# Documentación Técnica – Plataforma de Empleo

## 1. Resumen ejecutivo

La plataforma conecta directamente a empleadores y trabajadores (labours) a través de la publicación de empleos (jobs) y un flujo estructurado de postulación (job\_application) con su historial de proceso (application\_step). El modelo de datos es modular, auditable y escalable, incorporando un catálogo flexible de roles para soportar futuras ampliaciones sin afectar la compatibilidad.

## 2. Objetivos del sistema

- Permitir el registro y autenticación de usuarios.  
- Diferenciar perfiles de empleador y trabajador.  
- Habilitar que los empleadores publiquen ofertas de trabajo.  
- Facilitar que los trabajadores apliquen a dichas ofertas.  
- Gestionar el proceso de aplicación con estados formales.  
- Proveer trazabilidad mediante un historial auditable.  
- Preparar el sistema para nuevos roles (ej. administrador, moderador).

## 3. Modelo de datos

### 3.1 Gestión de roles

• user\_role: Catálogo de roles (ej. employer, labour, admin).

• app\_user\_role: Relación N:M que asigna uno o varios roles a cada usuario.

### 3.2 Identidad y perfiles

• app\_user: Identidad base de la cuenta (login).

• employer\_profile: Perfil 1:1 con app\_user para publicar empleos.

• labour\_profile: Perfil 1:1 con app\_user para aplicar a empleos.

### 3.3 Publicación de empleos

• job\_status: Enum que define el ciclo de vida de un job (draft, open, paused, closed).

• job\_modality: Enum que define modalidad del trabajo (full\_time, part\_time, contract, etc.).

• job: Oferta publicada con ubicación, modalidad, salarios, cupos y fechas.

### 3.4 Postulación y proceso

• application\_status: Enum que define estados de la aplicación (submitted, under\_review, etc.).

• job\_application: Registro de la postulación de un labour a un job.

• application\_step: Historial de cambios de estado en la postulación.

## 4. Relaciones y cardinalidades

- Un usuario puede tener múltiples roles (user\_role ↔ app\_user\_role ↔ app\_user).  
- app\_user se asocia 1:1 con employer\_profile o labour\_profile.  
- employer\_profile publica múltiples jobs.  
- labour\_profile aplica a múltiples jobs (job\_application).  
- Cada job\_application genera múltiples application\_steps.

## 5. Flujos de negocio

1. Registro y creación de perfil (employer o labour).

2. Employer publica un job con modalidad, salario y condiciones.

3. Labour aplica al job, generando un job\_application.

4. Employer gestiona la postulación, cambiando estados.

5. Cada cambio queda registrado en application\_step.

6. El proceso termina en estado final: hired, rejected o withdrawn.

## 6. Reglas de negocio y validaciones

- Un labour solo puede aplicar una vez a cada job (unicidad job\_id + labour\_id).  
- Transiciones de estados controladas por workflow.  
- Los cupos (openings) no deben superar lo definido.  
- Fechas coherentes: published\_at ≤ closing\_date.  
- Salarios coherentes: salary\_min ≤ salary\_max.

## 7. Índices y rendimiento

- Índices en job por employer\_id, status, published\_at y closing\_date.  
- Índices en job\_application por status y applied\_at.  
- Índice en application\_step por application\_id y changed\_at.