

SiMed

Ingeniería de software

Presentado a:

Carlos Andrés Parra Acevedo

Integrantes:

Johann Fernando Trigos Niño
José Esteban Betín Díaz
Katty Alexandra Peña Nieto
Johan Miguel Céspedes Ortega
Javier Alfonso Becerra Sánchez

Versión

1.0

Pontifica Universidad Javeriana

Facultad de ingeniería

Especialización en arquitectura empresarial de software

Bogotá, Colombia

2020

1. Tabla de contenido

1.	TABLA DE CONTENIDO	2
2.	LISTADO DE TABLAS	3
1.	INICIO	4
1.1.	ALCANCE:	4
1.2.	WBS.....	4
1.3.	RECURSOS	4
1.4.	CRONOGRAMA	5
1.5.	PRESUPUESTO	7
1.6.	RIESGOS.....	7
1.7.	6.1 POSITIVOS	7
1.8.	6.2 NEGATIVOS	8
2.	REQUISITOS	10
2.1.	REQUISITOS FUNCIONALES	10
2.2.	REQUISITOS NO FUNCIONALES	10
2.3.	ESTIMACIÓN- PLANNING POKER	11
3.	DISEÑO	12
3.1.	ESTILO DE ARQUITECTURA	12
3.2.	HERRAMIENTAS POR UTILIZAR	13
3.3.	DIAGRAMA DE CASOS DE USO	14
3.4.	DIAGRAMA DE CLASES	15
3.5.	DIAGRAMA DE ENTIDAD RELACIÓN	15
3.6.	DIAGRAMA DE COMPONENTES	16
3.7.	DIAGRAMA DE DESPLIEGUE.....	17

2. Listado de Tablas

Tabla 1 Recursos humanos del proyecto	5
Tabla 2 Recursos técnicos del proyecto	5
Tabla 3 Cronograma.....	6
Tabla 4 Riesgos positivos	8
Tabla 5 Riesgos negativos	9

Listado de Ilustraciones

Ilustración 1 Work Breakdown Structure	4
Ilustración 2 Costos.....	7

1. INICIO

1.1. Alcance:

SiMed es una aplicación Web que permite la localización de los centros médicos más cercanos a los usuarios de Bogotá, además contemplando la especialidad requerida y con mayor disponibilidad de atención.

1.2. WBS

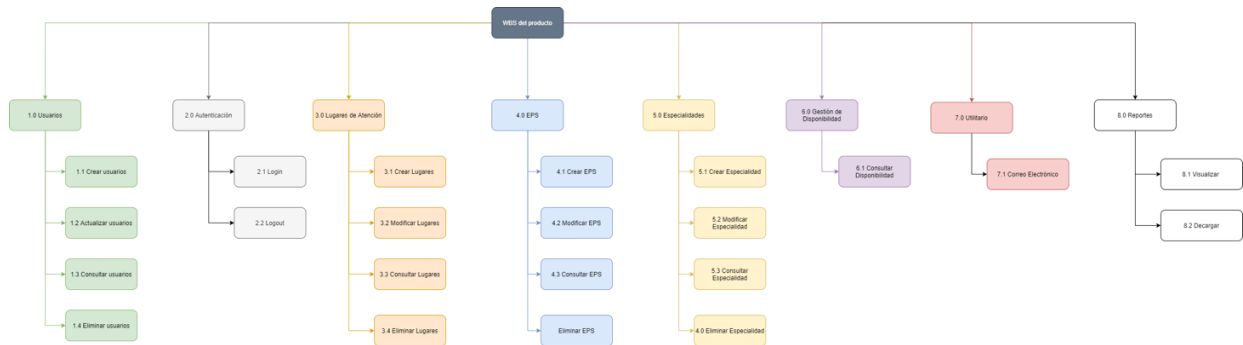


Ilustración 1 Work Breakdown Structure

1.3. Recursos

Este proceso de gestión del talento que acompañará el proceso de ejecución del proyecto debe incluir los procesos que los organizan y quienes realizan el proceso de dirección y control del proceso, para esto se tienen en cuenta las fases del proyecto donde es necesario la definición de roles, qué interrelación tienen entre ellos y las actividades de competencia que debe realizar cada rol, además de ello es necesario tener en cuenta el tiempo que se invertirá por hora/labor del proceso.

