【アイデア決めの流れ】

1. Who ペルソナを決める (困っている人の具体性を高める)
2. What「負」を決める
3. プロダクトの「キャッチフレーズ」を決める
4. How 我々の提案手法で実装する機能を決める
5. 競合他社の調査と比較
6. 技術選定

1. Who ペルソナを決める (困っている人の具体性を高める)

- **幼少期に祖母から散歩中、雑学を教わった思い出をきっかけに、今も日中に一人で散歩することを趣味にしている20代の女子学生**

学生系のペルソナ (創と実体験に基づく)

- 習慣的に日中に趣味で1人散歩をしている
- 20代の男子学生
- 普段の散歩道に飽きている
- 道端の花や風景、日常生活にまつわる雑学を学びたいが、自分で調べたりするのが面倒
- 散歩中に見つけた面白い発見は誰かに共有したいが、相手がいなくて悲しい

- たまに散歩をしている20代女性
- 幼少期はおばあちゃんが教えてくれて、散歩が楽しかった（例：グミの木が食べられるって教えてもらって、みんなで食べて盛り上がった）
- 今は植物の種類を知らなさ過ぎて、散歩中に楽しめない

- 少し病みがちな浪人生
- 神経症気味で気分転換を促されるものの、何をしていても受験について考えてしまう

- 20代男性
- モテたい
- 運動は始めた
- 知識があるとメロいらしいモテるためには教養が必要だと思って、身近な様々な知識が欲しい
- ランニングしているときの時間も有効に使いたい

社会人系のペルソナ

佐藤花さん

- 在宅勤務の20代後半女性
- 都内で一人暮らし
- 家からほぼ出ないため、健康/体型維持のために近所を散歩しているが、マンネリ気味
- 最近気分が乗らず散歩をサボることもある
- 散歩中気になることはあるがわざわざ調べるほどではないかぁと思いがち
- 楽しく散歩をするため、新しい発見を得たい
- 撮った写真や気付きを後から見返したい

- オフィス勤務の30代の女性
- 取引先や職場とのやりとりに疲れている
- 仕事や悩みのことを忘れてのんびり散歩を楽しみたい
- 花を見るのは好きで、見かけた花の写真を撮ったりはするが、詳しくは知らない

- 散歩好きな女性
- 知らない道を歩くのが楽しい一方、夜しか散歩できずに身の安心を考えて知っている道しか散歩できない

- 50代男女
- 運動不足を指摘されて、気にして散歩を始めたものの、単調に歩くだけではつまらない
- 老後に備えて体力づくりをしたいor趣味
- しかしそのモチベがわかない

- 社会人
- 趣味で小説を書いている
- 日々の生活に忙殺されている
- 手軽にクリエイティブな思考になりたいと考えている

子育て中主婦系のペルソナ

- 子育て中の主婦
- 子供の好奇心を養いたいと思っているものの、自身の知識不足からどのように教えればよいのか悩んでいる

- 40代女性
- 小学生の息子を持つ
- 虫取り大好き歩くの大好き公園大好き勉強大嫌い
- 勉強のきっかけとして、自然に触れがちな息子に面白い知識を与えたい

- 6歳(年長)の息子を持つ30代前半の女性
- 普段から息子と近所を散歩している
- 息子に、この植物は何？と聞かれることが多いが、すぐにはわからない
- 子供に＋αの雑学や、逆質問をすることで、息子の考える力と知識を養いたいと考えている。

2. What「負」を決める

- **散歩道は変えたくないが、普段の散歩道に飽きてきている**
- **幼少期の新しいものに満ち溢れていた景色を思い出したい (童心に帰りたい)**
- **道端の花や風景、日常生活にまつわる雑学を学びたいが、面白ポイントにも気付けないし、自分で調べたりするのは面倒**
- **散歩中に見つけた面白い発見を誰かに共有したいが、相手がいなくて悲しい**

3. プロダクトの「キャッチフレーズ」を決める

- **忘れていた「どうして？」を見つけにいこう。AIカメラと問いで作る、新しい散歩地図。**

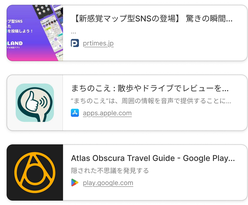
4. How 我々の提案手法

業態：スマホアプリ (Webアプリ)

- **カメラで植物や昆虫、空模様などを撮影すると、その名前と「雑学・面白ポイント(←後で固有名詞を作る)」を教えてくれる機能、**
- **発見物に関連する、知的好奇心を刺激するような答えのない問いを、一問出題する機能 (例：子供の無邪気な問い)、**
- **気に入った発見物の写真・質問内容・回答を地図上にプロットし、他のユーザーに共有する機能、**
- **他のユーザの投稿を地図上で閲覧できる機能、**

5. 競合他社の調査と比較

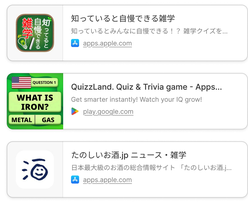
散歩系アプリ



地図系アプリ

植物系アプリ

雑学系アプリ



6. 技術選定

バックエンド

- Docker Compose (仮想環境構築)
  - Webサーバー
  - APIサーバー
  - DBサーバー (PosgreSQL)
- uv (Python パッケージ管理)
- Flask (FAST APIサーバー)
- 予選ではngrokを使う
- 将来的なデプロイ先 (Supabase)

フロントエンド

- Next.js (TypeScript)
- MUI

画面

- 認証
  - ログイン
  - 新規登録
- 地図表示画面
  - フィルター機能（自分/他人）
- 一覧画面
  - 距離順一覧
- コレクション画面（余裕があれば）
- カメラで読み取る画面
- コメント画面
- AIからの返答画面
  - 公開設定
  - キャンセルボタン
  - 保存ボタン

機能

- フロントエンド
  - 認証周り
  - 地図を表示する機構
  - カメラを起動する
  - 距離順タイムライン
- バックエンド
  - 撮影した画像から物体を検知して「名前」や「雑学」を返す (挑戦的) (バックエンドより)
    - Web検索やファインチューニング
    - APIをゴリゴリ使う路線 - 鈴木くんが技術的調査をする (どのモデルを使うか(GPT, Gemini など) + 画像をどうやって投稿するか)
    - Amazon Bedrock、GCP(VertexAI) などなどをゴリゴリ使ったら簡単になるんじゃない？路線
- 画像にまつわる「問い」を返す (挑戦的) (バックエンドより)
- データベースに投稿を保存する機構 (バックエンドより) - 泉 担当 (S3やGCP使う路線か、サーバー上に保存するか 自動化は must)