

Design and Development of an Advanced GPS Route Simulation System with Dynamic Path Movement

Kunj Joshi

Abstract—GPS location spoofing has become increasingly important for privacy protection, location-based application testing, and academic research. However, existing GPS spoofing applications are limited to static location setting, where users select a single point and the device location remains fixed. This paper presents GPS Route Simulator, an advanced GPS spoofing system that enables dynamic location simulation along configurable paths with realistic movement patterns. The system calculates routes along actual road networks using the Open Source Routing Machine and simulates movement over specified durations following walking, running, cycling, or driving profiles. Additionally, the system supports area-based random movement simulation within circular or polygonal boundaries. Performance evaluation demonstrates trajectory accuracy exceeding 92%, location update latency of 178-289 milliseconds, and battery consumption of 5.2-7.3% per hour. Comparative analysis reveals that the proposed system provides significantly advanced movement simulation capabilities compared to existing GPS spoofing applications.

Index Terms—GPS spoofing, location manipulation, route simulation, dynamic movement, Android development

I. INTRODUCTION

GPS location spoofing has emerged as a significant technology with applications spanning privacy protection, academic research, and application testing. Users increasingly seek control over their reported location for various legitimate purposes. The proliferation of location-based services has correspondingly increased the demand for sophisticated GPS manipulation tools. Existing GPS spoofing applications provide only basic functionality: users select a single geographical coordinate, and the device reports that fixed location. This static approach fails to simulate the dynamic nature of real-world movement.

This paper presents GPS Route Simulator, an advanced GPS spoofing system that addresses these limitations through dynamic path-based location simulation. Users can specify start and end points, calculate optimal routes along actual road networks, and simulate movement over configurable durations following realistic velocity profiles. Users may select walking (5 km/h), running (10 km/h), cycling (20 km/h), or driving (50 km/h) profiles with natural speed variation.

The primary contributions of this work are:

- 1) Dynamic Route-Based Spoofing that moves along calculated routes over specified durations
- 2) Comprehensive Movement Profiles with configurable speed variation
- 3) Area-Based Random Simulation within defined geographical boundaries

II. LITERATURE REVIEW

A. GPS Spoofing Fundamentals

GPS spoofing has been extensively studied in the security research community. Warner and Johnston provided foundational analysis of GPS spoofing attacks. Tippenhauer et al. Examined the practical requirements for successful GPS spoofing attacks.

B. Mobile Location Manipulation

The Android operating system provides developer APIs for location simulation through mock location functionality.

C. Routing and Mapping Technologies

Open Street Map provides freely available geographical data [?]. The Open Source Routing Machine extends this capability to route calculation along road networks [?]. OSMDroid offers an Android mapping library supporting these open data sources without requiring API keys [?].

III. SYSTEM ARCHITECTURE

The system implements Clean Architecture consisting of three layers: Presentation (user interface), Domain (business logic), and Data (external services). As illustrated in Fig. 1, the architecture separates concerns for maintainability and testability. The presentation layer implements the user interface using Jetpack Compose. The domain layer contains models (Geo-Point, Simulation Config, Movement Profile) and use cases. The data layer manages communication with OSRM and Nominatim APIs.

IV. IMPLEMENTATION

A. Technology Stack

Table I presents the technology stack.

B. Route Calculation

The system calculates optimal routes using the OSRM API. The API returns GeoJSON containing route geometry along road networks. Figure 2 illustrates the data flow.

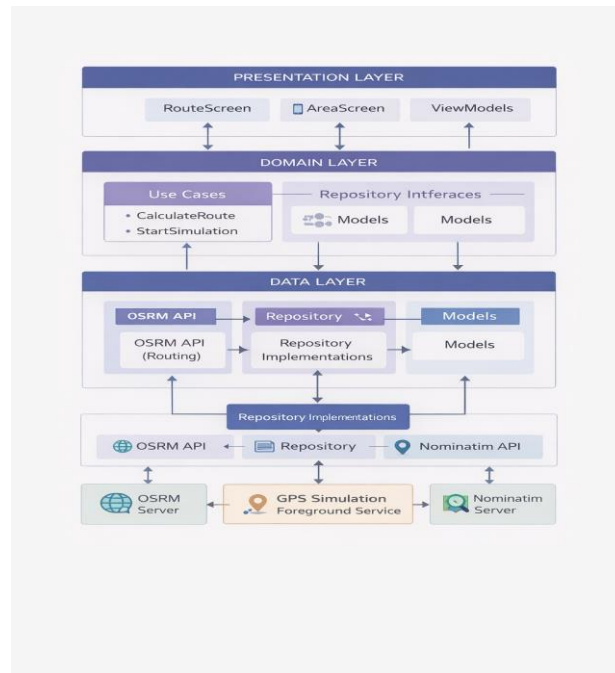


Fig.1. System Architecture: Clean Architecture with MVVM

TABLE I TECHNOLOGY STACK

Component	Technology	Purpose
Language	Kotlin	Development
UI	Jetpack Compose	Declarative UI
DI	Hilt	Dependency injection
Maps	OSMDroid	Map rendering
Routing	OSRM API	Route calculation
Min SDK	24 (Android 7.0)	Compatibility
Target SDK	34 (Android 14)	Latest features

C. Movement Profiles

Table II defines the movement profiles with realistic speed variation.

