International Journal of Science and Research Archive
eISSN: 2582-8185
Cross Ref DOI: 10.30574/ijsra
Journal homepage: https://ijsra.net/

(REVIEW ARTICLE)

Check for updates

# GyroLock: A Gyroscopic implementation for privacy Protection

Biplav Sharma *, Komal Kumari, Ramya Dorai, Sunil Kumar Singh and Sudan Neupane

*Department of Information Science and Engineering, JAIN (Deemed-to-be-university), Faculty of Engineering and Technology, Jakkasandra Post, Bengaluru - Kanakapura Rd, Bengaluru, Karnataka 562112, INDIA.*

International Journal of Science and Research Archive, 2024, 11(02), 1545–1554

Publication history: Received on 03 March 2024; revised on 12 April 2024; accepted on 15 April 2024

Article DOI: https://doi.org/10.30574/ijsra.2024.11.2.0642

## Abstract

GyroLock is a smartphone application designed to prevent the snatching of cell phones. Given the significant role that mobile devices play in an increasingly digital environment, certain actions carried out in public settings can pose a high level of danger and have negative consequences. The gyro lock solution utilizes the gyroscope sensor in the phone to promptly secure the device when it detects abrupt, erratic, and disordered motions, similar to those associated with theft. The motion detection feature serves to prevent any unauthorized individual from physically extracting your personal and sensitive data from the program.

The program is highly intuitive, allowing users to easily enable features or customize them according to their environment and personal preferences. GyroLock enhances device safety by allowing users to customize the sensitivity of motion detection and activate the service. Gyrolock will enhance client services by offering automated and immediate reactions to suspected cases of theft. This enhances the safety and reassurance of consumers while reducing the dangers linked to data theft and loss.

**Keywords:** GyroLock; Gyroscopic sensors; Privacy protection; Authentication; Motion-based authentication

## 1. Introduction

In our current scenario, convenient and safe techniques are becoming more and more important. When concerns about digital privacy are growing, passwords and biometrics are the two main concerns to which the world has attracted attention. In recent years, where it has acted as a traditional authentication method. It has been constantly falling short of offering strong defence against new threats in the world of technology. As a result, creative solutions that may improve security without sacrificing user ease are in high demand. Our proposal, GyroLock is a ground-breaking technology that takes advantage of the gyroscopic sensors included in contemporary mobile devices to address this problem. Gyroscopes present a unique chance to develop dynamic and customized authentication techniques based on the user's bodily motions. Gyroscopes were originally intended for orientation and motion sensing[1]. With our project, GyroLock can create motion-based signatures to help protect user privacy by examining the minute details of these movements in three dimensions.

GyroLock is important not just because it strengthens security but also because it solves usability and accessibility issues. In contrast to conventional authentication techniques that could need extra hardware or complicated steps, GyroLock makes use of the built-in sensors included in the majority of smartphones, making it easily usable by a broad spectrum of users. Its dependence on natural gestures also streamlines the authentication procedure, improving user experience without sacrificing security. GyroLock's ability to withstand shoulders suffering attacks- a common threat in which adversaries try to obtain authentication credentials by watching user inputs- is one of its main features.

* Corresponding author: Biplav Sharma

Gyroscopic movements are hard to duplicate, which makes them a strong barrier against password or PIN hacking attacks [2], GyroLock also has a wide range of applications, which expands its possibilities outside of mobile.[3]

