

## Evaluación I

NOMBRE ALUMNO: \_\_\_\_\_ RUT: \_\_\_\_\_

FECHA: \_\_\_\_\_ NOTA: \_\_\_\_\_

### **Unidad de Aprendizaje I:**

Tecnologías del lado del  
Servidor.

### **Aprendizaje esperado**

1.1.- Realiza programa del lado  
del servidor, de acuerdo con la  
sintaxis del lenguaje.

### **Actividades**

1. Desarrollo de formulario  
que valide datos por lado  
servidor y que cree  
dinámicamente una lista  
de elementos.

## I. Presentación

Se le solicita la creación de una página web para atenciones médicas.

### • Criterios de evaluación

Esta actividad, considera los siguientes criterios de evaluación:

- 1.1.1 Utiliza variables y operaciones del lenguaje según requerimiento, analizando situaciones problemáticas establecidas.
- 1.1.2 Codifica instrucciones estructuras de decisión y operadores según requerimiento.
- 1.1.3 Codifica instrucciones de repetición según requerimiento.
- 1.1.4 Codifica instrucciones que implementen funciones y arreglos según requerimiento.

## II. Instrucciones

1. Utilice documentación vista en clases si así lo desea.
2. Puede utilizar frameworks para el diseño si así lo desea.
3. Desarrollar formulario web según requerimientos.
4. Una vez finalizado el proyecto comprimir todo en un archivo con el siguiente nombre “Evaluacion1\_NombreAlumno.zip”.
5. Cualquier uso de ia será penalizado con nota mínima.
6. Subir el archivo en ambiente de aprendizaje -> Programación Back End -> Evaluación I.

## III. Actividades

### 1. Desarrollo sitio formulario web

- a. Crear el proyecto Django:
  - Iniciar un nuevo proyecto Django llamado **gestion\_atenciones**.
  - Crear la aplicación **agendas** dentro del proyecto.
- b. Definir las clases y lógica:
  - Implementar la clase **Atencion** con los atributos **rut**, **nombre\_paciente**, **motivo\_atencion**, **fecha**, y **hora**.
  - Implementar la clase **Agenda** con la lista de atenciones y los métodos necesarios (**agregar\_atencion**, **eliminar\_atencion**, **modificar\_atencion**, y **buscar\_atencion\_por\_rut**).
  - **Asegúrese que no se pueden agregar atenciones si el rut ya existe o si la fecha y hora ya están ocupadas.**

**Nota:** La fecha se recibe como un tipo **string** mientras que la hora debe ser de tipo **int**.
- c. Crear las vistas:
  - Vista para listar todas las atenciones agendadas.
  - Vista para agregar una nueva atención (validar que no se repita el rut ni la combinación de fecha y hora).
  - Vista para eliminar una atención utilizando el rut.
  - Vista para modificar la fecha y la hora de una atención, asegurando que no se repita la nueva fecha y hora.
- d. Crear los templates:
  - Listar atenciones: Mostrar la lista de atenciones en un template, utilizando bucles para recorrer la lista de objetos y mostrando un mensaje si no hay atenciones.
  - Agregar atención: Crear un formulario para agregar una nueva atención, validando que los campos estén correctamente llenos.
  - Modificar atención: Crear un formulario para modificar la fecha y hora de una atención existente.
  - Eliminar atención: Agregar un enlace en el listado para eliminar atenciones.
  - Puede crear templates adicionales a estos.
  - **Envié mensajes correspondientes a los templates en caso de lograr o no la creación, eliminación y modificación.**
- e. Configurar las URLs:
  - Definir las rutas necesarias para acceder a cada vista: **listar\_atenciones**, **agregar\_atencion**, **eliminar\_atencion**, y **modificar\_atencion**.