

Evaluación 3

ÁREA ACADÉMICA	Tecnologías De Información Y Ciberseguridad	CARRERA	Ingeniería en Informática
ASIGNATURA	TI2041 Programación Back End	FECHA	04-11-2025
DURACION	120 minutos	DOCENTE	ROBERTO ALVEAL ORTEGA
Unidad de Aprendizaje	N° 3	PORCENTAJE EVALUACION	30%
Puntaje máximo	83 puntos / 49 puntos para una nota 4.0	Calificación:	

NOMBRE		RUT	
---------------	--	------------	--

INSTRUCCIONES GENERALES:

1. La nota 4.0 se obtiene logrando un 60% del puntaje total.
2. Utilice lápiz pasta en sus respuestas.
3. Preocúpese de la redacción, ortografía y legibilidad de sus respuestas.
4. Está prohibido el préstamo (o solicitud) de materiales durante la evaluación.
5. Se prohíbe el uso de celulares, iphone, ipod o similares durante la evaluación

Criterios de evaluación

- 2.1.1 Codifica funciones que realicen operaciones CRUD sobre una base de datos según requerimiento.
- 2.1.2 Configura conexión a una base de datos según requerimiento.
- 2.1.3 Utiliza Sesiones de validación de acceso y estado de colecciones según requerimiento, seleccionando métodos y procedimientos establecidos.
- 2.1.4 Cumple con las tareas asignadas en el tiempo definido.

Puntaje	Nota	Puntaje	Nota	Puntaje	Nota	Puntaje	Nota	Puntaje	Nota	Puntaje	Nota	Puntaje	Nota	Puntaje	Nota	Puntaje	Nota	Puntaje	Nota	Puntaje	Nota
0.0	1.0	10.0	1.6	20.0	2.2	30.0	2.8	40.0	3.4	50.0	4.0	60.0	4.6	70.0	5.2	80.0	5.8	90.0	6.4	100.0	7.0
1.0	1.1	11.0	1.7	21.0	2.3	31.0	2.9	41.0	3.5	51.0	4.1	61.0	4.7	71.0	5.3	81.0	5.9	91.0	6.5	101.0	7.1
2.0	1.1	12.0	1.7	22.0	2.3	32.0	2.9	42.0	3.5	52.0	4.1	62.0	4.7	72.0	5.3	82.0	5.9	92.0	6.5	102.0	7.1
3.0	1.2	13.0	1.8	23.0	2.4	33.0	3.0	43.0	3.6	53.0	4.2	63.0	4.8	73.0	5.4	83.0	6.0	93.0	6.6	103.0	7.2
4.0	1.2	14.0	1.8	24.0	2.4	34.0	3.0	44.0	3.6	54.0	4.2	64.0	4.8	74.0	5.4	84.0	6.0	94.0	6.6	104.0	7.2
5.0	1.3	15.0	1.9	25.0	2.5	35.0	3.1	45.0	3.7	55.0	4.3	65.0	4.9	75.0	5.5	85.0	6.1	95.0	6.7	105.0	7.3
6.0	1.4	16.0	2.0	26.0	2.6	36.0	3.2	46.0	3.8	56.0	4.4	66.0	5.0	76.0	5.6	86.0	6.2	96.0	6.8	106.0	7.4
7.0	1.4	17.0	2.0	27.0	2.6	37.0	3.2	47.0	3.8	57.0	4.4	67.0	5.0	77.0	5.6	87.0	6.2	97.0	6.8	107.0	7.4
8.0	1.5	18.0	2.1	28.0	2.7	38.0	3.3	48.0	3.9	58.0	4.5	68.0	5.1	78.0	5.7	88.0	6.3	98.0	6.9	108.0	7.5
9.0	1.5	19.0	2.1	29.0	2.7	39.0	3.3	49.0	3.9	59.0	4.5	69.0	5.1	79.0	5.7	89.0	6.3	99.0	6.9	109.0	7.5

1.- INSTRUCCIONES

La presente actividad tiene como objetivo evaluar el desarrollo de una aplicación web utilizando el framework Django, haciendo uso de operaciones CRUD y manejo de sesiones de usuario, así como la correcta implementación de servicios en la nube mediante AWS.

La actividad debe ser desarrollada de forma individual y será presentada presencialmente en la próxima clase, utilizando servidores desplegados en AWS.

