

# Estructuras de Datos

## Proyecto Primera Evaluación – 2023 PAO 2 Sistema para Gestión de Contactos

### LEA ESTE DOCUMENTO DETENIDAMENTE

Este es un proyecto **grupal**. Por tanto, todos los miembros del grupo deben involucrarse en el diseño e implementación de la solución que se solicita. La entrega del proyecto incluirá un reporte de auto y coevaluación, donde cada miembro del grupo calificará sus contribuciones al proyecto y las de sus compañeros.

**Es su responsabilidad agruparse a tiempo para la entrega del proyecto.** Usted debe indicar su grupo en el Aula Virtual hasta el martes 24 de octubre. Los estudiantes que no se hayan unido a ningún grupo para entonces, serán asignados aleatoriamente por el profesor. A partir de entonces, no se aceptarán cambios en la conformación de los grupos. **EN NINGUNA CIRCUNSTANCIA SE ACEPTARÁN ENTREGAS INDIVIDUALES.**

Si algún grupo experimenta el incumplimiento sistemático de alguno de sus miembros, esta situación debe ser reportada al profesor **TAN PRONTO COMO SE PRESENTE**. Cualquier estudiante cuyo incumplimiento sea comprobado, será retirado del grupo y **DEBERÁ IMPLEMENTAR EL PROYECTO DE MANERA INDIVIDUAL SOBRE EL 50% DEL PUNTAJE POSIBLE.**

Si algún estudiante siente que otro (u otros) toma(n) el control del grupo y no permite(n) a los demás miembros hacer aportes o colaborar en el proyecto, debe reportar esta situación al profesor **TAN PRONTO COMO SE PRESENTE**. Cualquier estudiante que pretenda monopolizar el trabajo del proyecto o impida a los demás hacer aportes será separado del grupo y **DEBERÁ IMPLEMENTAR EL PROYECTO DE MANERA INDIVIDUAL Y SOBRE EL 50% DEL PUNTAJE POSIBLE.**

Si algún estudiante reporta de manera extemporánea cualquiera de las dos situaciones detalladas en los dos párrafos precedentes, será penalizado con su 25% de la nota total obtenida en el proyecto. Reporte novedades al profesor **TAN PRONTO COMO ÉSTAS SE PRESENTEN.**

Considere las políticas de buena conducta académica que se explicaron en la primera clase del curso respecto al plagio y demás violaciones del código de Ética de la ESPOL. Los autores del proyecto deben ser usted y sus compañeros de grupo (no otras personas, **ni motores de Inteligencia Artificial**). Cualquier indicio de lo contrario, será reportado a las unidades correspondientes para el tratamiento pertinente.

Finalmente,  **siga las instrucciones de este documento** para evitar inconvenientes. Si tiene dudas, consulte al profesor en lugar de asumir cosas que pueden ser incorrectas.

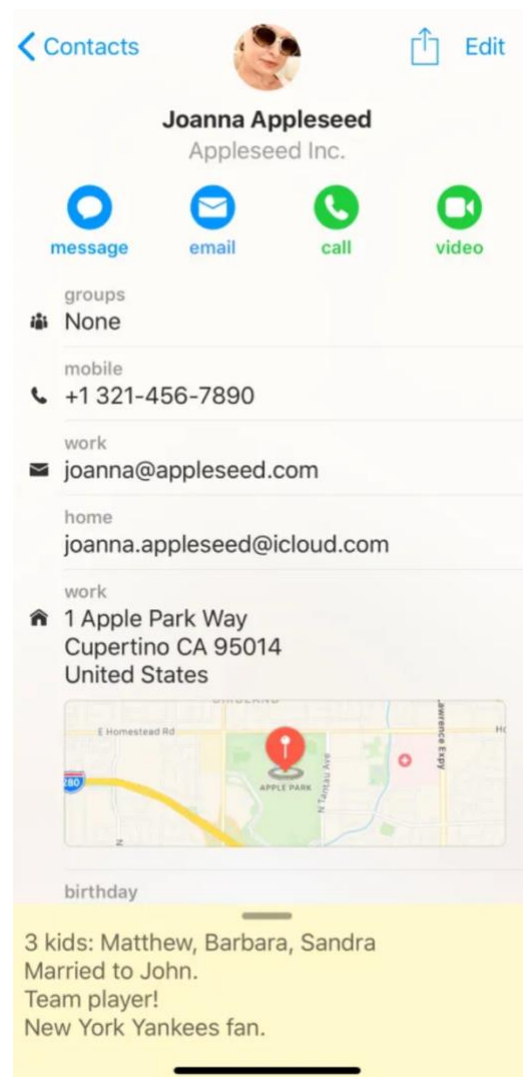
El incumplimiento de instrucciones explícitas indicadas en este documento o indicadas en nuestras sesiones teóricas derivará, de manera inapelable, en la penalización y rebaja de puntos sobre su entrega.

