





UNIVERSIDAD TECNOLÓGICA DE LA SELVA DIVISIÓN DE TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN

INGENIERÍA EN DESARROLLO Y GESTIÓN DE SOFTWARE

DOCUMENTACIÓN DE INTEGRADORA SPRINT 2 SISTEMA DE RESERVAS Y CITAS MÉDICAS EN LÍNEA "MED APPOINT"

Integrantes del equipo:

MATRICULA	NOMBRES
092010009	ADRIANA CECILIA MARCELIN ARAUJO
092010238	JOAHAN GUTIÉRREZ ÁLVARO
092010632	LEANDRO MONTEJO JIMÉNEZ
092010156	MARCO ANTONIO LÓPEZ GÓMEZ

Grado: 10°

Grupo: "A"

Turno: Vespertino

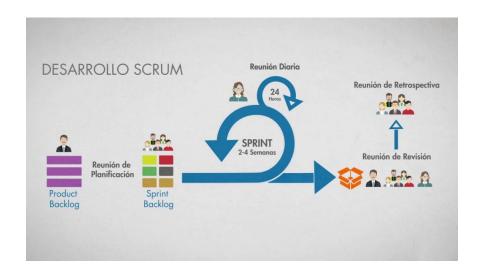
20 DE OCTUBRE DE 2023

ÍNDICE DE CONTENIDO

I. METODOLOGÍA ÁGIL	1
II. SPRINT BACKLOG	3
III. ROLES (METODOLOGÍA ÁGIL)	5
IV. PRODUCT BACKLOG	7
V. RELEASE PLAN	8
VI. DIAGRAMA DE CONTEXTO	9
VII. MODELO DE DATOS	10
Script de la base de datos	11
Diccionario de datos	14
VIII. DIAGRAMA DE ARQUITECTURA DE SOFTWARE	18
IX. PROTOTIPO VISUAL	19

I. METODOLOGÍA ÁGIL

La elección de la metodología ágil de Scrum para nuestro proyecto, se basa en diversas ventajas que esta metodología ofrece. En primer lugar, Scrum promueve la adaptabilidad y flexibilidad, lo que es crucial en un entorno de desarrollo de software donde los requisitos y las tecnologías pueden evolucionar rápidamente.



Esto nos permitirá responder de manera efectiva a cambios en el mercado o en las necesidades de los usuarios. Además, Scrum fomenta una comunicación constante y transparente entre los miembros del equipo y los stakeholders, lo que garantiza que todos estén al tanto del progreso y puedan proporcionar retroalimentación en etapas tempranas del proyecto. La entrega incremental de funcionalidades en cada iteración (sprint) nos permitirá ofrecer valor a nuestros usuarios de manera más rápida y frecuente, lo que puede ser fundamental para mantener la satisfacción y el compromiso de los clientes.

Asimismo, Scrum facilita la identificación temprana de posibles problemas o desafíos, lo que nos permite abordarlos de manera proactiva. En última instancia, esta metodología fomenta una cultura de mejora continua, lo que significa que estaremos constantemente buscando maneras de optimizar el proceso y la calidad

del producto final. Por todas estas razones, Scrum se presenta como la metodología más adecuada para garantizar el éxito de nuestro proyecto a desarrollar.

II. SPRINT BACKLOG

HISTORIA DE USUARIO	SPRINT	FECHA DE INICIO	FECHA DE FIN	ACTORES
Documentación técnica	1	07 de agosto de 2023	18 de agosto de 2023	Adriana Cecilia Marcelin Araujo Joahan Gutiérrez Álvaro Leandro Montejo Jiménez Marco Antonio López Gómez
Marco de planificación CANVAS	1	21 de agosto de 2023	31 de agosto de 2023	Adriana Cecilia Marcelin Araujo Joahan Gutiérrez Álvaro Leandro Montejo Jiménez Marco Antonio López Gómez
Plan de entregas (Release plan)	1	05 de septiembre de 2023	07 de septiembre de 2023	Joahan Gutiérrez Álvaro
Diagramas de la representación visual del sistema (De contexto y de arquitectura)	1	08 de septiembre de 2023	11 de septiembre de 2023	Joahan Gutiérrez Álvaro
Diseño del modelo de datos	2	12 de septiembre de 2023	14 de septiembre de 2023	Adriana Cecilia Marcelin Araujo
Diseño de la interfaz de usuario	2	15 de septiembre de 2023	19 de septiembre de 2023	Marco Antonio López Gómez
Creación de la base de datos	2	20 de septiembre de 2023	25 de septiembre de 2023	Leandro Montejo Jiménez
Registro y autenticación de usuarios	2	26 de septiembre de 2023	04 de octubre de 2023	Leandro Montejo Jiménez Joahan Gutiérrez Álvaro
Creación del módulo reservar cita	2	05 de octubre de 2023	20 de octubre de 2023	Leandro Montejo Jiménez

