

اسپراکسا چیه؟

اسپراکسا یک پلتفرم آنلاین تمرین «مکالمه واقعی» برای مهاجران و نوآموزان زبان در فنلاند است؛
نه مثل دولینگو و کتاب زبان که لغت و گرامر یاد می‌دهند، بلکه جایی که کاربر وارد یک سناپریوی واقعی می‌شود و با یک «پارتner مکالمه هوش مصنوعی» تمرین می‌کند؛ مثل:

- صحبت در Kela برای پرسیدن درباره‌ی یک کمک‌هزینه
- تماس برای گرفتن نوبت پزشک
- مکالمه با مدیر یا همکار سر کار
- تمرین مصاحبه شغلی
- صحبت با معلم مدرسه‌ی بچه

هدف اسپراکسا:

کمک به کاربر تا اعتماد به نفس مکالمه پیدا کند، نه این‌که فقط لغت بلد باشد.
کاربر در یک محیط امن، بدون خجالت، هی سناپریوها را تکرار می‌کند، اشتباه می‌کند، اصلاح می‌شود و کمک راحت حرف می‌زند.

هسته محصول:

- وب‌اپ(Browser-based)، بدون نیاز به نصب
- کاربر سناپری انتخاب می‌کند، وارد مکالمه می‌شود
- از میکروفن (یا در نسخه ساده‌تر اول: چت متنی) استفاده می‌کند
- سیستم جواب می‌دهد، مکالمه را ادامه می‌دهد و در پایان:
 - ۲-۳ نکته کوتاه به کاربر می‌دهد (مثلاً یک جمله مودبانه‌تر، یک عبارت طبیعی‌تر)
 - احساس «تشویق» ایجاد می‌کند، نه نمره قرمز و خطأ

پلتفرم برای چه کسانی است؟

- مهاجران در دوره‌های اینتگریشن و کلاس زبان
- Job seeker که مصاحبه و مکالمه کاری می‌خواهد
- والدین مهاجر که باید با مدرسه، پزشک، جامعه صحبت کنند
- دانشجویان بین‌المللی که می‌خواهند با محیط دانشگاه و کار پاره‌هفت وفق پیدا کنند

۲. دستورالعمل برای برنامه‌نویس) بریف فنی MVP اسپراکسا(

۱. کلیات فنی

- نوع محصول: وب‌اپ ریسپانسیو (روی موبایل، تبلت، لیپتاپ)
- نسخه اول MVP: با تمرکز روی یک زبان (فنلاندی) و یک نوع کاربر(Learner)
- معماری:

(React / Vue / Next ...Frontend: ○ هر فریمورک مدرن)

AI / Speech-to-Backend: ○
Node / Python (هـر زبان معمول ...) کـه بـتواند به راحـتی به سـرویسـهـای-Text متـصل شـود

الـزـامـی: •

- سـیـسـتـمـ حـسـابـ کـارـبـرـیـ سـادـهـ (ثـبـتـنـامـ، لـاـگـینـ، فـرـامـوـشـیـ رـمـزـ)
- مدـیرـیـتـ سنـارـیـوـ (مـعـلـمـ / اـدـمـینـ بـتـوـانـدـ سـنـارـیـوـ تـعـرـیـفـ وـ وـیرـایـشـ کـنـدـ)
- ذـخـيـرـهـ پـيـشـرـفتـ کـارـبـرـ (کـدـامـ سـنـارـیـوـهـاـ، چـنـدـ بـارـ، سـطـحـ اـعـتمـادـ بـهـ نـفـسـ)

۲.۲. نقشـهـایـ کـارـبـرـیـ (User Roles)

حدـاقـلـ درـ نـسـخـهـ اـوـلـ:

۱. کـارـبـرـ عـادـیـ (Learner)

- ثـبـتـنـامـ وـ وـرـوـدـ
- اـنـتـخـابـ سـنـارـیـوـ
- اـجـرـایـ مـكـالـمـهـ
- دـيـدـنـ فيـدـبـكـ وـ اـمـتـياـزـ اـعـتمـادـ بـهـ نـفـسـ /ـ پـيـشـرـفتـ

۲. Admin / Content Designer (ادـمـينـ سـنـارـیـوـ)

- لـاـگـينـ اـدـمـينـ
- سـاخـتـنـ /ـ وـيرـايـشـ /ـ حـذـفـ سـنـارـيـوـهاـ
- دـسـتـبـنـدـيـ سـنـارـيـوـهاـ (خـدـمـاتـ دـوـلـتـيـ، مـحـلـ کـارـ، مـدـرـسـهـ، مـصـاحـبـهـ شـغـلـيـ، ...)
- تعـيـينـ سـطـحـ سـخـتـىـ (A1-B2)

درـ فـازـ بـعـدـ مـیـتـوانـ Teacherـ جـداـ تـعـرـیـفـ کـرـدـ، فـعـلـاـ اـدـمـينـ کـافـیـ استـ.

