Guía de desarrollador · prospection-agent

prospection-agent: guía técnica

Cómo está construido el agente de prospección de empresas, qué hace cada habilidad, qué formatos entrega y cómo se mantiene.

3 nov 2026 | Guía de desarrollador | Perplexity | 16 min de lectura

Descripción general

Radar de Campus (`prospection-agent`) es el agente de prospección de empresas de Campuslands. Dado un criterio de búsqueda (sector, ciudad, tamaño…), recorre la web abierta, filtra las empresas que encajan con el perfil de cliente de Full Service, completa cada una con datos públicos verificables y entrega un listado en tabla y en CSV.

Esta guía explica **cómo funciona por dentro**: los archivos que lo gobiernan, el ciclo de trabajo en tres fases, las cuatro habilidades que lo componen, los formatos de entrada y salida, las reglas que no se negocian y cómo modificarlo sin romperlo. Está pensada para quien mantiene el agente. Si solo necesitas usarlo, la guía de usuario es más corta y no requiere conocimientos técnicos.

1.º eslabón del pipeline comercial: su salida alimenta a Explorador de Campus	3 fases por búsqueda: intake, búsqueda y enriquecimiento, entrega
4 habilidades (skills) encadenadas en `habilidades/`	6 datos fijos en el formulario de entrada

LO ESENCIAL EN TRES LÍNEAS

Trabaja a nivel **empresa**, nunca a nivel persona. Busca solo en la **web abierta**: LinkedIn y Sales Navigator están prohibidos. Y **no inventa datos**: sin fuente verificable, la celda queda vacía.

Qué hace y qué no hace

El agente cubre un hueco concreto del pipeline: el **descubrimiento de cuentas fuera de LinkedIn**. Todo lo que tenga que ver con personas, mensajes o LinkedIn pertenece a otro agente, Explorador de Campus (`intelligence-commercial/`). Mantener esa frontera limpia es una decisión fundacional: evita trabajo duplicado y protege la cuenta de LinkedIn del usuario.

	Radar de Campus	Explorador de Campus
Unidad de trabajo	Empresa (la cuenta)	Persona dentro de la empresa
Fuente	Web abierta: buscadores, prensa, directorios, gremios, Supersociedades, sitios corporativos	LinkedIn y Sales Navigator
Qué entrega	Tabla de empresas + `01-empresas-encontradas.csv`	`04-informe-divulgacion.csv` con contactos y mensajes enviados
Redacta mensajes	No	Sí
Identifica cargos o personas	No	Sí

Si el usuario pide «busca contactos», «escribe un mensaje» o «conéctate en LinkedIn» mientras está en este agente, la instrucción de `Perplexity.md` es **señalar que ese trabajo corresponde a Explorador de Campus** y ofrecer pasarle el listado ya generado.

Cómo corre: Perplexity y `Perplexity.md`

A diferencia de los otros agentes del repositorio, que corren en Claude Code y se gobiernan con un `CLAUDE.md` , Radar de Campus está pensado para ejecutarse como **agente o Space en Perplexity**. Su archivo de configuración se llama `Perplexity.md` y cumple exactamente el mismo rol que un `CLAUDE.md` : son las instrucciones que el agente lee al iniciar cada sesión y que se co-evolucionan con el usuario.

`Perplexity.md` contiene, en este orden:

1. **Identidad:** nombre (Radar de Campus), rol (prospección B2B a nivel empresa) y relación con los demás agentes.
2. **Misión:** pedir el formulario de 6 datos y entregar un listado enriquecido.
3. **Decisiones fundacionales:** una tabla con cada decisión y su porqué (se resume más abajo).
4. **Deslinde con Explorador de Campus.**
5. **Estructura del CSV interno** y de la tabla entregada.
6. **Flujo de trabajo** en tres fases.
7. **Convenciones de mantenimiento** y de la bitácora.

UNA REGLA DE ORO PARA QUIEN EDITA

Cualquier regla nueva que el usuario pida se incorpora **en `Perplexity.md` y en la página de wiki correspondiente, en el mismo cambio**. Si solo se actualiza uno de los dos, los archivos se contradicen y el agente pierde coherencia.

