دستیار جلسه
نقشه راه و کارهای آینده

این سند هر آنچه را که *پس از* MVP می‌آید یک‌جا گرد می‌آورد: نقشه‌ی راه محصول به‌تفکیک اسپرینت، سه قابلیت پیشرفته‌ی متمایزکننده (راهنمای برگزارکننده، هم‌راستایی اسناد، تحلیل احساسات)، چرخه‌ی کیفیت عملیاتی (MLOps)، و مجموعه‌ای از ایده‌های افق‌بلندتر پژوهشی. این، همراهِ «ایده‌های آینده»ی سند ۱ (چشم‌انداز) و سند ۲ (سامانه‌ی فعلی) است.

> **راهنمای وضعیت.** اسپرینت ۱ (همان MVP) تکمیل شده است. در میان قابلیت‌های پیشرفته، **راهنمای برگزارکننده** هم‌اکنون پیاده‌سازی شده (`GET /api/meetings/{id}/facilitation`)؛ هم‌راستایی اسناد و تحلیل احساسات طراحی شده‌اند اما هنوز ساخته نشده‌اند.

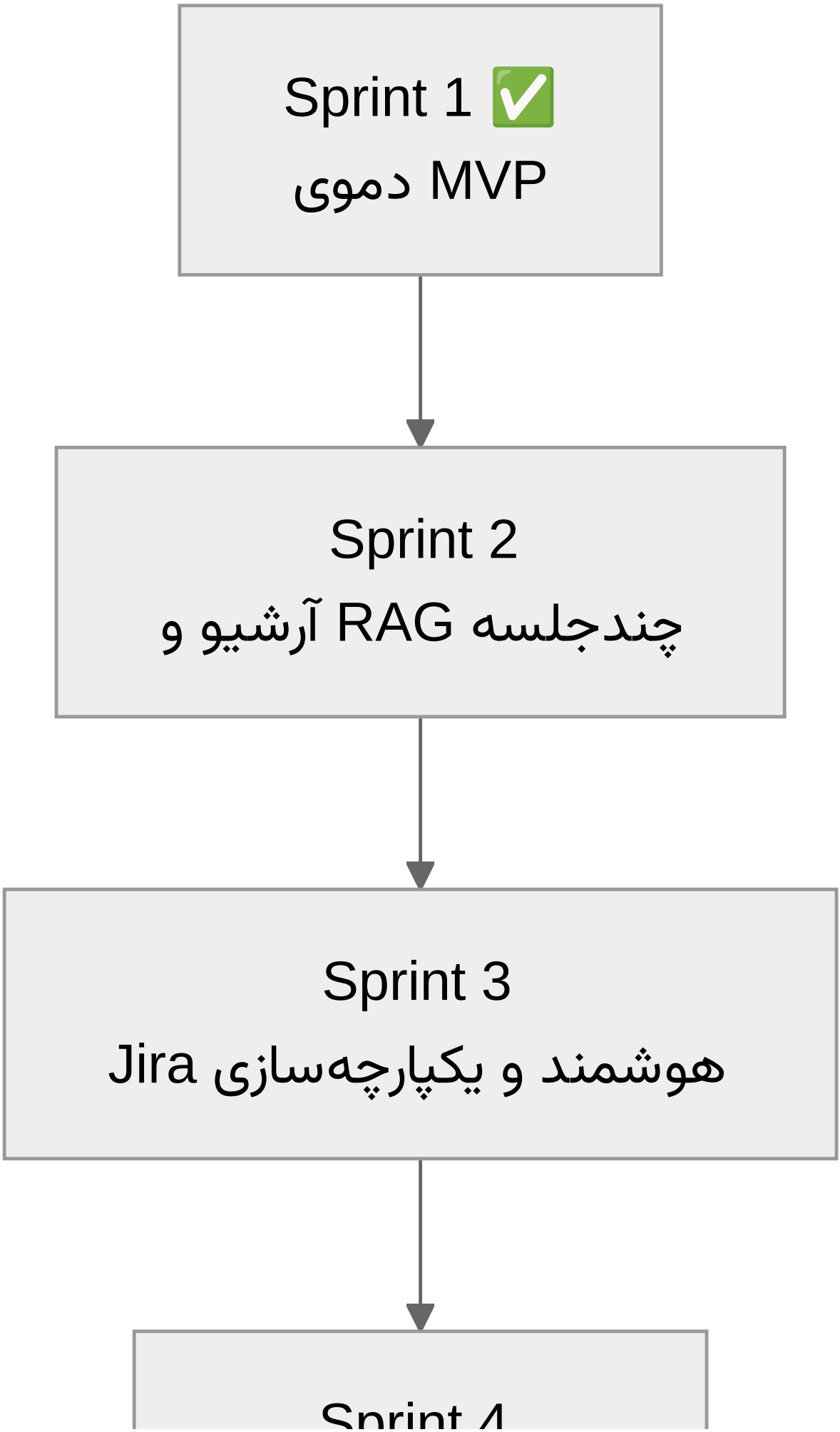
> **برای همیشه خارج از محدوده:** رونویسی زنده و جریانی حین جلسه و diarization حرفه‌ای گویندگان. رونویسی تک‌مرحله‌ای Gemini از صوت آپلود/ضبط‌شده، نیاز محصول را بدون یک خط لوله‌ی جدای STT پوشش می‌دهد.

