تصویر کلی پروژه: دقیقاً قراره چه چیزی بسازی؟

تو این پروژه، شما یک دیتاست واقعی‌نما از ۵۰ هزار متخصص فنی دارید و باید پیش‌بینی کنید چه کسی احتمالاً مهاجرت کاری می‌کند و چه کسی نه.

اما نکته مهم اینجاست:
نمره فقط برای ساختن یک مدل با عدد خوب نیست.
نمره اصلی برای اینه که نشان بدی:

1. مسئله را درست فهمیدی و درست تعریف کردی
2. داده را علمی و تمیز آماده کردی
3. مدل‌ها را درست مقایسه کردی (نه سلیقه‌ای)
4. نتیجه را قابل توضیح و قابل دفاع کردی
5. به عدالت، خطا، ریسک و استفاده واقعی فکر کردی

یعنی در نهایت باید یک «سیستم تصمیم‌یار قابل اتکا» تحویل بدی، نه صرفاً یک فایل کد.

خروجی‌هایی که باید تحویل بدهی (و چرا مهم‌اند)

1) نوت‌بوک اصلی (Q1 تا Q20 + Capstone)

این نوت‌بوک قلب پروژه است. هر سوال باید یک بخش مجزا داشته باشد.
کسی که نوت‌بوک را باز می‌کند باید بتواند مرحله‌به‌مرحله بفهمد:

- سؤال چی بوده
- شما چه روشی انتخاب کردید
- نتیجه چه شده
- چرا این نتیجه منطقی است

2) گزارش نهایی (حداکثر ۲۵ صفحه)

این برای «داستان علمی پروژه» است.
یعنی حتی اگر کسی کد را نبیند، از روی گزارش بفهمد کارتان چقدر درست و حرفه‌ای بوده.

3) بسته کد

کد باید ماژولار و قابل اجرا باشد.
یعنی با یک دستور مشخص (یا چند دستور واضح) پروژه اجرا شود.
اگر کسی مجبور شود با دست فایل‌ها را تغییر دهد، کیفیت مهندسی پایین حساب می‌شود.

4) پاسخ تشریحی سوالات

برای سوال‌های مفهومی و ریاضی (مثل Elastic Net یا استنباط) باید توضیح تشریحی روشن بدهی، نه فقط خروجی عددی.

5) خلاصه مدیریتی (1–2 صفحه)

برای مخاطب غیر فنی:
«مدل چه می‌گوید؟ چقدر قابل اعتماد است؟ کجا باید احتیاط کنیم؟»

حداقل استاندارد فنی (چیزهایی که اگر رعایت نکنی، پروژه از نظر حرفه‌ای ناقص است)

Seed ثابت

اگر امروز و فردا نتایجت فرق کند، بازتولیدپذیر نیست.
پس همه‌جا seed مشخص داشته باش.

تفکیک train/validation/test

این بخش خیلی مهمه.
اگر مرز این‌ها قاطی شود، نتیجه‌ها بیش‌برآورد می‌شوند و در واقعیت خراب می‌شوند.

کنترل نشت داده (Data Leakage)

خیلی از پروژه‌ها همینجا نابود می‌شوند.
هر فیچری که بعد از زمان تصمیم‌گیری به وجود آمده باشد نباید وارد مدل شود.

ثبت محیط اجرا

نسخه پایتون و پکیج‌ها باید مشخص باشد تا خروجی شما در سیستم دیگر هم تکرار شود.

توضیح ساده و دقیق هر بلوک

بلوک A — مبانی

Q1) صورت‌بندی مسئله و چرخه عمر داده‌محور

اینجا باید نشان بدهی می‌فهمی «مدل» فقط یک قطعه از کل سیستم است.

باید روشن کنی:

- دقیقاً چه تصمیمی می‌خواهیم با مدل پشتیبانی کنیم؟
  (مثلاً شناسایی افراد با ریسک مهاجرت بالا برای برنامه نگهداشت)

- موفقیت را با چه معیارهایی می‌سنجیم؟
  (فقط Accuracy کافی نیست؛ AUC، F1، Calibration مهم‌اند)

- چه ریسک‌هایی داریم؟
  (نشت داده، تغییر رفتار کاربران، تغییر سیاست مهاجرت، خطای لیبل)

- بعد از استقرار چه می‌شود؟
  (چه چیزی را پایش می‌کنیم؟ هر چند وقت؟ چه زمانی بازآموزی؟)

خروجی خوب: یک دیاگرام چرخه عمر + جدول ریسک‌ها و راه‌حل.

Q2) کیفیت داده و EDA

این بخش یعنی قبل از مدل‌سازی باید «چشم باز» روی داده داشته باشی.