Creación del módulo consultar cita	3	25 de octubre de 2023	29 de octubre de 2023	Leandro Montejo Jiménez Marco Antonio López Gómez
Implementar la función de enviar recordatorios automáticos	3	30 de octubre de 2023	05 de noviembre de 2023	Leandro Montejo Jiménez Marco Antonio López Gómez
Creación del módulo médicos	3	06 de noviembre de 2023	10 de noviembre de 2023	Joahan Gutiérrez Álvaro Adriana Cecilia Marcelin Araujo
Plan de pruebas al sistema web	4	13 de noviembre de 2023	21 de noviembre de 2023	Joahan Gutiérrez Álvaro Adriana Cecilia Marcelin Araujo Marco Antonio López Gómez
Publicación del sistema web	4	22 de noviembre de 2023	28 de noviembre de 2023	Leandro Montejo Jiménez
Informe final	4	29 de noviembre de 2023	01 de diciembre de 2023	Adriana Cecilia Marcelin Araujo Joahan Gutiérrez Álvaro Leandro Montejo Jiménez Marco Antonio López Gómez

III. ROLES (METODOLOGÍA ÁGIL)

ROLES	DESCRIPCIÓN	PROCESOS EN LOS QUE PARTICIPA	RESPONSABILIDADES	DATOS DEL INTEGRANTE
PRODUCT OWNER	Persona responsable de asegurar que el equipo aporte valor al negocio. y priorizar el backlog del producto, tomando decisiones estratégicas para maximizar el valor del producto entregado.	 ✓ Sprint Backlog ✓ Product Backlog ✓ Release Plan ✓ Plan de pruebas 	 ✓ Definir y priorizar el backlog del producto ✓ Comunicación con los interesados ✓ Tomar decisiones ✓ Clarificar requisitos ✓ Gestionar expectativas ✓ Representar los intereses del cliente y del negocio 	Leandro Montejo Jiménez 984 205 6758 montejo1jwn@gmail.com
SCRUM MASTER	Responsable de facilitar la implementación efectiva de Scrum, eliminando obstáculos y promoviendo una cultura de mejora continua en el equipo de desarrollo.	✓ Sprint Backlog✓ Release Plan✓ Plan de pruebas	 ✓ Organizar y conducir reuniones de planificación de sprint ✓ Fomentar la colaboración y el autoorganización del equipo ✓ Promover la mejora continua ✓ Evaluar y gestionar el rendimiento del equipo 	Adriana Cecilia Marcelin Araujo 919 111 2020 marcelinaraujo02@gmail.com

SCRUM TEAM	Su función principal es la eficiente y colaborativa creación y entrega del proyecto, cumpliendo con estándares de calidad y objetivos de negocio.	 ✓ Planeación estratégica ✓ Modelo CANVAS ✓ Sprint Backlog ✓ Release Plan ✓ Diagrama de contexto ✓ Modelo de datos ✓ Diagrama de arquitectura de seconda de la función de la función de la función. ✓ Mecanismos de Seguridad Intercambio y Almacenamies Información. ✓ Validación de Datos Ingresa Usuario y Bloqueo de Iny Código Malicioso. ✓ Registro de Usuarios, Malicioso de Contraseñas ✓ Integración de Contraseñas ✓ Integración con Web Servicion y/o de Terceros, Mediante Inseguro de Información ✓ Plan de Pruebas ✓ Publicación del Software ✓ Conclusiones y Recomendacion 	ncionalidad	marcelinaraujo02@gmail.com rajo de Joahan Gutiérrez Álvaro 919 168 1223 joahan.alvaro8@gmail.com
---------------	---	---	-------------	--

