

Web 版 AI 旅行规划师项目实现指南

1. 项目概述

- 项目目标：**构建一个帮助用户快速生成个性化旅行计划、预算分析和实时辅助的 Web 应用。
- 目标用户：**有中短期旅行计划的个人或家庭，特别是需要多端同步和语音交互的用户。
- 核心价值：**降低旅行规划门槛，提供可执行的路线、预算、住宿与餐饮推荐，并支持行程管理与实时调整。

2. 总体架构设计



3. 功能模块分解

1. 用户管理

- 注册/登录/重置密码，支持 OAuth（如 GitHub/微信）
- 个人偏好设置：常用预算范围、兴趣标签、旅行节奏
- 多计划管理：存储、复制、归档旅行计划

2. 智能行程规划

- 文本输入：表单收集目的地、日期、预算、人数、偏好
- 语音输入：录音上传 → 语音识别 → 自动填表
- 行程生成流程：调用 LLM 生成结构化 JSON → 后端解析存库 → 前端渲染
- 结果内容：每日行程、交通建议、住宿推荐、餐饮与活动建议、注意事项

3. 费用预算与管理

- 初始预算生成：基于 LLM 和预设模板估算各类支出
- 费用记录：支持手动录入/语音录入，按类别汇总
- 预算提醒：预算使用率高时触发通知（邮件或站内提醒）

4. 云端同步与实时辅助

- Supabase Realtime 或 WebSocket 推送更新
- 行程更新、费用变更即时同步到所有登录设备
- 可选：行程临近提醒、天气提示

5. 地图与导航集成

- 高德地图展示每日路线，标记景点/餐厅
- 单点详情页展示路线规划、步行/驾车时间估算

4. 技术选型建议

- 前端：Next.js 14 (App Router) + TypeScript + Tailwind CSS + React Query + Zustand/Recoil
- 后端：Next.js API Routes 或独立 Node.js (Express/NestJS)，包容边缘函数部署
- 数据库：Supabase PostgreSQL (含 Auth、Storage、Edge Functions)
- AI 服务：优先阿里云百炼（助教可用），备用 OpenAI / DeepSeek；使用 LangChain/自研 Prompt 模板
- 语音识别：科大讯飞语音听写（流式 WebSocket）；若使用阿里云，可走 NLS SDK
- 地图：高德地图 JS API (Web)+ Web 服务接口 (POI、路线)
- 容器化：Docker + docker-compose，本地开发配合 Supabase CLI 或 Docker
- CI/CD：GitHub Actions (格式检查、测试、构建 Docker 镜像并推送)

5. 开发环境与初始化

- 创建 Next.js 项目：`npx create-next-app@latest ai-travel-planner`
- 配置 TypeScript、ESLint、Prettier、Husky (pre-commit 运行 lint/test)

3. 集成 Tailwind CSS: `npx tailwindcss init -p`

4. 创建 `.env.local`, 包含:

- `SUPABASE_URL`、`SUPABASE_ANON_KEY`
- `LLM_PROVIDER` (`bailian` / `openai`) 与 `BAILIAN_API_KEY` (或 `OPENAI_API_KEY`), 如需走 <https://dashscope.aliyuncs.com/compatible-mode/v1> 只需填到 `/v1` 级别, 接口会自动补 `/chat/completions`
- `BAILIAN_API_BASE_URL` (默认 <https://dashscope.aliyuncs.com/api/v1/services/aigc/text-generation/generation>, 兼容模式可填 `.../compatible-mode/v1`)、`BAILIAN_MODEL`
- `NEXT_PUBLIC_AMAP_KEY` (高德 JS SDK 前端 Key)
- `NEXT_PUBLIC_AMAP_SECURITY_CODE` (高德 JS SDK 安全密钥, 开启「安全密钥」后必填)
- `AMAP_REST_KEY` (高德 Web 服务 Key, 用于 POI 与地理编码)
- `IFLYTEK_APP_ID/KEY` 等

5. 初始化 Supabase 项目: 定义表结构、行级安全策略 (RLS)

6. 建立 GitHub 仓库并推送初始代码, 准备 README

6. 数据库与实体设计 (Supabase/PostgreSQL)

