|  |  |  |
| --- | --- | --- |
|  | **Carátula para entrega de prácticas** | |
| Facultad de Ingeniería | | Laboratorio de docencia |

Laboratorios de computación

salas A y B

|  |  |
| --- | --- |
| *Profesor:* | García Morales Karina |
|  |  |
| *Asignatura:* | Fundamentos de la Programación |
| *Grupo:* | 22 |
| *No. de práctica(s):* | 1 |
| *Integrante(s):* | 1 |
| *No. de lista o brigada:* | 17 |
| *Semestre:* | 2026-1 |
| *Fecha de entrega:* |  |
| *Observaciones:* |  |
|  |  |

CALIFICACIÓN: \_\_\_\_\_\_\_\_\_\_

**La computación como herramienta de trabajo del profesional de ingeniería.**

**- Objetivos.**

El alumno conocerá y utilizará herramientas de software que ofrecen las Tecnologías de la Información y Comunicación que le permitan realizar actividades y trabajos académicos de forma organizada y profesional a lo largo de la vida escolar, tales como manejo de repositorios de almacenamiento, búsquedas de información especializada y revisión de información arrojada por generadores de contenido mediante la escritura de un prompt.

**-Actividades:**

▪ Realizar búsquedas de información especializada.

▪ Revisar y validar contenido creado por inteligencia artificial generativa.

▪ En casa, crear un repositorio de almacenamiento en línea.

**-Introducción:**

El uso de dispositivos de cómputo y comunicación se vuelve fundamental para el

desempeño de muchas actividades, las cuales pueden ser de la vida cotidiana,

académica, profesional, empresarial e inclusive de entretenimiento.

Como futuros profesionales de la ingeniería, los estudiantes de esta disciplina requieren

conocer y utilizar las herramientas de las Tecnologías de la Información y

Comunicación (TIC) que les apoyen tanto en sus tareas académicas como en su próxima

vida profesional. De la gran gama de herramientas TIC existentes, en esta práctica nos

enfocaremos en las herramientas para manejo de repositorios de almacenamiento,

buscadores de información especializada en Internet y revisión de información

arrojada por generadores de contenido, las cuales permitirán a los estudiantes realizar

las siguientes actividades en apoyo a sus tareas académicas:

• Almacenamiento de la información de manera organizada en repositorios que

sean accesibles, seguros y que la disponibilidad de la información sea las 24

horas de los 365 días del año.

• Búsqueda de información especializada en Internet.

• Revisar información que arrojen las herramientas de generación de contenidos

**- Desarrollo de la Practica.**

Control de Versiones

Un controlador de versiones es un sistema de software que lleva a cabo el registro de los

cambios sobre uno o más archivos (sin importar el tipo de archivos) a lo largo del tiempo.

Estos sistemas permiten regresar a versiones específicas de nuestros archivos, revertir y

comparar cambios, revisar quién hizo ciertas modificaciones, así como proteger nuestros

archivos de errores humanos o de consecuencias no previstas o no deseadas. Además, un

control de versiones nos facilita el trabajo colaborativo y nos permite tener un respaldo

de nuestros archivos.

Actualmente esta herramienta es sumamente importante para los profesionistas del

software, sin embargo, su uso se extiende a diseñadores, escritores o cualquiera que

necesite llevar un control más estricto sobre los cambios en sus archivos.**Tipos de Sistemas de Control de Versiones**

Sistema de Control de versiones Local

Sistema de Control de Versiones Centralizado:

Estos sistemas están pensados para poder trabajar de manera colaborativa, por lo que un servidor central lleva el control de las versiones y cada usuario descarga los archivos desde ese servidor y sube sus cambios al mismo.

Sistema de Control de Versiones Distribuido:

En estos sistemas, los usuarios tienen una copia exacta del proyecto, así como todo el

registro de las versiones, de esta manera si el servidor remoto falla o se corrompe, los

usuarios pueden restablecer el servidor con sus propias copias de seguridad y obtener

los cambios en los archivos directamente del equipo de otros usuarios.

