

# Quether Challenge

Cèlia Sánchez Morato

Blai Bofill Butinyà

Héctor García Sorbera



UAB THE HACK

Escola d'enyinyeria UAB

<b>Talent Gap Analyzer: De setmanes a hores. De percepcions a dades.....</b>	<b>2</b>
<b>Informe Explicativo del Proyecto.....</b>	<b>2</b>
1. Resumen Ejecutivo.....	2
2. Arquitectura de Datos y Captura de Talento.....	3
2.1. Arquitectura de Datos Separada.....	3
2.2. Formulario de Captura de Talento (Barra Lateral).....	3
3.1. Página de inicio.....	5
3.2. Implementación de la API (IA) en la Vista Estratégica.....	6
4. Análisis Táctico de Roles (Vista "Mánagers").....	7
Implementación de la API (IA) en la Vista Táctica.....	8
5. Dashboard de Talento Actual (Vista "RRHH").....	9
<b>Qué Hace Cada Archivo.....</b>	<b>12</b>
1. Archivos de Código (.py).....	12
2. Archivos de Datos (en la carpeta data/).....	12
<b>Guía de Ejecución y Arquitectura de Código.....</b>	<b>14</b>
Para ejecutar los códigos:.....	14
<b>Posibles Futuras Implementaciones.....</b>	<b>15</b>
Extracción y Análisis Automático de CVs.....	15
Modelo de Clustering para Descubrimiento de Roles.....	15
PCA (Principal Component Analysis) para Optimización de Habilidades.....	15
Modelo Multi-Classifier para Predicción de Roles.....	16
<b>Conclusiones del Proyecto.....</b>	<b>17</b>

# Talent Gap Analyzer: De setmanes a horas. De percepcions a dades.

## Informe Explicativo del Proyecto

### 1. Resumen Ejecutivo

El Talent Gap Analyzer es una web que, con la ayuda de la inteligencia artificial, ayuda a convertir procesos de análisis de la brecha de talento manuales a un diagnóstico interactivo instantáneo.

Usando como lenguaje principal el Python y varias librerías entre las que se encuentran Streamlit, Pandas y Lang Chain, Talent Gap Analyzer transforma una base de datos de diversos inputs en un output proporciona información muy útil para la empresa.



Página de inicio de la web Talent Gap Analyzer

Las respuestas dadas proporcionan información en diferentes ámbitos:

1. **Vista Estratégica (General):** Para la alta dirección. Diagnóstica donde hay más talento y habilidad dentro de la empresa, y los puntos dónde es más débil y en que deben incrementar para lograr los cambios deseados.
2. **Vista Táctica (Managers):** Para los directores de departamento. Permite encontrar al mejor candidato, tanto interno como externo, para un rol en el futuro.
3. **Vista de RRHH (Talento Actual):** Proporciona como están distribuidos los talentos de la plantilla.

La aplicación, con los datos dados por los inputs, pondera a los candidatos (internos o externos) y, aparte, usa API de IA Generativa (Groq) para crear textos ejecutivos sobre la situación del candidato en base a los objetivos buscados.

## 2. Arquitectura de Datos y Captura de Talento

Talent Gap Analyzer posee una barra lateral, en la que podemos poner información de nuevos candidatos tanto internos como externos en la empresa.

The image displays two wireframe mockups of the 'Captura de Talento' (Talent Capture) sidebar. Both mockups show a 'Formulario de Nuevo Talento' (New Talent Form) with the following fields:

- Tipo de Talento:** Radio buttons for 'Interno (Empleado Actual)' (Internal Employee) and 'Externo (Candidato / CV)' (External Candidate / CV). The 'Interno' option is selected in the left version and 'Externo' is selected in the right version.
- Selección de Habilidades:** A dropdown menu labeled 'Choose options'.
- Datos Básicos:** Fields for 'Nombre Completo' (Full Name) and 'Email o Contacto' (Email or Contact).
- Datos del Empleado:** Fields for 'Chapter', 'Strategy', 'Rol Actual' (with 'Junior' selected), 'Manager' (with 'N/A' selected), and a note to 'Define el Nivel de cada Habilidad (1-10)' (Define the level of each skill (1-10)).
- Adjuntar CV (Opcional):** A section with a 'Drag and drop file here' input field and a 'Browse files' button. The right version includes this section.
- Buttons:** 'Añadir Talento' (Add Talent) at the bottom of both forms.

Slide bar con un formulario para nuevos talentos

### 2.1. Arquitectura de Datos Separada

Para mantener la integridad de los datos, la aplicación utiliza dos archivos CSV distintos:

- [data/talento\\_actual.csv](#): Almacena la plantilla de empleados internos.
- [data/candidatos\\_externos.csv](#): Almacena el *pool* de talento de candidatos externos.

