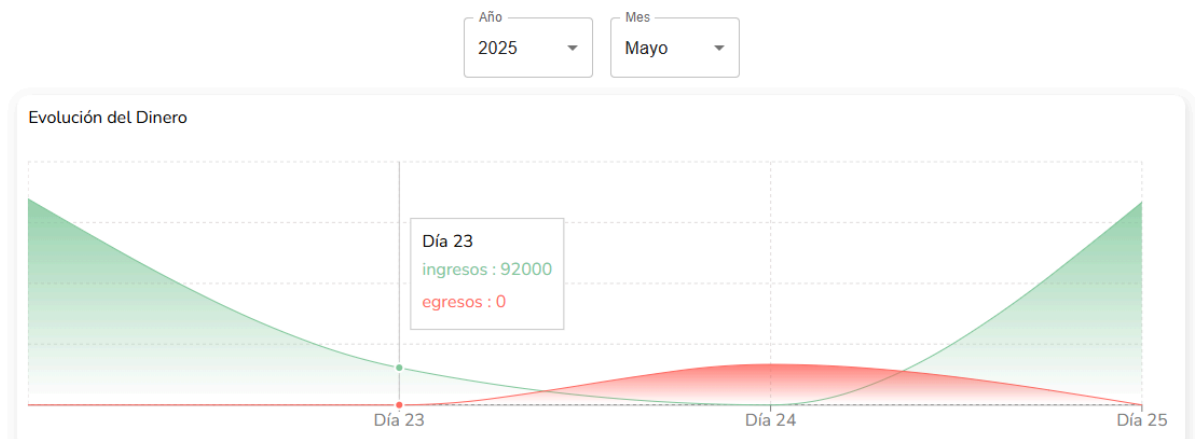
HISTORIAL DE PRECIOS

| FECHA     | CATEGORÍA | PRECIO ADULTO | PRECIO NIÑO |
|-----------|-----------|---------------|-------------|
| 24/5/2025 | carpas    | \$13500       | \$11000     |
| 24/5/2025 | cabañas   | \$22000       | \$19000     |
| 5/3/2025  | carpas    | \$13000       | \$10000     |
| 5/3/2025  | cabañas   | \$21500       | \$18000     |

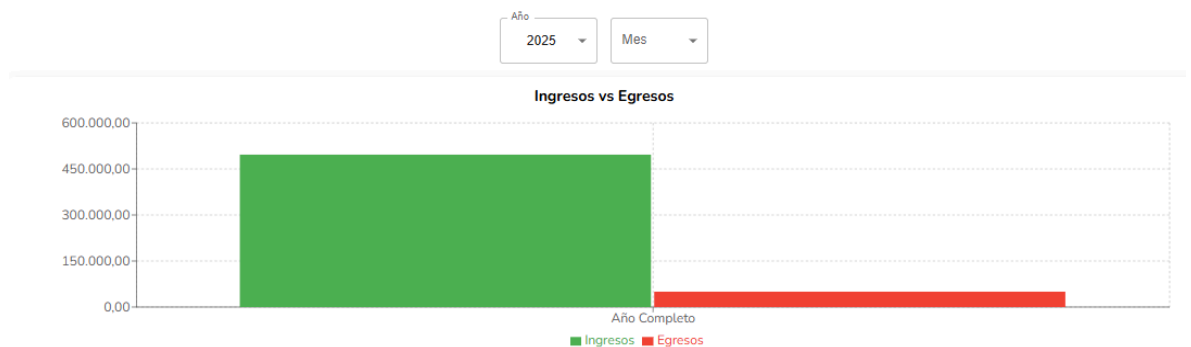
## Gráficos de contabilidad

La sección de contabilidad también contará con los siguientes gráficos que ayudarán a la toma de decisiones y control de ingresos y egresos dentro del camping.

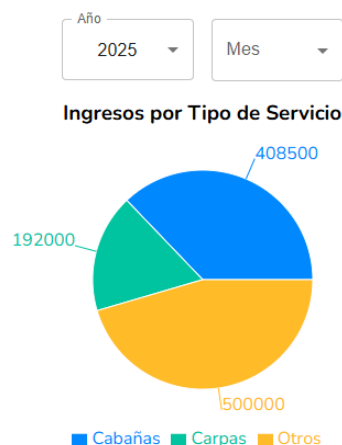
- Gráfico de la evolución del dinero: mostrará como el flujo de dinero en el camping se ve alterado a lo largo del tiempo, se permitirá al usuario filtrar según un mes específico o ver la evolución monetaria de todo el año.



- Gráfico de ingresos vs egresos: se trata de un gráfico de barras que permitirá a los administradores comparar la cantidad de ingresos y egresos según un mes o año específico.



- Gráfico de Ingresos por tipo de servicio: En esta sección se visualizan los servicios que más ingresos generaron para el camping, lo cual resulta especialmente útil para identificar cuáles son los más demandados según la época del año. La información se presenta dividida en tres categorías principales: reservas de cabañas, alojamiento en parcelas y otros ingresos, que abarcan todas aquellas entradas de dinero no relacionadas directamente con reservas.



- Widgets de contabilidad: brindan una visión rápida y actualizada del estado financiero del camping. Permiten consultar de forma inmediata: Ingresos totales, Gastos totales, Dinero disponible (balance actual). La información puede filtrarse por periodo, ya sea diario, mensual o anual.



## Empleados

Esta sección registra una lista con todos los empleados que hay en el camping, aquí se muestra una foto del empleado, su nombre, apellido y su estado el cual será Activo en caso de estar trabajando actualmente y Inactivo si es que no se encuentra trabajando en este momento.

FILTRAR EMPLEADOS

+ Añadir Empleado

| Foto  | Nombre   | Apellido | Estado   |                 |
|---|----------|----------|----------|-----------------|
|  | Saul     | Goodman  | Activo   | Ver información |
|  | Paula    | Perez    | Inactivo | Ver información |
|  | Jorge    | Newman   | Activo   | Ver información |
|  | Gabriela | Montes   | Activo   | Ver información |

Se podrá agregar a un nuevo empleado desde el botón “añadir empleado”, en donde el administrador deberá rellenar una planilla con la información relevante del nuevo empleado así como también añadir una foto y colocar un email válido. Este email será usado para que el nuevo empleado pueda acceder al sistema y también se le proveerá una contraseña temporal que se le pedirá cambiar en su primer ingreso al sistema.

