



Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas

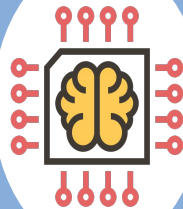
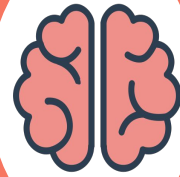
Propuesta de arquitectura de la plataforma para la gestión, análisis y procesamiento de datos BrainSSys.

Autor: Jesús Enrique Fuentes González

Tutor : Dr.C. Arturo Orellana García

Introducción

La **neurociencia** es el campo de la ciencia que estudia el **sistema nervioso** y todos sus aspectos.



La **neuroinformática** es la **rama** de la **neurociencia** que involucra en la investigación a **ingenieros informáticos**.

Introducción

3

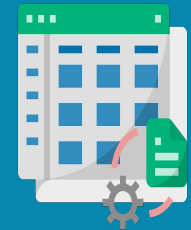
01

Plataformas: LORIS y CBRAIN



02

Gestión parcial de los datos,
incompatibilidad con sistemas heredados



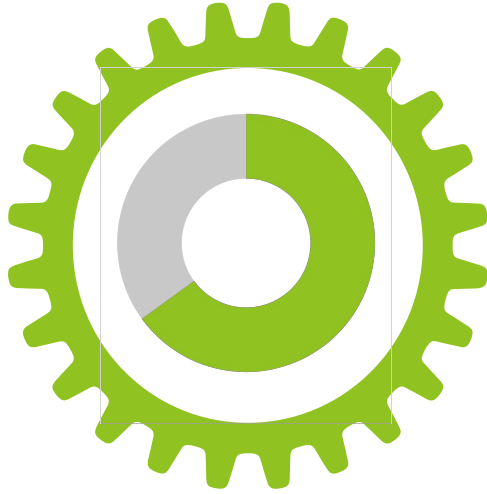
03

Resultados de investigación opacos, de poca confianza



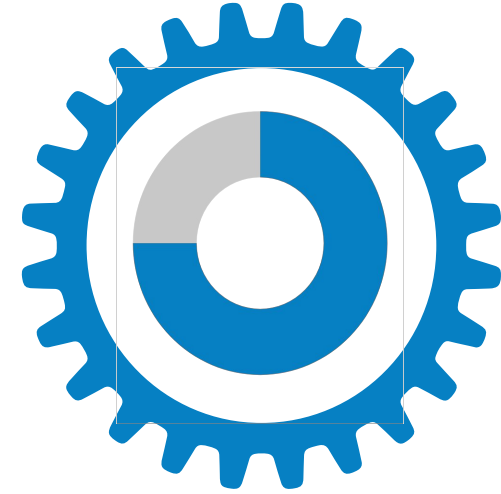
Problema científico

Las **plataformas** de investigación de neurociencias **carecen** de un conjunto de **funcionalidades** y **características** asociadas a su arquitectura de software que son necesarias para el correcto **funcionamiento**, **mantenimiento** e **integración** de las mismas en los centros de investigación.



Objeto de estudio

El proceso de diseño de la arquitectura de software.



Campo de acción

La arquitectura de software en las neurociencias.

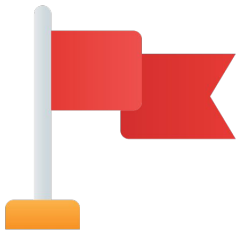
Objetivo general

6

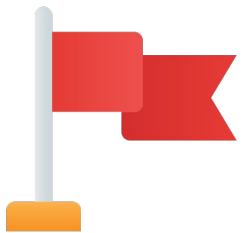
Diseñar la arquitectura de software de un sistema de gestión y procesamiento de datos de neurociencias que propicien su mantenibilidad, escalabilidad e integración.

Tareas Investigativas

7



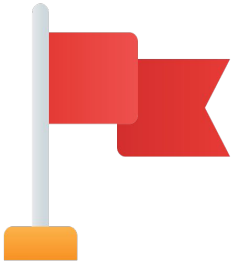
Elaboración de un **marco teórico-metodológico** referente a las arquitecturas de software presentes en plataformas usadas en el ámbito de la neurociencia y los elementos fundamentales del objeto de estudio.



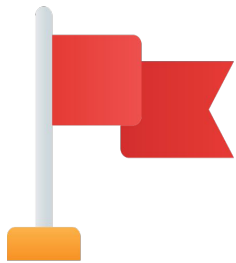
Diagnóstico de las **arquitecturas de software** de productos establecidos en el campo de gestión y el procesamiento de neurodatos para identificar buenas prácticas, errores y elementos a reutilizar.

Tareas Investigativas

8



Diseño de la arquitectura de software para la gestión y procesamiento de neurodatos.



Validación de la propuesta de solución a partir de los métodos definidos en la investigación.

Antecedentes

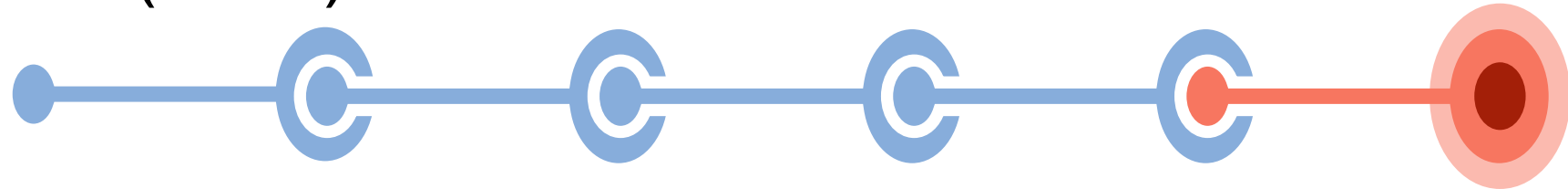
9

1989

Proyecto "Cerebro
Humano" (**HPB**)

2004 - 2020

Marco de Información
de Neurociencia

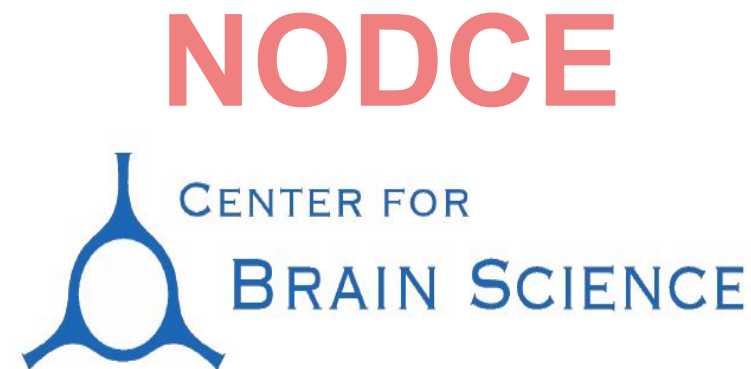


2003

Grupo de Información
Cerebral (**BIG**)

Sistemas homólogos

10



CBRAIN



Sistemas homólogos

11

Atributos/Plataformas	LORIS	CBRAIN	NODCE
Arquitectura	Monolítica basada en capas	Microlito basado en componentes	Microservicios, basado en componentes vía servicios
Almacenamiento de datos	Sql, datos estructurados combinado con sistemas de archivos	Puede tener disímiles fuentes de datos.	Datos estructurados y no estructurados combinando varias bases de datos con sistemas de archivos
Distribución de recursos	Centralizado	Distribuido	Híbrido
Administración de procesos	Delega en CBRAIN si está disponible	Sistema hecho a medida	Gestionan los recursos en tiempo real, dependiendo de la demanda

Sistemas homólogos

12

Atributos/Plataformas	LORIS	CBRAIN	NODCE
Código Abierto	Si	Si	No
Mantenibilidad	Media	Media	Alta
Extensibilidad	Baja	Media	Alta
Escalabilidad	Media	Alta	Alta
Reusabilidad	Baja	Media	Media