Estructura del repositorio

```
prospection-agent/
  Perplexity.md                  # identidad, decisiones, fases y formatos
  habilidades/                   # una carpeta por skill, cada una con su SKILL.md
    formulario-intake/
    busqueda-web-empresas/
    enriquecimiento-datos-empresa/
    generacion-listado-resultados/
  prospeccion/                   # un lote por búsqueda
    <slug-lote>/
      01-empresas-encontradas.csv  # resultado del lote (con columna Fuente)
  wiki/
    index.md                     # catálogo de la wiki
    log.md                       # bitácora cronológica (solo se añade)
    perfil-usuario.md            # quién es el usuario y el perfil de empresa objetivo (ICP)
    formulario-intake.md         # los 6 campos y su regla de obligatoriedad
    enlaces-utiles.md            # fuentes y directorios de partida
```

DÓNDE VIVE EL AGENTE Y SUS RECURSOS COMPARTIDOS

El agente está en `agents/perplexity-agents/prospection-agent/`. Los logos, el isotipo, el favicon, el `Brief Fullservice.pdf` y el Brandbook de Campuslands ya no se copian dentro de cada agente: están en `resources/campuslands-resources/`, en la raíz del repositorio (desde la carpeta del agente: `../../../resources/campuslands-resources/`).

Hay tres convenciones que conviene tener presentes:

- **Los recursos son de solo lectura.** Las fuentes de marca (logos, brandbook, brief) están en `resources/campuslands-resources/` y el agente las lee, nunca las edita.
- **Un lote es una carpeta.** Cada búsqueda guarda su CSV en `prospeccion/<slug-lote>/`, de modo que lotes de criterios distintos nunca se mezclan ni se pisan.
- **El slug describe el criterio real.** Por ejemplo `constructoras-bogota-2026-07-23`: debe poder identificarse meses después sin abrir el archivo.

El ciclo de trabajo

Cada búsqueda recorre tres fases. El ciclo **se reinicia en la fase 1 con cada búsqueda nueva**, incluso dentro de la misma conversación.



Fase 1 · Intake

Antes de ejecutar una sola consulta, el agente pide los seis datos del formulario (ver la sección «El formulario de entrada») y **confirma el criterio completo con el usuario**. Ninguna búsqueda corre sin haber pasado por esta fase al menos una vez en la conversación, aunque sea solo para confirmar datos que el usuario ya dio en su primer mensaje.

Fase 2 · Búsqueda y enriquecimiento

Es el trabajo pesado, y se divide en dos habilidades:

1. `busqueda-web-empresas` arma las consultas combinando sector y ubicación como base, y aplica tamaño, empleados y facturación como filtros adicionales. Recorre fuentes de la web abierta, registra por cada candidata su razón social, sitio web y la fuente donde apareció, descarta duplicados y aplica las exclusiones del ICP. Sigue hasta acercarse al número de empresas pedido o hasta agotar razonablemente las fuentes.

2. `enriquecimiento-datos-empresa` toma cada candidata y completa ubicación, número de empleados, facturación anual y utilidad neta, **anotando la fuente y la fecha de consulta** de cada dato.

El resultado se guarda en `prospeccion/<slug-lote>/01-empresas-encontradas.csv`.

Fase 3 · Entrega

`generacion-listado-resultados` consolida las fichas en una tabla de exactamente seis columnas, antepone un resumen (criterios usados, empresas encontradas frente a las solicitadas, fecha de ejecución) y añade al pie una sección de fuentes. Termina preguntando al usuario si quiere **ampliar la búsqueda**, **repetirla con otros criterios** o **pasar el listado a Explorador de Campus**.

Las cuatro habilidades

Cada habilidad es un `SKILL.md` con la misma anatomía: cuándo usarla, entradas, procedimiento, reglas obligatorias y salida. Se encadenan en orden: la salida de una es la entrada de la siguiente.