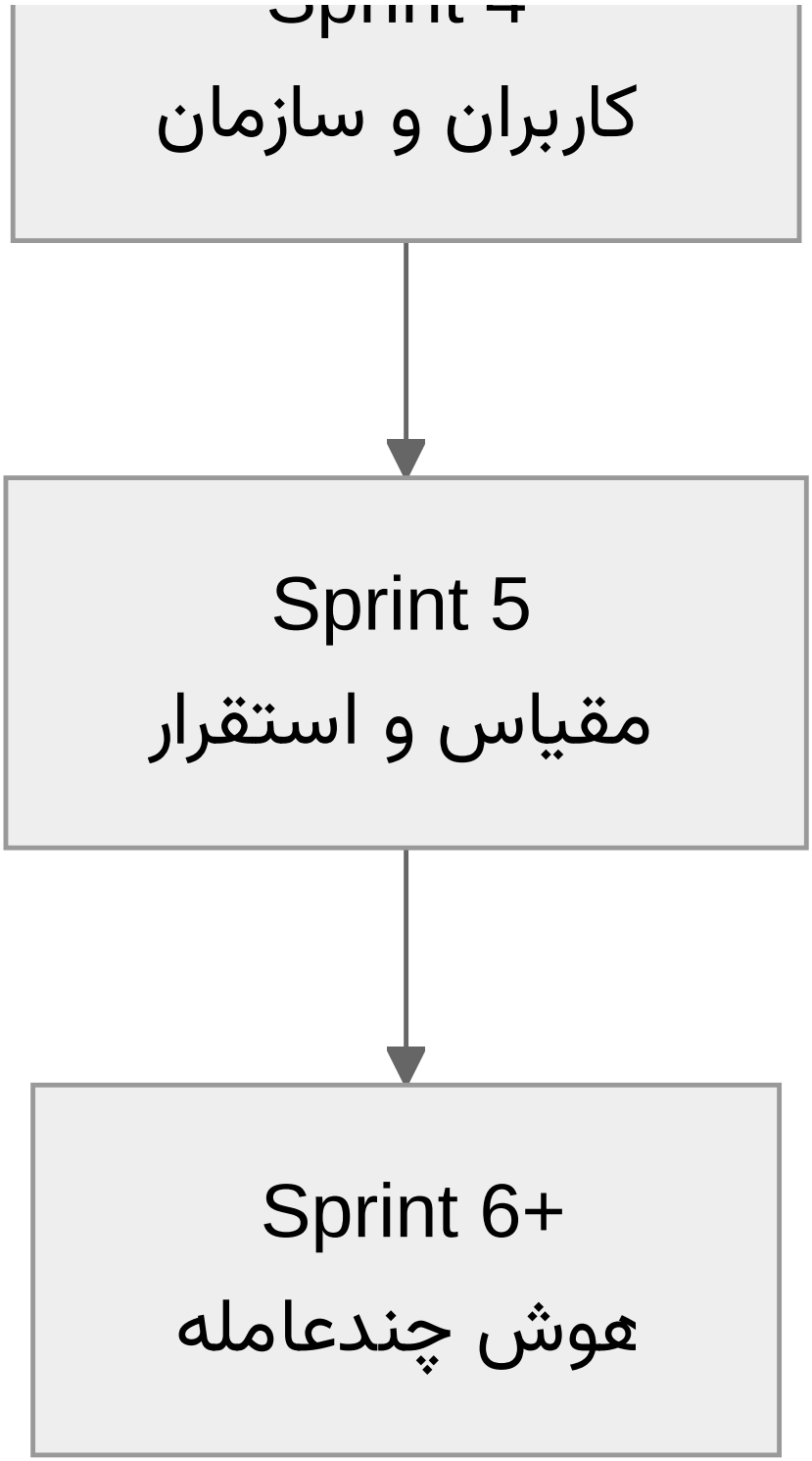
فهرست

1. ۱. نقشه راه در یک نگاه
2. ۲. اسپرینت ۲ — آرشیو و جستجوی سازمانی
3. ۳. اسپرینت ۳ — Jira و گردش کار عملیاتی
4. ۴. اسپرینت ۴ — کاربران، فضای کار و تقویم
5. ۵. اسپرینت ۵ — مقیاس، کیفیت و استقرار
6. ۶. اسپرینت ۶+ — محصول هوشمند سازمانی
7. ۷. قابلیت‌های پیشرفته (متمایزکننده‌ها)
    1. راهنمای برگزارکننده

2. هم‌راستایی SOW / قرارداد
3. تحلیل احساسات
8. ۸. MLOps و چرخه‌ی کیفیت
9. ۹. اولویت‌بندی
10. ۱۰. وابستگی‌های فنی
11. ۱۱. ایده‌های افق‌بلندتر پژوهشی

۱. نقشه راه در یک نگاه


Sprint 1 ✅
MVP دموی
Sprint 2
چندجلسه RAG آرشیو و
Sprint 3
هوشمند و یکپارچه‌سازی Jira
Sprint 4



قوس نقشه‌ی راه یک پیشروی عمدی است: از یک جلسه (MVP) به یک حافظه‌ی سازمانی از جلسات (اسپرینت ۲)، به گردش‌کار عملیاتی واقعی (اسپرینت ۳)، به یک محصول واقعاً چندمستأجره (اسپرینت‌های ۴ و ۵)، و سرانجام به هوش چندعامله‌ی متمایز (اسپرینت ۶+).

۲. اسپرینت ۲ — آرشیو و جستجوی سازمانی

هدف: از «یک جلسه» به «حافظه‌ی سازمانی جلسات» برسیم.

قابلیت	توضیح	پیاده‌سازی پیشنهادی
RAG چندجلسه‌ای	پرسش درباره‌ی چند جلسه با فیلتر تاریخ، پروژه، شرکت‌کننده.	metadata Chroma: `org_id`, `date` , `project` ؛ فیلتر در query.
صفحه آرشیو	لیست جلسات با جستجوی متنی روی عنوان/خلاصه.	SQLite FTS5 روی `summary` , `title` .
برچسب و پروژه	دسته‌بندی جلسات (standup، planning، review).	ستون‌های `project_key` , `tags` در SQLite.
خروجی خلاصه	Export به Markdown / PDF برای اشتراک با مدیر.	قالب Astro یا WeasyPrint از همان `MeetingAnalysis` .
مقایسه جلسات	«تصمیمات sprint قبل در برابر الان» — یک عامل روی چند meeting_id.	`compare_agent` + top-k از هر جلسه.