El módulo `modules/data_loader.py` se encarga de leer ambos archivos, junto con los archivos de configuración JSON ([org\\_config.json](#) y [vision\\_futura.json](#)), y unirlos en memoria (`pd.concat`) en un único DataFrame. Esto permite que el resto de la aplicación analice ambos *pools* de talento de forma unificada.

### 2.2. Formulario de Captura de Talento (Barra Lateral)

La barra lateral (`st.sidebar`) contiene el "Formulario dinámico RRHA" para añadir nuevos perfiles:

- **Selector de Tipo:** Un `st.radio` (fuera del formulario) permite al usuario elegir entre "Interno (Empleado Actual)" o "Externo (Candidato / CV)". Este selector actualiza la interfaz del formulario en tiempo real.
- **Campos Dinámicos:**
  1. Si se selecciona "Interno", el formulario muestra campos para "Chapter" y "Manager".
  2. Si se selecciona "Externo", estos campos se ocultan y se reemplazan por un `st.file_uploader` para "Adjuntar CV".
- **Asignación de Habilidades (1-10):** El formulario implementa un sistema de dos pasos:
  1. El usuario selecciona una o más habilidades (cargado con las previas de `org_config.json`).
  2. Por cada habilidad seleccionada, aparece dinámicamente una barra deslizante, permitiendo al usuario poner un nivel de competencia preciso del 1 al 10, replicando la estructura de datos de `talento_actual.csv`.
- **Lógica de Guardado:** Al enviar, el formulario escribe los datos en el CSV correspondiente (interno o externo) y fuerza un refresco de la caché (`st.cache_data.clear()`) para que los nuevos datos estén disponibles al instante y queden guardados en la empresa.

Selecciona Habilidades

Gestión de Proy... ×
Analítica Digital (... ×
...

Define el Nivel de cada Habilidad (1-10)

Nivel para: Gestión de Proyectos (S-PM)

1

7
10

Nivel para: Analítica Digital (S-ANALYTICS)

4

4

Nivel para: Automatización (n8n) (S-AUTOM)

8

8

Añadir Talento

Selecciónador i rankeador de habilidades para nuevos talentos

### 3. Dashboard Estratégico (Vista "General" / Ejecutiva)

Esta es la página de inicio, diseñada para ofrecer un diagnóstico de alto nivel.

#### 3.1. Página de inicio

- **Métricas de Alto Nivel:** Tres KPIs (`st.metric`) que resumen la salud de la empresa:
  - **Talento en Riesgo (Fuga):** Cuenta a los empleados con `"retention_risk": "Media"`.
  - **Empleados Sobre-asignados:** Cuenta a los empleados cuya suma de `dedicación_actual` supera el 95%.
  - **Vacíos Críticos (Contratación):** Identifica cuántos roles futuros no tienen ningún candidato interno con un score > 50%.

Dashboard Estratégico (Vista General)		
Talento en Riesgo (Fuga)	Empleados Sobre-asignados	Vacíos Críticos (Contratación)
5 Empleados	10 Empleados	1 Roles

KPIs Salud Empresa

Ver Detalles del Diagnóstico					
Talento en Riesgo		Capacidad Excedida (>95%)		Vacíos Críticos (Roles <50%)	
	nombre	rol_actual	chapter		
2	Pablo Reyes	Martech Architect	Martech	0	Jordi Casals
4	Marc Serra	Data Analyst	Martech	1	Marta Soler
5	Adrián López	Creative Director	Creative	2	Pablo Reyes
7	Sergio Martín	Performance Media Specialist	Performance	3	Núria Aguilera
9	Claudia Riera	Social Media Strategist	Influency	4	Marc Serra
				5	Adrián López
				6	Nerea Vidal
				7	Sergio Martín
				8	Laura Torres
				9	Claudia Riera

Pestaña con el Diagnóstico detallado: 5 talentos en Riesgo, Talentos que exceden en dedicación y Roles con un vacío.

- **Gráfico 1: Cobertura de Roles Futuros (Plotly):**

- Este gráfico de barras horizontales muestra el score del **mejor candidato interno** para cada rol futuro.
- Las barras están coloreadas (Rojo/Verde) para señalizar un "Vacío Crítico" (<50%) o un rol "Cubierto" (>50%), permitiendo a la dirección enfocar sus esfuerzos de contratación.