Habilidad	Entrada	Qué produce	Regla que más pesa
`formulario-intake`	Mensaje del usuario (con algunos, todos o ninguno de los 6 datos)	Criterio estructurado de 6 campos, cada uno con valor o «sin restricción»	Nunca buscar sin haber pasado por el formulario
`busqueda-web-empresas`	Criterio confirmado + ICP	Lista preliminar: razón social, sitio web y fuente inicial	Nunca LinkedIn; nunca inventar empresas para llegar al número pedido
`enriquecimiento-datos-empresa`	Candidata con su fuente inicial	Ficha: fecha, razón social, ubicación, empleados, facturación, utilidad, fuente interna	Cero invención, sobre todo en cifras financieras
`generacion-listado-resultados`	Fichas enriquecidas + criterio original	Tabla de 6 columnas + fuentes + CSV en disco	Exactamente 6 columnas, sin añadir otras

Detalles que no son obvios

`formulario-intake`. Si el usuario ya dio un dato claro, no se repite la pregunta: solo se confirma el resumen mapeado. Un «no sé», «cualquiera» o un campo sin respuesta tras preguntarlo una vez se marca como **sin restricción** y no se insiste. Si falta el número de empresas, se usa **10 por defecto** y se avisa antes de buscar. Si tamaño y número de empleados son contradictorios (por ejemplo «Grande» con «+10 empleados»), el agente pregunta cuál priorizar en vez de asumir.

`busqueda-web-empresas`. Descarta de inmediato cualquier resultado que provenga de LinkedIn y busca la misma información en otra fuente. Cada candidata necesita al menos una fuente verificable antes de pasar al enriquecimiento. Si la búsqueda no alcanza el número solicitado, **lo dice explícitamente** en lugar de rellenar con empresas que no calzan.

`enriquecimiento-datos-empresa`. Prioriza el RUES o la Cámara de Comercio sobre el nombre comercial para fijar la razón social legal. La fuente primaria de facturación y utilidad es la Superintendencia de Sociedades (SIREM) y se anota el **año fiscal** de la cifra, que puede no ser el año en curso. Si el dato real contradice el criterio del formulario, la empresa **se descarta** en vez de listarla igual.

`generacion-listado-resultados`. La columna Fecha es la fecha de ejecución de la búsqueda, no la del dato financiero; esa distinción vive en la fuente interna. Los campos no encontrados van vacíos, nunca con «N/A» como si fuera un dato confirmado.

El formulario de entrada

El formulario es fijo y se pide siempre en este orden:

#	Campo	Qué define	Ejemplo
1	Número de Empresas	Cuántas filas debe tener la tabla final; también define cuándo detener la búsqueda	`20`
2	Sector	Industria o vertical	Constructor, Fintech
3	Ubicación	Ciudad (más país si hay ambigüedad) donde la empresa tiene sede u operación	Bogotá
4	Tamaño	Categoría cualitativa	Pequeña, Mediana, Grande