Lenguajes

13



Python
v3.10



YAML
v3.8



UML
v2.0



TOML
v1.6

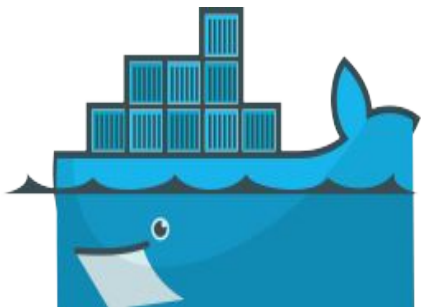
Herramientas

14



Visual Studio

v1.68



docker

v20.10



git

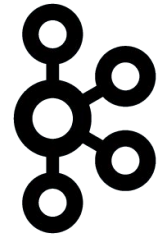
v2.36

Tecnologías

15



Traefik



kafka

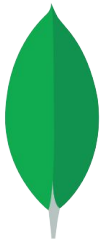


elasticsearch

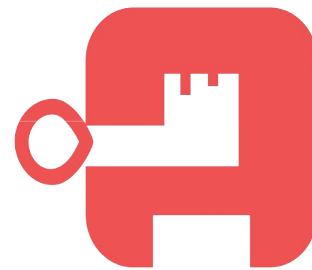


MINIO

NGINX



mongoDB®



authentik

Metodología

16

Adoptar una estrategia de desarrollo incremental



Solapar las diferentes fases del desarrollo

Basar la calidad del resultado en el conocimiento táctico

Historia de usuario

17

Baja

Revisar dato y metadatos del repositorio

Como **ingeniero de datos** quiero **revisar los datos subidos al repositorio** para **asegurar que los metadatos generados asociados al mismo sean consistentes**

- El sistema notifica a los ingenieros cuando un neurodato se insertó y sus metadatos fueron extraídos.
- Si hay varios ingenieros de datos su revisión queda sometida a una condición de carrera.

Dado un neurodato en “espera” **cuando** un ingeniero de datos accede al sistema este debe mostrale la notificación asociada **entonces** será capaz de revisar el dato y sus metadatos asociados para editarlos, aprobarlo o denegarlo.

