# Taller de Programación

# Informe del Modelo de Diseño

### Grupo 6

Integrantes

 Lucas Helguera
 CI: 4.785.828-4

 Bruno Romero
 CI: 5.382.171-4

 Baltasar Seijas
 CI: 5.478.011-5

 Flavio Cubón
 CI: 5.132.468-9

 Gaston Medeiros
 CI: 5.274.708-4

Docente: Leonel Peña

# Índice

Índice	2
1 Introducción	3
1.1 Propósito	3
1.2 Alcance	3
1.3 Estructura del Documento	3
2 Organización Lógica	3
3 Realización de Casos de Uso	4
3.1 Colaboración Servidor Central	4
3.2 Estructura	5

### 1 Introducción

## 1.1 Propósito

El propósito de este documento es brindar una descripción general del Modelo de Diseño.

### 1.2 Alcance

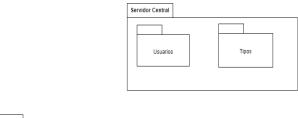
El informe del Modelo de Diseño presenta una abstracción de la solución lógica al problema. Incluye las colaboraciones que realizan cada uno de los casos de uso del Modelo de Casos de Uso.

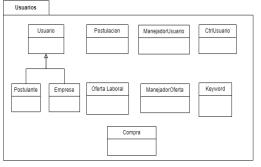
### 1.3 Estructura del Documento

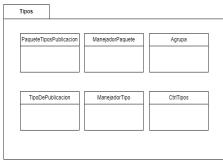
El documento está dividido en tres secciones. La segunda sección presenta la organización lógica del sistema en paquetes de diseño. La tercera sección presenta las colaboraciones con la realización de los casos de uso incluidos en el Modelo de Casos de Uso. Por último, la cuarta sección presenta los criterios generales adoptados para el diseño de las colaboraciones.

# 2 Organización Lógica

A los efectos de organizar la capa lógica se definen paquetes de diseño. Cada paquete de diseño puede contener clases de diseño y otros paquetes de diseño. En esta sección se incluyen los paquetes de diseño existentes y las clases incluidas en cada uno de ellos, sin incluir las relaciones entre éstas.