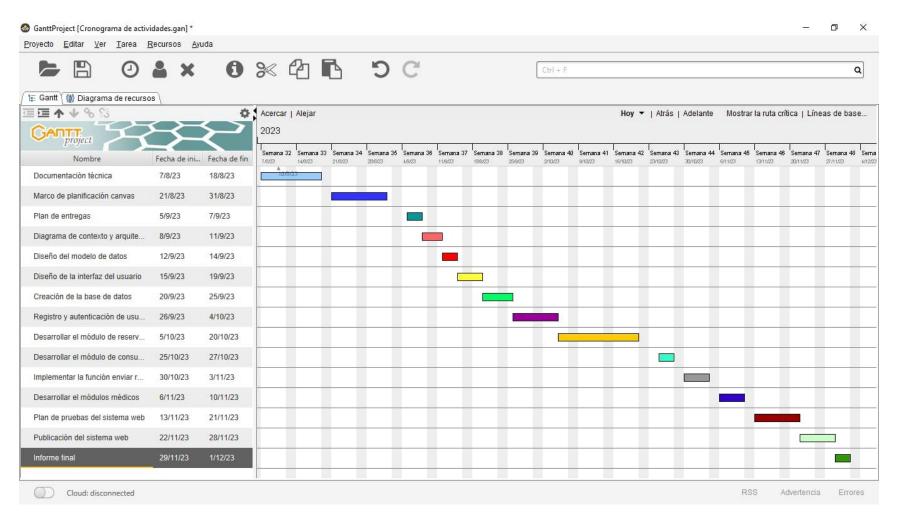
IV. PRODUCT BACKLOG

Link de invitación de acceso a nuestra plataforma empleada: https://trello.com/invite/medappoint/ATTI094ac8394e1749a034d284b9a10b58307C8DFACE

HISTORIA DE USUARIO	FECHA DE INICIO	FECHA DE FIN
Documentación técnica	07 de agosto de 2023	18 de agosto de 2023
Marco de planificación CANVAS	21 de agosto de 2023	31 de agosto de 2023
Plan de entregas (Release plan)	05 de septiembre de 2023	07 de septiembre de 2023
Diagramas de la representación visual del sistema (De contexto y de arquitectura)	08 de septiembre de 2023	11 de septiembre de 2023
Diseño del modelo de datos	12 de septiembre de 2023	14 de septiembre de 2023
Diseño de la interfaz de usuario	15 de septiembre de 2023	19 de septiembre de 2023
Creación de la base de datos	20 de septiembre de 2023	25 de septiembre de 2023
Registro y autenticación de usuarios	26 de septiembre de 2023	04 de octubre de 2023
Creación del módulo reservar cita	05 de octubre de 2023	20 de octubre de 2023
Creación del módulo consultar cita	25 de octubre de 2023	29 de octubre de 2023
Implementar la función de enviar recordatorios automáticos	30 de octubre de 2023	05 de noviembre de 2023
Creación del módulo médicos	06 de noviembre de 2023	10 de noviembre de 2023
Plan de pruebas al sistema web	13 de noviembre de 2023	21 de noviembre de 2023
Publicación del sistema web	22 de noviembre de 2023	28 de noviembre de 2023
Informe final	29 de noviembre de 2023	01 de diciembre de 2023