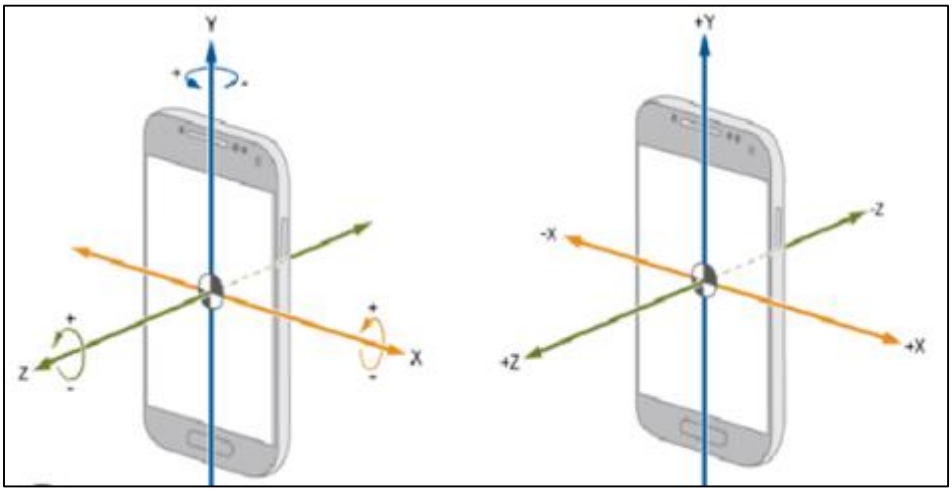


**Figure 1** Orientation overview of a Gyroscope working within smartphone devices

This introduction lays the groundwork for discussing GyroLock, a revolutionary motion-based authentication system for privacy protection. An in-depth analysis of GyroLock's design ideas, implementation methodologies, and assessment findings is presented in the following sections. These sections aim to demonstrate the technologies, as well as their effectiveness and prospective digital identities in a world that is becoming increasingly interconnected. Given the current state of the digital world, where there is a persistent threat to individuals' privacy, it is more vital than ever before to improve procedures. In spite of the fact that they were once considered sufficient. The traditional methods of verification, such as fingerprints and passwords, are becoming increasingly susceptible to hacking and are becoming more dangerous. As a consequence of this, there is a great demand for carrier and inventive solutions that may offer security without compromising user convenience. Gyroscopes are a cutting-edge method of identification that makes use of the gyroscopic sensors that are included in modern smartphones [4]. Because these smartphones were designed with orientation and motion tracking in mind, gyroscopes provide a unique opportunity to develop authentication systems that are activated by the motions of the user.Through the analysis of these movements and complex patterns in three dimensions, GyroLock creates dynamic, individualized motion-based signatures that function as extremely secure authentication elements.

## 2. Literature Review

The accelerometer, a sensor that detects tilt and linear acceleration, is used in a number of current applications for theft detection. "ShakeLock" [5] is one such example, which locks the phone when the user gives it a strong shake. ShakeLock works well for pocket theft; however, it could not work well in situations where there is little to no shaking action. GyroLock can provide a more accurate solution for snatching situations because of its concentration on the gyroscope, which detects rotational movement.

Research by Khan et al. [6] suggests a context-aware lock screen system that takes into account elements like location and time of day. For example, the lock screen may be enabled in unfamiliar places but disabled when the user is at home. GyroLock might incorporate this kind of contextual awareness. Imagine a system that automatically raises the lock sensitivity when the user's phone is not connected to their home WiFi network.

The main purpose of GyroLock is to lock the phone right away after grabbing it. A second layer of protection could be added by using more biometric authentication techniques, like fingerprint or facial recognition. Acampora et al.'s studies [7] indicate that motion detection and biometric authentication together provide a strong barrier against unwanted access.

GyroLock prioritizes reducing false positives, or instances in which the phone locks as a result of inadvertent motions. Zhao et al.'s research [8] investigates machine learning methods to distinguish between grabbing gestures and regular phone usage patterns. By lowering unintentional lockouts, using such techniques in GyroLock can greatly enhance the user experience.

GyroLock would ideally enhance current Android security features like PINs and lock screen passwords. Wei et al.'s research [9] investigates the idea of layered security, in which several security methods cooperate. In the event of a phone theft, GyroLock might serve as the first line of defense, locking the device right away, while the current lock screen presents an extra barrier for the would-be thief.

## 3. Problem Definition and Solution

Because of the growing number of cases in which mobile devices can be taken or snatched from users while they are in public places, the development of GyroLock is a necessary solution to the fundamental problem. Because cell phones are becoming an increasingly important part of our lives and are used to store important personal and professional information, the physical security of these devices is becoming an increasingly important concern, particularly while they are being used and are not locked. There is a glaring vulnerability when it comes to the instant physical theft of the item, although advanced digital security features such as encryption, passwords, and biometrics are there. GyroLock is designed to address this precise vulnerability.