۲.۳. ماـژـوـلـهـایـ اـصـلـیـ (MVP)

۱. ماـژـوـلـ اـحـراـزـ هـويـتـ (Auth)

ثـبـتـنـامـ باـ:

- اـیـمـیـلـ +ـ پـسـورـدـ (سـادـهـ)
- لـاـگـينـ /ـ لـاـگـاـوتـ
- اـمـكـانـ Reset Password
- ذـخـيـرـهـ سـادـهـ پـرـوـفـاـيلـ کـارـبـرـ
- نـامـ پـاـكـ

- سطح تقریبی زبان – (A1, A2, B1, B2) کاربر انتخاب می‌کند

۲.۳.۲. داشبورد کاربر (Learner Dashboard)

وقتی کاربر وارد می‌شود، این‌ها را ببیند:

- خوش‌آمدگویی کوتاه
- سطح کاربر (مثلًاً A2 – “Early Integration Learner”)
- لیست سناریوهای پیشنهادی:
 - بر اساس سطح زبان
 - بر اساس دسته‌بندی (Kela) ، کار، مدرسه، مصاحبه
- چند شاخص ساده:
 - تعداد سناریوهای تکمیل شده
 - آخرین سناریوی تمرین شده
 - نمودار خیلی ساده از تعداد سشن‌ها در ۷ روز اخیر (اگر وقت شد، اگر نه، فقط عدد)

۲.۳.۳. صفحه انتخاب سناریو (Scenario List & Detail)

Scenario List:

فیلترها:

- سطح: A1, A2, B1, B2
- دسته‌بندی: Civic / Services Workplace School & Family Job Interview
- هر سناریو باید این اطلاعات را نمایش دهد:
 - عنوان (مثلًاً: “پرسیدن درباره کمک‌هزینه بچه در Kela”)
 - سطح CEFR
 - مدت تخمینی (مثل ۳-۵ دقیقه)

Scenario Detail:

توضیح کوتاه:

- «کجا هستی؟) «مثلاً(Kela office
 - «هدف مکالمه چیست؟) «مثلاً: پرسیدن درباره eligibility
 - دکمه: "شروع تمرین"
-

۲.۳.۴. محیط مکالمه (Conversation UI)

این مهمترین بخش پروژه است.

الزامات: UI

- نمایش زمینه (Context) سناریو در بالای صفحه) مثلاً: «شما در صفحه Kela هستید، الان نوبت شماست.».
- یک پنل مکالمه (چتمنند) که شامل:

 - پیام «سیستم» نقش مأمور Kela ، مدیر، معلم و (...)
 - پیام‌های «کاربر»

- روش ورودی کاربر:

MVP پیشنهادی:

- حتماً حالت متى (Text input) را پیاده‌سازی کنید تا MVP بدون وابستگی به کیفیت STT هم کار کند.
- امکان اضافه کردن ورودی صوتی (میکروفون Speech-to-Text + به عنوان مرحله دوم.
- یک دکمه "ارسال" / "Send" / "برای متن")
- اگر صوت اضافه شد:

 - دکمه "شروع ضبط" / "Stop"

الزامات منطقی:

- هر سناریو به صورت یک گراف مکالمه prompt + AI / state machine / ساده یا یک مدل "AI" پیاده‌سازی شود:

- سیستم اولین جمله را می‌گوید
- کاربر پاسخ می‌دهد (متن یا متن تبدیل شده از صدا)
- جمله کاربر را به مازول «موتور مکالمه» می‌فرستد Backend
- موتور مکالمه بر اساس:

 - متن کاربر
 - وضعیت فعلی سناریو
 - هدف سناریو

→ یک پاسخ مناسب و کوتاه تولید می‌کند + سیگنال برای "پایان / ادامه مکالمه"

نکته مهم:

لازم نیست در این مرحله ساختار AI را خیلی پیچیده کنید. می‌توان:

- یا از یک / LLM سرویس هوش مصنوعی بیرونی استفاده کرد که با Prompt مهندسی شده مکالمه را مدیریت کند
- یا برای آغاز کار، یک مدل سادهتر / rule-based سکوی آماده مکالمه (Dialogflow, Rasa) و (...) استفاده شود، به شرطی که خروجی را در فرمت استاندارد برگرداند.

۲.۳.۵. مأذول فیدبک (Feedback Engine)

هدف فیدبک: تشویق + ۲-۳ پیشنهاد مشخص، نه نمره قرمز.