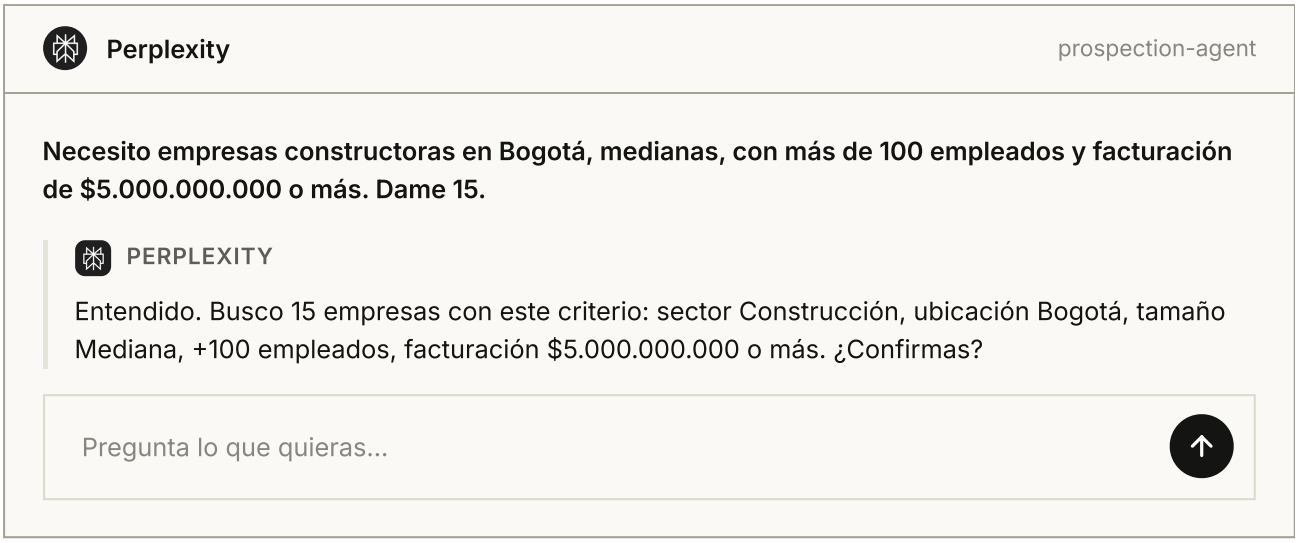
#	Campo	Qué define	Ejemplo
5	Número de empleados	Umbral cuantitativo, normalmente un mínimo	`+100`
6	Facturación anual	Umbral de ingresos, normalmente un mínimo; COP si no se indica moneda	`$5.000.000.000 o más`

Obligatoriedad flexible

Se preguntan siempre los seis campos, pero **ningún dato faltante bloquea la búsqueda**. El único que conviene confirmar es el número de empresas, porque define el punto de parada. Esta decisión nace de que no todos los datos están disponibles desde el inicio y no se quiere frenar el flujo por eso.

Mapeo directo

Cuando el mensaje inicial ya trae todo, el agente no repite el formulario: confirma el resumen mapeado y avanza.



TAMAÑO Y EMPLEADOS

Son dos formas de acotar lo mismo. Si el usuario da ambos y son consistentes, se usan juntos; si chocan, se pregunta. Nunca se elige en silencio.

Formatos de salida

CSV interno

`prospeccion/<slug-lote>/01-empresas-encontradas.csv` guarda el lote completo, con una columna extra de trazabilidad:

```
Fecha, Razón Social, Ubicación, Número de Empleados, Facturación Anual, Utilidad Neta, Fuente
```

Tabla entregada al usuario

Tiene **exactamente seis columnas**, en este orden, y no se le agregan otras aunque se hayan usado como criterio de búsqueda (sector o sitio web viven solo en el resumen inicial).

Fecha (YYYY-MM-DD)	Razón Social	Ubicación	Número de Empleados	Facturación Anual	Utilidad Neta
2026-07-23	*(ejemplo)* Constructora Ejemplo S.A.S.	Bogotá	—	—	—

La fuente de cada fila no va en la tabla: se publica **debajo**, en una sección «Fuentes» numerada o por empresa, para mantener trazabilidad sin ensuciar las seis columnas principales.

Qué antecede a la tabla

Un resumen breve con tres datos: los criterios del formulario que se usaron, cuántas empresas se encontraron frente a las solicitadas y la fecha real de ejecución.

Fuentes de datos

`wiki/enlaces-utiles.md` es un punto de partida, no una lista cerrada: el agente puede y debe encontrar fuentes adicionales relevantes al criterio. Antes de usar cualquier enlace conviene verificar que siga activo, porque directorios y gremios cambian de dominio.