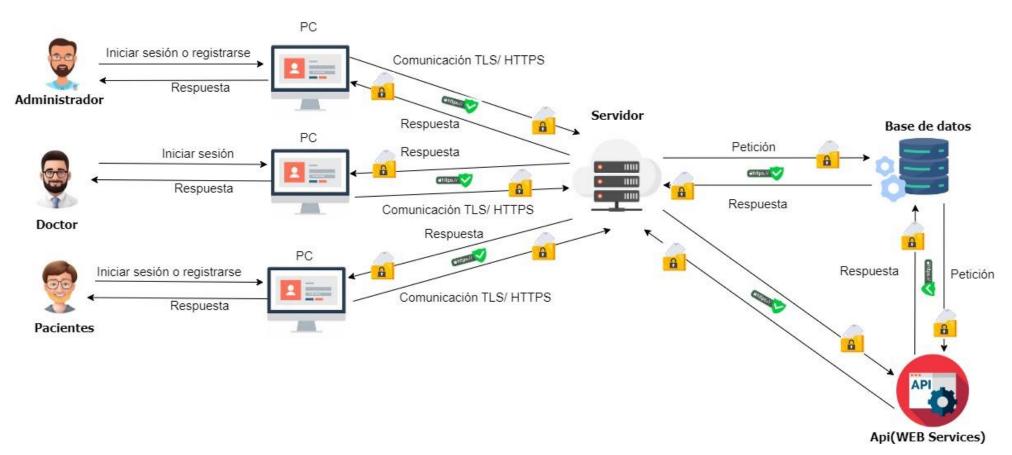
V. RELEASE PLAN

Esta secuencia de entregas nos proporciona una hoja de ruta clara, permitiéndonos una gestión efectiva de recursos y comunicación entre el equipo desarrollador, la cual sirve también para evaluar el progreso y la efectividad del desarrollo a lo largo del tiempo.



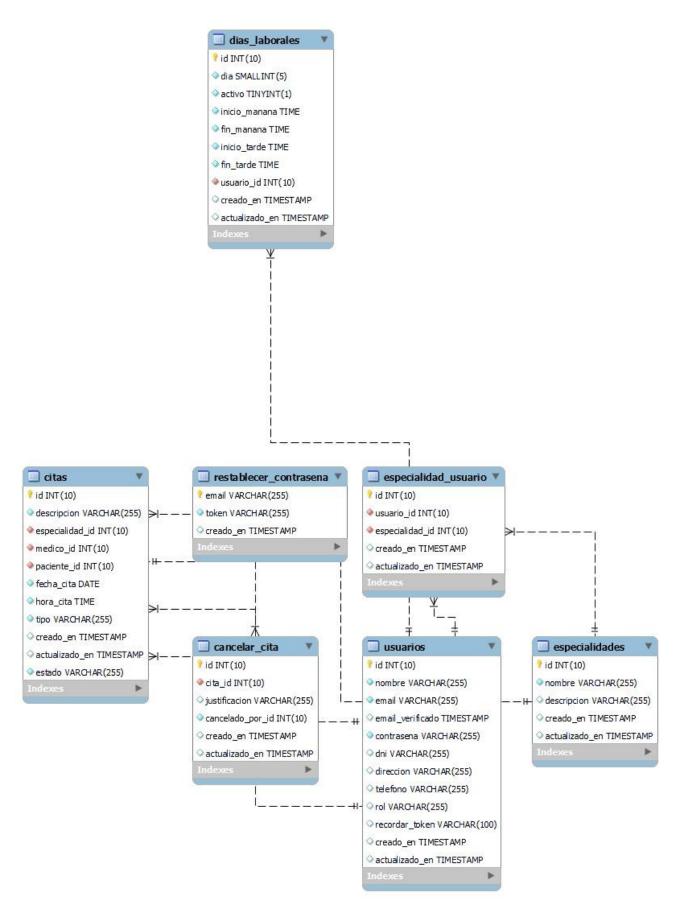
VI. DIAGRAMA DE CONTEXTO

A continuación, se presenta una representación gráfica que ilustra las interacciones entre un sistema especifico y su entorno. Muestra las entradas y salidas del sistema, así como las entidades o elementos externos con los que interactúa. Este tipo de diagrama proporciona una visión general de alto nivel, enfocándose en las conexiones esenciales sin detallar la funcionalidad interna del sistema. Esto facilita la comprensión de cómo el sistema se integra en su entorno y cómo se relaciona con otros componentes.



