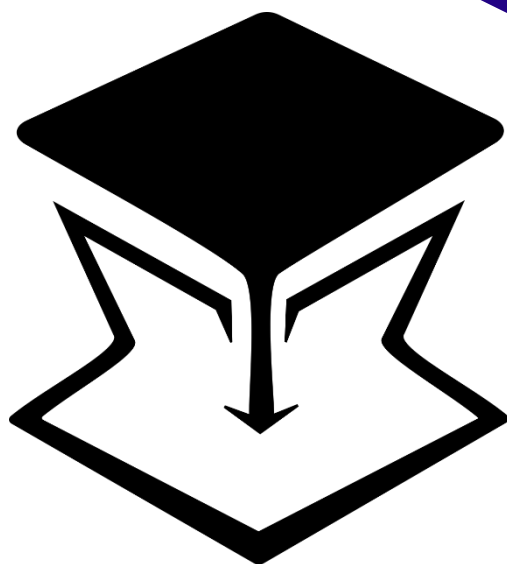




SIGMA

Arquitectura del Sistema SIGMA

3-BYTE Team

Adrián PORTILLO, Glenda FLORES, Franco ROSAS

Laboratorio de Desarrollo de Software

Unidad Académica Río Gallegos

Universidad Nacional de la Patagonia Austral

3-BYTE
TEAM



Un modelo es un conjunto de herramientas conceptuales para describir datos, sus relaciones, su significado y sus restricciones de consistencia.

Es el proceso de analizar los aspectos de interés para una organización y la relación que tienen unos con otros.

Resulta en el descubrimiento y documentación de los recursos de datos del negocio.

Es una tarea difícil, bastante difícil, pero es una actividad necesaria cuya habilidad solo se adquiere con la experiencia.

Tabla de contenido

<i>Introducción</i>	<i>4</i>
<i>Propósito</i>	<i>4</i>
<i>Alcance</i>	<i>4</i>
<i>Referencias</i>	<i>4</i>
<i>Panorama General</i>	<i>5</i>
Representación Arquitectónica	5
Objetivos Arquitectónicos y Restricciones	5
<i>Objetivos Generales</i>	<i>6</i>
<i>Objetivos Específicos</i>	<i>6</i>
<i>Descripción de Procesos</i>	<i>6</i>
Vista de Caso de Uso	7
<i>Descripción de los Actores</i>	<i>7</i>
Vista de Componentes.....	7
Vista de Procesos.....	8
Vista de Capas	9
Vista de Despliegue	10

Arquitectura del Sistema

Introducción

Este documento de Arquitectura del sistema tiene como finalidad proveer un panorama completo del diseño, estructura y comportamiento del sistema para el proyecto SIGMA. El cual es una herramienta que permitirá la realización de la estimación para proyectos de desarrollo de software, a través de Puntos de Caso de Uso (PCU) y Puntos de Historia de Usuario (PHU), y está diseñado para facilitar la gestión y análisis de los recursos y tiempos involucrados en proyectos ágiles. El documento detalla la arquitectura desde una perspectiva técnica, los cuales se ajustan a los requisitos del proyecto y a las necesidades específicas de los usuarios.

Propósito

El propósito del presente documento es servir como referencia principal para el diseño arquitectónico de SIGMA. Se detallan los componentes más importantes del sistema, sus relaciones, y los aspectos técnicos críticos que garantizan la funcionalidad y rendimiento del mismo. Este documento está dirigido principalmente a los desarrolladores y equipos técnicos que participarán en la implementación y mantenimiento del sistema.

Alcance

El alcance de este documento incluye la descripción de los principales componentes del sistema SIGMA, sus interacciones y las decisiones técnicas fundamentales que influyen en su comportamiento. Este documento afecta a todas las áreas relacionadas con el desarrollo, pruebas e implementación del sistema, y está alineado con los requerimientos funcionales y no funcionales previamente definidos en la fase de análisis.

