

PONTIFICIA UNIVERSIDAD CATÓLICA DE
VALPARAÍSO

FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA INFORMÁTICA

**INFORME PROYECTO SISTEMA DE INFORMACION
SIA FINAL**

GABRIEL FUENTES

ALEXIS ESCOBAR

ARIEL PONCE

PROGRAMACION AVANZADA(INF2236-2)
INGENERIA EN INFORMÁTICA

SEPTIEMBRE 2025



SIA1.1 - El presente proyecto aborda la gestión de una Bolsa de Trabajo, enfocándose en la optimización del proceso de vinculación entre postulantes y puestos laborales. El objetivo principal es establecer un mecanismo que permita asociar de manera automática a los postulantes con las vacantes disponibles, considerando principalmente el nivel de correspondencia entre sus competencias (habilidades de cada persona) y los requisitos solicitados por cada puesto.

Para ello, se parte de un registro estructurado de información, que incluye:

- **Datos de las personas(postulantes)**, como nombre, identificación y el conjunto de competencias que posee cada uno, acompañadas de un nivel de dominio.
- **Datos de las competencias**, que representan las habilidades o conocimientos relevantes para los distintos ámbitos laborales.
- **Datos de los puestos de trabajo**, en los que se definen requisitos mínimos de competencias (y niveles asociados) que determinan la idoneidad de un candidato.

El análisis de estos datos permite al sistema realizar una comparación objetiva entre las características de las personas (postulantes) y las exigencias de cada puesto, con el fin de filtrar y sugerir de manera automática a los candidatos más adecuados.

Entre las funcionalidades principales a implementar, se destacan:

- **Representación y almacenamiento de postulantes**, competencias y puestos de trabajo en estructuras adecuadas.
- **Módulo de autoselección de candidatos**, que evalúa el cumplimiento de los requisitos establecidos en cada puesto y determina qué postulantes cumplen o superan dichos criterios.
- **Posibilidad de registrar manualmente nuevos elementos**, ya sean postulantes, competencias o vacantes, para mantener la información actualizada.
- Generación y visualización de listados en consola que permitan al usuario conocer tanto los candidatos seleccionados como los registros existentes en el sistema.

SIA1.2 - Se realizó un diseño de clases del dominio representando las entidades principales del sistema (Puesto, Persona, Competencia, Gestión De Puestos).

Diagrama conceptual:

Código fuente en los archivos:

- Competencia.java
- Persona.java
- Puesto.java
- GestionPuesto.java

Se puede observar cada clase en un archivo por separado dentro del package “proyectopoo”.

SIA1.3 – Todos los atributos de todas las clases fueron definidos como privados y se implementaron métodos de lectura (getters) y escritura (setters).

En cada archivo de cada clase, se pueden observar:

- Competencia: desde la línea 6 a 7 y 21 hasta la línea 41
- Persona: desde la línea 9 a 13 y 24 hasta la línea 82
- Puesto: desde la línea 6 a 11 y 22 hasta la línea 86
- Empleado: desde la línea 7 a 9 y 19 hasta la línea 54
- Administrador: desde la línea 7 a 8 y 17 hasta la línea 49

SIA1.4 – Los datos iniciales se encuentran en el archivo “ProyectoPOO.java”, desde la línea 33 hasta la línea 103.

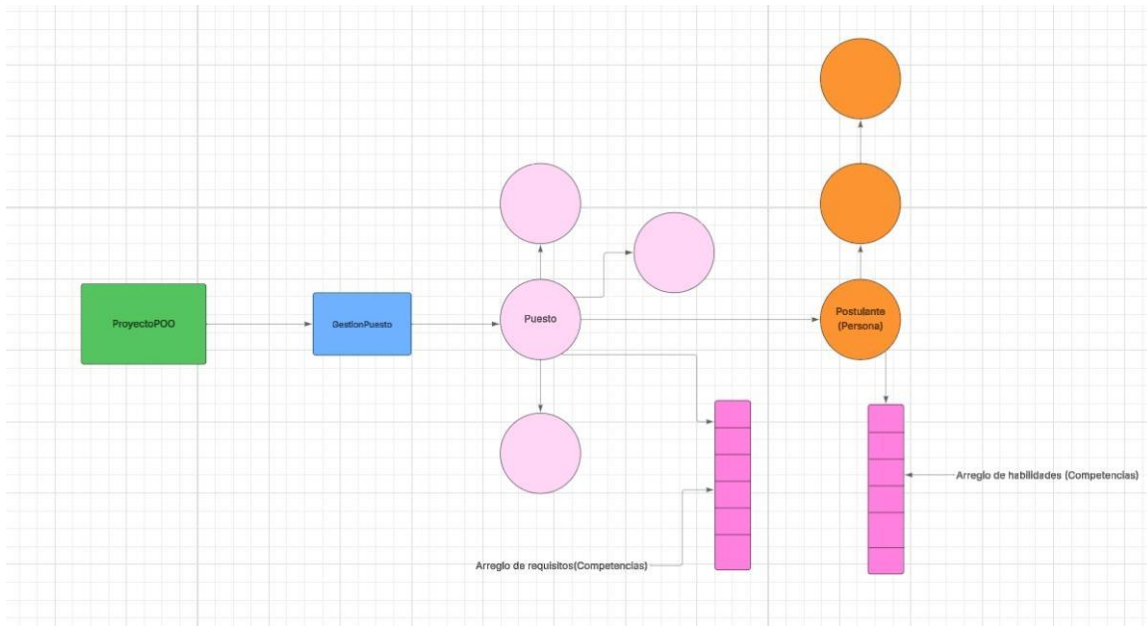
SIA1.5 – Dos colecciones de objetos, con la 2ª anidada