معیار پذیرش

- سه جلسه ingest شده؛ یک پرسش cross-meeting با citation از حداقل دو جلسه.
- جستجوی آرشیو زیر ۵۰۰ms (تا حدود ۱۰۰ جلسه).
- Export خلاصه‌ی یک جلسه بدون خطای RTL.

۳. اسپرینت ۳ — Jira و گردش کار عملیاتی

هدف: تسک‌ها واقعاً به افراد درست در Jira برسند، نه issueهای ناشناس.

قابلیت	توضیح	فنی
نگاشت assignee	«سارا» در transcript ← یک `accountId` در Jira.	جدول `speaker_jira_map` ؛ UI تنظیمات (آماده شده).

`jira_agent`	پیش‌نمایش، حذف تکراری، پیشنهاد epic/link.	ابزارهای Pydantic AI: `search_existing_issues`, `preview_create`.
OAuth در Jira	جایگزین token شخصی — مناسب تیم.	Atlassian OAuth 2.0؛ ذخیره‌ی refresh token رمزنگاری‌شده.
ویرایش قبل از ارسال	کاربر تسک‌ها را در UI ویرایش کند، بعد bulk create.	state موقت؛ POST فقط پس از تأیید.
وضعیت همگام	نمایش issue key و لینک Jira در صفحه‌ی جلسه.	`jira_keys_json` برای هر تسک.
قالب issue	فیلدهای سفارشی پروژه (story points، component).	map از `JIRA_PROJECT_KEY` + یک config YAML.

۴. اسپرینت ۴ — کاربران، فضای کار و تقویم

هدف: چندکاربره‌ی واقعی و اتصال به ابزارهای روزمره.

قابلیت	توضیح	فنی
احراز هویت	ورود با ایمیل/رمز یا SSO سازمانی.	JWT session؛ یا OIDC (Keycloak / Google Workspace).
فضای کار (workspace)	داده‌ی هر سازمان جدا — جلسات، نقشه، Jira، speaker.	`org_id` روی همه‌ی جداول؛ middleware tenancy.
نقش‌ها (RBAC)	admin · editor · viewer — چه کسی Jira create کند.	جدول `memberships`.
تقویم (متادیتا)	import عنوان/زمان/شرکت‌کنندگان از Google Calendar — transcript همچنان	Calendar API read-only؛ پیش‌پرکردن فرم جلسه.

	paste/upload.	
ایمیل خلاصه	ارسال خودکار خلاصه + لینک جلسه به شرکت‌کنندگان.	SMTP یا Resend/Cloudflare Email؛ قالب HTML RTL.
اعلان Slack/Teams	پیام کوتاه: N تسک جدید + لینک.	Incoming webhook.

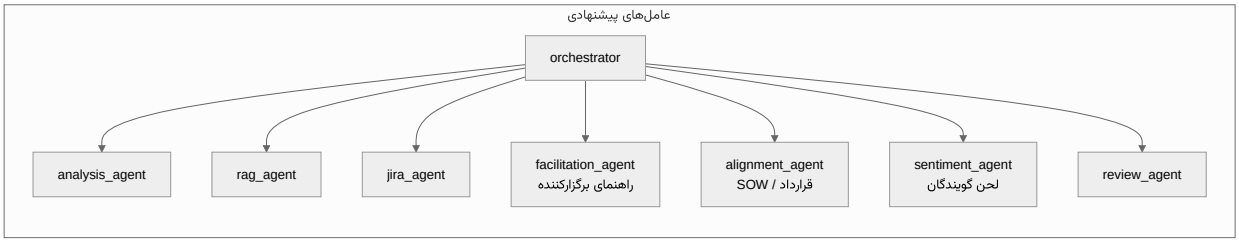
۵. اسپرینت ۵ — مقیاس، کیفیت و استقرار

هدف: آماده برای دموی سازمانی و پایش در production.

قابلیت	توضیح	فنی
PostgreSQL	جایگزین SQLite برای چند worker و backup.	مهاجرت SQLAlchemy + Alembic از schema فعلی.
Vector در مقیاس	هزاران chunk بدون کندی.	pgvector یا Chroma Cloud؛ retention policy.
صف ingest	تحلیل جلسه‌ی طولانی در background.	Redis + worker (Celery / ARQ)؛ polling وضعیت در UI.
Observability	ردیابی latency Gemini و خطای Jira.	OpenTelemetry + log ساختاریافته؛ یک dashboard.
استقرار	Docker Compose؛ اختیاری Cloud Run / VPS.	frontend + API + Postgres در یک compose.
prompt و eval	مجموعه‌ی golden transcript؛ regression کیفیت خلاصه.	pytest + امتیاز LLM-judge یا rubric انسانی.
محدودیت نرخ	جلوگیری از سوءاستفاده از API.	rate limit per org؛ cache embedding.

