فصلنامه علمی - پژوهشی فرماندهی و کنترل، سال هشتم، شماره چهار، زمستان ۱۴۰۳

طراحی و پیاده‌سازی سامانه ردیاب وسایل نقلیه مبتنی بر پروتکل MQTT اینترنت اشیاء تحت سیستم‌عامل اندروید

رضا صالحی[۱]، محمدحسین مدنی*[۲]

تاریخ دریافت: ۱۴۰۳/۱۰/۰۹ تاریخ پذیرش: ۱۴۰۴/۰۶/۰۳

چکیده

امروزه موقعیت‌یابی پیوسته و دقیق یکی از نیازهای اساسی حوزهٔ حمل‌ونقل هوشمند و سامانه‌های ناوبری است. اینترنت اشیا با فراهم‌سازی بستر ارتباطی میان اشیای هوشمند، نقش مهمی در توسعه سامانه‌های نوین ردیابی وسایل نقلیه ایفا می‌کند. در این پژوهش، یک سامانه نرم‌افزاری ردیاب وسایل نقلیه مبتنی بر پروتکل پیام‌رسان MQTT و مدل شبکه چهارلایه TCP/IP تحت سیستم‌عامل اندروید طراحی و پیاده‌سازی شده است. ساختار سامانه از سه مؤلفه اصلی شامل سرور، اپلیکیشن کاربر و مدیر تشکیل شده است. اپلیکیشن کاربر موقعیت مکانی را از طریق ناوبری ماهواره‌ای و مکان‌یابی سلولی تعیین و داده‌ها را به کارگزار MQTT مستقر بر روی سرور ارسال می‌کند. کارگزار MQTT با ایجاد پل ارتباطی میان کاربر و مدیر، داده‌ها را در پایگاه‌داده ذخیره کرده و به اپلیکیشن مدیر منتقل می‌کند تا اطلاعات مکانی کاربران بر روی نقشه نمایش داده شود. نتایج آزمایش‌های میدانی سامانه در دو محیط با شرایط چندمسیری بالا و پایین نشان می‌دهد، دقت موقعیت‌یابی با معیار RMSE به ترتیب برابر۲.۳۳۷ و ۱.۴۱۸ متر برای هزار نمونه است. پیاده‌سازی این سامانه به‌صورت کاملاً نرم‌افزاری انجام شده و نتایج حاصل، کارایی سامانه را در افزایش دقت موقعیت‌یابی، حذف وابستگی به سخت‌افزار اختصاصی، کاهش هزینه پیاده‌سازی و پایداری در ارتباطات شبکه‌ای نرم‌افزاری تأیید می‌کند.

واژگان کلیدی: اینترنت اشیاء (IoT)، پروتکل پیام‌رسان MQTT، شبکه چهارلایه TCP/IP، ردیابی وسایل نقلیه، موقعیت‌یابی ترکیبی.

[۱] کارشناس ارشد، مهندسی برق- مخابرات سیستم، شرکت ارتباطات سیار ایران rezasalehiii۹۶@gmail.com

[۲] *(نویسنده مسئول)، استاد و عضو هیئت علمی دانشگاه صنعتی مالک اشتر تهران madani۱۳۵۵@yahoo.com

[Downloaded from ic4i-journal.ir on 2026-10-21]

۱. مقدمه

موقعیت‌یابی امروزه به طور گسترده در کاربردهای عمومی و سامانه‌های ناوبری مورداستفاده قرار می‌گیرد. وسایل نقلیه زمینی به سطح بالایی از دقت نیاز دارند تا بتوانند مسیرهای از پیش تعریف‌شده را دنبال کرده و از محدوده‌های مجاز خارج نشوند. روش‌های موقعیت‌یابی در شبکه‌های سلولی مبتنی بر اصول مکان‌یابی هستند؛ به این صورت که گیرنده با استفاده از داده‌های دریافتی از یک یا چند فرستنده مرجع، موقعیت خود را با الگوریتمی مشخص محاسبه می‌کند. فرستنده‌های مرجع می‌توانند ماهواره‌های ناوبری[۱] یا ایستگاه‌های پایه سلولی[۱] باشند. باتوجه‌به پوشش جهانی و دقت بالا، ناوبری ماهواره‌ای یکی از اصلی‌ترین فناوری‌های مکان‌یابی محسوب می‌شود، درحالی‌که مکان‌یابی مبتنی بر شبکه‌های تلفن همراه به‌عنوان راه‌حلی مکمل در شرایطی که دید ماهواره‌ای به دلیل انسداد سیگنال در محیط‌های شهری یا فضاهای بسته وجود ندارد، به کار می‌رود [۱].

روش‌های مختلفی برای تعیین موقعیت وجود دارد. از مهم‌ترین آن‌ها می‌توان به سه‌وجهی‌بندی[۲] و مجاورت[۳] اشاره کرد که معمولاً برای تخمین موقعیت دستگاه تلفن همراه بر اساس پارامترهای اندازه‌گیری سیگنال‌های رادیویی استفاده می‌شوند [۲].

اینترنت اشیاء[۴] (IoT) مفهومی است که ارتباط و تبادل داده میان دستگاه‌های هوشمند را از طریق اینترنت برای پایش، شناسایی و مدیریت خودکار محیط‌های ناهمگن فراهم می‌کند. اینترنت اشیاء با ادغام حسگرها، محرک‌ها، درگاه‌ها و نرم‌افزارها، زمینه‌ساز ایجاد سامانه‌های کنترل پیشرفته و کاهش مداخله انسان می‌شود. هر «شیء» در اینترنت اشیاء می‌تواند یک واحد فیزیکی دارای پردازنده، حسگر و محرک باشد که از طریق اتصال سیمی یا بی‌سیم به شبکه متصل است. این اشیا با استفاده از شناسه‌های یکتا (ID) در شبکه شناسایی و مدیریت می‌شوند [۳].

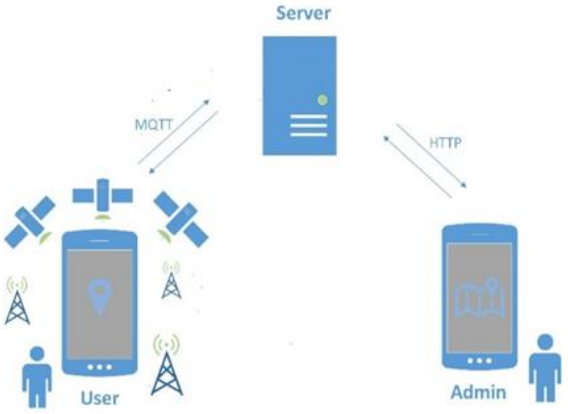
در معماری اینترنت اشیا، انتقال پیام میان دستگاه‌ها نقشی اساسی دارد، زیرا هر گره برای کنترل سامانه باید بتواند دستورات را به سایر گره‌ها ارسال کند. یکی از پروتکل‌های سبک و کارآمد انتقال پیام در این حوزه، MQTT[۵] است که برای دستگاه‌های کم‌مصرف و شبکه‌های با پهنای باند محدود طراحی شده است. درحالی‌که پروتکل‌های دیگری مانند XMPP[۶] و CoAP[۷] نیز برای انتقال داده در محیط‌های خاص توسعه یافته‌اند، MQTT به دلیل سادگی، قابلیت اطمینان در تضمین کیفیت خدمات[۸] و کارایی بالا در ارتباطات ناهم‌زمان، در سامانه‌های اینترنت اشیا و پیام‌رسانی بلادرنگ کاربرد فراوانی دارد. بااین‌حال، حفظ ترتیب ارسال و تحویل صحیح پیام‌ها در صورت ارسال هم‌زمان چند پیام همچنان یک چالش محسوب می‌شود [٤].

