

## 福岡宗像サイクリング観光促進アプリの実施報告

佐藤乃果<sup>†</sup>, 嶺田あんず<sup>†</sup>, 田中康一郎<sup>†</sup>

(†九州産業大学理工学部情報科学科)

## 1 はじめに

近年、スマートフォンの普及に伴い、地域における観光情報の提供やサービス案内を目的としたモバイルアプリの活用が広がっている。中でも、サイクリング観光においては、移動経路や周辺施設の情報をリアルタイムに取得できるアプリが、観光客の利便性向上に寄与することが期待される。昨年度、福岡県宗像市を対象としたサイクリング観光促進用スマートフォンアプリ「むなチャリ！」のiOS版をリリースした[1]。本年度は、宗像市の要望に基づき、同アプリのAndroid版を開発している。本研究では、Ktorを用いた通信処理の実装やAndroidエンジンの採用による安全なHTTPS通信の確保など、Android版開発の実装方法とその課題解決について報告する。

## 2 開発環境

Android版アプリの開発にはKotlin 2.2.10を用い、開発環境としてAndroid Studio (Narwhal Feature Drop 2025.1.2 Patch 1)を使用している。サーバ環境として、AlmaLinux release 9.6上で動作するPHP 8.1.32 (CLI, Zend Engine v4.1.32, Zend OPcache v8.1.32) およびMariaDB 10.11.10を使用した。

## 3 設計

本アプリ「むなチャリ！」Android版は、サイクリングコースの閲覧[2]、観光地情報の確認、デジタルクーポンの利用などを主な機能として実装している。その中で、本研究ではユーザ登録処理におけるサーバ通信を対象とした。ユーザ登録処理では、初期設定画面において利用者が年齢や性別を入力し、その情報をサーバへ送信する。サーバはユーザIDを発行し、アプリはそのIDを端末内に保存することで今後のクーポン利用や利用履歴管理に用いる。

## 4 実装

本研究では、まず通信ライブラリとしてOkHttpを用いて実装を行った。しかし、OkHttpはJavaベースのライブラリであり、将来的なKotlin環境との親和性や拡張性を考慮すると、Kotlin製のライブラリであるKtorを用いた方が、今後の開発の観点から望ましいと判断した。

そこで、Ktorを用いた通信処理の実装を検討した。しかし、実際に設計するとCIOエンジンで通信処理を行うと、通信エラーが発生した。この原因は、CIOエンジンは認証機関のルート証明書互換性問題とCIOの証明書チェックの厳密さが組み合わさってエラーが発生したと考えられる。

上記問題を回避するため、図1に示すような設定を行い動作させることができた。開発段階ではtrustManagerの設定を変更し、証明書の検証をスキップする方法を用いる。さらに、serverNameをnullに設定することで、証明書に記載されたドメイン名と実際のアクセス先ホスト名が一致しているかを検証する仕組みを無効化できる。しかしな

がら、このような検証を回避する設定はセキュリティ上のリスクが高く、本番環境においては適用すべきではない。そこでCIOエンジンの代わりに従来から利用されているAndroidエンジンを採用することで、これらの設定を行わなくても、スマートフォンに最初から備わっている安全に通信するための仕組みを活用し、信頼できるサーバとのHTTPS通信を安全に行えるようになった。

```
199
200 // HTTPS設定
201 https {
202     // カスタムTrustManagerを設定
203     trustManager = @SuppressLint( ...value = "CustomX509TrustManager") object {
204         javax.net.ssl.X509TrustManager {
205             @SuppressLint( ...value = "TrustAllX509TrustManager")
206             override fun checkClientTrusted(
207                 chain: Array<out java.security.cert.X509Certificate?>, authType: String?
208             ) {
209                 // クライアント証明書のチェックをスキップ
210             }
211
212             @SuppressLint( ...value = "TrustAllX509TrustManager")
213             override fun checkServerTrusted(
214                 chain: Array<out java.security.cert.X509Certificate?>, authType: String?
215             ) {
216                 // サーバー証明書のチェックを緩和
217                 // 実際の検証を行う場合は、ここでchainをチェック
218             }
219
220             override fun getAcceptedIssuers(): Array<java.security.cert.X509Certificate?> {
221                 return arrayOf()
222             }
223         }
224
225         // ホスト名検証を無効化
226         serverName = null
227     }
228
229 // 接続タイムアウト設定
230 endpoint {
231     connectTimeout = 30000
232     requestTimeout = 30000
233 }
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
```

図1: セキュリティリスクのある通信処理コード

## 5 まとめ

本研究では、福岡県宗像市を対象としたサイクリング観光促進アプリ「むなチャリ！」のAndroid版開発における通信処理の実装について報告した。開発当初はOkHttpを用いたが、将来的なKotlin環境との親和性を考慮し、Ktorを採用した。しかし、CIOエンジンでは認証機関のルート証明書互換性問題と証明書チェックの厳密さが組み合わさり、通信エラーが発生した。このエラーを回避するためには、セキュリティリスクを伴うコードを記述する必要があった。そこで、Androidエンジンを採用することで、安全にHTTPS通信を実現することができた。今後もリリースに向けて、両OS版アプリの機能安定性の確認や利用状況の分析を進めるとともに、得られたデータを活用した観光行動の可視化や地域施策への応用を検討する。

## 参考文献

- [1] 佐藤一晴, 井上誠斗, 安部萌, 田中康一郎. 福岡宗像サイクリング観光促進アプリの開発. 2024年度第32回電子情報通信学会九州支部学生会講演会, p. D-44.
- [2] 日高祥平, 竹内勇人, 田中康一郎. モバイルOSごとに構築した動画閲覧機能の実装と評価. 2025年度電子情報通信学会九州支部学生会講演会, p. D-10.