۶. اسپرینت ۶+ — محصول هوشمند سازمانی

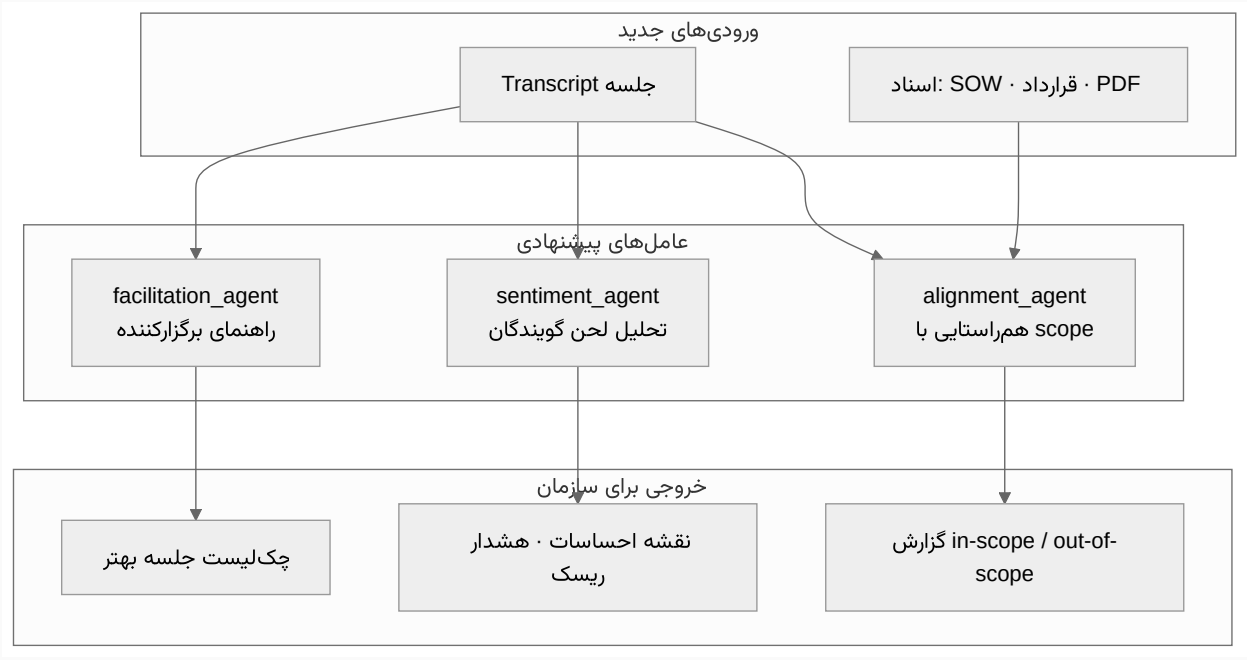
هدف: تمایز رقابتی — فراتر از «خلاصه + Jira».



ایده	ارزش
ردیابی تصمیمات	پیگیری تصمیمات بین جلسات — «آیا تصمیم X اجرا شد؟»
داشبورد action items	همه‌ی تسک‌های باز از همه‌ی جلسات، با فیلتر assignee.
قالب جلسه	prompt مخصوص standup / retro / planning.
تحلیل مشارکت	نسبت صحبت هر speaker؛ هشدار جلسه‌ی یک‌طرفه.
تعارض تصمیم	یک عامل که جلسه‌ی جدید را با تصمیمات قبلی مقایسه می‌کند.
پیشنهاد follow-up	تهیه‌ی خودکار دستور کار بعدی از تسک‌های باز.
Confluence / Notion	انتشار صورتجلسه در wiki سازمان.
نسخه‌گذاری تحلیل	ویرایش انسانی روی خلاصه؛ re-run فقط بخشی از خط لوله.
حریم خصوصی	ماسک PII · retention · export برای GDPR.
on-prem	استقرار داخل شبکه؛ مدل self-hosted اختیاری.

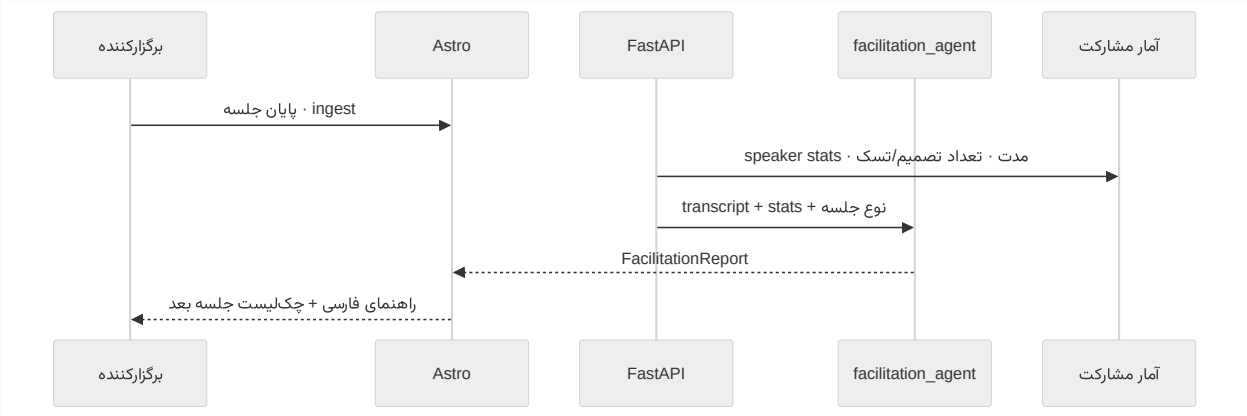
۷. قابلیت‌های پیشرفته (متمایزکننده‌ها)

سه قابلیت، دستیار را از یک یادداشت‌بردار به یک محصول هوش‌جلسه (meeting intelligence) تبدیل می‌کنند. هر کدام یک عامل جدید و در صورت لزوم یک ورودی جدید می‌افزایند.



۷.۱ راهنمای برگزارکننده *(پیاده‌سازی شده)*

پس از تحلیل هر جلسه، سامانه به **برگزارکننده (facilitator)** پیشنهاد عملی و مربی‌گونه می‌دهد: چه چیزی خوب بود، کجا وقت هدر رفت، و جلسه‌ی بعدی چطور کوتاه‌تر و مؤثرتر برگزار شود.



چک‌لیست دستی (امروز — بدون AI)

قبل از جلسه

- ☐ هدف یک‌خطی و خروجی مورد انتظار (تصمیم / تسک / اطلاع‌رسانی) مشخص باشد.
- ☐ دستور کار با زمان تقریبی هر بند (timebox) ارسال شود.
- ☐ فقط افراد ضروری دعوت شوند — نقش هر نفر روشن باشد.
- ☐ اسناد مرجع (SOW، قرارداد) از قبل در دسترس باشد.