## Introducción

Una aplicación de gestión de contactos suele incluir atributos básicos para cada contacto: nombre, número de teléfono, foto, y dirección. No obstante, actualmente, los perfiles de contactos se han vuelto más complejos y suelen incluir múltiples atributos adicionales, como los listados a continuación, algunos de los cuales se muestran en la figura:

- Dirección de domicilio o trabajo, descriptiva y/o con ubicación geográfica (link de Google Maps, por ejemplo).
- Varios emails: de uso personal, de trabajo, etc.
- Una lista de números telefónicos,
- Identificadores de distintas redes sociales.
- Una colección de fotos.
- Varias fechas de Interés: cumpleaños, aniversarios, renovación de suscripción o de contrato, pagos pendientes, etc.
- Contactos relacionados:
  - Por ejemplo, el contacto de una empresa podría tener asociado el contacto de su director o de alguien que trabaja en un área en particular (ventas, coordinación, etc).
  - El contacto de nuestro médico podría tener asociado también el de su asistente (para agendar citas).
  - El contacto de nuestro mejor amigo podría tener asociado el de su pareja y los de algunos familiares.

Algunas de estas características están incluidas en aplicaciones de gestión de contactos existentes. Les sugerimos investigar cómo estas funcionalidades se presentan típicamente para inspirar su propio diseño.



En este proyecto, usted implementará y utilizará estructuras de datos (específicamente colecciones lineales) para implementar una aplicación de gestión de contactos con características similares a las detalladas arriba. Así, usted y sus compañeros de grupo implementarán una interfaz gráfica de JavaFX que permitirá visualizar un listado de contactos (personas y empresas). Para esto deberán, además, diseñar e implementar clases para representar a los contactos y sus atributos, y a las demás entidades involucradas en este problema.

## Requerimientos Mínimos

Su proyecto debe consistir en una interfaz gráfica que permita a los usuarios visualizar la lista de contactos, crearlos, editarlos y eliminarlos. El diseño de su interfaz gráfica puede basarse en alguna aplicación existente (que debe ser mencionada en su reporte), pero también se espera que usted y su equipo introduzcan elementos de diseño de su autoría.

Considere además que cada contacto tiene varios atributos, y cada uno de estos atributos tiene un nombre o descripción (p. ej., “teléfono del trabajo”) y un valor asociado (p. ej., “0912345678”). Considere que el listado de atributos no siempre es el mismo (por ejemplo, una empresa no tiene apellido). Sin embargo, la aplicación debe ser consistente, por lo que debería existir un “listado maestro” de atributos que son comunes a todos los contactos.

Como funcionalidad mínima, su aplicación debe facilitar a los usuarios las siguientes operaciones:

- Ver y navegar la lista de contactos de la aplicación. La navegación debe poder realizarse en dos sentidos (hacia adelante y hacia atrás) y de manera “circular” (es decir, al llegar al final de la lista de contactos y avanzar hacia adelante, la aplicación debe mostrar el primer contacto. Asimismo, al avanzar hacia atrás, estando en el primer contacto, la aplicación debe mostrar el último.
- Ver y navegar las fotos asociadas a un contacto también de manera circular y en dos sentidos. Todo contacto deberá tener más de una foto en su lista.
- Crear, editar y remover contactos. Al crear un contacto nuevo se deberían tener al menos dos “plantillas” de contactos típicos: persona o empresa. Dichas plantillas deben tener atributos razonables para cada escenario. Note que la creación de un contacto incluye, además, indicar sus fotos asociadas. Su equipo debe decidir si quieren incluir más de dos plantillas básicas.
- Crear, editar y remover los atributos de un contacto existente. Por ejemplo, añadir una dirección a un contacto, o editar el usuario de Instagram de un contacto, o remover una foto asociada a un contacto.
- Ordenar la lista de contactos por, al menos, tres de los siguientes criterios:
  - Apellido y primer nombre
  - Cantidad de atributos contenidos
  - Fecha de cumpleaños más cercana (solo aplicable para los contactos que tengan dicha información).
  - Empresa
  - País de residencia
  - Tipo de contacto (persona natural o empresa)
  - Existencia de un atributo en particular (por ejemplo contactos con usuario de Instagram)
  - Otros que usted y su equipo considere pertinente
- Filtrar la lista de contactos por, al menos, tres de los atributos listados anteriormente. Por ejemplo, el usuario debería poder ver todos los contactos que residen en Quito o todos aquellos nacidos en febrero.
- Ver los contactos asociados de un contacto existente y visualizar los detalles estos. Es decir, al dar clic a un contacto asociado, la aplicación debería mostrar la información del contacto cliqueado.