Al hacer click en el botón “Ver información” el administrador será redirigido a la planilla de datos del empleado, la cual cuenta con todos sus datos personales y con datos relevantes para la administración, tales como puesto, estado, salario y la cantidad de permisos que tiene, estos datos podrán ser modificados por los usuarios administradores y podrán elevar el rol de un usuario empleado al de usuario administrador.

**Gabriela Montes**

Editar

Puesto:

Mucama

Estado:

Activo

Salario base:

500000

Permisos:

Administrador

Fecha de ingreso:

2025-04-14T00:00:00.000Z



**Información personal**

DNI:

39823423

Cumpleaños:

1998-10-10

Teléfono:

380942343

Email:

gabriela@gmail.com

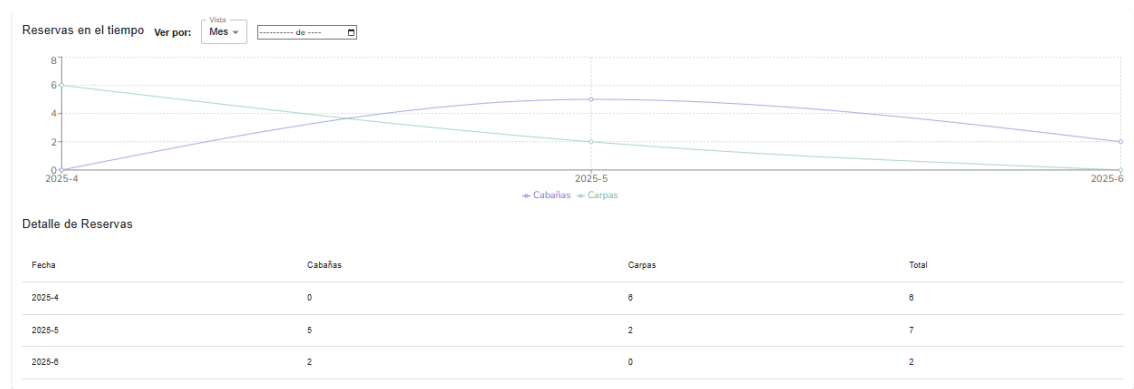
También estará disponible un calendario interactivo que permitirá a los administradores asignar tareas adicionales a los empleados, más allá de sus funciones habituales.

Una vez asignada, la tarea aparecerá automáticamente en el calendario del empleado dentro de la página principal (Home), de modo que sea lo primero que visualice al iniciar sesión en el sistema. Esto facilita la organización del trabajo diario y permite un seguimiento claro de las responsabilidades asignadas.

## Admin Portal

- Reservas en el tiempo: Se muestra la cantidad de reservas en cabañas y carpas se efectuaron durante un determinado tiempo y se puede ver la evolución de las

reservas según el tiempo elegido. El usuario puede ver las reservas según un día específico, según un mes específico o de todo el año.



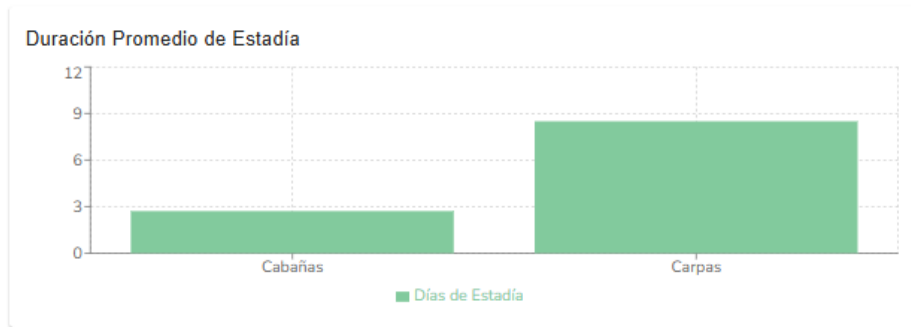
- Ranking de reservas por cabaña: Este gráfico ordena las cabañas en función de la cantidad de reservas recibidas, posicionando en los primeros lugares a las más demandadas. Los resultados pueden filtrarse por mes o año, lo que permite analizar el rendimiento de cada cabaña en distintos periodos del año. Esta herramienta es útil para que los administradores identifiquen cuáles cabañas generan mayor interés entre los visitantes y puedan: Asegurar su correcto mantenimiento, Implementar mejoras en aquellas con menor demanda, Tomar decisiones informadas sobre promociones o redistribución de recursos.

Ranking de Cabañas más Reservadas

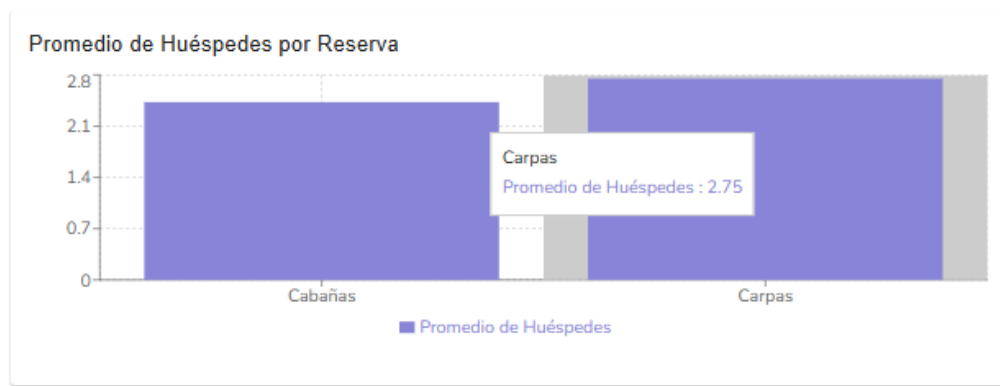
Filtrar por mes:  Año:

| Puesto | Cabaña   | Reservas |
|--------|----------|----------|
| 1º     | cabaña 2 | 2        |
| 2º     | Cabaña 5 | 1        |
| 3º     | Cabaña 1 | 2        |
| 4º     | cabaña 4 | 2        |

- Duración promedio de estadía: Este indicador muestra el promedio de días que los huéspedes permanecen en el camping, ayudando a comprender el comportamiento general de las estadías y facilitar la planificación operativa.



- Promedio de Huéspedes por reserva: Permite conocer cuántas personas, en promedio, se alojan por cada reserva, lo que ayuda a dimensionar correctamente la capacidad de atención y los recursos necesarios para cada grupo.





