IBIMA Publishing
Communications of International Proceedings
https://ibimapublishing.com/p-articles/46SE/2025/4622525/
Vol. 2025 (36), Article ID 4622525

RUTMath – a Free and Open-Source Educational Application for Learning Mathematics*

Mariusz BORKOWSKI, Dariusz KROL, Piotr KROL,
Arkadiusz POLEC and Wojciech WEGRZYN

Faculty of Electrical and Computer Engineering, Rzeszow University of Technology,
Rzeszow, Poland

Correspondence should be addressed to: Mariusz BORKOWSKI, marbor@prz.edu.pl

* Presented at the 46th IBIMA International Conference, 26-27 November 2025, Ronda, Spain

Abstract

The aim of this work is to present an open-source, cross-platform educational application designed to support the acquisition of mathematical skills by early school-age children. The tool combines classical arithmetic exercises (addition, multiplication, multiplication tables, unit conversion) with gamification elements such as duel mode and a reward system. The application is based on the principles of the Singapore method, fostering the development of mathematical competence through visualizations and an intuitive interface adapted to the perceptual abilities of young users. It operates fully offline and does not store data in the cloud, which enhances both its accessibility and security.

The project's source code is available on the GitHub platform, while distribution is carried out via the F-Droid repository. Released under the GPLv3 license, the application supports the idea of free software, and its multilingual support enables global application in the field of education. The project also constitutes an element of the promotional strategy of the Rzeszów University of Technology, emphasizing its commitment to social engagement, openness, and innovation.

Keywords: mobile learning, educational technology, open-source software

Introduction: aim, motivation, and project assumptions

In recent years, the market for mobile educational applications has been developing dynamically. However, even a cursory analysis of the contents of the most popular Android software repository, namely Google Play, Google LLC (2025), reveals a clear dominance of commercial solutions, whose primary purpose appears to be profit generation rather than genuine support of the didactic process. Although many of these applications are formally labeled as „free", in practice they often follow a shareware or freemium model, where basic functionality is provided at no cost but users are encouraged to make additional payments – through microtransactions, advertisements, or subscriptions. Such solutions are frequently associated with the presence of intrusive advertising content or non-intuitive interfaces leading to unintended, "accidental" purchases, which, in the case of young users, constitutes a particular ethical and educational concern.

Moreover, these applications often incorporate advertisements, microtransaction systems, and tracking mechanisms, raising legitimate concerns, especially regarding their use by children.

Another important issue concerns the graphical design employed in many educational applications. Instead of supporting concentration and cognitive development, their interfaces are often modeled after mobile games –

Cite this Article as: Mariusz BORKOWSKI, Dariusz KROL, Piotr KROL, Arkadiusz POLEC and Wojciech WEGRZYN, Vol. 2025 (36) "RUTMath – a Free and Open-Source Educational Application for Learning Mathematics" Communications of International Proceedings, Vol. 2025 (36), Article ID 4622525, https://doi.org/10.5171/2025.4622525

with bright colors, animations, and reward systems intended to maintain user attention. While such techniques may be effective in the context of entertainment, their application in a didactic environment, particularly when working with early school-age children, may be undesirable.

At the same time, an analysis of free and open-source software repositories indicates a significant shortage of modern and ethically aligned educational tools. A review of the F-Droid catalog – an alternative to Google Play that provides applications free from advertising and tracking mechanisms – shows that only three projects with a didactic character and mathematical components can be identified:

- TuxMath (TuxMath Team (2025))– an educational game in the form of a classic arcade-style shoot'em up, in which the user neutralizes incoming comets while simultaneously solving arithmetic problems,

- Primary (F-Droid Team (2025)) – a simple yet diversified educational game designed primarily for primary school students, covering tasks in arithmetic, sorting, clock reading, spelling, and plural forms,

- Brainjogging (Brainjogging Team (2025)) – an application offering a set of short logic and memory games (e.g., sequence memorization or simple arithmetic operations). Due to its nature, however, it cannot be regarded as a fully-fledged didactic tool; it is characterized by considerable simplicity, lack of adjustable difficulty levels, and absence of progress-tracking mechanisms.

Among these projects, Primary appears to hold the greatest didactic potential, although it should be noted that its last update dates back to 2018.