Here are some specific problems that GyroLock aims to solve:

### 3.1. Quick Unauthorized Access to Device Content

The main danger associated with device theft is the quick unauthorised access to sensitive data, including banking apps, contacts, messages, and personal information. GyroLock prevents this by sensing the quick, abrupt motion of a potential snatch attempt and locking the device right away, preventing access before any damage is done. This quick reaction is essential because it protects the device in the vital first few seconds after a theft.

### 3.2. Limited Reaction Time

In the event of a snatch, the user has very little time to respond; most are unable to physically lock their device the instant a grab takes place. Here, GyroLock's automated response is essential since it takes action without waiting for a user's input, guaranteeing that the device is immediately secured—a crucial feature in unexpected situations like this one.

### 3.3. Risks of Post-Theft Data Breach

After a device has been stolen, there is a possibility that an attempt will be made to get access to the owner's personal and business information. This scenario is known as a post-theft data breach. Consequently, this could lead to theft of identity, loss of financial resources, and invasion of privacy. GyroLock significantly reduces this danger by ensuring that the device is secured immediately after a theft. This reduces the likelihood that the data will be exploited and provides a crucial security buffer during the process.

### 3.4. User Anxiety and Deterrence:

The fear of having their mobile devices stolen can have a significant impact on the decisions that users make regarding where and how they use their mobile phones. This worry can also limit the convenience and functioning of these devices. Through the provision of a proactive security mechanism, GyroLock alleviates this concern, enabling users to interact with their gadgets in a more unrestricted manner and with a reduced sense of concern over theft.

### 3.5. Over-reliance on Post-Theft Solutions

Although remote wiping and tracking are useful tools in case of theft, they are preventive measures only; they do not stop the first unauthorized entry. GyroLock's proactive locking mechanism eliminates the need for these after-the-fact fixes by providing a first line of defence that takes action before the burglar has an opportunity to take advantage of the stolen item.

By resolving these issues, GyroLock provides a proactive and prompt solution to a prevalent and upsetting issue that not only improves the physical security of mobile devices but also plays a critical role in protecting users' digital lives

## 4. Importance of GyroLock

GyroLock is more than just an app; it's a powerful security feature created to reduce the serious risk of mobile device snatching. In the current digital era, where mobile devices play a major role in both our personal and professional lives and store a wealth of sensitive data that needs to be protected against unexpected physical threats, this tool is priceless.

Some of GyroLock's major importance is given as follows:

### 4.1. Snatch Detection

GyroLock's innovative detection technology is precisely calibrated to recognise the sudden, distinctive motion of a device being snatched, guaranteeing prompt device lockdown. As the first line of defence in mobile security, this quick reaction is essential for preventing unwanted access and safeguarding sensitive data.

### 4.2. Variable Sensitivity Settings

GyroLock provides a variable sensitivity setting through which the user can fine-tune the application according to his handling style and environmental conditions. Adjustable sensitivity settings will then provide fewer false positives and more confidence in the functionality of the application.

### 4.3. Smooth Background Operation

GyroLock allows the user to work with his gadget seamlessly as it provides smooth background operation without interrupting the gadget's user. The users always get continuous security due to this seamless integration without affecting the user experience or the performance of the device.

### 4.4. No Extra Hardware

GyroLock utilises the built-in sensor of a smartphone, hence, this solution needs no extra hardware. It is very inexpensive and can be widely used.

### 4.5. Start-on-Boot Option

By activating this feature, the app starts automatically with the device and ensures that security is active from boot, thus enabling continuous protection and giving the user security peace of mind from startup to shutdown.

With these added features, GyroLock would present an alternative in mobile security that is advanced and more user-focused—one that answers the immediate challenge of device-snatching risks while at the same time assuring users that their devices, along with priceless data at these points, are safe from unwanted access.

## 5. Methodology

Throughout our research and study, we discovered several different ways to develop our application, which is called GyroLock. Hence, we choose a few of them to help us create our application. To construct the needed APK file, we can use of a variety of programming languages, including C/C++, Flutter, Java, Javascript, and others. For our use case, We‘ve decided to build our application package (APK) utilising the Java programming language and Android Studio as our primary and primary integrated development environment (IDE).

