Reducing Costs and Energy Consumption in Marathon Tracking System: A Bluetooth Beacon Solution

FRANCESCO BACCHIN (2089122), University of Padua, Italy
LEILA DARDOURI (2080473), University of Padua, Italy

Supervising a marathon can be a complex task, particularly when the goal is to monitor each participant and prevent any cheating attempt, such as taking shortcuts. Moreover, collecting accurate and comprehensive data about the runners' performance is another critical aspect. This paper proposes an alternative solution to traditional RFID-based systems for tracking runners during marathons. The proposed system utilizes Bluetooth Low Energy (BLE) beacons placed at every kilometer around the marathon path, and an Android app on each runner's smartphone that listens for the beacons and records the time each beacon is detected. This approach eliminates the need for RFID tags and expensive readers. Furthermore, the challenge of battery life for smartphones is addressed avoiding the usage of GPS technology. The collected data is sent to a server for analysis, which checks the validity of the race by comparing the beacons IDs recorded by the app with the expected ones. The proposed system provides an efficient and cost-effective solution for tracking runners during marathons, while minimizing the impact on smartphone batteries.

Additional Key Words and Phrases: BLE beacons, bluetooth, wireless, Android, marathon tracking system

1 INTRODUCTION

With the rise in popularity of marathons in recent years, as evidenced by a 255% increase in marathon participation in the US from 1980 to 2021 [5], it is important to explore cost-effective methods for managing such events. While larger events, such as the annual New York City Marathon, may have ample funds to support sophisticated tracking systems, this may not always be the case for smaller events organized by minor organizations or cities, ranging from local charity marathons to half marathons that are open to non-professional runners.

The needs of these smaller events are different, and RFID tags have emerged as a popular solution. These tags, which are typically attached to a runner's shoe or race number bib, emit a unique code that is detected by radio receivers placed at strategic locations along the route. This system enables real-time tracking, as evidenced by the TCS New York City Marathon App, which displays the position of each runner in real-time [6]. However, while RFID tags are relatively inexpensive, the equipment required to scan them can be costly, amounting to thousands of dollars.

This expense can drive up entry fees and limit access to them. Moreover, smaller competitions may not require such a sophisticated system, but could benefit from an inexpensive and reliable way of tracking runners to ensure that they follow the prescribed route and record their overall timing. To address this need, we explored an alternative solution based on Bluetooth beacons - a type of Bluetooth Low Energy (LE) hardware transmitter that broadcasts its identifier. Bluetooth offers the advantage of enabling runners to use their own smartphones as tracking devices, rather than relying on additional support systems, thereby reducing costs and promoting ecological sustainability.

For devices emitting the signal, dedicated beacon devices or regular smartphones with a suitable application installed may be utilized. Our proposed solution adopts the latter approach to further reduce the overall cost of the system, although the detection range may be, as a result, somewhat compromised.

The paper is organized as follows: section 2 provides an overview of related works on the topic; section 3 outlines the functionalities offered by our proposed solution and how they are implemented; section 4 reports an usage example

Authors' addresses: Francesco Bacchin (2089122), University of Padua, Padua, Italy, francesco.bacchin.3@studenti.unipd.it; Leila Dardouri (2080473), University of Padua, Padua, Italy, leila.dardouri@studenti.unipd.it.

performed by us. In Section 5, we analyze the battery consumption of our proposed solution, including a comparison with other popular GPS-based options. Section 6 discusses potential future directions for this work. Finally, section 7 provides some concluding remarks.

2 RELATED WORKS

This section provides an overview of the primary proposed solutions for marathon timing systems documented in the literature. Each solution represents a distinct approach towards developing a precise, affordable and reliable timing system for marathons. In particular, this examination centers on the employment of Bluetooth Low Energy (BLE) technology, which is also the technology utilized in our proposed solution.

Perez-Diaz-de-Cerio et al. (2018) [2] developed a timing system that enables real-time monitoring of runners' positions through the utilization of mobile phones on the organizers' side, and BLE sensors on the participants. Then, they conducted a simulation of the system in a real-world scenario, utilizing data acquired from the Barcelona Marathon to obtain realistic mobility patterns. Their research also revealed that BLE can serve as a supplementary system for RFID timing and provide additional data, such as blood pressure information, in the transmitted data packets.