ورودی مأذول فیدبک:

- متن آخرین پاسخ کاربر
- خلاصه سناریو و مرحله فعلی
- تاریخچه کوتاه مکالمه Optional:

خروجی پیشنهادی به صورت JSON:

```
{  
    "confidence_score": 0.0-1.0"  
    , "clarity_score": 0.0-1.0"  
    , "comment": "پیام تشویقی کوتاه",  
    "tips": [  
        "یک جمله مودبانه‌تر پیشنهادی",  
        "یک عبارت طبیعی‌تر یا ساختار بهتر",  
        "نسخه پیشنهادی جمله کاربر",  
        "": "suggested_phrase"  
    ]  
}
```

نمایش در UI:

- یک باکس فیدبک زیر مکالمه یا در پایان سناریو:
- یک متن تشویقی (مثلًا: «خیلی خوب توضیح دادی.»)
- ۱-۲ نکته پیشنهادی
- یک جمله پیشنهادی با امکان کلیک برای شنیدن (در آینده)

۲.۳.۶. اتمام سناریو و ذخیره پیشرفت

در پایان هر سناریو:

- نمایش خلاصه:
 - "این سناریو را کامل کردی".
 - "نکته‌های اصلی که امروز پاد گرفتی" (۲-۳ bullet)
- از کاربر بخواهیم یک Self-rating بدده:
 - سؤال: «بعد از این تمرین، چقدر برای این موقعیت احساس اطمینان می‌کنی؟ (۱ تا ۵)»
 - ذخیره در دیتابیس:
 - user_id
 - scenario_id
 - زمان شروع / پایان
 - self_confidence_rating
 - چند متریک ساده (طول مکالمه، تعداد نوبت‌ها)

۴. پنل ادمین / سناریو دیزاینر (Admin Panel)

ویژگی‌های ضروری:

۱. لیست سناریوها
 - عنوان، دسته، سطح، وضعیت (Active / Draft)
۲. فرم ساخت سناریو (Scenario Editor) با فیلهای حداقی:
 - عنوان
 - دسته (Service / Work / School / Interview / Other)
 - سطح (A1-B2)
 - توضیح کوتاه برای کاربر
 - هدف سناریو – (goal) متن ساده (مثلاً: "درخواست اطلاعات درباره کمک‌هزینه")
 - پایه برای موتور مکالمه (Prompt) اگر از LLM استفاده می‌شود
۳. امکان فعل / غیرفعال کردن سناریو
۴. در فاز بعد: امکان دیدن آمار استفاده از هر سناریو (منابع، میانگین self-confidence و ...)

۵. بگزارش‌گیری و دیتا (نسخه ساده اولیه)

برای شروع، لازم است Backend این دیتای حداقی را بتواند استخراج کند:

• تعداد کاربران ثبت‌نام‌شده

• تعداد سناریوهای اجراشده در یک بازه زمانی

• میانگین self-confidence که کاربران بعد از سناریو داده‌اند

• لیست 5 top سناریوی پر استفاده

رابط کاربری ساده) حتی در حد / JSON یک صفحه ساده (HTML کافی است.
در مراحل بعد می‌توان داشبورد گرافیکی اضافه کرد.

۲.۶. ملاحظات / UX اخلاقی (خیلی مهم برای این پروژه)

• هیچ پیام خطابه شکل «اشتباه است / غلط است» نمایش داده نشود.

• لحن فیدبک باید همیشه تشویقی باشد:

◦ «خوب توضیح دادی، می‌توانی این عبارت را هم امتحان کنی» ...

• از رنگ‌های خشن (قرمز هشدار) برای فیدبک زبانی استفاده نشود.

• کاربر باید حس کند:

◦ «اینجا جای تمرين امن است، می‌توانم اشتباه کنم».

این دقیقاً با روح بیزنس‌پلن و تأکید بر اعتمادبهنه‌نفس و امنیت روانی همخوان است.

SPRAXA — FULL BUSINESS PLAN

۲.۷. جمع‌بندی برای برنامه‌نویس

نسخه اول پلتفرم باید این ۴ کار را حتماً انجام بدهد:

۱. کاربر بتواند ثبت‌نام کند و وارد شود.

۲. از میان سناریوها یکی را انتخاب کند.

۳. داخل محیط مکالمه چند نوبت گفتگو انجام دهد (متن + در صورت امکان صوت).

۴. در پایان، فیدبک دوستانه بگیرد و پیشرفت‌نشان نخیره شود.

باقي چیزها (اپ موبایل، چندزبانگی، گزارش‌های پیچیده، گیمیفیکیشن، ...) کاملاً قابل اضافه‌کردن در فازهای بعدی است.