# BENEFICIOS POST-IMPLEMENTACIÓN

En un entorno altamente competitivo como lo es el entorno turístico, la implementación de soluciones tecnológicas puede conseguir que un negocio destaque sobre el resto de competidores. La digitalización resulta clave para mejorar la operatividad y la experiencia del usuario, por lo cual se considera que la implementación del sistema Creek en conjunto con la página web para reservas traerán múltiples beneficios para el Camping Arroyito.

La implementación de estas soluciones tecnológicas traen consigo una serie de beneficios tanto tangibles como intangibles, los cuales se procederán a detallar a continuación:

## 1. Optimización de la operatividad y eficiencia administrativa:

- Se reducirán los tiempos que los empleados dedican a realizar tareas repetitivas y tediosas que los inhiben de utilizar al máximo sus capacidades físicas y mentales, es decir, que podrían dedicar mayor esfuerzo y concentración a realizar tareas que les permitan elevar sus conocimientos y que sean más útiles a largo plazo. Se estima que reduciendo la cantidad de tareas repetitivas que se realizan en el día a día del camping tales como enviar fotos de las cabañas disponibles, responder dudas, fijarse manualmente la disponibilidad, anotar manualmente los ingresos y gastos, se puede llegar a ahorrar un 40% del tiempo que se utilizaba en estas tareas y pueden concentrarse más en satisfacer las necesidades que tengan los clientes hospedándose en el camping.
- Disminución de errores administrativos: gracias a la automatización de reservas y pagos, atrás quedaron los días en donde se daban errores graves tales como cobrar precios viejos, asignar clientes a cabañas ya ocupadas o entregar habitaciones sin haber realizado la necesaria rutina de limpieza y mantenimiento. Los errores humanos ocurren, por lo que el uso de la automatización para ayudarlo con sus tareas operativas y de limpieza evita errores que resultan del exceso de trabajo o la distracción del personal. Nivelics y Deloitte, dos grandes

empresas que ofrecen soluciones tecnológicas afirman que automatizar tareas reduce el error humano en un 50% - 60% y por lo tanto eso es lo que se espera para con el Camping Arroyito al implementar este software de gestión.

- Los datos se verán consolidados en un único sistema, lo cual eliminará el uso de múltiples planillas que contienen los datos de los clientes, las reservas, los pagos, estados de las cabañas, etc. Todos estos datos podrán ser visualizados simplemente entrando al sistema lo cual hace que el proceso de consultar información sea más rápido y eficiente.
- Los reportes creados relacionados al estado actual del camping y sus datos históricos permitirán una mejor toma de decisiones que llevarán a mejorar los resultados del camping. Datos tales como la cantidad de reservas por mes, las cabañas que son más reservadas, el promedio de días que se hospedan las personas, entre otros datos, servirán como predictores que permitan a la gestión del camping a prepararse para épocas de mayor flujo de clientes y mejorar la estadía.

## 2. Mejora en la Experiencia del Cliente

- Los clientes ahora podrán hacer reservas en línea a cualquier horario y en cualquier día, ya no existirá la limitación de esperar a que sea un horario laboral para poder recibir una respuesta de un empleado del camping y poder realizar una reserva. Esto ayudará a aumentar el porcentaje de reservas anuales en el camping debido a que se elimina el factor de impaciencia y frustración que pueden sentir algunos clientes al no recibir una respuesta inmediata y que provoque que busquen otra alternativa de hospedaje.
- Mayor transparencia: La plataforma proporciona información actualizada en tiempo real sobre precios, cabañas disponibles, el estado de las reservas y también se mandan correos electrónicos para informar al usuario sobre los distintos estados de su reserva para brindar mayor tranquilidad y evitar la incertidumbre que muchos usuarios pueden llegar a experimentar lo cual aumenta la percepción de calidad y destaca al camping sobre la competencia.
- Se espera un aumento en las valoraciones positivas en Google, plataforma globalmente utilizada para consultar la opinión de usuarios que ya visitaron el

recinto. Clientes potenciales se ven más susceptibles a hacer una reserva en lugares que cuenten con una buena valoración y reseñas por parte de otros clientes, con lo cual obtener reseñas positivas puede llegar a multiplicar por 2 la probabilidad de elección de nuevos visitantes.

### 3. Impacto económico

- Será más fácil hacer un seguimiento de los ingresos y egresos, ya no será necesario utilizar una planilla de Excel en donde es posible que existan errores o que los empleados se olviden de registrar un pago, ahora todo el flujo de cobro estará automatizado. Además al permitir cargar los comprobantes al lado de cada pago se elimina la posibilidad de pérdida de información.
- Se reducirá la morosidad de los pagos, al crear recordatorios sobre pagos pendientes y señas no realizadas, será casi imposible pasar por alto un pago adeudado lo cual reduce la posibilidad de perder dinero a causa de una distracción.
- Se prevé un incremento del 20% al 35% en reservas anuales debido a la visibilidad online del camping y a la mejora en la experiencia del usuario, esto por consecuente aumentará también las ganancias del camping
- Debido al aumento de las ganancias el camping puede invertir parte de sus ingresos en mejorar la instalación. Se han planteado varios planes de incluir más espacios para mejorar la comodidad y estadía de los huéspedes, tales como tener un lugar de recreación para niños, mejorar el área del comedor y posiblemente aumentar la cantidad de cabañas para reservas. Se espera que estos objetivos puedan alcanzarse en un lapso de 3 a 5 años gracias a la nueva presencia digital que tendrá el camping.
- Al tener un mejor control de los ingresos, egresos y el total de dinero restante hay probabilidades de que se disminuya el desperdicio de dinero debido a una mala gestión de gastos. Los gráficos y reportes generados permitirán llevar un mejor control de las salidas de dinero que tiene el camping y cómo esto impacta sobre el dinero disponible del establecimiento.

# IMPACTO SOCIAL

El impacto social de la implementación de un sistema de gestión web para Camping Arroyito se manifiesta en múltiples dimensiones que abarcan desde la mejora en la calidad de vida de los clientes y empleados hasta el fortalecimiento de la inclusión digital y el desarrollo sostenible. Este proyecto, al modernizar la operación del camping, no solo resuelve problemas técnicos y operativos, sino que también genera beneficios tangibles e intangibles para diferentes segmentos de la sociedad los cuales detallaremos a continuación:

Desde la perspectiva de los clientes, la incorporación de herramientas digitales como reservas en línea, consultas de disponibilidad en tiempo real y servicios automatizados reduce las barreras logísticas que suelen dificultar la planificación de una estadía. Por ejemplo, si una familia desea organizar una escapada de fin de semana puede gestionar su reserva desde la comodidad de su hogar, sin necesidad de realizar llamadas telefónicas, mandar mensajes de whatsapp o trasladarse físicamente para confirmar disponibilidad. Este nivel de accesibilidad y conveniencia mejora notablemente la experiencia del usuario, generando satisfacción y fidelidad hacia el establecimiento. Además, la transparencia en los procesos como la visualización en tiempo real de tarifas y disponibilidad contribuye a una percepción positiva del servicio, fortaleciendo la confianza del cliente en la empresa.