Chun-I Sun et al. (2019) [7] implemented a marathon timing technology using BLE communication technology. The authors perform pre-validation of the athletes' physiological sensory data that is sent out by the same timing system. The accuracy and performance of the timing and positioning technology are verified through various experiments, such as analysis of Received Signal Strength Indicator (RSSI), time synchronization using Network Time Protocol (NTP), and testing of BLE tag's battery life. The system also uses GPS signals to enhance the accuracy of time synchronization. The authors estimate that the system can provide a timing error of under ±156 ms for each athlete.

In Jussi's study (2021) [4], a Bluetooth-based timing solution was proposed and assessed for small sports events that employed smartphones as timing transponders. The solution included two applications: RaceConnect and RaceStation. The former was installed on each athlete's smartphone and transmitted some data, such as the event UUID, the hashed device serial code, and the runner's personal tracking id. The latter was installed on several smartphones near the finish line and acted as Bluetooth servers that listened to the broadcasted data from the athletes' devices. The RaceStation application recorded all readings from the session and sent them to an online collector that handled the time calculations to create the final race ranking. Results showed good reliability and precision compared to official RFID recorded times.

Tosi et al. (2017) [8] conducted a comprehensive analysis of the utilization of BLE in small, compact, and embedded sensors for various applications. The study reviewed the performance of BLE technology, including factors such as throughput, power consumption, latency, and maximum reachable range, with the objective of identifying the current limitations of this technology. The review concludes that actual applications of BLE technology typically exhibit limited throughputs of about 100 kbps and have a maximum reachable range of a few tens of meters, while the maximum number of nodes in the network is usually fewer than 10. Furthermore, power consumption and latency are dependent on numerous parameters.

Chokchai (2018) [1] proposed a cost-effective and high-performance UHF RFID system based on an Arduino-compatible IoT board for timing marathon competitions. The system consists of an antenna for the reader and the tag, a program for reading and writing data into the tags, and an Android application for displaying the results. The system demonstrated a coverage range of eight running lanes and achieved accurate classification of runners.

In this paper, we propose a novel solution for timing systems in marathon events, which utilizes Bluetooth technology. The proposed system involves the adoption of Bluetooth beacons as checkpoints along the marathon route, and the usage of a developed application that is installed on each participant's smartphone. The main goal of the system is to monitor each participant and prevent fraudulent activities, such as shortcut-taking.

3 FUNCTIONALITIES AND IMPLEMENTATION

The primary outcome of this work is the development of an Android Kotlin-written application. The app follows the Model-View-ViewModel (MVVM) architecture pattern and employs Jetpack Compose, a modern UI toolkit for building native Android applications using a declarative approach. To detect beacons, the application is integrated with the Android Beacon Library, an open-source library developed by Radius Networks for Android devices.

Additionally, we utilize the Firebase set of hosting services provided by Google. Specifically, the Firebase Authentication service is utilized to enable back-end user authentication, while the Firebase Realtime Database is employed as a cloud-hosted NoSQL database to store data in real-time. Additionally, the Cloud Functions for Firebase service is used to execute back-end JavaScript code in response to events triggered by HTTPS requests.

In order to use the application, users are required to register by providing their name, surname, email address and a chosen password, followed by the login process. This is a fundamental step to associate the runner with the detections made through the phone. Once logged in, the user can choose from the available marathons and will be prompted to grant the permissions needed by the application to scan for beacons. It should be noted that the password can be changed later by the user in the designated section of the application.