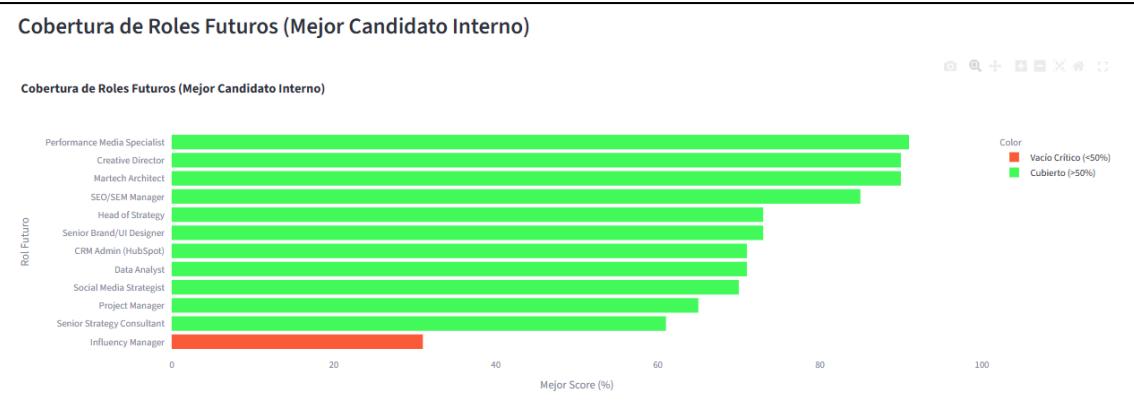


Gráfico 1. Cobertura de Roles Futuros: Score del mejor candidato interno para cada rol futuro.

- **Gráfico 2: Carga de Trabajo por Proyecto (Plotly):**

- Este gráfico de barras agrega la columna `dedicación_actual` de todos los empleados para mostrar la carga total que consume cada proyecto (ej. "Royal", "Arquimbau", "I+D").

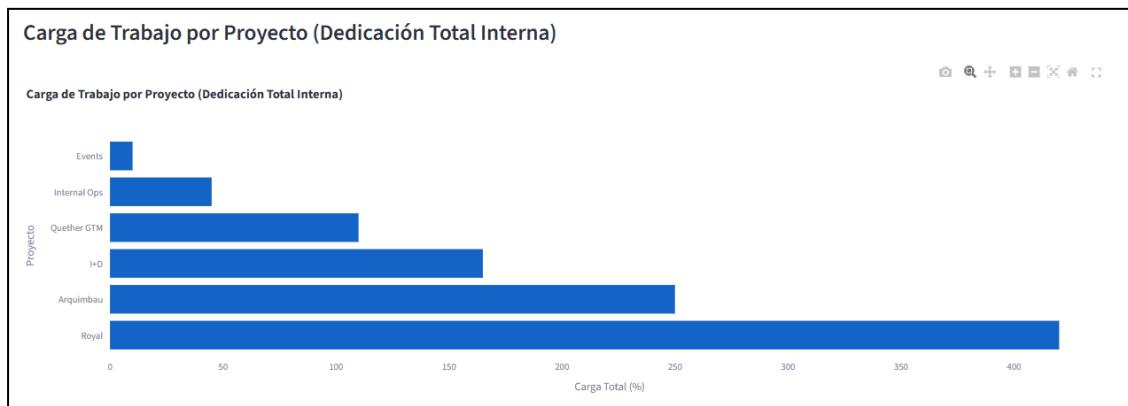


Gráfico 2. Carga de Trabajo por Proyecto: carga total que consume cada proyecto.

### 3.2. Implementación de la API (IA) en la Vista Estratégica

- **Funcionalidad:** Botón "Generar Resumen Ejecutivo (IA)", que cumple el requisito de "Narrativas ejecutivas".
- **Cómo Funciona:**
  1. **Recopilación de Datos:** El código recopila las listas de empleados en riesgo, empleados sobre-asignados y los roles en vacío crítico.
  2. **Llamada a la API:** Se puede llamar a la función de `generar_resumen_ejecutivo` (en `modules/recommendations.py`).
  3. **Construcción del Prompt:** Esta función construye un prompt que instruye a la IA (Groq API vía LangChain) para que actúe como un "Consultor de Estrategia de Talento".

4. **Respuesta:** La API da una respuesta sobre los problemas y posibles estrategias para obtener una solución para estos.

Generar Resumen Ejecutivo (IA)

Resumen Ejecutivo (Generado por IA)

Narrativa Ejecutiva

Quether enfrenta tres desafíos críticos que afectan su capacidad para alcanzar sus objetivos y mantener un crecimiento sostenible. En primer lugar, la empresa tiene un alto riesgo de fuga de talento, lo que significa que varios de sus mejores empleados están considerando abandonar la empresa. Esto incluye a importantes miembros del equipo como Pablo Reyes, Marc Serra, Adrián López, Sergio Martín y Claudia Riera. Además, la empresa también enfrenta una sobrecarga de trabajo en varios departamentos, con empleados como Jordi Casals, Marta Soler, Pablo Reyes, Núria Aguilera, Marc Serra, Adrián López, Nerea Vidal, Sergio Martín, Laura Torres y Claudia Riera que están sobredimensionados. Por último, la empresa tiene vacíos críticos en roles futuros, específicamente en la posición de Influency Manager, lo que podría afectar su capacidad para innovar y mantenerse competitiva.

