



fidÉлитas
INGENIERÍA EN SISTEMAS DE COMPUTACIÓN

DN01 Documento de Arquitectura

Coromoto Access

HOJA DE CONTROL

Proyecto	Coromoto Access		
Entregable	DN01 Documento de arquitectura		
Autores	José Mario Ulloa Vargas. Alejandro Rivero Reyes. Jonathan Esquivel Guillén. Nicole Torres Rodríguez.		
Versión/Edición	0102	Fecha Versión	12 de diciembre de 2024
		Nº Total de Páginas	26

REGISTRO DE CAMBIOS

Versión	Causa del Cambio	Responsable del Cambio	Fecha del Cambio
1.0	Elaboración del Documento	Todos	28-11-2024
1.1	Modificación del documento	Jonathan Esquivel Guillén	10-12-2024
1.2	Correcciones del documento	Alejandro Rivero Reyes	12-12-2024
1.3	Correcciones Finales	Alejandro Rivero, José Ulloa y Nicole Torres	12-12-2024

Contenido

HOJA DE CONTROL.....	2
REGISTRO DE CAMBIOS.....	2
1. Introducción.....	3
2. Propósito.....	3
3. Descripción del sistema.....	4
4. Objetivos y Restricciones arquitectónicas.....	5
4.1. Plataforma tecnológica.....	6
4.1.1. Aplicación.....	7
4.2. Seguridad.....	9
4.3. Tecnologías utilizadas.....	10
4.4. Herramientas utilizadas para el desarrollo.....	12
SQL (Structured Query Language).....	12
SQL Server Management 20.....	13
Hostinger.....	13
JavaScript.....	13
CSS.....	13
HTML.....	13
Bootstrap.....	13
4.5. Estándares de desarrollo.....	13
4.5.1. Interfaces.....	14
4.5.2. Datos de diseño.....	16
4.6. Componentes.....	17
4.6.1. Servidor de pruebas.....	17
4.6.2. Servidor de producción.....	18
4.6.3. Base de datos.....	19
4.6.4. Diagrama entidad-relación (si aplica).....	20
4.6.5. Diagrama de infraestructura.....	23
5. Control de Cambios y Versiones.....	23
5.1.1. GitHub.....	24
6. Referencias.....	25

1. Introducción

El objetivo principal es establecer de manera detallada las funcionalidades y características que debe incluir el sistema para garantizar su eficiencia, seguridad y cumplimiento con las necesidades del cliente.

Asimismo, está pensado para que vaya en aumento, permitiendo adaptarse a futuras exigencias del negocio.

Entre sus principales características destacan la sincronización en tiempo real de la disponibilidad de habitaciones, la actualización automatizada de registros de limpieza y la generación de reportes que facilitan el control operativo.

Este documento también define el alcance del proyecto, los pasos a seguir durante el desarrollo y los procedimientos para gestionar cambios, manteniendo así un enfoque organizado y flexible. Adicionalmente, se especifican los roles de los diferentes usuarios del sistema, de forma que sus funciones se adapten a las necesidades específicas de los administradores, empleados y clientes.

2. Propósito

Este sistema tiene como propósito principal transformar la forma en que se gestiona y opera la Villa Coromoto, poniendo como prioridad tanto la eficiencia en los procesos internos como la experiencia del usuario. A través de la automatización de tareas repetitivas y la integración de herramientas modernas, buscamos facilitar el trabajo diario del equipo, permitiendo que las operaciones sean más fluidas y organizadas.

Además, este proyecto está diseñado para crecer y adaptarse a las necesidades futuras de la Villa, asegurando que pueda evolucionar con el tiempo. Queremos crear un sistema que no sea solo práctico, sino también flexible. Por eso, nos estamos enfocando en aplicar prácticas profesionales durante todo el desarrollo, desde el diseño hasta la implementación, para garantizar un producto final bien estructurado y de alta calidad. En otras palabras, queremos que este sistema sea una herramienta confiable que realmente haga la diferencia.

3. Descripción del sistema

La plataforma web estará diseñada para gestionar de manera integral la administración de la VillaCoromoto, cuenta con modelos de reservas de habitaciones, que permite a los usuarios realizar la consulta y poder modificar las reservas con respecto al día de llegada y cantidad de usuarios, al consultar la disponibilidad de habitaciones en tiempo real, lo cual al llenar los datos de registros, llega una alerta a los empleados para completar el registro y mantener la comunicación con el cliente, además de incluirlos en la base de datos de forma automática, la interfaz intuitiva será para el manejo de empleados, inventario de activos, gestiones que realizan los empleados como vacaciones, consultas de pago entre otras.

Sobre las funcionalidades se incluye la de registro histórico de facturas, esto porque la empresa maneja el sistema de hacienda y al realizar los pagos, le llega al cliente copia automática de hacienda, lo cual permite mantener los históricos de facturas.

Entre las funciones que realizará el sistema Villa Coromoto sobre los accesos al sistema tenemos:

1. **El módulo Catálogo de Hospedajes:** Permitirá gestionar la información relacionada con las habitaciones disponibles. Esto incluye diferentes tipos de habitaciones que contiene el complejo, especificando las tarifas según la temporada, se especifica horarios de check-in y check-out.
2. **El módulo Gestión y Registro de Reservas:** Permitirá administrar de manera eficiente las reservas de los clientes. Se podrán registrar nuevas reservas y gestionar las existentes, especificando fechas de inicio y fin, así como la cantidad de huéspedes. También se controla el estado de la reservación, indicando si la habitación está ocupada o disponible
3. **El módulo Inventario de Activos por Habitación:** Se encargará de gestionar y controlar los activos disponibles en cada habitación. Cada artículo se identificará mediante un ID único y se clasificará en diferentes categorías, como electrónicos, muebles, entre otros. Se registran detalles específicos de cada activo, incluyendo modelo, número de serie y una descripción del mismo