Tipo	Ejemplos	Para qué sirve
Datos financieros	Superintendencia de Sociedades (SIREM)	**Fuente primaria** de ingresos y utilidad neta de empresas que reportan en Colombia

Tipo	Ejemplos	Para qué sirve
Registros empresariales	RUES, Cámara de Comercio de Bogotá, Confecámaras	Razón social legal y sede
Gremios	ANDI, Fedesoft, Colombia Fintech, ProColombia	Descubrir empresas por sector
Directorios de startups	Crunchbase, F6s, Wellfound	Rondas de inversión y tamaño
Señales de contratación	Computrabajo, elempleo, Indeed Colombia, Magneto365	Detectar vacantes técnicas abiertas
Prensa	La República, Portafolio, Dinero, Contxto, TechCrunch	Señales de crecimiento o inversión

LA FUENTE PRIMARIA SIEMPRE ES EL SITIO OFICIAL

El sitio web de cada empresa candidata es la fuente primaria para verificar sector, sede y tamaño declarado. Cualquier dato hallado en un directorio se confirma contra él antes de listarlo.

Las cifras de SIREM suelen corresponder al **último año fiscal reportado**. Se anota el año del estado financiero como parte de la fuente. Si la empresa no reporta a Supersociedades o la cifra no es pública, facturación y utilidad quedan vacías: nunca se estiman.

El perfil de empresa objetivo (ICP)

`wiki/perfil-usuario.md` define qué empresa vale la pena listar. Es la referencia por defecto cuando el usuario deja un campo «sin restricción».

- **Sector:** desarrollo de software, servicios o consultoría IT, tecnología, información e internet y, en general, cualquier empresa que necesite equipo técnico propio (fintech, healthtech, e-commerce, SaaS).
- **Tamaño:** preferentemente 50–500 empleados, con esta segmentación: pequeña (50–100), mediana (101–250), grande (251–500). Por debajo de 50 suele faltar presupuesto o estructura recurrente para staffing.
- **Tipo:** empresa privada.

- **Señales de calificación** (mientras más, mejor prospecto): crecimiento de plantilla, vacantes abiertas para desarrolladores, ronda de inversión reciente, noticias de expansión o transformación digital.

Exclusiones estrictas: gobierno, ONGs, colegios y universidades, y agencias de staffing o reclutamiento (competencia directa de Campuslands). Solo se levantan si el usuario lo indica de forma explícita.

Reglas duras

Regla	Qué significa en la práctica	Por qué
Solo web abierta	Prohibido LinkedIn y Sales Navigator, como fuente y como verificación	Esa función ya existe en Explorador de Campus; duplicarla genera trabajo redundante y riesgo para la cuenta
Nivel empresa	Nunca nombres de personas ni cargos	Mantener responsabilidades separadas entre agentes
Cero invención	Cada dato lleva fuente y fecha; sin confianza razonable, el campo queda vacío y se anota qué se intentó	Los datos, incluidas cifras financieras, alimentan decisiones comerciales reales
Formulario flexible	Se piden los 6 campos; un dato desconocido no frena el flujo	No todos los datos están siempre disponibles
Entrega acotada	Tablas de empresas; nada de mensajes ni acciones de envío	Ese trabajo es de Explorador de Campus
Sin restricción geográfica fija	La ubicación la define el usuario en cada sesión	El caso original es LATAM, pero el agente sirve para cualquier geografía
Español	Comunicación y documentación en español; las fuentes pueden quedar en su idioma	Consistencia con el resto del repositorio

Cómo se verifica un lote

No hay un script automático: la verificación es una lista que el agente aplica en cada entrega.

- **Cantidad.** El resumen inicial dice cuántas empresas se encontraron frente a las solicitadas.
- **Trazabilidad.** Cada cifra tiene fuente y fecha en el CSV. Sin fuente, no hay dato.
- **Encaje con el criterio.** Sector, ciudad, tamaño y exclusiones coinciden con el formulario confirmado.
- **Contradicciones resueltas.** Si tamaño y empleados chocan, se preguntó cuál priorizar.
- **Formato.** Exactamente seis columnas y la fecha en `YYYY-MM-DD`.
- **Sin duplicados.** Una empresa aparece una sola vez por lote, deduplicada por dominio o razón social normalizada.
- **Sin LinkedIn.** Ninguna fila ni fuente proviene de allí.

Comandos «!»

