

落とし物の報告アプリの 作成



2023年度プロジェクトリサーチ

『落とし物の報告アプリの作成』 活動報告書

後藤桃衣

1. 背景

落とし物をする時、駅や交番へ連絡する以外に探す手段が限られ、貴重品でなければ諦めてしまうこともある。反対に、街で落とし物を見つけても、それら全てを届けるには手間がかかる。そこで、「落とし物発見者が落とし物情報を地図上に投稿し共有することで、落とし主は自分の移動経路から簡単に探し出せるのではないか」と考えた。

2. 目的

街で見つけた落とし物を手軽に撮影・共有できる機能を提供する iOS アプリを開発し、開発環境の構築、コーディング、バージョン管理等の実践を通して、アプリ開発のいろはを学ぶことが活動の目的である。

ただし、当アプリ（現時点）では、貴重品の悪用リスクを考慮し、緊急性の低い落とし物（例：キーホルダー）を想定する。アプリを通じて、一見すると貴重品でなくても、落とし主にとって思い入れのあるアイテムが、再び手元へ戻るための支援をする。

3. 計画

開発未経験かつ使用言語の知識も皆無の状態から単独での挑戦であるため、ChatGPT（GPT-3.5）を活用する。活動期間内での目標はアプリのリリースではなく、予算 0 円でアイデアを具現化し、プロトタイプを作成することである。特に、落とし物の報告はユーザーの善意に依存するため、投稿するハードルをできる限り下げられるよう、「手軽に投稿できる UI/UX の設計」に焦点を当てる。

使用するツールを以下に示す。

- 開発環境：Xcode
- 言語：Swift
- その他：Git, SourceTree, ChatGPT（GPT-3.5）

4. 開発方法

コーディングでは、以下の A または B の方法をとる。

- ChatGPT でコードの枠組みを生成し、その内容を ChatGPT に自ら解説させ、その後公式ドキュメント等を参考にコードを修正する
- 公式ドキュメント等を参考に自分でコード書く

Swift の知識が皆無だったため、生成したコードを ChatGPT に解説させて、その後も質問を繰り返すことで、文法やフレームワークについての理解を深めた。また、個人で効率的に開発するため、Git やその他ツールを活用し、開発状況を管理した。

5. 活動経過

総活動時間は 196 時間、活動経過は以下の通りである。

- [1] アプリの方針の決定
- [2] 練習用環境の構築, Git の勉強
- [3] 本番環境の構築
- [4] コーディング
- [5] ポスター制作
- [6] デモ動画制作

6. 実装内容

スプラッシュ画面（起動画面）、メイン画面、カメラ画面、投稿画面の 4 つの画面を作成した。各画面での実装機能を以下に示す。

■ スプラッシュ画面（起動画面）

■ メイン画面

地図の表示、位置情報の取得、現在地の表示・追跡、現在地に戻るボタン、投稿開始ボタンと画面遷移、投稿の反映(マップピンの表示)、投稿情報の閲覧(マップピン押下で吹き出しの表示)、住所のコピー

■ カメラ画面

再撮影、キャンセル、写真のトリミング、写真の圧縮、画面遷移

■ 投稿画面

日時と住所の自動入力、写真の反映、文字入力、戻る・キャンセルボタン(入力情報の削除)と画面遷移、投稿完了ボタンと画面遷移

<成果物>

アプリ名：『Itemmap』（アイテムマップ）

以下の URL にデモ動画を置く。（QR コードからも閲覧可能）

https://drive.google.com/file/d/16bCWZqoPGpc516t-1f_ItkX2D2Fnc2wA/view?usp=sharing



また、画面遷移と各機能の紹介を図 1 に示す。



図 1

<「手軽に投稿できる UI/UX の設計」における工夫点>

① 機能面

落とし物発見時に、歩きながらも即座にシャッターを押せるよう、ボタン1つでカメラが即起動する仕様にした。その後の投稿手続きも、投稿画面への写真の自動追加や、撮影時の日時と位置情報の取得と自動入力によって、簡略化した。また、投稿画面では複数の入力項目を作ること虽も可能だが、手軽さを重視するために、敢えて入力欄は1つに絞り、この入力欄も任意入力とした。

② デザイン面

指の動きを考慮したボタン配置、編集不可の項目の色など、配置や色の選択に特に気を配り、ユーザーが直感的に利用できるデザインを目指した。さらに、ロゴ、マップピン、吹き出し表示はオリジナルにデザインし、中でも、マップピンは、投稿された落とし物画像を、アイコンとして組み込むことで、探しやすさを向上させた。

その他の工夫点としては、現在地に戻るボタンや各画面でのキャンセルボタンなど、操作性を向上させるると便利な機能を実装したことや、各媒体に対応するデザイン設計、動作の高速化のための画像の圧縮処理の追加などが挙げられる。また、複数人での開発にも対応するよう、他人が見ても理解しやすいファイル構成を心がけた。

7. 活動結果

今回採用した SwiftUI や MapKit は比較的新しいフレームワークであり、日本語のドキュメントが少ないため苦労した。また、ChatGPT が生成するコードは不完全であるため、結局は自分で理解し修正することが必須であり、二度手間となることも多かった。しかし、ChatGPT に生成したコードを自ら解説させることは、文法の学習に非常に有用であり、予備知識ゼロで始めた Swift にも十分に慣れることができた。さらに、全くの無知であった開発環境の構築法や Git を用いたバージョン管理などについても、活動を通して学習でき、今回は個人開発だったが、GitHub 等のツールを用いた複数人での開発方法の概観を得ることができた。

8. 展望

アプリの本格的な運用を視野に入れると、落とし物情報のデータベース化とトランザクション管理、貴重品や個人情報分かるものの投稿の制限が必須となる。

また、他に追加できる機能として、投稿に対する他ユーザーによる追記や会話形式での情報共有、過去 24 時間の移動経路の表示と常時更新、落とし物に気が付いた時点での過去 24 時間の移動経路の保存と移動経路上の投稿の通知、落とし主側からの検索依頼の投稿、落とし物場所へのナビゲーションもしくは他の地図アプリとの連携等が挙げられる。

今回の活動では「0 円で開発」するため、課金せず ChatGPT (GPT-3.5) を使用したが、ChatGPT Plus (GPT-4.0) や GitHub Copilot を使用することでより効率よく開発が進んだだろう。今後も開発を継続する場合は、これらのツールや、11 月上旬にコーディング能力が GPT-4 を上回ったと話題になった Phind の使用等も検討したい。

ポスターセッションでは大変好評いただき、「リリース待ってます」という期待の声を多数頂戴した。現時点ではリリースの目処は立っていないが、リリースも視野に入れつつ必要事項をクリアし、便利な追加機能等も検討していきたい。