

Tecnología colaborativa para investigadores: implementación y optimización de un repositorio GitLab para gestión de la configuración y versionado

Damián Ferrara¹, Emanuel Rodriguez², Romeo Lorenzo Monfroglio³, Manuel Robles⁴,
María Ayelén Díaz Lapérgola⁵, Leopoldo Nahuel⁶

GIDAS - Grupo de I&D Aplicado a Sistemas informáticos y computacionales
Universidad Tecnológica Nacional (UTN) - Facultad Regional de La Plata (FRLP)
Av. 60 esq. 124 s / n° CP 1900, La Plata, Buenos Aires, Argentina

[1dferrara@frlp.utn.edu.ar](mailto:dferrara@frlp.utn.edu.ar), [2erodriguez@frlp.utn.edu.ar](mailto:erodriguez@frlp.utn.edu.ar),
[3rmonfrolio@alu.frlp.utn.edu.ar](mailto:rmonfrolio@alu.frlp.utn.edu.ar),
[4mroblesrobles@alu.frlp.utn.edu.ar](mailto:mroblesrobles@alu.frlp.utn.edu.ar), [5madlapergola@frlp.utn.edu.ar](mailto:madlapergola@frlp.utn.edu.ar),
[6lnahuel@frlp.utn.edu.ar](mailto:lnahuel@frlp.utn.edu.ar)

Resumen

Partiendo de la instalación del servidor Lenovo ThinkServer SR530 en el laboratorio, Grupo de Investigación y Desarrollo Aplicado a Sistemas Informáticos y Computacionales (GIDAS) a cargo del grupo “Transformación digital en organizaciones aplicando tecnologías de infraestructura informática basada en computación en la nube” (TDO), se presentaron nuevos desafíos relacionados con la descentralización y versionado de información y código.

Para dar solución a estos, el grupo anteriormente mencionado ha investigado sobre herramientas de gestión de la configuración y decidió implementar GitLab por su licencia de software de código abierto, amplia popularidad, uso y documentación existente acerca de ella.

La implementación de GitLab en nuestra infraestructura logrará una significativa mejora en la gestión y control de la información, al centralizar los repositorios en nuestro servidor; aseguramos un acceso más eficiente a los proyectos y documentos importantes, evitando la dispersión y pérdida de información que podría ocurrir. Además, podremos facilitar la colaboración entre los equipos y garantizar la seguridad y confidencialidad.

La adopción de esta herramienta fortalecerá el control, coordinación y eficiencia del grupo. A pesar de ciertas dudas iniciales, este último se adaptó y niveló conocimientos para aprovechar los beneficios que puede proporcionar GitLab en el servidor para todo el laboratorio.

Palabras Claves: gestión de la configuración, GitLab, repositorio, versionado.

1 Introducción

En búsqueda de eficiencia, agilidad, control y gestión en el ámbito de investigaciones científica del Grupo de Investigación y Desarrollo Aplicado a Sistemas Informáticos y Computacionales (GIDAS) ha llevado al laboratorio a adquirir el servidor Lenovo ThinkServer SR530 [1], un hito significativo en el crecimiento del mismo, que ha permitido al grupo disponer de una infraestructura propia. El equipo “Transformación digital en organizaciones aplicando tecnologías de infraestructura informática basada en computación en la nube” (TDO), perteneciente a este mismo laboratorio, comenzará a prestar servicios entorno a infraestructura IT.

En el artículo “Una experiencia de implementación de infraestructura informática: recorriendo el camino desde lo académico hasta la instalación y puesta en funcionamiento” [2], publicado en el año 2022, se describe el fin con el que fue instalado este equipo.

Sin embargo, tras esta implementación, el grupo se enfrenta a desafíos importantes relacionados con la descentralización y el versionado, tanto en la gestión de información como en el desarrollo de código fuente. Para dar solución a esta problemática realizamos una investigación de herramientas que nos permitan solucionarlo de manera sencilla y eficaz.

Por otro lado, también buscamos una herramienta que nos brinde el servicio de “Tickets”, lo que nos facilitará la administración y seguimiento de problemas que puedan surgir. Además, la capacidad de versionado de archivos que nos posibilite la modificación de estos, permitiendo tener un historial completo de cambios y la posibilidad de retroceder a versiones anteriores si es necesario. Esta característica es invaluable para mantener un registro preciso de los progresos, solucionar problemas y evitar confusiones en el trabajo colaborativo.