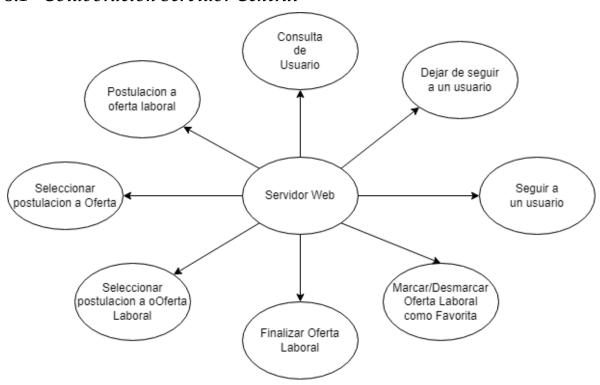


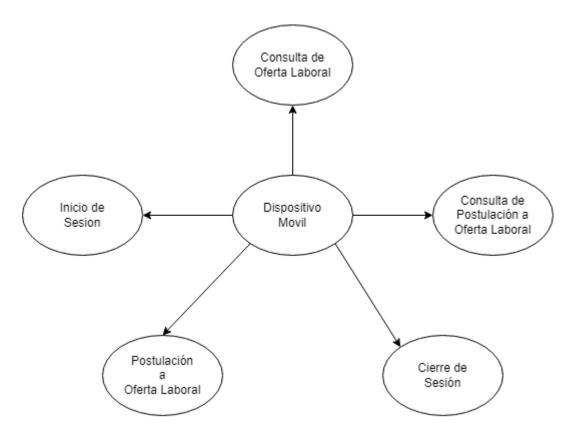


## 3 Realización de Casos de Uso

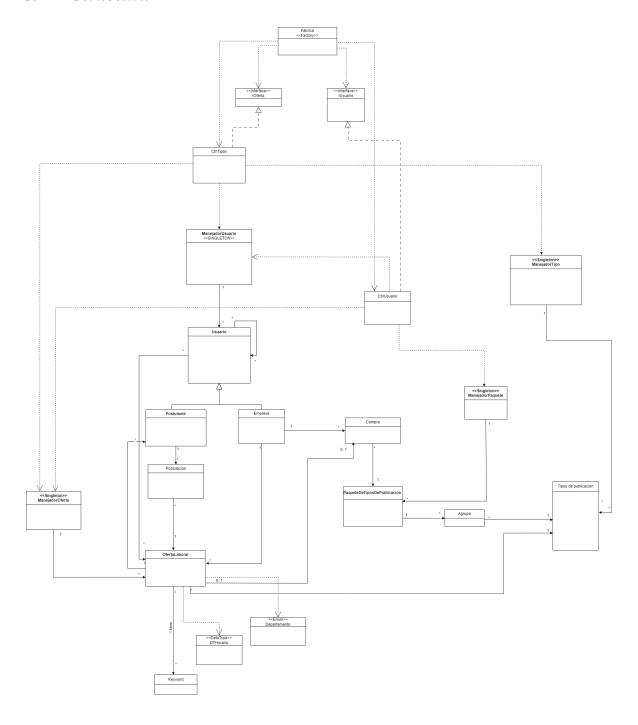
Las realizaciones, están expresadas en términos de estructura e interacciones entre instancias que respeten dicha estructura. Concretamente, la parte estructural de una realización es un *diagrama* de clases conteniendo clases del modelo; la parte dinámica es un conjunto de *diagramas* de interacción que ilustran el flujo de mensajes entre instancias de las clases de la parte estructural correspondiente.

## 3.1 Colaboración Servidor Central





# 3.2 Estructura



< <singleton>&gt; ManejadorPaquete</singleton>	< <singleton>&gt; ManejadorTipo</singleton>	< <singleton>&gt; ManejadorOferta</singleton>	ManejadorUsuario < <singleton>&gt;</singleton>
- instance : ManejadorPaquete	- instance : ManejadorTipo	- instance : ManejadorOferta	- instance : ManejadorUsuario
<u>+ getInstance()</u> : ManejadorPaquete	+ getInstance() : ManejadorTipo	+ getInstance(): ManejadorOferta	+ getInstance(); ManejadorUsuario
+agregarPaquete(p:PaqueteDeTipoDePublicacion)	+obtenerTipo(nomTipo:string):TipoPubl	+obtenerOferta(nomOferta:string):Ofert	+obtenerUsuario(nickName:string): Usuario
+existePaquete(nombrePaquete:string): bool	+agregar(tipo: TipoPublicacion)	+AgregarOferta(O: Oferta)	+ ObtenerEmpresas(): Set(string)
+obtenerPaquete(nombrePaquete:string): PaqueteDeTiposDePublicacion		+existeOferta(nombre:string) : bool	+ getAllNicks() : Set <string></string>
Fabrica			+ getDataUsuario(nombre):DataUsuario
< <factory>&gt;</factory>			+ UnicidadNombreCorreo(nickName:string,correo:string):bool
+getIUsuario(): IUsuario +getIOferta(): IOferta			+ CrearPostulante(nickName, nombre, apellido, correo, fechaNac, nacionalidad):bool
garoiataly, ionata			+ crearEmpresa(nickName, nombre, apellido, correo, descripcion, link):bool

\*Interface\*
Interface\*
Interfac

+agregarTipoPublicacion(nombrePaquete, nombreTipo, cant)				
	CtrlUsuario			
	+ingresarDatosPostulacion(nomPostulante,CVRed,motivacion,nomEmpresa,nomOferta:string,fechaPos:Date): bool			
	finalizarOferta(nomOferta)			
	elegirOrden(nomPostulante, nomOferta, Orden)			
	follow( seguidor, seguir)			
	unfollow(seguidor, seguir)			
	esSeguidor(seguidor, seguir): bool			
	aceptarRechazarOferta(nomOferta, cambio)			
	ingresarOferta(nickname, nombreTipo, nombre, descripcion, horario, remuneracion, ciudad, depto, fecha, keyword) : bool			
	+seleccionarOfertaLaboral(nombre : string): DTOfertaLaboral			
	+seleccionarEmpresa(nickname: string): Set(string)			
	+getDataOferta(nomOferta:string):DTOfertaLaboral			
	+obtenerOfertasDeEmpresa(nickname:string) : Set(string)			
	+listarPostulantes():set(string)			
	+editarDatosBasicosUsuario(d: DataUsuario)			
	+IngresarDatosPostulacion(nomPostulante,CVRed,motivacion,nomEmpresa,nomOferta:string,fechaPos:Date, video:String):bool			
	+mostrarDatosUsuario(nickname):DataUsuario			
	+listarUsuarios():set(string)			