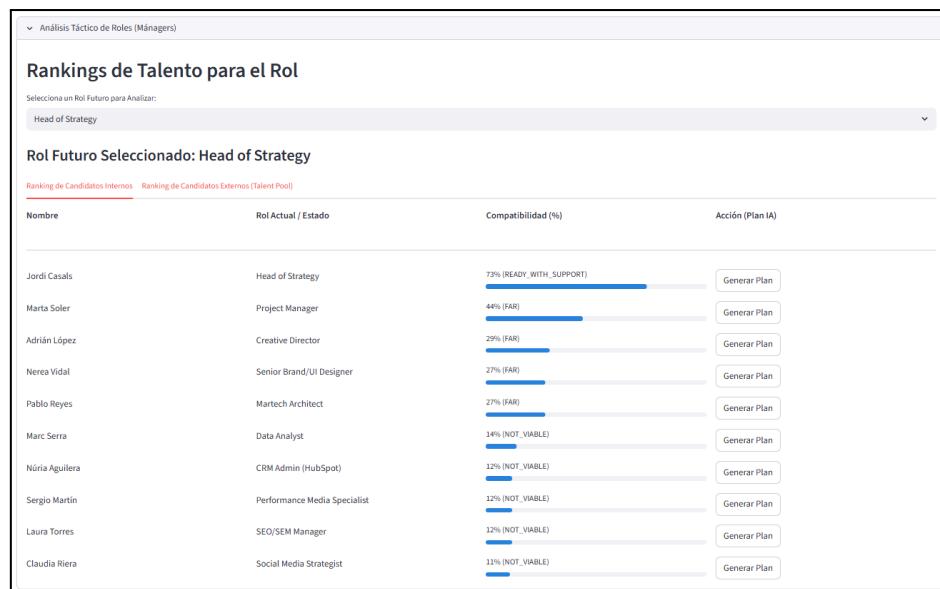
Para abordar estos desafíos, recomendamos implementar una estrategia de desarrollo de talento integral que se centre en el fomento de la retención de empleados, la optimización del uso de recursos humanos y la identificación y desarrollo de talento para cubrir vacíos críticos. Esto podría incluir la implementación de programas de capacitación y desarrollo de carrera, la creación de un entorno de trabajo motivador y atractivo, y la identificación y desarrollo de candidatos internos para cubrir vacíos críticos. Al abordar estos desafíos de manera integral, Quether podrá mantener a sus mejores empleados, optimizar su uso de recursos humanos y mantener su competitividad en el mercado.

Ejemplo de Respuesta con IA integrada

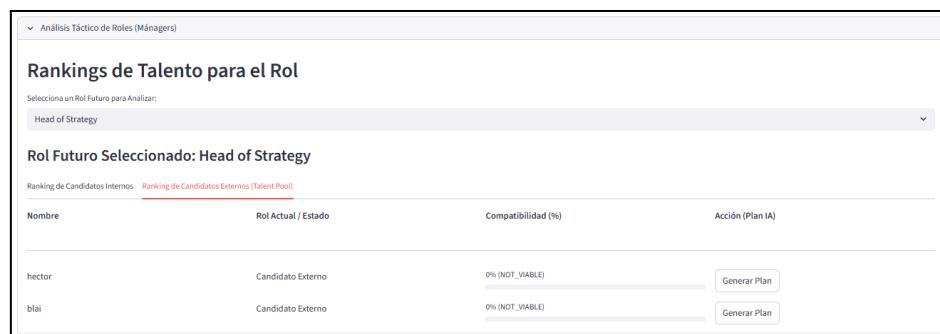
## 4. Análisis Táctico de Roles (Vista "Mánagers")

Esta sección (`st.expander`) está diseñada para máñagers que buscan cubrir una vacante específica.

- **Selector de Rol:** Un desplegable (`st.selectbox`) permite al máñager elegir un rol de `vision_futura.json`.
- **Interfaz de Pestañas (`st.tabs`):** La app presenta dos rankings limpios y separados:
  - **Ranking de Candidatos Internos**
  - **Ranking de Candidatos Externos (Talent Pool)**
- **Algoritmo de Compatibilidad (El "Core" del Reto):**
  - Cada persona en los rankings es evaluada por la función `calcular_compatibilidad_total` usando el algoritmo de ponderación oficial:
    - **50% Habilidades:** Cobertura de *skills* y su nivel.
    - **25% Responsabilidades:** Similitud de experiencia (usando Scikit-learn).
    - **15% Ambiciones:** Alineación de aspiración profesional (`ambiciones`).
    - **10% Dedicación:** Disponibilidad (`dedicación_actual`).
- **Clasificación de Banda (Nivel 1):**
  - El *score* porcentual se muestra en una barra de progreso (`st.progress`) y se traduce a las 5 bandas oficiales requeridas por el reto: **READY (>85%)**, **READY\_WITH\_SUPPORT** (70-85%), **NEAR** (50-70%), **FAR** (25-50%), y **NOT\_VIABLE (<25%)**.