4. **El módulo Gestión de Empleados:** Permitirá administrar la información relacionada con los empleados de la empresa. Incluyendo el registro de datos personales y de contacto, así como la asignación de roles y responsabilidades específicas para cada empleado.
5. **El módulo Control de Limpieza:** Este permitirá gestionar el proceso de limpieza de las habitaciones de una forma más detallada. Esto incluye un registro de limpieza que almacena los nombres de los responsables de la limpieza, además del estado de cómo se encuentra la habitación y el cómo quedó después de la limpieza y algunos comentarios alusivos
6. **El módulo Control de Facturas:** Este módulo se encargará de enviar a los clientes de Villa Coromoto sus respectivas facturas y almacenarlas todas las facturas generadas en el sistema. Además, el sistema debe permitir tanto al cliente como al administrador descargar las facturas, en el caso del cliente solo descargara su respectiva factura y en el caso del administrador la de cualquier cliente, por medio de un archivo XML.
7. **El módulo Catálogo de Clientes:** Permite registrar y gestionar la información de los clientes, esto incluye información importante como nombres, datos de contacto como direcciones y números de teléfono y por último detalles de la persona, en este caso el cliente, los cuales se almacenará en una base de datos.
8. **El Módulo de Seguridad:** Se encargará de asignar y gestionar los permisos para los diferentes tipos de usuarios, estos van desde los clientes, empleados y hasta los administradores.
9. **El módulo de RRHH:** Este se diseñará con la finalidad de administrar eficientemente las solicitudes del personal de la Villa Coromoto, esto ayudando a la eficacia del módulo de gestión de empleados por medio de la implementación de turnos, nómina (Salario, nombramiento), gestión de vacaciones (día completo o medio día,) con previa aprobación del manager y evaluaciones de desempeño
10. **El módulo de Parámetro:** Este apartado es un módulo de configuración, donde se podrán configurar los métodos de pago, además de los tipos de reservaciones, el tipo de moneda a pagar (dólares, colones, euros), todos estos datos, deberán contar con su respectivo id para un cambio de forma rápida de ser requerido

Con estos módulos lograremos que VillaCoromoto esté automatizado a su totalidad, con el fin que hospedarse en esta Villa vacacional sea sencillo y eficaz

4. Objetivos y Restricciones arquitectónicas

El sistema debe ser diseñado para manejar un aumento en la carga de trabajo sin comprometer su rendimiento, garantizando escalabilidad y eficiencia operativa incluso en situaciones de alta demanda. Además, debe asegurar una disponibilidad continua, minimizando los tiempos de inactividad para garantizar la continuidad del servicio en todo momento.

La disponibilidad es otro aspecto crítico; el sistema debe estar siempre accesible, minimizando al máximo los tiempos de inactividad para garantizar la continuidad del servicio, se espera que el sistema cumpla con estándares predefinidos, como tiempos de respuesta rápidos y un procesamiento eficiente de los datos.

La seguridad es fundamental. El sistema debe proteger los datos sensibles y cumplir con los estándares establecidos para evitar vulnerabilidades que puedan comprometer su integridad o la privacidad de los usuarios. Se debe garantizar una estructura modular y bien documentada. Esto facilita la implementación de cambios o actualizaciones sin que afecten negativamente a otras partes del sistema, asegurando su adaptabilidad y evolución en el tiempo.

Sobre las restricciones arquitectónicas tomamos en cuenta que se depende de programas de software de terceros como las bases de datos, y se depende de las actualizaciones, los cuales pueden cambiar las políticas de uso o modificaciones a los sistemas.

Tomando en cuenta el hosting se buscó uno con varios nodos, el cual si uno se cae entra a funcionar el otro manteniendo el sistema web en línea.

4.1. Plataforma tecnológica

En este apartado de plataforma tecnológica se abordan las herramientas y servicios que proporcionan el entorno que se necesita para la el desarrollo, implementación y funcionamiento del sistema de VillaCoromoto, donde se incluye tanto el hardware como el software, aplicaciones, seguridad y tecnologías, todo el entorno completo donde se ejecuta y mantiene el sistema, estas decisiones técnicas influirán en el diseño de la implementación para esto tenemos las siguientes definiciones:

Lenguajes de programación: El lenguaje o los lenguajes utilizados para desarrollar la aplicación en este caso C#, JavaScript, HTML y CSS.

Según AWS (2024): JavaScript es un lenguaje de programación que los desarrolladores utilizan para hacer páginas web interactivas. Desde actualizar fuentes de redes sociales a mostrar animaciones y mapas interactivos, las funciones de JavaScript pueden mejorar la experiencia del usuario de un sitio web. Como lenguaje de scripting del lado del servidor, se trata de una de las principales tecnologías de la World Wide Web.

Seguridad: Se utilizarán herramientas y servicios para garantizar la seguridad del sistema, como cifrado, autenticación, autorización y control de acceso.

Sistemas operativos: Principalmente el sistema se realizará en computadoras con sistemas operativos de Windows, este sistema tiene como objetivo funcionar en los sistemas operativos de MacOS (Mac) y Windows.

Bases de datos: Para CoromotoAccess, se planea utilizar un sistema de gestión de bases de datos (DBMS) como SQL Server Management 20. Dado que contamos con experiencia en el desarrollo con este sistema, nos facilita el proceso de almacenar y gestionar la información de la empresa.

4.1.1. Aplicación

En este apartado relacionado con la Capa de arquitectura para la aplicación web, se trabajará utilizando microservicios con un enfoque de cliente-servidor. Estos servicios se comunican entre sí por medio de las APIs, a continuación, se plantea como se va a desarrollar la aplicación:

Capa de Frontend (Cliente): Esta capa se encarga de la interacción con el usuario final, el frontend es la responsable de la presentación de los datos y de gestionar las interacciones con el usuario, utilizando para esto las tecnologías de HTML/CSS para la estructura y estilizar las páginas, además JavaScript para la interacción dinámica de la página, y el framework de .net para la creación de las interfaces utilizando el lenguaje de c#, que se detalla más adelante en el documento.