In order to build a GyroLock application that is capable of functioning, we need to request a certain authorization from both the user and the device. The manifest file is responsible for collecting both of these permissions. The responsibility of the manifest file is to gather the necessary permissions and to provide an environment that is suitable for the application. Additionally, We need numerous permissions in our manifest to ensure that our application runs properly. These permissions include RECIEVE_BOOT_SERVICES, VIBRATE, WAKE_LOCK, and others. All of these permissions collaborate to ensure that the program runs without any errors.

One of our primary goals is to develop an application that is not only as visually appealing as feasible but also user-friendly, compatible, and simple to install and operate. Holding these goals in mind, we began the process of developing a fundamental foundation within Android Studio. Before beginning the development of our java based application in Android Studio we decided to create a flow chart for every step throughout the building process:

### 5.1. Flowchart

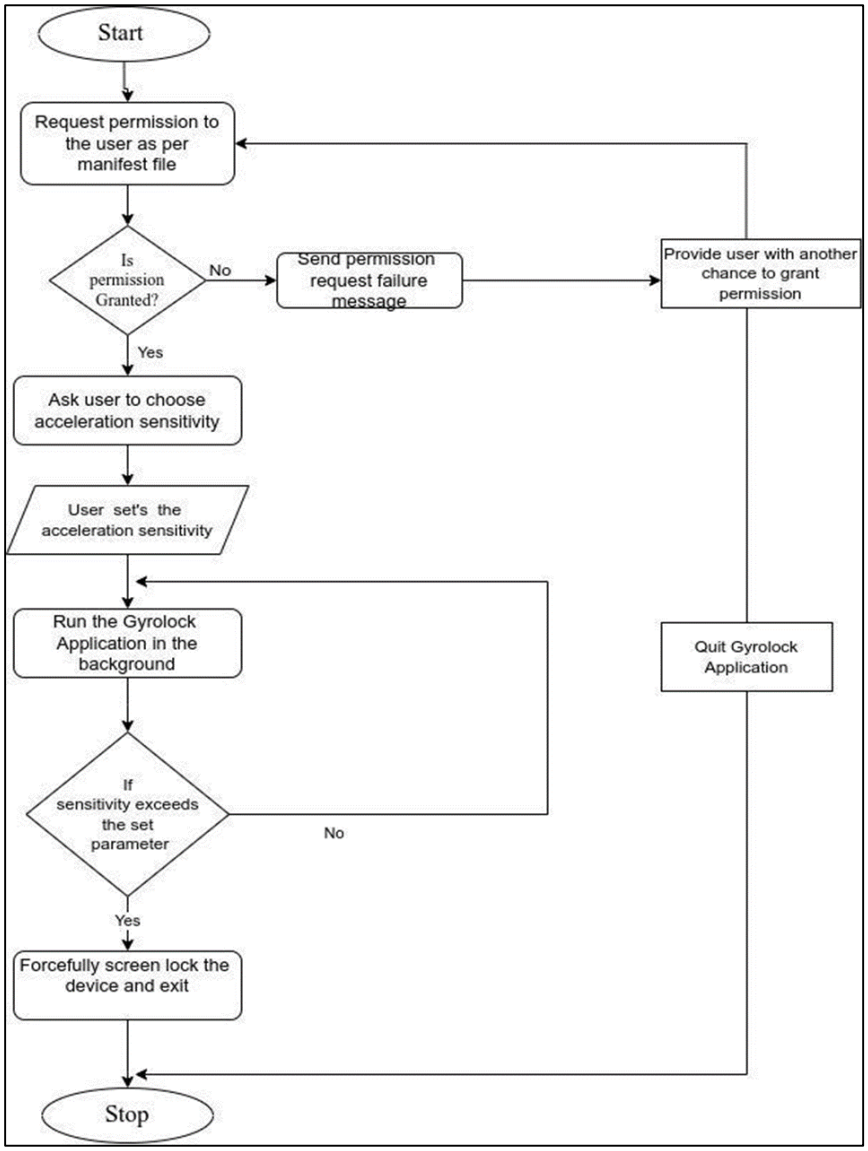


**Figure 2** Diagrammatic overview and technical Flow chart for the Gyrolock's algorithm

By the diagram above, let's discuss how our application will work :-

*Step 1: Start*

Launching the application is the first step that is extremely important since it establishes the foundation for all of the activities that will follow. In order to achieve this, it is necessary to load the required libraries, set up the data structures, and create connections to the necessary services.

