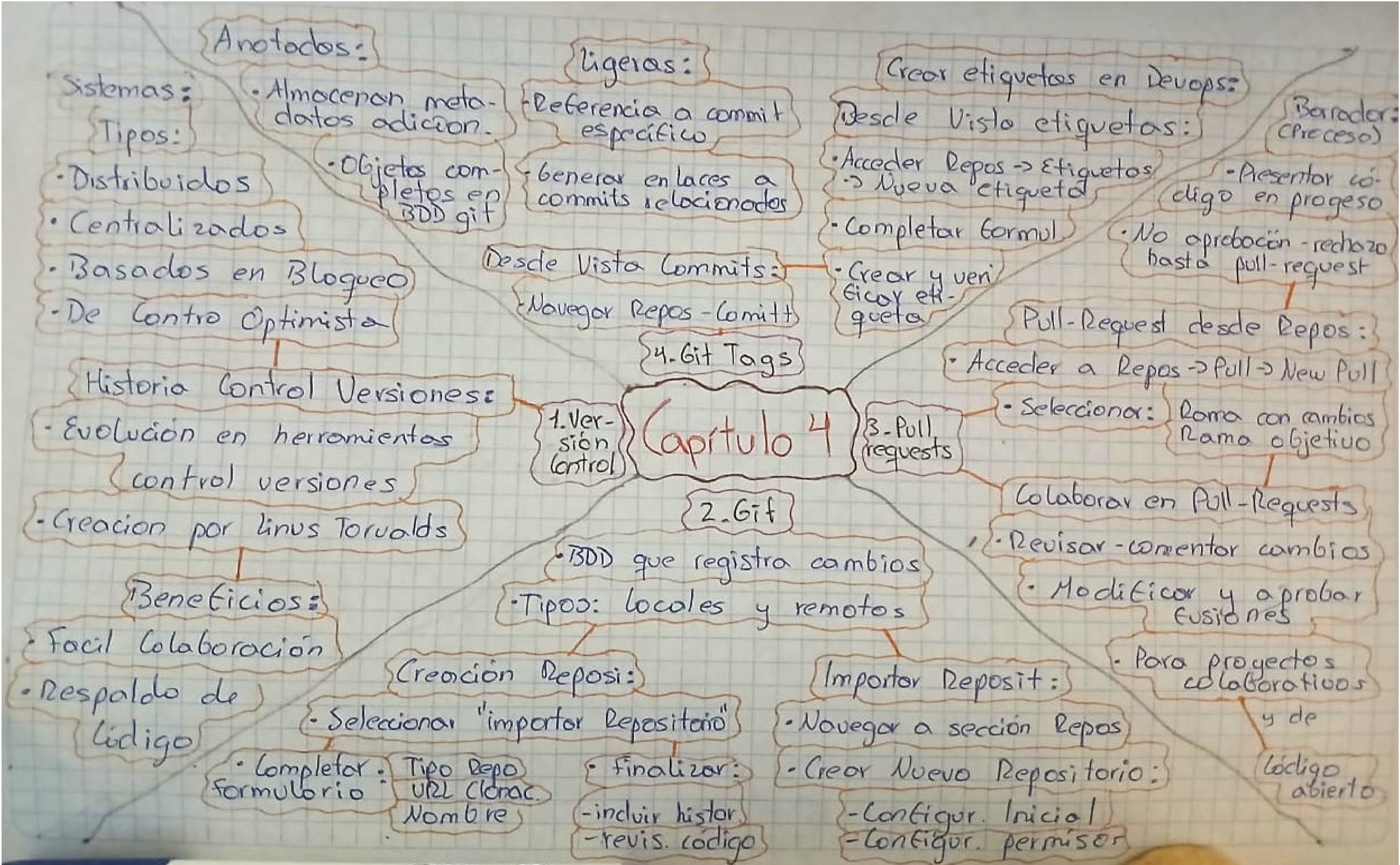


Mapa Mental ”Control de Version con Azure Repos”

Cuenca Buele Ronald Paúl, Carrera de Computación, Universidad Nacional de Loja, FEIRNR (ronald.cuenca@unl.edu.ec)
Advisor: Prof.Cristian Narváez G.

graphics

Imagen del Mapa Mental



Conclusiones

- C01: Azure DevOps nos proporciona herramientas poderosas para poder manejar el control de versiones mediante Git, haciendo un importe fácil de repositorios mediante plataformas externas como GitHub. Su integración de pull requests y posibilidad de realizar revisiones de código optimiza la colaboración dentro de los equipos de desarrollo, fometnado culturas de retroalimentación constante y mejoras evolutivas.
- C02: Los pull requests nos son de gran ayuda ya que gestionan los cambios en el código de manera controlada. Los equipos pueden revisar, comentar y aprobar los cambios antes de fusionarlos en la rama principal, garantizando así la calidad del código. Su uso de etiquetas (Git Tags) permite marcar puntos específicos en la historia del código, siendo esto de suma importancia para identificar versiones estables o de lanzamiento, mejorando la trazabilidad y la gestión de versiones.
- C03: Azure DevOps no tan solo tiene limitantes que se llegan hacia la gestión del código fuente, sino que también estas automatizan el proceso de integración y entrega continua (CI/CD) mediante los Azure Pipelines, permitiendo a los equipos poder reducir los errores manuales, optimizar sus entregas y mantener la calidad del software a lo largo del ciclo de vida. Esta combinación de herramientas como las de los pull requests, Tags y algunas opciones de automatización deja en evidencia de que Azure DevOps es una plataforma integral para el desarrollo ágil y controlado del software.

References

- [1] A. Nwodo y A. Silver, Beginning Azure DevOps: Planning, Building, Testing and Releasing Software Applications on Azure. Wiley Sons, Inc., John, 2023.

Enlace — URL

<https://drive.google.com/file/d/1BMOmns-EMjRc0nWoBnxpK-Z1NGVVVkyZ/view?usp=sharing>