
ULASALLE

INTEGRICODE

Versión 1.0

Documento de Arquitectura de Software

1. Introducción

1.1 Propósito

Este documento proporciona una visión general arquitectónica del Sistema de INTEGRICODE. Para ello se presentan diversas vistas arquitectónicas para atender diferentes aspectos del sistema. Con éste se pretende capturar y establecer las decisiones arquitectónicas significativas, que han sido definidas para el desarrollo del sistema.

1.2 Alcance

El presente documento tiene como objetivo describir la arquitectura del sistema INTEGRICODE, una aplicación web diseñada para la detección de plagio en los códigos presentados por estudiantes. El documento abarca las interacciones entre los distintos actores del sistema, las funciones disponibles para cada usuario, y la estructura tecnológica que sostiene el sistema. Además, detalla cómo se gestionan los componentes tanto a nivel de frontend como de backend, así como la integración de la API para tareas específicas como el web scraping. Se incluyen diagramas que ilustran las vistas del sistema desde la perspectiva de los casos de uso, el desarrollo, y el despliegue.

1.3 Definiciones, Acrónimos, y Abreviaturas

- API: Application Programming Interface. Interfaz que permite la comunicación entre diferentes componentes del sistema o entre sistemas distintos.
- Web Scraping: Técnica para extraer información de páginas web de forma automatizada.
- Frontend: Parte de la aplicación con la que los usuarios interactúan directamente (interfaz de usuario).
- Backend: Parte de la aplicación que gestiona la lógica de negocio, el acceso a la base de datos y otras operaciones detrás de escena.
- XAMPP: Paquete de software que proporciona un entorno de servidor local para desarrollo web.
- MySQL: Sistema de gestión de bases de datos relacional de código abierto.

1.4 Visión general

Este documento presenta la arquitectura del sistema INTEGRICODE a través de diversas vistas que permiten entender sus funcionalidades y componentes. Las vistas incluidas en el documento son:

- Vista de casos de uso: Descripción de las interacciones entre los usuarios (estudiantes y docentes) con el sistema, mostrando las funcionalidades clave disponibles para cada actor.
- Vista de desarrollo: Explicación de cómo está organizado el código del sistema, sus módulos principales, y cómo se conectan entre ellos.
- Vista de despliegue: Muestra cómo el sistema está implementado en un entorno de servidor local, utilizando tecnologías como XAMPP, PHP, y MySQL, y cómo los diferentes componentes están interrelacionados.

Cada una de estas vistas se describe en detalle en las secciones correspondientes para facilitar la comprensión de la arquitectura global del sistema.