Modelo de diseño C4

18

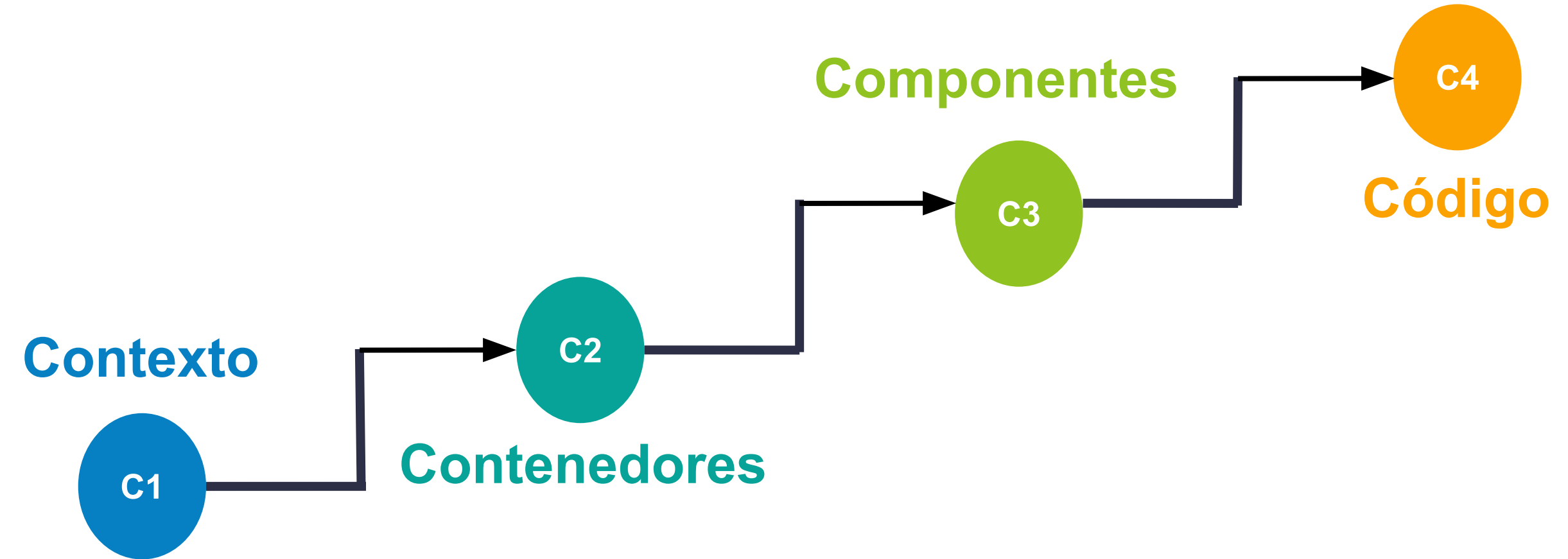


Diagrama de contexto del sistema

19

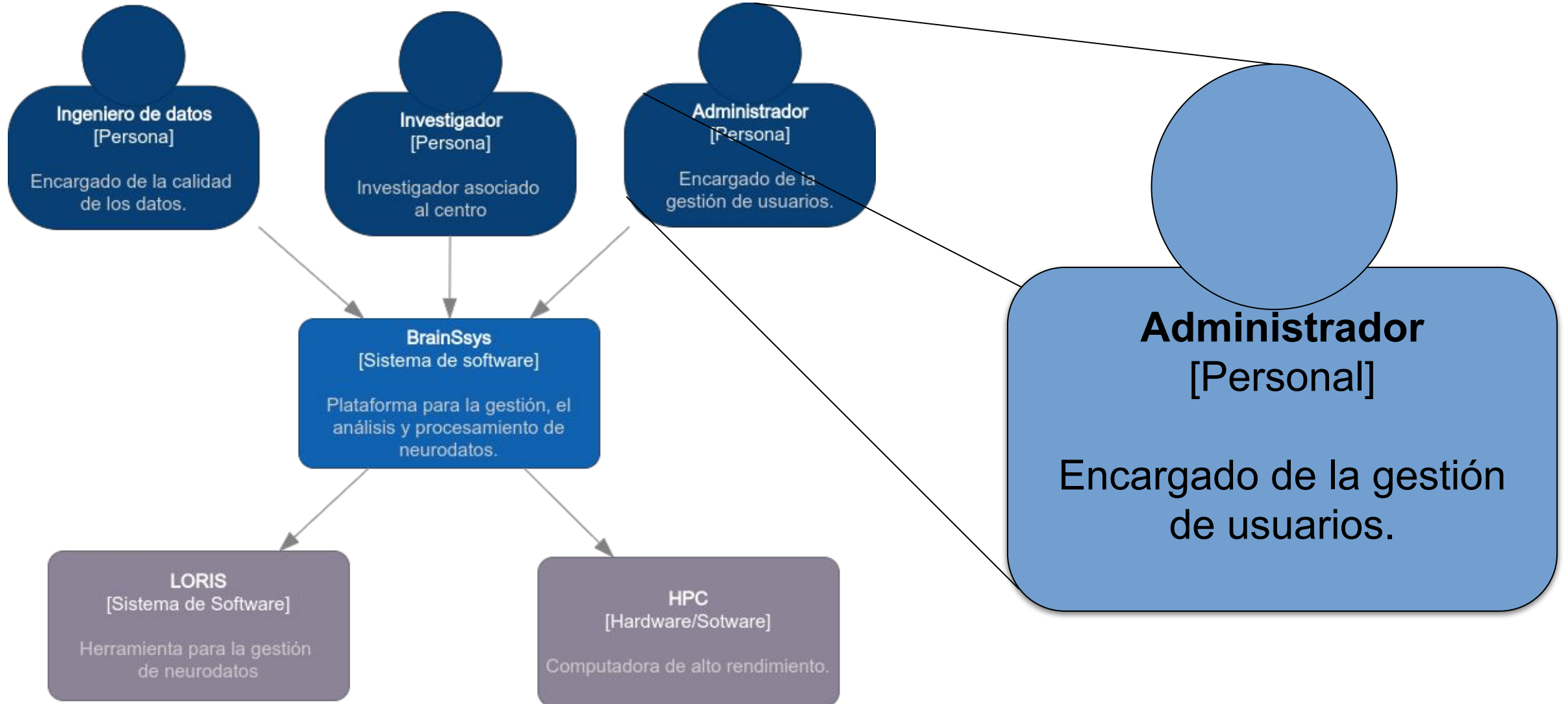


Diagrama de contexto del sistema

20

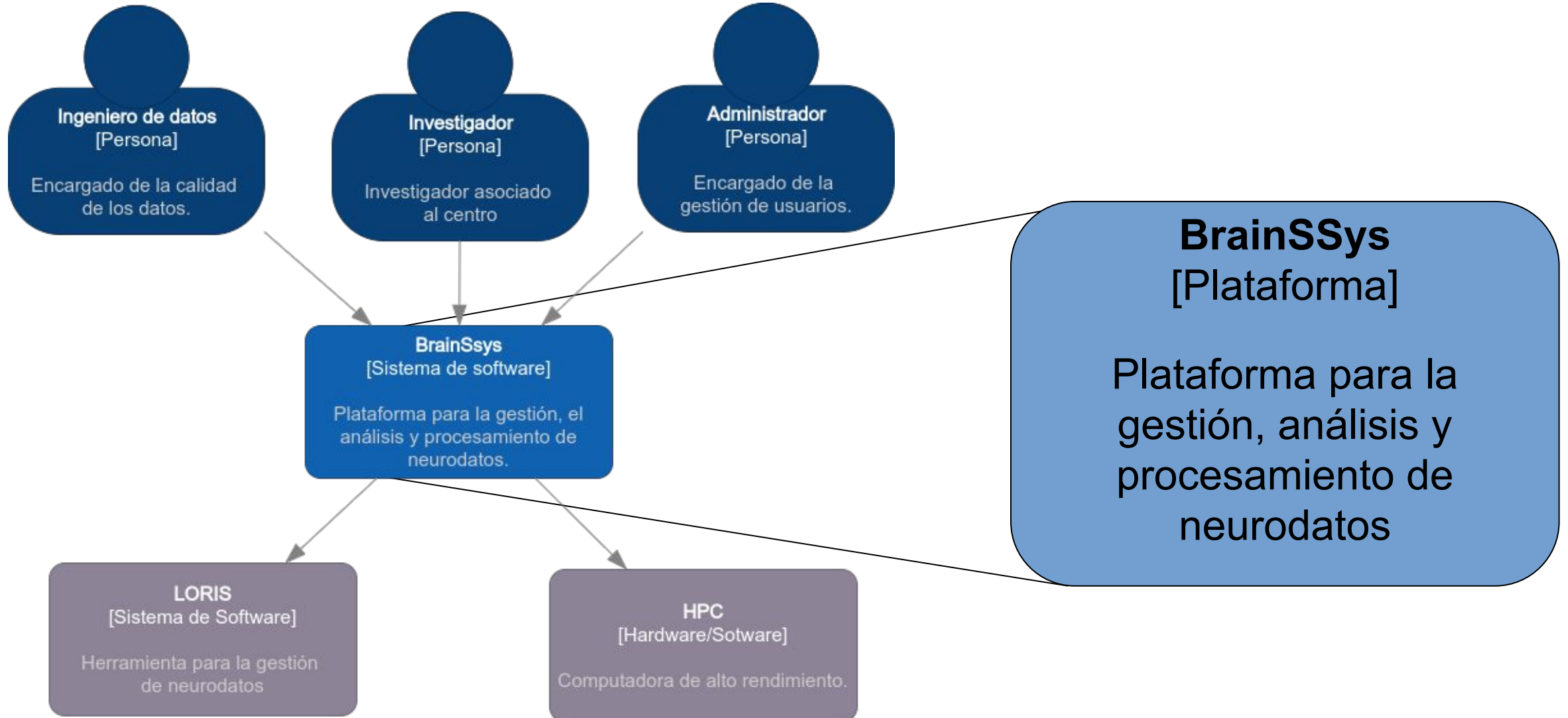