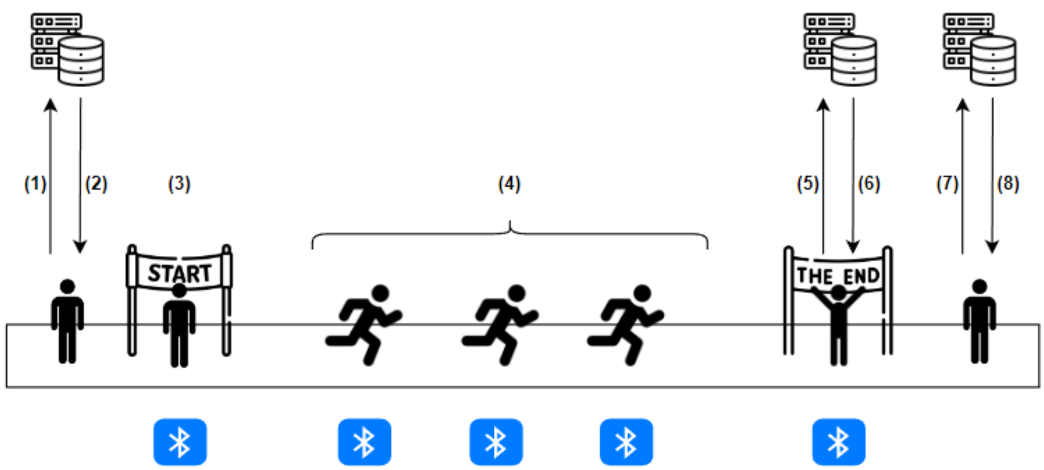


Fig. 1. Flow of interactions between the app and the server.

Upon selection of a marathon from the available options, the application sends an HTTPS request to the server, communicating the user's choice (Figure 1, point (1)). The server responds with the IDs of the first and last beacon (Figure 1, point (2)). This step is essential for initiating the stopwatch function when the phone detects the first beacon when the marathon starts (Figure 1, point (3)) and stopping it after the last one, at the end of the marathon. The app stores all detected beacons with their respective timestamps during the race in the smartphone's memory (Figure 1, point (4)) and pushes them to the server only after detecting the last beacon (Figure 1, point (5)). The last beacon is recognized by comparing the IDs of each beacon received by the application with the ID stored as the last beacon's ID.

This approach has the benefit of being unaffected by areas with poor signal strength, which could otherwise hinder the signal transmission of individual beacons. Additionally, it is more energy-efficient as fewer connections are made.

As soon as the list of detected beacons is received from the app, a Firebase function is triggered to determine the validity of the race. This is achieved by comparing the IDs of the detected beacons against the IDs of the beacons placed along the path. If all the IDs match, the race is considered valid; otherwise, it is not. The presence of unknown beacons does not affect the validity of the race, as they are simply disregarded. The server sends the outcome of the validation to the application (Figure 1, point (6)). Subsequently, users may request the current ranking of the marathon from the server (Figure 1, point (7)), and the server will respond accordingly (Figure 1, point (8)). To view the most up-to-date ranking, users can press the "refresh" button, since the ranking is updated continuously as other participants finish the race.

During the race, each time a beacon is detected, the system records the universally unique identifier (UUID) of the beacon, a timestamp obtained from the athlete's mobile phone, and the relative time. The relative time is calculated by subtracting the timestamp of the first beacon from the timestamp of the current beacon. These data are added to a list named "passedBeacons", which represents all the beacons collected by the athlete during the race. To determine the final ranking, the relative time is used instead of the absolute time. This eliminates the need to synchronize the timestamps of each participant's mobile phone to a common clock. By using relative time, we ensure that the time recorded for each participant is accurate and reliable, regardless of any discrepancies between their mobile phone clocks.

Once the RunMarathon application obtains the list of beacons from the server, it sets the UUID of the first and last beacons. The timer is automatically triggered as soon as the application detects the first beacon, which is located at the starting line and is potentially emitted by a simple mobile phone. At the start of the race, the phone begins transmitting the UUID of the first beacon, and the timer automatically starts in every RunMarathon application. The first beacon is also added to the passedBeacons list. The RunMarathon application is designed to continue detecting beacons even when the phone screen is locked, thanks to its ability to run in background. This feature helps to reduce battery consumption. When the application is running in the background, a notification is displayed to indicate its status to the user.

To ensure the accuracy and reliability of the time associated to the recorded intermediate beacons (i.e., those detected between the first and the last), the passedBeacons list only includes those detected within a distance of 2.5 meters from the athlete's phone. This is particularly important when moving at a faster pace, where delays in beacon detection can lead to inaccuracies. When a beacon is detected and added to the passedBeacons list, the stopwatch screen displays the corresponding beacon number and its associated relative time, as illustrated in Figure 2.