ringresarPostulante(nickName, nombre, apellido, correoElectronico, fechaNacimiento, nacionalidad):bool

listarUsuarios() : Set <string></string>
ingresarPostulante(nickName, nombre, apellido, correoElectronico, fechaNacimiento, nacionalidad):bool
ingresarEmpresa(nickName, nombre, apellido, correoElectronico, descripcion, link):bool
listarUsuarios():set(string)
mostrarDatosUsuario(nickname):DataUsuario
In gresar Datos Postulacion (nom Postulante, CVRed, motivacion, nom Empresa, nom Oferta: string, fecha Pos: Date): booling the properties of the propertie
editarDatosBasicosUsuario(d: DataUsuario)
listarOfertasIngresadas(nickEmpresa):Set(String)
esSeguidor(seguidor, seguir): bool
iniciarSesion(nickname, clave)
aceptarRechazarOferta(nomOferta, cambio)
finalizarOferta(nomOferta)
elegirOrden(nomPostulante, nomOferta, Orden)
follow( seguidor, seguir)
unfollow(seguidor, seguir)
esSeguidor(seguidor, seguir): bool
finalizarOferta(nomOferta)
elegirOrden(nomPostulante, nomOferta, Orden)
follow( seguidor, seguir)
unfollow(seguidor, seguir)

Illustatio
IngresarDatosPostulacion(nomPostulante, CVRed, motivacion, nomEmpresa, nomOferta string, fechaPos: Date): booi

CtrtTipos

\*ingresarDatosTipoPublicacion(d: DataTipoPublicacion) : bool

\*ingresarDatosPaquete(nombre, descripcion string vailidez.int, descuento double, fechaAlta:Date):bool

\*obtenerDatosPaquete(nombrePaquete: string): DTPaqueteTiposPublicacion

\*obtenerDatosTipoPublicacion(nombreTipoPublicacion: string): DTTipoPublicacion

\*obtenerDatosTipoPublicacion(nombreTipoPublicacion: string): DTTipoPublicacion

\*listarEmpresas() : Set(string)

\*seleccionarCertat.abora(inombre : string): DTOfertat.aborat

\*ingresarCerta(nickname, nombreTipo, nombre, descripcion, horario, remuneracion, ciudad, depto, fecha, keyword): bool

\*listarPaquetes():Set(string)

\*listarPaquetes():Set(string)

\*agregarTipoPublicacion():Set(string)

\*agregarTipoPublicacion(nombrePaquete, nombreTipo, cant)

\*

-listarUsuarios() : Set<String>

#### Usuario

-nickname : string -mail: string

-nombre : string -apellido : string

setDatosBasicos(DataUsuario)

#### PaqueteDeTiposDePublicacion

nombre : string

descripcion: string

descuento: double

fecha: Date /costo: double

periodoValidez: int

#### Tipos de publicacion

nombre: string

descripcion: string

exposicion: int

duracion: int

costo: double

fecha: Date

#### Postulante

-fechaNacimiento : Date

-nacionalidad : string

+postularAOferta(OfertaLaboral)

#### Empresa

-nombre : string

-descripcion: string

-link: string

#### Postulacion

-fechaPostulacion: Date

-resumenCV :string

-descripcion : string

-video : string

#### Compra

fechaCompra: Date

/vencimiento : Date

# Keyword

-nombre : string

Agrupa

cantidad: int

# OfertaLaboral

-nombre : string

-ciudad: string

-departamento: Departamento

-horario: DTHorario

-remuneracion: int

-fecha: Date

/costo: double

+agregarPostulante(pos:Postulante)

+existePostulante(nomPostulante):bool

#### <<Enum> Departamento

Montevideo

Artigas

Canelones

CerroLargo

Colonia

Durazno

Flores

Florida

Lavalleja

Maldonado

Paysandú

Rivera

Rocha RíoNegro

Salto SanJosé

Soriano

Tacuarembó

TreintaYTres

<<DataType>> DTHorario

inicio: LocalTime

fin: LocalTime