

**Seguridad informática.**

Presentado a:

LEYDY KATHERINE LESMES FERNANDEZ

NRC: 10-57595

Presentado por:

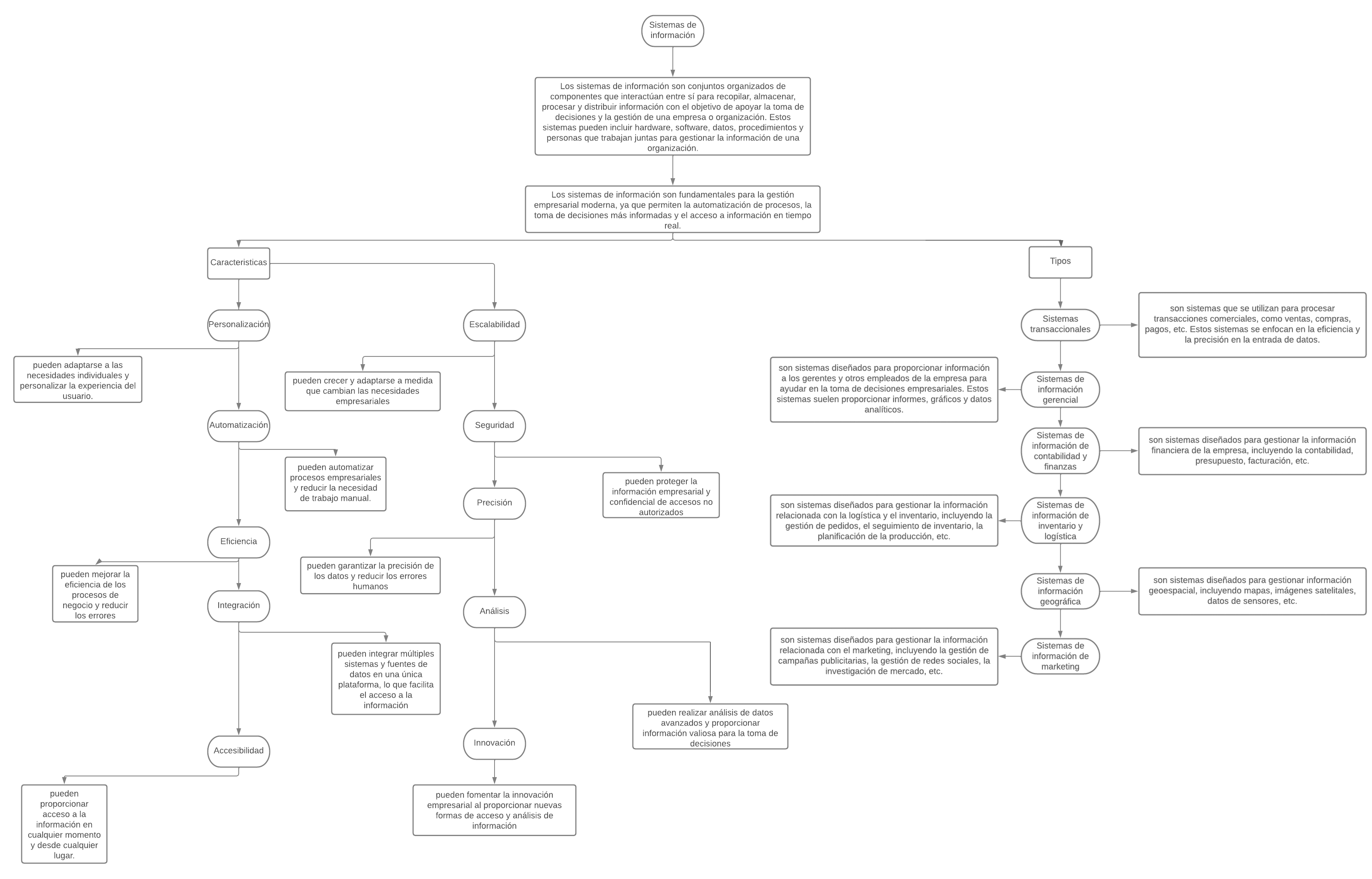
Anderson Gómez ID: 412199

Universidad Minuto de Dios

Facultad de Ingeniería

Bogotá – 2022

**1.Realice un mapa de ideas donde se sinteticen las ideas principales mencionadas en el video Recurso 1 ¿Qué son los sistemas de información? Características, tipos y ciclo de vida que encuentra en la sección de recursos del aula virtual.**