Referencias

Diseño - Modelo 4+1 - Documento 1 (Software Architecture Document). Disponible en:
<https://proyecto-semesteral.readthedocs.io/en/latest/6%20-%20Design.html>

Arquitectura de software. Dirección de gestión del ordenamiento social de la propiedad. Disponible en: <https://www.ant.gov.co/wp-content/uploads/2022/12/INTI-PT-002-DOCUMENTO-ARQUITECTURA-DE-SOFTWARE-1.pdf>

Elaboración de aplicación informática para la precalificación y negociación de precios en el marco de la negociación conjunta de precios y compra de medicamentos para Centroamérica y República dominicana. Disponible en:
https://intercoonecta.aecid.es/Gestin%20del%20conocimiento/Manual%20T%C3%A9cnico%20Sistema%20Precalificaci%C3%B3n%202011_.pdf

Documento de Arquitectura Del Software de TecnoHelp. Disponible en:
<https://repositorio.uniajc.edu.co/server/api/core/bitstreams/a3f0e265-d67e-43b6-8c91-7f7c68179cbe/content>

Panorama General

Este documento está estructurado en varias secciones que cubren los aspectos claves de la arquitectura del sistema SIGMA. Iniciando con la introducción, propósito y alcance, seguido por la descripción de la arquitectura, incluyendo vistas de casos de uso, lógica, procesos e implementación. También abarca los objetivos arquitectónicos, restricciones, y los diagramas UML que ilustran la estructura del sistema.

Representación Arquitectónica

La arquitectura del sistema SIGMA se representa mediante varias vistas clave que cubren diferentes aspectos del sistema.

Casos de Uso: Se describe cómo interactúan los actores (Project Managers, Desarrolladores, Invitado, Administrador). Este es un punto clave para SIGMA ya que se usa para gestionar estimaciones, las cuales se pueden basar en Puntos de Historia de Usuario o Puntos de Caso de Uso, dependiendo de la metodología seleccionada.

Vista Lógica: Se enfoca en cómo se organizan los componentes internos del sistema. SIGMA tiene módulos como la gestión de estimaciones, proyectos, usuarios. Aquí se detalla cómo se estructuran y comunican.

Vista de Procesos: Describe cómo se gestionan varias operaciones al mismo tiempo, como cuando se están creando estimaciones, generando reportes o procesando datos en paralelo

Objetivos Arquitectónicos y Restricciones

Los principales objetivos arquitectónicos de SIGMA incluyen que sea un diseño escalable, reutilizable y seguro, que permita el crecimiento del sistema a medida que se utilice en diferentes proyectos.

Las restricciones clave incluyen:

- **Portabilidad:** El sistema debe ser fácil de desplegar en diferentes entornos.
- **Cumplimiento de especificaciones de seguridad:** Debe garantizar que los datos del proyecto estén protegidos y que el acceso esté controlado según roles claramente definidos, como el Administrador, Project Manager, Desarrollador y Usuario.

Restricciones adicionales:

- El sistema debe ser escalable para soportar un creciente número de usuarios y proyectos.
- Rendimiento: Debe proporcionar tiempos de respuesta eficientes.

Objetivos Generales

El objetivo general de SIGMA es mejorar el proceso de estimación en proyectos de software mediante el uso de Puntos de Caso de Uso (PCU) y Puntos de Historia de Usuario (PHU). El sistema busca optimizar la gestión del tiempo y los recursos involucrados, brindando a los usuarios una herramienta eficiente que reemplace las metodologías manuales, como plantillas de Excel, y mejore la toma de decisiones en la planificación y seguimiento de proyectos.

Objetivos Específicos

Entre los objetivos específicos de SIGMA destacan:

- Desarrollar un sistema flexible y escalable que permita gestionar estimaciones de proyectos en diferentes etapas de desarrollo.
- Implementar funcionalidades que faciliten la creación de reportes y gráficos comparativos de las estimaciones.
- Garantizar que el sistema soporte diferentes roles de usuario (Administrador, Project Manager, Desarrollador e Invitado), permitiendo el acceso adecuado a las funcionalidades según los permisos asignados.
- Asegurar que el sistema integre un panel de control visual para facilitar la visualización y el seguimiento del progreso del proyecto.

Descripción de Procesos

SIGMA incorpora procesos claves que permiten la gestión eficiente de proyectos. Estos incluyen la creación y modificación de estimaciones, la generación de reportes y gráficos, y la asignación de tareas a los miembros del equipo.

Los procesos en SIGMA se enfocan en cómo los usuarios interactúan con el sistema y cómo fluye la información entre ellos y los distintos componentes del software. Esto asegura que los datos se gestionen correctamente y que las estimaciones se controlen de forma eficiente durante cada fase o iteración del proyecto.