表名	关键字段	说明
<code>users</code>	<code>id</code> , <code>email</code> , <code>display_name</code> , <code>default_budget</code> , <code>preferences JSONB</code>	Supabase Auth 自动创建, 扩展 profile
<code>trips</code>	<code>id</code> , <code>user_id</code> , <code>title</code> , <code>destination</code> , <code>start_date</code> , <code>end_date</code> , <code>budget</code> , <code>travelers JSONB</code> , <code>tags</code>	行程主表
<code>trip_days</code>	<code>id</code> , <code>trip_id</code> , <code>date</code> , <code>summary</code> , <code>notes</code>	每日行程概览
<code>activities</code>	<code>id</code> , <code>trip_day_id</code> , <code>type</code> , <code>start_time</code> , <code>end_time</code> , <code>location</code> , <code>poi_id</code> , <code>cost</code> , <code>details JSONB</code>	景点/餐饮/交通等活动
<code>expenses</code>	<code>id</code> , <code>trip_id</code> , <code>category</code> , <code>amount</code> , <code>currency</code> , <code>source</code> , <code>memo</code> , <code>created_at</code>	费用记录
<code>voice_inputs</code>	<code>id</code> , <code>trip_id</code> , <code>user_id</code> , <code>transcript</code> , <code>audio_url</code> , <code>status</code>	语音输入留存

表名	关键字段	说明
trip_intents	<code>id</code> , <code>user_id</code> , <code>voice_input_id</code> , <code>raw_input</code> , <code>structured_payload JSONB</code> , <code>confidence</code> , <code>field_confidences JSONB</code> , <code>source</code>	语音/文本描述拆解后的结构化结果，便于多端回填
sync_logs	<code>id</code> , <code>trip_id</code> , <code>change</code> , <code>created_at</code>	变更日志（可驱动通知）

- 启用 RLS：仅行程拥有者可读写；支持分享可选令牌。
- 对 `activities.location` 使用 PostGIS（可选）提高地理查询能力。

7. 行程生成与 AI 工作流

1. Prompt 模板：

- 输入：目的地、天数、预算、同行人、偏好、已有计划限制
- 输出：结构化 JSON，包括每日时间段、交通方式、费用估算、注意事项

2. 调用流程：

- 前端提交参数 → 后端校验 → 触发 LLM
- 若使用阿里云百炼：调用 `dashscope` SDK，选 `qwen-max` 或 `tongyi-turbo`
- 在后端进行 JSON Schema 验证，失败时重试或回退到模板
- ⚠ 兼容 OpenAI 的 `https://dashscope.aliyuncs.com/compatible-mode/v1` 仅需填写到 `/v1`，其余路径由后端自动补齐；若走第三方兼容网关请将 `LLM_PROVIDER` 设为 `openai`，否则会得到 502 错误

3. 后处理：

- 拆分每日活动写入 `trip_days`、`activities`
- 调用高德 POI 搜索补全经纬度与图片
- 计算预算汇总写入 `trips.budget_breakdown`

4. 实时辅助：

- 用户手动调整活动时，可触发轻量 LLM 调整建议或推荐替代

8. 费用预算与管理实现

- 预算估算基于 LLM 给出的类别（住宿/交通/餐饮/门票/杂项）
- 费用记录：
 - 表单或语音上传（语音识别后触发 NLP 分类器或规则分类）
 - 支持多币种，统一换算为主币种

- 数据可视化：
 - 饼图展示分类占比、折线图展示日支出
- 提醒机制：
 - 若实际支出超过预算的 80%，推送站内通知或邮件

9. 语音功能设计

1. 前端：

- 使用 MediaRecorder 录音，生成 PCM/MP3 Blob
- 录音组件提供状态（录制中/上传/完成）、噪音提示

2. 后端：

- 将音频上传至 Supabase Storage 暂存
- 按 `VOICE_RECOGNIZER_PROVIDER` 调用真实服务：目前接入科大讯飞语音听写流式 WebSocket（iflytek），同时保留 `mock` 降级与 `openai` 备选
- 对识别文本进行意图解析，自动填充表单或新增费用条目

3. 容错：

- 识别失败时提示重试
- 支持手动编辑识别结果
- 通过 `VOICE_RECOGNIZER_TIMEOUT_MS` 控制 API 超时（默认 45s），超时返回友好提示

配置说明：

- `VOICE_RECOGNIZER_PROVIDER=iflytek` 时，需要配置 `IFLYTEK_APP_ID`、`IFLYTEK_API_KEY`、`IFLYTEK_API_SECRET` 并启用讯飞“语音听写·流式 WebSocket”接口；可通过 `IFLYTEK_DOMAIN`、`IFLYTEK_LANGUAGE`、`IFLYTEK_ACCENT` 等参数控制领域与语种。
- `VOICE_RECOGNIZER_PROVIDER=openai` 可作为备选方案，需提供 `OPENAI_API_KEY`、`OPENAI_VOICE_MODEL` 等。
- `VOICE_RECOGNIZER_PROVIDER=mock` 时，可通过 `VOICE_RECOGNIZER MOCK TRANSCRIPT` 设置固定演示文案。

