6-8-2025

Instituto Superior Adventista de Misiones

# EXAMEN FINAL

Asignatura: Programación I

Nombre del estudiante: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Duración del examen: 2 horas

## Introducción

Este examen final consiste en el desarrollo completo de un módulo funcional de software en un entorno de trabajo profesional. Deberás conectar tu proyecto a un repositorio remoto en GitHub, desarrollar un CRUD básico utilizando PHP, diseñar y crear la estructura SQL correspondiente y seguir políticas de trabajo predefinidas sobre nomenclaturas y organización de carpetas. No habrá teoría, solo práctica. El trabajo debe subirse correctamente al repositorio con la estructura solicitada.

## Criterios de Evaluación

|  |  |
| --- | --- |
| Criterio | Puntaje |
| Conexión del repositorio remoto y uso correcto de comandos Git | 15 pts |
| Estructura y funcionamiento correcto del CRUD en PHP | 35 pts |
| Creación y funcionalidad de la base de datos con SQL | 25 pts |
| Organización del proyecto y cumplimiento de las políticas de nomenclatura | 15 pts |
| Entrega completa en GitHub dentro del tiempo establecido | 10 pts |
| Total | 100 pts |

## Contexto del Problema

Una empresa de mensajería necesita registrar sus entregas diarias mediante un sistema simple que permita administrar sus envíos. Se solicita desarrollar un CRUD (Crear, Leer, Actualizar y Eliminar) que permita gestionar los envíos visualizando los envíos realizados en el tablero principal del cliente.

Cada envío debe tener: ID, nombre del cliente, dirección, código postal, estado del envío y fecha. Debés:  
- Crear un repositorio en GitHub llamado `crud-envios-final` y conectarlo con tu proyecto local usando los comandos Git.  
- Subir el proyecto completo a la rama `master`.  
- Se adjunta la base de datos:  
- Desarrollar el CRUD utilizando PHP (sin frameworks).  
- Cumplir con las políticas de nomenclatura y estructura de carpetas indicadas abajo.

## Políticas de trabajo y nomenclatura

- Todas las variables deben usar la nomenclatura **camelCase**.  
- Los nombres de archivos deben usar la nomenclatura **snake\_case,** estar en minúsculas, sin espacios, separados por guiones bajos (`\_`). Ejemplo: `form\_envios.php`.  
- Las carpetas principales deben ser:  
 \* `config/` → conexión a la base de datos.  
 \* `controllers/` → lógica de manejo de datos.  
 \* `views/` → formularios HTML.  
- No se permite el uso de frameworks externos.  
- El proyecto debe subirse completo al repositorio remoto antes del fin del examen.