



Diario de Ingeniería - Tecnologías y Sistemas Web III.

⚙ Status	Done
----------	------

Semana 1 (Viernes 19 de Septiembre).

CMS (Content Management System), también conocidos como gestores de contenido pre fabricados.

Plataforma o software que permite crear, gestionar y publicar contenido digital sin necesidad de programar desde cero (ya tiene estructurado definidas para ser utilizadas), por lo que facilita la administración de textos, imágenes, menús, usuarios y funcionalidades mediante una interfaz gráfica y uno de sus objetivos es agilizar el desarrollo web y permitir que personas sin conocimientos técnicos profundos puedan mantener un sitio activo.

WordPress.

Uno de los CMS más utilizados en el mundo (utilizado un 43% de todos los sitios web que hay en internet). Es característico por su facilidad de uso, gran cantidad de extensiones y plantillas que permiten extender sus funciones. Trabaja mediante un sistema de temas basados en diseño, y complementos para agregar

características como blogs, tiendas en línea o sistemas de membresías. Está basado en PHP y utiliza MySQL como base de datos.

Joomla.

CMS de código abierto orientado a la creación de sitios web más estructurados o complejos. Destaca por su variedad de funciones, como su sistema de gestión de usuarios avanzado, su arquitectura modular y la posibilidad de crear portales, intranets, catálogos y aplicaciones web robustas. También está construido en PHP y usa MySQL o MariaDB como base de datos.

Drupal.

CMS altamente flexible y orientado al desarrollo de proyectos complejos, escalables y con estructuras de datos que pueden ser personalizables. Tiene un sistema mas robusto, con roles avanzados y un control profundo sobre la arquitectura del contenido. Es muy utilizado en sitios institucionales y gubernamentales debido a su seguridad y capacidad de personalización.

Servidores de la universidad.

URL: <https://10.90.30.219>

User: **SK**

Contraseña: **Parda99***

Credenciales usados localmente.

PHPMYADMIN

Usuario: **LuisAle**

Contraseña: **LuisAleLR-0303**

WORDPRESS

Usuario: **Alejandro03**

Contraseña: **AlejandroLR-03030**

Semana 2 (Viernes 26 de Septiembre).

Introducción de PHP.

Lenguaje de programación del lado del servidor enfocado en la creación de aplicaciones web dinámicas. Permite procesar formularios, gestionar archivos, conectarse a bases de datos y generar contenido HTML de forma automática. En cuatrimestres anteriores ya se había trabajado con PHP de manera superficial, por lo que en este cuatrimestre se profundiza en su estructura, su sintaxis y su integración con herramientas modernas del entorno web.

Fundamentos de php.

En el código que se nos brinda como recurso de aprendizaje, se incluyen variables, tipos de datos, estructuras de control, arreglos, funciones y el manejo de capturas (peticion y respuesta).

miSitio Inicio Tipos Estructuras Subrutinas Captura 1 Captura 2 Captura 3 Combo 1 Prueba		
Descripcion	Tipo	Valor
Lógico	boolean	1
Cadena	string	cadena
Entero	integer	13
Flotante	double	45.3
Fecha	string	11/Dec/2025
Hora Unix	integer	1765469081
Hora NO Unix	integer	10:04:41

Tipos Enteros

Descripcion	Valor
Decimal +	1234
Decimal -	-123
Octal	83
Hexadecimal	26

Uso de Vectores

Hola Mundo
 69
 45.4
 1

Array ([0] => Hola Mundo [1] => 69 [2] => 45.4 [3] => 1)

Hola Mundo
 69
 45.4
 1

Lectura de archivos .csv con valores separados por ; y creación de tablas.

Permite abrir y procesar archivos **.csv** donde los datos están estructurados por punto y coma (;).

Una vez interpretado el archivo, su contenido puede transformarse en estructuras tabulares dentro de una página web.

Este tipo de procesamiento facilita:

- Mostrar **datos generales del archivo**, como nombre, tipo y tamaño.
- Detectar y representar la **estructura de los encabezados**, interpretándolos como atributos o columnas.
- Construir una **tabla HTML interactiva**, capaz de ordenar, paginar y buscar información.

Enfoque útil para visualizar grandes cantidades de datos provenientes de sistemas externos.

miSitio	Inicio	Gráficas	Opción 2	Opción 3	Opción 4
---------	--------	----------	----------	----------	----------

Procesando archivo.