VII. MODELO DE DATOS

En la siguiente imagen, podemos observar el diseño del esquema o estructura de almacenamiento de la base de datos del sistema web a desarrollar. Donde cada tabla representa una entidad y sus atributos, mientras que las relaciones entre las tablas se establecen a través de claves primarias y claves foráneas. Este modelo proporciona una estructura eficiente para organizar y recuperar información de manera sistemática y coherente, lo que facilita la gestión y consulta de datos en sistemas de gestión de bases de datos relacionales.



Script de la base de datos

-- Creando la base de datos

```
CREATE DATABASE IF NOT EXISTS `citas_prueba` /*!40100 DEFAULT CHARACTER SET utf8 COLLATE utf8_spanish_ci */;
```

USE `citas_prueba`;

-- Creando la tabla especialidades

CREATE TABLE IF NOT EXISTS 'especialidades' (

'id' int(10) unsigned NOT NULL AUTO_INCREMENT,

`nombre` varchar(255) NOT NULL,

`descripcion` varchar(255) DEFAULT NULL,

`creado_en` timestamp NULL DEFAULT NULL,

`actualizado_en` timestamp NULL DEFAULT NULL,

PRIMARY KEY ('id')

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;

-- Creando la tabla usuarios

CREATE TABLE IF NOT EXISTS 'usuarios' (

'id' int(10) unsigned NOT NULL AUTO_INCREMENT,

`nombre` varchar(255) NOT NULL,

`email` varchar(255) NOT NULL,

`email_verificado` timestamp NULL DEFAULT NULL,

`contrasena` varchar(255) NOT NULL,

`dni` varchar(255) DEFAULT NULL,

`direccion` varchar(255) DEFAULT NULL,

`telefono` varchar(255) DEFAULT NULL,

'rol' varchar(255) DEFAULT NULL,

`recordar_token` varchar(100) DEFAULT NULL,

`creado_en` timestamp NULL DEFAULT NULL,

`actualizado_en` timestamp NULL DEFAULT NULL,

PRIMARY KEY ('id')

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;

-- Creando la tabla citas

CREATE TABLE IF NOT EXISTS 'citas' (

'id' int(10) unsigned NOT NULL AUTO_INCREMENT,

```
`descripcion` varchar(255) NOT NULL,
```

PRIMARY KEY ('id'),

FOREIGN KEY ('especialidad_id') REFERENCES 'especialidades' ('id') ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY ('medico_id') REFERENCES 'usuarios'('id') ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY ('paciente_id') REFERENCES 'usuarios'('id') ON DELETE CASCADE ON UPDATE CASCADE

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;

-- Creando las demás tablas con las claves foráneas correspondientes

CREATE TABLE IF NOT EXISTS `cancelar_cita` (

'id' int(10) unsigned NOT NULL AUTO_INCREMENT,

PRIMARY KEY ('id'),

FOREIGN KEY (`cita_id`) REFERENCES `citas`(`id`) ON DELETE CASCADE ON UPDATE CASCADE

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4 unicode ci;

CREATE TABLE IF NOT EXISTS 'dias_laborales' (

'id' int(10) unsigned NOT NULL AUTO_INCREMENT,

'dia' smallint(5) unsigned NOT NULL,

[`]especialidad_id` int(10) unsigned NOT NULL,

[`]medico_id` int(10) unsigned NOT NULL,

[`]paciente_id` int(10) unsigned NOT NULL,

[`]fecha_cita` date NOT NULL,

^{&#}x27;hora cita' time NOT NULL,

[`]tipo` varchar(255) NOT NULL,

[`]creado_en` timestamp NULL DEFAULT NULL,

[`]actualizado_en` timestamp NULL DEFAULT NULL,

[`]estado` varchar(255) NOT NULL DEFAULT 'Reservada',

[`]cita_id` int(10) unsigned NOT NULL,

^{&#}x27;justificacion' varchar(255) DEFAULT NULL,

[`]cancelado_por_id` int(10) unsigned NOT NULL,

[`]creado_en` timestamp NULL DEFAULT NULL,

[`]actualizado_en` timestamp NULL DEFAULT NULL,

[`]activo` tinyint(1) NOT NULL,

[`]inicio_manana` time NOT NULL,

[`]fin_manana` time NOT NULL,

```
`inicio_tarde` time NOT NULL,
```

PRIMARY KEY ('id'),

FOREIGN KEY (`usuario_id`) REFERENCES `usuarios`(`id`) ON DELETE CASCADE ON UPDATE CASCADE