2. Vista de escenarios

2.1 Diagrama



2.2 Descripción de la Vista

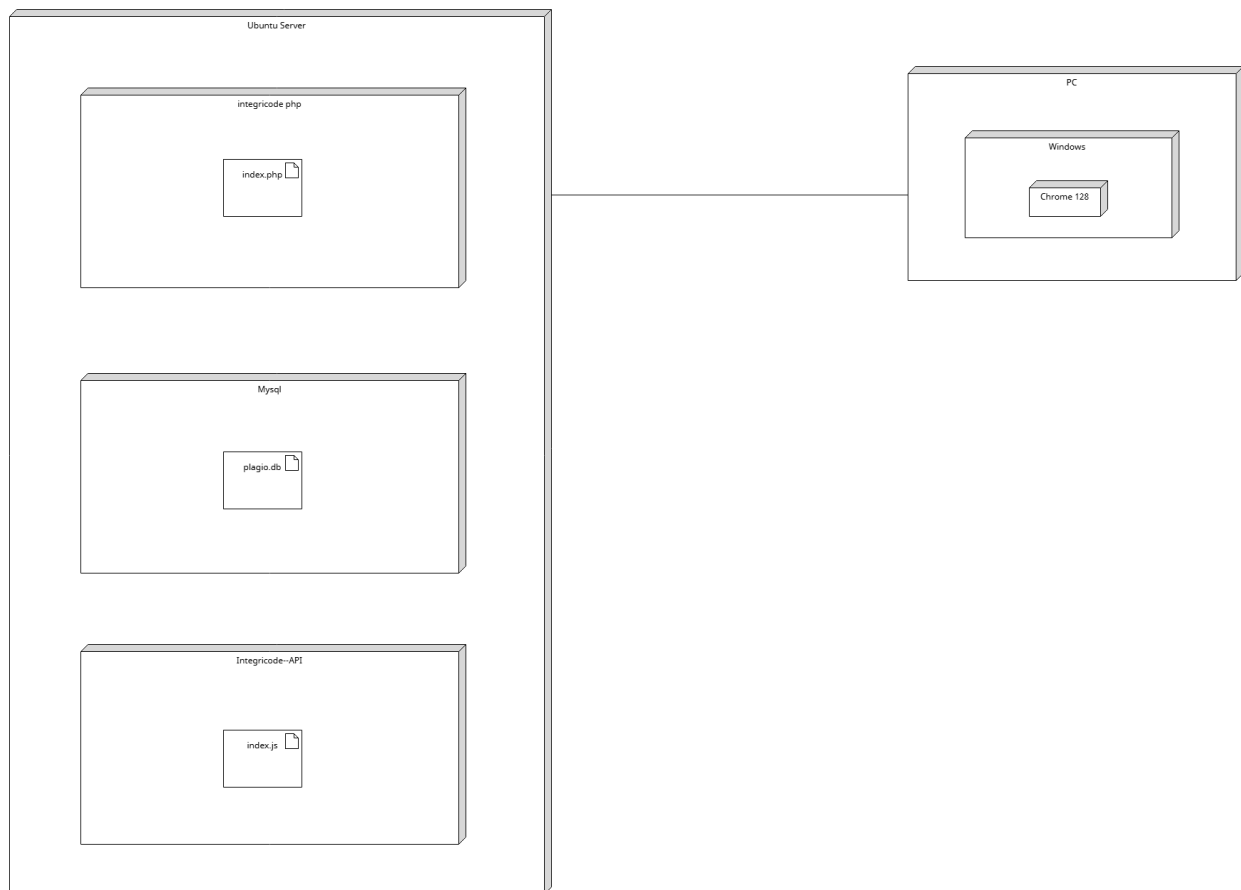
Este diagrama ilustra los casos de uso del sistema, que tiene dos actores principales: el

Estudiante y el Docente. Cada uno interactúa con el sistema en diferentes capacidades. El estudiante puede realizar varias acciones, como Ingresar a curso, que le permite acceder a un curso específico. También puede Enviar Enlace, lo que implica subir el enlace compartido de su código a la plataforma para su revisión. Además, el estudiante tiene la capacidad de Loguear Usuario, que es el proceso de iniciar sesión en la plataforma.

Por otro lado, el Docente tiene un conjunto de funcionalidades más orientadas al análisis y gestión. Puede Analizar códigos de estudiantes, lo que le permite revisar los enlaces de código enviados para comprobar posibles plagios. El docente también puede Gestionar cursos, lo que implica administrar los cursos que tiene asignados, y Gestionar Actividades, donde organiza las tareas y proyectos dentro de cada curso. Por último, tiene la capacidad de Generar Reporte de Similitud, que le permite generar un informe que muestra el nivel de plagio o similitud entre los códigos enviados por los estudiantes.