10. 地图与路线

- 引入高德 JS SDK，配置安全密钥与 `referer` 白名单。前端 `TripMap` 组件会根据活动坐标生成 Marker，缺失点位时会优先读取 `details.latitude/longitude`，再回退到 JS SDK

地理编码与服务端 `/api/geocode`。

- 组件职责：

- 行程概览地图：展示所有活动点位与真实路线
- 单日地图：按时间顺序动态绘制路线，支持活动卡片 ↔ 地图点位联动
- POI 详情卡片：展示地址、营业时间、评分、照片

- 后端调用高德 Web 服务：

- POI 搜索 (`/place/text`) 匹配 LLM 返回的地点，并写入 `activities.details.latitude/longitude`
- 地理编码 (`/v3/geocode/geo`) 为纯文本地点提供坐标
- 路线规划：新增 `/api/directions` 代理
`https://restapi.amap.com/v5/direction/{driving,walking,cycling}`，输出 polyline。前端按活动间两两请求真实路线，不再使用直线连线。

- 缓存策略：

- 服务端对 POI/路线可结合 Supabase Edge Functions 或 Redis 做持久缓存
- 前端 `TripMap` 内部以 `Map` 形式缓存 geocode 与路线结果，避免同页多次请求
- 若路线接口失败，自动降级为两点之间的直线，保证地图元素始终可见

11. 前端页面与组件规划

- `/landing`：产品介绍、主要卖点
- `/login`, `/register`, `/forgot-password`
- `/dashboard`：行程列表、快速创建
- `/planner/new`：表单 + 语音输入 + 偏好面板
- `/trips/[tripId]`：
 - 行程概览、每日卡片、地图、预算小组件
 - 子路由 `/itinerary`, `/budget`, `/notes`
- 公共组件：语音按钮、地图容器、行程时间轴、费用表格、LLM 生成进度条

12. 后端接口设计（示例）

方法	路径	描述
<code>POST</code>	<code>/api/auth/login</code>	登录（若使用 Supabase，可直接用 auth 客户端）
<code>POST</code>	<code>/api/trips</code>	创建行程并触发生成

方法	路径	描述
GET	/api/trips	获取用户所有行程
GET	/api/trips/:id	获取单个行程详情 (含活动、费用)
PATCH	/api/trips/:id	更新行程摘要、偏好
POST	/api/voice-inputs	上传语音并触发识别
POST	/api/trip-intents	解析语音/文本描述 并生成结构化行程意图
POST	/api/expenses	新增费用
GET	/api/expenses?tripId=	获取费用列表
POST	/api/llm/generate	手动触发行程生成 (如重新规划)
GET	/api/directions? origin=lng, lat&destination=lng, lat&mode=driving	调用高德路线规划， 返回 polyline，前端 用来绘制真实路径

- 使用 JWT/Session 验证用户身份。
- 对 AI 调用设置速率限制 (Supabase Edge Functions + Redis)。

13. 状态管理与同步

- React Query: 处理 API 请求、缓存与乐观更新
- Zustand/Recoil: 存储 UI 状态 (语音录制、临时表单数据)
- Supabase Realtime: 监听 `activities`、`expenses` 表的 insert/update，自动刷新前端
- Web Push/邮件: 通知预算超支、行程即将开始等

14. 测试策略

- 单元测试: 组件 (React Testing Library)、服务函数 (Jest)
- 集成测试: Next.js API Routes + Supabase 测试实例
- 端到端: Playwright/Cypress，覆盖行程创建、语音输入回填、费用录入流程
- Mock 外部服务: 使用 MSW/Mock Service Worker 模拟高德与语音 API
- 负载与速率测试: 对 LLM 调度器与地图接口进行限流测试

15. 部署与运维

1. Docker 化

- 前端与后端同仓库：Next.js 多阶段构建（builder + runner）
- 使用 `docker-compose` 本地启动：`web`、`supabase`、`edge-functions`（可选）

2. 生产部署

- Vercel 部署前端 + Edge Functions，或自建服务器（Docker + Nginx）
- Supabase 托管数据库与 Auth
- 对象存储使用 Supabase Storage；若需大文件可迁移至 OSS/S3