NOMBRE	ID	PERFIL	RESPONSABLE
Gerente del proyecto	GP	Líder nato que debe tener una alta capacidad de comunicación y motivación para llevar un liderazgo con el equipo de ejecución del proyecto, y además evaluar las actividades que comprenden el trabajo en equipo con los interesados.	José Betín
Líder de desarrollo	LD	Capacidad de coordinar y organizar actividades de los integrantes de desarrollo, tiene que llevar a cabo procesos de liderazgo, uso de recursos disponibles, realizar seguimiento y retroalimentación de las actividades.	Johann Trigos
Líder de configuración	LC	Rol encargado de la habilitación de las herramientas tecnológicas necesarias para todo el ciclo de vida de desarrollo de la aplicación.	Johann Trigos

Arquitecto de software	AS	Como liderazgo técnico, este debe tener la capacidad de tomar las óptimas decisiones tomando en cuenta los requisitos no funcionales del proyecto, este debe tener en cuenta la definición de tecnología y su mejora continua.	Johan Céspedes
Analista QA Senior	AQA	Para lograr la gestión de la calidad del proyecto, es necesario responsables en revisar flujos desde la perspectiva del usuario final cuestionando que podría pasar si se realiza una acción en específico. Debe tener un alto nivel de abstracción y capacidad de rastrear amenazas.	Katty
Desarrollador Front-End	FED	Expertos técnicos en el desarrollo de software del lado de la vista, responsable de la capa de presentación que implementan técnicas a base de decisiones del arquitecto, actuando con eficiencia y alto nivel de compromiso, este es un recurso muy importante debido que es el que recrea la idea principal de visualización del Software.	Javier Becerra José Betín
Desarrollador Back-End	BED	Expertos técnicos en el desarrollo de software, responsable de la lógica del sistema implementando las técnicas sujetas de las decisiones del arquitecto, actuando con eficiencia y alto nivel de compromiso, este es un recurso muy importante debido que quien desarrolla la lógica del aplicativo y permite que la vista presente la información requerida por el usuario.	Katty Johan Céspedes

Tabla 1 Recursos humanos del proyecto

Otros recursos necesarios para el desarrollo del proyecto que van a ser tenidos en cuenta para adquisición son los siguientes:

CONCEPTO	CARACTERÍSTICAS	CANTIDAD
Computador	Portátil o escritorio mínimo procesador Core i7 4ta generación +, SO a elección.	5
Internet	Hasta 30 Megs bajada y hasta 6 Megs subida de Internet.	5

Tabla 2 Recursos técnicos del proyecto

1.4. Cronograma

A continuación se presenta el cronograma general para el proyecto SiMed:

Actividad\Semana	1	2	3	4	5	6	7	8
Definición proyecto								
Conformación equipo								
Iteración 1								
Análisis								
Diseño								
Implementación								
Pruebas								
Primer DEMO								

<i>Iteración 2</i>								
<i>Análisis</i>								
<i>Diseño</i>								
<i>Implementación</i>								
<i>Pruebas</i>								
<i>Producción</i>								
<i>Refinamiento</i>								
<i>Segundo DEMO</i>								
<i>Postmortem</i>								