Diagrama de contexto del sistema

21

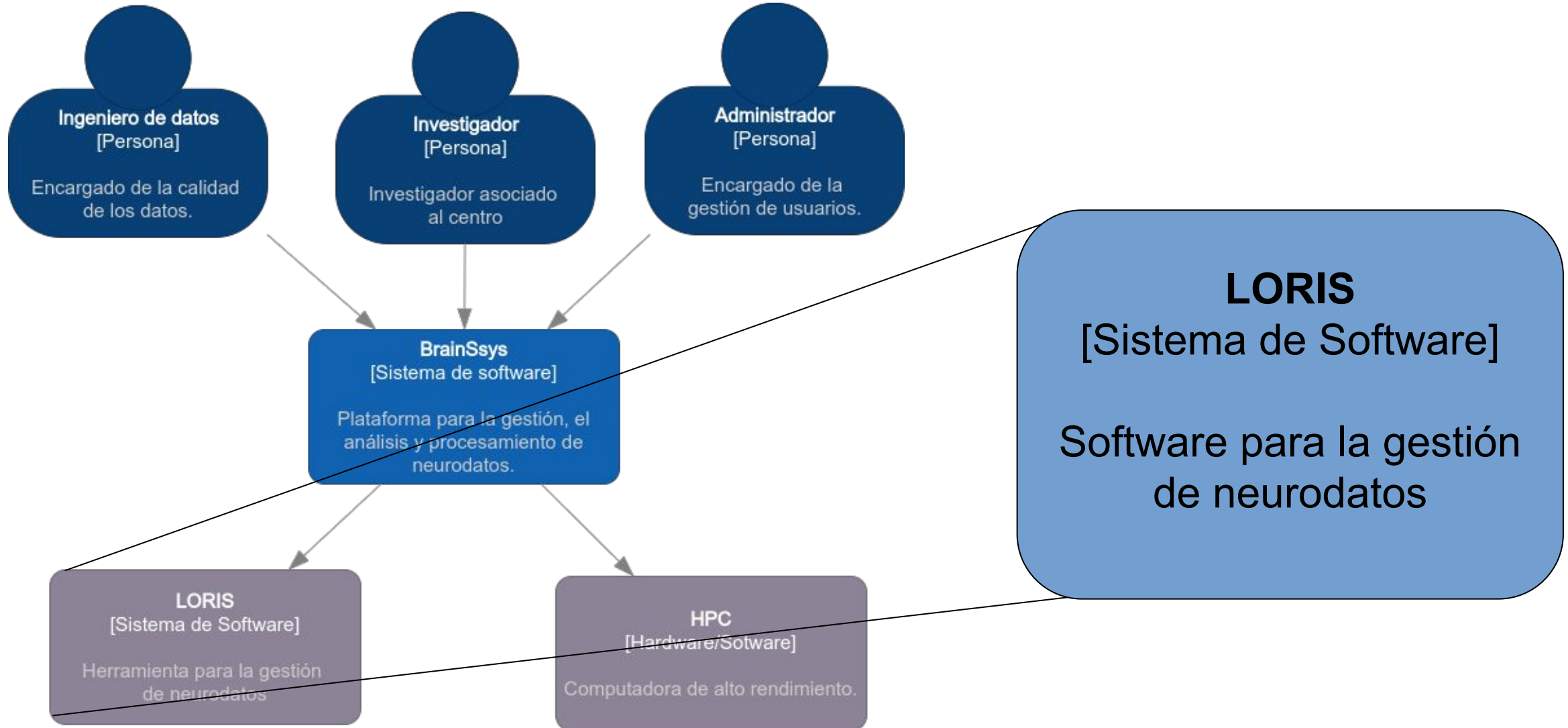


Diagrama de contenedores

22

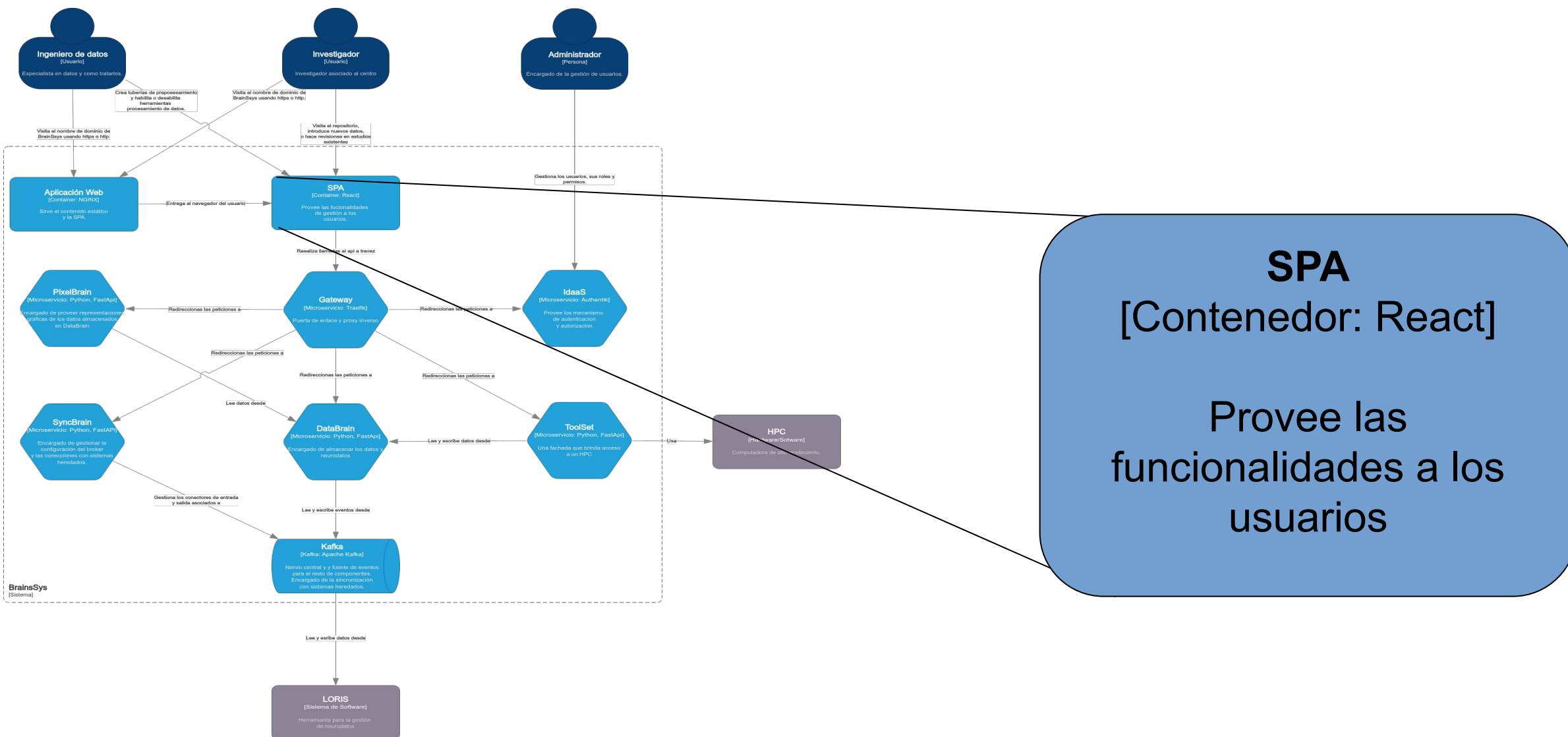


Diagrama de contenedores

23