**Repositorio**

Es el directorio de trabajo usado para organizar un proyecto, aquí se encuentran todos

los archivos que integran nuestro proyecto, y en el caso de Git, todos aquellos que son

necesarios para llevar a cabo el control de versiones.

**Repositorio Local**

Es aquel que se encuentra en nuestro propio equipo y solo el dueño del equipo tiene

acceso a él.

**Repositorio Remoto**

Es aquel que está alojado en la nube, esto quiere decir, que se encuentra en un servidor

externo, el cual puede ser accedido desde Internet y que nos va a permitir tener siempre

a la mano nuestros archivos. Algunos de los proyectos que usan como base Git son:

github.com, bitbucket.org o gitlab.com, todos ofreciendo diferentes características.

**Github**

Es una plataforma de almacenamiento para control de versiones y colaboración. que

permite almacenar nuestros repositorios de una forma fácil y rápida, además nos da

herramientas para el mejor control del proyecto, posibilidad de agregar colaboradores,

notificaciones, herramientas gráficas y mucho más. Actualmente Github cuenta con más

de 31 millones de usuarios haciéndola la plataforma más grande de almacenamiento de

código en el mundo.

**Operaciones en un repositorio**

**Agregar:**

Esta operación agrega archivos en nuestro repositorio para ser considerados en el nuevo

estado guardado del proyecto. Por lo general son los archivos creados o que tienen

nuevas modificaciones.

**Commit:**

Esta operación se encarga de registrar los archivos agregados para generar un nuevo

estado (o versión) en nuestro repositorio, un commit puede registrar uno o más archivos,

y van acompañados de una explicación de lo que agregamos o cambiamos.

**Ramas (Branches):**

Nuestro repositorio se puede ver como un árbol, donde la rama principal (generalmente

llamada master) contiene nuestro trabajo revisado y funcionando. Una rama es una

bifurcación de otra rama en la cual podemos realizar nuevas modificaciones y pruebas

sin afectar los archivos que ya funcionan, una vez que hayamos terminado las nuevas

modificaciones sobre esa rama, se puede fusionar (merge) con la rama padre y ésta

tendrá los nuevos cambios ya aprobados.

**Almacenamiento en la nube**

El almacenamiento en la nube (o cloud storage, en inglés) es un modelo de servicio en

el cual los datos de un sistema de cómputo se almacenan,se administran y se respaldan

de forma remota, normalmente en servidores que están en la nube y que son administrados

por el proveedor delservicio. Estos datosse ponen a disposición de los usuarios a través

de una red, como lo es Internet.

**Google Drive, OneDrive, iCloud o Dropbox** son algunos espacios de almacenamiento

en la nube. Además, Google Drive (Google) y OneDrive (Outlook) cuentan con

herramientas que permiten crear documentos de texto, hojas de cálculo y presentaciones,

donde el único requisito es tener una cuenta de correo de dichos proveedores.

Este tipo de herramientas hace posible editar un documento y compartirlo con uno o

varios contactos, de tal manera que todos pueden trabajar grupalmente en un solo

documento.

Por lo tanto, los documentos creados pueden ser vistos, editados, compartidos y

descargados en cualquier sistema operativo, ya sea Windows, Mac OS o Linux, y en

cualquier dispositivo con capacidad de procesamiento como teléfonos inteligentes,

tabletas y computadoras.

**Actividades de tarea:**

1.- Investiga de dos repositorios además de GitHub con sus ventajas y desventajas.

2.- Realiza un cuadro comparativo de tres inteligencias artificiales que no vimos en el laboratorio, pueden ser de investigación, creación de presentaciones, generación de infografías, etc.

3.- Genera tu cuenta en GIT y sube tu práctica en Word a GitHub, al final de la práctica se debe agregar esta liga de GitHub(prueba que tu liga funcione en otro explorador diferente al que te encuentras en sesión, este punto se evalúa con el acceso a la liga y la validación de que se subió la práctica a GIT).

1.- Investiga de dos repositorios además de GitHub con sus ventajas y desventajas.

