

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

LENGUAJE DE PROGRAMACIÓN 2

Examen 1
(Segundo semestre 2021)

Indicaciones generales:

- Duración: 2 horas.
- Materiales o equipos a utilizar:
ESTÁ PERMITIDO USAR APUNTES, DIAPOSITIVAS DE CLASE E INTERNET.
El Internet será utilizado solo para consultar documentación de programación.
Está prohibido toda forma de conversación con otras personas.
Se les recuerda que, de acuerdo con el reglamento disciplinario de nuestra institución, constituye una falta grave copiar del trabajo realizado por otro estudiante o cometer plagio para el desarrollo de este examen.
- **El uso apropiado de buenas prácticas de programación (sobre todo aquellas vistas en clase) será tomado en cuenta para la calificación.**

Puntaje total: 20 puntos

Parte Práctica: (20 puntos)

La oficina de *Dirección de Gestión de la Investigación* (DGI) de nuestra universidad desea contar con un software que le permita gestionar la información relacionada a los grupos de investigación. En este sentido, se le ha solicitado que realice la programación de las funcionalidades de “registro, listado y selección” de grupos de investigación. Para esto, se cuenta con un diseño previo de la interfaz (Figura 01 y Figura 02). Se le solicita descarga el proyecto base en C# o JAVA de PAIDEIA y realizar la programación de estas funcionalidades.

Formulario de Gestión de Grupos de Investigación

Nuevo Guardar Buscar Cancelar

Gestión de Grupos de Investigación PUCP

Datos Generales Integrantes

Nombre del Grupo: HCI, Design, User Experience, Accessibility & Innovation Technologies Research Group

Siglas: HCI-DUXAIT

Año de Fundación: 2019

Interdisciplinario: ☒ Si ☐ No

Sección: INGENIERÍA INFORMÁTICA

Foto del Grupo:

Descripción:

El grupo DUXAIT es un equipo que tiene como objetivo la investigación, el desarrollo y la difusión de trabajos y proyectos de investigación que contribuyan en las áreas de la Interacción Humano-Computador (HCI) y en el Desarrollo Tecnológico e Innovación.

Archivo PDF Información Adicional: Subir Descargar

C:\Users\fredd\Desktop\Info_Adicional.pdf

Figura 01. Interfaz gráfica – Datos Generales

Un grupo de investigación tiene los siguientes atributos: nombre, siglas, año de fundación, si es interdisciplinario o no, una descripción, una foto y un archivo PDF con información adicional. Asimismo, pertenece a una sección. Además de sus datos generales, posee integrantes, los cuales pueden ser profesores o alumnos de la universidad. En este formulario de registro, se podrán agregar/quitar a los integrantes que pertenecen al grupo de investigación. Asimismo,

cuando se liste un profesor se desea que se visualice su categoría en la cuarta columna. Mientras que cuando se lista un alumno, se desea que se visualice su CRAEST en la cuarta columna. De igual manera, en la segunda columna, se debe visualizar el tipo de integrante (Profesor o Alumno) (Ver Figura 02).

Gestión de Grupos de Investigación PUCP

Datos Generales **Integrantes**

Código PUCP: ...

Nombre Completo: + -

	Código PUCP	Tipo	Nombre Completo	Categoría/CRAEST
	19973102	Profesor	NICOL STEFANY RUIZ CASTILLO	AUXILIAR
▶	19992984	Profesor	WILMER JOEL SIFUENTES PAREDES	PRINCIPAL
	20112303	Alumno	CRISTIAN AVILA JIMENEZ	90.44
	20129830	Alumno	ELIZABETH DE LA CRUZ PEREZ	87.22

Figura 02. Interfaz gráfica - Integrantes

Se ha diseñado la base de datos que se estaría utilizando (Figura 03). En PAIDEIA encontrará un Script que permitirá generar todas las tablas y sus relaciones, así como los procedimientos almacenados que deben emplearse. Emplee el script en su instancia de base de datos MySQL de AWS.