Capa Backend (Servidor): El backend es responsable de la lógica de negocio y el acceso a la base de datos. Aquí se gestionan las solicitudes recibidas del frontend y se procesan según las reglas del negocio.

Capa de Base de Datos: La base de datos es la encargada de almacenar y recuperar los datos necesarios para la aplicación.



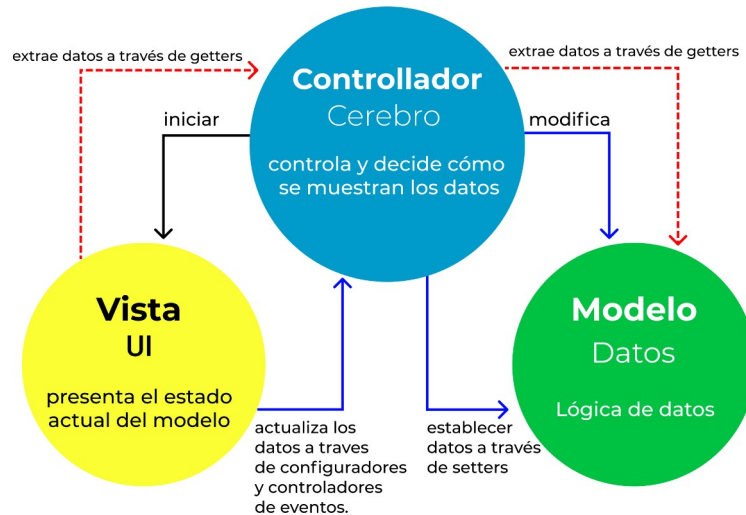
Adicionalmente para la realización del sistema se utilizará la arquitectura MVC (Model-View-Controller)

Según (Hernández, 2021), la arquitectura Model-View-Controller (MVC) es un patrón de diseño que separa una aplicación en tres componentes principales, resultando en los siguientes:

- **Modelo (Model).** Este representa los datos y la lógica de negocio del sistema. Se encarga del acceso a los datos almacenados, ya sea en una base de datos o en otros medios. También define las reglas de negocio que se deben cumplir.
- **Vista (View).** Esta capa se encarga de la presentación de los datos. Por tanto, interactúa con el usuario, muestra la información y recibe la entrada. Se diseña para que sea independiente del modelo. Es decir, que, como en la hexagonal, se puede cambiar sin afectar a los datos subyacentes.

- Controlador (Controller). Es el intermediario entre los dos componentes anteriores. Procesa las solicitudes o entradas del usuario; las procesa y realiza las operaciones necesarias en el modelo, como, por ejemplo, actualizarlo, y decide qué vista es la que se mostrará después con los resultados.

Patrones de Arquitectura MVC



Fuente: Hernández, 2021.

4.2. Seguridad

La seguridad en cualquier sistema web debe basarse en principios sólidos para garantizar la protección de los datos y la privacidad de los usuarios. Uno de los principios fundamentales es el **acceso mínimo requerido**, que implica que los usuarios, independientemente de su perfil, solo deben tener acceso a la información y funcionalidades estrictamente necesarias para desempeñar su rol en el sistema. Implementar este principio ayuda a reducir el riesgo de brechas de seguridad y asegura que la información sensible sea accesible solo para las personas adecuadas.

El sistema va a tener autenticación de Usuarios: La **autenticación** es el proceso mediante el cual el sistema verifica la identidad de un usuario que intenta acceder. En un sistema seguro, se deben emplear métodos de autenticación robustos para asegurar que solo usuarios legítimos puedan acceder, esto por medio de contraseñas, basados en una

política de contraseña fuerte que contenga mínimo 11 caracteres, entre los que están combinación de letras, número y caracteres especiales.

La autenticación de usuarios está basada en los roles o perfiles, en los cuales están:

- **Administrador:** Acceso total a todas las funcionalidades y configuraciones del sistema, incluyendo la gestión de usuarios y configuración avanzada.
- **Limpieza:** Acceso limitado, tendrá el acceso al área de limpieza y visualizará sus datos de perfil, su rendimiento, vacaciones y otras opciones relacionadas a los empleados, este no podrá modificar datos, solo podrá confirmar si se realiza una limpieza.
- **Usuario Regular:** Acceso limitado a ciertas funcionalidades según las necesidades del negocio (por ejemplo, acceso solo a sus datos o a ciertas secciones del sistema).

4.3. Tecnologías utilizadas

Lenguaje de programación: C# y .NET en Visual Studio, que es el entorno de desarrollo integrado (IDE). Este entorno aumenta la productividad y reduce errores de sintaxis, al contar experiencia en estos lenguajes, nos facilita el desarrollo del sistema en conjunto de su ambiente amigable, se estima realizar un sistema eficiente.

Aplicaciones Web: Con ASP.NET Core, que permite la creación de aplicaciones web modernas con MVC (Modelo-Vista-Controlador) y APIs RESTful.

Frameworks y Bibliotecas Comunes en C#:

- **ASP.NET Core:** Framework para construir aplicaciones web, APIs y servicios.
- **Entity Framework Core:** ORM (Object-Relational Mapper) para facilitar la interacción entre la base de datos y la aplicación en C#.
- **Bootstrap:** Biblioteca de diseño front-end para crear interfaces de usuario responsivas y modernas.
- **jQuery:** Biblioteca de JavaScript que simplifica la manipulación del DOM y la gestión de eventos.