*Step 2: Request Permission*

Requesting permission is the second and crucial step that is followed right after launching the application. In this step, the GyroLock application asks the user for the required permission as written within the manifest file to run the application smoothly without any errors.

*Step 3: Check Permission*

The third step is to check whether the user has given the required permission to the app or not. If the permission is granted, then it moves towards step 4; otherwise, it sends a request failure message to the user. Once it has sent the

request failure message to the user, it provides the user with another chance to grant permission and redirects to step 2. If this process is done several times, then it quits the GyroLock application.

*Step 4: Permission Granted*

The main interface of the GyroLock application becomes accessible to the user once they grant the required permissions. This access is a crucial point in time where the user moves from the app's setup to its interaction, allowing them to customize their security experience.

*Step 5: Choose Sensitivity*

The user is asked to adjust the sensitivity level of GyroLock on a scale from 1 to 10 after reaching the app's main interface. This step is essential for adjusting how the app reacts to movement on the device and enabling users to customize the app's functionality to suit their own usage habits and surroundings.

*Step 6: Set the sensitivity*

Once the sensitivity level is defined by the user, the application's monitoring threshold is adjusted appropriately. This makes sure that GyroLock triggers the lock mechanism only when the intensity of the device's movements reaches or exceeds the sensitivity level that has been selected.

*Step 7: Run GyroLock in background*

Once the sensitivity is set, GyroLock begins monitoring silently. That way, the app will keep an eye on the device's motion and compare it to the threshold you've set. It will then be prepared to start the lock protocol as soon as the detected motion matches the sensitivity level you've chosen.

*Step 8: Check for excessive sensitivity.*

Here, GyroLock keeps an eye on your device's motions in the background and compares them to the sensitivity threshold you set. There will be immediate action taken by the application if the movement intensity is greater than this threshold. Unless something goes wrong, it will stay in the monitoring phase, making sure it is always alert and ready to respond to the device's dynamics.

*Step 9: Forcefully lock the device and exit*

As soon as GyroLock determines that the device's motion goes beyond the threshold that user had specified, it locks the device to immediately. After this, the app will notify the user that the device has been locked because it detected movement. This way, the user will know that the app is taking precautions.

*Step10: Stop*

After the device has been locked, GyroLock terminates its active monitoring operation. Once the device is locked it requires device pin or password to re unlock the device for enhance security and android policy.