Para los empleados, el sistema de gestión representa una mejora sustancial en su carga de trabajo. Al automatizar procesos los procesos más tediosos y recurrentes del día a día, se reducen el estrés y el cansancio permitiendo que los trabajadores se concentren en actividades de mayor valor añadido. Por ejemplo, en lugar de dedicar tiempo a registrar manualmente las reservas, enviar fotos e información a todos los potenciales clientes que se comuniquen mediante Whatsapp, estar atentos a no cometer errores en los cobros y entre otras tareas que pueden resultar aburridas y pesadas, los empleados pueden enfocarse en brindar una atención personalizada a los visitantes o en supervisar el mantenimiento de las instalaciones. Este cambio no solo incrementa la eficiencia operativa, sino que también mejora el ambiente laboral al disminuir el estrés asociado con tareas rutinarias y errores humanos. En consecuencia, los empleados se sienten más motivados y

comprometidos, lo que repercute positivamente en la calidad del servicio ofrecido.

Desde una perspectiva empresarial, la modernización de las operaciones posiciona al camping como un negocio innovador y orientado a la calidad del servicio, destacándose en un mercado cada vez más competitivo. Al adoptar soluciones tecnológicas, Camping Arroyito puede atraer tanto a clientes locales como a turistas nacionales e internacionales que valoran una experiencia cómoda, moderna y adaptada a las expectativas del mundo digital. Este enfoque innovador no solo mejora la reputación del establecimiento, sino que también lo prepara para enfrentar los desafíos del turismo contemporáneo, como la necesidad de adaptarse rápidamente a nuevas demandas del mercado y a cambios en los hábitos de consumo.

#### Fomento del Desarrollo Local

La modernización del Camping Arroyito mediante la implementación de soluciones tecnológicas también tiene un impacto significativo en el desarrollo económico y social de la región, extendiéndose mucho más allá de los beneficios directos para clientes y empleados:

El impacto positivo también se extiende a la comunidad local de Rodeo, San Juan, donde se encuentra ubicado el camping. La mejora en la experiencia de los visitantes fomenta la recomendación boca a boca, atrayendo a un mayor número de turistas al área. Un incremento en el flujo turístico tiene efectos directos en la economía local: negocios como restaurantes, tiendas de souvenirs, artesanos y proveedores de servicios experimentan un aumento en sus ventas y oportunidades de crecimiento. Además, el crecimiento de visitantes impulsa el desarrollo de infraestructura en la región, generando beneficios que se extienden más allá del camping. Negocios como restaurantes, bares y cafeterías se ven beneficiados por la llegada de más visitantes que buscan disfrutar de la gastronomía regional. Esto no solo incrementa las ventas de estos establecimientos, sino que también incentiva la diversificación de la oferta gastronómica, con la incorporación de platos típicos, productos orgánicos y servicios personalizados que enriquecen la experiencia turística. Por otro lado, tiendas locales, como aquellas que venden artesanías, productos regionales, ropa y equipo para actividades al aire libre, experimentan un

incremento en sus ventas. Los turistas, al encontrar servicios de calidad en el camping, tienden a extender su estadía y explorar los alrededores, lo que amplifica su interacción con los negocios locales. Además, los emprendedores tienen la oportunidad de crear nuevos productos o servicios enfocados en satisfacer las necesidades específicas de los visitantes, como guías turísticas, transporte local, alquiler de bicicletas, caballos o experiencias de deporte náutico las cuales son el fuerte de la región.

El incremento del turismo también representa una oportunidad para revitalizar y promover los eventos culturales y recreativos de la zona. Festividades tradicionales, ferias artesanales, competencias deportivas y conciertos al aire libre pueden aprovechar el flujo de turistas para aumentar su convocatoria y relevancia. La modernización del camping y su sistema de gestión web también permite que este espacio actúe como un punto de difusión para estos eventos, facilitando que los visitantes estén informados y puedan participar activamente.

Además, el camping mismo puede convertirse en un escenario para organizar actividades lo cual servirá para dar una experiencia aún más personalizada e innovadora. Estos eventos, además de atraer turistas, generan oportunidades de empleo para organizadores, artistas y otros proveedores de servicios relacionados.

Otro beneficio se presenta para los jóvenes de la comunidad, quienes tendrán acceso a nuevas oportunidades laborales y de capacitación. La expansión del turismo puede llevar a la creación de puestos de trabajo en el mismo Camping Arroyito o incluso en otras áreas como guía turística, la gastronomía, la organización de eventos, entre otros.

Finalmente, la modernización del camping y su compromiso con la sostenibilidad lo posicionan como un destino ideal para el turismo responsable, una tendencia que se ha vuelto cada vez más popular. Esto puede generar una ventaja competitiva significativa frente a otros establecimientos que aún no han adoptado prácticas similares. Al digitalizar sus procesos y adoptar un enfoque sostenible, Camping Arroyito contribuye a la preservación del medio ambiente y también fortalece su relación con los clientes que buscan una experiencia distinta a la que se puede obtener en un hotel o alquilando un departamento o casa.

# CONCLUSIÓN

Al finalizar este proyecto y reflexionar sobre el recorrido realizado, puedo reconocer el gran desafío que representó desde sus inicios. Se trató del primer proyecto de esta magnitud en el cual debí aplicar de manera integral los conocimientos adquiridos a lo largo de mi formación como estudiante de Ingeniería en Sistemas. Además, fue la primera vez que asumí la responsabilidad de desarrollar por mi misma cada uno de los aspectos que componen un sistema completo, desde la identificación de requerimientos, la creación de la base de datos, el desarrollo del código, el testing y hasta la etapa de despliegue del sistema para su uso.

El proceso implicó investigar las tecnologías necesarias, aprender a utilizarlas, asegurar que el sistema cumpliera con los requerimientos del usuario, fuera intuitivo para su uso cotidiano y sostenible a lo largo del tiempo. Cada etapa demandó paciencia, dedicación, aprendizaje continuo y sobre todo determinación.