Características de C#:

- **Orientado a Objetos:** C# es un lenguaje de programación orientado a objetos, lo que significa que utiliza conceptos como clases, objetos, herencia, polimorfismo y encapsulación. Esto facilita la organización del código y su reutilización.
- **Sintaxis Clara y Limpia:** C# tiene una sintaxis bastante similar a otros lenguajes de programación como C, C++ y Java, lo que facilita la transición para programadores con experiencia en esos lenguajes.
- **Compatibilidad con LINQ (Language Integrated Query):** LINQ permite realizar consultas a fuentes de datos (como bases de datos, colecciones en memoria, archivos XML, etc.) de manera intuitiva y eficiente dentro del mismo lenguaje.
- **Asincronía:** C# ofrece características avanzadas para la programación asincrónica, como la palabra clave `async` y `await`, que permiten ejecutar operaciones sin bloquear el hilo principal de ejecución, mejorando la eficiencia en aplicaciones de alto rendimiento, como las que realizan operaciones de entrada/salida (I/O).
- **Multiplataforma (Con .NET Core y .NET 5/6+):** Aunque originalmente C# estaba vinculado principalmente a Windows, con la llegada de .NET Core y posteriormente .NET 5/6+, ahora se pueden ejecutar aplicaciones C# en diferentes sistemas operativos como Linux y macOS, lo que lo hace aún más versátil.

SQL Server Management 20:

SQL Server Management 20 es una herramienta utilizada para gestionar bases de datos relacionales. A través de esta herramienta, se pueden realizar operaciones como la creación, modificación y consulta de bases de datos, así como la gestión de los datos almacenados, utilizar esta herramienta nos aporta las siguientes características:

- **Consultas a Bases de Datos:** SQL Server Management 20 permite realizar consultas complejas a las bases de datos, tanto para recuperar datos (con `SELECT`), como para insertar (`INSERT`), actualizar (`UPDATE`) o eliminar (`DELETE`) datos.
- **Definición de Esquemas:** Con SQL Server Management 20 se pueden definir y modificar la estructura de las bases de datos (tablas, índices, relaciones, vistas, procedimientos almacenados, etc.).
- **Transacciones:** SQL Server Management 20 soporta la gestión de transacciones, lo que permite agrupar varias operaciones en una sola unidad de trabajo. Las

transacciones aseguran que las bases de datos se mantengan consistentes incluso en presencia de errores o fallos del sistema.

- **Normalización de Bases de Datos:** SQL Server Management 20 se utiliza para estructurar las bases de datos de forma eficiente, aplicando reglas de normalización para reducir la redundancia y mejorar el rendimiento.

4.4. Herramientas utilizadas para el desarrollo

Durante el desarrollo del proyecto y con el fin de llegar a un final satisfactorio se utilizarán las siguientes herramientas:

Nombre	Descripción
Visual Studio 2022	plataforma de lanzamiento creativa que puede utilizar para editar, depurar y compilar código y, finalmente, publicar una aplicación (<i>Visual Studio: IDE y Editor de Código Para Desarrolladores de Software y Teams</i> , 2024)
C#	Lenguaje de programación desarrollado por Microsoft como parte de su plataforma .NET. Es un lenguaje de propósito general, orientado a objetos y diseñado para ser simple, moderno y seguro

SQL (Structured Query Language)	SQL es un lenguaje estándar para gestionar bases de datos relacionales. A través de SQL se pueden realizar operaciones como la creación, modificación y consulta de bases de datos, así como la gestión de los datos almacenados.
SQL Server Management 20	Sistema de gestión de bases de datos relacionales (RDBMS), diseñado para almacenar, recuperar y administrar datos de manera eficiente, ofreciendo funcionalidades avanzadas para el manejo de datos en aplicaciones empresariales y soluciones analíticas.
Hostinger	Ofrece servicios de hosting web, dominios y herramientas relacionadas con el desarrollo de sitios web entre desarrolladores, pequeñas empresas y usuarios que buscan soluciones económicas para alojar sus sitios web.
JavaScript	Lenguaje de programación interpretado, ligero y de alto nivel diseñado principalmente para agregar interactividad y dinamismo a los sitios web, se usa en su mayoría para, desarrollo web, junto con HTML y CSS.
CSS	Lenguaje de estilo utilizado para describir la presentación de documentos HTML o XML. CSS define cómo se muestran los elementos en pantalla, impresoras o en otros medios, y permite separar el contenido de un sitio web de su diseño visual.
HTML	Es un lenguaje de marcado que utiliza etiquetas para definir la estructura y el contenido de una página web, como texto, imágenes, enlaces, tablas y formularios.
Bootstrap	Framework de código abierto para el desarrollo de interfaces web, que facilita la creación de sitios y aplicaciones web modernas, responsivas y con un diseño consistente.

4.5. Estándares de desarrollo

En el desarrollo de aplicaciones, la calidad y la coherencia son fundamentales para garantizar una experiencia fluida y eficiente tanto para los usuarios como para los desarrolladores. Por eso, establecer estándares claros desde el inicio es esencial. Estos estándares no solo ayudan a construir interfaces atractivas y fáciles de usar, sino que también aseguran que el código sea limpio, seguro y fácil de mantener a lo largo del tiempo. Al seguir estos lineamientos, se facilita la colaboración en equipo, se optimiza el rendimiento y se crea un producto más accesible e intuitivo para todos.

A continuación, se detallan los principios y buenas prácticas que guían nuestro enfoque en el desarrollo de interface de usuario y en la codificación de la aplicación para los cuales se incluyen los siguientes principios:

- **Diseño Responsivo:** La interfaz debe adaptarse a una amplia variedad de dispositivos y tamaños de pantalla, garantizando que la aplicación sea usable tanto en dispositivos móviles como en escritorios. El diseño debe ser fluido y dinámico.
- **Accesibilidad:** La aplicación cumple y es accesible para personas con discapacidades. Esto incluye el uso adecuado de colores, contrastes, etiquetas en los elementos interactivos y la compatibilidad con lectores de pantalla.
- **Consistencia Visual:** Los elementos visuales (botones, iconos, tipografía, colores) deben ser coherentes en toda la aplicación. Las interfaces deben ser intuitivas, con elementos de navegación claros y fáciles de encontrar.
- **Interactividad:** Los botones y elementos interactivos deben ser fácilmente identificables y deben proporcionar un cambio visual (como un color o animación) al pasar el cursor o al hacer clic.