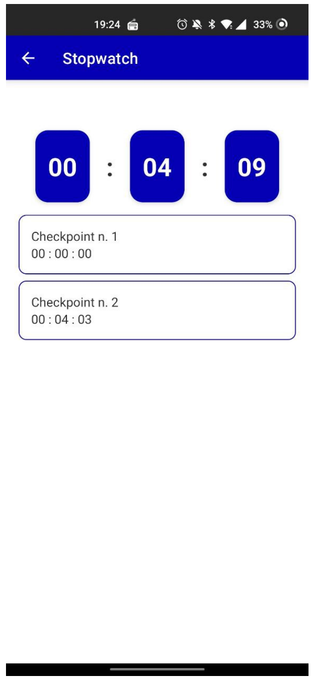


Fig. 2. Stopwatch screen during a marathon.

The same method of measurement applied to the intermediate beacons is also employed for the last beacon, situated at the finish line. To approximate the end of the race, we rely on the detection of each athlete within 2.5 meters from the finish line. As soon as the UUID of the last beacon is detected at a distance of less than 2.5 meters, the race concludes automatically. The athlete can then submit their recorded data, consisting on the passedBeacons list, to the server for validation purposes and to register their time in the final ranking.

To ensure accurate measurement of beacons during the race, it is necessary to apply some filtering techniques during the detection phase. All UUIDs associated with the race beacons must have the same first 24 digits. The last 8 digits are used to uniquely identify each beacon from the others required for the competition. Additionally, a beacon that has already been registered in the passedBeacons list will be automatically ignored if detected again, in order to avoid duplicates in the list.

In order to maintain uniformity in the level of sensitivity of the signal across all participants, each athlete is required to wear their phone in a designated phone armband on their right arm. This precaution is taken to prevent any potential signal attenuation or blockage due to objects obstructing the path between the phone and the beacon, which could otherwise vary among athletes. Additionally, the beacons will be placed on the right side of the path to enhance guidance towards the phone.

4 RETRIVED DATA AND RESULTS

To test the precision and reliability of the tracking and timing measurement of the system described earlier, we conducted a small-scale test. The considered path consisted of five beacons: three of which were intermediate, one was at the starting line, and one at the finish line. The beacons were placed on the right side of the path, with a distance of one kilometer each. To emit the beacons, we have used the Beacon Simulator app -available for free on the Play Store-installed on five Android smartphones. This simulator allows you to set parameters related to a beacon, including the transmitted UUID, and emulating its functioning. The two race participants were asked to place their phone on their right arm at shoulder height. Table 1 and table 2 show the results of the readings taken during the race of each athlete, compared to the actual time, recorded using a stopwatch.

Beacon	RunMarathon (mm:ss)	Actual time (mm:ss)	Error (s)	Beacon was detected
1	00:00	00:00	-	Yes
2	04:31	04:28	+3	Yes
3	09:59	10:01	-2	Yes
4	14:48	14:49	-1	Yes
5	20:12	20:10	+2	Yes

Table 1. Timing results of athlete 1.

Beacon	RunMarathon (mm:ss)	Actual time (mm:ss)	Error (s)	Beacon was detected
1	00:00	00:00	-	Yes
2	04:45	04:46	-1	Yes
3	10:05	10:08	-3	Yes
4	15:00	15:02	-2	Yes
5	21:02	21:00	+2	Yes

Table 2. Timing results of athlete 2.

The obtained results demonstrate that the application successfully detected all the beacons, and the time measurement error between the application and actual time may vary by ± 3 seconds. This variation is attributed to detection difficulties, as beacons may report distances that are either greater or smaller than the actual ones. The error is further compounded when the position is detected while the participant is in motion, as is the case in a marathon.

5 BATTERY CONSUMPTION

To assess the battery consumption of our application, we conducted a comparative analysis with two popular GPS-based running tracking applications, namely "adidas Running" and "Strava". We utilized two distinct smartphones: a Redmi Note 7 with a 4000 mAh LiPo battery, and a One Plus Nord with a 4115 mAh LiPo battery. The first one was purchased in 2019 and has a damaged battery that results in high power consumption, even under normal usage conditions. The second one is slightly newer, as it was purchased in 2020. The age of the smartphone is significant as batteries degrade over time and this affected the results retrieved.