3. 环境配置

- `.env` 不入库，使用 `.env.example` 提示变量
- 设置 CI 密钥（Supabase、LLM、语音、地图）

4. 日志与监控

- 前端：Sentry/LogRocket 记录异常
- 后端：Supabase Logs，配合 OpenTelemetry（可选）

16. 文档与提交流程

1. README 内容要求

- 项目简介、功能展示（截图/GIF）
- 环境变量说明、开发/生产启动步骤
- Docker 镜像获取与运行方法（示例命令）
- API Key 提供方式：若非阿里云百炼，需在 README 中提供 key 并保证 3 个月内有效

2. PDF 文件

- 包含 GitHub 仓库地址、README 全文、关键截图
- 可使用 `md-to-pdf` 或 `pandoc` 自动生成

3. Git 历史

- 保留细粒度提交（功能模块、修复、文档）
- 提交信息规范：`feat: add itinerary generator`, `chore: configure supabase`

4. 交付物

- GitHub 仓库链接
- Docker 镜像文件（可通过 GitHub Releases 上传 tar）

- README + PDF + API Key 信息

17. 开发迭代建议

迭代	时长	目标
Iteration 0	1 天	需求澄清、架构设计、环境搭建
Iteration 1	3 天	完成认证、基础页面、Supabase 表结构
Iteration 2	4 天	实现行程生成流程、地图展示
Iteration 3	3 天	加入费用管理、语音识别、同步
Iteration 4	2 天	打磨 UI/UX、增加测试、文档与部署
Iteration 5	1 天	准备 Docker 镜像、PDF 导出、最终检查

- 每个迭代结束进行演示与回顾

18. 风险与缓解

- 第三方 API 受限：准备备用服务（地图、语音、LLM）与降级策略
- LLM 输出不稳定：加入 JSON Schema 验证、重试、人工编辑界面
- 预算数据准确性：结合公开价格 API 或预设平均值，允许用户校正
- 语音隐私：上传前提醒用户、仅存储必要数据、定期清理音频
- 多端同步冲突：对行程/费用更新设置乐观锁或版本号

19. 验收标准与演示脚本

1. 注册新用户 → 登录 → 录入偏好
2. 使用语音描述旅行需求 → 自动生成行程
3. 查看行程地图与每日安排 → 调整活动顺序
4. 记录一笔费用 → 查看预算变化
5. 在移动端浏览器登录 → 确认行程同步
6. 导出行程为 PDF（可选）

20. 后续扩展方向

- AI 推荐可视化：行程与预算的可操作卡片
- 支持多语言界面与多货币换算

- 离线模式（PWA）与缓存策略
- 社区功能：分享行程、评价景点
- 实时数据整合：天气、航班、突发事件提醒

提示：在 README 中显式说明助教可用的 API Key（若非阿里云百炼），并确保镜像内默认配置指向测试环境。提交前检查 Docker 镜像能在无源码的情况下独立运行。

21. 详细开发计划

Iteration 0 (第 1 天)：项目启动与基线搭建

- PM/Tech Lead
 - 最终确认需求范围、验收标准与里程碑；整理风险清单、备选第三方服务。
 - 细化信息架构与数据流图，冻结技术选型。
- 全体
 - 建立协作工具：GitHub 项目、Issue 模板、CI/CD 框架、知识库。
- Deliverables
 - 项目甘特图/看板、架构图、环境变量清单、开发规范文档。

Iteration 1 (第 2-4 天)：基础框架与账号体系

- 前端
 - 初始化 Next.js + Tailwind + TypeScript；搭建全局布局与 UI 设计系统。
 - 实现登录/注册/重置密码页，集成 Supabase Auth 与会话管理。
- 后端
 - 创建基础数据库 schema (`users`, `trips`) 并配置 RLS。
 - 实现认证相关 API 封装（若使用 Supabase 客户端需编写服务层）。
- 测试/运维
 - 配置 ESLint、Prettier、Husky；在 GitHub Actions 中加入 lint/test pipeline。
- Deliverables
 - 可运行的基础 Web 框架、账号流程 Demo、CI 任务成功运行。

Iteration 2 (第 5-8 天): 行程创建与 LLM 生成主流程

- 前端
 - 完成新建行程表单 (文本输入 + 偏好设置 UI)。
 - 实现生成状态页面 (进度条、错误处理、重试按钮)。
- 后端
 - 接入 LLM 服务，定义 Prompt 模板与 JSON Schema 校验逻辑。
 - 完成行程生成 API: 写入 `trips`, `trip_days`, `activities`。
 - 构建高德 POI 匹配服务 (Edge Function 或后端服务)。
- 测试
 - 为 LLM 调度编写集成测试 (Mock LLM & 高德接口)。
- **Deliverables**
 - 从表单到行程展示的完整闭环，包含至少一个真实示例。