**2.Amplie y profundice los conceptos sobre Arquitecturas de cómputo, Modelo Von Neuman y Máquinas de Turing y mediante un cuadro sinóptico sintetice sus principales características.**

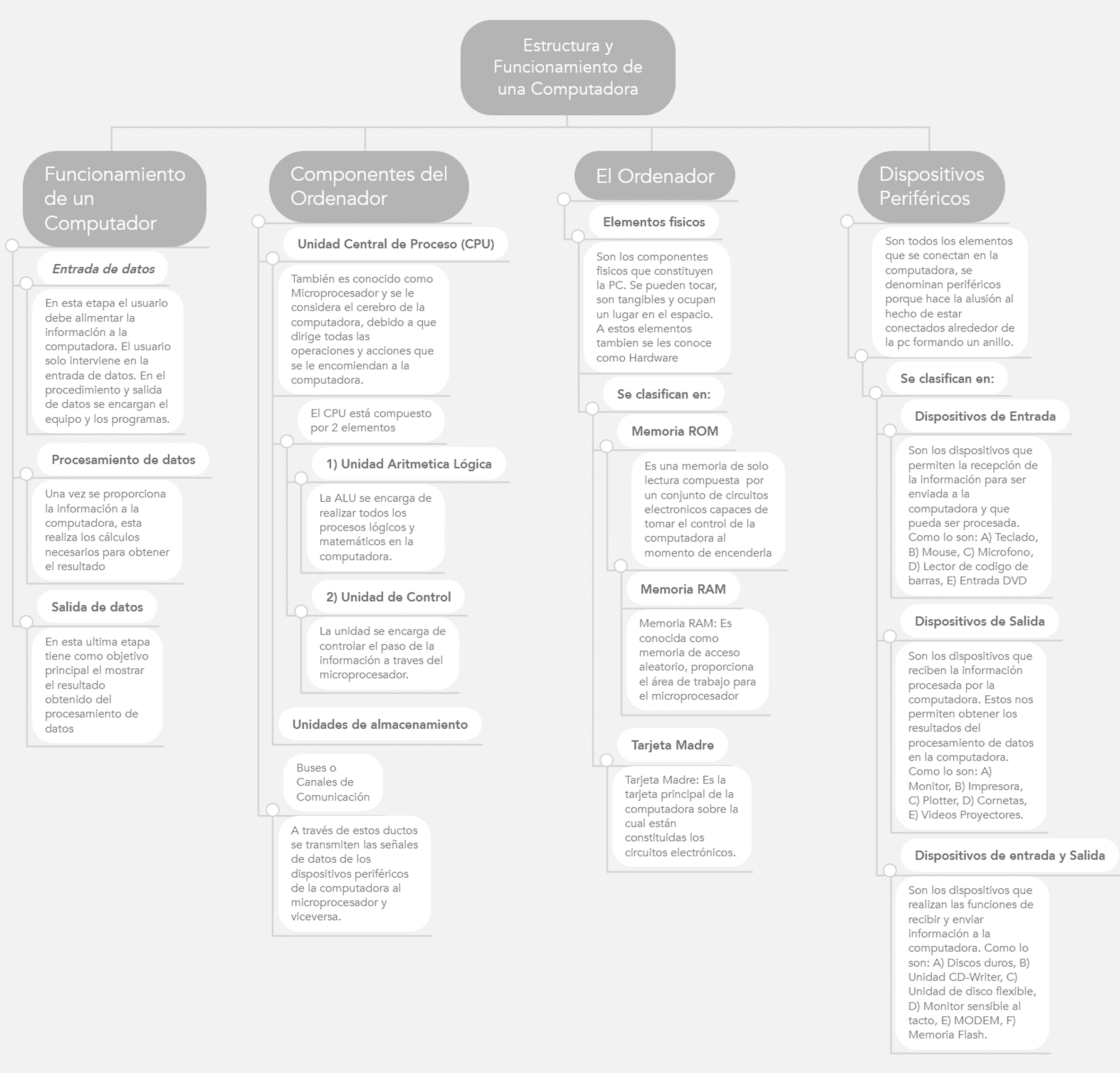
**Opción 1: (Ingresar desde el navegador)**