Vista de Caso de Uso

Los casos de uso clave en SIGMA representan las funcionalidades principales del sistema y abarcan aspectos como la gestión de proyectos, estimación de esfuerzos por Puntos de Caso de Uso (PCU) y Puntos de Historia de Usuario (PHU), la generación de reportes y gráficos y la gestión de usuarios. Cada caso de uso refleja cómo los actores interactúan con el sistema, y cómo estas interacciones conforman la arquitectura del sistema. Estos casos de uso son fundamentales para garantizar que el sistema cumpla con los requisitos de estimación y control de proyectos en tiempo real.

Descripción de los Actores

Los actores principales de SIGMA incluyen al Administrador, el Project Manager, el Desarrollador y el Invitado, cada uno con roles y responsabilidades específicas:

- **Administrador:** Responsable de la gestión global del sistema, incluyendo la creación y eliminación de proyectos, así como la gestión de usuarios y permisos.
- **Project Manager:** Encargado de crear y modificar estimaciones, asignar tareas y generar reportes de progreso, supervisando el desarrollo del proyecto.
- **Desarrollador:** Registra el tiempo dedicado a las tareas y sigue el progreso de sus asignaciones dentro del proyecto.
- **Invitado:** Tiene acceso limitado al sistema, pudiendo visualizar reportes y gráficos sin poder realizar modificaciones.

Vista de Componentes

La arquitectura lógica de SIGMA está basada en una estructura de componentes interrelacionados, organizados en módulos.

En la parte del cliente, los usuarios interactúan con el navegador, el cual procesa las interfaces diseñadas en HTML, CSS y JavaScript. Estas interfaces envían solicitudes al servidor de aplicaciones, donde el código PHP gestiona la lógica del sistema.

Por un lado, el servidor de aplicaciones interactúa con una base de datos MySQL para almacenar y recuperar la información necesaria. Por otro lado, también se conecta a servicios externos, como la API de autenticación de Google, para permitir un acceso seguro mediante Google Auth.

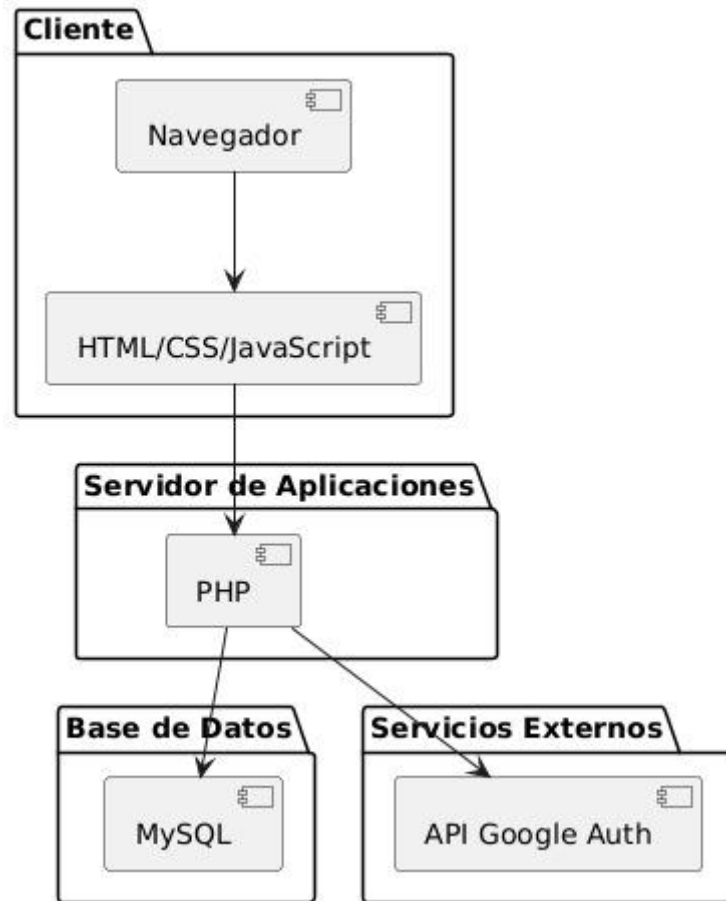


Imagen 1: Vista de componentes de SIGMA

Vista de Procesos

El sistema SIGMA está compuesto por procesos que interactúan entre sí para gestionar las estimaciones y la administración de proyectos. Estos procesos se dividen en:

- **Procesos ligeros:** Incluyen tareas que requieren menos recursos, como la validación de los datos ingresados por los usuarios (creación de estimaciones, asignación de permisos).
- **Procesos pesados:** Incluyen operaciones más complejas que requieren mayor capacidad de procesamiento, como la generación de reportes y gráficos.