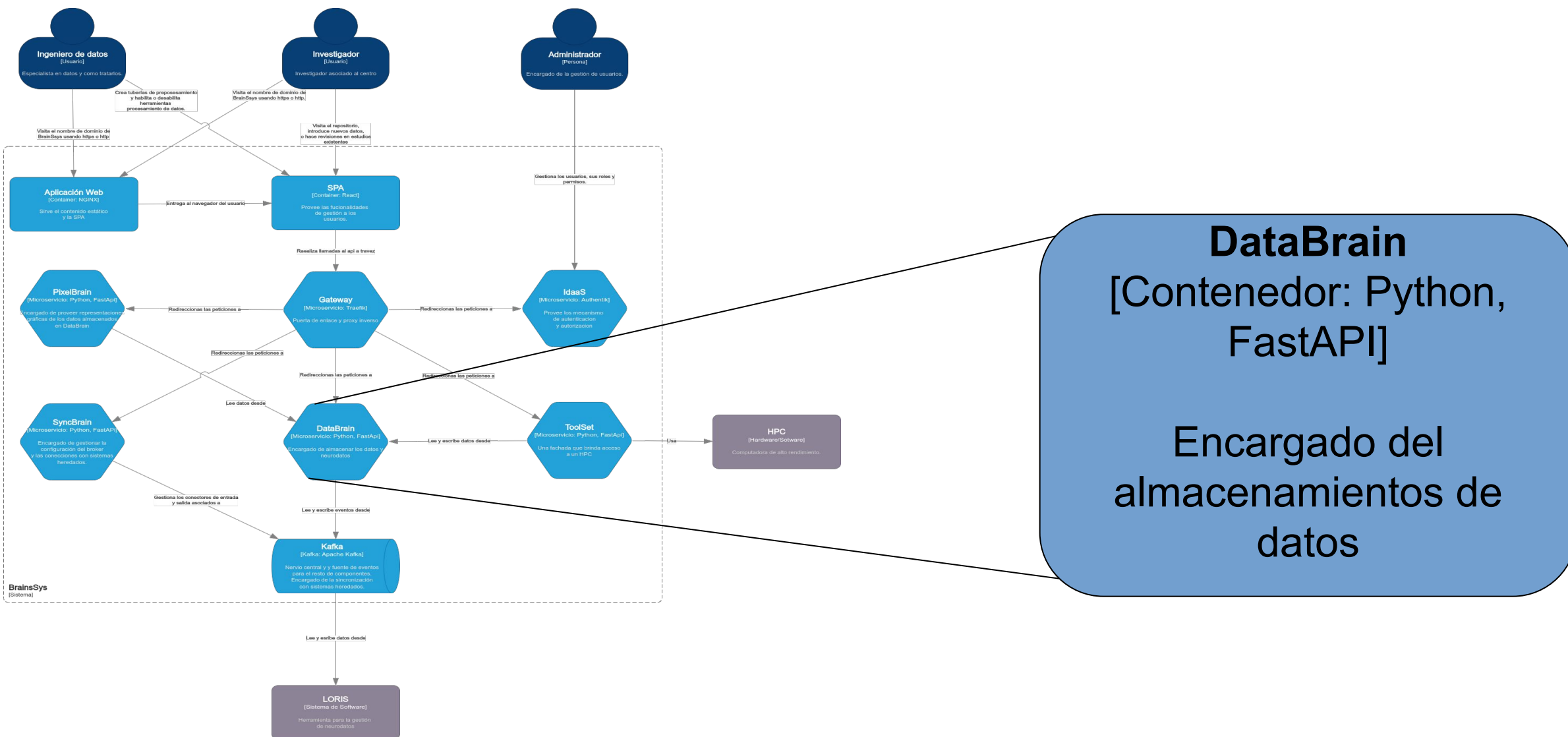
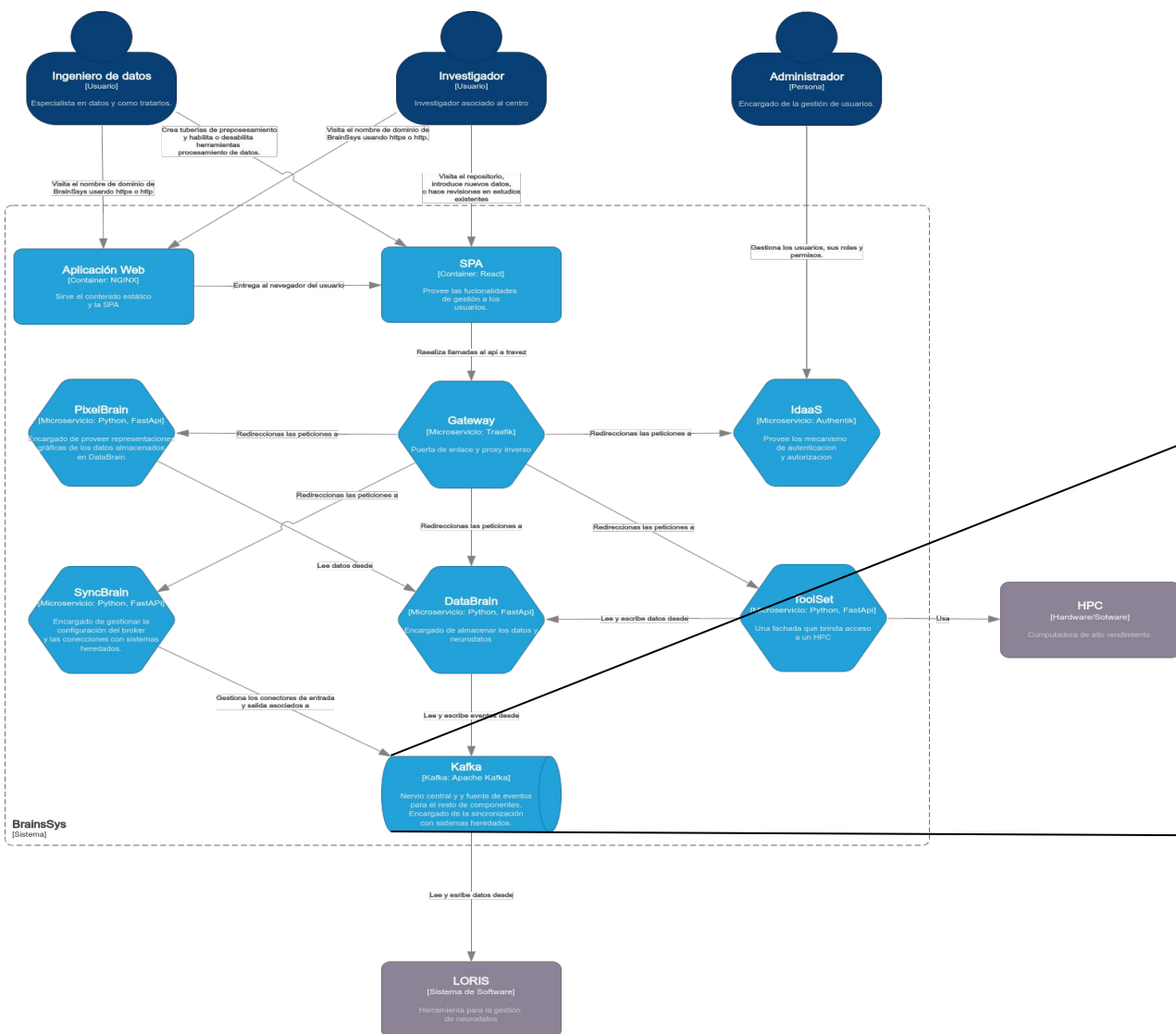


Diagrama de contenedores

24



KAFKA
[Broker: Apache Kafka]

Nervio central, fuente de eventos para el resto de los componentes.