Iteration 3 (第 9-11 天): 地图展示与基础可视化

- 前端
 - 集成高德地图组件，展示行程点位与路线。
 - 实现行程详情页 (时间轴、活动卡片、地图联动)。
- 后端
 - 提供地图数据接口 (POI 元数据、路线缓存)。
 - 优化 `activities` 查询，提供分页/筛选支持。
- 测试
 - 编写地图组件可视化回归测试 (视觉或关键 DOM 断言)。
- **Deliverables**
 - 行程详情页面可交互地图、活动与地图联动体验。

Iteration 4 (第 12-14 天): 语音输入与费用管理

- 前端
 - 开发语音录制组件 (状态、权限、重试)。
 - 费用录入界面 (表格、分类筛选、统计图表)。
- 后端
 - 接入语音识别 API, 创建 `voice_inputs`, `expenses` 相关服务。

- 实现预算提醒规则、通知触发机制（站内或邮件）。
- 启用 Supabase Realtime 监听费用与活动更新。
- 测试
 - 端到端测试覆盖语音上传→文本回填→自动创建费用条目。
- Deliverables
 - 完整语音输入链路、费用管理模块、实时同步能力。

Iteration 5 (第 15-16 天): 打磨、部署与文档交付

- 前端
 - UI/UX 打磨，适配移动端，新增提示与空状态。
 - 增加行程导出（PDF/打印视图或分享链接）。
- 后端/运维
 - 完成 Dockerfile、多阶段构建；编写 docker-compose 样例。
 - 配置生产环境 .env 管理、安全审查、速率限制。
 - 准备监控与日志方案（Sentry/Supabase Logs）。
- 文档
 - 更新 README（运行说明、API Key、Docker 镜像、部署指南）。
 - 生成提交所需 PDF（嵌入 README、截图、仓库链接）。
- Deliverables
 - 可部署 Docker 镜像、README+PDF 完整文档、演示脚本校验通过。

持续性工作（贯穿所有迭代）

- 对关键模块实施 Code Review，确保提交历史清晰且粒度合理。
- 维护测试覆盖率目标：单元 $\geq 70\%$ ，端到端覆盖核心用户路径。
- 定期备份 Supabase 数据（开发环境），并记录 API Key 有效期。

22. 代理执行任务清单（顺序完成）

执行进度提醒：任务 1-36 已完成（最新完成任务 36：行程地图定位修复），继续推进后续交付。

1. 需求确认与环境检查

- 阅读实现指南，列出所有外部服务账号需求。
- 在 README 起草环境变量与密钥清单草稿。

2. 仓库初始化

- 使用 `create-next-app` 创建项目骨架，提交初始代码。
- 配置 TypeScript、ESLint、Prettier、Husky，并验证 lint。

3. UI 底层搭建

- 集成 Tailwind CSS，搭建全局布局与主题色。
- 实现通用组件：按钮、表单输入、加载态、通知 Toast。

4. 认证与用户管理

- 连接 Supabase Auth，完成注册/登录/重置密码页面。
- 创建用户偏好表 `user_profiles`，实现偏好表单与保存。

5. 数据库结构落地

- 在 Supabase 中创建 `trips`, `trip_days`, `activities`, `expenses`, `voice_inputs` 等表。
- 编写 RLS 策略与初始 SQL 迁移脚本，提交到仓库。

6. 基础 API 层

- 在 Next.js API Routes 中实现通用响应封装与错误处理。
- 创建 `GET/POST /api/trips` 基础 CRUD，连通前端仪表盘列表。