Los procesos se comunican mediante el intercambio de mensajes y eventos, utilizando hilos de control para gestionar múltiples solicitudes al mismo tiempo sin afectar el rendimiento del sistema.

Vista de Capas

La arquitectura de SIGMA está organizada en varias capas que garantizan la modularidad y separación de responsabilidades. Estas capas permiten una mayor flexibilidad y mantenimiento del sistema, asegurando que cualquier cambio en una capa no afecte negativamente a las otras. Las principales capas son:

- **Capa de Presentación:** Esta capa contiene la interfaz de usuario, implementada en tecnologías web, que interactúa con los usuarios finales (Administrador, Project Manager, Desarrollador e Invitado). Los componentes clave en esta capa incluyen formularios para la creación de proyectos, visualización de estimaciones y generación de reportes.
- **Capa de Lógica de Negocio:** Aquí se encuentra la funcionalidad principal de SIGMA, como la validación de estimaciones, el cálculo de puntos de caso de uso e historia de usuario, y la generación de gráficos y reportes. También es responsable de la aplicación de los roles y permisos.
- **Capa de Persistencia:** responsable del almacenamiento de datos en bases de datos relacionales. Esta capa incluye componentes que manejan el acceso a la información de proyectos, estimaciones y usuarios, garantizando que los datos estén disponibles para su consulta o modificación en cualquier momento.

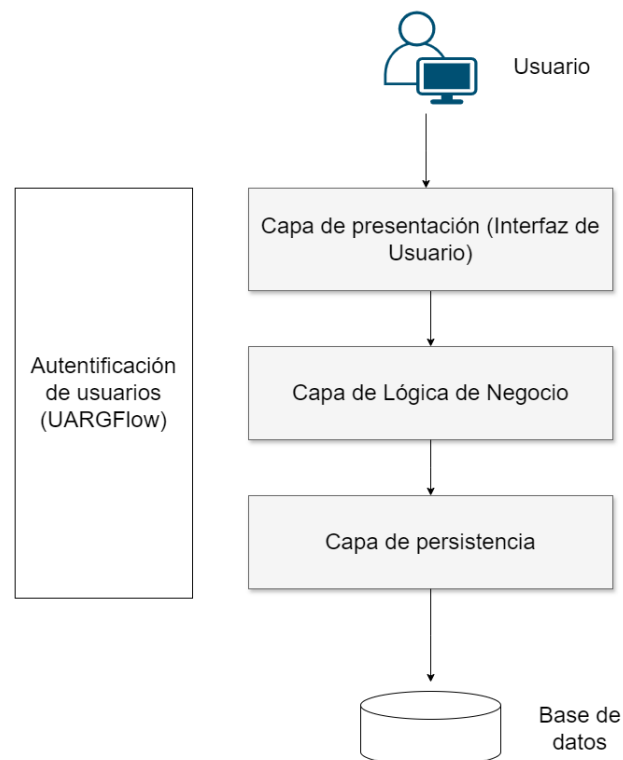


Imagen 2: Vista de capas de SIGMA

Vista de Despliegue

En el diagrama de despliegue que se muestra, se presenta la arquitectura de una aplicación web que será ejecutada en un entorno de producción utilizando contenedores Docker.

La aplicación está compuesta por tres contenedores principales:

1. **Contenedor PHP:** Este contenedor contiene la aplicación web escrita en PHP, la cual es el núcleo del sistema.
2. **Contenedor de Base de Datos:** Este contenedor utiliza MySQL como gestor de base de datos para almacenar toda la información relacionada con la aplicación.
3. **Contenedor Nginx:** Nginx actúa como servidor web y es el responsable de recibir las solicitudes externas y redirigirlas internamente al contenedor PHP.

Todos estos contenedores están conectados a través de una **red interna de Docker**. Esta red interna permite la comunicación entre los contenedores de manera segura y eficiente, sin necesidad de exponer los servicios directamente al exterior.