## Sobre Funcionalidad Opcional

Las funcionalidades descritas arriba son obligatorias y constituyen los requerimientos mínimos para que su proyecto sea aceptado y admitido a ser calificado sobre **hasta el 80%** de la nota máxima. Este proyecto, sin embargo, es también una oportunidad para ser creativo. Así, usted y sus compañeros de grupo pueden implementar funcionalidad extra que contribuya a una mejor versión de la aplicación solicitada.

A continuación, se listan algunas funcionalidades opcionales que usted y sus compañeros de grupo podrían implementar para aplicar a puntos extra:

**Búsqueda Avanzada:** Permitir la búsqueda de contactos basada en múltiples criterios a la vez, como apellido y ubicación (p. ej., *Listar todos los contactos de apellido Miranda nacidos en abril, o todas las empresas con dirección en Guayaquil*).

**Etiquetado de Contactos:** Ofrecer la opción de etiquetar contactos con categorías personalizadas (amigos, trabajo, familia, etc.), y luego filtrar o buscar basado en esas etiquetas.

**Exportar/Importar Contactos:** Permitir la exportación e importación de contactos a/desde formato vCard.

**Favoritos y Acceso Rápido:** Permitir marcar contactos como favoritos para un acceso rápido.

**Grupos de Contactos:** Ofrecer la posibilidad de crear y gestionar grupos de contactos (por ejemplo, el grupo “familia”, o el grupo “amigos cercanos”).

**Inicio de sesión:** Manejar distintos usuarios, cada uno de los cuales tiene un directorio de contactos propios. Al iniciar sesión, la aplicación le permitiría al usuario correspondiente explorar únicamente su lista de contactos asociados. También podría existir un contacto administrador, que pueda ver y modificar la lista de contactos de todos los contactos registrados en el sistema.

Adicionalmente, usted y su grupo quedan en libertad de implementar cualquier funcionalidad que consideren relevante y útil para este proyecto, siempre que dicha funcionalidad haga un uso pertinente y apropiado de las estructuras de datos revisadas en el curso durante el primer parcial.

En general, si su implementación es excepcional, su proyecto podría superar el 100% de la nota.

## Modelamiento del Problema y Consideraciones Técnicas

Como habrá notado, gran parte de la funcionalidad solicitada en este proyecto puede ser lograda con listas: los contactos manejados en la aplicación pueden ser almacenados en una lista circular doblemente enlazada (al igual que las fotos asociadas a cada contacto). Esto permitiría al usuario iterar sobre los contactos hacia adelante hacia atrás. Su aplicación debe consistir en una interfaz gráfica de JavaFX que permita al usuario realizar las operaciones indicadas arriba y todas las demás que sean pertinentes para lograr los requerimientos mínimos.

Dados los objetivos que se persiguen en este curso, usted y sus compañeros de grupo serán responsables de implementar las estructuras que consideren pertinentes para la solución de este problema. **El uso de las clases ArrayList y LinkedList de Java está explícitamente prohibido. Violar esta prohibición invalidaría su proyecto y le otorgaría una nota automática de cero (0).** Las demás estructuras de datos del Java Collection Framework (pilas, colas, conjuntos, mapas) pueden ser utilizadas sin problemas, así como las clases que permiten la comparación (Comparable, Comparator) e iteración de elementos (Iterator, ListIterator).