Tabla 3 Cronograma

1.5. Presupuesto

Nombre Proyecto: SiMed				
Sigla: SiMed				
	Tarifa Hora	Cantidad Horas	SubTotal	Total
ÉPICAS				
1. Registro de usuarios		36		\$450.000
Gerente de proyecto	45.000	2	\$90.000	
Líder de Configuración	20.000	6	\$120.000	
Arquitecto de Software	25.000	4	\$100.000	
Analista QA	18.750	6	\$112.500	
Desarrollador Front End	18.750	9	\$168.750	
Desarrollador Back End	18.750	9	\$168.750	
2. Autenticación de usuarios		8		\$170.625
Gerente de proyecto	45.000	0,5	\$22.500	
Líder de Configuración	20.000	1	\$20.000	
Arquitecto de Software	25.000	1	\$25.000	
Analista QA	18.750	1,5	\$28.125	
Desarrollador Front End	18.750	2	\$37.500	
Desarrollador Back End	18.750	2	\$37.500	
3. Lugares de atención		42		\$893.750
Gerente de proyecto	45.000	2,5	\$112.500	
Líder de Configuración	20.000	5	\$100.000	
Arquitecto de Software	25.000	5,5	\$137.500	
Analista QA	18.750	6	\$112.500	
Desarrollador Front End	18.750	14	\$262.500	
Desarrollador Back End	18.750	9	\$168.750	
4. EPS		32		\$726.250
Gerente de proyecto	45.000	2	\$90.000	
Líder de Configuración	20.000	4	\$80.000	
Arquitecto de Software	25.000	5	\$125.000	
Analista QA	18.750	5	\$93.750	
Desarrollador Front End	18.750	9	\$168.750	
Desarrollador Back End	18.750	9	\$168.750	
5. Ubicación de usuario		9		\$173.125
Gerente de proyecto	45.000	0	\$0	
Líder de Configuración	20.000	1	\$20.000	
Arquitecto de Software	25.000	0,5	\$12.500	
Analista QA	18.750	1,5	\$28.125	
Desarrollador Front End	18.750	3	\$56.250	
Desarrollador Back End	18.750	3	\$56.250	
6. Disponibilidad de atención		42		\$870.000
Gerente de proyecto	45.000	2	\$90.000	
Líder de Configuración	20.000	4	\$80.000	
Arquitecto de Software	25.000	4	\$100.000	
Analista QA	18.750	8	\$150.000	
Desarrollador Front End	18.750	14	\$262.500	
Desarrollador Back End	18.750	10	\$187.500	
7. Reportes		40		\$895.000
Gerente de proyecto	45.000	4	\$180.000	
Líder de Configuración	20.000	2	\$40.000	
Arquitecto de Software	25.000	6	\$150.000	
Analista QA	18.750	4,5	\$84.375	
Desarrollador Front End	18.750	10,5	\$196.875	
Desarrollador Back End	18.750	13	\$243.750	
8. Especialidades		37		\$788.750
Gerente de proyecto	45.000	2	\$90.000	
Líder de Configuración	20.000	4	\$80.000	
Arquitecto de Software	25.000	6	\$150.000	
Analista QA	18.750	5	\$93.750	
Desarrollador Front End	18.750	10	\$187.500	
Desarrollador Back End	18.750	10	\$187.500	

9. Postmortem				
Gerente de proyecto	45.000	4	\$180.000	
Lider de Configuración	20.000	4	\$80.000	
Arquitecto de Software	25.000	4	\$100.000	
Analista QA	18.750	2	\$37.500	
Desarrollador Front End	18.750	2	\$37.500	
Desarrollador Back End	18.750	2	\$37.500	
6. Reservas			\$3.586.143	
	Tarifa Mes	Cantidad	SubTotal	
Costos directos				
Servicios Cloud AWS	\$7.600	8	\$60.800	
Licencia Antivirus	\$12.500	5	\$187.500	
Computador Portatil	\$2.000.000	5	\$30.000.000	
			Total	
	Tarifa Mes	Cantidad	SubTotal	
Otros Costos				
2. Costos Indirectos				
Internet	\$120.000	15	\$1.800.000	
			Total	
			Total Sin IVA	
			Total Con IVA	

Ilustración 2 Costos

1.6. Riesgos

1.7. 6.1 Positivos

Riesgo	Impacto	Probabilidad de ocurrencia
--------	---------	----------------------------

Mayor uso de la aplicación por aumento de enfermedades en Bogotá	Alto	Media
Necesidad de ubicar un centro de atención con baja aglomeración	Alto	Media
Tecnología ya conocida por el equipo	Alto	Alto
Disponibilidad de recursos	Alto	Alto
Experiencia técnica	Alto	Alto
Posibilidad de apoyo de instituciones del estado	Alto	Bajo
Necesidad del producto en el mercado	Alto	Alto

Tabla 4 Riesgos positivos

1.8. 6.2 Negativos

Riesgo	Impacto	Probabilidad de ocurrencia	Mitigación
Disponibilidad de los datos de los lugares de atención	Alto	Alto	Uso de bases de datos públicas del gobierno que cuenten con información de la capacidad de las EPS (ya se encuentra base de datos del gobierno de la capacidad instalada de camas)
Falta de capacidad tecnológica de las EPS para integrarse con la aplicación	Alto	Media	Trabajar con los datos públicos de los lugares de atención asociados a las EPS cargando la información a la plataforma.
Posibilidad de entrada de competidores con el mismo mercado objetivo	Medio	Media	Establecer estrategias de diferenciación que permitan tener una ventaja competitiva, esto acompañado de una arquitectura que permita hacer cambios en pocos días para dar respuesta ante cualquier cambio en el mercado
La aplicación no tenga buena acogida	Alto	Media	Implementar estrategias de

			mercadeo e imagen para impulsar el uso de la aplicación
La normativa colombiana restrinja el uso y distribución de la aplicación	Medio	Baja	Permitir configurar la información que puede ser mostrada en caso de que se restrinja el uso de la información
Cambio en la estructura de las EPS y lugares de atención	Alto	Baja	Plantear un modelo de relación de los componentes flexible