**URL:** [Teoría de la computación y desarrollo de las computadoras modernas - MindMeister Mapa Mental](https://www.mindmeister.com/map/2748816573)

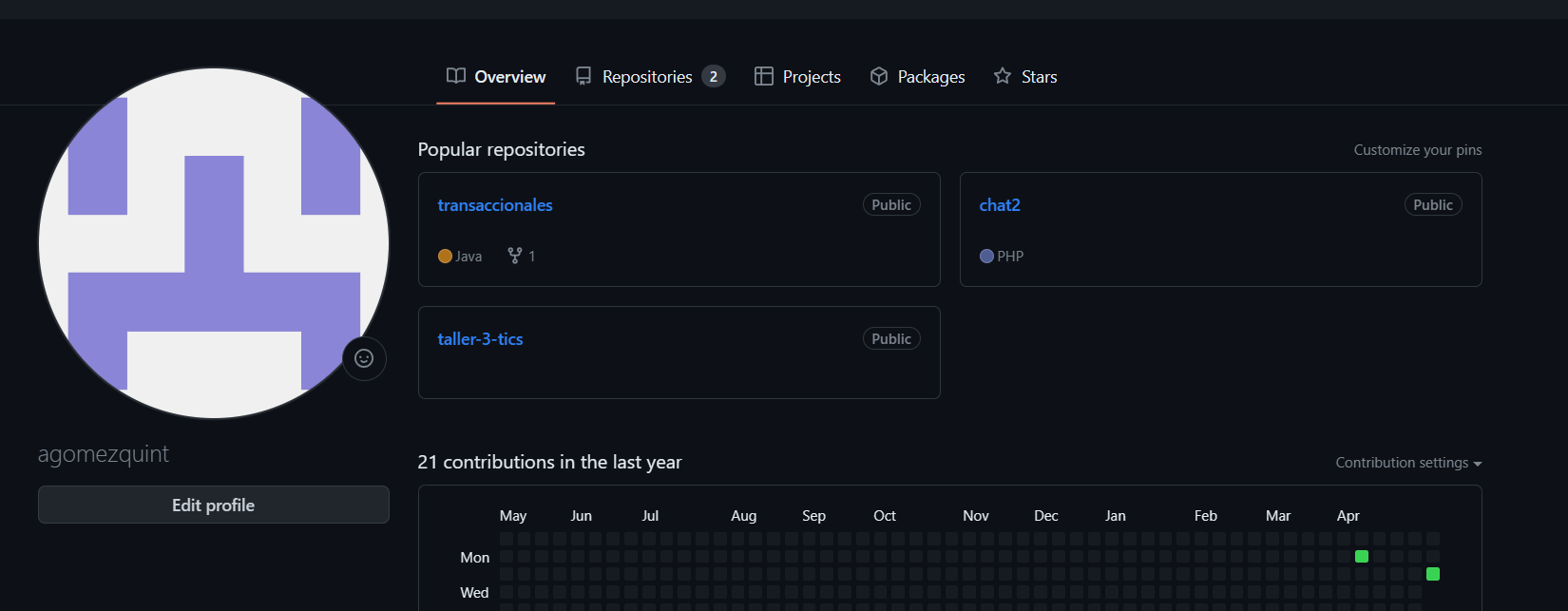
**Opción 2: (Ingresar desde el navegador)**

[**https://www.mindmeister.com/map/2748816573**](https://www.mindmeister.com/map/2748816573)

**3. Para un profesional del área de ingeniería de sistemas, sin importar el énfasis que adopte en su carrera, es fundamental conocer como es la estructura funcional de una computadora. Investigue y recopile de forma gráfica (esquema, mapa conceptual, infografía etc) la estructura funcional de una computadora. Acompañe el gráfico con un breve texto explicativo que describa el gráfico diseñado.**