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;

CREATE TABLE IF NOT EXISTS `especialidad_usuario` (

'id' int(10) unsigned NOT NULL AUTO_INCREMENT,

PRIMARY KEY ('id'),

FOREIGN KEY (`usuario_id`) REFERENCES `usuarios`(`id`) ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY ('especialidad_id') REFERENCES 'especialidades'('id') ON DELETE CASCADE ON UPDATE CASCADE

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4 unicode ci;

CREATE TABLE IF NOT EXISTS 'restablecer_contrasena' (

`email` varchar(255) NOT NULL,

PRIMARY KEY ('email')

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;

[`]fin_tarde` time NOT NULL,

[`]usuario_id` int(10) unsigned NOT NULL,

[`]creado_en` timestamp NULL DEFAULT NULL,

[`]actualizado_en` timestamp NULL DEFAULT NULL,

[`]usuario_id` int(10) unsigned NOT NULL,

[`]especialidad_id` int(10) unsigned NOT NULL,

[`]creado_en` timestamp NULL DEFAULT NULL,

[`]actualizado_en` timestamp NULL DEFAULT NULL,

[`]token` varchar(255) NOT NULL,

[`]creado_en` timestamp NULL DEFAULT NULL,

♣ Diccionario de datos

		TABLA "citas"	
Campo	Tipo de dato	Longitud	Descripción
id	int	10	Identificador único de la cita
descripcion	varchar	255	Descripción de la cita.
especialidad_id	int	10	Identificador de la especialidad médica asociada a la cita.
medico_id	int	10	Identificador del médico asignado a la cita.
paciente_id	int	10	Identificador del paciente que tiene la cita.
fecha_cita	date		Fecha en que está programada la cita.
hora_cita	time		Hora en que está programada la cita.
tipo	varchar	255	Tipo de cita.
creado_en	timestamp		Fecha y hora de la creación del registro.
actualizado_en	timestamp		Fecha y hora de la última actualización del registro.
estado	varchar	255	Estado de la cita.

	TABLA "restablecer_contraseña"				
Campo	Tipo de dato	Longitud	Descripción		
email	varchar	255	Dirección de correo electrónico asociada al reseteo de contraseña.		
token	varchar	255	Token generado para el proceso de reseteo de contraseña.		
creado_en	timestamp		Fecha y hora de creación del registro.		

		TABLA "cancelar_cita"	
Campo	Tipo de dato	Longitud	Descripción
id	int	10	Identificador único de la cita cancelada.
cita_id	int	10	Identificador de la cita original que fue cancelada.
justificacion	varchar	255	Justificación de la cancelación,
cancelado_por_id	int	10	Identificador del usuario que canceló la cita.
creado_en	timestamp		Fecha y hora de creación del registro.
actualizado_en	timestamp		Fecha y hora de la última actualización del registro.

	T.	ABLA "especialidades"	
Campo	Tipo de dato	Longitud	Descripción
id	int	10	Identificador único de la especialidad.
nombre	varchar	255	Nombre de la especialidad.
descripcion	varchar	255	Descripción de la especialidad.
creado_en	timestamp		Fecha y hora de creación del registro.
actualizado_en	timestamp		Fecha y hora de la última actualización del registro.

