



## PROYECTO PRÁCTICO NO. 2

---

### Personal URL

#### Descripción

A través de un sistema automatizado se quieren controlar los datos de todo el personal perteneciente a la Universidad Rafael Landívar, ya sea estudiante o trabajador. De todo el personal se conoce el nombre, primer y segundo apellidos y el DPI. Los estudiantes, en particular, pueden ser de pregrado o de postgrado. En cada caso se conoce el carnet, la facultad en la que estudia, año en que ingresó y la nota de cada curso que recibe. Para los estudiantes de postgrado se conoce el tipo de estudio que realiza (Maestría o Doctorado).

Por otro lado, del personal trabajador se conoce el salario, fecha de inicio (año, mes y día) y el departamento en el que labora. Entre los trabajadores existen docentes y no docentes. De los primeros se conoce las asignaturas que imparte, mientras que de los no docentes se conoce el cargo ocupacional.

#### Enunciado

Crear una solución utilizando C++, Windows forms. Diseñe las clases que considere necesarias para modelar los conceptos descritos en el problema.

Suponiendo que en el sistema se implementará una lista de estudiantes de pregrado, una de estudiantes de postgrado, una de Trabajadores no docentes y otra de trabajadores docentes. Implemente los siguientes requisitos funcionales:

- Gestionar estudiantes de pregrado y posgrado.
- Gestionar trabajadores docentes y no docentes.
- Devolver el nombre completo de una persona dado el DPI
- Mostrar un listado con los estudiantes de Doctorado, Pregrado o Postgrado y exportar la información a un archivo CSV ordenado por el primer apellido y debe guardarse con el siguiente formato  
<carnet>,<apellidos>,<nombre>,<año de ingreso>,<DPI>,<facultad>,<curso y nota 1>,<curso y nota 2>,...<curso y nota n>



- Mostrar un listado con los docentes y exportar la información a un archivo CSV ordenado por el primer apellido y debe guardarse con el siguiente formato <código de empleado>,<apellidos>,<nombre>,<DPI>,<fecha de inicio>,<salario>,<curso 1>,<curso 2>,...<curso n>
- Mostrar un listado con los datos del personal trabajador no docente y exportar la información a un archivo CSV ordenado por el primer apellido y debe guardarse con el siguiente formato <código de empleado>,<apellidos>,<nombre>,<DPI>,<fecha de inicio>,<cargo ocupacional>,<salario>
- Calcular el salario promedio de todos los trabajadores (docentes, no docentes o ambos).
- Determinar la cantidad de estudiantes en Maestría o por facultad.
- Dado un curso determinar la cantidad de docentes que lo imparten y un listado con los estudiantes que lo reciben ordenados alfabéticamente por el nombre o por nota
- Dado un estudiante de pregrado o postgrado conocer su promedio
- Dado un estudiante de pregrado conocer su nota más alta y el curso al que corresponde dicha nota
- Dada una facultad, conocer el alumno de pregrado con mejor promedio

El programa deberá mostrar un menú por cada uno de los requisitos funcionales mencionados anteriormente y manejar errores y excepciones.

Para la implementación del programa utilice Listas doblemente enlazada. Los métodos de ordenamiento y de búsqueda quedan a su consideración.

**Nota:** gestionar incluye mostrar, insertar, modificar y eliminar

El programa debe poder leer y agregar listados de estudiantes a partir de un archivo CSV, de la siguiente manera:

- Para estudiantes el formato será: <apellidos>,<nombre>,<año de ingreso>,<DPI>,<facultad>,<pregrado, postgrado o doctorado>,<curso y nota 1>,<curso y nota 2>,...<curso y nota n>
  - El carnet debe generarse con 5 dígitos aleatorios concatenado con los dos últimos dígitos del año en que ingresó el estudiante (<valor aleatorio><dos últimos dígitos del año en que inició el estudiante>) Ej: 2491221.



- Para trabajadores el formato será: <apellidos>, <nombre>, <DPI>, <fecha de inicio>, <salario>, <true si es docente, false si no lo es>, [<cargo ocupacional> | <curso 1>, <curso 2>, ... <curso n>]
  - El código de empleado se calcula de la misma manera que el carnet de estudiante
  - Si no es un trabajador docente, no debe impartir cursos, pero se sabe su cargo ocupacional
  - Si es un trabajador docente no se espera un cargo ocupacional sino los cursos que imparte

## Entregables

- Archivo comprimido o repositorio en GitHub con:
  - Carpeta de Solución del Proyecto (código fuente con documentación interna)
  - Programa ejecutable (no incluir si es repositorio de GitHub)
  - Documentación Externa
- La documentación externa contiene:
  1. Carátula
  2. Introducción
  3. Análisis
    - a. Entradas
    - b. Salidas
    - c. Procesos
    - d. Restricciones
  4. Diseño
    - a. Diagrama de Flujo
  5. Conclusiones
  6. Recomendaciones
  7. Referencias
    - a. Librerías utilizadas y su utilización
  8. Anexos
    - a. Manual de Usuario



### Aspectos para evaluar

- Validación de errores.
- Adecuada aplicación de los conocimientos.
- Calidad de la documentación: ortografía, orden, limpieza y que esté completa.
- Calidad de la solución propuesta: que solucione el problema (que haga lo que requiere el sistema de reserva) en forma eficaz.
- Funcionalidad del programa: debe cumplir a cabalidad con todos los requerimientos.
- Evidencia de la creación del programa y dominio de los conceptos utilizados.
- Creatividad.

### Ponderación

Documentación	10
Análisis	15
Diseño	15
Codificación	60
Total	100

### Consideraciones

- Se trabajará en parejas.
- Deberá realizarse la **creación de estructuras propias, así como la utilización de punteros.**
- Toda solución presentada debe compilar correctamente para poder tener derecho a revisión.
- La utilización de código de terceros debe estar completamente documentado, referenciado y justificada su utilización, debe demostrarse el dominio completo de lo implementado.
- **Se podrá demandar que en la calificación presencial del proyecto se realicen cambios de funcionalidad.**