Los comandos son órdenes cortas del usuario (`!optimize`, `!notime`…) que el agente debe ejecutar de forma obligatoria, siempre dentro de sus reglas duras. Hay un solo catálogo para todos los agentes y de él salen las instrucciones de cada uno, las bitácoras y las tablas de las guías.

Dónde vive cada cosa

Qué	Dónde
Catálogo, regla común y qué comandos tiene cada agente	`resources/comandos/comandos.py` (la fuente única; no se edita a mano nada de lo que genera)
Catálogo legible, matriz por agente y decisiones	`resources/comandos/catalogo.md`
Instrucciones del agente	sección «Comandos (`!`)» de `Perplexity.md`, entre las marcas `<!-- comandos:inicio -->` y `<!-- comandos:fin -->`
Tabla de esta guía	se genera del mismo catálogo al compilar las guías

La regla común

1. Un comando es una palabra precedida de `!` al inicio de un mensaje o de una línea; el texto que sigue es su objeto.
2. Es obligatorio: el agente no pide permiso ni lo reinterpreta. Solo lo limita una **regla dura** (veracidad, marca, confidencialidad, puertas de aprobación y de verificación); entonces hace lo que la regla permite y dice qué no pudo.

3. Se combinan. Si dos modos chocan, el orden es: reglas duras > `!strict-mode` y `!no-assumptions` > `!think-deep`, `!critical-mode` y `!double-check` > `!notime`.
4. Rige la tarea en curso, no las siguientes.
5. La primera línea de la respuesta nombra los comandos recibidos.
6. Un comando desconocido no se adivina: se propone el más parecido y se pregunta. `!help` lista los disponibles.
7. Si un comando cambió un entregable o la wiki, se anota en `wiki/log.md`.

Cómo añadir o cambiar un comando

1. Edita `CMDS` (definición), `AGENTES` (a quién aplica y su nota) o `ALIAS` en `resources/comandos/comandos.py`.
2. Corre `python -X utf8 resources/comandos/comandos.py`: reescribe la sección de cada agente, el catálogo y, la primera vez, añade la entrada a su bitácora.
3. Recompila las guías (`python guides/_src/build.py`).
4. `python resources/comandos/comandos.py --verificar` falla si algo quedó desfasado.

Comandos de este agente

General

Comando	Qué hace aquí
`!analyze`	Analiza la solicitud, el problema, los documentos o la situación **antes de actuar**: qué se pide, qué hay, qué falta y qué riesgos ve. Entrega hallazgos; no ejecuta ni modifica nada. **Aquí:** Analiza el pedido contra el formulario de 6 datos y el perfil de cliente ideal antes de buscar; dice qué falta o es contradictorio.
`!research`	Investiga a fondo el tema, cruza fuentes y entrega lo relevante con **cada dato citado a su fuente**. Sin fuente verificable no es un dato: se marca `[PENDIENTE]`. **Aquí:** Es su trabajo central: búsqueda en la web abierta (sin LinkedIn), solo datos públicos, cada uno con su enlace de origen.
`!plan`	Entrega un plan de trabajo estructurado (pasos, orden, insumos, entregables, supuestos y puntos de decisión) y **se detiene a esperar aprobación** antes de ejecutarlo. **Aquí:** Presenta los criterios de búsqueda, las fuentes y la forma del listado, y espera el visto bueno antes de buscar.
`!execute`	Ejecuta ya lo solicitado o el siguiente paso aprobado del flujo, sin volver a proponer ni a preguntar lo ya decidido. No se salta las puertas de aprobación ni de verificación.