Primera colección: “HashMap <String, Puesto> puestos” en GestionPuesto.java , la cual es iniciada en la línea 12 de ProyectoPOO.java con el nombre de “sistema”.

Segunda colección anidada:

- En cada Puesto, existe un arreglo “LinkedList<Persona> postulantes” en la línea 10 de Puesto.java.
- Además, un “ArrayList<Competencia> requisitos” asociados al puesto , línea 11 del archivo Puesto.java.

Representación gráfica de la anidación:



SIA1.6 – Sobrecarga de métodos

Se implementó una sobrecarga de métodos en la clase “Competencia”, específicamente en las líneas 64 a 74 se evidencian dos métodos “cumpleCon” sobrecargados y en la clase “Persona” en la línea 108 y 153 con los dos métodos “agregarHabilidad”.

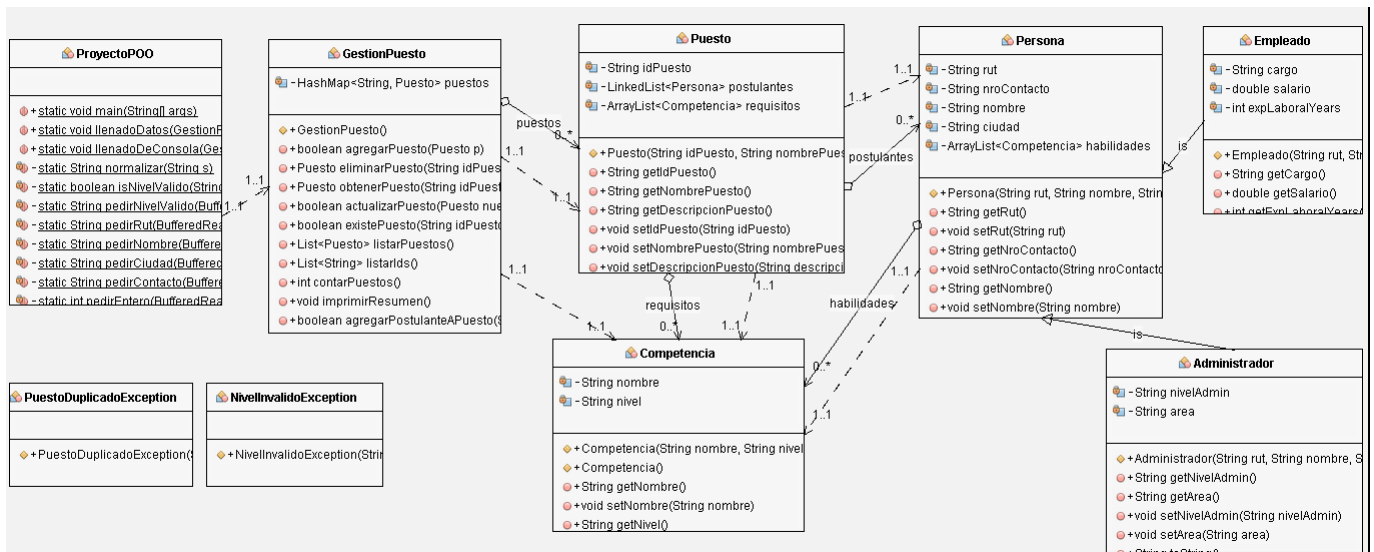
SIA1.7 – En GestionPuesto.java se usa un Map (HashMap) para almacenar los puestos por id (String), se evidencia en la línea 8 de la clase GestionPuesto.

SIA1.8 – En el archivo “ProyectoPOO”, desde la línea 114 hasta la línea 350 se puede observar todas las opciones que ofrece el menú y las instrucciones del código.

SIA1.9 – Se usa “System.out.println” en varias partes por ejemplo en la línea 104 lo cual cumple el requisito de mostrar datos sin ventanas.

SIA1.10- <https://github.com/GaboCrvzy/Bolsa-De-Trabajo-Java>

SIA2.1 – Diagrama de clases UML.



SIA2.2 – Desde la línea 250 hasta la línea 310 del archivo “GestionPuesto” se puede observar la persistencia de datos utilizando un archivo CSV, ademas, desde la línea 313 hasta la línea 440 se observa el sistema batch.