چه کارهایی؟

- نوع ستون‌ها، مقادیر گمشده، تکراری‌ها، مقادیر غیرمنطقی

- پرت‌ها را با حداقل دو روش پیدا کن (IQR و z-score مثلاً)

- حداقل ۸ نمودار معنادار بکش (نه صرفاً برای پر کردن)

- یک تابع preprocessing قابل استفاده مجدد بساز (و تست کن)

چرا مهم است؟
چون اگر داده را نفهمی، هر مدلی هم بسازی در بهترین حالت عدد ظاهری خوب می‌دهد، نه نتیجه قابل اعتماد.

بلوک B — استنباط آماری و روایت بصری

Q3) استنباط

اینجا باید نشان بدهی که فرق همبستگی، معناداری آماری و علیت را می‌فهمی.

- آیا مطالعه مشاهده‌ای است یا آزمایشی؟
- یک CI معتبر بده (و درست تفسیر کن)
- یک آزمون فرض کامل انجام بده (H0/H1، سطح معنی‌داری، نتیجه)
- پیش‌فرض‌های آزمون را بررسی کن

نکته مهم:
p-value کوچک ≠ اثر بزرگ یا علت قطعی.
فقط می‌گوید داده با فرض صفر سازگار نیست.

Q4) مصورسازی برای تصمیم‌گیر

این بخش درباره «انتقال درست معنا» است.

- یک داشبورد روایت‌دار بساز: KPIها واضح باشند
- رنگ، مقیاس، ترتیب نمودارها دلیل داشته باشد
- یک نمودار گمراه‌کننده نشان بده و نسخه درستش را ارائه کن

مثال خطای رایج:
محور y بریده شده و اختلاف را مصنوعی بزرگ نشان می‌دهد.

بلوک SQL — C و مهندسی داده

SQL (Q5) پیشرفته

باید نشان بدهی query تحلیلی بلد هستی، نه فقط select ساده.

- moving average سه‌ساله با window function
  رتبه‌بندی/دهک‌بندی با rank/ntile
- cohort analysis با CTE

اینجا کیفیت تحلیل زمانی و ساختار query خیلی مهم است.

Q6) نشت داده + معماری مقیاس‌پذیر

دو چیز همزمان می‌خواهند:

1. منطق علمی: حذف فیچرهای نشتی با دید زمانی
2. منطق مهندسی: طراحی معماری داده (Bronze/Silver/Gold + Feature Store)

باید توضیح بدهی:

- داده خام کجاست؟
- داده تمیز کجاست؟

- فیچر نهایی برای مدل کجاست؟
- چطور مطمئن می‌شوی همان فیچری که در train بوده در serving هم همان است؟

بلوک D — مدل‌سازی نظارت‌شده و بهینه‌سازی

Q7) مدل‌های خطی/لجستیک + Elastic Net

هدف اینه که از پایه علمی دور نشی.

- یک baseline معقول بساز
- تابع هزینه Elastic Net را بنویس و توضیح بده
- ضرایب را تفسیر کن (علامت، اندازه، CI، p-value)
- درباره پایداری ضرایب صحبت کن

چرا مهم؟
چون baseline قوی و قابل تفسیر بهترین نقطه مرجع برای مدل‌های پیچیده‌تر است.

Q8) مقایسه SGD / Momentum / Adam روی ravine

اینجا می‌خواهند بفهمند optimization را واقعاً می‌فهمی.

- مسیر همگرایی را نشان بده

- نوسان در راستای شیب تند را توضیح بده
- بگو برای ناهمگنی مقیاس فیچرها کدام optimizer مناسب‌تر است و چرا

Q9) مقایسه خانواده مدل‌ها

اینجا باید «منصفانه» مقایسه کنی:

- SVM/KNN
- Tree/RF
- Boosting (ترجیحاً XGBoost)

و الزاماً:

- CV درست
- tuning درست
- تحلیل خطا (فقط جدول عدد کافی نیست)

نکته حرفه‌ای:
مدلی بهتر است که علاوه بر عملکرد، پایداری و تفسیر و هزینه محاسباتی مناسبی هم داشته باشد.

بلوک E — بدون نظارت

Q10) کاهش بُعد

- PCA را درست انجام بده و EVR را توضیح بده
- یک روش دیگر (t-SNE/UMAP/...) اضافه کن
- در مورد معنی فضای نهفته با احتیاط حرف بزن (خصوصاً t-SNE)

Q11) خوشه‌بندی

- KMeans + Elbow + Silhouette
- DBSCAN هم اجرا کن
- پایداری خوشه‌ها را بررسی کن

هدف:
فقط عدد خوشه ندهی؛ باید بگویی خوشه‌ها در دنیای واقعی چه معنایی دارند.