7. 行程创建表单

- 完成新建行程表单 UX（目的地、日期、预算、偏好）。
- 支持保存为草稿并展示在仪表盘。

8. LLM 集成基础（已完成：引入百炼 SDK 封装、Prompt 模板与 JSON Schema 重试验证）

- 接入首选 LLM（默认阿里云百炼），编写 Prompt 模板与 SDK 封装。
- 创建 JSON Schema 验证器与重试机制（失败回滚草稿）。

9. 行程生成流程联通（已完成：新增 `/api/llm/generate` 接口、生成进度界面与错误处理）

- `POST /api/llm/generate`：接收表单数据、触发 LLM、解析结果入库。
- 前端展示生成进度、结果成功页及错误提示。

10. 行程详情页面（已完成：新增 `/trips/[id]` 详情页、活动时间/备注编辑与删除能力）

- 构建 `/trips/[id]` 页面：行程纲要、每日卡片列表。
- 实现基础活动编辑（时间、备注、删除）。

11. 高德地图集成（已完成：在行程详情页嵌入高德地图，活动卡片联动点位并绘制每日路线）

- 行程详情页加载高德 JS SDK，展示行程点位及连续路线。
 - 活动卡片与地图点位双向联动，支持点击高亮与信息窗提示。
12. **POI 数据完善** (已完成：后端调用高德 Web 服务补全活动 POI、经纬度与照片信息，并做请求级缓存)
- 在生成流程中调用高德 POI 搜索，填充 `activities` 的 `poi_id`、经纬度与图片。
 - 使用请求级缓存与限流策略避免重复外呼，为后续接入 Supabase Edge 缓存做准备。
13. **预算估算与展示** (已完成：解析 LLM 预算 JSON 并在行程详情页展示分类摘要与剩余提醒)
- 后端存储 `budget_breakdown` 字段，解析出分类占比、提示信息。
 - 详情页新增预算组件，展示总额、类别进度条及超支预警。
14. **费用记录功能** (已完成：提供费用 CRUD 接口及前端支出管理面板)
- 实现 `GET/POST /api/expenses`，返回分类汇总、总额与最新记录。
 - 详情页新增费用面板，支持分类筛选、进度图示及快速新增支出。
15. **语音录制前端** (已完成：封装可复用的语音录制组件，接入新建行程与费用面板，实现录制/回放/上传状态提示)
- 使用 `MediaRecorder` 实现语音录制、回放、上传 UI。
 - 加入状态提示（录制中、上传中、失败重试）。
16. **语音识别后端** (已完成：新增 `/api/voice-inputs` 上传接口，落盘 Supabase Storage，返回识别文本与费用草稿信息)
- 将音频上传至 Supabase Storage 并调用语音识别 API。
 - 将识别文本保存至 `voice_inputs`，回填表单或自动创建费用。
17. **实时同步** (已完成：前端订阅 Supabase Realtime，自动刷新行程活动与费用面板)
- 启用 Supabase Realtime 监听 `activities`、`expenses` 表。
 - 前端通过 React Query 订阅实现无刷新更新。
18. **通知与提醒** (已完成：预算超阈值提醒与行程开始前 24 小时站内提示)
- 预算超过阈值时发送站内通知（或集成邮件服务）。
 - 行程即将开始前 24 小时推送提醒（定时任务/Edge Function）。
19. **导出与分享** (已完成：提供 `/trips/[id]/share` 只读分享页、复制链接与一键 PDF 打印能力)
- 提供行程导出 PDF 或打印视图，支持下载。
 - 生成分享链接（可选受限访问）。
20. **端到端测试** (已完成：新增 Playwright 配置与示例用例，覆盖分享页回退与打印模式)

- 使用 Playwright/Cypress 编写关键流程测试：登录、行程生成、费用录入、语音识别。
 - 配置测试数据隔离与 CI 集成。
21. **性能与安全加固**（已完成：为 LLM/语音接口新增速率限制、启用全局 API 中间件限流与安全响应头，并在生产环境隐藏敏感错误详情）
- 引入速率限制（LLM/语音 API），审查敏感日志。
 - 开启 CSP、安全头、中间件限流。
22. **Docker 化**（已完成：新增多阶段 Dockerfile、提供 `docker-compose.yml` 与示例环境变量文件，支持本地一键启动 Web 与 Supabase/Postgres 环境）
- 编写多阶段 Dockerfile，支持生产构建。
 - 编写 `docker-compose.yml`（含 Supabase/代理服务）供本地/助教运行。
23. **CI/CD 完善**（已完成：新增 `.github/workflows/ci.yml`，串联 lint/test/build，并在推送 `main` 时构建并推送 Docker 镜像；确立语义化版本与标签规范）
- GitHub Actions: `quality-checks` 负责 `npm ci`、`npm run lint`、`npm run test:e2e` 与 `npm run build`，失败即阻断后续流程。
 - Docker 发布：`docker` 任务仅在推送 `main` 分支时运行，使用 Buildx 推送至 GHCR（`ghcr.io/<仓库>:latest` 与 `:v<package.version>`）。
 - 版本策略：沿用语义化版本（SemVer），发布前使用 `npm version [patch|minor|major]` 同步 `package.json`，并在仓库创建匹配的 `vX.Y.Z` 标签。
24. **文档与示例数据**（已完成：重写 README，新增 `.env.example` & 示例行 JSON，说明本地/容器启动流程）
- README：新增快速开始、环境变量说明、脚本、Docker 部署、CI 策略与常见问题；保持中文叙述保证交付完整性。
 - 环境变量：在仓库根目录创建 `.env.example`，对关键密钥给出默认值与注释（含 LLM、地图、语音、通知等配置）。
 - 示例数据：在 `docs/examples/sample-trip.json` 提供台北 3 日行程示例，覆盖预算拆分、每日活动、费用细节，便于演示与测试。
25. **PDF 交付物**（已完成：新增 `scripts/export-pdfs.mjs`，基于 Playwright+marked 生成 README 与实现指南 PDF，并输出到 `docs/output/`）
- 脚本：运行 `npm run export:pdf` 自动渲染 Markdown → HTML → PDF，首次需执行 `npm install` 与 `npx playwright install chromium`。
 - 产物：`docs/output/README.pdf` 与 `docs/output/AI旅行规划师项目实现指南.pdf`；脚本在受限环境下可加 `--with-escalated_permissions` 保证浏览器启动。
 - 文档：README 新增操作说明，提醒生成的 PDF 在交付前打包提交给评审。
26. **最终检查与交付**（已完成：执行 lint 校验、整理测试记录并列出交付清单，确认 Docker 镜像与 PDF 文档准备就绪）