در این پژوهش، یک سامانه ردیابی وسایل نقلیه مبتنی بر پروتکل MQTT و تحت سیستم‌عامل اندروید طراحی و پیاده‌سازی شده است. این سامانه از سه مؤلفه اصلی شامل اپلیکیشن کاربر، سرور[۹] مرکزی و اپلیکیشن مدیر تشکیل شده است. اپلیکیشن کاربر داده‌های مکانی دریافتی از ماهواره‌های ناوبری و شبکه‌های کمکی را به‌صورت مختصات جغرافیایی به پایگاه‌داده سرور ارسال می‌کند. کارگزار[۱۰] MQTT نصب‌شده بر روی سرور، پل ارتباطی میان کاربر و مدیر را فراهم کرده و داده‌ها را در پایگاه‌داده ذخیره می‌نماید. در مقابل، اپلیکیشن مدیر با اتصال به سرور، اطلاعات مکانی کاربران را دریافت و بر روی نقشه نمایش می‌دهد.

شکل ۱ نمای کلی از ساختار و نحوه ارتباط میان مؤلفه‌های سامانه پیشنهادی را نشان می‌دهد.

[۱] Global Navigation Satellite System
[۱] Base Station
[۲] Trilateration
[۳] Proximity
[۴] Internet of Things (IoT)
[۵] Message Queuing Telemetry Transport
[۶] Extensible Messaging and Presence Protocol
[۷] Constrained Application Protocol
[۸] Quality of Service (QoS)
[۹] Server
[۱۰] Broker

[Downloaded from ic4i-journal.ir on 2026-10-21]



شکل ۱. نمای کلی از ساختار و نحوه ارتباط مؤلفه‌های سامانه پیشنهادی.

به‌طورکلی، موضوعات موردبررسی این مقاله عبارت‌اند از:

- بررسی مدل شبکه مورداستفاده در اینترنت اشیاء
- تحلیل و تشریح پروتکل پیام‌رسان MQTT و سازوکار عملکرد آن
- طراحی و پیاده‌سازی نرم‌افزاری سامانه ردیاب تحت سیستم‌عامل اندروید[۱]
- تحلیل خطای موقعیت‌یابی در محیط‌های با چندمسیری[۲] کم‌وزیاد و بررسی خطای سرعت لحظه‌ای با معیار RMSE

ساختار مقاله به‌این‌ترتیب است: بخش ۱ مقدمه، بخش ۲ پیشینه تحقیق، بخش ۳ زیرساخت‌های اینترنت اشیاء و پروتکل MQTT، بخش ٤ طراحی و پیاده‌سازی سامانه موقعیت‌یاب، بخش ۵ ارزیابی تجربی سامانه، و در نهایت بخش ٦ نتیجه‌گیری است.

۲. پیشینه تحقیق

در حوزهٔ اینترنت اشیاء، مقالات [۵] و [٦] به بررسی معماری‌های لایه‌ای، پروتکل‌های ارتباطی و جنبه‌های امنیتی این فناوری پرداخته‌اند. این پژوهش‌ها ضمن ارائهٔ طبقه‌بندی جدیدی از فناوری‌های اینترنت اشیاء، برخی از کاربردهای بالقوه و تأثیرگذار آن بر زندگی انسان را نیز معرفی کرده‌اند. ازآنجاکه ایجاد ارتباط مؤثر میان اشیاء و تصمیم‌گیری بهینه بر اساس داده‌های جمع‌آوری‌شده از چالش‌های اساسی در اینترنت اشیاء است، در [۷] مجموعه‌ای از پروتکل‌های ارتباطی لایه‌ی کاربرد برای تحقق این هدف پیشنهاد شده است. در [۸] نیز علاوه بر بررسی قابلیت همکاری[۳] در اینترنت اشیاء، انواع فناوری‌ها و پروتکل‌های ارتباطی مبتنی **بر** IP [٤] تحلیل شده‌اند. همچنین در [۹] و [۱۰]، پروتکل‌های پیام‌رسانی از جمله MQTT موردبررسی قرار گرفته و نحوهٔ انتشار و اشتراک پیام‌ها و مدل کارگزار پیام‌رسان آن تشریح شده است. در ادامه، این پروتکل به‌عنوان پایهٔ ارتباطی سامانه پیشنهادی مورداستفاده قرار گرفته است.

در مطالعهٔ [۱۱]، از ارتباط خودرو با خودرو[۵] و خودرو با زیرساخت برای ردیابی وسایل نقلیه و ارائهٔ مسیریابی به کاربران استفاده شده است. این روش با استفاده از Android Things که به طور ویژه برای دستگاه‌های اینترنت اشیای کم‌مصرف و دارای حافظه محدود طراحی شده، پیاده‌سازی گردیده و یک شبکهٔ درهم‌تنیده[٦] ایجاد کرده است. در [۱۲] نیز یک اپلیکیشن اندرویدی برای امدادرسانی در شرایط اضطراری توسعه‌یافته است. این برنامه با استفاده از الگوریتم Dijkstra و رابط برنامه‌نویسی Google Maps API، نزدیک‌ترین مسیر ممکن را برای رسیدن سریع آمبولانس به محل حادثه تعیین می‌کند و در نتیجه، زمان امدادرسانی کاهش‌یافته و احتمال نجات جان افراد افزایش می‌یابد.

در پژوهش‌های [۱۳] تا [۱۷]، سامانه‌های ردیابی خودرو با استفاده از فناوری‌های ارتباطی پیامکی و GPRS[۷] و بر پایهٔ موقعیت‌یابی ماهواره‌ای GPS[۸] طراحی و ارزیابی شده‌اند. در این سامانه‌ها، از میکروکنترلر برای کنترل ماژول‌های GPS و GSM/GPRS استفاده شده است. سامانه ردیابی در فواصل زمانی مشخص، مختصات جغرافیایی را از طریق GPS دریافت کرده و از طریق GSM/GPRS به پایگاه‌داده مرکزی ارسال می‌کند. بررسی این روش‌ها نشان می‌دهد که نرخ تبادل داده پایین بوده

[۱] Android Operating System
[۲] Multipath
[۳] Interoperability
[۴] Internet Protocol
[۵] Vehicle to Vehicle (V۲V)
[۶] Interconnected Network
[۷] Global Positioning System
[۸] General Packet Radio Service

[Downloaded from ic4i-journal.ir on 2026-10-21]

و از پروتکل‌های پیام‌رسانی بهینه استفاده نمی‌شود؛ ازاین‌رو، این رویکردها از نظر کارایی ارتباطی با محدودیت مواجه‌اند. علاوه بر این، نیاز به طراحی سخت‌افزار اختصاصی باعث افزایش هزینهٔ پیاده‌سازی می‌شود. همچنین، دقت پایین مکان‌یابی ناشی از استفادهٔ صرف از فناوری ماهواره‌ای از نقاط ضعف این روش‌ها است. بر همین اساس، طراحی یک سامانهٔ نرم‌افزاری ردیابی وسایل نقلیه مبتنی بر فناوری‌های جدید و بهره‌گیری از تلفن‌های هوشمند می‌تواند خلأهای موجود را برطرف کرده و دقت و کارایی سامانه‌های ردیابی را بهبود بخشد.

۳. زیرساخت‌های اینترنت اشیاء

۱.۳. مدل شبکه TCP/IP

مدل شبکهٔ TCP/IP [۱] یکی از مهم‌ترین مدل‌های ارتباطی مورداستفاده در سامانه‌های اینترنت اشیاء است که مجموعه‌ای از پروتکل‌ها را در قالب چهارلایهٔ اصلی سازمان‌دهی می‌کند. این لایه‌ها از پایین‌به‌بالا شامل لایه دسترسی به شبکه، لایه اینترنت، لایه انتقال و لایه کاربرد هستند که در جدول ۱ نمایش داده شده‌اند [۱۸].

جدول ۱. دسته‌بندی پروتکل‌های مدل TCP/IP در چهارلایه [۱۸].

لایهTCP/IP	پروتکل‌های مرتبط با اینترنت اشیاء
لایه کاربرد	MQTT, HTTP, CoAP, AMQP, XMPP
لایه انتقال	TCP, UDP, DTLS
لایه اینترنت	IPV٤/٦, ٦LoWPAN, RPL
لایه دسترسی به شبکه	Ethernet, Wi-Fi, GSM, LTE, LPWAN

۱.۱.۳. لایه دسترسی به شبکه

این لایه شامل بخش فیزیکی ارتباط است و نحوهٔ اتصال هر دستگاه به شبکه را از طریق سخت‌افزار تعیین می‌کند. داده‌ها در این لایه به‌صورت باینری انتقال می‌یابند.