4.5.1. Interfaces

Las interfaces son los puntos de contacto que establecen un contrato que permite el intercambio de información entre elementos que forman parte de la arquitectura de un sistema de software. Estos elementos pueden ser lógicos (ej. módulos), dinámicos (ej. objetos) o físicos (ej. nodos de hardware).

Las interfaces en el contexto de arquitectura de software son esenciales para definir cómo los distintos componentes o elementos de un sistema interactúan entre sí, asegurando que la comunicación sea clara, estandarizada y predecible. En otras palabras, las interfaces actúan como contratos que determinan qué tipo de información puede ser intercambiada, cómo se realiza ese intercambio y qué expectativas deben cumplirse.

Para esto tenemos los siguientes tipos de Interfaces:

1. **Lógicas:** Definen cómo los módulos de software interactúan entre sí. Por ejemplo, una API que un módulo de software puede usar para comunicarse con otro.
2. **Dinámicas:** Involucran objetos o instancias en tiempo de ejecución. Por ejemplo, el envío de mensajes o eventos entre objetos dentro de una aplicación.
3. **Físicas:** Implican la comunicación entre hardware o nodos físicos, como la interfaz de red entre dos servidores, o el protocolo de comunicación entre una computadora y un dispositivo periférico.

Nombre	Descripción
Módulo Catálogo de Hospedaje	Interfaz lógica. Donde el cliente ve un catálogo de las habitaciones disponibles, se ven tarifas, días disponible etc..
Módulo Gestión y Registro de Reservaciones	Interfaz lógica. EL cliente realiza la reservación especificando las fechas de llegada y salida y la cantidad de huéspedes para asegurar la reserva exitosa de la villa.
Módulo Inventario de Activos por Habitación	Interfaz lógica. Donde el administrador y la persona encargada realizan consulta de los activos, incluyendo filtros para ayudar con el manejo de datos.
Módulo Gestión de empleados	Interfaz lógica. El administrador registra la información de los empleados
Módulo Control de	Interfaz lógica, que permite asignar tareas de limpieza por

Limpieza	cada habitación.
Módulo histórico de facturas	Interfaz lógica, Consulta el histórico de facturas.
Módulo Catálogo de Clientes	Interfaz lógica que registra y gestiona la información de los clientes en base de datos.
Módulo de Seguridad	Interfaz lógica, el administrador asigna roles y privilegios a cada usuario
Módulo de RRHH	Interfaz lógica, el personal registra y ve información sobre evaluaciones, vacaciones control de pagos.
Módulo de Parámetros	Interfaz lógica, donde el administrador podrá realizar cambios en la página web, tales como, cambio de idioma, métodos de pago, etc.

4.5.2. Datos de diseño

En el mundo del diseño, hay varios elementos esenciales que ayudan a crear una experiencia visual efectiva y atractiva para el usuario. Uno de los más importantes las Tipografías donde encontramos los títulos **primarios**, que son los encabezados principales de una página o sección. Estos títulos deben ser lo más destacado visualmente, ya que guían al usuario y le ayudan a entender rápidamente de qué trata el contenido. Por eso, suelen ser más grandes y tener un peso tipográfico más fuerte, para que capten la atención desde el primer momento.

Otro aspecto clave es la **cromática**, o la paleta de colores que se elige para el diseño. Los colores no solo tienen un impacto estético, sino que también juegan un papel importante en la coherencia visual del proyecto. Es fundamental seleccionar colores que se complementen entre sí y que transmitan la sensación adecuada. Además, especificar los códigos exactos de cada color en formatos como RGB, HEX o CMYK garantiza que los colores se mantengan consistentes, sin importar dónde o cómo se muestren.

En la **iconografía**, se refiere al uso de íconos y símbolos visuales para comunicar ideas o acciones de forma clara y rápida. Los íconos son como un "lenguaje universal" que permite a los usuarios entender la información sin necesidad de leer largas explicaciones. Además, cuando los íconos se usan de manera coherente, en un mismo estilo y con colores armoniosos, ayudan a mejorar la navegación y hacen que la experiencia en sitios web o aplicaciones sea mucho más fluida y agradable. En resumen, estos tres elementos, como el

título, la cromática y la iconografía, trabajan en conjunto para crear un diseño no solo bonito, sino también funcional y fácil de usar para el usuario.

Tipografías	
Títulos Primarios	Lora (cursiva)
Títulos Secundarios	Courgette (cursiva)
Texto Plano	Arial
Acentos y enlaces	Segoe ui
Cromática	
<ul style="list-style-type: none"> • Color primario: Pantone: #101820 • Color secundario: Gris: #b2b4b2 • Texto Plano: Negro: #000000 • Color de acento: No aplica. • Enlaces: Azul: #2952ff 	
Iconografía	
<p>Para los iconos de la página web se utilizará Bootstrap. Bootstrap es una biblioteca de código abierto que proporciona una amplia gama de iconos vectoriales escalables que se pueden personalizar fácilmente, Además estos iconos se modificaran por medio de CSS, para obtener una mayor personalización.</p>	

4.6. Componentes

4.6.1. Servidor de pruebas

Un servidor de pruebas es una herramienta esencial en cualquier proyecto de sistema web, ya que ofrece un espacio seguro para probar el funcionamiento del sistema antes de lanzarlo en vivo. Al usar un servidor de pruebas, los desarrolladores pueden simular condiciones reales y detectar posibles fallos o problemas de rendimiento, sin que afecten a los usuarios finales. Este entorno permite ajustar detalles del sistema, como la interacción con bases de datos, la carga de tráfico y la compatibilidad con diferentes dispositivos, asegurando que el producto final sea robusto y esté listo para el uso real.