Tabla 5 Riesgos negativos

2. REQUISITOS

2.1. Requisitos funcionales

Los requisitos funcionales se plantean a través de épicas e historias de usuario:

Épica	Descripción
Disponibilidad de atención	El sistema debe permitir al usuario visualizar la disponibilidad de los lugares de atención que fueron consultados de acuerdo con la especialidad y su ubicación, este requisito permitirá al usuario elegir y llegar al lugar con mayor disponibilidad de atención para sus necesidades medicas
Registro de usuarios	El sistema debe permitir registrar usuarios para el acceso a los servicios de la plataforma, durante el registro se diligencian algunos parámetros y son persistidos en la base de datos
Autenticación de usuarios	El sistema debe permitir la autenticación de usuarios a la plataforma.
Ubicación del usuario	El sistema debe permitir encontrar la ubicación del usuario.
EPS	El sistema debe permitir realizar la gestión de las EPS
Lugares de atención	El sistema debe permitir realizar la gestión de los lugares de atención
Especialidades	El sistema debe permitir CRUD las especialidades que tienen los centros de atención
Reportes	El sistema debe permitir visualizar reportes de uso del producto

El detalle de las historias de usuario de cada épica se puede ver en el siguiente archivo:

[Anexo historias de usuario](#)

2.2. Requisitos no funcionales

1. Adecuación funcional.

- Completitud funcional: el sistema debe cubrir todas las tareas y objetivos del producto planteado

2. Usabilidad.

- Estética de la interfaz de usuario: el sistema debe agradar y satisfacer la interacción con el usuario en cualquier dispositivo por el que se conecte.

3. Portabilidad.

- Adaptabilidad: La aplicación web debe poseer un diseño “Responsive” para garantizar una adecuada visualización en navegadores de diferentes dispositivos (computadores personales, dispositivos tableta y teléfonos inteligentes).
- Se debe confirmar el correcto funcionamiento en el navegador Chrome.

4. Seguridad

- En la interacción entre el Back-end y Front-end debe intermediar un JWT

5. Mantenibilidad

- El aplicativo estará dividido en componentes para poder tener una modificación más rápida, efectiva y eficiente.
- El sistema se dividirá en componentes reutilizables.

2.3. Estimación- PLANNING POKER

Issue Key	Summary	Story Points
1	Registrarme en el sistema	5
2	Modificar mis datos de EPS, Nombres o Apellidos, documento de identidad	3
3	Consultar la información de usuario	3
4	Inactivar el usuario	2
5	Recibir el correo de notificación de registro	5
6	Iniciar Sesión	2
7	Cerrar Sesión	2
8	Llevar crear nuevos lugares de atención	5
9	Editar lugares de atención	5
10	Consultar lugares de atención	13
11	Consultar lugares de atención administrador	5
12	Eliminar lugares de atención	3
13	Registrar en el sistema las nuevas EPS	5
14	Modificar los datos de la EPS	5
15	Consultar la información de EPS	3
16	Inactivar las EPS	3
17	dar permisos de ubicación a la aplicación	8
18	visualizar la disponibilidad en el listado de lugares de atención	8
19	ver la disponibilidad del lugar seleccionado	13
20	ver el reporte de usuarios nuevos del último mes	5
21	ver el reporte de la cantidad de usuarios que ingresaron a la aplicación del último mes	5
22	ver el reporte de usuarios nuevos para el rango de fechas seleccionados	5
23	ver el reporte de la cantidad de usuarios que ingresaron a la aplicación para el rango de fecha seleccionado	5
24	Crear Especialidades Médicas	5
25	Editar Especialidades Médicas	5
26	Consultar Especialidades Médicas	3
27	Consultar Especialidades Médicas	2
28	Eliminar Especialidades Médicas	2

3. DISEÑO

3.1. Estilo de arquitectura

La arquitectura en capas es un estilo arquitectónico que nos permite promover la descomposición de tareas en grupos y responsabilidades, cada uno de estos grupos tiene un nivel de abstracción de acuerdo con su tarea, de tal manera que sea individualizar tareas y responsabilidades dentro de SiMed.

Así mismo es importante resaltar que se empleará el patrón cliente servidor, este patrón nos permitirá ejercer mayor control en la etapa inicial del proyecto, así mismo se espera evitar situaciones como la redundancia o inconsistencia de información en las bases de datos, por lo cual para un comienzo resulta benéfico emplear este tipo de patrones.

