

	PRUEBA TÉCNICA PRÁCTICA PHP INGENIERO DE DESARROLLO	IN-FR-21 V2
---	--	--------------------

REQUERIMIENTOS

Para esta prueba técnica, se te pedirá desarrollar un CRUD de registro de documentos utilizando PHP. Deberás demostrar tu dominio en PHP, arquitecturas MVC y en la implementación de frameworks que operen con arquitecturas multicapa. La aplicación que desarrolles debe presentar una interfaz sencilla, implementada con una plantilla JavaScript. Es fundamental que asegures la correcta aplicación de la arquitectura multicapa en tu solución.

CARACTERÍSTICAS

- Login de usuario (especificar en el read.me el usuario y la contraseña)
- Logout de usuario.
- Tabla o grilla de datos de los documentos Búsqueda de registro de documentos.
- Creación de registro de documentos.
- Edición de registro de documentos.
- Eliminación de registro de documentos.

MODELO DE DATOS

- Tabla PRO_PROCESO esta tabla debe venir precargada con al menos 5 procesos, ejemplo: {PRO_ID: 1, PRO_NOMBRE: "Ingeniería", PRO_PREFIJO: "ING"}
- Tabla TIP_TIPO_DOC esta tabla debe venir precargada con al menos 5 tipos de documentos, ejemplo: {TIP_ID: 1, TIP_NOMBRE: "Instructivo", TIP_PREFIJO: "INS"}
- Tabla DOC_DOCUMENTO es la tabla principal donde se almacenan los registros de los documentos, el programa debe crear un código único consecutivo para cada documento con el siguiente lineamiento:

TIP_PREFIJO – PRO_PREFIJO – <Consecutivo único>

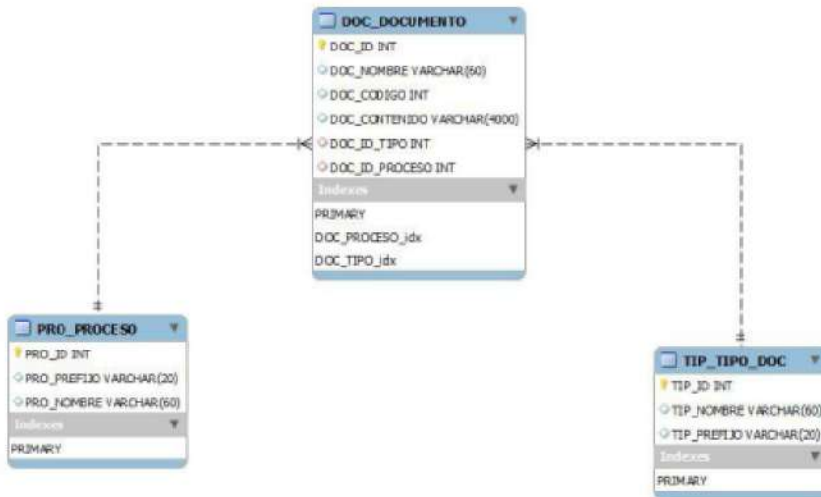
Ejemplo de código: Documento: INSTRUCTIVO DE DESARROLLO, Proceso: (ING) Ingeniería, Tipo: (INS) Instructivo, Código: INS-ING-1.

Ejemplo de registro de documento: DOC_ID: 1, DOC_NOMBRE: "INSTRUCTIVO DE DESARROLLO", DOC_CODIGO: "INS-ING-1", DOC_CONTENIDO: "texto grande con el contenido del documento", DOC_ID_TIPO: 1, DOC_ID_PROCESO: 1

Nota: No se deben repetir los consecutivos, es decir, si el consecutivo 1 ya fue usado en un documento con el mismo Tipo y Proceso, este número no debe volver a utilizarse.

Valores agregados: En la edición del registro del documento, si el tipo o el proceso del documento cambian, se debe recalcular el código.

A continuación, se presenta la estructura de la base de datos:



ENTREGABLES

- Instrucciones de despliegue, o si prefiere un demo disponible en línea.
- Diagrama de clases u otros diagramas que considere necesarios para ilustrar la solución.
- Es muy importante entregar los requerimientos solicitados completos.
- El código fuente de la aplicación.

TECNOLOGÍAS

- Backend: La aplicación debe realizarse utilizando PHP, usando el patrón MVC, implementando arquitectura por capas en el framework de su preferencia.
- Frontend: Cualquier tecnología en el cliente que consuma los servicios REST es permitida.

CRITERIOS DE EVALUACIÓN

- Funcionamiento correcto de la aplicación.
- Indicaciones de instalación. (readme)
- Arquitectura MVC o multicapa
- Principios SOLID, en especial SRP e Inyección de dependencias.
- Implementación de front sencillo

PUNTOS EXTRAS

- Buena documentación en Readme
- Uso de composer.
- MVC sin uso de framework o arquitectura multicapa
- Uso de ORM
- Historial de git con comentarios claros y concisos.
- Pruebas de unidad/integración con phpunit

Nota: Cualquier prueba con indicios de uso de IA como chatGPT o bard será descartada.