Ranking de Talento de candidatos internos para el Rol seleccionado



Ranking de Talento de candidatos externos para el Rol seleccionado

## Implementación de la API (IA) en la Vista Táctica

- Funcionalidad:** Botón "Generar Plan" al lado de cada candidato, cumpliendo el requisito de "Planes de desarrollo personalizados".
- Cómo Funciona:**
  - Recopilación de Datos:** El código recopila los datos del candidato específico y el rol seleccionado.
  - Llamada a la API:** Se llama a la función `generar_plan_desarrollo`.
  - Construcción del Prompt:** Esta función construye un *prompt* táctico, instruyendo a la IA para que actúe como un "coach de talento" y analice la brecha de habilidades de esa persona.
  - Respuesta:** La API devuelve un plan de acción personalizado, que se muestra en un `st.expander` debajo del botón.

Ver Plan de Desarrollo para Marta Soler

**Narrativa de Talento**

Marta Soler es una Project Manager experimentada con una ambición senior y un deseo claro de avanzar en su carrera para convertirse en Head of Strategy. Aunque tiene una base sólida en la gestión de proyectos, hay áreas clave donde necesita mejorar para alcanzar su objetivo. Estamos comprometidos con su crecimiento y desarrollo, y estamos aquí para apoyarla en su camino hacia la estrategia líder.

**Plan de Desarrollo Personalizado**

Para cerrar la brecha de habilidades clave y alcanzar su objetivo, hemos identificado las siguientes tres acciones concretas y cuantificables:

- 1. Mejorar habilidades en OKRs y Roadmapping**
  - **Objetivo:** Aumentar el nivel de habilidades en OKRs y Roadmapping a 8/10 dentro de los próximos 6 meses.
  - **Acción:** Asistir a un curso de formación en OKRs y Roadmapping de 2 días, con un costo de \$1,500.
  - **Medición del éxito:** Realizar un ejercicio de OKRs y Roadmapping en un proyecto futuro, con un seguimiento y evaluación del desempeño en un plazo de 3 meses.
- 2. Desarrollar habilidades en Análisis de Negocio**
  - **Objetivo:** Aumentar el nivel de habilidades en Análisis de Negocio a 8/10 dentro de los próximos 9 meses.
  - **Acción:** Participar en un programa de mentoría con un experto en Análisis de Negocio durante 3 meses, con un costo de \$3,000.
  - **Medición del éxito:** Realizar un análisis de negocio para un proyecto futuro, con un seguimiento y evaluación del desempeño en un plazo de 6 meses.
- 3. Mejorar habilidades en Gestión de Stakeholders**
  - **Objetivo:** Aumentar el nivel de habilidades en Gestión de Stakeholders a 8/10 dentro de los próximos 6 meses.
  - **Acción:** Asistir a un taller de formación en Gestión de Stakeholders de 1 día, con un costo de \$800.
  - **Medición del éxito:** Realizar un ejercicio de gestión de stakeholders en un proyecto futuro, con un seguimiento y evaluación del desempeño en un plazo de 3 meses.

Estas acciones serán evaluadas y ajustadas según sea necesario para garantizar el progreso y el éxito de Marta en su camino hacia la estrategia líder.

Plan de desarrollo personalizado generado con IA

## 5. Dashboard de Talento Actual (Vista "RRHH")

Esta sección ([st.expander](#)) proporciona una visión profunda de la plantilla interna actual para Recursos Humanos.

- **Filtro por Chapter:** Permite a RRHH analizar un departamento a la vez.
- **Tabla de Empleados:** Un [st.dataframe](#) con la lista de empleados internos filtrada.
- **Gráfico 3: Promedio de Habilidades (Plotly):**
  - Un gráfico de barras horizontales que muestra el nivel promedio (1-10) de cada habilidad dentro del *chapter* seleccionado, identificando fortalezas y debilidades formativas.
- **Gráfico 4: Grafo Interactivo de Habilidades (Pyvis/NetworkX):**
  - Esta es la visualización más avanzada, que muestra las conexiones de talento.
  - **Nodos:** Los empleados se representan como nodos azules y las habilidades como nodos verdes.
  - **Aristas (Líneas):** Se dibuja una línea si un empleado tiene un nivel superior a 4 en una habilidad.
  - **Grosor de la Arista:** El grosor de la línea es directamente proporcional al nivel de habilidad (1-10), permitiendo a RRHH identificar visualmente a los expertos clave (nodos con líneas gruesas) y las habilidades raras (nodos con pocas conexiones).
  - **Interactividad:** El grafo es totalmente interactivo, permitiendo al usuario hacer zoom y arrastrar los nodos para explorar las relaciones.