Our measurements were performed with the objective of capturing the change in battery percentage over a 240-minute period, with both phones starting at 100% battery level. We ensured that no other application was running in the background, and the screens were locked during the tests. The battery level of both phones was checked after 30, 60, 120, and 240 minutes, and the results obtained are presented below.

	Battery decrease		
Smartphone	**RunMarathon**	**adidas Running**	**Strava**
Redmi Note 7	-6%	-9%	-8%
One Plus Nord	-2%	-3%	-4%

Table 3. Battery decrease comparisons results after 30 minutes.

	Battery decrease		
Smartphone	**RunMarathon**	**adidas Running**	**Strava**
Redmi Note 7	-10%	-19%	-15%
One Plus Nord	-5%	-7%	-9%

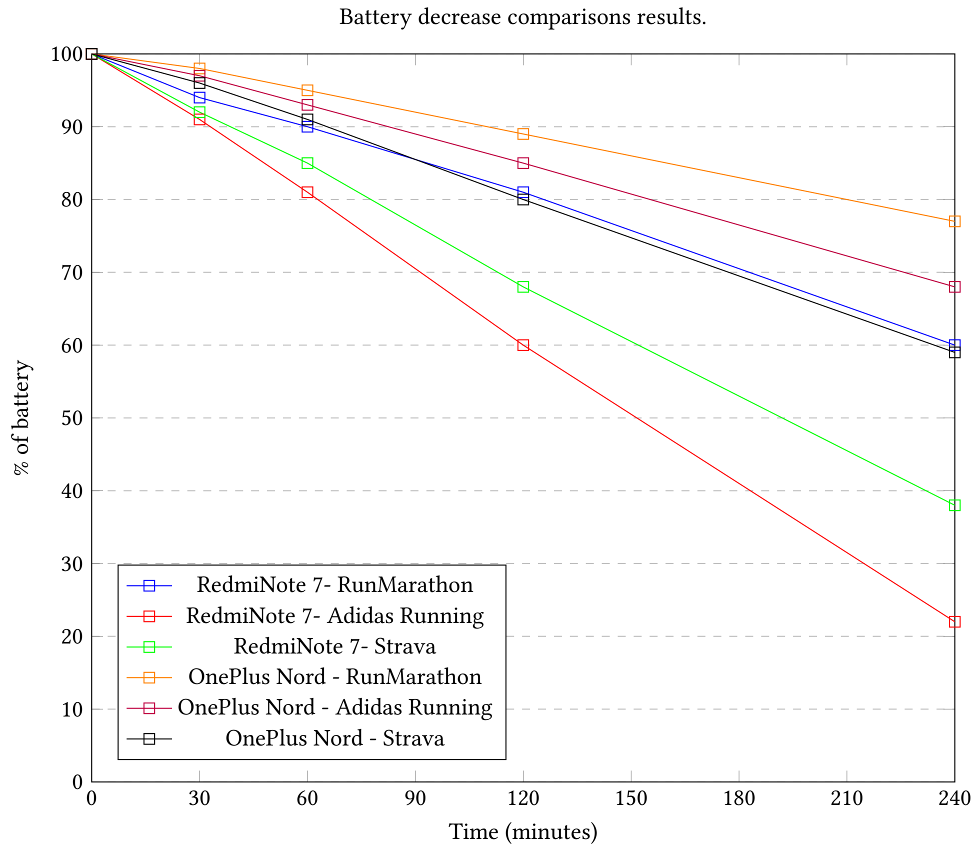
Table 4. Battery decrease comparisons results after 60 minutes.

	Battery decrease		
Smartphone	**RunMarathon**	**adidas Running**	**Strava**
Redmi Note 7	-19%	-40%	-32%
One Plus Nord	-11%	-15%	-20%

Table 5. Battery decrease comparisons results after 120 minutes.

	Battery decrease		
Smartphone	**RunMarathon**	**adidas Running**	**Strava**
Redmi Note 7	-40%	-78%	-62%
One Plus Nord	-23%	-32%	-41%

Table 6. Battery decrease comparisons results after 240 minutes.