3. Vista física

3.1 Diagrama



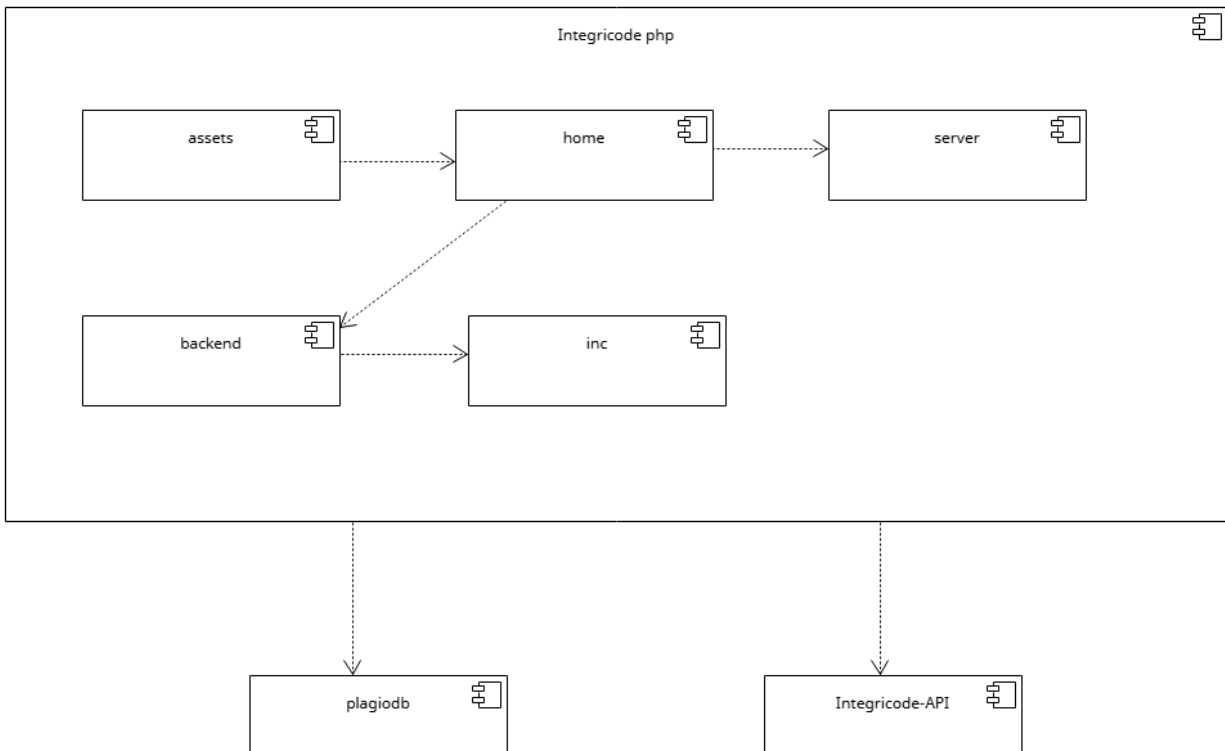
3.2 Descripción de la Vista

El diagrama de despliegue muestra cómo INTEGRICODE opera en un servidor Ubuntu, donde se aloja la infraestructura principal del sistema. El servidor contiene tres elementos fundamentales: el archivo index.php, que actúa como el núcleo del sistema al gestionar las solicitudes iniciales y redirigir a los módulos adecuados; la base de datos plagio.db, donde se almacena toda la información sobre estudiantes, enlaces de código y reportes de similitud; y el

archivo index.js, correspondiente a la API de INTEGRICODE, responsable de realizar tareas especializadas como el web scraping de los enlaces de código enviados por los estudiantes. Los usuarios acceden al sistema desde un equipo con Windows utilizando el navegador Chrome 128, lo que refleja un entorno distribuido, con el servidor Ubuntu centralizando todos los procesos y recursos necesarios para la aplicación.

4. Vista de desarrollo

4.1 Diagrama



4.2 Descripción de la Vista

El diagrama de desarrollo describe la estructura interna de INTEGRICODE, organizada de manera modular para garantizar la escalabilidad y el mantenimiento del sistema. La aplicación se compone de módulos que cumplen funciones específicas y trabajan de manera interconectada. El módulo Assets gestiona los recursos estáticos, como imágenes y hojas de estilo CSS, necesarios para la interfaz de usuario. El módulo Home se encarga de presentar la página inicial del sistema, donde los usuarios interactúan por primera vez. Por su parte, el módulo Server actúa como intermediario, procesando las solicitudes de los usuarios y distribuyéndolas a los módulos correspondientes, mientras que el Backend implementa la lógica del negocio y se conecta con la base de datos plagiodb, que asegura la persistencia de datos críticos. Además, el módulo Inc contiene funcionalidades reutilizables, como la configuración de la conexión a la base de datos. Finalmente, el archivo index.js de la API complementa las operaciones del backend con análisis más avanzados, como la extracción y procesamiento de datos de los enlaces de código enviados por los estudiantes. Todos los módulos trabajan en conjunto para proporcionar un sistema funcional, robusto y eficiente.