بلوک F — Deep Learning, NLP, LLM

Q12) یک مدل عصبی جدولی + یک مدل توالی/متن

می‌خواهند ببینند مدل عمیق را کورکورانه استفاده نمی‌کنی.

- MLP روی جدول

- یک RNN/LSTM/GRU/CNN روی داده توالی/متن
- مقایسه واقعی با baseline کلاسیک

نتیجه‌گیری خوب:
اگر مدل کلاسیک بهتر بود، باید صادقانه بگویی و دلیلش را تحلیل کنی.

Q13) طراحی LLM Agent

فقط «استفاده از LLM» نیست، طراحی فرایند است:

plan → retrieve → reason → verify

باید تعریف کنی:

- چطور خطای hallucination را کم می‌کنی؟
- معیار ارزیابی کیفیت چیست؟
- چه محدودیت امنیتی/دسترسی برای ابزارها می‌گذاری؟

بلوک G — عدالت و حاکمیت

Q14) عدالت و سوگیری

اینجا خیلی مهم است چون خروجی مدل روی آدم‌ها اثر دارد.

- تحلیل زیرگروهی انجام بده (کشور، تحصیلات، سابقه)

- تبعیض غیرمستقیم (proxy discrimination) را بررسی کن
- روند human-in-the-loop تعریف کن
- مسیر اعتراض/بازبینی تصمیم داشته باش

پیام کلیدی:
مدل دقیق ولی ناعادلانه، برای تصمیم‌گیری واقعی قابل قبول نیست.

بلوک H — Capstone (یکپارچه‌سازی)

اینجا همه چیز باید به هم وصل شود:

1. pipeline کامل و leakage-safe
2. model card کامل
3. SHAP محلی + سراسری
4. جدول عدالت + توصیه استقرار + برنامه پایش

این بخش نشان می‌دهد شما فقط «مدل‌ساز» نیستی، بلکه «طراح سیستم» هستی.

بلوک I — پایش تولید و تحلیل پیشرفته (Q15–Q20)

Q15) Calibration + Threshold Policy

- منحنی کالیبراسیون
- Brier یا ECE
- دو آستانه:
  - یکی برای بهترین F1
  - یکی بر اساس هزینه کسب‌وکاری (FN vs FP)

هدف: تصمیم‌گیری بهتر، نه صرفاً خروجی احتمال خام.

Q16) Drift Detection

- پنجره مرجع و جاری بساز
- PSI برای عددی‌ها
- JS divergence برای دسته‌ای‌ها
- آستانه‌های هشدار/بحرانی و trigger بازآموزی تعریف کن

Q17) Counterfactual Recourse

- برای موارد نزدیک مرز تصمیم، کمترین تغییر لازم را پیدا کن
- فقط تغییرهای واقع‌بینانه را پیشنهاد بده

- نرخ موفقیت recourse و هزینه تغییر را گزارش کن
- بعد اخلاقی‌اش را هم تحلیل کن

Q18) Temporal Validation

- ارزیابی زمانی غلطان انجام بده
- افت عملکرد در زمان را بسنج
- ارتباط با drift را بررسی کن

اگر ستون زمان نداری: fallback منطقی و مستند بده.

Q19) Uncertainty Quantification

- روش conformal یا مشابه
- پوشش تجربی در چند سطح اطمینان
- تحلیل پهنا و under-coverage

هدف: بدانی مدل کجا «مطمئن نیست» تا آن موارد به انسان ارجاع شوند.

Q20) Fairness Mitigation

- fairness مدل پایه را اندازه بگیر
- یک روش مداخله اجرا کن (مثل reweighing)
- قبل/بعد را مقایسه کن
- با قید سیاستی تصمیم نهایی بده (قبول/رد)

کجاها معمولاً نمره کم می‌شود؟

1. نشت داده کشف نشده یا بی‌توضیح
2. CV یا split نادرست
3. تفسیر اشتباه p-value یا CI
4. نبود تحلیل عدالت
5. نداشتن مستندات بازتولید (نسخه پکیج، seed، دستور اجرا)

اصول اخلاقی که باید رعایت شود

- هر منبع بیرونی باید ارجاع داشته باشد
- نتیجه بد یا منفی را حذف نکن

- اگر از مدل زبانی کمک گرفتی، شفاف بگو کجا و چطور

جمع‌بندی خیلی ساده

این پروژه یعنی ساختن یک سامانه تصمیم‌یار که:

- **فنی درست** باشد
- **آماری درست** باشد
- **اخلاقی قابل دفاع** باشد
- **مهندسی‌اش قابل اجرا در عمل** باشد