۲.۱.۳. لایه اینترنت

در این لایه، آدرس‌دهی منطقی و مسیریابی بسته‌های داده میان‌گره‌های مبدأ و مقصد انجام می‌شود. پروتکل‌های IPv٤ و IPv٦ متداول‌ترین گزینه‌ها برای آدرس‌دهی دستگاه‌های اینترنت اشیاء هستند.

۳.۱.۳. لایه انتقال

وظیفهٔ این لایه فراهم‌سازی ارتباط انتها به انتها[۲]، کنترل ازدحام و اطمینان از تحویل صحیح و به ترتیب داده‌ها است. پروتکل‌های TCP و UDP[۳] در این لایه عمل می‌کنند.

٤.۱.۳. لایه کاربرد

در این لایه، پیام‌رسانی و تعامل نرم‌افزارها با یکدیگر صورت می‌گیرد. پروتکل‌هایی نظیر MQTT و HTTP[٤] از مهم‌ترین گزینه‌ها برای پیاده‌سازی ارتباطات در سطح کاربرد محسوب می‌شوند.

۲.۳. پروتکل پیام‌رسانیMQTT

MQTT یک پروتکل سبک و کارآمد برای تبادل پیام در بستر شبکه‌های اینترنت اشیاء است. این پروتکل در سال ۱۹۹۹ طراحی شد و بر اساس الگوی ارتباطی انتشار/اشتراک[۵] عمل می‌کند [۱۹]. تمرکز اصلی در طراحی آن، کاهش مصرف پهنای باند، کاهش نیاز به منابع محاسباتی و اطمینان از تحویل مطمئن پیام‌ها است. ویژگی‌هایی مانند سادگی، سبک بودن، متن‌باز بودن و قابلیت اطمینان بالا، MQTT را به گزینه‌ای مناسب برای محیط‌های محدود و دستگاه‌های کم‌مصرف تبدیل کرده است.

MQTT معمولاً بر روی پروتکل TCP/IP اجرا می‌شود و از درگاه[٦] ۱۸۸۳ برای ارتباطات عادی و از درگاه ۸۸۸۳ برای ارتباطات امن مبتنی بر SSL/TLS[۷] استفاده می‌کند[۲۰].

معماری MQTT شامل سه جزء اصلی است (شکل ۲):

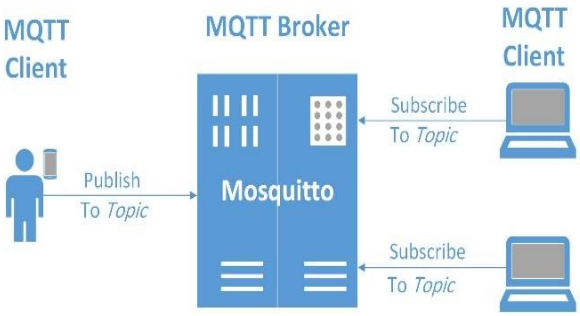
[۱] Transport Control Protocol/Internet Protocol
[۲] End-to-End
[۳] User Datagram Protocol
[٤] Hyper Text Transfer Protocol
[۵] Publish/Subscribe
[٦] Port
[۷] Secure Sockets Layer/Transport Layer Security

[Downloaded from ic4i-journal.ir on 2026-10-21]

۱- **ناشر**[۱]: تولیدکنندهٔ داده‌ها یا پیام‌ها،

۲-**کارگزار**[۲]: واسط مرکزی برای مدیریت پیام‌ها،

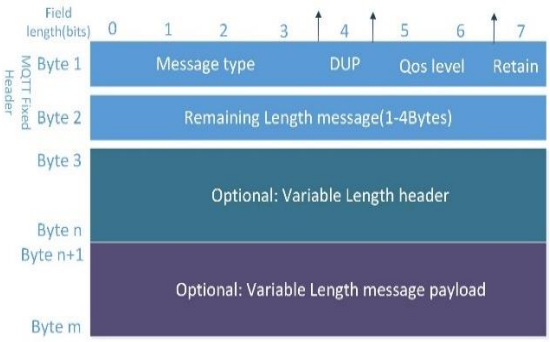
۳-**مشترک**[۳]: دریافت‌کنندهٔ داده‌ها بر اساس موضوعات انتخابی.



شکل ۲. معماری ساده پروتکل MQTT [۲۱].

در این ساختار، ناشر پیام‌ها را به کارگزار ارسال می‌کند و کارگزار بر اساس موضوع پیام‌ها را برای مشترکین مرتبط توزیع می‌نماید. [۲۱]. ساختار کلی پیام MQTT همانطور که در شکل ۳ مشاهده می‌شود، از سه بخش تشکیل شده است [۲۲]:

- **سرایند ثابت** (Fixed Header): شامل اطلاعات کنترلی (۲ بایت اجباری) است.
- **سرایند متغیر** (Variable Header): شامل پارامترهای اختیاری
- **بار پیام** (Payload): شامل داده‌های اصلی انتقال‌یافته است.

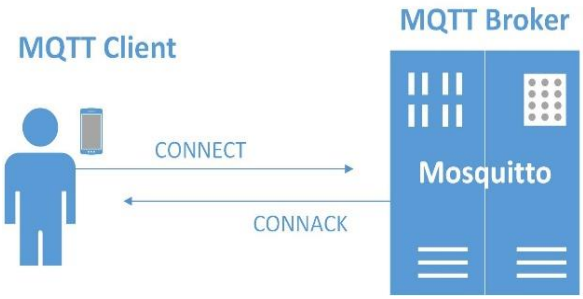


شکل ۳. ساختار پیام در پروتکل MQTT [۲۲].

برای شروع ارتباط، کاربر پیامی از نوع CONNECT به کارگزار ارسال می‌کند (شکل ٤). برخی از فیلدهای مهم در این پیام عبارت‌اند از:

- **شناسه کاربر**: شناسهٔ یکتای هر کاربر است.
- **جلسهٔ پاک‌سازی** (Clean Session): تعیین‌کنندهٔ نوع نشست (موقت یا پایدار) است.
- **نام کاربری و گذرواژه:** جهت احراز هویت به کار می‌رود.
- **پیام وصیت** (Will Message): اطلاع‌رسانی در صورت قطع ناگهانی ارتباط است.
- **زمان ماندگاری** (Keep Alive Time): مدت زمانی که بدون ارسال پیام، اتصال معتبر باقی می‌ماند. CT زمان اتمام اتصال و KAT زمان ماندگاری است که به‌صورت رابطه (۱) محاسبه می‌شود.

$$CT = 1.5 \times KAT \qquad (۱)$$



شکل ٤. برقراری اتصال بین کارگزار و کاربر در MQTT [۲۱].

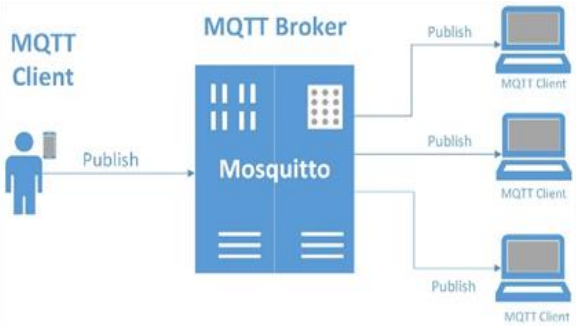
پس از برقراری ارتباط، کاربر می‌تواند پیام‌هایی از نوع PUBLISH ارسال کند (شکل ٥). پیام انتشار شامل اطلاعات زیر است:

- **موضوع** (Topic): کانال ارتباطی بین ناشر و مشترک است.
- **سطح کیفیت خدمات** (QoS): تعیین‌کنندهٔ میزان اطمینان از تحویل پیام (۰، ۱ یا ۲) است.