Trabajar con un cliente real y desarrollar una solución orientada a satisfacer una necesidad concreta me permitió comprender profundamente el impacto que un sistema informático puede generar en una empresa familiar, como es el caso del Camping Arroyito. Al finalizar este proyecto, se alcanzaron los objetivos planteados inicialmente: el camping puede ahora digitalizar y centralizar toda la información clave para su organización y operación mediante el sistema Creek. Además, se ha implementado una página web que permite a los clientes consultar información actualizada del camping y gestionar sus reservas de manera cómoda, eficiente y transparente, lo cual además contribuirá al crecimiento de la presencia web del camping. El sistema también incluye un módulo de control de accesos, que refuerza la seguridad en el manejo de datos sensibles. Esto garantiza una adecuada protección de la información. Por otro lado, permite la visualización atractiva y fácil de los datos en tiempo real del camping favorece un análisis más rápido y preciso, y a su vez optimiza la toma de decisiones estratégicas. Aunque se logró cubrir la mayoría de las necesidades detectadas, reconozco que algunas funcionalidades quedaron fuera del alcance de esta etapa del proyecto. Por ello, tengo la intención de seguir desarrollando y ampliando este sistema en el futuro, acompañando así el crecimiento del camping.

Esta experiencia ha sido sumamente enriquecedora. Me permitió crecer profesionalmente, reafirmar mi pasión por la programación y por la resolución de problemas, así como también adquirir habilidades valiosas en la interacción con un cliente y en la entrega de una solución personalizada, útil y perdurable.

Soy consciente de que aún tengo un largo camino por recorrer, con mucho por aprender y mejorar, pero me siento profundamente agradecida por los conocimientos adquiridos durante mi paso por la Universidad Católica de Córdoba. Cada una de las materias cursadas aportó elementos clave que hicieron posible la concreción de este proyecto. Sin duda, esta experiencia representa un hito significativo en mi desarrollo como futura profesional.



# ANEXO I

## CÓDIGO UTILIZADO EN PRUEBAS

### I.1) Pruebas automatizadas para Usuarios:

```
process.env.NODE_ENV = 'test';
import { test } from 'node:test';
import assert from 'node:assert/strict';
import request from 'supertest';
import { app } from '../index.js';
import { disconnectFromDB } from './testing_db.js';

let createdUserId;
console.log("Iniciando tests de Users...");

test('POST /register debe registrar un nuevo usuario', async () => {
  const newUser = {
    email: 'pruebaemail@example.com',
    password: 'securepassword',
    first_name: 'Test',
    last_name: 'User',
    phone: '123456789',
    dni: '24880200',
    birthday: '01-01-1999',
    occupation: 'Tester'
  };
  const response = await request(app).post('/register').send(newUser);
  assert.strictEqual(response.status, 200);
  assert.ok(response.body._id);
  createdUserId = response.body._id;
});

test('GET /users/:id debe obtener un usuario por ID', async () => {
  const response = await request(app).get(`/user/${createdUserId}`);
  assert.strictEqual(response.status, 200);
  assert.strictEqual(response.body.email, 'pruebaemail@example.com');
});

test('PUT /users/:id debe actualizar un usuario', async () => {
  const updatedData = { first_name: 'UpdatedName' };
  const response = await request(app).put(`/user/${createdUserId}`).send(updatedData);
  assert.strictEqual(response.status, 200);
  assert.strictEqual(response.body.first_name, 'UpdatedName');
});
```

```

    const response = await
request(app).put(`/user/${createdUserId}`).send(updatedData);
    assert.strictEqual(response.status, 200);
    assert.strictEqual(response.body.first_name, 'UpdatedName');
});

test('PUT /users/changePassword/:id debe cambiar la contraseña del usuario',
async () => {
    const newPassword = { password: 'newsecurepassword' };
    const response = await
request(app).put(`/user/changePassword/${createdUserId}`).send(newPassword);
    assert.strictEqual(response.status, 200);
});

test('DELETE /users/:id debe eliminar un usuario', async () => {
    const response = await request(app).delete(`/user/${createdUserId}`);
    assert.strictEqual(response.status, 200);
});

test('Cerrando', async() => {
    await disconnectFromDB();
    process.exit(0);
});

```

## I.2) Pruebas automatizadas para Autenticación:

```

process.env.NODE_ENV = 'test';
import { test } from 'node:test';
import assert from 'node:assert/strict';
import request from 'supertest';
import { app } from '../index.js';
import { disconnectFromDB } from './testing_db.js';

let authToken;
let createdUserId;
console.log("Iniciando tests de Auth...");

test('POST /register debe registrar un nuevo usuario', async () => {
    const newUser = {
        email: 'usuario@example.com',
        password: 'securepassword',
        first_name: 'Carlos',

```

```

    last_name: 'Gómez',
    phone: '987654321',
    dni: '87654321',
    birthday: '1992-06-15',
    occupation: 'Gerente'
  });
  const response = await request(app).post('/register').send(newUser);
  assert.strictEqual(response.status, 200);
  assert.ok(response.body._id);
  createdUserId = response.body._id;
});

test('POST /login debe autenticar un usuario y devolver un token', async ()
=> {
  const credentials = {
    email: 'usuario@example.com',
    password: 'securepassword'
  };
  const response = await request(app).post('/login').send(credentials);
  assert.strictEqual(response.status, 200);
  assert.ok(response.headers['set-cookie']);
  authToken = response.headers['set-cookie'][0];
});

test('GET /profile debe devolver la información del usuario autenticado',
async () => {
  const response = await request(app).get('/profile').set('Cookie',
authToken);
  assert.strictEqual(response.status, 200);
  assert.strictEqual(response.body.email, 'usuario@example.com');
});

test('POST /logout debe cerrar la sesión del usuario', async () => {
  const response = await request(app).post('/logout').set('Cookie',
authToken);
  assert.strictEqual(response.status, 200);
  assert.deepStrictEqual(response.body, { success: true });
});

test('Cerrando', async() => {
  await disconnectFromDB();