Dashboard de Talento Actual (Vista de RRHH)

## Dashboard de Talento Actual

Filtrar por Chapter:

Todos

	nombre	chapter	rol_actual	manager
0	Jordi Casals	Strategy	Head of Strategy	None
1	Marta Soler	Strategy	Project Manager	Jordi Casals
2	Pablo Reyes	Martech	Martech Architect	Jordi Casals
3	Núria Aguilera	Martech	CRM Admin (HubSpot)	Pablo Reyes
4	Marc Serra	Martech	Data Analyst	Pablo Reyes
5	Adrián López	Creative	Creative Director	Jordi Casals
6	Nerea Vidal	Design	Senior Brand/UI Designer	Adrián López
7	Sergio Martín	Performance	Performance Media Specialist	Jordi Casals
8	Laura Torres	Performance	SEO/SEM Manager	Sergio Martín
9	Claudia Riera	Influency	Social Media Strategist	Adrián López

Promedio de Habilidades en: Todos

Nivel de Habilidad Promedio (0-10)

skill_nombre	nivel
Storytelling	8.5
Admin. CRM (HubSpot)	8.0
SEO/SEM	7.8
Diseño UI/UX	7.5
Google Ads	7.3
SQL y Python	7.2
Modelado de Datos	7.1
OKRs y Roadmapping	6.9
Copywriting	6.7
Analítica Digital	6.5
IA Generativa	6.3

Dashboard de Talento Actual: Tabla de empleados internos filtrada y Gráfico 3 del promedio de las habilidades dentro del chapter seleccionado.

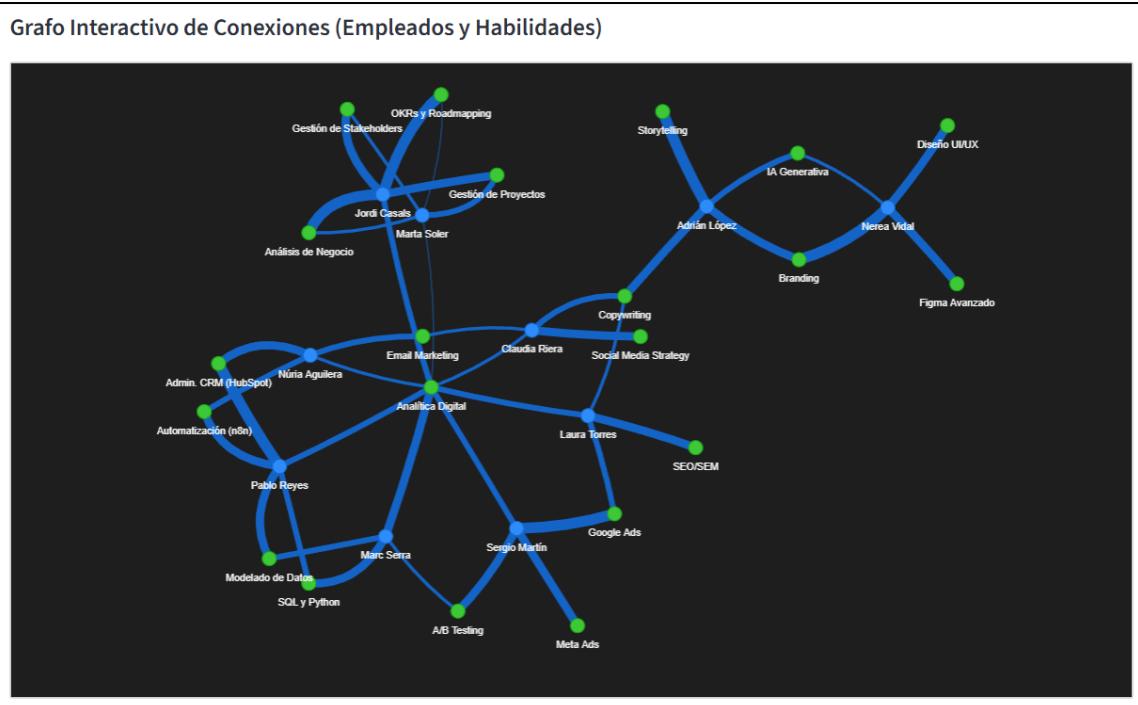


Gráfico 4. Grafo Interactivo de Habilidades: Conexiones de talentos con sus habilidades

# Qué Hace Cada Archivo

El código del proyecto sigue la siguiente estructura:

## 1. Archivos de Código (.py)

- **app.py**
  - Este es el archivo principal que ejecutas con `streamlit run app.py`.
  - **Qué hace:** Dibuja **toda la interfaz de usuario** (títulos, botones, gráficos, *sliders* y *expanders*).
  - Organiza la aplicación en sus 3 secciones (Dashboard Estratégico, Análisis Táctico, Dashboard RRHH).
  - Llama a los otros módulos (`data_loader`, `compatibility`, `recommendations`) para obtener los datos y realizar los cálculos.
- **modules/data\_loader.py**
  - **Qué hace:** Su única función es leer y preparar los datos.
  - Lee los archivos `org_config.json` y `vision_futura.json`.
  - Lee los dos archivos CSV: `talento_actual.csv` y `candidatos_externos.csv`.
  - **Une** los dos CSV en un único *DataFrame* de Pandas para que la app pueda analizar a internos y externos juntos.
  - Limpia las columnas de texto (ej. `{"S-OKR": 9}`) y las convierte en diccionarios de Python (`ast.literal_eval`).
- **modules/compatibility.py**
  - **Qué hace:** Es el corazón del Nivel 2. Su única función es calcular el *score* de compatibilidad.
  - Contiene la lógica del **algoritmo de ponderación**: 50% Habilidades, 25% Responsabilidades, 15% Ambiciones y 10% Dedicación.
  - Usa `scikit-learn` para comparar el texto de las responsabilidades.
- **modules/recommendations.py**
  - **Qué hace:** Su única función es conectarse a la API de Groq vía LangChain.
  - Contiene la función `generar_resumen_ejecutivo` (para el dashboard estratégico).
  - Contiene la función `generar_plan_desarrollo` (para los planes tácticos de cada candidato).
  - Se encarga de construir los *prompts* (instrucciones) para la IA.

## 2. Archivos de Datos (en la carpeta `data/`)

- **org\_config.json**
  - Define la "configuración" de la empresa.
  - Contiene la lista de **Roles** (ej. "R-STR-LEAD") y la lista de **Habilidades** (ej. "S-OKR").
- **vision\_futura.json**

- Define la "visión" de la empresa.
- Contiene la lista de **roles\_necesarios** que la empresa quiere contratar (esto rellena los selectores de la app).
- **talento\_actual.csv**
  - La base de datos de tus **empleados internos**.
  - Contiene sus niveles de habilidad, dedicación y metadatos de riesgo.
- **candidatos\_externos.csv**
  - La base de datos de tu "**Talent Pool**" **externo**.
  - El formulario de "Candidato / CV" guarda los datos en este archivo.

# Guía de Ejecución y Arquitectura de Código

Esta sección detalla los pasos técnicos para ejecutar la aplicación y describe la función de cada archivo de código en el proyecto.

## Para ejecutar los códigos:

**Paso 1: Configurar el Entorno e Instalar Dependencias** Desde la terminal, en la carpeta raíz del proyecto ([Hackaton](#)), ejecute:

# 1. Activar el entorno virtual

`.\venv\Scripts\activate`

# 2. Instalar las librerías base del reto

`pip install -r requirements.txt`

# 3. Instalar las librerías adicionales para IA y Grafos

`pip install langchain-groq networkx pyvis`

**Paso 2: Configurar la Clave de API** Para que las funciones de IA (Nivel 3) funcionen, crea un archivo llamado `.env` en la carpeta raíz. Después, se añade la clave de API de Groq dentro de este archivo:

`GROQ_API_KEY="su_clave_secreta_aqui"`

**Paso 3: Ejecutar la Aplicación** Una vez configurado el entorno, lance la aplicación:

`streamlit run app.py`

La aplicación se abrirá automáticamente en su navegador web.

# Posibles Futuras Implementaciones

Aunque el prototipo actual es una solución integral que cumple con los requisitos del reto (Nivel 1, 2 y 3), la arquitectura permite un *roadmap* de expansiones muy potente. Las siguientes implementaciones se centrarían en pasar de un sistema de "soporte a la decisión" a un sistema "predictivo".

Cabe destacar que, si bien estas ideas son el siguiente paso lógico, **no se implementaron en el prototipo actual debido a la limitación de los datos de muestra**. El dataset [talento\\_actual.csv](#) contiene solo 10 empleados, lo cual es insuficiente para entrenar modelos de Machine Learning complejos sin caer en un severo *underfitting* (modelos que no pueden generalizar patrones) o obtener resultados estadísticamente irrelevantes.

A continuación, se detallan las implementaciones futuras, asumiendo un dataset más robusto (ej. +100 empleados):

## Extracción y Análisis Automático de CVs

- **Objetivo:** Automatizar completamente la ingesta de datos del "Talent Pool" externo.
- **Implementación:** El formulario de "Candidato Externo" actualmente incluye un [st.file\\_uploader](#) para "Adjuntar CV". La siguiente iteración utilizaría un modelo de IA (OCR + LLM) para leer el PDF o DOCX, extraer automáticamente las [habilidades](#) y [responsabilidades\\_actuales](#), y estimar su nivel (1-10) basándose en el contexto del CV.
- **Beneficio:** Eliminaría la entrada manual de datos para candidatos externos, permitiendo a RRHH construir el "Talent Pool" de forma masiva y rápida.