Adicionalmente, poder otorgar diferentes niveles de acceso y permisos, para asegurarnos de que cada equipo acceda únicamente a la información relevante para sus tareas, garantizando la seguridad y confidencialidad de los datos sensibles.

Asimismo, el equipo TDO podrá prestar estos servicios personalizados de configuración para el laboratorio. Lo que implica que los diferentes equipos del GIDAS podrán tener entornos de trabajo adaptados a sus necesidades específicas, lo que mejorará la eficiencia y la productividad.

2 Aportes realizados

Contar con una herramienta de gestión para la configuración proporciona varios beneficios a nuestro laboratorio de investigación.

La problemática principal es la descentralización de la configuración, ya que cada equipo de investigación utiliza su propio repositorio en distintos proveedores de la nube, como GitHub y GitLab. Estos son creados por becarios o el coordinador del proyecto, lo que puede ocasionar conflictos de acceso o dificultades para la gestión de los usuarios cuando el administrador original deja su función.

Al tener instalada una herramienta de gestión de la configuración en nuestro servidor local, obtendremos un mayor control y coordinación con los distintos equipos de investigación. Podremos identificar los elementos de configuración que cada equipo necesita y ofrecer un servicio más personalizado. Además, brindaremos capacitación, recomendaciones y mejores prácticas para el uso de la gestión de la configuración.

Esta herramienta nos permitirá tener un control exhaustivo de las versiones y cambios de la configuración, con soporte dedicado para este servicio. Además, el equipo de investigación TDO, lleva más de cuatro años capacitándose en infraestructura, lo que nos permitirá ofrecer

un servicio y soporte profesional, algo que no siempre se encuentra en los proveedores gratuitos que se están utilizando.

Asimismo, podremos tener un control más sólido en la gestión de las baselines, siguiendo las mejores prácticas y estándares como normas ISO e ITIL, entre otros.

Con esta implementación también podremos comenzar a automatizar gradualmente nuestras tareas de gestión de la configuración. Esto implica realizar diversas acciones de forma automática mediante el uso de herramientas específicas, como sistemas de control de versiones, sistemas de seguimiento de problemas y plataformas de integración continua con GitLab CI.

3 Implementar

Para dar solución a los desafíos presentados en el grupo GIDAS se ha investigado dentro del equipo TDO sobre herramientas de gestión de la configuración; se decidió implementar GitLab [3] por su amplia popularidad, uso y documentación existente acerca de ella. GitLab es un manejador de repositorios Git basado en web con manejo de wiki y seguimiento de tickets.

Con el fin de mantener un historial de versiones, en todo momento se implementó esta herramienta para versionar el código y la documentación necesaria en los distintos equipos de investigación.

Los repositorios en ella son espacios donde se almacena el código fuente de un proyecto. Estos funcionan utilizando el sistema de control de versiones Git, que registra y administra los cambios en el código a lo largo del tiempo. Los desarrolladores pueden clonarlo en sus máquinas locales, realizar cambios en el código y luego enviar esos cambios de vuelta al repositorio.

Estos últimos permiten organizarse en ramas, conocidos también como “branch”, que son copias independientes del código. De este modo los desarrolladores pueden trabajar en paralelo en diferentes características o soluciones sin afectar la rama principal.

Al poseer un control de versiones contamos con el beneficio de un análisis y seguimiento de los cambios realizados, podemos comparar el funcionamiento con versiones antiguas y tenemos la capacidad de identificar los inconvenientes para su futura solución.

Si hablamos de colaboración este se logra mediante comentarios, revisiones de código y aprobaciones. Varios usuarios pueden trabajar en el mismo repositorio, realizar cambios en diferentes branch y fusionar sus cambios cuando sea necesario. GitLab también proporciona opciones de gestión de permisos para controlar el acceso y las acciones permitidas en los repositorios.

Además, incluye funciones de integración continua (CI) que permiten automatizar tareas como compilar, probar y desplegar el código en diferentes entornos. Los repositorios se pueden configurar para ejecutar pipelines de CI/CD [4] automáticamente cuando se realizan cambios en el código.

En GitLab tenemos el sistema de issues que son un punto donde se inicia una conversación sobre el código, permiten discutir la implementación de una nueva idea, sugerir propuestas de features, reportar bugs y fallos, obtener reportes y planear implementaciones de código [5].