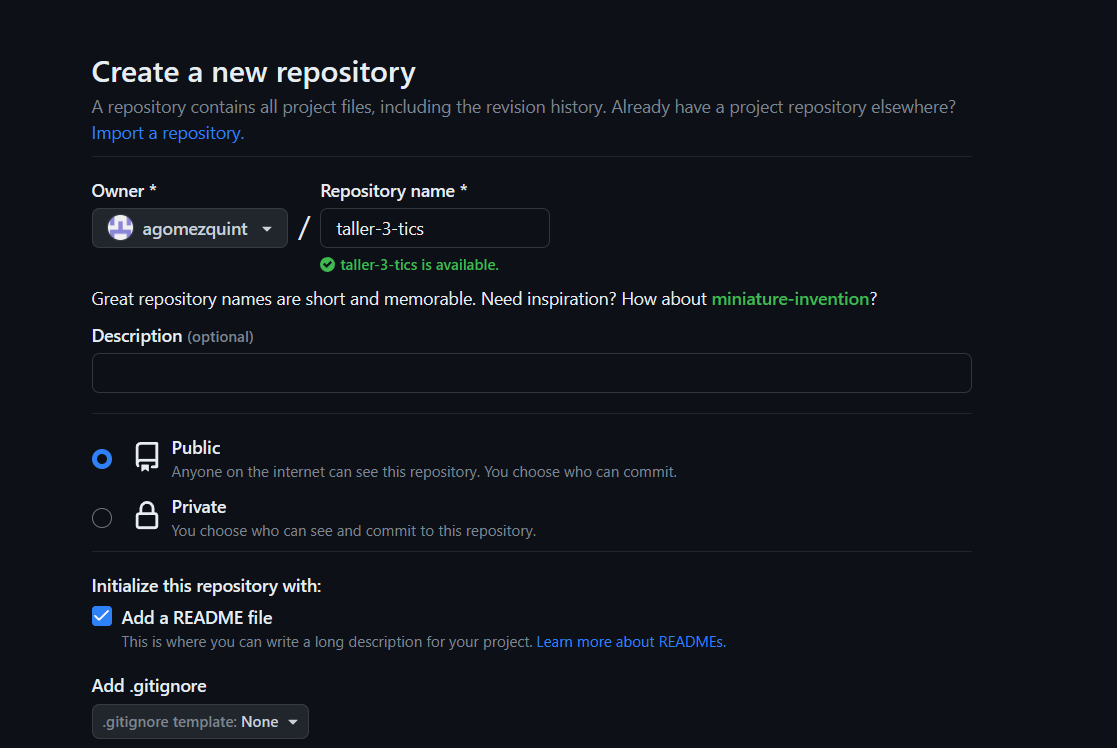


**. Leer el recurso 4. Introducción a GIT Y GIT HUB y realizar a práctica que a continuación se solicita.**

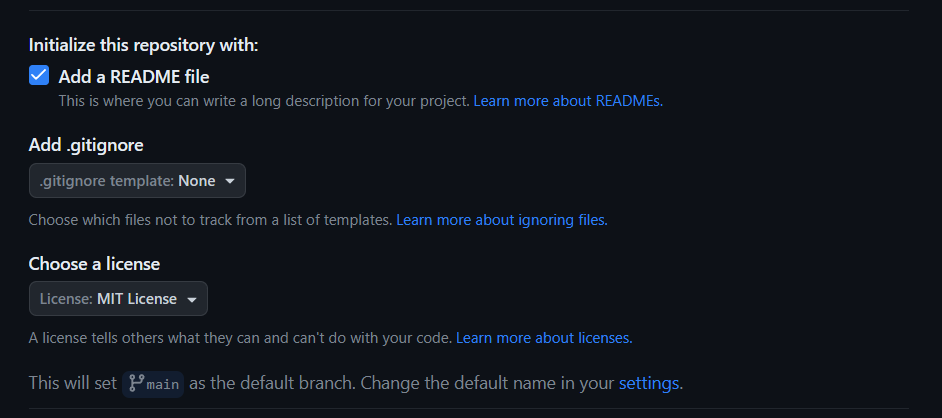


**Inicialice un repositorio local git en una carpeta denominado repositorios-ini**

**❑ Crear un repositorio remoto en git hub denominado taller-3-tics**



**❑ Crear los archivos README, .gitignore para unity y licencia tipo MIT**



**❑ Enlazar el repositorio local con el remoto de git hub**

**❑ Efectuar un git pull para actualizar el repositorio local**

**❑ Crear en él repositorio local un documento de Word donde recopile la siguiente**

**investigación:**

**- ¿Qué es y para qué sirve el archivo .gitignore ?**

es un archivo de configuración utilizado por el sistema de control de versiones Git para especificar los archivos y directorios que deben ser ignorados por Git en una determinada repositorio, este archivo se coloca en el directorio raíz del repositorio Git y se lee automáticamente por Git en cada operación de seguimiento o registro de cambios. Al agregar patrones de nombres de archivo o directorio al archivo .gitignore, se asegura que estos archivos no sean versionados por Git y no sean incluidos en los cambios registrados.