```

```
process.exit(0);
});
```

### I.3) Pruebas automatizadas para Reservas:

```
process.env.NODE_ENV = 'test';
import { test } from 'node:test';
import assert from 'node:assert/strict';
import request from 'supertest';
import { app } from '../index.js';
import { disconnectFromDB } from './testing_db.js';

let createdBookingId;
console.log("Iniciando tests de Bookings...");

test('POST /bookings debe crear una reserva', async () => {
  const newBooking = {
    lodge: '67b3e8effa36567d56d54d79',
    checkIn: '2025-10-07T00:00:00.000Z',
    checkOut: '2025-10-11T00:00:00.000Z',
    user: '682be492427c566d515638c9',
    numberOfAdults: 1,
    numberOfChildren: 2,
    totalAmount: 5000
  };
  const response = await
request(app).post('/bookings/createBooking').send(newBooking);
  assert.strictEqual(response.status, 200);
  assert.ok(response.body._id);
  createdBookingId = response.body._id;
});

test('GET /bookings debe devolver estado 200', async () => {
  const response = await request(app).get('/bookings');
  assert.strictEqual(response.status, 200);
});

test('GET /bookings/:id debe devolver una reserva específica', async () => {
  const response = await request(app).get(`/bookings/${createdBookingId}`);
  assert.strictEqual(response.status, 200);
  assert.strictEqual(response.body._id, createdBookingId);
});
```

```

test('DELETE /bookings/:id debe eliminar una reserva', async () => {
  const response = await request(app).delete(`/bookings/${createdBookingId}`);
  assert.strictEqual(response.status, 200);
});

test('Cerrando', async() => {
  await disconnectFromDB();
  process.exit(0);
});

```

#### I.4) Pruebas automatizadas para Empleados:

```

import { test, before } from 'node:test';
import assert from 'node:assert/strict';
import request from 'supertest';
import { app } from '../index.js';
import { disconnectFromDB } from './testing_db.js';

let createdEmployeeId;
let createdUserId;

console.log("Iniciando tests de Employees...");

test('POST /employees debe crear un empleado', async () => {
  const newEmployee = {
    email: 'empleadon@example.com',
    first_name: 'Juan',
    last_name: 'Pérez',
    dni: '65234189',
    phone: '123456789',
    birthday: '01-01-1999',
    job: 'Recepcionista',
    base_salary: 50000,
    start_date: '02-05-2025',
  };

  const response = await request(app).post('/employees/').send(newEmployee);
  assert.strictEqual(response.status, 200);
  assert.ok(response.body._id);

  createdEmployeeId = response.body._id;
  createdUserId = response.body.user;
});

```

```

test('GET /employees debe devolver estado 200', async () => {
  const response = await request(app).get('/employees/');
  assert.strictEqual(response.status, 200);
});

test('GET /employees/:id debe devolver un empleado específico', async () => {
  const response = await request(app).get(`/employees/${createdEmployeeId}`);
  assert.strictEqual(response.status, 200);
  assert.strictEqual(response.body._id, createdEmployeeId);
});

test('PUT /employees/update/:id debe actualizar un empleado', async () => {
  const updateData = {
    userId: createdUserId,
    statusName: 'Activo',
    base_salary: 55000
  };
  const response = await
request(app).put(`/employees/update/${createdEmployeeId}`).send(updateData);
  assert.strictEqual(response.status, 200);
  assert.strictEqual(response.body.base_salary, 55000);
});

test('DELETE /user/:id debe eliminar una reserva', async () => {
  try {
    if (createdEmployeeId) {
      await EmployeeModel.deleteOne({ _id: createdEmployeeId });
    }
    if (createdUserId) {
      await request(app).delete(`/user/${createdUserId}`);
    }
  } catch (err) {
    console.error("Error eliminando registros de prueba:", err);
  }
});

test('Cerrando', async() => {
  await disconnectFromDB();
  process.exit(0);
});

```

I.5) Pruebas automatizadas para Cabañas:

```
process.env.NODE_ENV = 'test';
import { test } from 'node:test';
import assert from 'node:assert/strict';
import request from 'supertest';
import { app } from '../index.js';
import { disconnectFromDB } from './testing_db.js';
let lodgeId;

console.log("Iniciando tests de Lodges...");

test('GET /lodges debe devolver estado 200 y un array', async () => {
  const response = await request(app).get('/lodges');
  assert.strictEqual(response.status, 200);
  assert.ok(Array.isArray(response.body));
});

test('POST /lodges debe crear una cabaña', async () => {
  const newLodge = {
    name: "Cabaña Test",
    description: "test",
    services: "test",
    capacity: 4
  };

  const response = await request(app).post('/lodges').send(newLodge);
  console.log("Status:", response.status);
  console.log("Body:", response.body);
  assert.strictEqual(response.status, 200);
  assert.strictEqual(response.body.name, newLodge.name);
  lodgeId = response.body._id;
});

test('PUT /lodges/:id debe actualizar una cabaña', async () => {
  const updatedData = { name: "Cabaña Modificada" };

  const response = await
request(app).put(`/lodges/${lodgeId}`).send(updatedData);
  assert.strictEqual(response.status, 200);
  assert.strictEqual(response.body.name, updatedData.name);
});
```

```

test('POST /lodges_x_status/ debe crear una nueva entrada con estado ocupado',
  async () => {

    console.log(lodgeId);
    const response = await request(app).post(`/lodge_x_status`).send({
      lodge: lodgeId,
      booking: "682be31d427c566d515638c8",
      status: "67affd400cb373c285f445f2",
      comment: "lol",
    });

    console.log("Status:", response.status);
    console.log("Body:", response.body);
    assert.strictEqual(response.status, 200);
    assert.ok(response.body._id);
  });

test('DELETE /lodges/:id debe eliminar una cabaña', async () => {
  const response = await request(app).delete(`/lodges/${lodgeId}`);
  assert.strictEqual(response.status, 200);

});

test('Cerrando', async() => {
  await disconnectFromDB();
  process.exit(0);
});

```

## I.6) Pruebas automatizadas para Carpas:

```

import { test } from 'node:test';
import assert from 'node:assert/strict';
import request from 'supertest';
import { app } from '../index.js';
import { disconnectFromDB } from './testing_db.js';

let createdTentId;
console.log("Iniciando tests de tents...");

test("POST /tents debe crear una nueva carpa", async () => {
  const newTent = {