Datos Generales.

Nombre	Datos - PHP.csv
Tipo	text/csv
Tamaño	0.86 MBs

Estructura.

Cedula	Nombre	Sexo	eCivil	Nacido	Salario	Provincia	Politico
string	string	string	string	string	string	string	string

Datos.

Show **10** entries

Search:

Cedula	Nombre	Sexo	eCivil	Nacido	Salario	Provincia	Politico
101030051	Wright Flores, Ignacio	F	U	10/04/1978	1140000	Alajuela	Acción Ciudadana
101240489	Ugalde López, Yvone	M	C	14/03/1967	1235000	Guanacaste	Acción Ciudadana
101550906	Ibarra Pacheco, Wenceslao	M	S	13/08/1994	1710000	Limón	Renovación Costarricense
101880443	Ojeda Hidalgo, Nidia	F	V	23/07/1999	0	Limón	Republicano Social Cristiano
102970379	Sáenz Sanchez, Ulises	M	D	29/11/1999	2850000	Alajuela	Liberación Nacional
103570803	Espinoza Urbina, Pedro	M	S	03/08/1974	190000	Heredia	Acción Ciudadana
105810612	Ballesteros Tijerino, Ofelia	F	V	05/04/1998	1330000	Heredia	Renovación Costarricense
107670543	Ureña Huertas, Victor	M	U	01/05/1988	1140000	Heredia	Republicano Social Cristiano
108900815	Jerez Cespedes, Zacarías	M	V	24/07/1990	1045000	Cartago	Liberación Nacional
108980604	Mendez Abarca, Gabriela	F	V	23/07/1974	2470000	Alajuela	Frente Amplio

Showing 1 to 10 of 100 entries

Previous **1** 2 3 4 5 ... 10 Next

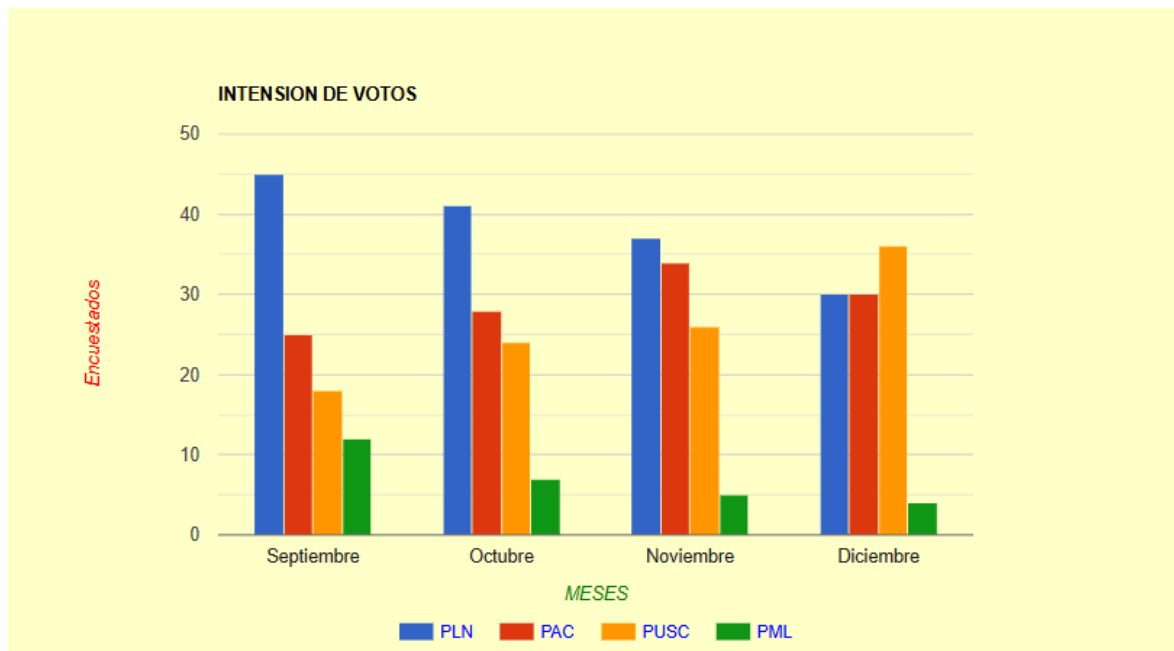
http://localhost/claas_01/index.php

Además muestra como utilizar un API de Google para la creación de Gráficas.