Es una herramienta que nos deja gestionar y dar seguimiento a problemas, tareas y solicitudes relacionadas con proyectos de desarrollo de software. Los usuarios pueden crear tickets para informar sobre problemas o solicitar new features, asignar responsables, establecer prioridades y agregar comentarios. Estos se organizan en un tablero de proyecto,

donde se pueden visualizar y gestionar de forma eficiente. Los miembros del equipo pueden colaborar en estos mediante comentarios, debates y adjuntando archivos. A medida que los tickets avanzan, se actualizan para reflejar el progreso, como cambiar el estado, registrar tiempo dedicado y vincular con “commits” o solicitudes de extracción. GitLab también se integra con otras herramientas y permite automatizar acciones y notificaciones.

3 Proyecto, relevamiento y desarrollo de experiencia

Cuando emprendimos esta experiencia y decidimos incursionar en GitLab para el servidor del GIDAS se nos presentaron diversas dudas a priori: ¿Qué tipo de información necesitan los equipos que sea gestionada? ¿Cómo los demás grupos del GIDAS almacenan su código actualmente? ¿Cómo se usaría desde el lado administrador GitLab? ¿De qué modo se debería gestionar los usuarios y los roles? ¿Cómo administrar los tickets?

Poco a poco fuimos logrando una intensa tarea de nivelación de conocimientos y adaptación a la nueva herramienta. Al ser un grupo de personas con diferentes experiencias en IT, debimos recaudar la información necesaria para poder entender, trabajar y, posteriormente, aplicar lo aprendido a el servidor en cuestión.

La información de los integrantes actuales de nuestro laboratorio puede ser pedida mediante mail y encuestas con preguntas abiertas, de la manera: ¿Dónde guardan actualmente el versionado de su código? ¿Un administrador o varios? ¿Qué pasos siguen en caso de querer compartir entre sus miembros el código en cuestión? ¿Cuál es su plataforma de preferencia? ¿Qué opinan de un lugar centralizado de almacenamiento común?

Con esta información podemos brindar un mejor servicio para todos los equipos; una mejor organización y un flujo continuo de código firmemente almacenado en infraestructura local, donde pueda ser accesible desde cualquier lugar del que tenga acceso un grupo de investigadores.

4 Trabajo futuro

Al utilizar GitLab como herramienta en nuestro servidor podemos dar servicios de CI/CD [4]. Tendremos a disposición CI porque validamos si el código funciona correctamente antes de enviarlo a producción y así notificar a los desarrolladores de lo ocurrido. Por otro lado, aplicamos CD que consiste en que el código válido, pase a desplegarse automáticamente en un servidor en producción.

Teniendo en cuenta una filosofía basada en DevOps, tanto el equipo de operaciones como el equipo de desarrollo trabajarán en varias fases para fomentar la colaboración entre los equipos de infraestructura y desarrollo.

5 Conclusiones

Al adoptar una herramienta de gestión de la configuración en nuestro servidor local, fortaleceremos el control, la coordinación y la eficiencia en la gestión de los distintos equipos de investigación. Además, mejoraremos la calidad de nuestros servicios y seguiremos avanzando hacia la automatización para optimizar nuestras prácticas y procesos de desarrollo.

Por otra parte, se pretende lograr que la implementación sistemática de distintas herramientas dé soluciones a los diferentes equipos de investigación; teniendo como último objetivo implementar la filosofía DevOps [6] en el área de infraestructura.

Para esto, se pretende de manera escalonada ir incorporando tecnologías de contenedores, orquestadores, infraestructura como código, CI/ CD [4], entre otras.

Referencias

1. ThinkSystem SR530 Setup Guide. (2020). Lenovofiles.com.
https://thinksystem.lenovofiles.com/help/topic/7X07/setup_guide.pdf
2. Ferrara, D. Nahuel, L. Di Santo, L. Basalo, A. Bertuzzi Gaspari, A. Rodríguez, E.: Una experiencia de implementación de infraestructura informática: recorriendo el camino desde lo académico hasta la instalación y puesta en funcionamiento: XXVIII Congreso Argentino de Ciencias de la Computación (CACIC), pp. 510—514. (2023) <http://sedici.unlp.edu.ar/handle/10915/149422>
3. GitLab Docs, <https://docs.gitlab.com/ee/install/index.html>
4. GitLab CI/CD, <https://docs.gitlab.com/ee/ci/>
5. Gitlab – Platzi, <https://platzi.com/clases/1526-gitlab/19282-planificacion-en-gitlab-issues/>
6. Belalcázar, A: Arquitectura de un Data Center con Herramientas DevOps, pp. 21—25. (2017) <http://sedici.unlp.edu.ar/handle/10915/63838>