- 自测：运行 `npm run lint` 确保代码规范；记录 E2E 依赖（需 Supabase + Playwright）供验收者复现。
- 测试报告：在 `docs/test-report.md` 汇总自动化与手动检查结果，标明环境依赖与潜在风险。
- 交付物：确认 `docs/output/*.pdf`、Docker 镜像构建指令、README 操作指南齐备，可据此发布最终 Release。

27. 语音识别服务接入

(已完成：接入科大讯飞流式 WebSocket 并保留 mock/openai 降级)

- 在 `src/lib/voice/recognizer.ts` 中实现基于讯飞流式接口的鉴权、分片推送与结果解析。
- 识别结果统一回填 Supabase 与前端，支持 `iflytek`、`mock`、`openai` 三种模式灵活切换。
- 更新 `.env.example`、`.env.docker` 与 README，补充讯飞参数说明及可选的 OpenAI 备选配置。

28. 语音识别体验验证 (已完成：新增语音测试场景页与 Playwright 用例，验证成功/失败链路并补充测试记录)

- 新增 `/test/voice-scenarios` 仅在 `PLAYWRIGHT_BYPASS_AUTH` 启用时可访问，复刻行程备注与费用语音录入交互，方便端到端脚本驱动 `MediaRecorder`。
- 编写 `tests/e2e/voice-recorder.spec.ts`，通过 Mock `MediaRecorder` 与 `/api/voice-inputs` 响应覆盖成功转写与失败提示两条路径，并使用 Playwright 配置注入绕过令牌。
- 在 `docs/test-report.md` 追加测试条目，记录 `npx playwright test tests/e2e/voice-recorder.spec.ts` (2025-11-04, 本地带 bypass token) 执行结果，确保语音识别恢复消息、错误文案均符合预期。

29. 我的行程列表页 (已完成：构建 `/trips` 总览页、完善接口筛选并补充多状态反馈)

- 新增客户端页面 `src/app/trips/page.tsx`，支持状态切换、关键字搜索、本地排序，同时展示预算、日期区间与更新相对时间。
- 列表卡片内提供「查看详情 / 分享 / 继续生成」入口；页眉含「新建行程」「刷新列表」按钮，未登录时展示引导登录的空态提示。
- 更新 `GET /api/trips`：补充 `tags` 字段、按 `updated_at` 排序，并支持 `search` 查询参数（标题/目的地模糊匹配），满足列表筛选需求。
- 空列表、加载失败等场景均提供明确文案与重试操作，保证导航回流和多计划管理体验。

30. 多行程管理自测 (已完成：新增多行程端到端脚本、补充 README/指南说明与测试记录)

- 新增 `tests/e2e/my-trips.spec.ts`，覆盖「创建行程 -> 跳转生成页」以及「我的行程列表 → 详情页」两条链路（Mock `/api/trips` 与 `/api/trips/:id` 响应）。

- 调整 `/trips/[tripId]` 与 `/trips/[tripId]/generate` 客户端页面以兼容 Next.js 19+ 传递异步 `params`，确保测试与真实环境均可加载。
- 在 README 增补「我的行程页面使用提示」小节，并明确截图存放于 `docs/screenshots/my-trips-overview.png`；同时在本指南和 `docs/test-report.md` 中记录对应测试与验证结果。