حین جلسه

- ☐ تصمیمات را با جمله‌ی «تصمیم گرفتیم که...» و مسئول اعلام کنید.
- ☐ بحث‌های off-topic را به «پارکینگ‌لات» بسپارید.
- ☐ هر ۱۵ تا ۲۰ دقیقه یک خلاصه‌ی ۳۰ ثانیه‌ای از پیشرفت بدهید.

بعد از جلسه

- ☐ خلاصه + تسک‌ها ظرف ۲۴ ساعت منتشر شود.
- ☐ جلسه‌ی follow-up فقط در صورت وجود تسک باز برنامه‌ریزی شود.

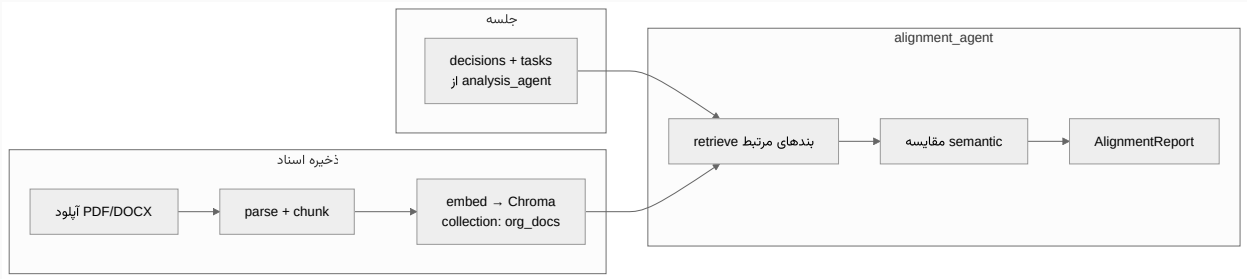
خروجی FacilitationReport

فیلد	توضیح
`what_went_well`	۲ تا ۳ نکته‌ی مثبت (مثلاً تصمیمات شفاف، مشارکت متعادل).
`improvements`	پیشنهاد مشخص (مثلاً «بخش X چهل‌درصد زمان را گرفت بدون تصمیم»).
`next_meeting_agenda`	پیش‌نویس دستور کار جلسه‌ی بعد از روی تسک‌های باز.
`timebox_suggestion`	پیشنهاد مدت کوتاه‌تر یا قالب متفاوت (مثلاً async update).
`coaching_summary`	خلاصه‌ی کوتاه و تشویقی برای برگزارکننده.
`facilitator_score`	امتیاز اختیاری ۱ تا ۵ بر اساس rubric.

`GET /api/meetings/{id}/facilitation` **API:** — پیاده‌سازی شده، با تب «راهنمای جلسه» در UI.

۷.۲ هم‌راستایی SOW / قرارداد *(برنامه‌ریزی‌شده)*

سازمان می‌تواند **Statement of Work**، قرارداد، یا RFP را آپلود کند. پس از هر جلسه، `alignment_agent` تصمیمات و تعهدات جلسه را با بندهای سند مقایسه می‌کند و هر مورد را **in-scope**، **out-of-scope**، یا **نیازمند تأیید حقوقی/مدیر** برچسب می‌زند.



مدل داده‌ی پیشنهادی

موجودیت	فیلدها
`OrgDocument`	`id` , `title` , `type` (sow|contract|rfp), `project_key` , `uploaded_at`
`AlignmentFinding`	`meeting_decision` , `doc_excerpt` , `status` (aligned|out_of_scope|unclear), `confidence` , `doc_id`
`AlignmentReport`	لیست findings + خلاصه‌ی فارسی برای مدیر پروژه.

API پیشنهادی

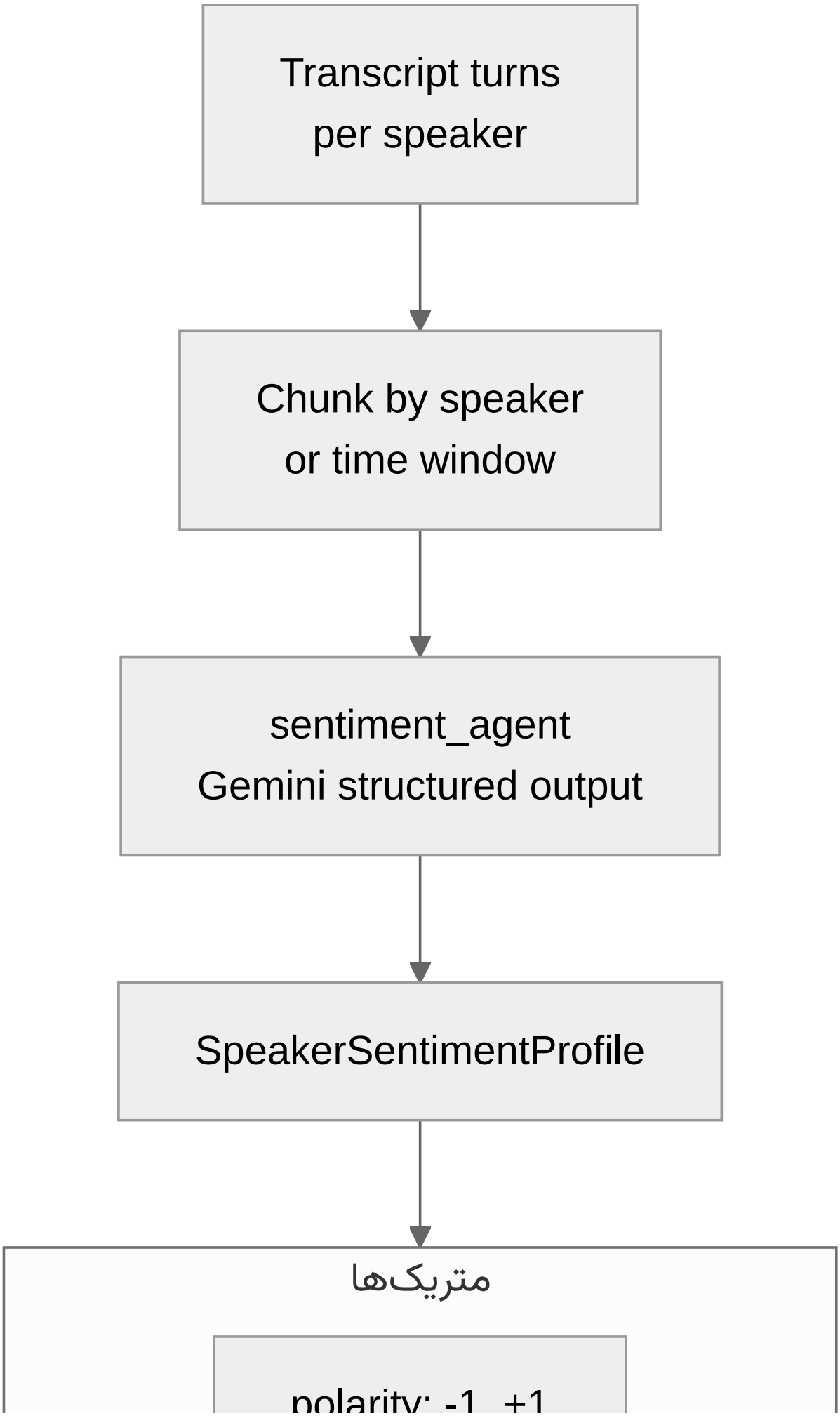
Method	Path	کار
POST	/api/documents	آپلود SOW/قرارداد · index در Chroma.
GET	/api/documents	لیست اسناد پروژه.