Tomando en cuenta el alcance de la aplicación a desarrollar definimos Elementos, relaciones, responsabilidades y colaboración para la implementación de SiMed, la implementación permite hacer una descomposición entre capas de vista, negocio y modelo, esto mejora en mantenibilidad, disminuye el nivel de complejidad de la implementación y favorece los tiempos de desarrollo por cuanto se espera hacer la entrega del aplicativo con un máximo de dos meses dando cumplimiento con los requisitos funcionales y no funcionales propuestos por el equipo de trabajo.

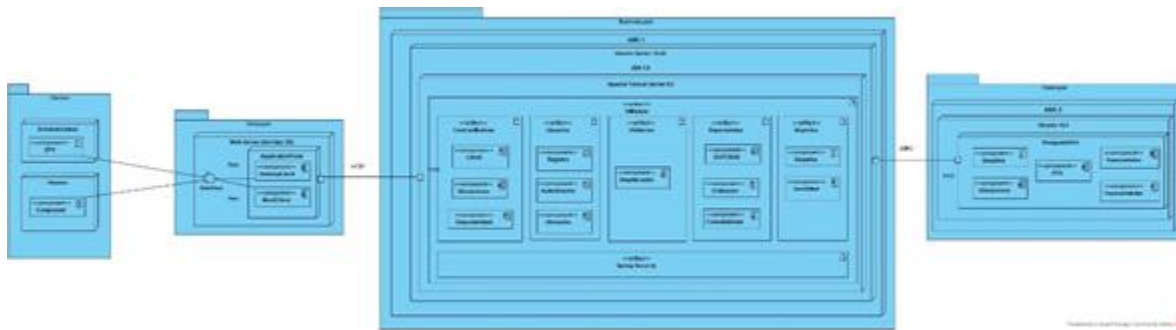


Ilustración 3 Diagrama de arquitectura en Capas SiMed

Capa de presentación: se empleará un servidor AWS 0, con un sistema operativo Ubuntu Server 18.04, allí se alojará, el *frontend* de la aplicación, a futuro se pretende desplegar el servicio de aplicación móvil para usuarios finales.

Capa de negocio: será implementada en un servidor AWS 1 con sistema operativo Ubuntu Server 18.04, JDK 1.8, servidor embebido de aplicaciones TomCat 8.0, se empaquetan servicios de aplicación, divididos en módulos centros médicos, usuarios, utilitarios, especialistas y reportes, por último, se implementa capa de seguridad de spring (Spring Security).

Capa de datos: será implementada en un servidor AWS 2 con sistema operativo Ubuntu Server 18.04, allí se alojará base de datos PostgresDB12, para la persistencia de datos de aplicación.

3.2. Herramientas por utilizar

Para la implementación de SiMed emplearan las siguientes herramientas que permitirían realizar lo planteado en la arquitectura.

Herramienta	Descripción
RAML	Para el desarrollo de los contratos, este ofrece una herramienta de código abierto para su diseño.
Node.js	Node.js es un entorno JavaScript de lado de servidor que utiliza un modelo asíncrono y dirigido por eventos. Soporta protocolos TCP, DNS y HTTP, importantes para el desarrollo de la SiMed.
Angular	Framework de desarrollo para JavaScript creado por Google. La finalidad de Angular es facilitarnos el desarrollo de SiMed, al tratarse de una aplicación web, proporciona herramientas para trabajar con los elementos de una web de una manera más sencilla y óptima.
Postgres	Sistema de gestión de bases de datos relacionales, su licencia es de código abierto por lo cual es soportado por una comunidad de desarrolladores.
Spring Boot	Framework de desarrollo, el cual será empleado para el desarrollo de apis con servicios REST no lo permite hacer de manera sencilla mediante sus anotaciones. Además de ello permite hacer uso de clientes REST y SOAP.
Java	Lenguaje de programación orientado a objetos, especialmente utilizado para el desarrollo de aplicaciones empresariales.