Cuando configuramos un servidor de pruebas, es fundamental tener en cuenta aspectos clave como la CPU, la RAM, el almacenamiento y el sistema operativo. La CPU es la parte que procesa toda la información y ejecuta las tareas del sistema, por lo que su capacidad determinará qué tan rápido funciona el sistema bajo carga. La RAM influye en la

velocidad con la que se pueden manejar múltiples procesos a la vez, algo crucial cuando el sistema está en uso por varios usuarios. El almacenamiento asegura que haya suficiente espacio para guardar datos, logs y archivos sin afectar el rendimiento. Y el sistema operativo es la base de todo, ya que debe ser compatible con las tecnologías que utiliza el proyecto para garantizar su buen funcionamiento. Todos estos componentes trabajan juntos para que el servidor de pruebas sea lo más parecido posible a un entorno de producción, permitiendo detectar y corregir problemas antes de que lleguen a los usuarios finales.

Características del servidor de pruebas	
Procesador (CPU)	Múltiples núcleos, con una frecuencia de reloj mayor a 3.5 GHz.
Memoria RAM	32 GB
Almacenamiento	Disco NVMe para mejorar los tiempos de lectura y escritura de 2 Tb
Tarjeta gráfica (GPU)	Nvidia GTX 1660 Super con 6 GB de video dedicado.
Sistema operativo	Windows 11

4.6.2. Servidor de producción

El servidor de producción es el entorno donde el sistema web finalmente se despliega y es accesible para los usuarios. A diferencia del servidor de pruebas, este servidor debe ser robusto y capaz de manejar la carga real de usuarios, transacciones y datos de manera eficiente. Para garantizar su rendimiento óptimo, es crucial que el servidor de producción cuente con un hardware y software adecuados. Esto incluye un procesador potente, suficiente memoria RAM y almacenamiento adecuado, todo respaldado por un sistema operativo que ofrezca estabilidad y seguridad. Tener un servidor de producción bien configurado asegura que el sistema web se mantenga rápido, disponible y confiable en todo momento.

En cuanto al hardware del servidor, se necesita un procesador con múltiples núcleos y una frecuencia de reloj superior a 3.5 GHz para garantizar un rendimiento fluido incluso en

momentos de alta demanda. La memoria RAM de 32 GB es esencial para manejar múltiples procesos y usuarios simultáneamente sin afectar la velocidad del sistema. El almacenamiento de 1 TB en disco duro proporciona el espacio necesario para almacenar grandes volúmenes de datos, registros y archivos de forma segura. En cuanto al sistema operativo, el servidor utiliza Windows 11, que ofrece un entorno seguro y actualizado, compatible con las aplicaciones y herramientas necesarias para ejecutar el sistema web de manera eficiente. La combinación de estos recursos garantiza que el servidor de producción esté preparado para ofrecer un rendimiento confiable y sin interrupciones a los usuarios finales.

4.6.3. Base de datos

Para el diseño de la base de datos se utilizará SQL Server Management 20, que es uno de los sistemas gestores de bases de datos más populares. Entre sus múltiples ventajas, podemos destacar que SQL Server Management 20 dispone de versiones para Windows, uno de los sistemas operativos más utilizados en la actualidad, por lo que la base de datos puede funcionar correctamente en esta arquitectura.

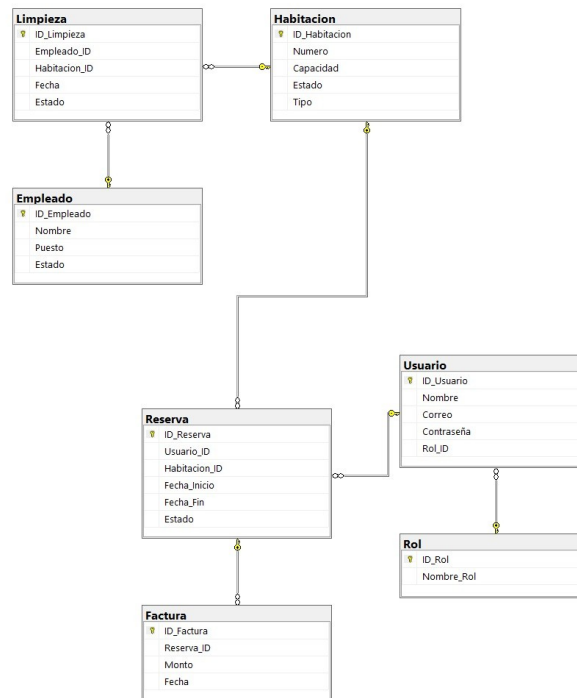
Según Castrillón-Estrada et al., (2008) Una base de datos es la organización estructurada de un conjunto de información con al menos una característica en común que permite su agrupación. Además, permite la recopilación de la información para su posterior recuperación, para lo cual generalmente ofrece un motor de búsqueda interno que utiliza características especiales de cada artículo, con el fin de lograr una rápida y eficaz ubicación.

Los datos que almacena la base de datos permanecerán de manera indefinida a fin de servir de apoyo a la investigación y planificación de nuevas iniciativas que tengan. No obstante, el administrador del sistema puede eliminar cualquier información almacenada en el sistema o modificar dicha información.