El **contenedor Nginx** es el único que se expone al exterior. Es decir, las solicitudes que llegan desde el Internet son gestionadas inicialmente por el contenedor Nginx, que está configurado para comunicarse con el **host del servidor** a través del puerto que se haya configurado en Docker. Luego, este contenedor redirige las solicitudes hacia el contenedor PHP para procesar la lógica de la aplicación.

Además, en la capa externa del sistema, hay un **firewall** que filtra el tráfico que llega desde Internet hacia el servidor. Este firewall garantiza que solo el tráfico autorizado pueda acceder al sistema, protegiendo así la infraestructura de posibles ataques o accesos no deseados.

En resumen, el flujo de funcionamiento es el siguiente:

- Los usuarios acceden a la aplicación a través de Internet.
- El tráfico es filtrado por un firewall.
- Las solicitudes llegan al servidor, donde Docker está en ejecución.
- El contenedor Nginx recibe las solicitudes y las redirige hacia el contenedor PHP.
- El contenedor PHP, si es necesario, interactúa con la base de datos MySQL.
- Finalmente, las respuestas se devuelven al usuario a través del mismo flujo.

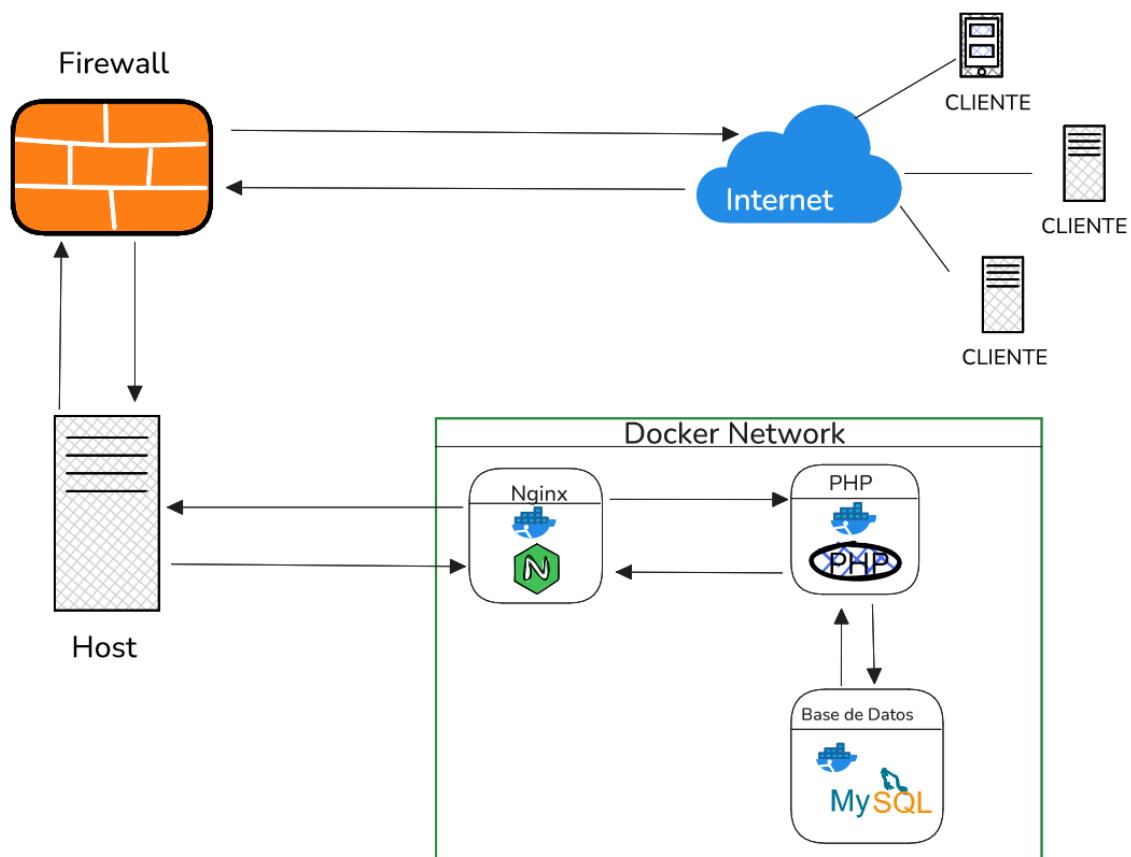


Imagen 3: Vista de despliegue de SIGMA