Diagrama de componentes

25

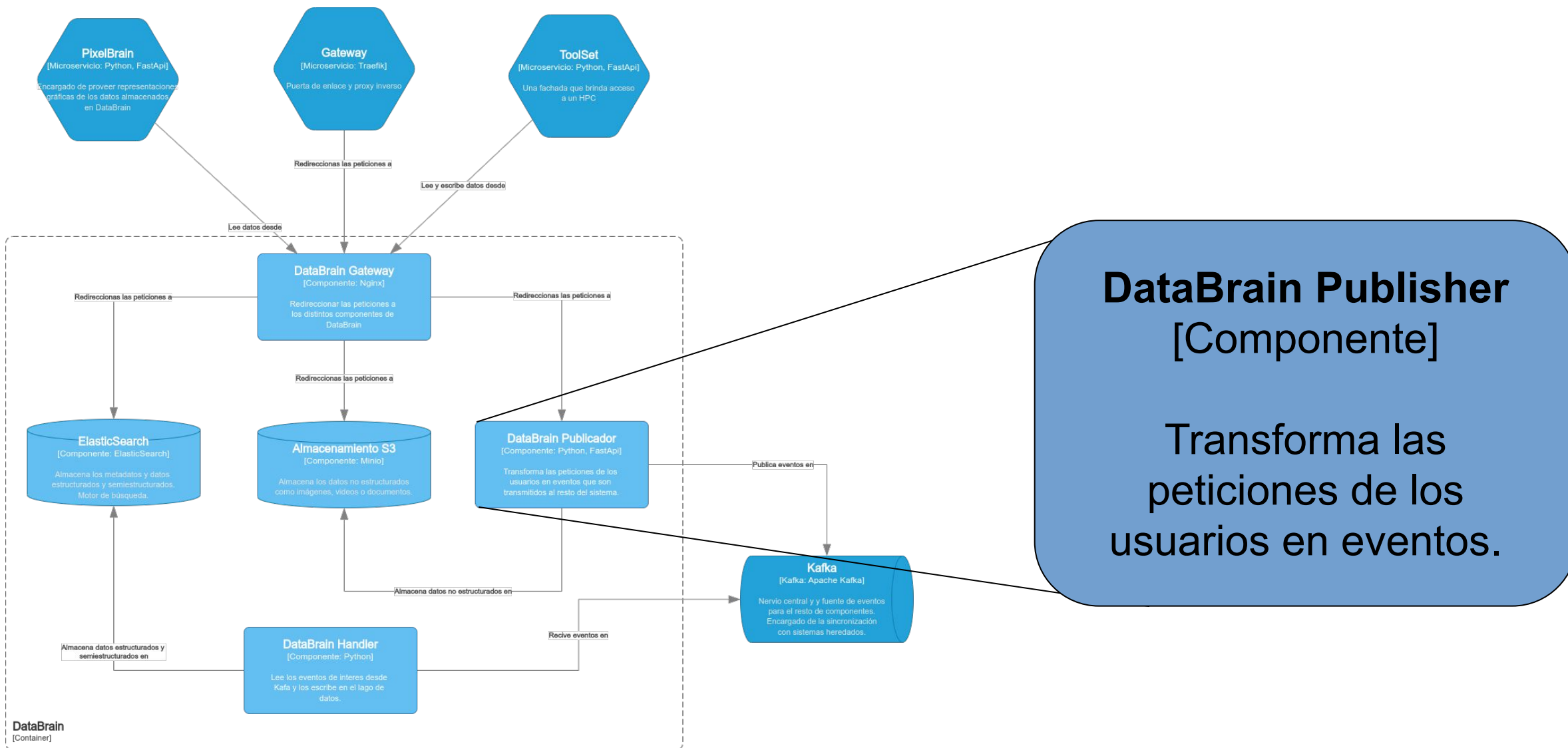


Diagrama de componentes

26

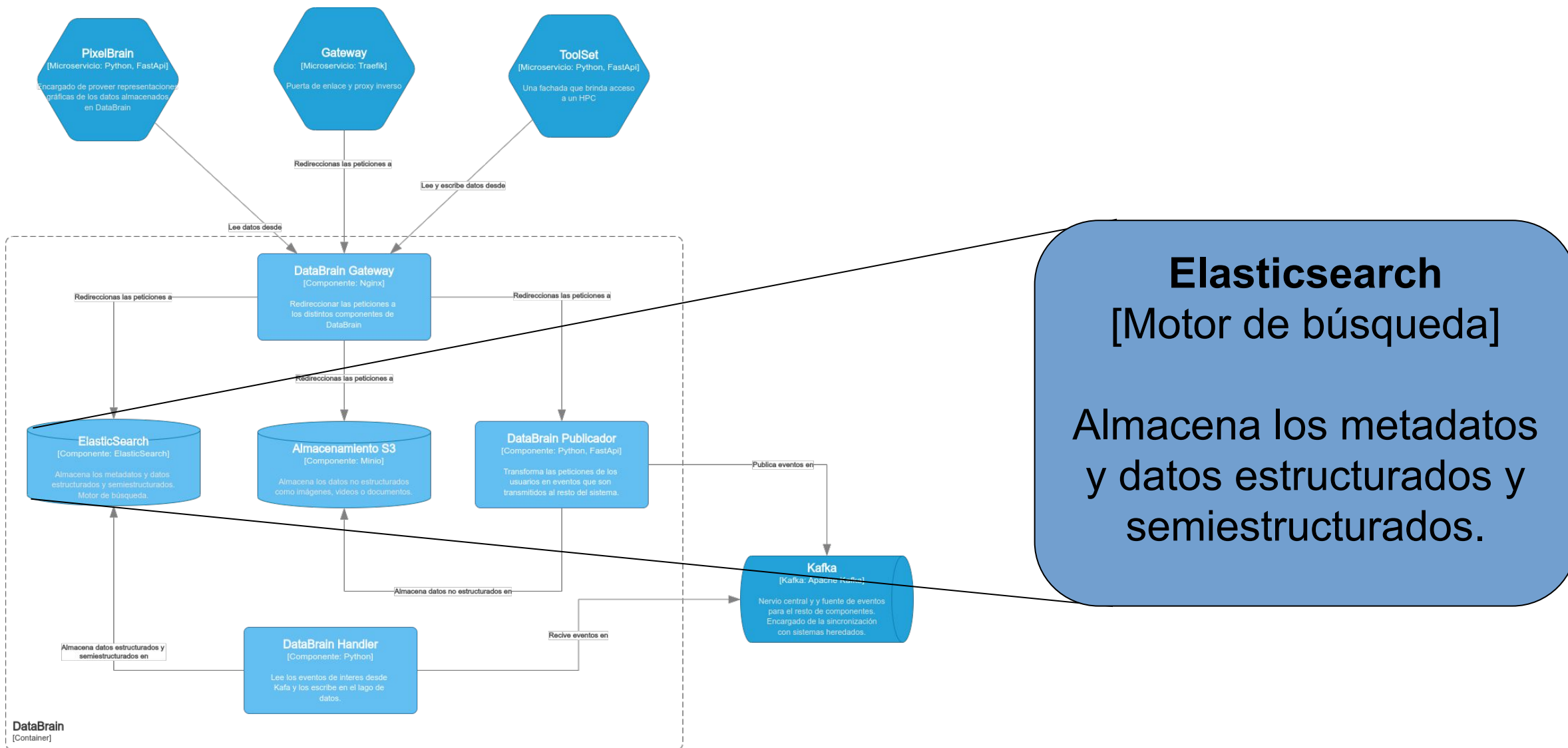
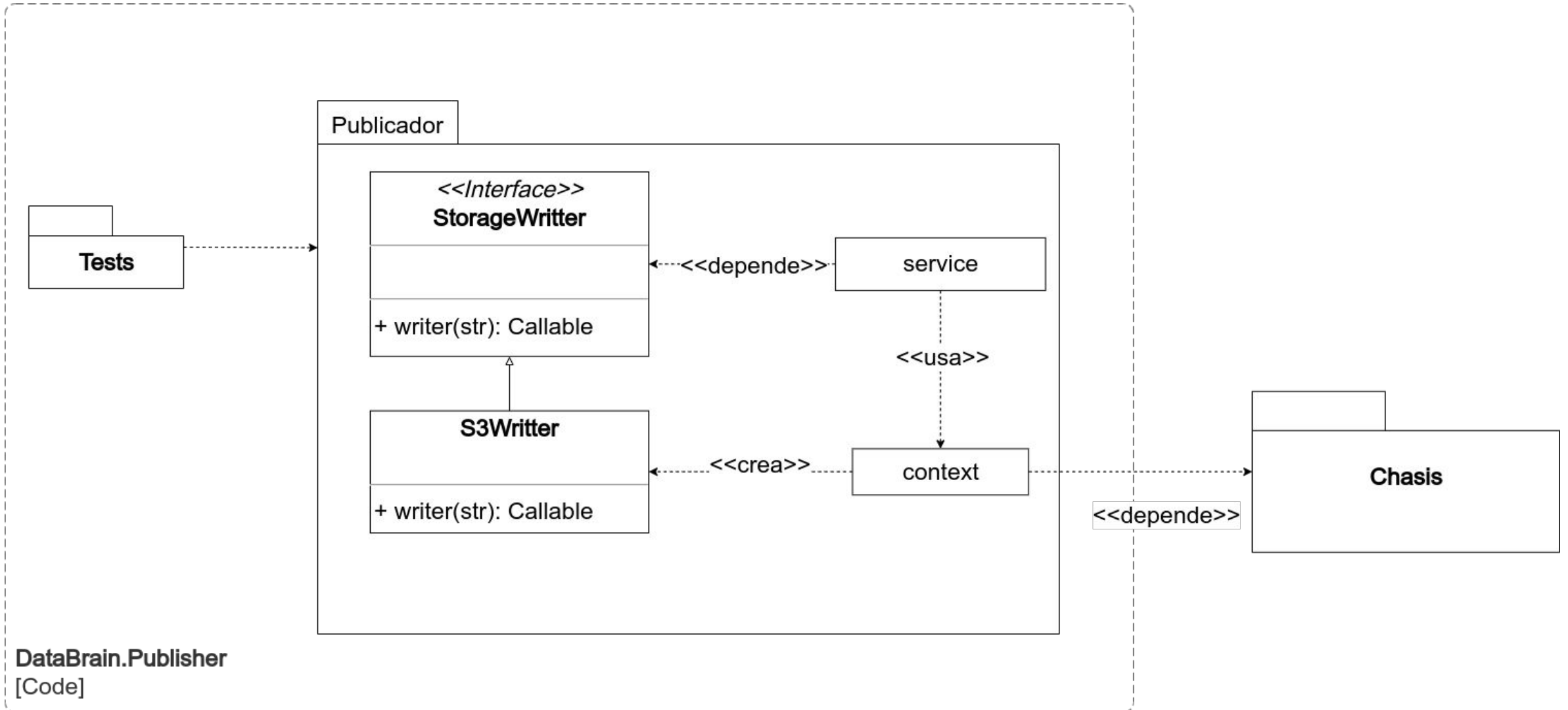