Herramientas de Construcción	
Entorno de Desarrollo	SQL Server Management Studio 20 (2022)
Lenguaje	SQL Transact-SQL
Sistema Gestor de Base de Datos	SQL Server 2022
Justificación	
<p>Se utilizara SQL server en este proyecto debido a la disponibilidad de usar SQL de forma gratuita, además de ´por su capacidad de adaptarse al sistema de Windows, para el servidor de prueba, gracias a su gran eficiencia y capacidad para la gestión de los datos de manera estructurada, las consultas las cuales son necesarias para el proyecto de VillaCoromoto permitirán actualizar, consultar e insertar datos de manera segura y rápida, lo que será compatible con los sistemas gestores de bases de datos quienes facilitaran la integración y escalabilidad, permitiendo el ac ceso a la información.</p>	

4.6.4. Diagrama entidad-relación (si aplica)

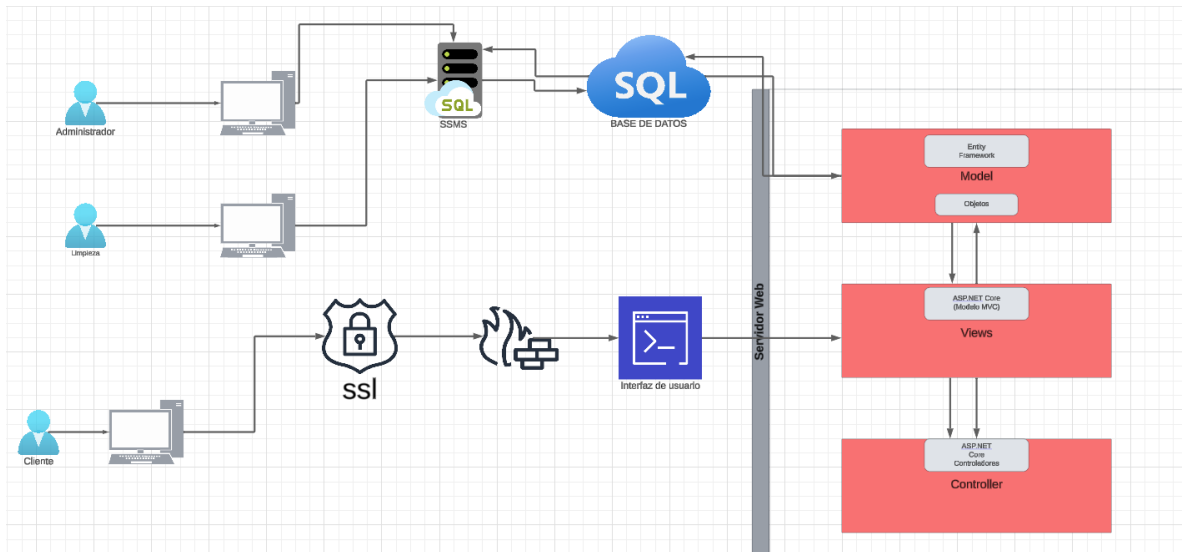
Un diagrama entidad-relación, también conocido como modelo entidad relación o ERD, es un tipo de diagrama de flujo que ilustra cómo las "entidades", como personas, objetos o conceptos, se relacionan entre sí dentro de un sistema. Los diagramas ER se usan a menudo para diseñar o depurar bases de datos relacionales en los campos de ingeniería de software, sistemas de información empresarial, educación e investigación. También conocidos como los ERD o modelos ER, emplean un conjunto definido de símbolos, tales como rectángulos, diamantes, óvalos y líneas de conexión para representar la interconexión de entidades, relaciones y sus atributos. A continuación, se presenta el diagrama entidad-relación de nuestro proyecto, creado por medio del uso de SQL Management 20, al ingresar el script planteado para nuestra página web, resultando de esta manera:



Fuente: elaboración propia. Imagen Diagrama Entidad-Relación CoromotoAccess.

4.6.5. Diagrama de infraestructura

Según Hackmetrix (2024), el diagrama de infraestructura es una representación visual que muestra cómo están organizados e interconectados los diferentes componentes de una infraestructura tecnológica. Este diagrama detalla elementos como servidores, redes, bases de datos, aplicaciones, sistemas de almacenamiento, contenedores, medidas de seguridad (como firewalls y VPN), y su interacción tanto interna como externa. A continuación, se presenta el diagrama de infraestructura para CoromotoAccess:



5. Control de Cambios y Versiones

Para gestionar los cambios y versiones del proyecto, se utilizará la plataforma GitHub, alojando el trabajo en la nube. Esta herramienta permite crear ramas individuales para cada integrante del equipo, además de un branch principal o main que actúa como el núcleo del proyecto. Cada rama estará en un repositorio, lo que facilita el trabajo independiente desde los equipos personales. Según GitHub (2024):

Un repositorio de Git abarca todos los archivos y carpetas de un proyecto, junto con el historial de revisiones de cada archivo. Se presenta como "confirmaciones", que son instantáneas de los cambios realizados a lo largo del tiempo. Las confirmaciones se organizan en distintas líneas dado que Git es un sistema de control de versiones distribuido (DVCS) y cualquier usuario con una copia del mismo puede acceder tanto al código completo como a su historial.

Esta funcionalidad resulta especialmente útil para el trabajo colaborativo, ya que cada miembro puede desarrollar en su propia rama y luego integrarla al main, lo que permite llevar un control detallado de los cambios realizados en el código.

5.1.1. GitHub

GitHub es una plataforma en la nube, con una comunidad de código abierto que ofrece varias características gratuitas que permitiéndonos en nuestro proyecto almacenar, compartir y trabajar el código elaborado para el sistema, además de que el uso de GitHub fomenta el trabajo colaborativo, ya que nos permite trabajar con diferentes usuarios, además, por medio de GitHub, podemos seguir y administrar los cambios realizados en el código. Con el uso de esta herramienta podemos presentar y compartir el trabajo a nuestros compañeros y hasta al administrador de CoromotoAccess y por último, el uso de este, permite poder migrar proyectos desde diferentes dispositivos, mediante el uso de repositorios, al agregar las cuentas de los compañeros. A continuación, se presenta la imagen de la creación del repositorio para el sistema CoromotoAccess.