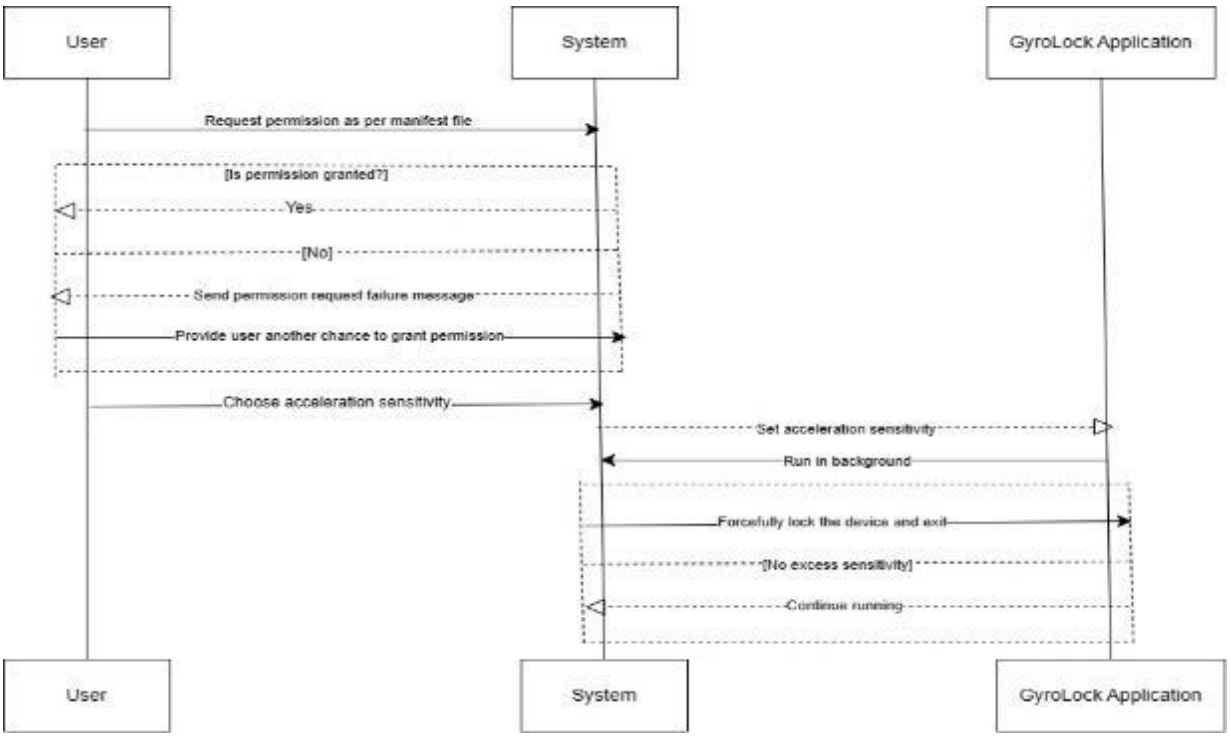


**Figure 3** Activity Diagram for the Technical aspect of Gyrolock algorithm

The figure presented above illustrates a functional activity diagram of the Gyrolock Application, providing an in-depth representation of the sequential interactions among the user, the system, and the application. The procedure begins when the user submits a request for permissions, as specified in the manifest file of the application. Subsequently, the system assesses this request, resulting in a binary result: if authorization is given, the process advances; otherwise, the user is notified of the rejection with an error message and is prompted to make another attempt to approve permission.

After obtaining permission, the user proceeds to establish the acceleration sensitivity settings, a decision that is recorded by the system in order to customise the operational parameters of the program. The background-running application constantly monitors the acceleration of the device. If the acceleration surpasses the sensitivity specified by the user, the program will take appropriate action by securely securing the device's screen and suspending its activity, thereby ensuring the enforcement of the intended security protocol. The application continues its background activity without interruption when the device works within its regular sensitivity range. The provided activity diagram offers a succinct and comprehensible graphical depiction of the application's logical progression from initiation to completion, showcasing the decision pathways derived from user engagement and system reactions.

### 5.2. Class Diagram For GyroLock

This UML class diagram describes the overall functioning of the GyroLock system. It outlines the system’s structure through the relationship between user interactions and the system's core classes: PermissionRequest, SensitivitySetting, and Gyrolock. The diagram offers a clear depiction of how these elements coalesce to facilitate the system's operations, ensuring both security and user-specific customization.

At the onset, the PermissionRequest class interfaces directly with the system user, managing security through methods that handle the solicitation and validation of access permissions. The boolean attribute, if granted, is pivotal in determining the outcome of a permission request, thereby controlling access to the system's functions. Conversely, the SensitivitySetting class provides mechanisms to adjust the system's responsiveness via the sensitivitylevel attribute. This class not only permits users to modify the system’s sensitivity to various operational thresholds through its methods but also validates these adjustments to maintain system integrity and performance.

Central to the system is the Gyrolock class, which implements the core functionality through methods that initiate system operations, enforce security protocols, and allow for the system's orderly termination. It integrates the

validations from the PermissionRequest and SensitivitySetting classes to maintain operational security and tailored responsiveness. This class serves as the operational nucleus of the GyroLock system, responding to user commands and system triggers, indicative of a robust and user-responsive security application.