SIA2.3 –

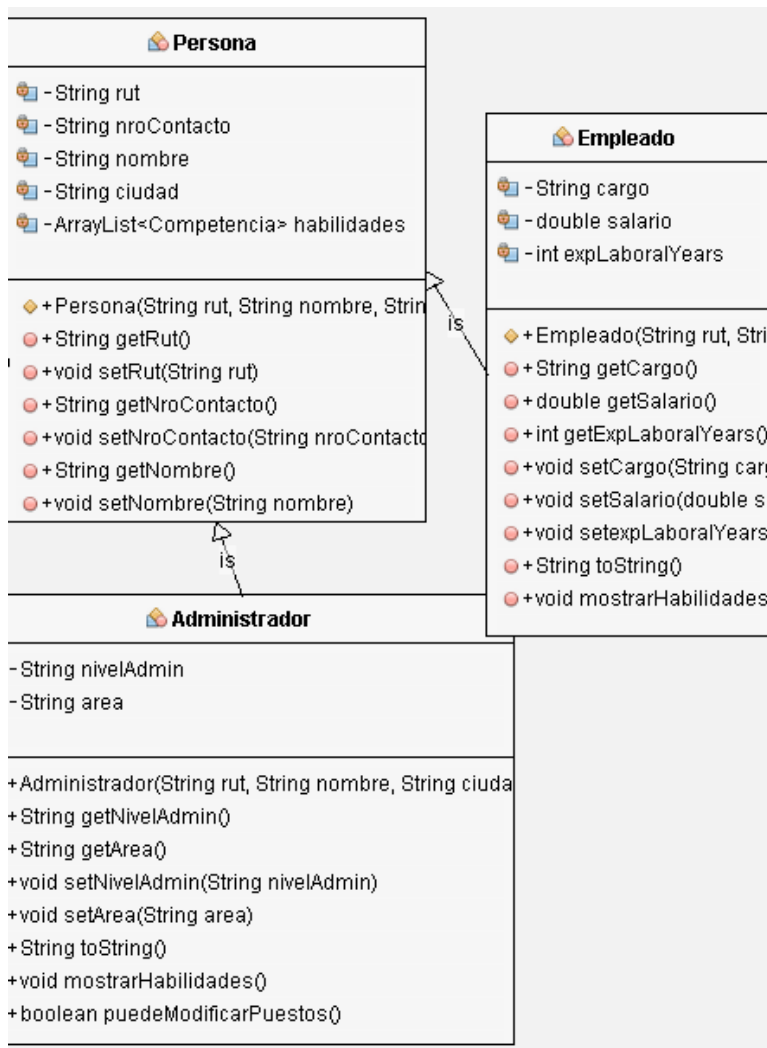
SIA2.4– En el archivo “ProyectoPOO”, desde la línea 226 hasta la línea 317 se puede observar las opciones de editar y eliminar para la segunda colección anidada (Persona/Postulante).

SIA2.5– Desde la línea 169 hasta la línea 211 del archivo “GestionPuesto” se observa la funcionalidad propia de ranking y selección de puntaje.

SIA2.6– El código cumple con la modularización y sigue con las buenas prácticas de documentación y legibilidad.

SIA2.7– Se evidencian 2 clases que usan sobreescritura de métodos. La clase “Empleado” la cual tiene la sobreescritura de método de la clase padre “Persona”, se evidencia en la línea 65 a 72 del archivo “Empleado.java” y la clase “Administrador” se evidencia en la línea 60 a 66 del archivo “Adminastrador.java”, el método sobrescrito es “mostrarHabilidades”.

El diseño es:



SIA2.8– Desde la línea 250 hasta la línea 310 del archivo “GestionPuesto.java” se observa el manejo de excepciones usando try-catch al igual que en el archivo “ProyectoPOO.java” se observa en la línea 130.

SIA2.9– Desde la línea 14 hasta la línea 33 del archivo “GestionPuesto” se observa la primera clase que se extiende de una Excepción utilizando la clase “PuestoDuplicadoException”, la siguiente está en el archivo “Competencia.java” desde la línea 30 hasta la línea 40, ahí se observa que se extiende de una Excepción utilizando la clase “NivelInvalidoException”.

SIA2.10–Desde la línea 418 hasta la línea 440 en el archivo “GestionPuesto” se observa el metodo que genera un reporte en archivo txt.

SIA2.11– <https://github.com/GaboCrvzy/Bolsa-De-Trabajo-Java>

SIA2.12–En el archivo “GestionPuesto” desde la línea 14 hasta la línea 33 se observa el metodo de agregar, desde la línea 35 hasta la línea 45 se observa el metodo de eliminar, desde la línea 57 hasta la línea 70 se observa el metodo para actualizar.

SIA2.13—en el archivo “GestionPuesto”, desde la línea 47 hasta la línea 55 hay un método de búsqueda de primer nivel, en el archivo “Puesto.java”, desde la línea 117 hasta la línea 125 se encuentra un método de búsqueda de segundo nivel, en el archivo “Persona.java”, desde la línea 159 hasta la línea 169 se observa un método de búsqueda de tercer nivel