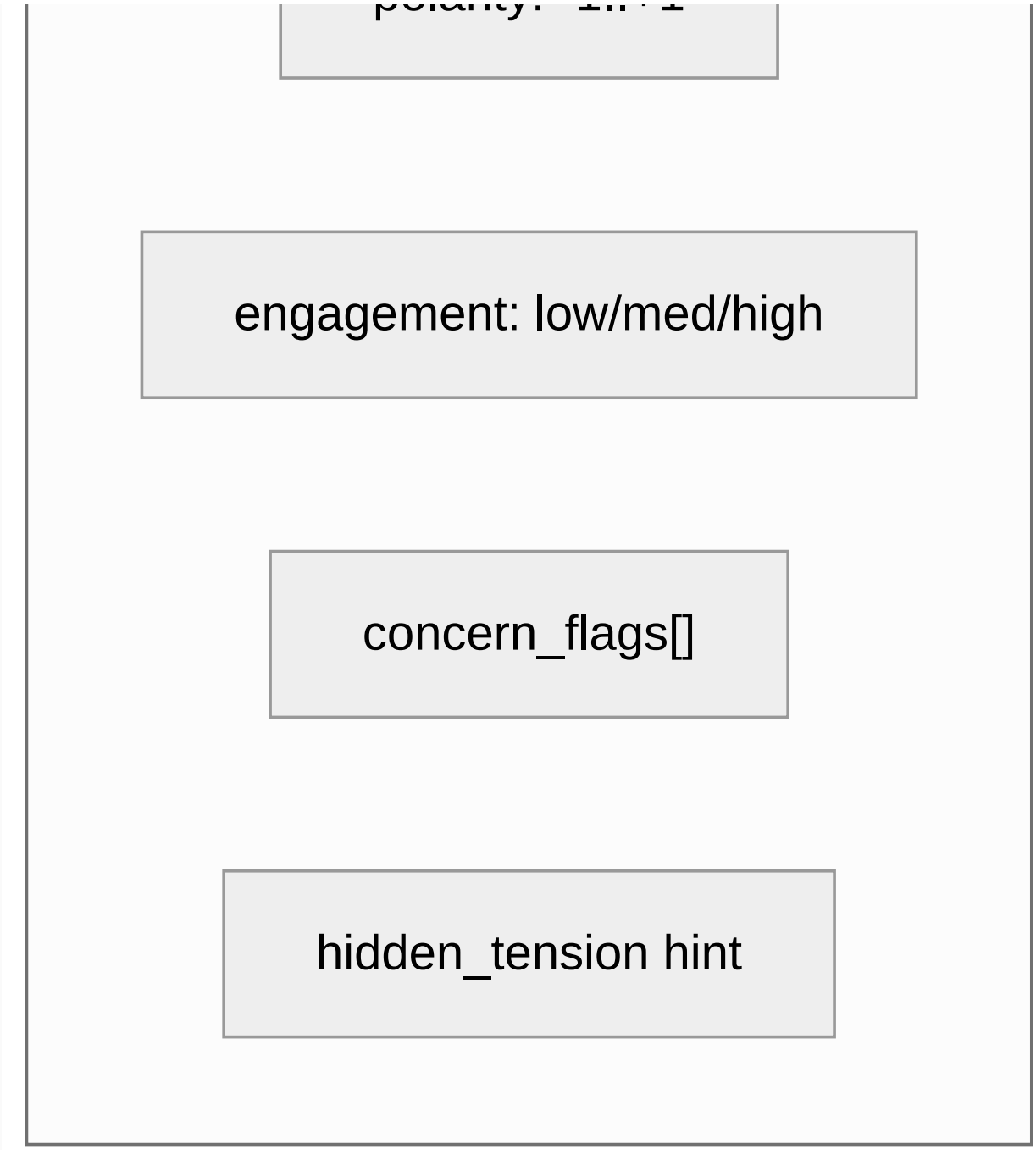
POST	/api/meetings/{id}/alignment	اجرای `alignment_agent` · گزارش مقایسه.