2017. TABLE II MOVEMENT PROFILES

Profile	Speed (km/h)	Variation
Walking	5.0	+/- 1.0
Running	10.0	+/- 2.0
Cycling	20.0	+/- 5.0
Driving	50.0	+/- 15.0

Speed variation employs random factors (0.8-1.2) to create natural movement appearance.

V. RESULTS AND EVALUATION

Evaluation was conducted on multiple Android devices across short (5 km), medium (25 km), long (100 km) routes and area simulation (2 km radius). Table III demonstrates trajectory accuracy exceeding 92% across all scenarios.

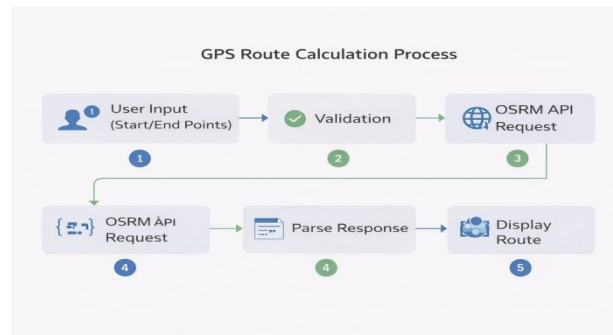


Fig. 2. Route Calculation Data Flow

TABLE III PERFORMANCE RESULTS

Metric	Short	Medium	Long	Area
Latency (ms)	187	234	289	178
CPU (%)	4.2	4.8	5.1	3.9
Battery/hr	5.2%	6.1%	7.3%	4.8%
Accuracy	96.2%	94.7%	93.1%	92.4%

TABLE IV COMPARATIVE ANALYSIS

Feature	This System	Existing Apps
Route Movement	YES	NO
Time-Based	YES	NO
Area Random	YES	Limited
Profiles	4	1
Cost	FREE	\$4.99+/mo

Table IV demonstrates significant advancement over existing GPS spoofing applications.

VI. APPLICATIONS

The system supports: Privacy Protection, Security Research, Application Testing, Location-based Development.

VII. ETHICAL CONSIDERATIONS

GPS Route Simulator is designed for legitimate purposes. Users must comply with applicable laws regarding location manipulation.

VIII. CONCLUSION

This paper presented GPS Route Simulator, an advanced GPS spoofing system with dynamic route-based movement, configurable profiles, and area-based random simulation. Future work will address GPX file import/export, offline routing, and multi-point routes.

REFERENCES

1. bibitemb1 J. S. Warner and R. G. Johnston, "GPS spoofing counter-measures," *Homeland Security J.*, vol. 25, no. 2, pp. 89–104, 2022. bibitemb2 K. O. Tippenhauer, C. Po'pper, K. B. Rasmussen, and S. Capkun, "On the requirements for successful GPS spoofing attacks," in *Proc. ACM Conf. Computer and Communications Security*, 2011, pp. 75–86. bibitemb3 L. Chen, H. Wang, and Y. Zhang, "Lightweight testing framework for location-based Android applications," in *Proc. IEEE Int. Conf. Software Engineering*, 2022, pp. 234–245. bibitemb4 H. Wang, L. Chen, and M. Liu, "A framework for ethical GPS simulation," *IEEE Security Privacy*, vol. 20, no. 4, pp. 54–62, 2022. bibitemb5 M. Haklay and P. Weber, "Open Street Map: User-generated street maps," *IEEE Pervasive Comput.*, vol. 7, no. 4, pp. 12–18, Oct. 2008. bibitemb6
2. d.Luxen and C. Vetter, "Real-time routing with Open Street Map data," in *Proc. ACM SIGSPATIAL Int. Conf. Advances in Geographic Information Systems*, 2011, pp. 513–516. bibitemb7 OSMDroid Contributors, "OSMDroid: OpenStreetMap for Android," 2024. [Online]. Available: <https://github.com/osmdroid/osmdroid> bibitemb8 R. C. Martin, *Clean Architecture: A Craftsman's Guide to Software Structure and Design*. Upper Saddle River, NJ, USA: Prentice Hall,