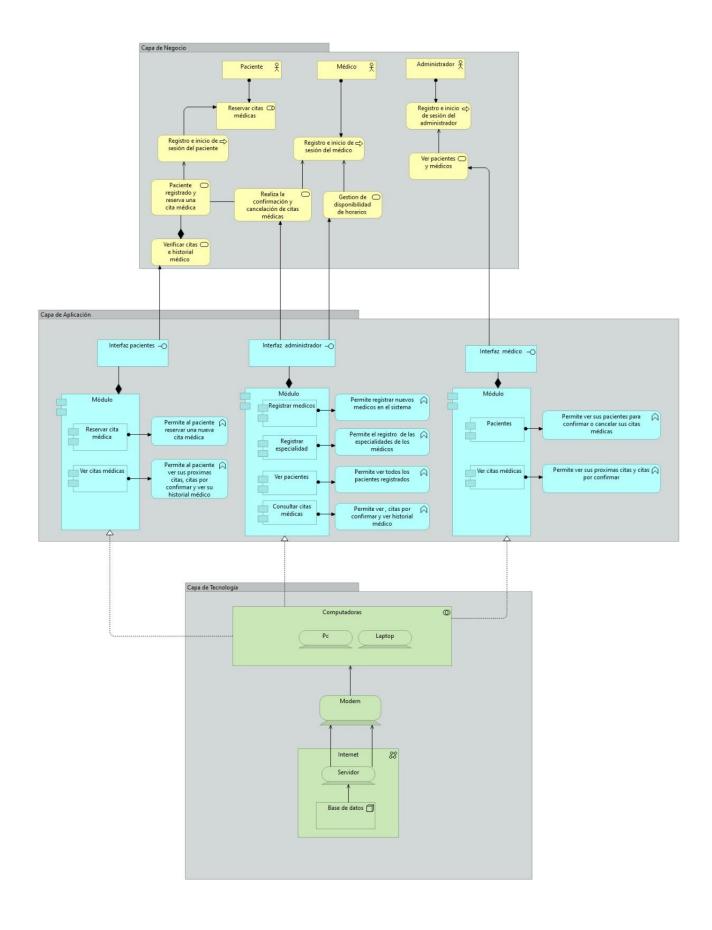
		TABLA "usuarios"	
Campo	Tipo de dato	Longitud	Descripción
id	int	10	Identificador único del usuario.
nombre	varchar	255	Nombre del usuario.
email	varchar	255	Dirección de correo electrónico del usuario.
email_verificado	timestamp		Fecha y hora de verificación del correo electrónico.
contraseña	varchar	255	Contraseña del usuario.
dni	varchar	255	Número de identificación del usuario.
direccion	varchar	255	Dirección del usuario.
telefono	varchar	255	Número de teléfono del usuario.
rol	varchar	255	Rol o tipo de usuario.
recordar_token	timestamp		Token de recordatorio de sesión.
creado_en	timestamp		Fecha y hora de creación del registro.
actualizado_en	timestamp		Fecha y hora de la última actualización del registro.

TABLA "días_laborales"				
Campo	Tipo de dato	Longitud	Descripción	
id	int	10	Identificador único del usuario.	
dia	smallint	5	Nombre del usuario.	
activo	tinyint	1	Dirección de correo electrónico del usuario.	
inicio_mañana	time		Fecha y hora de verificación del correo electrónico.	
fin_mañana	time		Contraseña del usuario.	
inicio_tarde	time		Número de identificación del usuario.	
fin_tarde	time		Dirección del usuario.	
usuario_id	int	10	Número de teléfono del usuario.	
creado_en	timestamp		Rol o tipo de usuario.	
actualizado_en	timestamp		Token de recordatorio de sesión.	

TABLA "especialidad_usuario"				
Campo	Tipo de dato	Longitud	Descripción	
id	int	10	Identificador único de la relación entre usuario y especialidad.	
usuario_id	int	10	Identificador del usuario asociado a la especialidad.	
especialidad_id	int	10	Identificador de la especialidad asociada al usuario.	
creado_en	timestamp		Fecha y hora de creación del registro.	
actualizado_en	timestamp		Fecha y hora de la última actualización del registro.	