> **هشدار محصول:** خروجی AI جایگزین مشاور حقوقی نیست — برای قراردادهای binding، تأیید انسان الزامی است. UI باید یک disclaimer واضح نشان دهد.

۷.۳ تحلیل احساسات (Sentiment) *(برنامه‌ریزی‌شده)*

روی **متن transcript** (نه صوت خام)، برای هر گوینده و هر بخش جلسه: لحن کلی (مثبت / خنثی / منفی)، **ناهماهنگی** (مثلاً کلمات مثبت با نگرانی پنهان)، و روند احساس در طول جلسه.


Transcript turns
per speaker
Chunk by speaker
or time window
sentiment_agent
Gemini structured output
SpeakerSentimentProfile
متریک‌ها
polarity: -1 ..+1



خروجی SpeakerSentimentProfile

فیلد	معنی
`speaker`	نام گوینده از transcript.
`overall_tone`	positive | neutral | negative | mixed.
`polarity_score`	عددی از -۱ تا +۱.

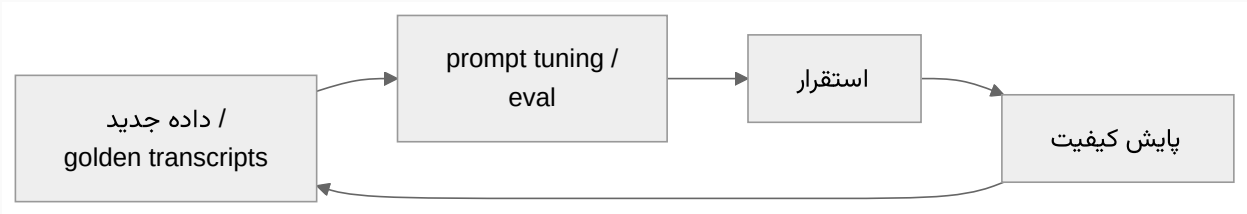
`concern_phrases`	نقل‌قول‌های کوتاهی که نگرانی یا مخالفت احتمالی نشان می‌دهند.
`hidden_dissatisfaction`	bool + توضیح (مثلاً «موافقت ظاهری اما تأکید بر ریسک»).
`timeline`	تغییر لحن در ابتدا / میانه / پایان جلسه.

API پیشنهادی: `GET /api/meetings/{id}/sentiment` — یک تب جدید کنار خلاصه/تسک/RAG.

> **اخلاق و حریم خصوصی:** تحلیل sentiment برای مدیریت تیم حساس است. فقط admin/facilitator باید آن را ببیند؛ باید opt-in سازمانی باشد، بدون برچسب‌گذاری منفی در گزارش‌های عمومی. چارچوب در UI: «سیگنالی برای گفت‌وگوی follow-up»، نه «قضاوت شخصیتی».

۸. MLOps و چرخه‌ی کیفیت

MVP مدلی را بازآموزی نمی‌کند — از یک LLM میزبانی‌شده استفاده می‌کند. بنابراین دغدغه‌ی عملیاتی، آموزش مدل نیست بلکه **تضمین کیفیت promptها و خروجی‌ها در طول زمان** است؛ دقیقاً همان‌جایی که یک محصول AI به‌آرامی و بدون سروصدا دچار افت می‌شود.



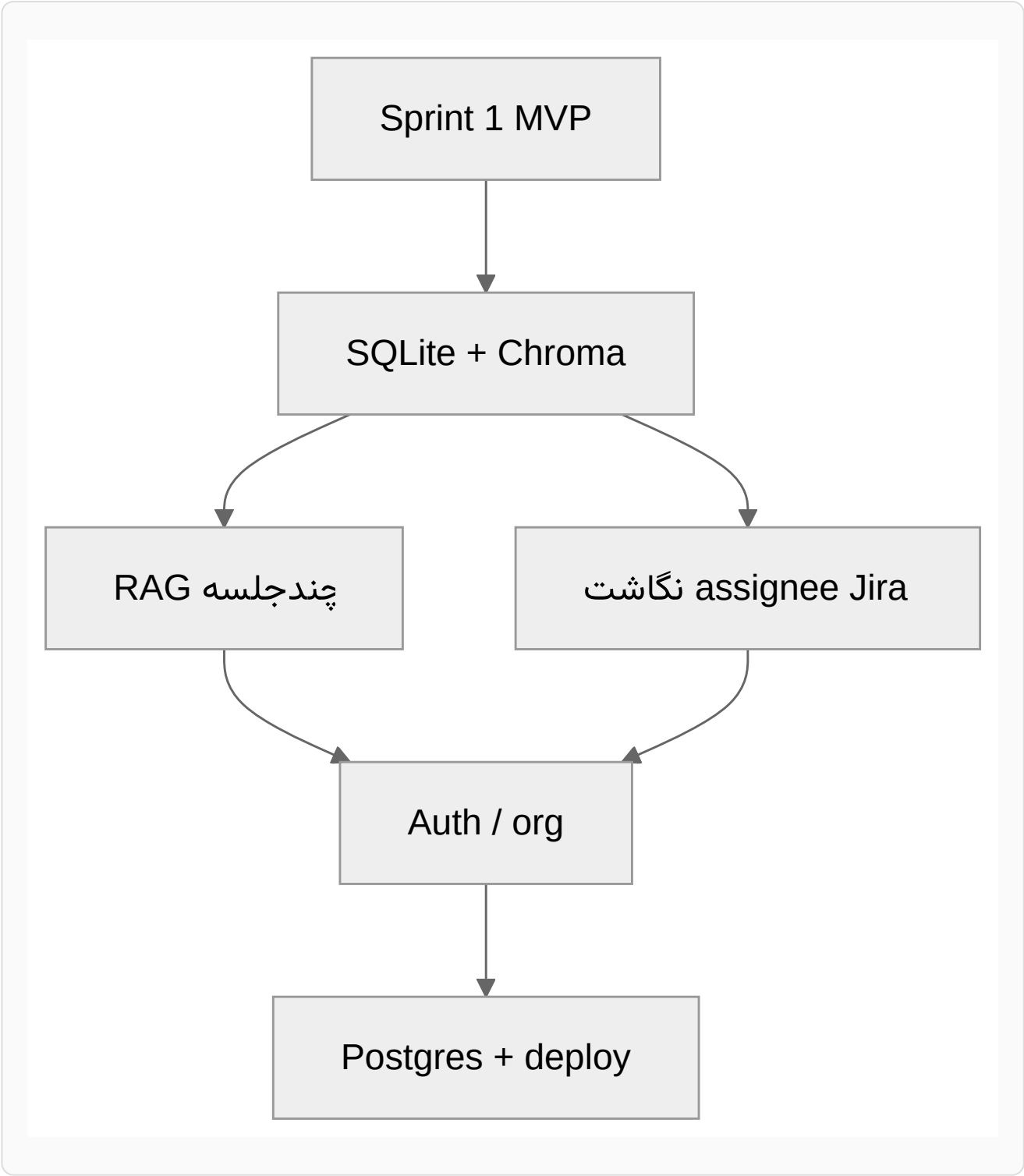
- **مجموعه‌ی golden:** مجموعه‌ای منتخب از transcriptها با خلاصه‌ی مرجع و تسک‌های مورد انتظار.
- **eval خودکار:** امتیازدهی هر نسخه‌ی جدید prompt/مدل در برابر مجموعه‌ی golden با LLM-judge و/یا rubric انسانی، به‌عنوان بخشی از CI.
- **نسخه‌گذاری prompt:** با promptها مثل artifactهای نسخه‌دار رفتار شود تا یک regression را بتوان به یک تغییر مشخص ردیابی کرد.
- **پایش drift:** رصد کیفیت خلاصه، دقت citation در RAG، و نرخ تبدیل Jira در طول زمان، و هشدار در صورت افت.