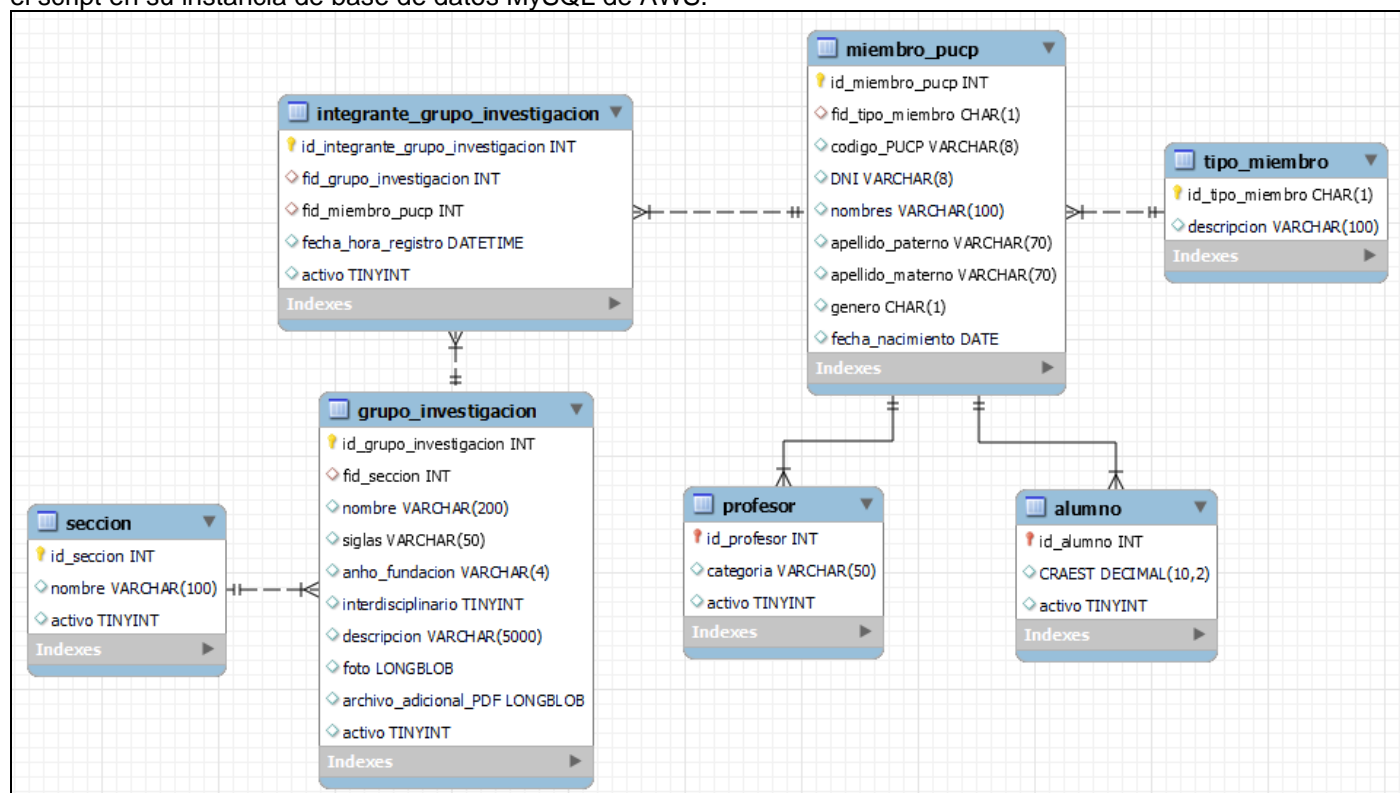


Figura 03. Diagrama ERR de la base de datos

No es necesario considerar validaciones. Asuma que el usuario ingresa toda la información correctamente y que no comete errores en el uso del producto de software.

Cuando liste los miembros PUCP, desarrollará una única función llamada “listarPorNombreCodigo(String nombreCodigo)” que permitirá devolver una lista de objetos de tipo “MiembroPUCP” donde se encuentren tanto alumnos como profesores. Identifique primero el tipo de miembro a través de la columna ‘id_tipo_miembro’ y luego instancie un objeto de tipo Alumno o Profesor dependiendo del tipo. Luego asigne el CRAEST o la categoría según sea el caso y finalmente agregue el objeto a la lista de objetos de tipo MiembroPUCP.