VIII. DIAGRAMA DE ARQUITECTURA DE SOFTWARE

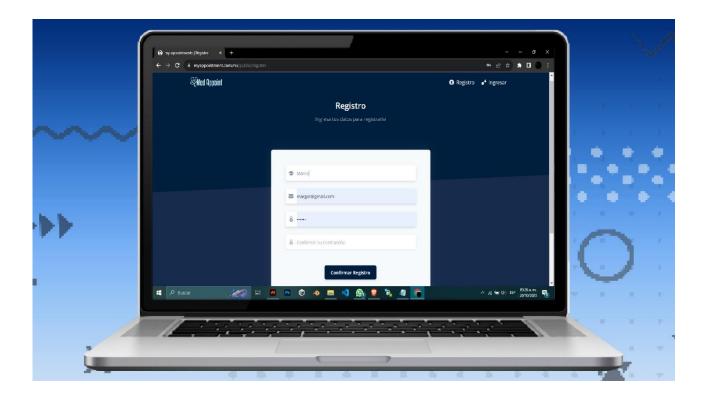
A continuación, se presenta el diagrama de arquitectura que ilustra la estructura y relaciones de los componentes de nuestro sistema de software, el cual nos proporciona una visión de alto nivel de cómo los diferentes elementos del sistema interactúan entre sí para lograr los objetivos del software. Incluye elementos como módulos, capas, interfaces, bases de datos y conexiones entre ellos, brindando una comprensión visual de la arquitectura subyacente del sistema de software.



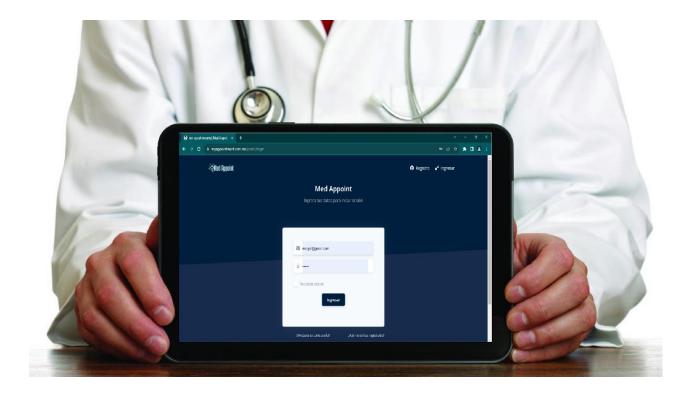
IX. PROTOTIPO VISUAL

En esta sección, se presenta el diseño visual del sistema web de reservas y citas médicas en línea, aproximado a la solución tecnológica tal y como la usará el usuario final.

-Mockup interfaz de login (registro de usuario)



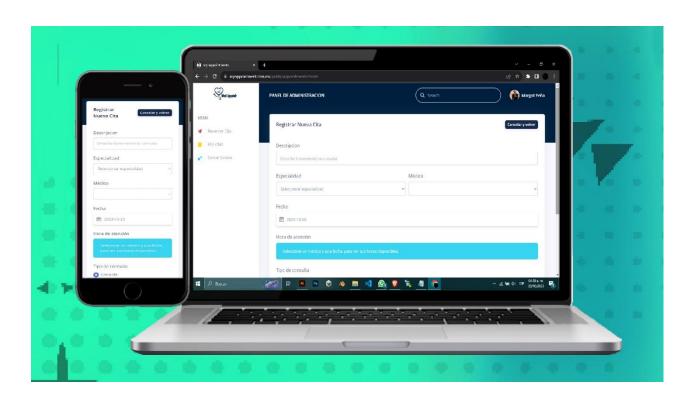
-Mockup interfaz de inicio de sesión



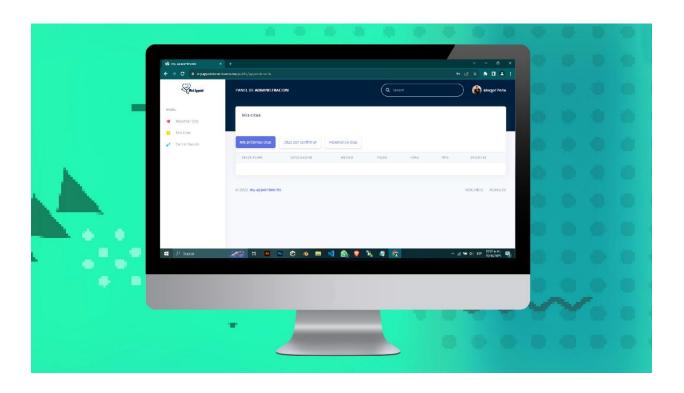
-Mockup panel de bienvenida



-Mockup formulario para agendar una cita médica



-Mockup panel de consultar cita



-Mockup sistema web responsive