The results obtained show that our application, RunMarathon, consumed less battery power than both the other two applications considered. For example, after 240 minutes of use, RunMarathon consumed 38% and 22% less battery power than "adidas Running" and "Strava" respectively, on the Redmi Note 7 smartphone. Whereas on the One Plus Nord smartphone, RunMarathon consumed 9% and 18% less battery power than "adidas Running" and "Strava", respectively.

6 POSSIBLE FURTHER DEVELOPMENTS

The main aim of the developed application was to test the efficiency of a Bluetooth-based tracking system. As a result, the application is still skeletal and could be improved in terms of both graphical design and functionalities. Additional desirable features for the application may include allowing users to search for available marathons, providing more detailed information about the athlete's performance (such as average speed, portion of path ran faster/slower, and calories burned), and integration with a smartwatch to track the heart rate. Additionally, incorporating more gamification features, such as badges, achievements, and leaderboards, could motivate users to continue training and participating in marathons, increasing their overall engagement with the application. Integration with social media could increase the application's overall reach and visibility, as it would allow users to share their progress and achievements with their followers.

Developing a Swift version of the application may also be necessary for wider adoption, as even though Android represents the majority with 72% of the market share, iOS accounts for a significant 27% of the total smartphone market share [3].

To reduce battery consumption even more, the current mechanism of perpetual scanning for beacons could be refined to scan only when a new beacon is supposed to be in range. This can be achieved by estimating the runner's speed and calculating the time it would take for him to reach the next beacon. The system can then go to sleep until the estimated time of arrival, thereby conserving battery life.

7 CONCLUSIONS

The analysis presented in Section 5 demonstrates that the proposed method reduces battery usage in mobile applications compared to GPS-based alternatives. Furthermore, the results of the tests described in Section 4 show that the Bluetooth-based system is a reliable and cost-effective alternative to RFID-based checkpoint tracking systems. The system is capable of detecting athletes passing through each checkpoint and storing their results in a remote database.

However, as discussed in Section 4, the timing measurement system's accuracy is affected by some issues, limiting its suitability for use in competitive sports events to an accuracy of ±3 seconds.

To overcome this limitation, an RFID-based detection system could be used only at the finish line, ensuring more precise final rankings while maintaining relatively lower costs than a system based solely on RFID devices. In conclusion, the proposed system offers an interesting option for timing and checkpoint tracking in marathons, with the caveat that its accuracy limitations must be taken into account for potential applications in specific contexts.

REFERENCES

[1] Chatchai Chokchai. 2019. Low Cost and High Performance UHF RFID System using Arduino Based on IoT Applications for Marathon Competition. (May 2019). https://ieeexplore.ieee.org/document/8713018

[2] David Perez-Diaz de Cerio, Ángela Hernández-Solana, Antonio Valdovinos, and Jose Luis Valenzuela. 2018. A Low-Cost Tracking System for Running Race Applications Based on Bluetooth Low Energy Technology. (March 2018). https://www.mdpi.com/1424-8220/18/3/922

[3] Gs.statcounter.com. 2023. *Mobile Operating System Market Share Worldwide Feb 2022 - Feb 2023*. https://gs.statcounter.com/os-market-share/mobile/worldwide

[4] Patana Jussi. 2021. *Implementation of Bluetooth based timing solution, Case small sports events*. Master's thesis. JAMK University of Applied Sciences, Finland. https://www.theseus.fi/handle/10024/511543

[5] Anne McCarthy. 2021. Why do people run marathons? *BBC* 1 (Oct. 2021). https://www.bbc.com/future/article/20210929-why-do-people-run-marathons

[6] Alexandra E. Petri and Ashley Wong. 2022. How to Track Runners in the New York City Marathon. *The New York Times* 6 (Nov. 2022). https://www.nytimes.com/article/track-runners-nyc-marathon.html

[7] Chun-I Sun, Jung-Tang Huang, Shih-Chi Weng, and Meng-Fan Chien. 2019. City Marathon Active Timing System Using Bluetooth Low Energy Technology. (Feb. 2019). https://www.mdpi.com/2079-9292/8/2/252/htm

[8] Jacopo Tosi, Fabrizio Taffoni, Marco Santacatterina, Roberto Sannino, and Domenico Formica. 2017. Performance Evaluation of Bluetooth Low Energy: A Systematic Review. (Dec. 2017). https://www.mdpi.com/1424-8220/17/12/2898