این کار با اسپرینت ۵ (مقیاس، کیفیت و استقرار) زمان‌بندی شده و همتای صادقانه‌ی نظریه‌ی AIOps/MLOps است: چرخه‌ی عمری که محصول با بلوغ خود اتخاذ می‌کند، به‌صورت واقع‌بینانه و بدون وعده‌ی اغراق‌آمیز در MVP.

۹. اولویت‌بندی

اولویت	مورد	دلیل
P0	RAG چندجلسه + آرشیو FTS	بیشترین ارزش نسبت به MVP.
P0	نگاشت assignee در Jira	درد واقعی تیم‌های عملیاتی.
P1	Auth + workspace	پیش‌نیاز مشتری سازمانی.
P1	ویرایش قبل از create در Jira	اعتماد کاربر به خروجی AI.
P1	راهنمای برگزارکننده (انجام‌شده) + مربی‌گری عمیق‌تر	تمایز فوری با infra سبک.
P1	هم‌راستایی SOW/قرارداد	ارزش برای PM و قراردادهای fixed-scope.
P2	PostgreSQL + queue	وقتی ترافیک از حد دمو فراتر رود.
P2	متادیتای تقویم + ایمیل خلاصه	کاهش اصطکاک ورود داده.
P2	تحلیل sentiment	فقط پس از RBAC و opt-in اخلاقی.
P3	multi-agent · Confluence	تمایز بلندمدت.

۱۰. وابستگی‌های فنی



گراف وابستگی، ترتیب اسپرینت‌ها را توضیح می‌دهد: پایه‌ی ذخیره‌سازی هم RAG چندجلسه و هم نگاشت assignee را ممکن می‌کند؛ هر دوی این‌ها چندمستأجری (auth/org) را برمی‌انگیزند؛ و بار واقعی مستأجران است که سرانجام PostgreSQL و یک استقرار production را توجیه می‌کند.

۱۱. ایده‌های افق‌بلندتر پژوهشی

فراتر از کار محصولیِ برنامه‌ریزی‌شده، چند مسیر می‌توانند پایه‌ی پژوهش یا توسعه‌ی پایان‌نامه باشند:

- **گراف تصمیمات بین‌جلسه‌ای.** مدل‌کردن تصمیمات به‌صورت گره و تشخیص تناقض، بازگشت، و تعهدات اجرانشده در طول تاریخچه‌ی جلسات یک سازمان.
- **مباحثه‌ی چندعامله برای خلاصه‌ها.** استفاده از یک عامل منتقد/بازبین برای به‌چالش‌کشیدن خلاصه‌ی نوبت اول، و سنجش اینکه آیا بازبینی خصمانه وفاداری واقعی را بهبود می‌دهد.
- **Chunking معنایی و گوینده‌آگاه.** مقایسه‌ی chunking مبتنی بر نوبت با chunking معنایی و گوینده‌آگاه از نظر دقت بازیابی روی transcriptهای فارسی.
- **ارزیابی وفاداری برای RAG فارسی.** ساخت یک harness ارزیابی برای دقت citation و نرخ hallucination، ویژه‌ی داده‌ی جلساتِ زبانِ کم‌منبع.
- **تحلیل حافظ حریم خصوصی.** خط لوله‌های on-device یا با PII ماسک‌شده تا جلسات حساس بدون خروج از شبکه‌ی سازمان تحلیل شوند.
- **کیفیت با انسان در حلقه.** ثبت ویرایش‌های انسانی روی خلاصه و تسک به‌عنوان سیگنال بازخورد برای بهبود prompt و انطباق per-organization.

مرتبط: شرح پروژه و چشم‌انداز · پیاده‌سازی فنی.