2.- Descripción del Proyecto a Desarrollar

Desarrollar una aplicación web que permita realizar el control de existencias de una bodega en la cual existen diversos materiales que son facilitados a docentes para realizar las clases en una institución. La actividad a realizar principalmente el administrador puede registrar nuevos usuarios, materiales y docentes, mientras que el pañol permite registrar el préstamo de materiales a docentes

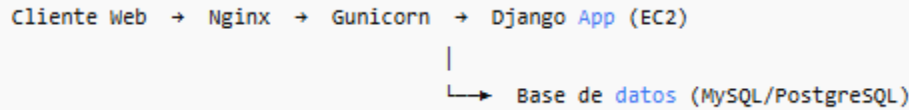
La aplicación debe considerar dos tipos de usuarios diferenciados por rol:

Rol	Permisos principales
Administrador	Registrar usuarios Pañol, docentes y materiales. Acceso completo a CRUD's.
Pañol	Registrar y visualizar préstamos de materiales a docentes. Sin acceso a administración.

El sistema debe permitir iniciar sesión, cerrar sesión y restringir el acceso a funcionalidades de acuerdo al rol del usuario autenticado.

3.- Arquitectura AWS obligatoria

Componentes	Requisito
EC2 Web	Amazon linux + acceso SSH con llave
Django	Entorno virtual + Gunicorn
Nginx	Reverse proxy para exponer app en puerto 80
EC2 BD	Base PostgreSQL o MySQL, con usuario dedicado, Oracle
Seguridad	No se permiten otros puertos abiertos a 0.0.0.0 excepto 80,443 y 22 y en BD solo los que corresponda



Las operaciones de CRUD (Create, Read, Update, Delete) sobre una base de datos mysql o postgres con django sobre servidores aws. Debe utilizar un servidor para los archivos web y un servidor de Base de datos

4.- Especificaciones de Desarrollo Django

1. El proyecto debe llamarse: **bodega**
2. La aplicación principal debe llamarse: **informatica**
3. Debe configurarse correctamente la conexión a la base de datos definida en AWS de manera local, sin exposición a la pública.
4. Generar las distintas vistas html
5. Crear página de inicio de sesión
6. Se deben implementar las siguientes funcionalidades:

4.1 CRUD obligatorios (solo administrador)

El usuario administrador debe realizar las operaciones CRUD que permitan la correcta funcionalidad del sistema, al menos debe contar con:

CRUD	Descripción
Docentes	Crear, listar, actualizar, eliminar docentes
Materiales	Crear, listar, actualizar, eliminar materiales
Administrador	Crear, listar, actualizar, eliminar administradores
Pañol	Crear, listar, actualizar, eliminar pañoles

Considere que puede activar o desactivar temporalmente un material, docente o pañol como mecanismo de control

4.2 Funcionalidad de préstamo (solo rol pañol)

- Registrar préstamo de material a docentes
- Visualizar listado de préstamos realizados

Asignar Materiales: Se solicita crear página que permita el préstamo de materiales a docentes, registrando fecha, docente, materiales y cantidad, solo el pañol puede realizar esta acción.

4.3 Control de sesión y seguridad

- Inicio de sesión obligatorio para ambos usuarios
- Restricción estricta por rol (no debe poder acceder a vistas ajenas)
- Evitar acceso a páginas por URL directa sin autenticación
- Sesiones no deben compartirse entre navegadores
- No expone credenciales ni claves en repositorio.
- Botón **logout**

4.4 Interfaz web

- Uso obligatorio de **Bootstrap, AdminLTE o similar**
- Uso de menú superior o lateral con navegación
- Inclusión de botón “Cerrar sesión” visible en todas las vistas internas
- Se acepta diseño libre mientras cumpla la estructura mínima solicitada
- Dependiendo su menú, en algún costado debe considerar una imagen tipo logo (invente un logo o cualquiera de internet), costado superior debe mostrar el botón que permite finalizar la sesión. Considere Plantilla base para cada vista
- Las vistas del administrador no pueden ser accesibles desde el usuario pañol
- Las vistas del pañol no pueden ser accesibles desde el usuario administrador
- Realizar uso correcto de sesión, las páginas no pueden ser accesibles por otro navegador.
- Validar las entradas para evitar inconsistencia en el registro de datos en la BD.