Google tiene servicios de visualización mediante APIs como **Google Charts**, que permiten convertir conjuntos de datos en gráficos dinámicos. PHP se utiliza para procesar la información, formatearla y enviarla al componente gráfico, mientras

que Google Charts se encarga del renderizado en el navegador mediante JavaScript.

Esto permite crear gráficos de barras, líneas o pastel integrados directamente en el sitio web, facilitando el análisis visual de la información proveniente de archivos .csv u otras fuentes.



http://localhost/class_02/index.php

Semana 3 (Viernes 3 de Octubre).

No hubo clases, fue el convivio con las otras sedes de la universidad y las actividades de las habilidades blandas.

Notas que tomé en la charla que dio York sobre inteligencia artificial con la API KEY de Gemini.

Uso de variables de entorno – `load_dotenv` .

El manejo de datos sensibles, como las API Keys, se realiza mediante variables de entorno.

`load_dotenv` permite cargar automáticamente estos valores desde un archivo `.env` para mantener el código seguro y ordenado.

Almacenamiento de la API Key de Gemini.

La clave de Gemini se registra dentro del archivo `.env` generado en el proyecto. Para evitar exponer la clave directamente en el código fuente y permite una gestión más segura y reutilizable.

Asignación del modelo y modo de operación.

La API de Gemini permite seleccionar el modelo con el que se quiere trabajar (por ejemplo, modelos orientados a texto, código o razonamiento).

Construcción del prompt.

El prompt es la instrucción que recibe el modelo. Debe estar redactado de forma **clara, precisa y coherente** para obtener mejores resultados. La calidad del prompt influye directamente en la calidad del análisis o generación de contenido.

Procesamiento de la solicitud.

Una vez enviado el prompt, la API procesa la petición y genera la respuesta según los parámetros definidos. PHP, Python u otros lenguajes pueden manejar esta solicitud dependiendo del proyecto.

Inicialización de funciones y recepción de resultados.

Las funciones creadas para estructurar la solicitud se ejecutan al inicio del proceso.

Luego, se espera la respuesta del modelo, que puede incluir sugerencias, análisis o mejoras al código ingresado.

Notas adicionales de la charla.

- En el ejemplo analizado, el modelo revisa el código completo y propone mejoras automáticamente. Sin embargo, **se debe validar lo que sugiere**, ya que el modelo podría recomendar cosas que no aplican o que requieren ajustes.
- Es necesario **instalar la librería correspondiente** para poder trabajar con la API (dependiendo del lenguaje usado).

Beneficios.

Análisis y depuración de código.

El modelo puede revisar código fuente, detectar errores, proponer mejoras, explicar funciones y acelerar la depuración.

Uso empresarial.

En ambientes corporativos, lo más recomendable es implementar un **servidor de IA interno**.

Esto evita exponer código confidencial en servicios de nube externos y protege la propiedad intelectual.

Uso en entornos locales.

A nivel local, este enfoque puede implementarse como un **ChatBot especializado** para tareas diarias, como análisis de fragmentos de código, refactorización o generación de documentación.

También puede adaptarse a múltiples lenguajes, ajustando apropiadamente los prompts para obtener mayor precisión.

¿Cómo obtener la API Key de Gemini?

URL directa (con sesión iniciada y vinculada a la misma cuenta usada en Android Studio):

<https://ai.google.dev/gemini-api/docs/api-key>

Pasos generales

1. Crear un nombre para la API Key.
2. Asociar un proyecto (opcional, pero recomendable para organización).
3. Guardar la clave en el archivo `.env`.
4. Preparar su uso con futuros servicios de Google como **Firebase**, lo cual facilita integraciones con Android Studio, React y otras tecnologías.

Semana 4 (Viernes 10 de Octubre).

Acceso a BD con PHP.

Permite conectar aplicaciones web con motores como MySQL para realizar operaciones de lectura, inserción, modificación y eliminación de datos. Esto se logra mediante extensiones como mysqli o PDO, que proporcionan funciones y métodos para ejecutar consultas y manejar resultados de forma segura.

SHA2 y MD5.

Algoritmos de hashing utilizados para transformar datos (generalmente contraseñas) en cadenas irreversibles.

- **MD5** es un método antiguo, rápido, pero vulnerable a colisiones y ataques de fuerza bruta.
- **SHA2** (como SHA-256 o SHA-512) ofrece mayor seguridad gracias a su estructura criptográfica más robusta.