31. 语音/文本行程需求解析（已完成：TripIntentAssistant + 本地解析器，语音/文本可一键填表）

- 新增 `src/types/trip-intent.ts` 与 `src/lib/trip-intent/parser.ts`，通过目的地词典+正则组合拆解目的地、日期/天数、预算、同行人数、偏好等字段，并计算字段置信度。
- `/planner/new` 右侧引入 `TripIntentAssistant`（语音+文本双入口），实时展示拆解结果、置信度条以及缺失字段提示，并支持保留原描述。
- 「应用到表单」会自动写入目的地、预算、标签及同行人草稿（成人/儿童拆分），同时将原始描述追加到备注，提示用户继续补齐未解析字段。

32. 行程需求解析服务（已完成：后端存储+API+语音链路联动）

- 新增 `supabase/migrations/20241115_add_trip_intents.sql` 建表，字段涵盖 `structured_payload`、`field_confidences`、`confidence`、`voice_input_id`，并启用 RLS + 更新时间触发器。
- 提供 `POST /api/trip-intents` (`src/app/api/trip-intents/route.ts`)，校验 `rawInput/source/voiceInputId`，复用 `parseTripIntent` 生成结构化 JSON，调用 `persistTripIntentDraft` 入库并返回 `TripIntentDraft`。
- `/api/voice-inputs` 在 `purpose=trip_notes` 时自动串联解析服务，语音识别成功即写入 `trip_intents` 并把 `tripIntent` 随响应返回，方便前端快速回填。

33. 联调与验证（已完成：前/后端埋点、解析 UI 兜底、Playwright 端到端用例）

- `TripIntentAssistant` 新增 `sendBeacon` 埋点 (`src/lib/analytics.ts`)，在成功/失败时记录耗时、缺失字段、错误原因，并在解析多次失败时显示提示卡片引导用户改用手动填写/稍后重试；`NEXT_PUBLIC_TRIP_INTENT_ANALYTICS_ENDPOINT` 可选上报到外部日志服务。
- 语音路径支持携带 `tripIntent` 直达前端，`/planner/new` 侧栏自动展示拆解结果并可应用到表单；新增体验指标（最近一次解析耗时/置信度）帮助调试。
- 新增 `tests/e2e/trip-intent-assistant.spec.ts` 覆盖「文本解析→应用填表」与「语音上传→自动拆解」全链路，并抽离 `tests/e2e/utils/voice.ts` 复用语音 Mock；`docs/test-report.md` 已记录最新执行结论。

34. 行程地图体验增强（已完成：新增筛选工具栏、活动细节卡片与类型图例，提升地图可读性）

- 地图上方新增「全部行程/按日筛选」工具栏与「重置视图」按钮，可快速聚焦单日路线并一键回到全局视角，符合第 10 节对概览/单日地图的并行展示建议。
- 选中点位后在地图内弹出活动详情卡片，呈现活动顺位、摘要、类型、时间与地址，支持随时清除选中，弥补 POI 详情卡片需求。
- 引入活动类型颜色图例与空态提示，帮助用户理解颜色含义，并在缺少位置信息时给出中文错误指引，保持可用性与指南推荐的友好反馈。

35.  行程地图交互优化（已完成：修复日期筛选失效、移除遮挡提示并强化路线高亮）

- 修正小方框筛选逻辑，避免已有活动选中时强制回退，点击任意日期即可只展示当日点位与路线，高亮状态清晰可见。
- 将活动详情移出地图叠加层，改为下方信息卡，提供「在地图中定位」「清除选择」双按钮，阅读与地图操作互不干扰。
- 依据当前筛选或选中活动自动调整标记与折线的不透明度/层级，突出重点路线，进一步贴合第 10 节对“地点+路线高亮”呈现的要求。

36.  行程地图定位修复（已完成：保障高德 SDK 初始化与重复定位体验）

- 为高德覆盖物绘制流程增加版本号，确保异步加载完成后会重新执行定位逻辑，避免地图一直停留在初始视角。
- 「在地图中定位」按钮与活动选中效果均会调用最新的标记信息，自动调整缩放并平移至目标坐标，消除定位无响应的问题。
- 再次核对 `@amap/amap-javascript-loader` 调用链与 Geocoder 插件加载，确认高德服务在加载失败时会输出中文错误提醒，方便排查 Key/SecurityCode 配置。