

# Trabajo Final Programación III

Sistema de Búsquedas laborales

EQUIPO 9:

Garcia, Ezequiel

Gonzalez, Franco

Vicente, Juan Martín

# Introducción



## OBJETIVO

El objetivo del trabajo práctico grupal es la aplicación de los conceptos teórico-prácticos adquiridos en la cátedra, para la realización de un sistema para la gestión de búsquedas laborales. Aplicar estos nuevos conocimientos relacionados a la programación orientada a objetos (clases y sus relaciones, herencia, polimorfismo), patrones de diseño, excepciones, serialización, concurrencia, interfaces gráficas y desarrollo de proyectos en el lenguaje de programación “Java”.

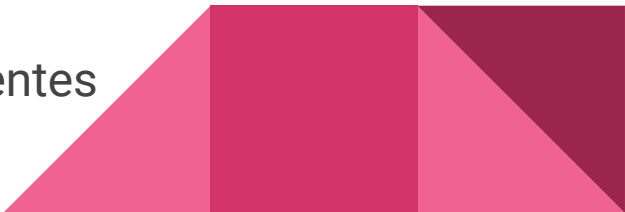
## ALCANCE

El sistema realizado tiene como alcance las “Consideraciones Generales sobre el Trabajo Práctico Grupal” otorgado por la cátedra y las consideraciones conversadas en las clases correspondientes.

# Modelo del Negocio

- Sistema
- Funcionalidades del Empleado Pretenso
- Funcionalidades del Empleador
- Funcionalidades del Administrador del sistema

## Capa de Datos

- Usuarios: Administrador, Empleadores y Empleados Pretensos
  - Tickets: de Empleados y de Empleadores
  - Clases donde se guardan los resultados de las diferentes Rondas
- 

# Controlador

- Controlador Base: Transición a los distintos controladores
- Controlador main
- Controladores registro usuarios: Administrador, Empleadores y Empleados
- Controladores funciones usuarios
- Controlador simulación Threads



# Vista


- Se utilizó una ventana distinta para cada especificación, tanto para los registros, las funcionalidades y la simulación de Threads como una vista “main” principal donde se da la opción de login
- Cada ventana responde a un único controlador (MVC)
- En cada vista se permite el retroceso para una mejor interacción usuario/sistema



# Serialización

- Se crearon las clases serializables que almacenen todos los datos que se busca persistir (usuarios, tickets, rondas).
- Una clase que permite el traspaso del modelo a estas clases serializables y viceversa.

# Concurrencia

- El objetivo fue desarrollar una simulación donde cada hilo se encuentre lo menos posible dentro de los métodos “synchronized”
  - Se realizó un paquete aparte para cumplir con lo establecido en el Trabajo Práctico, con una ventana separada.
- 

# Patrones Utilizados

- Singleton

Utilizado en la clase Sistema, Clase principal del programa realizado.

- Double Dispatch

Para generar puntuación de la locación en la comparación de Tickets.

- Factory

En la generación de Tickets.



# Patrones Utilizados

- MVC

Para la generación de las interfaces gráficas del sistema.

- Observer Observable

En la muestra de la simulación de la concurrencia.

- DAO DTO

Serialización y persistencia de los datos del sistema.





# Conclusiones

- Durante el trayecto del proyecto se implementaron los conocimientos sobre POO y patrones de diseño adquiridos durante el presente cuatrimestre.
- Se pudo trabajar con entornos para gestión de proyectos de Software como Eclipse, JDeveloper y GitHub, esta última herramienta muy utilizada en la industria en la actualidad.
- Si bien consideramos que el sistema generado cumple con las expectativas del proyecto, entendemos que es nuestro primer programa de esta magnitud, por lo que sabemos que todavía hay detalles a mejorar.



Muchas gracias