```



```

    first_name: "Micaela",
    last_name: "Vargas",
    email: "micaela@gmail.com",
    photo: "380848593",
    dni: "43444120",
    occupation: "Estudiante",
    checkIn: '2025-10-07T00:00:00.000Z',
    checkOut: '2025-10-11T00:00:00.000Z',
    numberOfAdults: 2,
    numberOfChildren: 3,
  };

  const response = await request(app).post("/tents").send(newTent);

  createdTentId = response.body._id;

  assert.strictEqual(response.status, 200);
  assert.ok(response.body._id)
  createdTentId = response.body._id;
});

test("GET /tents/:id debe devolver una carpa", async () => {
  const response = await request(app).get(`/tents/${createdTentId}`);

  console.log("Respuesta GET por ID:", response.body);

  assert.strictEqual(response.status, 200);
  assert.strictEqual(response.body._id, createdTentId);
});

test("GET /tents debe devolver todas las carpas", async () => {
  const response = await request(app).get("/tents");

  assert.strictEqual(response.status, 200);
});

test("GET /tents/occupiedPositions debe devolver posiciones ocupadas",
async () => {
  const response = await request(app).get("/tents/occupiedPositions");
  assert.strictEqual(response.status, 200);
});

```

```

test("DELETE /tents/:id debe eliminar una carpa", async () => {
  const response = await request(app).delete(`/tents/${createdTentId}`);
  assert.strictEqual(response.status, 200);
  assert.strictEqual(response.body, "Carpa eliminada");
});

test('Cerrando', async() => {
  await disconnectFromDB();
  process.exit(0);
});

```

### 1.7) Pruebas automatizadas para Contabilidad:

```

process.env.NODE_ENV = 'test';
import { test } from 'node:test';
import assert from 'node:assert/strict';
import request from 'supertest';
import { app } from '../index.js';
import { disconnectFromDB } from './testing_db.js';

let createdAccountingId;
console.log("Iniciando tests de Accounting...");

test('POST /accounting debe crear un registro contable', async () => {
  const newAccounting = {
    amount: 1000,
    totalAmount: 2000,
    remainingAmount: 1000,
    lodge: '682be31d427c566d515638c8',
    type: 'Ingreso',
    date: '2025-02-05T00:00:00.000Z',
    status: "pendiente",
  };
  const response = await
request(app).post('/accounting/createAccounting').send(newAccounting);
  assert.strictEqual(response.status, 200);
  assert.ok(response.body._id);
  createdAccountingId = response.body._id;
});

test('GET /accounting debe devolver estado 200', async () => {

```

```

    const response = await request(app).get('/accounting');
    assert.strictEqual(response.status, 200);
  });

test('GET /accounting/:id debe devolver un registro contable específico',
  async () => {
    const response = await
request(app).get(`/accounting/${createdAccountingId}`);
    assert.strictEqual(response.status, 200);
    assert.strictEqual(response.body._id, createdAccountingId);
  });

test('PUT /accounting/status/:id debe actualizar el estado de pago', async ()
=> {
    const response = await
request(app).put(`/accounting/status/${createdAccountingId}`).send({ status:
'parcial' });
    assert.strictEqual(response.status, 200);
    assert.strictEqual(response.body.status.status, 'parcial');
  });

test('GET /accounting/filter?filter=month debe devolver contabilidad del mes',
  async () => {
    const response = await request(app).get('/accounting/filter?filter=month');
    assert.strictEqual(response.status, 200);
    assert.ok(response.body.totalMoney !== undefined);
  });

test('DELETE /accounting/:id debe eliminar un registro contable', async () =>
{
    const response = await
request(app).delete(`/accounting/${createdAccountingId}`);
    assert.strictEqual(response.status, 200);
  });

test('Cerrando', async() => {
    await disconnectFromDB();
    process.exit(0);
  });

```

## I.8) Código de pruebas de la página web Camping Arroyito en Cypress:

```
describe('Prueba de login', () => {
  it('Debería loguearse correctamente con credenciales válidas', () => {
    cy.viewport(1280, 1000); //ancho x alto en píxeles

    cy.visit('http://localhost:5173/login');

    cy.get('input[type="email"]').type('pruebaemail@example.com');
    cy.get('input[type="password"]').type('securepassword');
    cy.get('button[class="primary"]').click();

    cy.url().should('include', '/');
    cy.contains('Test'); //que en algun lado diga nombre del user

    //selecciono fechas
    cy.get('.headerSearchText').first().click();
    cy.get('.rdrDayToday').first().click();
    cy.get('.rdrDayToday').first().next().click();
    cy.get('.headerSearchText').first().click(); //cierro el modal

    //selecciono cant de adultos y niños
    cy.get('.headerSearchText').eq(1).click();
    cy.get('.optionItem').eq(0).within(() => {
      cy.get('.optionCounterButton').contains('+').click();
    });
    cy.get('.optionItem').eq(1).within(() => {
      cy.get('.optionCounterButton').contains('+').click();
    });
    cy.get('.headerSearchText').eq(1).click(); //cierro el modal

    //click para buscar
    cy.get('button[class="bg-primary text-white p-3 rounded-full"]').click();

    //me lleva al /roomsPage y se cargan las cabañas disponibles
    cy.location('pathname', { timeout: 1000 }).should('eq', '/roomsPage');

    //se carga al menos una cabaña
    cy.get('.searchItemCard', { timeout: 3000 }).should('have.length.greaterThan',
```

```
0);

//click en la primera cabaña disponible
cy.get('button[class="cardButton"]').first().click();

//verifica redirección a la ficha del lodge
cy.url().should('include', '/lodges/');
cy.get('.lodgeTitle', { timeout: 3000 }).should('exist');

-

cy.get('.roomsSearchText').first().click(); // abrir el date picker

//avanzar al mes siguiente
cy.get('.rdrNextButton').click();
cy.get('.rdrMonthAndYearWrapper').should('exist');

//seleccionar el primer día habilitado
cy.get('.rdrMonth').last().find('.rdrDay:not(.rdrDayPassive)').eq(0).click();
//selecciona el segundo día habil
cy.get('.rdrMonth').last().find('.rdrDay:not(.rdrDayPassive)').eq(1).click();

//cerrar el date picker
cy.get('.roomsSearchText').first().click();

cy.get('.roomsSearchText').eq(1).click(); // abre opciones
cy.get('.optionItem').eq(0).within(() => {
  cy.get('.optionCounterButton').contains('-').click(); //suma adulto
});

cy.get('.optionItem').eq(1).within(() => {
  cy.get('.optionCounterButton').contains('-').click(); //suma niño
});
cy.get('.roomsSearchText').eq(1).click(); //cierra opciones

//verifica que el precio total cambió
cy.get('.priceRow.total span')
  .last()
  .invoke('text')
  .then((totalPriceText) => {
    expect(totalPriceText).to.match(/\$\d+/);
  });
```

```

        cy.get('.bookNowButton').should('not.be.disabled').click(); //click boton de
reserva

        //verifica redireccion
        cy.location('pathname', { timeout: 5000 }).should('include', '/account');

        //click para ir a user page
        cy.get('a.login').click();

        cy.contains('a', 'My Bookings').click();

        //espera a que se cargue alguna reserva
        cy.get('.text-gray-600').should('exist');

        //que haya al menos una con estado Pendiente
        cy.contains('Pendiente: Esperando transferencia').should('exist');

        cy.contains('button', 'Cancelar').should('not.be.disabled').click(); //click
boton cancelar

        //espera alerta
        cy.on('window:alert', (text) => {
            expect(text).to.contains('Reserva cancelada exitosamente');
        });
    });
});

it('muestra mensaje si login falla con 401', () => {
    cy.intercept('POST', 'http://localhost:5173/login', {
        statusCode: 401,
        body: { error: 'Email o contraseña incorrecto' },
    });

    cy.visit('http://localhost:5173/login');

    cy.get('input[type=email]').type('pruebaemail@example.com'); //mail existe
    cy.get('input[type=password]').type('wrongpassword'); //contraseña incorrecta

    cy.window().then((win) => {
        cy.stub(win, 'alert').as('alert');
    });
});