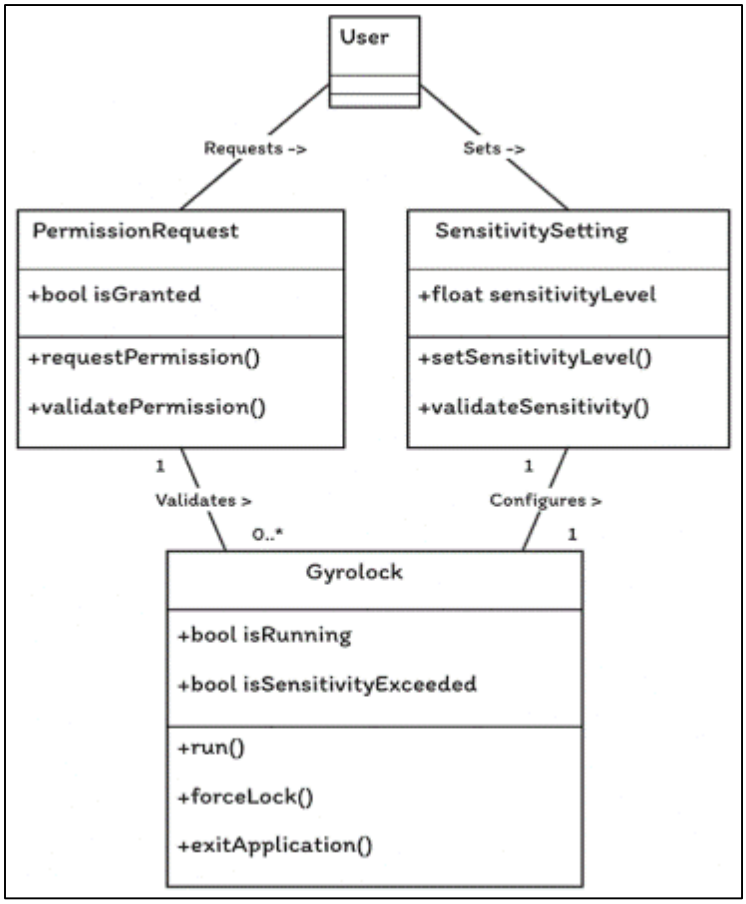


**Figure 4** Class Diagram for the Technical aspect of Gyrolock algorithm

## 6. Hardware and Software Used

We designed our application primarily with Android testing in mind, as it can be challenging to onboard an untested program on iOS. In the same way, two pieces of hardware were employed for the tests. Every piece of hardware has a role in both testing and building. Building is the responsibility of one, testing of the other. However, we are utilizing Android Studio as a Complete Builder for our application, and the following is a list of all the hardware and software specifications:

**Table 1** Hardware specifications of the device used.

| Device Name | Specifications | Purpose |
|---|---|---|
| MSI GF65 THIN 9SEXR | Intel® Core™ i7-9750H CPU @ 2.60GHz × 12 | Application Building for Android as Integrated Development Environment (IDE) |
| Google Pixel 7a | Google Tensor G2 (5 nm) octacore 8GB RAM | Testing and Bug Identification |

**Table 2** Software used for creating the application**.**

| Software Name | Specifications | Purpose |
|---|---|---|
| Android Studio | 64-Bit Program, Android Studio Giraffe \| 2022.3.1 Patch 2 (September 2023) | Used as an IDE |

The information shown here is a concise summary of the hardware and software that we make use of. First, let's get a quick understanding of each component. With regard to the computer component, we have used a powerpack device (GF65 THIN 9SEXR) that was created by MSI. This particular powerpack device is equipped with an Intel CPU core 17-9750H 2.60HZ and a 12 core processor. The graphics processing unit (GPU) is an RTX 2060 with 6 gigabytes, and it comes with 16 gigabytes of RAM. These parameters are more than sufficient to ensure that Android Studio runs well and that we are able to complete our work. The Integrated Development Environment (IDE) for Android Studio is the primary application that runs on this PC. For the Android phone, we have used the Google Pixel 7a, which is equipped with a Google Tensor G2 (5nm) processor and octacore 8GB RAM. This configuration is ideal for the seamless operation of our application that we have built.