1-GitLab

Ventajas:

* Código abierto (puedes instalar tu propia instancia en un servidor).
* Integración nativa con **CI/CD** (automatización de pruebas, despliegues, etc.).
* Mejor control de **privacidad** y permisos en proyectos privados.
* Se puede usar en la nube (GitLab.com) o en tu propia infraestructura.

Desventajas:

* La versión gratuita en la nube tiene menos funciones que GitHub.
* La interfaz puede ser más compleja para principiantes.
* Consume más recursos si instalas la versión self-hosted.

2. Bitbucket (de Atlassian)

Ventajas:

* Integración perfecta con herramientas de Atlassian como **Jira** y **Trello**.
* Permite repositorios **privados ilimitados** (incluso en planes gratuitos, aunque con límite de usuarios).
* Compatible con **Git** y también con **Mercurial** (aunque ya está en desuso).
* Buena opción para equipos que ya usan Atlassian.

Deventajas:

* Menos popular que GitHub y GitLab, por lo que hay menos comunidad y soporte.
* Su interfaz puede sentirse menos intuitiva que la de GitHub.
* Algunas integraciones avanzadas requieren plan de pago.

2.- Realiza un cuadro comparativo de tres inteligencias artificiales que no vimos en el laboratorio, pueden ser de investigación, creación de presentaciones, generación de infografías, etc.

| **IA** | **Tipo / Enfoque** | **Ventajas** | **Desventajas** |
| --- | --- | --- | --- |
| **Perplexity AI** | Búsqueda e investigación con IA | - ✅ Responde con fuentes verificadas en tiempo real. - ✅ Ideal para investigaciones académicas o trabajos con citas. - ✅ Interfaz clara y rápida. | - ❌ No siempre tiene tanta “creatividad” como otras IA. - ❌ Algunas funciones avanzadas requieren suscripción. |
| **Canva (con IA)** | Diseño gráfico y creación de presentaciones | - ✅ Genera presentaciones, infografías y diseños de forma automática. - ✅ Muy fácil de usar, incluso sin conocimientos de diseño. - ✅ Plantillas variadas y exportación directa. | - ❌ Limitado en personalización avanzada si no se paga el plan Pro. - ❌ La calidad del contenido depende mucho de las plantillas. |
| **Notion AI** | Productividad y organización con IA | - ✅ Resume documentos, genera ideas y estructura notas. - ✅ Integrado en el ecosistema de Notion (bases de datos, gestión de proyectos, etc.). - ✅ Muy útil para estudiantes y equipos de trabajo. | - ❌ Requiere suscripción aparte de Notion. - ❌ No es tan potente para investigación profunda o diseño. |

3.- Genera tu cuenta en GIT y sube tu práctica en Word a GitHub, al final de la práctica se debe agregar esta liga de GitHub(prueba que tu liga funcione en otro explorador diferente al que te encuentras en sesión, este punto se evalúa con el acceso a la liga y la validación de que se subió la práctica a GIT).

Genera tu cuenta en GIT.

Interfaz de usuario gráfica, Texto

El contenido generado por IA puede ser incorrecto.

Sube tu práctica en Word a GitHub.

Conclusiones:

La computación, mediante herramientas como repositorios Git y plataformas colaborativas, empodera al profesional de ingeniería al ofrecer control de versiones, automatización, seguridad y colaboración distribuida. La integración de inteligencia artificial —como en creación de presentaciones o visualización de datos— amplía aún más esta capacidad, al facilitar la comunicación, el análisis y la productividad con herramientas rápidas, creativas y seguras.

Liga de GitHub:

Referencias:

1. http://rypress.com/tutorials/git

2. https://git-scm.com/book/es/v1/EmpezandoAcerca-del-control-de- versiones

3. https://www.dropbox.com/

4. https://scholar.google.com/

5. https://www.microsoft.com/en-us/research/project/academic/

6. https://www.sciencedirect.com/topics/biochemistry-genetics-andmolecular-biology/springerlink

7. https://www.researchgate.net/

8. https://www.base-search.net/