[۱] Publisher
[۲] Broker
[۳] Subscriber

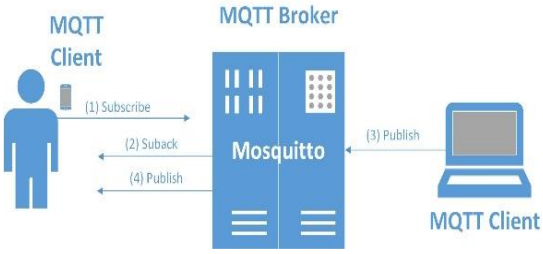
[Downloaded from ic4i-journal.ir on 2026-10-21]

- **پرچم بازیابی** (Retain Flag): آخرین مقدار پیام توسط کارگزار را ذخیره می‌کند.
- **ظرفیت بار** (Payload): شامل محتوای داده است.
- **شناسه بسته** (Packet Identifier)
- **پرچم تکرار** (DUP Flag): برای مدیریت تحویل پیام‌ها به کار می‌رود.



شکل ۵. فرآیند انتشار پیام در معماری MQTT [۲۱].

درصورتی‌که کاربر بخواهد پیام‌هایی با موضوع خاصی را دریافت کند، پیامی از نوع SUBSCRIBE به کارگزار ارسال می‌نماید. کارگزار پس از تأیید، پیام‌های مرتبط با آن موضوع را برای کاربر ارسال می‌کند (شکل ٦). در مقابل، پیام UNSUBSCRIBE برای لغو اشتراک به کار می‌رود.



شکل ٦. فرایند اشتراک و دریافت پیام‌ها در MQTT [۲۱].

بررسی ساختار مدل TCP/IP و عملکرد پروتکل MQTT در لایه‌ی کاربرد نشان می‌دهد که ماهیت اتصال گرا و انتشار/اشتراک این پروتکل، بستر مناسبی را برای طراحی سامانهٔ ردیاب وسایل نقلیه در محیط اینترنت اشیاء فراهم می‌کند. در بخش بعد، طراحی و پیاده‌سازی این سامانه توضیح داده می‌شود.

٤. طراحی و پیاده‌سازی سامانه

در این بخش، راهکار پیشنهادی برای طراحی و پیاده‌سازی یک سامانه موقعیت‌یاب وسایل نقلیه مبتنی بر اینترنت اشیاء و پروتکل پیام‌رسانی MQTT ارائه می‌شود. معماری کلی این سامانه شامل سه مؤلفه اصلی یعنی کاربر، سرور و مدیر است که تعامل میان آن‌ها در شکل ۷ نمایش‌داده‌شده است. وظیفه کاربر، جمع‌آوری و ارسال داده‌های مکانی به سرور است، سرور داده‌ها را ذخیره و پردازش می‌کند، و مدیر با استفاده از داده‌های ذخیره‌شده، موقعیت وسایل نقلیه را بر روی نقشه مشاهده و مدیریت می‌نماید.

١.٤. معماری کلی سامانه

ارتباط بین اجزای سامانه بر پایه مدل TCP/IP و با استفاده از پروتکل MQTT در لایه کاربرد برقرار می‌شود. در این ساختار، اپلیکیشن کاربر نقش ناشر، سرور نقش کارگزار، و اپلیکیشن مدیر نقش مشترک را دارد. داده‌های مکانی شامل طول و عرض جغرافیایی، ارتفاع، سرعت لحظه‌ای، زمان و شناسه کاربر، توسط اپلیکیشن کاربر تولید و از طریق MQTT به سرور ارسال می‌شوند. این داده‌ها در پایگاه‌داده ذخیره شده و توسط اپلیکیشن مدیر برای نمایش موقعیت کاربران مورداستفاده قرار می‌گیرند. روند کلی طراحی سامانه در ادامه تشریح شده است.

٢.٤. پیاده‌سازی سرور

سرور به‌عنوان هسته مرکزی سامانه، وظیفه مدیریت ارتباطات MQTT، ذخیره‌سازی داده‌ها و پاسخ‌دهی به درخواست‌های اپلیکیشن مدیر را بر عهده دارد. سیستم‌عامل انتخاب‌شده برای سرور، Ubuntu ٢٠،٠٤ لینوکس است و کارگزار MQTT به نام Mosquitto بر روی آن پیکربندی شده است. برای افزایش امنیت، سرور اوبونتو از فایروال UFW استفاده می‌کند تا اتصالات به سرویس‌های خاص از پیش تعریف شده مجاز باشد. به همین منظور پورت‌های ۱۸۸۳ و ۳۰۰۰ بر روی سرور تعریف و فعال شده است. مشخصات فنی سرور در جدول ۲ آورده شده است.

[Downloaded from ic4i-journal.ir on 2026-10-21]

جدول ۲. مشخصات سرور.

سیستم‌عامل	Ubuntu ۲۰,۰٤
کارگزار پیام MQTT	Mosquitto
درگاه های فایروال فعال شده	۱۸۸۳, ۳۰۰۰
نام پایگاه‌داده	SQLite
پلتفرم برای برنامه‌نویسی سمت سرور	NodeJS

شکل ۷. نمای کلی عملکرد سامانه موقعیت‌یاب.

برای ذخیره داده‌های مکانی، پایگاه داده‌ای سبک به نام SQLite ایجاد شده است. این پایگاه‌داده شامل جدولی با نام users.db است که دارای متغیرهای زیر است:

۱- شناسه کاربر (ID)

۲- سرعت (Speed)

۳-عرض جغرافیایی (Latitude)

۴- طول جغرافیایی (Longitude)

۵- ارتفاع (Altitude)

٦- نام کاربری (Username)

۷- زمان ثبت (Time)

ساختار جدول و نمونه داده‌های ثبت‌شده در شکل ۸ نشان‌داده‌شده است.

پیاده‌سازی سمت سرور با استفاده از NodeJS که یک پلتفرم برای برنامه‌نویسی سمت سرور است، انجام شده است. در این بخش، دو عملکرد اصلی پیاده‌سازی گردیده است:

- **عملکرد کاربر سمت سرور:** کلاس MqttHandler وظیفه اتصال به کارگزار Mosquitto، اشتراک در موضوع (تاپیک) به نام data با سطح کیفیت خدمات QoS=۲، دریافت پیام‌های حاوی اطلاعات مکانی از کاربران، و ذخیره آن‌ها در پایگاه‌داده را بر عهده دارد. هر پیام جدید منجر به درج یک سطر داده در پایگاه‌داده می‌شود.

شکل ۸. ساختار جدول پایگاه داده و نمونه داده‌های ذخیره‌شده

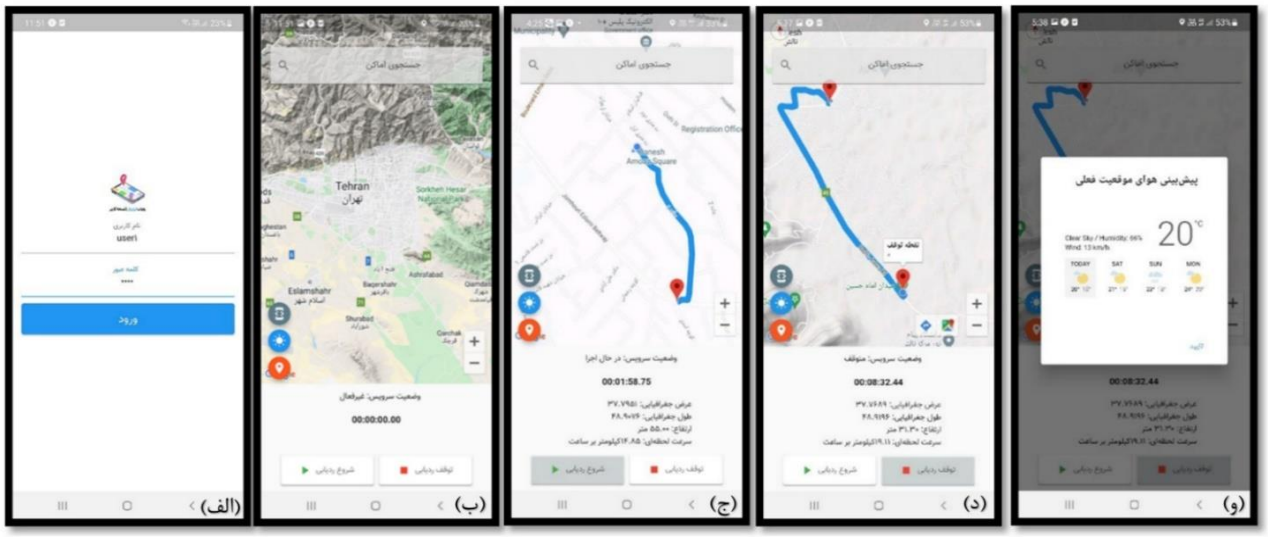
- **عملکرد مدیر سمت سرور:** کلاس MqttHandler داده‌های دریافتی ذخیره‌شده را بر اساس نام کاربری از پایگاه‌داده استخراج کرده و با قالب JSON در اختیار اپلیکیشن مدیر قرار می‌دهد تا موقعیت کاربران روی نقشه نمایش داده شود.