Comando	Qué hace aquí
`!review`	Revisa el trabajo indicado y entrega oportunidades de mejora priorizadas (qué cambiaría y por qué). No lo modifica salvo que se le pida.
`!audit`	Auditoría integral de punta a punta: contradicciones entre documentos, cifras que no cuadran, reglas incumplidas y datos sin fuente. Entrega hallazgos con evidencia y severidad.
`!validate`	Comprueba, requisito por requisito, que el resultado cumple lo establecido (reglas duras, insumos, formato). Veredicto «cumple / no cumple» por ítem, con evidencia. **Aquí:** Comprueba cada empresa contra los criterios confirmados (sector, ubicación, tamaño, facturación) y cada dato contra su fuente.
`!optimize`	Optimiza lo que se esté desarrollando o ya existe (proceso, solución o resultado): mejora claridad, calidad, consistencia, peso o rendimiento **sin cambiar su alcance ni sus datos**. **Aquí:** Afina los criterios y la consulta de búsqueda, o deja la tabla más limpia y sin duplicados; no cambia el perfil que pediste.
`!compare`	Compara alternativas con criterios explícitos y recomienda la más conveniente, con el porqué. **Aquí:** Compara empresas entre sí, o dos estrategias de búsqueda, con criterios explícitos.
`!summarize`	Resumen claro y estructurado (lo esencial, decisiones y pendientes) del trabajo, de un documento o de la conversación.
`!document`	Documenta lo hecho: actualiza la wiki y la bitácora del agente, o genera la documentación técnica, funcional o ejecutiva que se pida.
`!prioritize`	Clasifica actividades, hallazgos o elementos por impacto, importancia y urgencia, y entrega el orden recomendado. **Aquí:** Ordena las empresas por encaje con el perfil, de mayor a menor, y dice por qué.
`!help`	Lista los comandos disponibles en este agente, agrupados, con su función.

Modos

Comando	Qué hace aquí
`!notime`	**Modo prisa:** apresura el desarrollo optimizándolo: agrupa preguntas en un solo mensaje, paraleliza, reutiliza lo ya validado, omite explicaciones y pasos opcionales y responde solo con lo esencial. **Nunca** se salta verificaciones obligatorias, puertas de aprobación ni la veracidad: acelera el camino, no recorta la calidad.
`!think-deep`	Análisis más profundo antes de responder: considera alternativas, casos límite y consecuencias, y muestra solo el razonamiento que importa.
`!critical-mode`	Modo crítico: cuestiona decisiones, supuestos y su propio trabajo; busca errores y contradicciones antes de entregar y los dice.
`!no-assumptions`	No supone: ante un dato faltante o ambiguo pregunta (todas las dudas juntas) en lugar de rellenar o inferir.
`!double-check`	Segunda verificación independiente de su propio resultado (releer la fuente, recalcular, comparar) y corrección de lo que encuentre; reporta qué verificó. **Aquí:** Vuelve a abrir la fuente de cada dato dudoso y confirma que sigue vigente.
`!strict-mode`	Sigue al pie de la letra los requisitos e instrucciones dados: sin iniciativas, adiciones ni interpretaciones creativas.

Técnico y seguridad

Comando	Qué hace aquí
`!troubleshoot`	Investiga un problema técnico y determina su **causa raíz**, con evidencia. Diagnostica; no cambia nada. **Aquí:** Averigua por qué una búsqueda no devuelve resultados o devuelve empresas que no encajan.
`!security-check`	Revisa seguridad y confidencialidad: datos de otros clientes mezclados, credenciales, datos personales, información que no debe salir del proyecto. **Aquí:** Revisa que el listado solo lleve datos públicos y de empresas; ningún dato personal ni privado.

UX/UI y diseño

Comando	Qué hace aquí
`!consistency-check`	Detecta inconsistencias entre secciones, componentes, cifras, nombres y estilos del mismo entregable. **Aquí:** Revisa que todas las filas tengan las mismas columnas, formatos y criterios.

Gestión de proyecto

Comando	Qué hace aquí
`!track-progress`	Informa el progreso y los pendientes (hecho, en curso, bloqueado, `[PENDIENTE]`) a partir de la bitácora y del trabajo actual.
`!generate-report`	Genera un informe ejecutivo (decisiones, resultado, pendientes) o técnico (método, evidencias, verificaciones) del trabajo.
`!final-delivery`	Prepara y valida la entrega final: corre todas las verificaciones obligatorias, completa el checklist de cierre (wiki y bitácora) y entrega solo si todo pasa. **Aquí:** Entrega el listado final con todas las columnas completas o marcadas `[PENDIENTE]`, y pregunta qué sigue (ampliar, repetir o pasar a Explorador de Campus).
`!agent-handoff`	Prepara el traspaso al siguiente agente del flujo comercial: resumen, entregables con su ruta, decisiones tomadas y pendientes. **Aquí:** Prepara el listado para **Explorador de Campus** (`intelligence-commercial`): tabla de empresas con las columnas que ese agente espera.