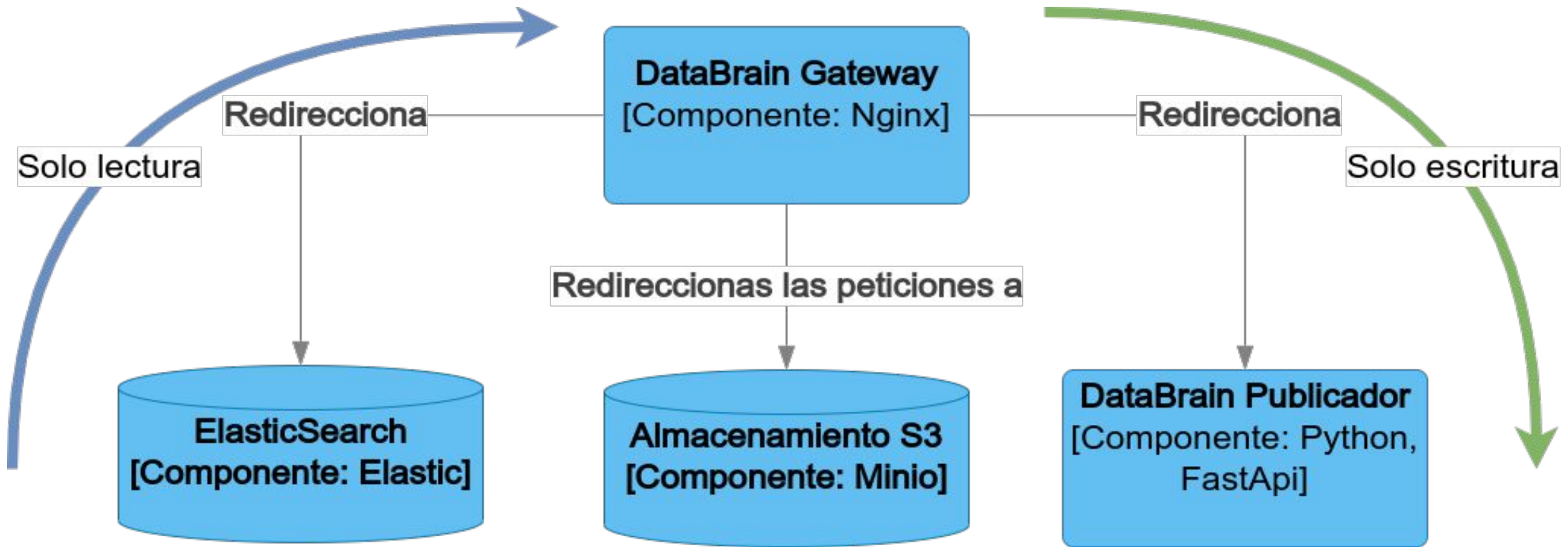
Diagrama de código

27



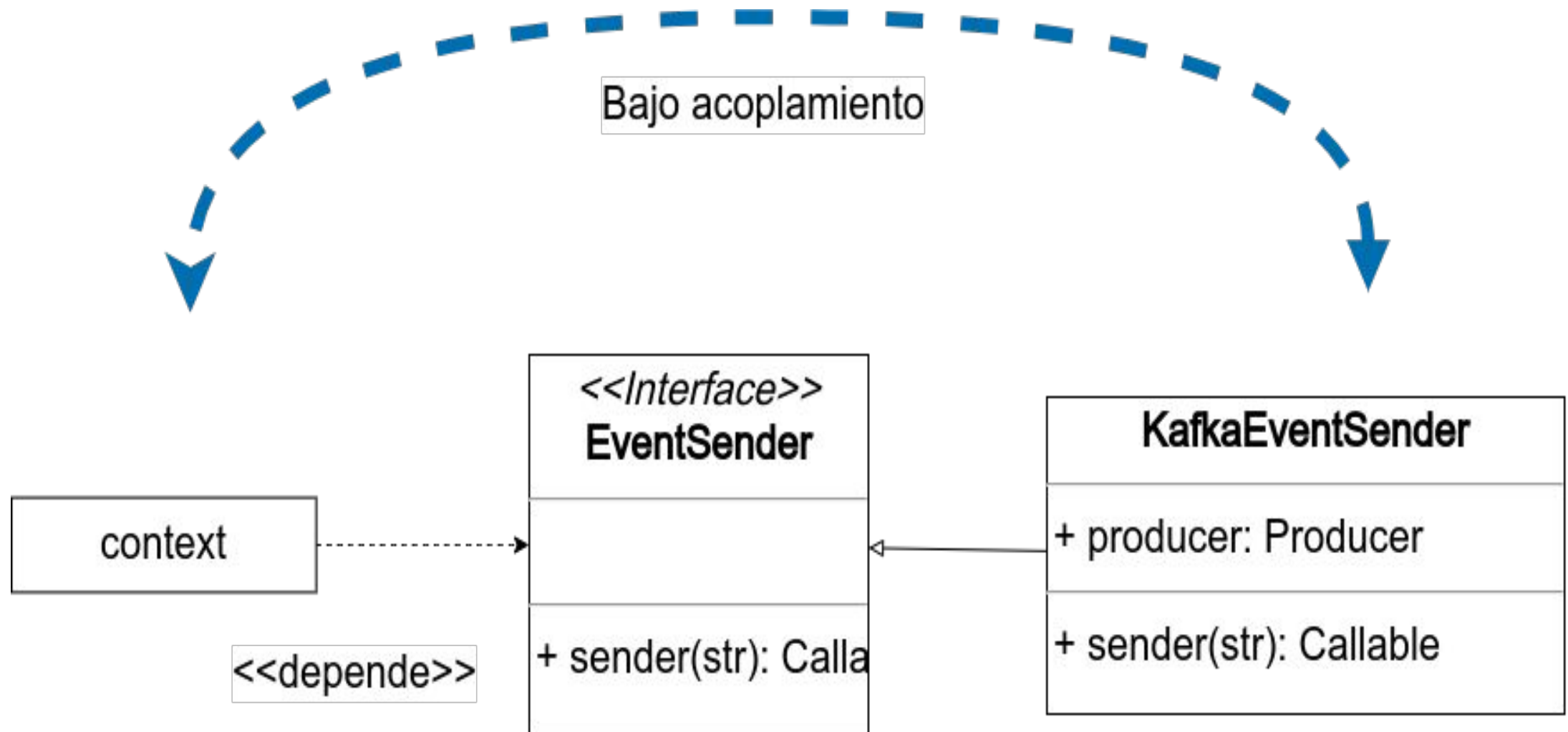
Patrones de diseño: CQRS

28



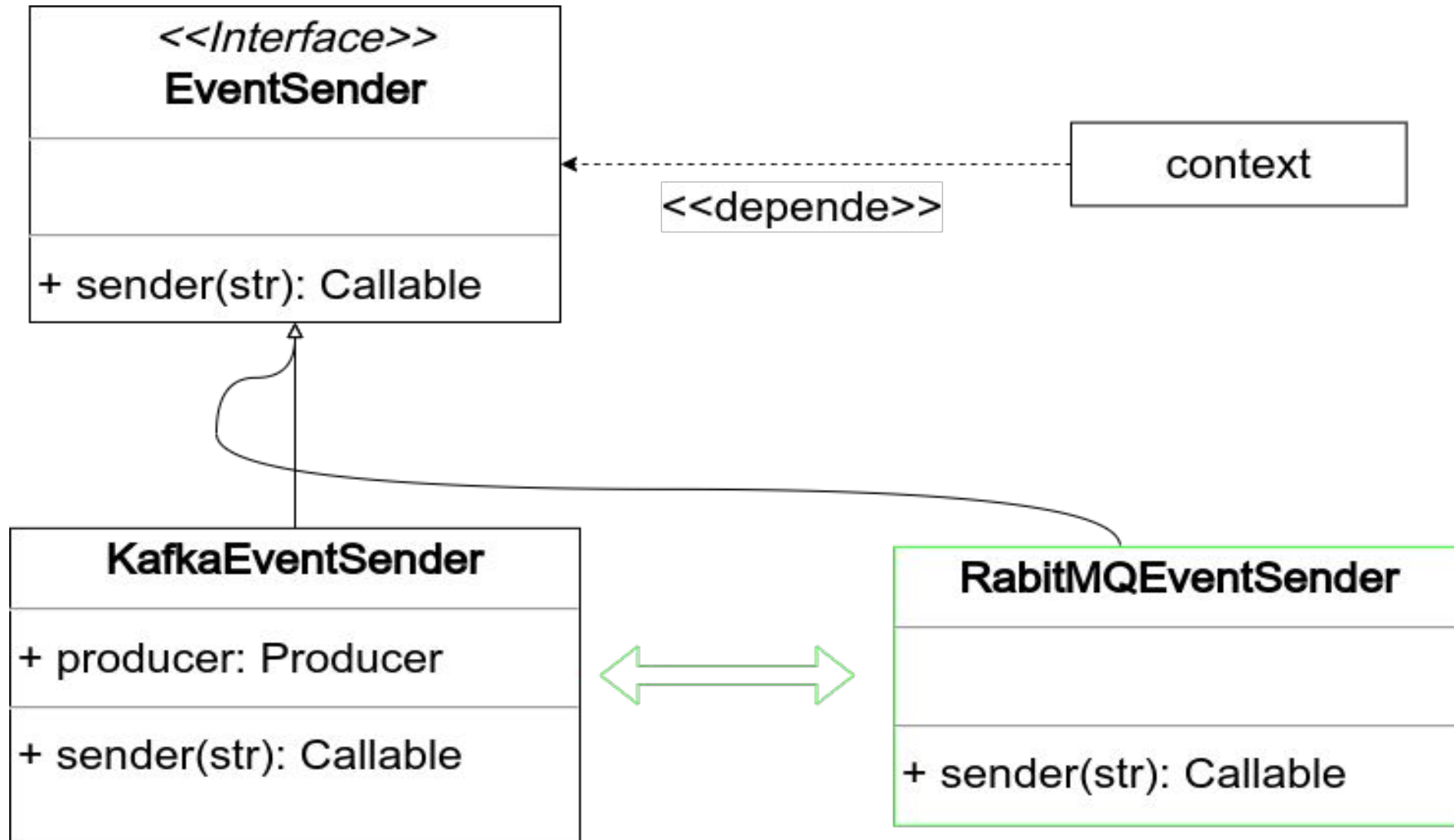
Patrones de diseño: GRASP

29



Patrones de diseño: SOLID

30



Persistencia

31

Persistencia políglota: Es un término de almacenamiento empresarial que se utiliza para describir la elección de **diferentes almacenes de datos** que admiten **diferentes tipos de datos** según sean necesarios.



Seguridad

32

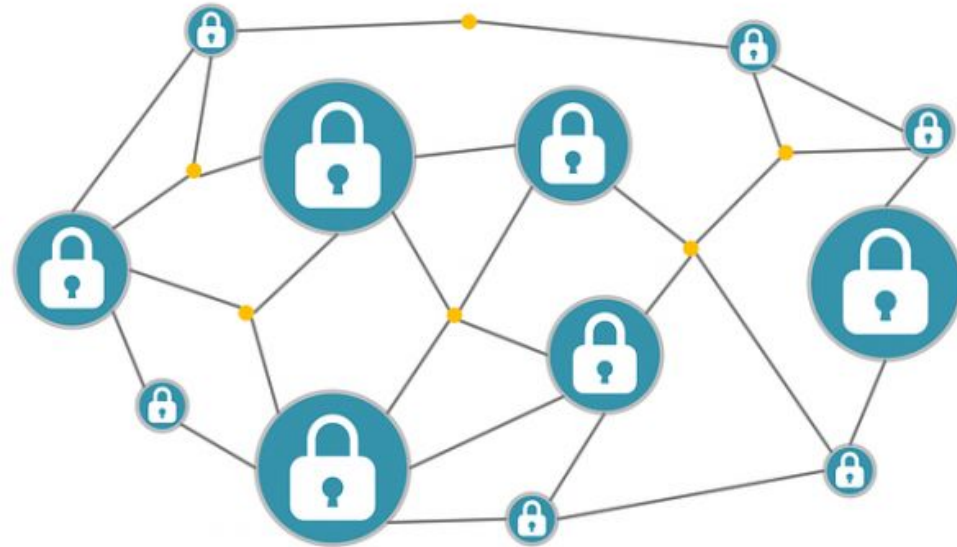


SSL

IDaaS



HTTPS

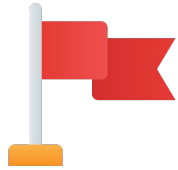


TLS



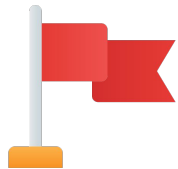
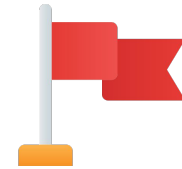
Pruebas unitarias

33



Las **pruebas unitarias** son realizadas a pequeñas **porciones de código**, por separado, para **verificar** su correcta **funcionalidad**.

Se pueden ir **efectuando** desde el **comienzo** de la **implementación**.



Se debe **determinar** qué **porcentaje** del **código** está **cubierto** por las **pruebas** unitarias y cuál no, esto permite determinar que **código** es **robusto** y está **probado**.

Pruebas unitarias

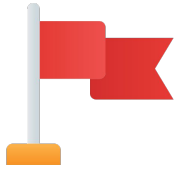
34

Reporte de cobertura

Módulo	Declaraciones	Perdido	Excluido	Cubierto
brainssys/chasis/__init__.py	2	0	0	100%
brainssys/chasis/extras/web/server.py	64	64	0	0%
...
brainssys/chasis/extras/web/web.py	72	72	0	0%
brainssys/chasis/logging.py	16	11	0	31%
Total	342	190	26	44%

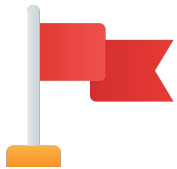
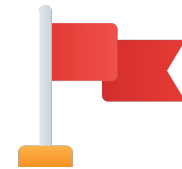
Pruebas de aceptación

35



Las **pruebas de aceptación** son creadas como los **criterios** de aceptación a las **Historias de Usuarios**, en cada ciclo de la iteración del desarrollo.