۳.٤. پیاده‌سازی اپلیکیشن نسخه کاربر

اپلیکیشن نسخه کاربر باهدف ردیابی وسایل نقلیه و ارسال اطلاعات موقعیت طراحی شده است. این اپلیکیشن از فنّاوری ارتباطی بی‌سیم Wi-Fi و شبکه سلولی (GSM/۳G/٤G) استفاده می‌کند. فنّاوری موقعیت‌یابی مورداستفاده این اپلیکیشن ترکیبی است، به عبارتی از چند فنّاوری جهت دست‌یابی به موقعیت مکانی بادقت بهینه بهره می‌برد. ازجمله این فنّاوری‌ها روش‌های سه‌وجهی‌بندی سلولی با GNSS و A-GNSS و همچنین استفاده از حسگرهای دستگاه تلفن همراه مانند قطب‌نما، اودومتر (مسافت‌سنج)، ژیروسکوپ (گردش نما)، شتاب‌سنج است.

[Downloaded from ic4i-journal.ir on 2026-10-21]

پروتکل ارتباطی شبکه این اپلیکیشن بر پایه مدل TCP/IP بوده و پروتکل پیام‌رسانی در این اپلیکیشن MQTT است. مصرف حافظه اپلیکیشن در حالت فعال‌بودن موقعیت‌یابی حدود ۸۰ مگابایت و در حالت غیرفعال کمتر از ۳۰ مگابایت است. این اپلیکیشن تحت سیستم‌عامل اندروید و با استفاده از فریم‌ورک می‌دهند. با فعال‌بودن سرویس مکان‌یابی، تابع getLocationUpdates تغییرات موقعیت را دریافت و علاوه بر نمایش روی نقشه، داده‌های طول جغرافیایی، عرض جغرافیایی، ارتفاع، سرعت و زمان را از طریق MQTT با QoS=۲ در تاپیک data منتشر می‌کند. شکل ۹ (ج) عملکرد سرویس ردیابی در



شکل ۹. اپلیکیشن موقعیت‌یاب نسخه کاربر.

Flutter توسعه‌یافته و زبان برنامه‌نویسی آن Dart است.

طراحی اپلیکیشن نسخه کاربر از صفحه ورود نام کاربری شروع می‌شود. در صفحه ورود[1]، اعتبارسنجی نام کاربری و رمز عبور انجام می‌شود و پس از تأیید، کاربر به صفحه اصلی منتقل می‌شود که شامل نقشه و گزینه‌های کنترل سرویس ردیابی است (شکل ۹ (الف) و ۱۰ (ب)).

صفحه اصلی شامل نقشه و امکانات ردیابی اپلیکیشن است. این صفحه امکان فعال‌سازی ردیابی کاربر را فراهم می‌کند. پس از فعال‌سازی سرویس ردیابی (گزینه شروع ردیابی)، تابع OnStart با بررسی دو شرط زیر، داده‌ها را به سرور ارسال می‌کند:

۱- حداقل جابه‌جایی ۱۵ متر نسبت به موقعیت قبلی

۲- سپری‌شدن حداقل ۳ ثانیه از آخرین ارسال داده

این شروط از ثبت داده‌های غیرضروری در حالت سکون جلوگیری کرده و میزان درخواست ارسال داده به سرور را کاهش حال اجرا را نشان می‌دهد.

تابع onStop نیز موجب توقف ردیابی می‌شود. (گزینه توقف ردیابی). وقتی ردیابی متوقف می‌شود، زمان‌سنج و سرویس دریافت موقعیت مکانی متوقف‌شده و مسیر پیمایش قابل‌مشاهده خواهد بود. شکل ۹(د) وضعیت سرویس متوقف‌شده را نشان می‌دهد.

کلاس MQTTWrapper مسئول مدیریت اتصال به کارگزار، اشتراک تاپیک، و انتشار پیام است. زمان ماندگاری در این کلاس ۶۰ ثانیه تعریف شده است. در صورت عدم ارسال پیام بیش از ۶۰ ثانیه، سرور ارتباط را قطع می‌کند.

اپلیکیشن علاوه بر قابلیت ردیابی، امکان پیش‌بینی وضعیت آب‌وهوا را نیز بر اساس طول و عرض جغرافیایی فعلی کاربر فراهم می‌کند (شکل ۹ (و)).

۴.۴. پیاده‌سازی اپلیکیشن نسخه مدیر

[1] Login Page

[Downloaded from ic4i-journal.ir on 2026-10-21]

اپلیکیشن نسخه مدیر جهت‌نمایش، نظارت و مدیریت کاربران ردیابی شده طراحی شده است. این اپلیکیشن نیز با Flutter توسعه‌یافته و از پروتکل‌های TCP/IP و MQTT برای تبادل داده با سرور استفاده می‌کند.

صفحه اصلی اپلیکیشن شامل فهرست کاربران ثبت‌شده است (شکل ۱۰(الف)). کاربر مدیر می‌تواند با انتخاب هر نام کاربری، به صفحه نقشه منتقل شود تا نقاط ردیابی شدهٔ آن کاربر را مشاهده نماید.

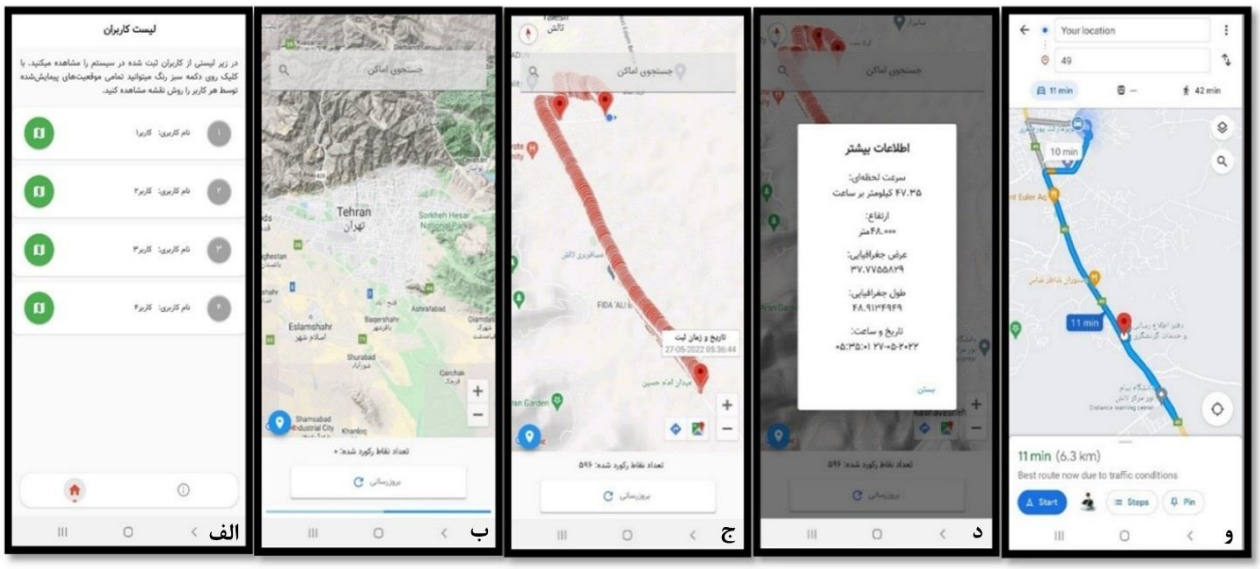
تابع getClientsData با ارسال درخواست POST به آدرس IP سرور در درگاه ۳۰۰۰، داده‌های مکانی مربوط به کاربر انتخابی را بازیابی کرده و آن‌ها را بر روی نقشه نمایش می‌دهد. برای هر نقطه، یک نشانگر همراه با اطلاعات زمان، سرعت، ارتفاع و مختصات جغرافیایی نمایش داده می‌شود (شکل ۱۰(ج) و ۱۰(د)).