Cuando trabajamos en un proyecto de software, a menudo hay archivos o directorios que no deben ser versionados por Git. Por ejemplo, archivos de configuración específicos del equipo o sistema operativo, archivos de compilación o archivos de registro temporales. Si estos archivos se incluyen en el control de versiones, pueden crear problemas de conflicto y aumentar innecesariamente el tamaño de la base de código.

El archivo .gitignore permite especificar patrones de nombres de archivos o directorios que deben ser ignorados por Git. Por ejemplo, se puede especificar que se deben ignorar todos los archivos que tienen una extensión .log o un nombre determinado, como "config.ini".

**- ¿Qué es y para qué sirve el archivo README.md ?**

Es un archivo de texto que se encuentra en la raíz de un proyecto de software y contiene información sobre el proyecto y cómo utilizarlo. El archivo README.md es un archivo de formato Markdown, que es un lenguaje de marcado ligero utilizado para formatear el texto en la web , este archivo es importante porque proporciona información básica sobre el proyecto, como su propósito, cómo instalarlo, cómo utilizarlo, cómo contribuir al proyecto, cómo reportar problemas y cómo contactar al equipo de desarrollo, tambien permite a otros desarrolladores comprender rápidamente el propósito y la funcionalidad del proyecto, así como cómo pueden contribuir al mismo.

**- ¿Qué tipo de licencias se pueden usar en un repositorio de git hub?**

En GitHub, los repositorios pueden utilizar diferentes tipos de licencias de software que determinan los términos y condiciones en los que se puede utilizar, modificar y distribuir el código fuente del software. A continuación, se describen algunos de los tipos de licencias de software que se pueden utilizar en un repositorio de GitHub:

* Licencias de código abierto: estas licencias permiten el acceso y la modificación del código fuente del software, y permiten su distribución libremente. Ejemplos de licencias de código abierto incluyen la Licencia MIT, la Licencia Apache, la Licencia GPLv3, y la Licencia BSD.
* Licencias de software propietario: estas licencias restringen el acceso y la modificación del código fuente del software, y su uso suele estar limitado a un número determinado de usuarios o dispositivos. Ejemplos de licencias de software propietario incluyen la Licencia Propietaria y la Licencia de Usuario Único.
* Licencias de dominio público: estas licencias permiten la libre distribución y uso del software sin restricciones, ya que renuncian a todos los derechos de autor y propiedad intelectual. Ejemplos de licencias de dominio público incluyen la Licencia de Dominio Público de Creative Commons y la Licencia de Unlicense.
* Licencias de atribución: estas licencias permiten la modificación y distribución del software, pero exigen que se reconozca la autoría original del código fuente. Ejemplos de licencias de atribución incluyen la Licencia Creative Commons y la Licencia BSD.

**❑ Efectuar los pasos necesarios para realizar un commit que diga “Se agregó archivo de investigación git hub”**

**❑ Actualizar el repositorio remoto git hub con un git push**

**❑ Enviar a Aulas virtuales el enlace del repositorio de git hub creado**

**Referencias:**

Xataka. (2021, 13 mayo). ¿Qué es GitHub y qué le ofrece a los desarrolladores?. Xataka. <https://www.xataka.com/basics/que-github-que-que-le-ofrece-a-desarrolladores>

Hostinger. (s.f.). ¿Qué es GitHub? Tutorial para principiantes. Hostinger. <https://www.hostinger.co/tutoriales/que-es-github>

Belar, A. (s.f.). Tema 4: Estructura del ordenador. Universidad de Valladolid. Recuperado de <https://www.infor.uva.es/~belar/IG/Tema4_Estructura_Ordenador.pdf>