

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
CENTRO UNIVERSITARIO DE OCCIDENTE
DIVISIÓN DE CIENCIAS DE LA INGENIERÍA
INTRODUCCIÓN A LA PROGRAMACION 2



MANUAL TECNICO

DIEGO ALESSANDRO GONZALEZ BRAVO / 202230939

Introducción

Este documento describe la implementación y funcionamiento del sistema de gestión de anuncios, específicamente la funcionalidad para aprobar y denegar anuncios pendientes. Se detallan los componentes utilizados, la estructura del código y la interacción entre el frontend y el backend.

Herramientas de desarrollo.

Las herramientas (IDES, lenguajes de programación) utilizadas para el desarrollo de la aplicación Manejo de tarjetas son:

- JAVA
 - Version openjdk version "21.0.4"
- Apache NetBeans
 - Version 22
- Sistema operativo de desarrollo
 - Ubuntu 24.04 LTS
- Jakarta servlet
 - Version 3.0.0

Estructura del Proyecto

El proyecto está estructurado en varias capas que incluyen:

- **Frontend:** Implementado en JSP con HTML, CSS y JavaScript.
- **Backend:** Servlets y clases de acceso a datos para interactuar con la base de datos.
- **Base de Datos:** Maneja la persistencia de datos de los anuncios.

Funcionalidades Implementadas

1. Listar Anuncios Pendientes

```
protected void doGet(HttpServletRequest request, HttpServletResponse response)
```

```
    throws ServletException, IOException {  
        Conexion conexion = new Conexion();  
        conexion.conectar();
```

```
  
        AnunciosSQL anuncioDAO = new AnunciosSQL(conexion.connection);  
        List<Anuncio> anuncios = anuncioDAO.obtenerAnunciosPendientes();
```

```
  
        request.setAttribute("anuncios", anuncios);  
        RequestDispatcher dispatcher =  
        request.getRequestDispatcher("/jsp/AnuncioAdmin.jsp");  
        dispatcher.forward(request, response);
```

```
    Conexion.closeConnection();  
}
```

2. Aprobar Anuncio

Cuando un administrador aprueba un anuncio, se actualiza la base de datos con la fecha de inicio, fecha final y el costo del anuncio. Esto se realiza mediante una llamada AJAX que mantiene al usuario en la misma página.

Código AJAX en JavaScript

```
function aprobarAnuncio(idAnuncio) {  
    var fechaInicio = $('#fechaInicio' + idAnuncio).val();  
    var fechaFinal = $('#fechaFinal' + idAnuncio).val();  
    var pago = $('#pago' + idAnuncio).val();  
  
    $.ajax({  
        url: 'proyecto1/AprobarAnuncio',  
        type: 'POST',  
        data: {  
            idAnuncio: idAnuncio,  
            fechaInicio: fechaInicio,  
            fechaFinal: fechaFinal,  
            pago: pago  
        },  
        dataType: 'json',  
        success: function (response) {  
            if (response.status === "success") {  
                alert(response.message);  
                // Opcional: actualizar la tabla o realizar otras acciones  
            } else {  
                alert("Error: " + response.message);  
            }  
        },  
        error: function (xhr, status, error) {  
            alert("Error: " + error);  
        }  
    });  
}
```

Código del Servlet para Aprobar Anuncio

3. Denegar Anuncio

La funcionalidad para denegar un anuncio sigue una estructura similar a la de aprobación, y se puede implementar de manera similar en el código del JavaScript y del servlet.

Base de Datos

Se utiliza una tabla **Anuncios** con las siguientes columnas relevantes:

- **idAnuncio**: Identificador del anuncio.
- **texto**: Contenido del anuncio.
- **url_video**: URL del video asociado.
- **path_imagen**: Ruta de la imagen del anuncio.
- **activo**: Estado del anuncio.
- **fecha_inicio**: Fecha de inicio de la actividad del anuncio.
- **fecha_final**: Fecha de finalización de la actividad del anuncio.
- **pago**: Costo asociado al anuncio.
- **nombre_anunciante**: Nombre de la persona que publica el anuncio.
- **tipo**: Tipo de anuncio.