همچنین با لمس هر نقطه، امکان مسیریابی به آن موقعیت مکانی از طریق نرم‌افزارهای مسیریاب فراهم شده است (شکل ۱۰(و)).

برای به‌روزرسانی نقشه، دکمه «به‌روزرسانی» در صفحه تعبیه شده است که با اجرای تابع refreshMarkers داده‌ها را مجدداً از سرور دریافت و نمایش می‌دهد (شکل ۱۰(ب)).

۵.٤. جمع‌بندی بخش طراحی

نتایج پیاده‌سازی نشان می‌دهد که استفاده از پروتکل MQTT در لایه کاربرد مدلTCP/IP، امکان تبادل داده سبک، سریع و مطمئن را بین کاربران و سرور فراهم می‌کند. اندازه‌گیری‌های انجام شده نشان می‌دهد میانگین زمان پردازش و ارسال هر بسته داده MQTT کمتر از نیم ثانیه است. طراحی مبتنی بر Flutter نیز توسعه هم‌زمان برای پلتفرم‌های اندروید و iOS (سیستم‌عامل دستگاه‌های شرکت اپل) را میسر ساخته است. ترکیب این دو فناوری موجب شده تا سامانه طراحی‌شده علاوه بر کارایی بالا، از سادگی، مقیاس‌پذیری و مصرف بهینه پهنای باند برخوردار باشد.



شکل ۱۰. اپلیکیشن موقعیت‌یاب نسخه مدیر.

۵. ارزیابی تجربی

به‌منظور ارزیابی عملکرد سامانه پیشنهادی و بررسی میزان اثربخشی آن در شرایط محیطی مختلف، آزمایش‌های میدانی در دو نوع محیط انجام شد. هدف از این ارزیابی، بررسی کیفیت ارتباطات شبکه‌ای، پایداری سامانه در شرایط قطع و وصلی شبکه، و تحلیل دقت موقعیت‌یابی در حضور و عدم حضور اثر چندمسیری است.

دو مدل محیطی مورد آزمایش عبارت‌اند از:

مدل ۱: محیط‌های با چندمسیری پایین (نظیر بزرگراه‌ها و مناطق برون‌شهری با تراکم کم موانع) با میانگین قدرت سیگنال به نویز $C/N_0 = ٤٢$ dB-Hz

[Downloaded from ic4i-journal.ir on 2026-10-21]

مدل ۲: محیط‌های با چندمسیری بالا (مناطق شهری با تراکم زیاد ساختمان‌های بلند و بازتاب سیگنال‌ها) با میانگین قدرت سیگنال به نویز C/N۰ = ۳۱ dB-Hz

۱.۵. تحلیل ارتباطات شبکه‌ای

برای ارزیابی عملکرد ارتباطی سامانه، کیفیت اتصال بین اپلیکیشن کاربر و کارگزار MQTT در هر دو مدل محیطی بررسی شد.

در مدل ۱ (محیط‌های برون‌شهری)، پوشش شبکه سلولی در برخی نواحی ضعیف بوده و گاهی ارتباط کاربر با شبکه به‌صورت موقت قطع می‌شود. در چنین شرایطی، ارتباط با کارگزار MQTT نیز مختل می‌گردد. هرچند در این مناطق، داده‌های مکانی بادقت بالا از طریق سامانه ناوبری ماهواره‌ای دریافت می‌شود، اما پایداری ارتباط شبکه‌ای پایین‌تر است.

برای جلوگیری از ازدست‌رفتن داده‌ها در هنگام قطع ارتباط، در سامانه پیشنهادی زمان ماندگاری برابر با ۶۰ ثانیه تعریف شده است. این پارامتر باعث می‌شود تا در صورت قطع موقت شبکه، پیام‌ها در صف نگهداری شده و بلافاصله پس از اتصال مجدد، به کارگزار ارسال شوند. این قابلیت مانع از حذف داده‌های موقعیت و تضمین‌کنندهٔ تداوم ثبت مسیر حرکتی است. در محیط‌های ناپایدارتر نیز می‌توان مقدار زمان ماندگاری را افزایش داد تا بازهٔ تحمل قطع ارتباط بیشتر شود.

بر اساس نتایج آزمایش‌های میدانی، بیشترین میزان افت ارتباط در تونل‌های طولانی برون‌شهری مشاهده شد؛ جایی که هم سیگنال‌های ماهواره‌ای و هم پوشش شبکه سلولی تضعیف شدید پیدا می‌کنند. در این حالت، نقاط موقعیت‌یابی در محدوده تونل ثبت نمی‌شوند، اما پس از بازیابی ارتباط، مسیر حرکت به‌صورت پیوسته با پلی‌لاین[۱] (قابلیت امکان متصل‌کردن دونقطه موقعیت ثبت شده پشت‌سرهم با یک خط) بر روی نقشه رسم می‌گردد تا مسیر طی شده به طور تقریبی قابل بازیابی باشد.

در مدل ۲ (محیط‌های شهری متراکم)، شبکه سلولی از پایداری بالاتری برخوردار بود و هیچ قطع ارتباط قابل‌توجهی مشاهده نشد. بااین‌حال، به دلیل وجود چندمسیری شدید، دقت موقعیت‌یابی کاهش یافت که در بخش بعدی تحلیل می‌شود.

۲.۵. تحلیل خطای موقعیت

یکی از مهم‌ترین منابع خطا در موقعیت‌یابی ماهواره‌ای، پدیدهٔ چندمسیری است. در این حالت، سیگنال‌های ارسالی از ماهواره پس از بازتاب از سطوح مختلف (مانند ساختمان‌ها یا موانع فلزی) با تأخیر به گیرنده می‌رسند و باعث انحراف در محاسبه موقعیت واقعی می‌شوند.

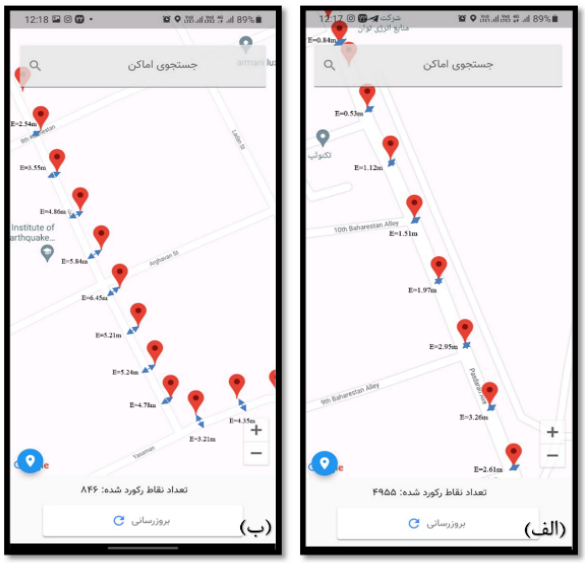
برای ارزیابی دقت سامانه در حضور اثر چندمسیری، از معیار خطای ریشه میانگین مربعات[۲] (RMSE) مطابق رابطه (۲) استفاده شده است:

$$RMSE_{Position} = \sqrt{\frac{1}{N}\sum_{k=1}^{N}[x_{measured}(k) - x_{true}]^2} \qquad (۲)$$

که در آن، $x_{measured}$ موقعیت واقعی و x_{true} موقعیت اندازه‌گیری‌شده توسط سامانه است.

۳.۵. تحلیل خطای موقعیت‌یابی در محیط‌های با چندمسیری پایین

در محیط‌های با چندمسیری پایین (مدل ۱)، آزمایش میدانی با ۱۰۰ نقطه موقعیت‌یابی انجام شد. شکل ۱۱(الف) موقعیت‌های واقعی و اندازه‌گیری‌شده را نشان می‌دهد.