Se utilizan para resguardar información sensible evitando guardar datos en texto plano.

AJAX (Asynchronous JavaScript and XML).

Permite realizar peticiones al servidor sin recargar la página completa. Esto mejora la experiencia del usuario al actualizar solo la parte necesaria de la interfaz, integrándose con PHP para obtener o enviar información de manera dinámica.

Rutas absolutas y relativas.

- **Rutas absolutas:** indican la ubicación exacta de un archivo dentro del sistema, desde la raíz del servidor.
- **Rutas relativas:** describen la ubicación de un archivo respecto al archivo actual.

Recursos y autoridades sobre carpetas en Linux.

En Linux, los permisos y autoridades determinan qué usuarios pueden leer, escribir o ejecutar archivos y directorios. Se gestionan por medio de comandos como `chmod`, `chown` y `chgrp`, y son esenciales para que aplicaciones PHP accedan correctamente a imágenes, cargas de archivos y datos sin comprometer la seguridad del servidor.

MVC con la guía.

MVC con la guía.

El patrón **Modelo–Vista–Controlador (MVC)** organiza un proyecto en tres componentes:

- **Modelo:** gestiona los datos y la lógica de negocio.
- **Vista:** presenta la información al usuario.
- **Controlador:** recibe las peticiones, coordina la interacción entre modelo y vista.

Con los conceptos y guías explicados, se nos asigna un trabajo de clases, que consistía en realizar ciertas adaptaciones a una guía para el montaje de la base de datos de Northwind en MySQL y del website basado en PHP, para darle soporte a la tabla de Categories.

Semana 5 (Viernes 17 de Octubre).

No asistí a clases.

MyBox (Como un Dropbox pero la version de York).

Proyecto diseñado como una alternativa simplificada a servicios de almacenamiento en la nube como **Dropbox** o **Google Drive**. Con el objetivo de

comprender cómo funcionan los sistemas de gestión de archivos en entornos web, incluyendo la organización de directorios, permisos de lectura/escritura y el uso del servidor web para manejar contenido dinámico.

El proyecto utiliza tecnologías como:

- **Apache2**, para servir los archivos y gestionar rutas.
- **PHP con CGI**, para procesar solicitudes, manejar operaciones de carga, descarga y listado de archivos.
- **Estructuras de carpetas y permisos**, esenciales para el control de acceso y la seguridad del sistema.
- **Interfaz visual incluida en un archivo .zip**, que permite visualizar el contenido y simular un gestor de archivos en línea.

Y el proposito es aprender a configurar el entorno del servidor, gestionar derechos efectivos y comprender el flujo interno de una aplicación de almacenamiento en la nube desde cero.



Universidad Técnica Nacional

Ejemplo

Inicio

Carpetas

Otras ▼

Acerca de ▼

Cerrar Sesión

Autenticación

Usuario:

Contraseña:

Aceptar

[Registrarse Aquí](#)

Semana 6 (Viernes 24 de Octubre).

Con base en la introducción que nos dieron a MyBox y el código, se nos asignó el laboratorio para desarrollar, en el cual había que tomar el código como referencia, había que mantener ciertas funcionalidades que ya estaban como:

1. El proceso para autenticar y registrar un usuario ante el sistema se mantiene.
2. El usuario podrá examinar el contenido de sus carpetas.
3. El usuario podrá subir, descargar, borrar sus propios documentos.

Y agregar las funcionalidades nuevas, el código montado en una VM, con FTP y Certificado SSL/TLS para el uso del protocolo HTTPS.

Semana 7 (Viernes 31 de Octubre).

Esta semana el decano de la universidad no mandó a clases virtuales por las lluvias (ese día no llovió XD).

Introducción a Firebase.

Plataforma de desarrollo web y móvil creada por Google que proporciona un conjunto de herramientas en la nube para construir aplicaciones escalables sin necesidad de gestionar servidores propios. Incluye servicios como bases de datos en tiempo real, Firestore, autenticación de usuarios, hosting, almacenamiento de archivos, funciones serverless y analíticas. Con el objetivo de simplificar el backend y permitir que el enfoque principal se dirija al desarrollo de la lógica y la experiencia del usuario.

En la clase virtual el profesor ejecutó una lista de pasos del uso del firebase con el fin de explicarnos cada funcionalidad, debido a que teníamos que implementarlo en el proyecto final.