Tabla 6 Herramientas del desarrollo

3.3. Diagrama de casos de uso

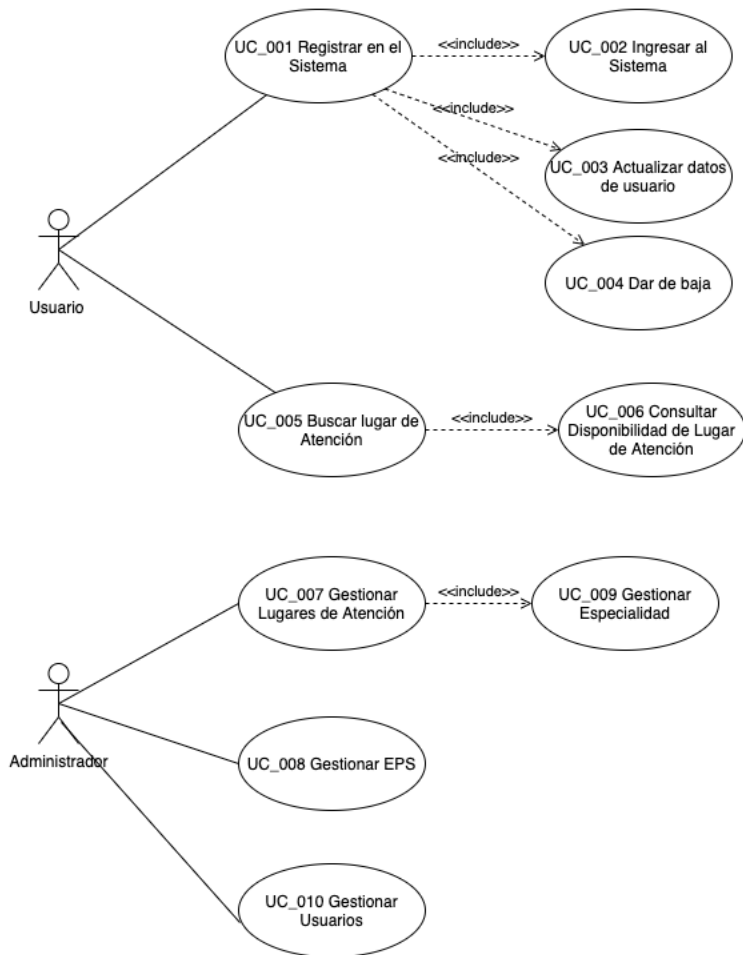


Ilustración 4 Casos de Uso

3.4. Diagrama de clases

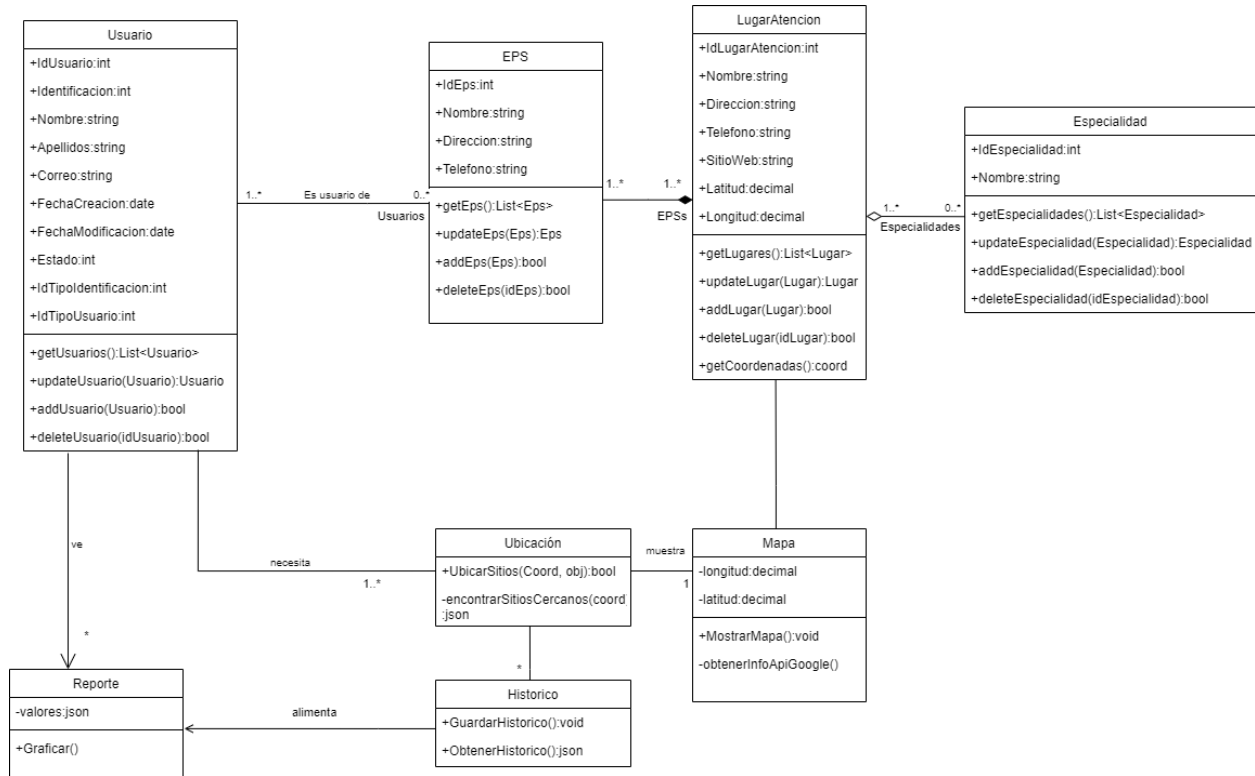