A continuación, se adjunta un video sobre el funcionamiento esperado: <https://youtu.be/uFrFzgLZ0g8>

Procedimientos almacenados a utilizar:

- LISTAR_SECCIONES_X_NOMBRE
- LISTAR_MIEMBROS_PUCP_X_NOMBRE_CODIGO_PUCP
- INSERTAR_GRUPO_INVESTIGACION
- INSERTAR_INTEGRANTE_DE_GRUPO_INVESTIGACION
- LISTAR_GRUPOS_X_NOMBRE
- LISTAR_INTEGRANTES_X_ID_GRUPO

Una vez culminado el ejercicio, suba su proyecto a PAIDEIA.

Rúbrica de Calificación:

- No se considerará puntaje alguno si es que el aspecto a evaluar no funciona a nivel de interfaz gráfica, por lo que se solicita que, a medida que va avanzando el *back-end*, realice la programación del *front-end*. Asimismo, la corrección de los apartados será progresiva. Para obtener el puntaje de un apartado en específico, todos los anteriores aspectos a evaluar deben estar correctamente desarrollados. Por ejemplo:
“no se considerará puntaje por la programación de la funcionalidad de registro, si es que para ese registro son necesarias algunas búsquedas y selecciones en el formulario y estas no se encuentran correctamente implementadas.”
 - El driver de conexión a base de datos debe estar referenciado con ruta relativa (no absoluta).
 - Se descontarán puntos significativamente si es que existen errores de compilación.
 - Debe utilizar la estructura vista en clase: proyectos relacionados a la capa de modelo, controlador y vista.
- A. Implementación de la clase DBManager. (1 punto).
B. Implementación de las clases SeccionDAO y SeccionMySQL. (2 puntos)
C. Implementación de la búsqueda, listado y selección de secciones a nivel de interfaz. (2 puntos)
D. Implementación de las clases MiembroPUCPDO y MiembroPUCPMySQL. (3 puntos)
E. Implementación de la búsqueda, listado y selección de miembros PUCP a nivel de interfaz (considerando botones agregar y quitar). (4 puntos)
F. Implementación de las clases GrupoInvestigacionDAO y GrupoInvestigacionMySQL. (4 puntos)
G. Implementación de la funcionalidad de registro de Grupos de investigación a nivel de interfaz. (2 puntos)
H. Implementación de la funcionalidad de búsqueda, listado y selección de Grupos de investigación a nivel de interfaz. (2 puntos)

Profesor del curso: **Dr. Freddy Paz**

San Miguel, 18 de octubre del 2021

ANOTACIONES DE PROGRAMACIÓN

(En el caso de C#) Para obtener el año de un datetimepicker utilizar:

```
this._grupoInvestigacion.AnhoFundacion = dtpAnho.Value.Year.ToString();
```

Para convertir de String (yyyy) a Datetime utilizar:

```
dtpAnho.Value = DateTime.ParseExact(_grupoInvestigacion.AnhoFundacion, "yyyy", null);
```

(En el caso de JAVA) Para obtener el año de un JYearChooser utilizar:

```
grupoInvestigacion.setAnhoFundacion(String.valueOf(dtpAnho.getValue()));
```

Para convertir de String (yyyy) a int utilizar:

```
dtpAnho.setValue(Integer.parseInt(grupoInvestigacion.getAnhoFundacion()));
```

(En el caso de JAVA) Para enviar un arreglo de bytes a un CallableStatement puede utilizar:

```
cs.setBytes("_foto", grupoInvestigacion.getFoto());
```

```
cs.setBytes("_archivo_adicional_PDF", grupoInvestigacion.getArchivoAdicionalPDF());
```

En caso desee recibir bytes:

```
grupo.setFoto(rs.getBytes("foto"));
```

```
grupo.setArchivoAdicionalPDF(rs.getBytes("archivo_adicional_PDF"));
```

(En el caso de C#) Para enviar un arreglo de bytes a un Command puede utilizar:

```
comando.Parameters.AddWithValue("_foto", grupoInvestigacion.Foto);
```

```
comando.Parameters.AddWithValue("_archivo_adicional_PDF", grupoInvestigacion.ArchivoAdicionalPDF);
```

En caso desee recibir bytes:

```
if (!lector.IsDBNull(lector.GetOrdinal("foto"))) grupo.Foto = (byte[])lector["foto"];
```

```
if (!lector.IsDBNull(lector.GetOrdinal("archivo_adicional_PDF"))) grupo.Foto =  
(byte[])lector["archivo_adicional_PDF"];
```