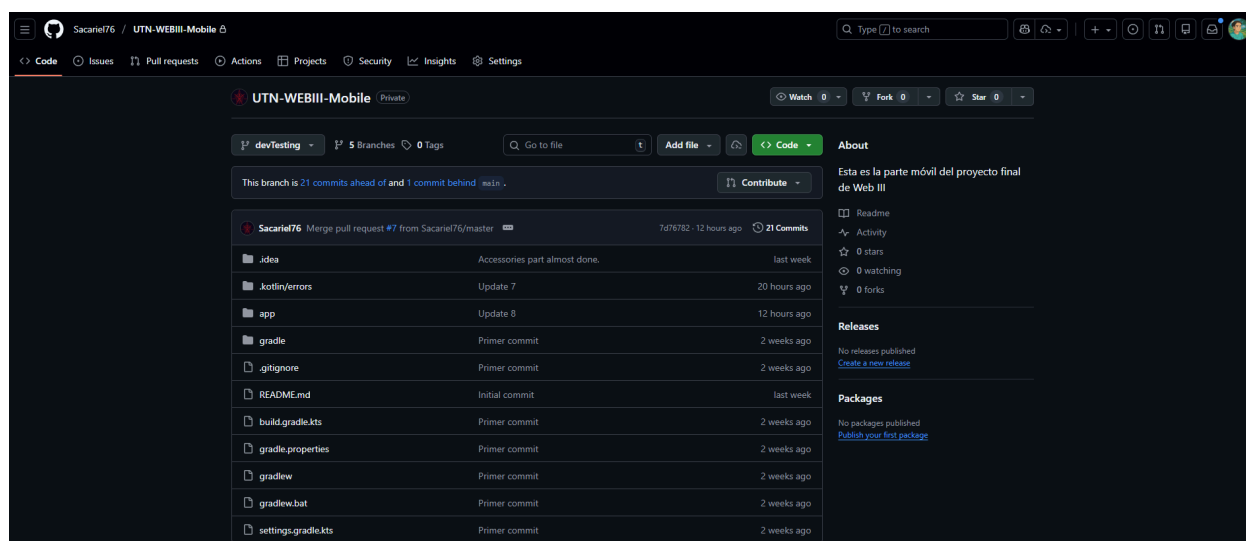
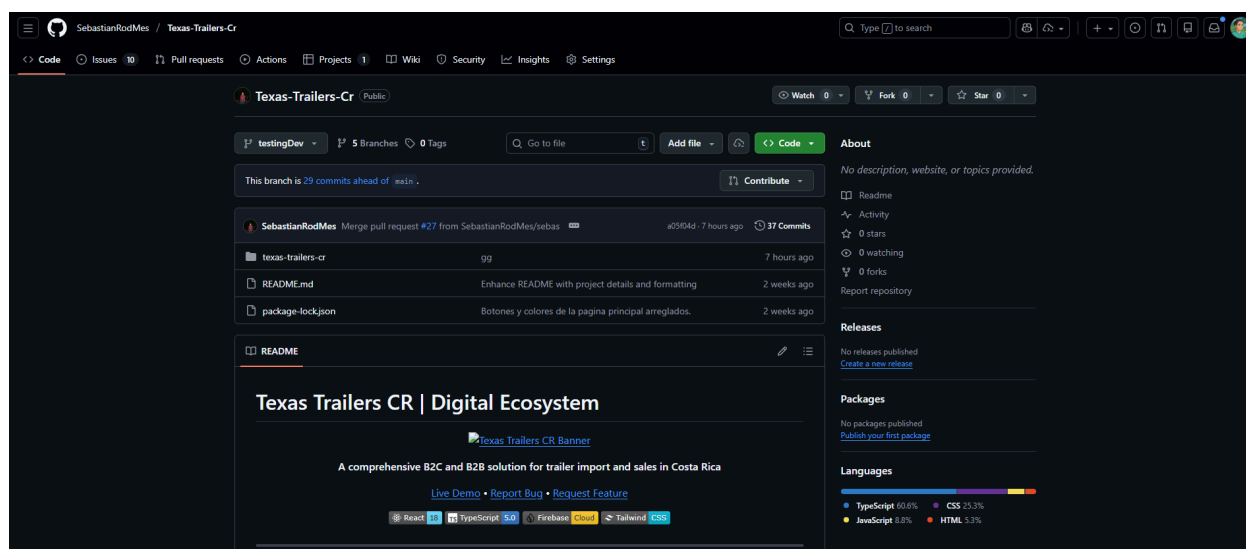
Y al final se nos asignaron 2 laboratorios, uno grupal y otro individual.