Redactar los criterios de aceptación con la **sintaxis de Gherkin** nos permite **diseñar pruebas** que modelen el comportamiento esperado.



Combinar las técnicas generales y los principios del **desarrollo guiado por pruebas**, junto con ideas del **diseño guiado por el dominio**.

Conclusiones



1

La elaboración de un marco teórico-metodológico permitió identificar los diseños arquitectónicos y como contribuye a las métricas de calidad de los software presentes en plataformas usadas en el ámbito de la neurociencia y los elementos fundamentales del objeto de estudio

2

Con el diagnóstico de las arquitecturas de software de LORIS, CBRAIN y NODCE se obtuvieron las características a tenerse en cuenta en el desarrollo de la propuesta de solución y se identificó la arquitectura de microservicios como el diseño de la misma.

Conclusiones

3

Del diseño de la arquitectura de software se obtuvo un conjunto de diagramas que modelan diferentes capas de abstracción de la propuesta de solución así como los principales patrones de diseño usados.

4

Se validó la arquitectura usando un prototipo funcional sometido a pruebas unitarias y de aceptación usando técnicas de desarrollo dirigido por pruebas y comportamiento y las herramientas brindadas por el entorno de desarrollo.

Recomendaciones

- Incluir **funcionalidades** que permitan mayor **observabilidad y trazabilidad** de los **microservicios**.
- Incluir **funcionalidades** que permitan hacer **capturas del estado** de los servicios para **disminuir el tiempo de recuperación** de errores en el sistema.



Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas

Propuesta de arquitectura de la plataforma para la gestión, análisis y procesamiento de datos BrainsSys.

Autor: Jesús Enrique Fuentes González

Tutor : Dr.C. Arturo Orellana García