

GRASP

ALTA EMPRESA

```
altaEmpresa(cuit:int, descripcion:char, nombre:char, razonSocial:char, contraseña:char, correo:char, nombreUsuario:char) {  
    EmpresaBD.persistir(Empresa.crear(cuit, descripcion, nombre, razonSocial, contraseña, correo, nombreUsuario))  
}
```

```
Empresa:: crear(cuit:int, descripcion:char, nombre:char, razonSocial:char, contraseña:char, correo:char, nombreUsuario:char){  
    e = self.new()  
    e.setCuit(cuit)  
    e.setDescripcion(descripcion)  
    e.setNombre(nombre)  
    e.setRazonSocial(razonSocial)  
    e.setContraseña(contraseña)  
    e.setCorreo(correo)  
    e.setNombreUsuario(nombreUsuario)  
    e.setValidado(FALSE)  
    return e  
}
```

EMPRESA::

```
validar(){  
    self.setValidado(TRUE)  
}
```

ALTA POSTULACION

```
altaPostulacion(fecha:date, postulante: Postulante, ofertaLaboral: OfertaLaboral){  
    if (ofertaLaboral.abierta())  
        GestorCorreo.mandarCorreo(cliente.getCorreo(), ofertaLaboral.getTitulo())  
    PostulacionBD.persistir(Postulacion.crear(fecha, postulante, ofertaLaboral))  
}
```

```
al))  
}
```

```
Postulacion:: crear(fecha:date, postulante: Postulante, ofertaLaboral: Oferta  
Laboral) {  
    p = self.new()  
    p.setFecha(fecha)  
    p.setPostulante(postulante)  
    p.setOferta(ofertaLaboral)  
    p.setEstado("Iniciada")  
    return p  
}
```

```
OfertaLaboral:: abierta() {  
    return self.estado == "Abierta"  
}
```

```
GestorCorreo(correo: string, oferta: string) {  
    Correol.enviarCorreo('Hola' + correo + 'Tu postulacion fue realizada con ex  
ito para la siguiente oferta' + oferta)  
}
```