Ilustración 5 Diagrama de clases

3.5. Diagrama de entidad relación

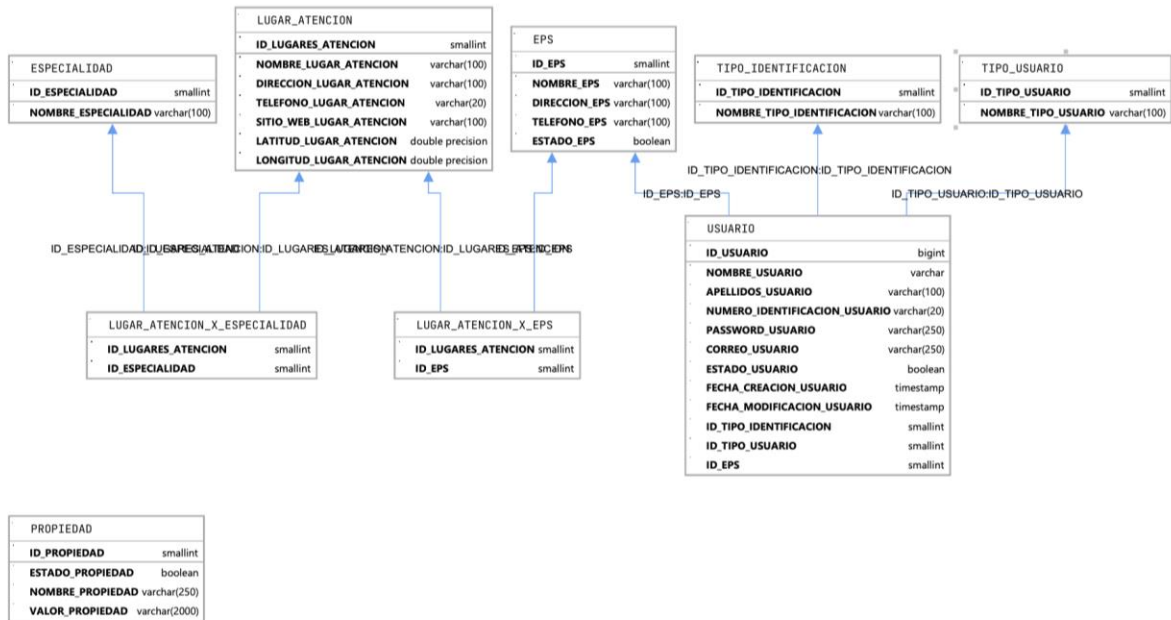


Ilustración 6 Diagrama Entidad Relación

3.6. Diagrama de componentes