Toda la información que deba ser “recordada” por su aplicación (por ejemplo, los contactos existentes, las fotos, los registrados en el sistema, etc.) se debe almacenar en archivos. De esta forma, los datos de su aplicación serán persistentes y estarán disponibles cada vez que ésta se ejecute, **sin necesidad de que alguien tenga que ingresar toda la información desde cero**. Usted es responsable de decidir la estrategia mediante la cual logrará la persistencia de los datos. Es decir, queda a discreción suya y de sus compañeros de grupo utilizar archivos de texto plano o serializar la información en archivos binarios. **No utilice almacenamiento que requiera motores de bases de datos.**

## Entregables

**Entrega parcial (40%): lunes 13 de noviembre 8:00pm**

Su proyecto de NetBeans (enviado como un archivo .zip) debe incluir, por lo menos:

- Implementación básica de la interfaz gráfica de usuario (GUI) en JavaFX.
- Diseño e implementación de las clases para representar a los contactos y sus atributos.
- Funcionalidad para visualizar la lista de contactos.
- Funcionalidad para crear nuevos contactos.
- Si se desea, en esta entrega, es permitido el uso de las clases ArrayList y LinkedList de Java.

## Entrega final (60%): **miércoles 29 de noviembre 11:59pm**

Esta entrega corresponde al proyecto de NetBeans que implementa correctamente la interfaz gráfica final de su proyecto, con –al menos– las funcionalidades mínimas detalladas anteriormente.

Su entrega deberá incluir además un archivo **.docx**. La plantilla específica de este documento se adjunta a este documento y deberá incluir:

- *Screenshots* de su interfaz explicando su funcionalidad,
- Una imagen de su clase principal, y
- Tabla de auto- y co-evaluación de los miembros del grupo

Su proyecto de Netbeans y el archivo **.docx** deben ser entregados a través del Aula Virtual en un único archivo comprimido **.zip**.

**No se aceptarán entregas atrasadas, sin reporte, o que no implementen los requerimientos mínimos.**

Recuerde que este **es un proyecto grupal**. Esto significa que usted debe trabajar **con** sus compañeros de grupo para sacar el proyecto adelante. Su grupo deberá reunirse periódicamente para coordinar y discutir acciones relacionadas al proyecto. Dividir funcionalidad y repartir responsabilidades podría no ser la mejor estrategia, sobre todo en las fases tempranas del proyecto.

## Presentaciones

Usted y sus compañeros de grupo deberán presentar su proyecto (el código subido al Aula Virtual) al profesor en una reunión presencial que tendrá lugar el **jueves 30 de noviembre de 11:00 a 13:00** en un laboratorio que será anunciado con anticipación. Esta reunión tendrá una duración de **15 MINUTOS** por grupo y se efectuará únicamente entre el profesor y **TODOS** los miembros del grupo. Allí, las observaciones sobre su entrega serán dadas de manera verbal.

**Solo los grupos que hayan entregado un proyecto funcional deberán presentarse a esta reunión.**

La reunión iniciará con uno de los miembros del grupo mostrando su proyecto en funcionamiento. Esta demostración debe evidenciar que el grupo ha cumplido con los requerimientos mínimos del proyecto y debe mencionar explícitamente toda funcionalidad extra que hayan implementado. **Su grupo debe definir con anticipación qué miembro del grupo estará a cargo de esta presentación inicial.**

La presentación inicial debe enfocarse en lo que su grupo ha implementado. **NO** debe ser sobre los requerimientos del proyecto --- el profesor tiene claro lo que se ha solicitado. Asimismo, el profesor conoce los nombres de los integrantes del grupo, por tanto, no es necesario presentar a sus compañeros. **Enfóquese en mostrar su proyecto en ejecución y en comunicar al profesor lo que su grupo ha logrado.**

Durante la reunión, cada integrante del grupo deberá responder, al menos, una pregunta sobre el proyecto. Para esto, todos los miembros del grupo deberán tener el código subido al Aula Virtual abierto en las

computadoras del laboratorio donde se realizará la presentación. Sus respuestas serán calificadas y la calificación individual recibida será uno de los dos factores multiplicativos que contribuirán a su nota en el proyecto.

## Instrucciones Finales

Recuerde: **Este es un proyecto grupal.**

Finalmente,  **siga las instrucciones de este documento**  para evitar inconvenientes. Si tiene dudas, consulte al profesor (en lugar de asumir cosas que pueden ser incorrectas). Los ayudantes de la materia podrían no ser la fuente de información más confiable cuando se trata de responder preguntas sobre el proyecto.

Sea oportuno(a) en cuanto a sus preguntas: **No se atenderán consultas sobre el proyecto después del 16 de noviembre de 2023.** Tampoco se responderán preguntas que consulten sobre instrucciones indicadas explícitamente en este documento. Antes de hacer preguntas, **LEA ESTE DOCUMENTO DETENIDAMENTE.**

¡Muchos éxitos a todos!