```

```

cy.get('button[class="primary"]').click();

    cy.get('@alert').should('have.been.calledWith', 'Email o contraseña incorrecto');
//se fija que dice la alerta
});

it('solo uno de dos usuarios puede reservar la misma cabaña', () => {
    const payload = {
        lodge: '67b3e8effa36567d56d54d79',
        checkIn: '2025-10-10',
        checkOut: '2025-10-13',
        numberOfAdults: 2,
        numberOfChildren: 0,
        totalAmount: 20000,
        origin: 'cliente',
    };

    const user1 = { _id: '6862c06576718babb0c0e254', first_name: 'Test', email:
'pruebaemail@example.com' };
    const user2 = { _id: '682be492427c566d515638c9', first_name: 'Mica', email:
'm@gmail.com' };

    //user 1
    cy.request('POST', 'http://localhost:3001/bookings/createBooking', { ...payload,
user: user1._id }).then((res1) => {
        expect(res1.status).to.eq(200);
    });

    //user 2 intenta lo mismo
    cy.request({
        method: 'POST',
        url: 'http://localhost:3001/bookings/createBooking',
        failOnStatusCode: false,
        body: { ...payload, user: user2._id }
    }).then((res2) => {
        expect(res2.status).to.not.eq(200);
        expect(res2.body.message).to.match(/La cabaña ya está reservada en esas
fechas./i);
    });
});

```

# BIBLIOGRAFÍA

- DoceTv Misiones(2024). La Provincia y Aves Argentinas firmaron un convenio para potenciar el turismo de naturaleza. Recuperado de <https://www.canal12misiones.com/noticias-de-misiones/turismo/la-provincia-y-aves-argentinas>
- BBVA (2025). ¿Qué es el turismo sostenible y por qué es clave para el planeta? Recuperado de <https://www.bbva.com/es/sostenibilidad/que-es-el-turismo-sostenible/>
- Vidable, C., Gho Silva, M., Nicolás, A., & Viel, J. (2024). PLAN DE NEGOCIOS Y MEJORA DE PROCESOS PARA COMPLEJO TURÍSTICO DE CABAÑAS Y CAMPING “ARROYITO”. AJEA (Actas De Jornadas Y Eventos Académicos De UTN), (AJEA 30). Recuperado a partir de <https://rtyc.utn.edu.ar/index.php/ajea/article/view/1446>
- Django Software Foundation (2025). Django documentation (v5.1). Recuperado de <https://docs.djangoproject.com/en/5.1/>
- GeeksforGeeks (s.f.). How much Python should you know to learn Django? Recuperado de <https://www.geeksforgeeks.org/how-much-python-should-you-know-to-learn-django/>
- Noble Desktop (s.f.). How difficult is it to learn Django? Recuperado de <https://www.nobledesktop.com/learn/django/how-difficult-is-it-to-learn-django>
- Rorbits (s.f.). Ruby on Rails Scalability. Recuperado de <https://www.rorbits.com/ruby-on-rails-scalability/>
- React.js (2025). React Documentation. Recuperado de <https://react.dev/>
- MongoDB (s.f.). MongoDB Blog. Recuperado de <https://www.mongodb.com/>
- RisingStack (s.f.). Node.js Tutorials and Articles. Recuperado de <https://blog.risingstack.com/>



- MongoDB (s.f.). Why Choose MongoDB? Recuperado de <https://www.mongodb.com/>
- Kinsta (2023). Mean Stack Explained. Recuperado de <https://kinsta.com/blog/mean-stack/>
- Curotec (s.f) Mean vs. Mern: Choosing the Right Tech Stack for your next project. Recuperado de <https://www.curotec.com/insights/mean-angular-vs-mern-react/>
- Express (s.f) Recuperado de <https://expressjs.com/>
- React Native. (s.f.). Recuperado de <https://reactnative.dev/>
- DesarrolloWeb (2023). Que es MVC. Recuperado de <https://desarrolloweb.com/articulos/que-es-mvc.html>
- Alexander Shvets (2014- 2025). Dive into design patterns. Recuperado de <https://refactoring.guru/es>
- Vercel (2023). What does Vercel do? Recuperado de <https://vercel.com/blog/what-is-vercel>
- Northflank (2025). Vercel vs Heroku: Which platform fits your workflow best? Recuperado de <https://northflank.com/blog/vercel-vs-heroku>
- Jay Shah(2024). Key Differences between Vercel and Render. Recuperado de <https://www.linkedin.com/pulse/key-differences-between-vercel-render-jay-shah-cdq4f/>
- Medium (2024). Render - Deploying Backends Made Easy. Recuperado de <https://medium.com/%40muskanverma/render-deploying-backends-made-easy-7d8e0ef878e5>
- Render (s.f) Web Services. Recuperado de <https://render.com/docs/web-services>
- Github (s.f). Entender las Github Actions. Recuperado de <https://docs.github.com/es/actions/about-github-actions/understanding-github-actions>
- Operto (s.f). Cómo las automatizaciones hoteleras reducen los errores del personal y ahorran ganancias. Recuperado de <https://es.operto.com/blog/hotel-error-management/>
- Nivelics (s.f) Automation: the path to operational efficiency in Large Companies. Recuperado de

<https://www.nivelics.com/en/blog/automation-operational-efficiency>

- The Hotel's Yearbook (2023). Recuperado de <https://www.hotelyearbook.com/edition/2023.html>