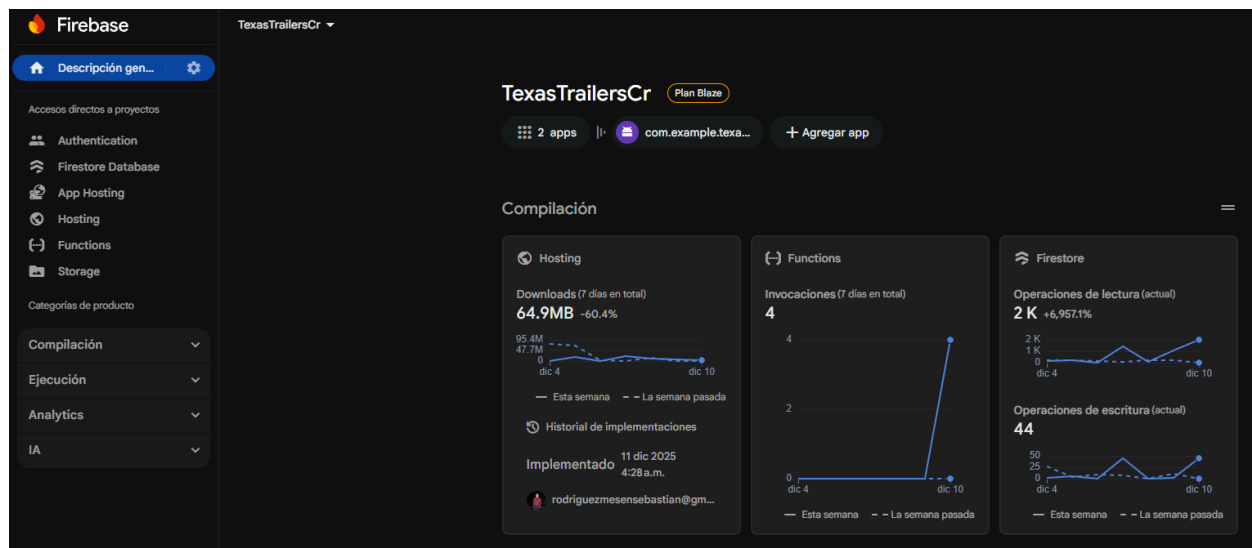
Semana 8 (Viernes 7 de Noviembre).

Se nos entregan las indicaciones para el desarrollo del proyecto final del curso, la pagina web para la empresa TEXAS TRAILERS de Orotina, enfocada en la venta de carretas y accesorios de alta calidad importados de USA, página que puede ser administrada por el dueño de la empresa desde una aplicación movil.

Semana 9 (Viernes 14 de Noviembre).

Una vez se definió el proyecto por realizar, y recibir aprobacion del profesor, empezamos con toda la preparacion del entorno de trabajo.





Semana 10 (Viernes 21 de Noviembre).

Sebastian Rodriguez, al ser el Tech Lead, nos compartió más información que nos podría ser de importancia en el desarrollo del proyecto, contexto de la empresa, lo que pide, forma de trabajo por medio de github (ramas e issues) y desarrollo de diagramas, todo eso puede ser consultado acá:

<https://drive.google.com/drive/folders/1WL8gxxhWVSQZur39eNIZIR8vWctGIJ966>

Semana 11 (Viernes 28 de Noviembre).

Continuación del desarrollo del proyecto.

Aplicación del examen I (Teórico sobre Firebase y Git).

Semana 12 (Viernes 5 de Diciembre).

Continuación en el desarrollo del proyecto y muestra al profesor de los avances que se tenían para recibir retroalimentación, posibles mejoras y ver los puntos que faltaban.

Semana 13 (Viernes 12 de Diciembre).

Entrega del proyecto final en el que se había venido trabajando.

Se asigna también la entrega del diario de ingeniería y el examen II que quedó pendiente por hacer para el lunes a las 11:59pm.

Note: Le hicimos la invitación a la organización de Github para la entrega de los labs y examen final, si ya se venció nos indicas para volverte a invitar :)