In terms of the software, we have used Android Studio as an integrated development environment (IDE) with these particulars. The 64-Bit Program, Android Studio Giraffe, will get the 2022.3.1 Patch 2 in September of 2023. The Integrated Development Environment (IDE) that is officially sanctioned for the development of Android applications is known as Android Studio. With the assistance of this powerful tool, developers have the ability to design Android apps of exceptional quality. It includes all of the tools that are required to build applications for Android. Using Android Studio, developers are able to construct an Android application that has all of the functionality they need, including the ability to write code, test it, and publish it.[10]

## 7. Conclusion

GyroLock is a potentially significant development in authentication technology, providing a flexible and user-friendly privacy protection solution. We have thoroughly assessed GyroLock's effectiveness and usability using a rigorous methodological framework that includes research design, implementation, data gathering, analysis, and iterative refinement. Our results highlight GyroLock's strong security features, which are demonstrated by its ability to withstand common authentication threats like impersonation attempts and brute-force attacks. GyroLock's integration of gyroscopic sensors allows it to dynamically authenticate users based on distinct motion signatures, hence improving security and lowering the possibility of unwanted access. Managing and controlling the data privacy of the user in a personalized way where there is [only access system to the owner or the person who owns that device, considering this, GyroLock provides a controlling mechanism to the user where the user can control the initiatives and can set the measures which activate when the device gets sudden theft attempt or the actions like that and gives total control of data on the hand of the user.

In addition, customers have shown high levels of pleasure and efficiency with GyroLock's intuitive user experience, as highlighted by our usability evaluations. GyroLock's smooth integration with current mobile devices and operating systems guarantees wide accessibility and user-friendliness in a variety of scenarios. Prioritizing privacy has also been crucial to our evaluation of GyroLock. GyroLock guarantees user privacy and confidentiality by protecting user data and reducing the possibility of illegal access or information leakage. GyroLock does this by abiding by privacy standards and legislation.

Overall, GyroLock presents a compelling solution for privacy protection through its innovative use of gyroscopic sensors and dynamic authentication mechanisms. As we continue to refine and optimize GyroLock based on our findings, we anticipate its broader adoption and integration into digital ecosystems, contributing to a more secure and user-friendly authentication landscape in the digital age.

## Compliance with ethical standards

*Disclosure of conflict of interest*

The authors have no conflict of interest to disclose.

## References

[1] Passaro, V. M. N., Cuccovillo, A., Vaiani, L., De Carlo, M., & Campanella, C. E. (2017, October 7). Gyroscope Technology and Applications: A Review in the Industrial Perspective. Sensors. https://doi.org/10.3390/s17102284

[2] 2GesturePIN: Securing PIN-Based Authentication on Smartwatches. (2019, June 1). IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/abstract/document/8795369

[3] Alruban, Abdulrahman & Al-obaidi, Hind & Clarke, Nathan & Li, Fudong. (2019).Physical Activity Recognition by Utilising Smartphone Sensor Signals. 10.5220/0007271903420351.

[4] Rayani, P. K., & Changder, S. (2022, June 9). Continuous user authentication on smartphone via behavioral biometrics: a survey. Multimedia Tools and Applications. https://doi.org/10.1007/s11042-022-13245-9

[5] tLock | F-Droid - Free and Open Source Android App Repository. (n.d.).

[6] Khan, Farhan, et al. "Contextual lock screen: A system for improving user experience of pattern based screen lock." 2014 International Conference on Future Internet of Things and Cloud. IEEE, 2014.

[7] Acampora, Anthony P., et al. "The design of a biometric authentication system for mobile device access control." IEEE Transactions on Dependable and Secure Computing 11.1 (2014): 23-37.

[8] Wenbo, et al. "Improving mobile anti-theft accuracy through user behavior modeling." Proceedings of the 10th ACM conference on embedded network sensor systems. ACM, 2012.1.

[9] Wei, Dongxiao, et al. "Layered security for mobile devices: Combining location-basedaccess control and intrusion detection." IEEE Transactions on Information Forensics and Security 8.4 (2013): 630-640.

[10] G. (2024, January 31). What is Android Studio? GeeksforGeeks. https://www.geeksforgeeks.org/overview-of-android-studio/