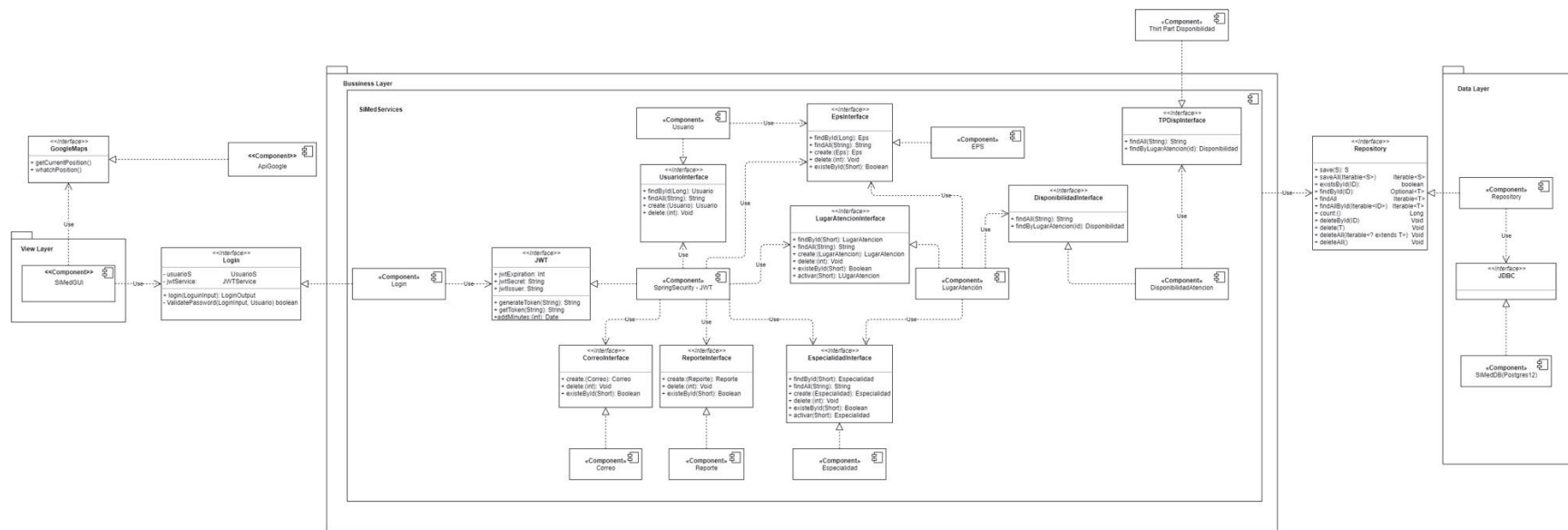


Ilustración 7 Diagrama de componentes

3.7. Diagrama de despliegue

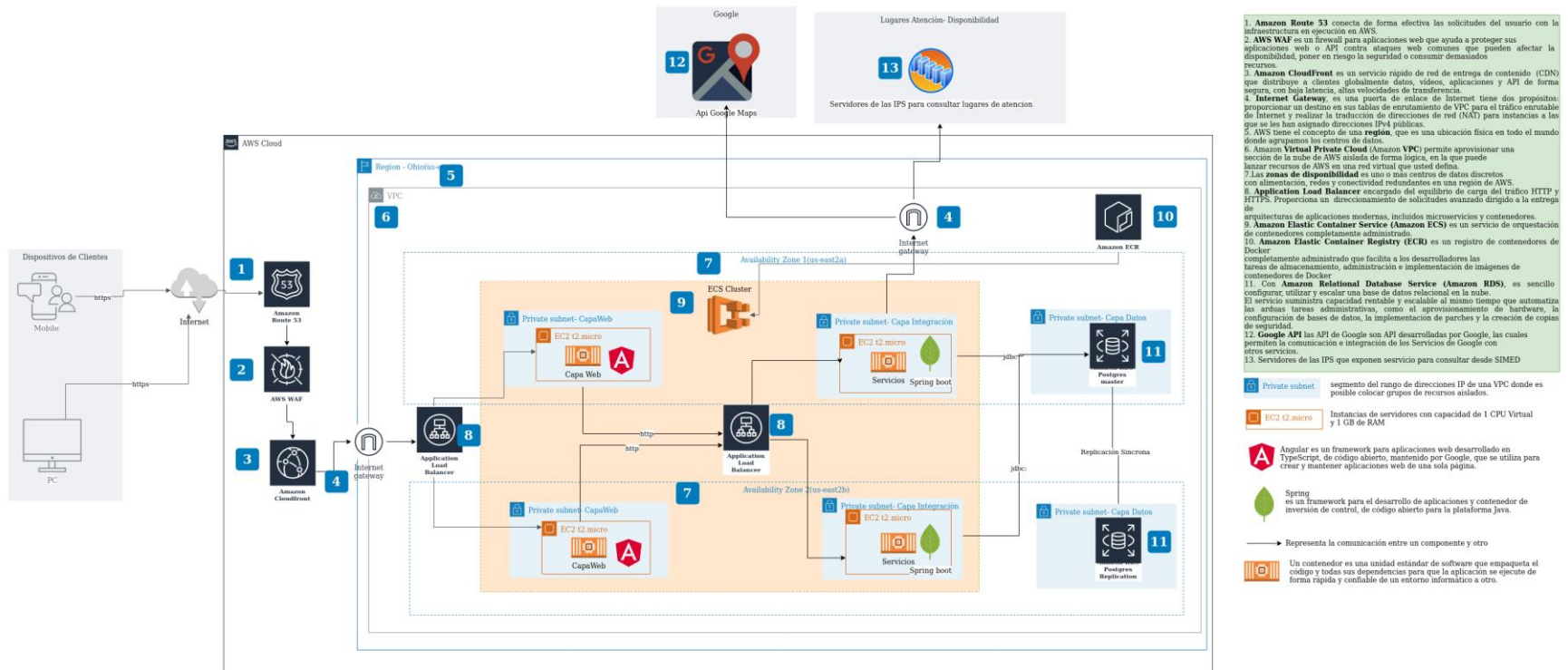


Ilustración 8 Diagrama de despliegue