Conclusiones

Se ha implementado un sistema eficiente para gestionar anuncios pendientes, permitiendo al administrador aprobar o denegar anuncios con una interfaz clara y responsive. Las interacciones se manejan de manera asíncrona, mejorando la experiencia del usuario al no recargar la página.

Diagramas

Diagrama de entidad relación E/R

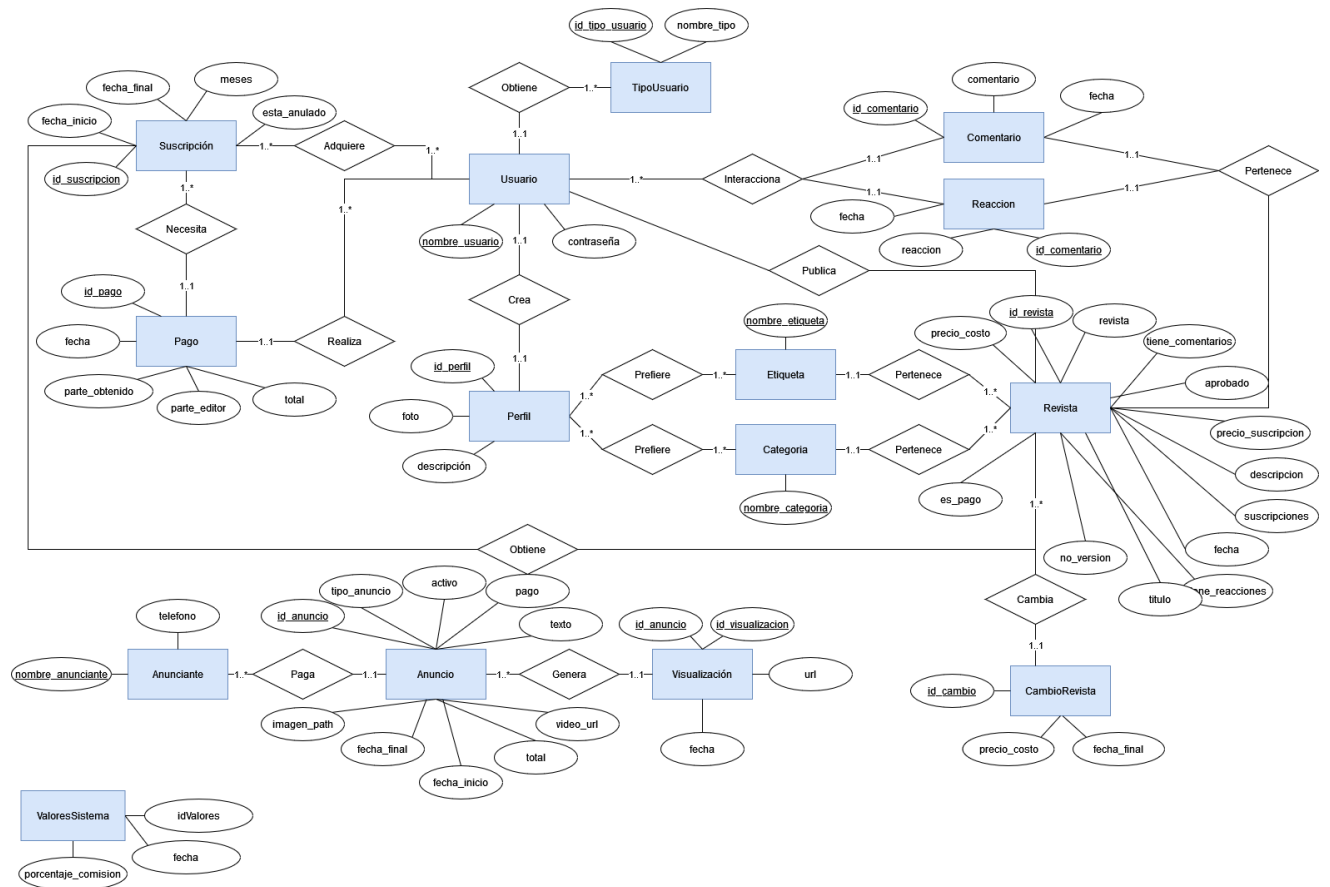


Diagrama de tablas

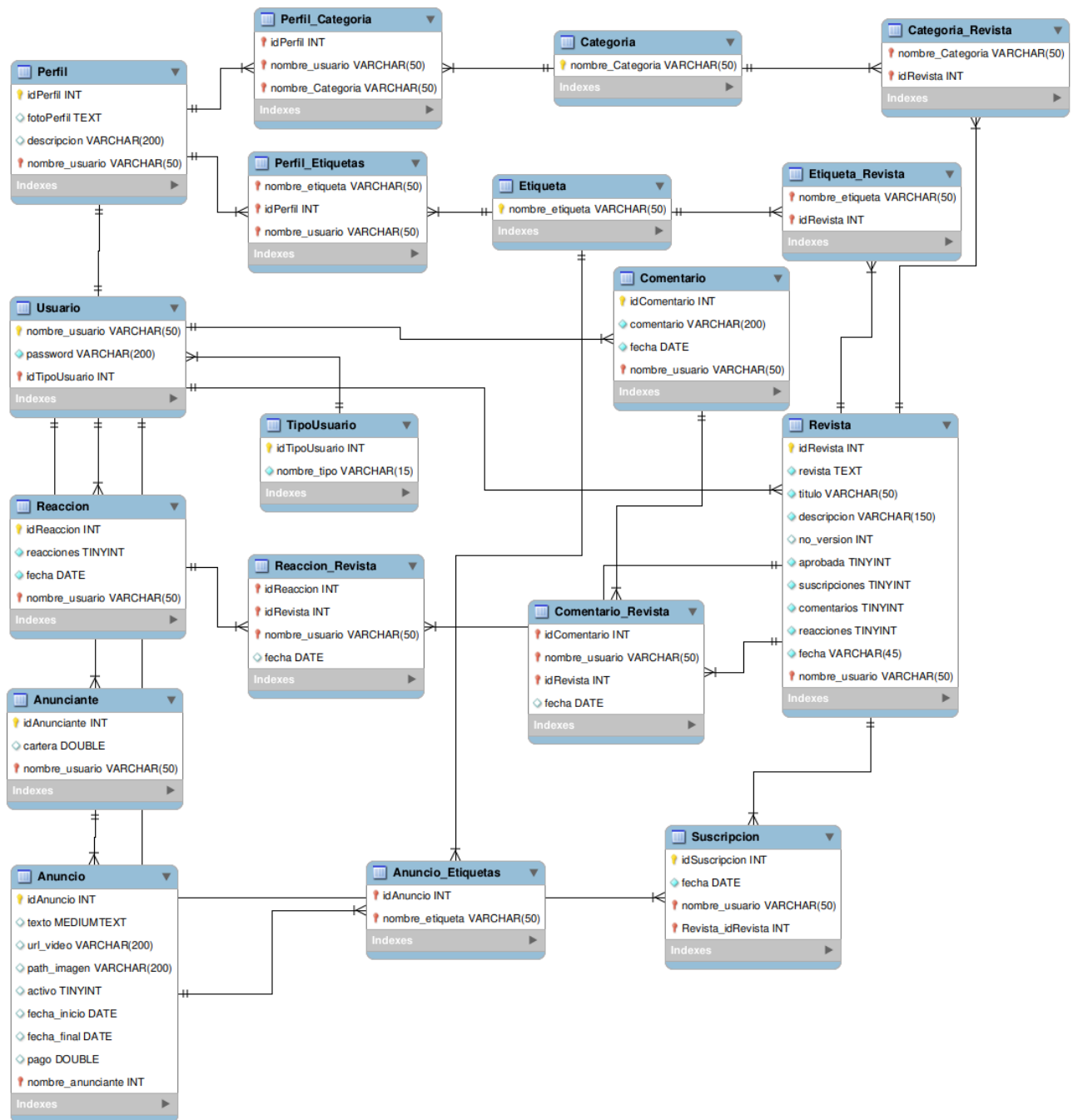


Diagrama de clases