The identified market gap provided the foundation for the development of an original application for learning mathematics, aimed at early primary school children and designed with full accessibility, transparency, and the promotion of education as a public good in mind. The application was created for the Android platform as a free software project, which means that its source code is fully available and open to further development both by educational institutions and independent developers. In contrast to many closed, paid solutions, this tool offers users complete freedom of use – without advertisements, microtransactions, or functional limitations.

Importantly, the project serves not only a didactic function but also a representational one. The application was developed by a team affiliated with Rzeszów University of Technology, and its open distribution serves as a tool for promoting the institution among prospective students and their parents. Such initiatives align with a broader trend of building relationships with the social environment through educational rather than purely informational value. In this context, free software acts as a carrier of the principles of transparency, openness, and social responsibility of the educational institution, which constitutes a distinct competitive advantage over commercially oriented applications. As a result, the project may be perceived not only as a didactic tool but also as an element of the university's communication and promotional strategy.

Application features

The developed application provides a range of interactive mathematical exercises tailored to the educational level of children in the early grades of primary school. The designed exercises aim to alleviate the teacher's or parent's workload in preparing examples and testing novice mathematics learners in arithmetic operations (Figure 1) or unit conversions of measurements and weights (Figure 2).

Tasks are presented in the form of a series of exercises, with difficulty increasing as the user progresses – subsequent levels are unlocked upon completion of previous ones.

The application incorporates gamification elements, such as a "duel" mode (Figure 3), which allows two players to compete simultaneously on a single device, and a reward system for correct answers and completion of consecutive tasks.

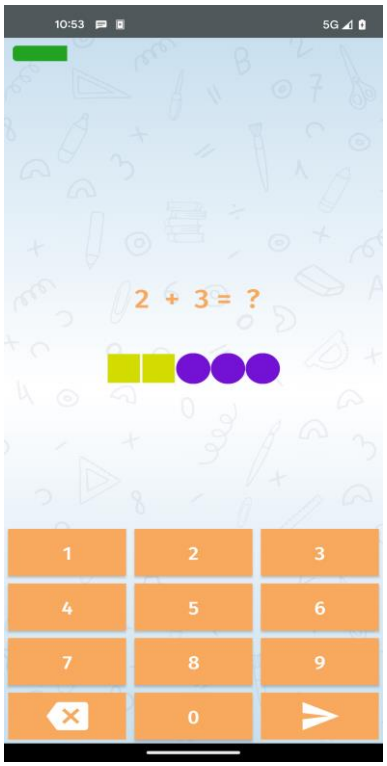

Fig 1. Application Window Appearance in Learning Mode – Arithmetic Operations

Fig 2. Application Window Appearance in Learning Mode – Units of Measurement

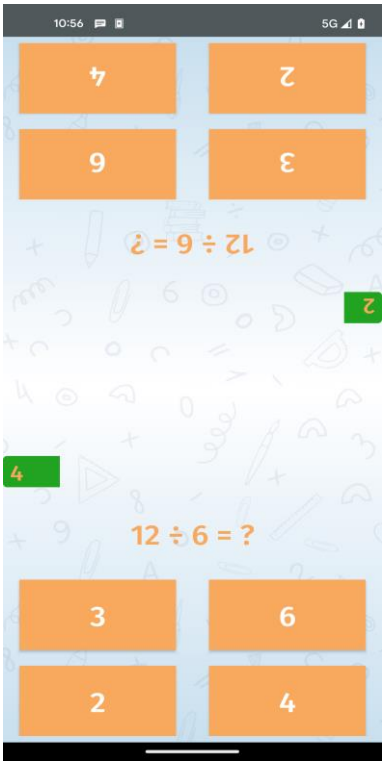

Fig 3. Application Window Appearance in "Battle" Mode

It should also be noted that the application implements elements of the Singapore method – selected exercises at the initial stages of the course are explained through the visualization of mathematical operations using colored blocks and graphical representations of concepts (Figure 1). This approach enables children not only to perform calculations but also to understand their structure and numerical relationships. The user interface was designed with the perceptual limitations of young children in mind – large icons, intuitive buttons, and a colorful graphical layout were used to enhance readability and encourage engagement with the application.

The application operates entirely offline, ensuring accessibility even in environments with limited Internet access and enhancing the security of user data. Progress information is stored locally on the device, without being transmitted to external servers.