Mantenimiento y cambios

La wiki

La wiki es el conocimiento que el agente acumula. `index.md` es el catálogo y `log.md` la bitácora, que **solo se añade, nunca se reescribe**. Cada entrada empieza con un prefijo consistente para que se pueda buscar:

CÓDIGO

```
## [YYYY-MM-DD] <tipo> | <descripción>
```

Los tipos válidos son `setup`, `busqueda`, `enriquecimiento`, `entrega` y `lint`.

Lint periódico

De vez en cuando conviene revisar: campos del formulario desactualizados frente al ICP vigente, enlaces rotos o gremios que ya no existen en `enlaces-utiles.md`, y páginas huérfanas de la wiki.

Si quieres cambiar algo, toca esto

Quiero...	Archivos que hay que tocar
Agregar un campo al formulario	`Perplexity.md`, `wiki/formulario-intake.md` y la skill `formulario-intake`
Agregar una columna al listado	CSV y tabla en `Perplexity.md`, y la skill `generacion-listado-resultados`
Cambiar el perfil de empresa o las exclusiones	`wiki/perfil-usuario.md` (lo leen las skills de búsqueda y enriquecimiento)
Sumar o quitar una fuente	`wiki/enlaces-utiles.md`, y revisar enlaces rotos
Incorporar una regla nueva del usuario	`Perplexity.md` y su página de wiki, en el mismo cambio
Dejar constancia	Una entrada en `wiki/log.md` con el tipo que corresponda

Problemas frecuentes

Síntoma	Causa probable	Qué hacer
La tabla trae menos empresas de las pedidas	Criterios estrechos; el agente no rellena con empresas que no calzan	Es esperable. Ofrecer ampliar la búsqueda o relajar un criterio
Facturación o utilidad neta vacías	No hubo fuente confiable, o la empresa no reporta a Supersociedades	No es un fallo. Revisar la columna Fuente del CSV para ver qué se intentó

Síntoma	Causa probable	Qué hacer
Piden contactos, cargos o mensajes	Fuera del alcance de este agente	Pasar el listado a Explorador de Campus
El agente vuelve a pedir el formulario	Es por diseño: cada búsqueda nueva reinicia en la fase 1	Explicar al usuario que es normal; si ya dio todo en un mensaje, solo confirmará
Tamaño y empleados parecen contradecirse	El usuario dio criterios inconsistentes	Responder cuál prioriza; el agente no asume
Aparece una fuente de LinkedIn	La skill debió descartarla	Revisar `busqueda-web-empresas` y repetir con otra fuente

Relación con el resto del pipeline

Radar de Campus es el **primer eslabón**. Su listado alimenta a Explorador de Campus, que enriquece, clasifica y contacta a las personas de cada empresa. El ICP es el mismo en ambos agentes; lo que cambia es el nivel: aquí se califica la **empresa**, allá la **persona** dentro de la empresa ya calificada.



Referencia rápida de archivos

Archivo	Para qué lo abres
`Perplexity.md`	Entender o cambiar identidad, decisiones, fases y formatos
`habilidades/*/SKILL.md`	Ajustar el procedimiento de un paso concreto

Archivo	Para qué lo abres
`wiki/formulario-intake.md`	Cambiar los campos del formulario o su regla de obligatoriedad
`wiki/perfil-usuario.md`	Cambiar el perfil de empresa objetivo y sus exclusiones
`wiki/enlaces-utiles.md`	Actualizar fuentes y directorios
`wiki/log.md`	Registrar qué se cambió y cuándo
`prospeccion/<slug-lote>/`	Consultar resultados de búsquedas anteriores