[۱] PolyLine

[۲] Root Mean Square Error – RMSE

[Downloaded from ic4i-journal.ir on 2026-10-21]

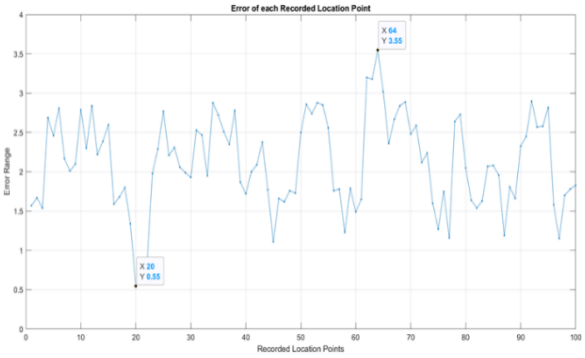
شکل ۱۱. خطای موقعیت اندازه‌گیری شده و موقعیت واقعی.

نتایج حاصل از این نقاط نشان داد که:

- **بیشترین خطا:** ۳٫۵۵ متر
- **کمترین خطا:** ۰٫۵۵ متر
- **میانگین خطا:** حدود ۱٫٤٥ متر

همان‌گونه که در نمودار شکل ۱۲ نشان‌داده‌شده است، تغییرات خطا در محدوده ۰ تا ٤ متر بوده و توزیع یکنواختی دارد. مقدار RMSE برای ۱۰۰ نقطه برابر با ۱٫٤٥٥٧ متر (مطابق رابطه ۳) و برای ۱۰۰۰ نقطه برابر با ۱٫٤۱۸ متر محاسبه شد. این نتایج بیانگر دقت بالای سامانه در محیط‌های با تداخل پایین است.

$$RMSE_{Position} = \sqrt{\frac{1}{100}\sum_{k=1}^{100}[x_{measured}(k) - x_{true}]^2} = 1.4557m \quad (۳)$$



شکل ۱۲. محدوده خطای ۱۰۰ نقطه موقعیت‌یابی شده در محیط‌های با چندمسیری پایین.

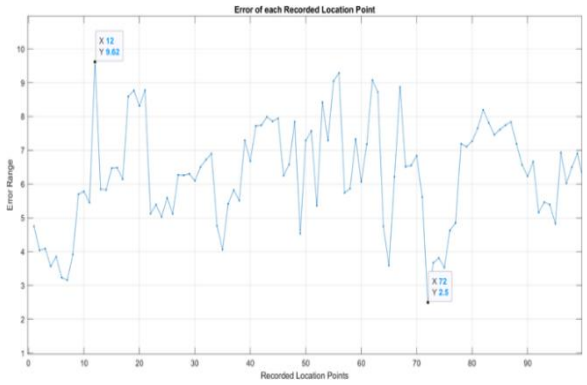
٤.٥. تحلیل خطای موقعیت‌یابی در محیط‌های با چندمسیری بالا

در محیط‌های شهری متراکم (مدل ۲)، همان‌طور که در شکل ۱۱(ب) مشاهده می‌شود، خطای موقعیت‌یابی به طور محسوسی افزایش‌یافته است. بر اساس ۱۰۰ نقطه آزمایش‌شده:

- بیشترین خطا: ۹٫٦۲ متر
- کمترین خطا: ۲٫٥ متر
- میانگین خطا: حدود ۲٫٥ متر

نمودار شکل ۱۳ توزیع خطاهای اندازه‌گیری‌شده را نشان می‌دهد. مقدار RMSE برای ۱۰۰ نقطه برابر با ۲٫٥۰۹٤ متر (مطابق رابطه ٤) و برای ۱۰۰۰ نقطه برابر با ۲٫۳۳۷٤ متر به دست آمد. این نتایج نشان‌دهنده تأثیر قابل‌توجه بازتاب‌ها و شکست سیگنال در محیط‌های با چندمسیری بالا است.

$$RMSE_{Position} = \sqrt{\frac{1}{100}\sum_{k=1}^{100}[x_{measured}(k) - x_{true}]^2} = 2.5094m \quad (۴)$$



شکل ۱۳. محدوده خطای ۱۰۰ نقطه موقعیت‌یابی شده در محیط‌های با چندمسیری بالا.

٥.٥. تحلیل مقایسه نتایج

نتایج تجربی نشان می‌دهد که دقت موقعیت‌یابی سامانه در محیط‌های مختلف با استفاده از روش ترکیبی GNSS، A-GNSS و حسگرهای دستگاه، عملکرد قابل‌قبولی داشته است. دقت موقعیت‌یابی ترکیبی در محیط‌های شهری به طور میانگین ۲ تا ۳ متر بهتر از سامانه‌های مبتنی بر GPS صرف بوده است.

به طور خلاصه:

- در محیط‌های با چندمسیری پایین، دقت موقعیت‌یابی بالاتر اما پایداری شبکه کمتر است.
- در محیط‌های با چندمسیری بالا، پایداری شبکه بیشتر ولی دقت موقعیت پایین‌تر است.

باوجود اثرات چندمسیری در مناطق شهری، مقادیر RMSE به‌دست‌آمده در حد قابل‌قبول برای سامانه‌های نرم‌افزاری تلفن همراه محسوب می‌شود. همچنین پروتکل MQTT توانسته است ارتباطی پایدار و مقاوم در برابر قطع‌های موقت شبکه برقرار کند.

٦. نتیجه‌گیری

در این پژوهش، یک سامانه نرم‌افزاری ردیاب وسایل نقلیه مبتنی بر اینترنت اشیاء و پروتکل پیام‌رسانی MQTT طراحی و

[Downloaded from ic4i-journal.ir on 2026-10-21]

پیاده‌سازی شد. سامانه پیشنهادی با ساختار سه‌لایه شامل کاربر، سرور و مدیر توسعه‌یافته و به‌صورت کاملاً نرم‌افزاری و بدون نیاز به سخت‌افزار اضافی بر بستر سیستم‌عامل اندروید اجرا می‌شود. هدف اصلی این سامانه، فراهم‌سازی بستری هوشمند و کارآمد برای ردیابی پیوسته وسایل نقلیه و تبادل داده‌های مکانی در شبکه‌ای سبک، مقیاس‌پذیر و کم‌هزینه بود.

در بخش پیاده‌سازی، از پروتکل MQTT به‌عنوان بستر ارتباطی سطح کاربرد استفاده شد تا ضمن کاهش حجم داده‌های ارسالی، قابلیت اطمینان تبادل پیام بین کاربر و سرور تضمین شود. همچنین برای موقعیت‌یابی دقیق‌تر، از روش ترکیبی ادغام ناوبری ماهواره‌ای، مکان‌یابی سلولی و حسگرهای تلفن همراه استفاده شد که سبب افزایش دقت مکانی در مقایسه با روش‌های مبتنی بر GPS صرف گردید.

نتایج آزمایش‌های میدانی در دو محیط با شرایط متفاوت چندمسیری نشان داد که دقت موقعیت‌یابی در محیط‌های با چندمسیری پایین (مانند بزرگراه‌ها) بسیار مطلوب بوده و مقدار RMSE برابر با ۱.۴۱۸ متر برای هزار نمونه نقطه حاصل شد. در مقابل، در محیط‌های با چندمسیری بالا (مانند مناطق شهری متراکم)، مقدار RMSE برابر با ۲.۳۳۷ متر به دست آمد که معادل حدود ۶۰ درصد افزایش خطا نسبت به محیط‌های با چندمسیری پایین است. این نتایج بیانگر آن است که سامانه پیشنهادی توانایی عملکرد قابل‌قبول در محیط‌های واقعی با شرایط متنوع شبکه‌ای و جغرافیایی را دارد.