## Modelo de Clustering para Descubrimiento de Roles

- **Objetivo:** Encontrar patrones y "roles ocultos" en la plantilla actual.
- **Implementación:** Se aplicaría un algoritmo de *clustering* no supervisado (como K-Means) a la matriz de habilidades de los empleados.
- **Justificación de la No-Implementación:** Con solo 10 empleados, los *clusters* resultantes no serían significativos. Con un dataset más grande, este modelo podría:
  - **Optimizar la Estructura:** Descubrir que dos roles que la empresa considera diferentes, en la práctica usan un *cluster* de habilidades casi idéntico.
  - **Descubrir Nuevos Roles:** Identificar un grupo de empleados que comparten un conjunto único de habilidades (ej. "IA + Creativo"), revelando un nuevo rol que la empresa tiene pero no ha definido formalmente.

## PCA (Principal Component Analysis) para Optimización de Habilidades

- **Objetivo:** Reducir el ruido y la redundancia en el modelo de compatibilidad.
- **Implementación:** La empresa define 21 habilidades, y es probable que muchas estén altamente correlacionadas. Se utilizaría un PCA para analizar estas correlaciones.

- **Justificación de la No-Implementación:** El PCA requiere una varianza de datos suficiente para identificar componentes principales. Con solo 10 *samples*, los resultados del PCA serían estadísticamente inestables.
- **Beneficio (a futuro):** Reduciría las 21 habilidades a 5 o 6 "Macro-Competencias" (ej. "Competencia Estratégica", "Competencia de Performance"), haciendo que el algoritmo de compatibilidad sea más rápido y robusto.

### Modelo Multi-Classifier para Predicción de Roles

- **Objetivo:** Predecir el rol más adecuado para un nuevo candidato basándose solo en sus habilidades.
- **Implementación:** Se entrenaría un modelo de Machine Learning supervisado (como un SVM) usando las *habilidades* como variables de entrada (X) y el *rol\_actual* como la variable objetivo (Y).
- **Justificación de la No-Implementación:** Esta es la razón más clara. Como se indica en la propuesta original, el *dataset* actual de 10 empleados hace **imposible** entrenar un modelo multiclasa preciso. El modelo sufriría de un *underfitting* severo (demasiada simplicidad), siendo incapaz de aprender las fronteras de decisión entre roles.
- **Beneficio (a futuro):** Con +100 empleados, el modelo podría sugerir automáticamente el rol más probable para un nuevo candidato y generar visualizaciones 2D de las "fronteras" de talento entre departamentos.

# Conclusiones del Proyecto

Al finalizar este hackathon, nuestra conclusión va más allá del código. El reto de Quether no era solo un desafío de automatización; era un desafío de *perspectiva*. Creemos que hemos superado las expectativas en cuatro áreas clave:

1. **El Descubrimiento de los "Datos Ocultos":** Nuestra verdadera innovación no fue solo implementar el algoritmo del Nivel 2, sino lo que descubrimos al explorar los datos. El archivo [talento\\_actual.csv](#) ocultaba la información más valiosa: [retention\\_risk](#) y [dedicación\\_actual](#). En ese momento, el proyecto cambió. Decidimos que la app *tenía* que sacar esto a la luz. Ese fue el nacimiento de nuestro Dashboard Estratégico, una función que no se pedía explícitamente, pero que se convirtió en el corazón de la "Vista General".
2. **Más Allá de Gráficos de Barras:** Estamos especialmente orgullosos de las visualizaciones. No nos conformamos con los gráficos de barras estándar de Plotly (aunque los usamos para los KPIs). Para la vista de RRHH, sabíamos que un grafo era la única respuesta. El grafo de red interactivo (Pyvis) que implementamos es nuestra visión de cómo RRHH debería ver el talento: no como filas en una tabla, sino como una red de conexiones vivas, donde el grosor de la línea (el nivel 1-10) realmente importa.
3. **Una IA con Doble Propósito (Nivel 3):** No nos limitamos a "añadir IA". Nuestra implementación de LangChain tiene una personalidad dual: es **táctica** (genera planes de desarrollo personalizados para los managers) y es **estratégica** (genera resúmenes ejecutivos para el CEO). Demostramos que la IA no es solo un *gadget*, sino una herramienta de soporte a la decisión en todos los niveles.
4. **De "Datos Sucios" a una Arquitectura Limpia:** El desafío más grande no fue la IA, fue la ingesta de datos. Los [org\\_config.json](#) y [vision\\_futura.json](#) venían en formato RTF. Superar eso y luego diseñar una arquitectura limpia (separando [internos](#) de [externos](#) en distintos CSVs) nos dio una base sólida y escalable.

**Conclusión Final:** No hemos entregado simplemente un "analizador de gaps". Hemos construido un motor de decisión proactivo. Este proyecto demuestra que los datos que una empresa ya tiene, incluso si están "sucios" u ocultos en columnas de metadatos, son la clave para predecir el riesgo, gestionar la capacidad y, en definitiva, construir el futuro.