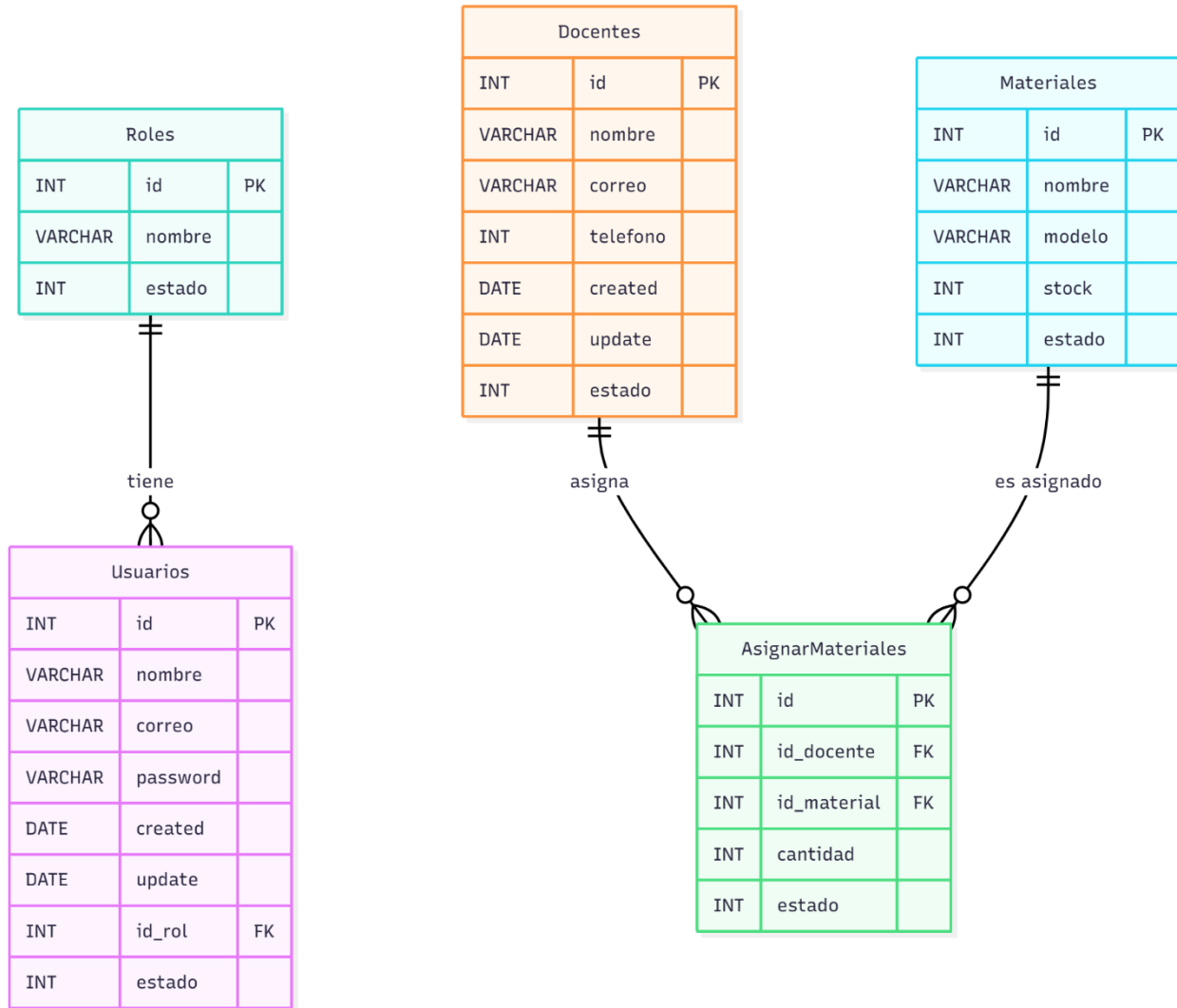
Para ambos CRUD debe realizar

- Crear archivo backend que permita realizar el registro en la BD (POST)
- Crear archivo backend que permita mostrar todos los datos de la tabla usuarios y materiales
- Crear archivo backend que permita actualizar un registro en la BD.
- Crear archivo backend que permita borrar un registro de la BD
- Generar acceso e inicio de sesión

Para asignar Materiales

- Crear archivo backend que permita realizar el préstamo en la BD (POST)
- Crear archivo backend y vista que permita visualizar la totalidad de materiales facilitado.

5.- Estructura de la base de datos



7.- Consideraciones

- Puede utilizar imágenes claras,
- Framework de vistas de su preferencia
- Utilice correctamente los nombres indicados para el proyecto y archivos.
- El ejercicio resuelto debe ser presentado en la próxima clase de manera presencial en su computador o equipo institucional, con máquinas implementadas en aws
- Aquellos trabajos que sean idénticos en más de un 50% serán considerados copias y tendrán la mínima evaluación.
- Los archivos deben ser subidos a la plataforma, debe realizar descarga en clases para presentación
- La rúbrica se encuentra disponible en la plataforma.