مزیت اصلی این سامانه در مقایسه با روش‌های مشابه، حذف وابستگی به سخت‌افزار جانبی، کاهش هزینه پیاده‌سازی، پایداری در ارتباطات شبکه‌ای و استفاده از پروتکل سبک MQTT برای تبادل داده‌ها است که موجب تأخیر کمتر در ارسال داده‌ها نسبت به پروتکل‌های مشابه شده است. علاوه‌برآن، طراحی رابط کاربری مناسب برای اپلیکیشن‌های کاربر و مدیر، قابلیت‌هایی نظیر مشاهده مسیر طی‌شده، نمایش زنده موقعیت روی نقشه، و پیش‌بینی وضعیت آب‌وهوا را فراهم کرده است.

در آینده، پیشنهاد می‌شود توسعه سامانه با افزودن قابلیت‌های یادگیری ماشین برای پیش‌بینی مسیر حرکت، بهینه‌سازی مصرف انرژی در شبکه MQTT، و به‌کارگیری الگوریتم‌های فیلتر کالمن جهت افزایش دقت موقعیت‌یابی دنبال شود. همچنین بهره‌گیری از پروتکل‌های امن مانند MQTT-S می‌تواند گام مؤثری در ارتقای امنیت و قابلیت اطمینان آن باشد.

۷. مراجع

[۱] M. Yassin and E. Rachid (۲۰۱۵), "A survey of positioning techniques and location based services in wireless networks," ۲۰۱۵ IEEE International Conference on Signal Processing, Informatics, Communication and Energy Systems (SPICES), pp. ۱–۵, doi: ۱۰,۱۱۰۹/SPICES.

[۲] J. A. del Peral-Rosado, R. Raulefs, J. A. López-Salcedo and G. Seco-Granados (۲۰۱۸), "Survey of Cellular Mobile Radio Localization Methods: From ۱G to ۵G," IEEE Communications Surveys & Tutorials, vol. ۲۰, no. ۲, pp. ۱۱۲۴–۱۱۴۸, Secondquarter, doi: ۱۰,۱۱۰۹/COMST..

[۳] T. Malche, P. Maheshwary and R. Kumar (۲۰۱۹), "Environmental Monitoring System for Smart City Based on Secure Internet of Things (IoT) Architecture," Wireless Personal Communications, vol. ۱۰۷, pp. ۲۱۴۳–۲۱۷۲, doi: ۱۰,۱۰۰۷/s۱۱۲۷۷-۰۱۹-۰۶۳۷۶-۰.

[٤] H. C. Hwang, J. Park and J. G. Shon (۲۰۱٦), "Design and Implementation of a Reliable Message Transmission System Based on MQTT Protocol in IoT," Wireless Personal Communications, vol. ۹۱, pp. ۱۷٦۵–۱۷۷۷, doi: ۱۰,۱۰۰۷/s۱۱۲۷۷-۰۱٦-۳۳۹۸-۲.

[۵] P. Sethi and S. R. Sarangi (۲۰۱۷), "Internet of Things: Architectures, Protocols, and Applications," Journal of Electrical and Computer Engineering, vol. ۲۰۱۷, Article ID ۹۳۲٤۰۳۵, ۲۵ pages, doi: ۱۰,۱۱۵۵/۲۰۱۷/۹۳۲٤۰۳۵.

[٦] E. Cavalcante, M. P. Alves, T. Batista, F. C. Delicato and P. F. Pires (۲۰۱۵), "An analysis of reference architectures for the internet of things," ۲۰۱۵ ۱st International Workshop on Exploring Component-based Techniques for Constructing Reference Architectures (CobRA), pp. ۱–٤.

[۷] S. Saritha and V. Sarasvathi (۲۰۱۷), "A study on application layer protocols used in IoT," ۲۰۱۷ International Conference on Circuits, Controls, and Communications (CCUBE), pp. ۱۵۵–۱۵۹, doi: ۱۰,۱۱۰۹/CCUBE.

[۸] M. Noura, M. Atiquzzaman and M. Gaedke (۲۰۱۹), "Interoperability in Internet of Things: Taxonomies and Open Challenges," Mobile Networks and Applications,

[Downloaded from ic4i-journal.ir on 2026-10-21]

[۱۸] A. Gerber and J. Romeo (n.d.), "Connecting all the things in the Internet of Things," IBM Developer. Available: https://developer.ibm.com/articles/iot-lp۱۰۱-connectivity-network-protocols/.

[۱۹] N. Naik (۲۰۱۷), "Choice of effective messaging protocols for IoT systems: MQTT, CoAP, AMQP and HTTP," in Proceedings of the IEEE International Systems Engineering Symposium (ISSE), pp. ۱–۷, doi: ۱۰,۱۱۰۹/syseng.

[۲۰] A. Banks and R. Gupta (۲۰۱٤), "MQTT Version ۳,۱,۱," OASIS Standard, October ۲۹. Accessed: October ۱۰, ۲۰۲۰. Available: http://docs.oasis-open.org/mqtt/mqtt/v۳,۱,۱/os/mqtt-v۳,۱,۱-os.html.

[۲۱] HiveMQ Team (۲۰۲۰), "MQTT Essentials E-Book," October ۲۰.

[۲۲] T. Jaffey (۲۰۱٤), "MQTT and CoAP, IoT Protocols," accessed August ۲۳, ۲۰۲۰. Available: https://eclipse.org/٦۱٥community/eclipsenewsletter/۲۰۱٤/february/article۲.php.

vol. ۲٤, pp. ۷۹٦–۸۰۹, doi: ۱۰,۱۰۰۷/s۱۱۰۳٦-۰۱۸-۱۰۸۹-۹.

[۹] B. Mishra and A. Kertesz (۲۰۲۰), "The Use of MQTT in M۲M and IoT Systems: A Survey," IEEE Access, vol. ۸, pp. ۲۰۱۰۷۱–۲۰۱۰۸٦, doi: ۱۰,۱۱۰۹/ACCESS.۲۰۲۰,۳۰۳٥۸٤۹.

[۱۰] M. Bender, E. Kirdan, M.-O. Pahl and G. Carle (۲۰۲۱), "Open-Source MQTT Evaluation," ۲۰۲۱ IEEE ۱۸th Annual Consumer Communications & Networking Conference (CCNC), pp. ۱–٤, doi: ۱۰,۱۱۰۹/CCNC٤۹۰۳۲,۲۰۲۱,۹۳٦۹٤۹۹.

[۱۱] V. V. R. Dantu, V. V. S. Sai Dasaradha and P. Sasikumar (۲۰۲۱), "Unified Automotive Location Tracking Using Android Things (IoT)," Wireless Personal Communications, vol. ۱۲۰, pp. ٦۳–۷۹, doi: ۱۰,۱۰۰۷/s۱۱۲۷۷-۰۲۱-۰۸٤۳٤-y.

[۱۲] A. Y. Khan, M. Á. Luque-Nieto, S. Batool et al. (۲۰۲۲), "SOS Application Under Android: Help Pro," Wireless Personal Communications, doi: ۱۰,۱۰۰۷/s۱۱۲۷۷-۰۲۲-۰۹۹۸٥-٤.

[۱۳] S. Lee, G. Tewolde and J. Kwon (۲۰۱٤), "Design and implementation of vehicle tracking system using GPS/GSM/GPRS technology and smartphone application," ۲۰۱٤ IEEE World Forum on Internet of Things (WF-IoT), pp. ۳٥۳–۳٥۸, doi: ۱۰,۱۱۰۹/WF-IoT.۲۰۱٤,٦۸۰۳۱۸۷.

[۱٤] M. Alkhedher (۲۰۱۲), "Hybrid GPS-GSM localization of automobile tracking system," International Journal of Computer Science and Information Technology, vol. ۳, doi: ۱۰,٥۱۲۱/ijcsit.۲۰۱۱,۳٦۰٦.

[۱٥] B. Sahoo and S. Rath (۲۰۱۲), "Integrating GPS, GSM and Cellular Phone for Location Tracking and Monitoring."

[۱٦] P. K. Harshadbhai (۲۰۱۳), "Design of GPS and GSM Based Vehicle Location and Tracking System."

[۱۷] S. V. Vanmore, N. Jadhav, S. Nichal, M. Patil and A. Patil (۲۰۱۷), "Smart vehicle tracking using GPS," IRJET, vol. ۰٤, no. ۰۳, pp. ٦٦–٦۹.

[Downloaded from ic4i-journal.ir on 2026-10-21]