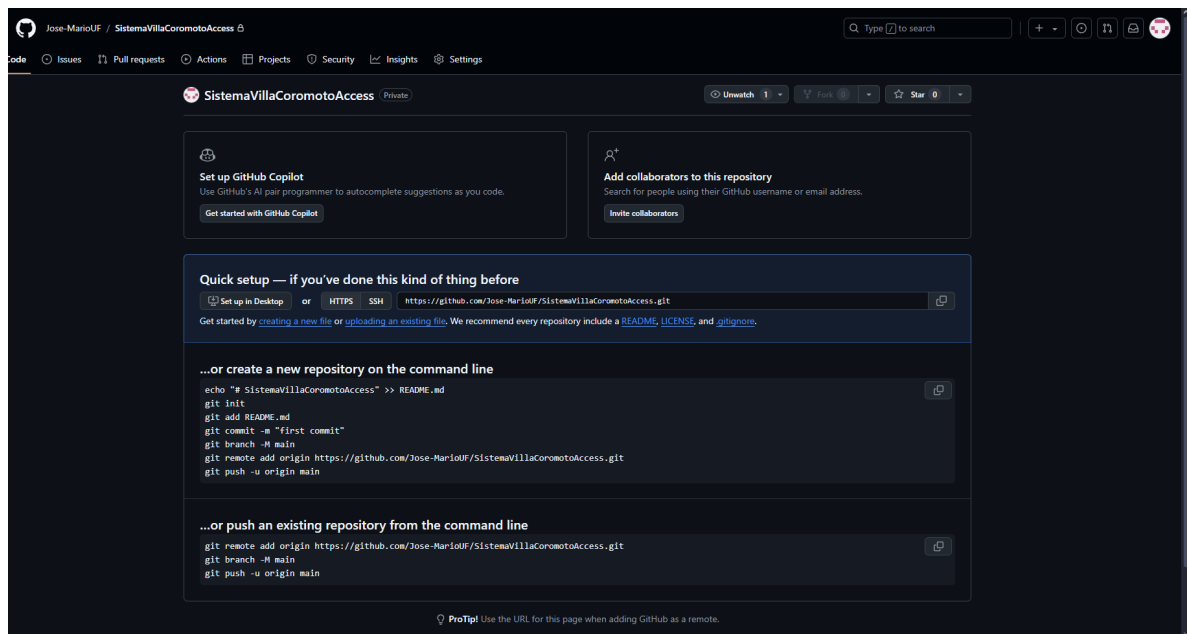


Imagen del repositorio creado.

Adicionalmente, Se adjunta link del proyecto en GitHub:
<https://github.com/Jose-MarioUF/SistemaVillaCoromotoAccess.git>

6. Referencias

AWS. (S/f). JavaScript. Amazon.com. Recuperado el 6 de diciembre de 2024, de <https://aws.amazon.com/es/what-is/javascript/>

Castrillón-Estrada, J. A., García Domínguez, J. C., Anaya Taboada, M., Rodríguez Berdugo, D., de la Rosa Barranco, D., & Caballero-Urbe, C. V. (2008). Bases de datos, motores de búsqueda e índices temáticos: herramientas fundamentales para el ejercicio médico. *Salud Uninorte*, 24(1), 95-119.

Erinstallato-ms. (s/f). *Descarga de SQL Server Management Studio (SSMS)*. Microsoft.com. Recuperado el 6 de diciembre de 2024, de <https://learn.microsoft.com/es-es/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver16>

GitHub. (2024). *Build and ship software on a single, collaborative platform*, s/f) <https://docs.github.com/es/get-started/using-git/about-git>

Hernández, R. D. (2021, junio 28). *El patrón modelo-vista-controlador: Arquitectura y frameworks explicados*. freecodecamp.org. <https://www.freecodecamp.org/espanol/news/el-modelo-de-arquitectura-view-controller-pattern/>

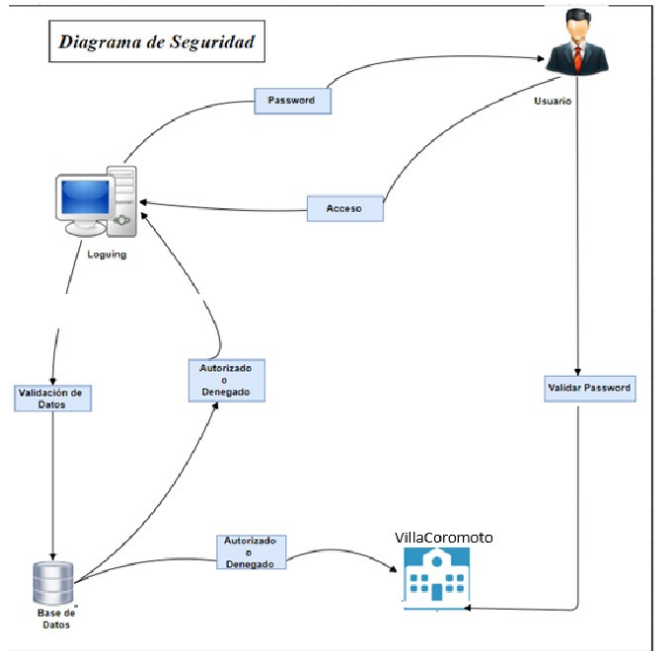
Lucidchat. (2024). *Qué es un diagrama entidad-relación*. Recuperado el 6 de diciembre de 2024, de <https://www.lucidchart.com/pages/es/que-es-un-diagrama-entidad-relacion>

MikeRayMSFT. (s/f). *SQL Server 2019: Requisitos de hardware y software - SQL Server*. Microsoft.com. Recuperado el 12 de diciembre de 2024, de <https://learn.microsoft.com/es-es/sql/sql-server/install/hardware-and-software-requirements-for-installing-sql-server-2019?view=sql-server-ver16>

Visual Studio: IDE y Editor de código para desarrolladores de software y Teams. (2024, 15 noviembre). Visual Studio. <https://visualstudio.microsoft.com/es/#:~:text=Podemos%20ayudarle.,Visual%20Studio%202022,%2C%20finalmente%2C%20publicar%20una%20aplicaci%C3%B3n>

7. Anexos

Anexo A. Diagrama de Seguridad.



Anexo B. Diagrama de flujos.