Technical architecture and applied solutions

The original version of the application was developed for the Android system using the Kotlin language (Skeen and Greenhalgh (2018); Subramaniam (2019)), based on the native Android SDK development environment. The use of this technology stack allowed full utilization of the platform's capabilities, including a responsive user interface, support for local data storage, and compatibility with devices of various screen resolutions. The project structure was organized according to the Model-View-ViewModel (MVVM) architecture (Phillips et al. (2019)), facilitating the separation of business logic from the presentation layer. Additional advantages of this approach include easier application testing, which contributes to its maintainability and development.

The project team also employed a range of supporting libraries, including Android Jetpack (ViewModel, Navigation), which enhanced code stability and simplified component lifecycle management. To ensure proper offline operation, user progress data and application configuration are stored locally using SharedPreferences and Room (Android SQLite ORM). The entire codebase was organized with clarity, modularity, and extensibility in mind, making the project potentially attractive to external contributors and educators wishing to adapt the application to their specific educational needs.

Migration to Kotlin Multiplatform

Due to the increasing interest in making the tool available to iOS users as well, there arose a need to transform the code to enable its cross-platform use without the necessity of maintaining two separate codebases. Maintaining two distinct versions of the application – one for Android and one for iOS – generates significant development and testing costs, as well as increases the risk of inconsistencies between versions.

Consequently, a decision was made to analyze available technologies enabling cross-platform application development. Among the considered solutions were frameworks such as Flutter (Dart), React Native (JavaScript/TypeScript), Xamarin (C#), and Kotlin Multiplatform (KMP). The final choice fell on the latter technology, as it allows sharing business logic across platforms while maintaining native user interfaces. This approach provides greater control over application performance on each platform and ensures a high-quality user experience.

The objective of the latest project development stage was the migration from a native solution based solely on Kotlin to a Kotlin Multiplatform (KMP) architecture. This transformation aimed to enable the sharing of essential logical components between the Android version and the planned iOS version of the application. Adopting a cross-platform approach was a key step toward increasing the project's scalability, both technically and organizationally. It allowed simplification of the development process, reduction of maintenance costs, and mitigation of the risk of functional inconsistencies between platforms, while preserving the full functionality and quality of the existing Android version.

As part of this process, a comprehensive audit of the RUTMath application source code was conducted to identify components that could be extracted as a shared part within the KMP model.

As part of the migration of the application to a cross-platform architecture, the existing Room database in the project was adapted. A key element of the solution was the use of the Adapter pattern, which encapsulated Android-specific Room entities and translated them into shared data models in the shared module. Bidirectional mapping between entities and data classes ensured type consistency and allowed the full functionality of the database to be preserved, including table relationships, advanced queries, and schema migration mechanisms with rollback support.

Additionally, the repository was equipped with multi-level caching, comprehensive transaction management (including nested transactions and deadlock detection), and incremental backup mechanisms. Existing data access objects remained in the Android module, with their functionality exposed through an abstraction layer enabling sharing with the business logic.

In the context of cross-platform dependency management, the migration of the dependency injection (DI) system focused on implementing a modular structure of definitions using the Koin Multiplatform library (Koin Team (2025)). After analyzing available tools (Kodein, Kotlin Inject, and manual implementations), Koin was chosen due to compatibility with existing code, maturity, strong community support, and robust debugging and testing tools.

The new dependency definition architecture was divided into modules:

- SharedModule – shared business dependencies and utilities,
- AndroidSharedModule – Android-specific implementations (e.g., Android context, Room, system integrations),
- IosSharedModule – iOS-specific implementations (Core Data, SQLite, iOS SDK integrations),
- and business logic modules such as GameUseCase and DataUseCase.

Component lifecycle management was extended with custom scopes, and platform-specific dependencies were separated into dedicated Android and iOS directories using expected/actual declaration patterns.
The implementation also included advanced DI container initialization mechanisms, configuration validation, detailed error logging, and support for mocks and isolated test environments.

For the future iOS implementation, the possibility of using alternative database solutions (such as Core Data, SQLite.swift, or FMDB) was anticipated, which would be implemented in the iOS-specific module while maintaining the same repository interface. A prototype implementation using SQLite for iOS was also prepared.

The iOS version of the application is currently in the final testing phase and will soon be publicly released.

Distribution and licensing model

From the outset, the project assumed full openness – both in terms of application availability and access to the source code. The application was released under the GNU General Public License v3.0 (GPLv3), which guarantees users the right to freely use, analyze, modify, and redistribute the program provided that the same license is maintained, Tsai (2008). The choice of GPLv3 emphasizes the social character of the initiative, protecting the project from closure by commercial entities and allowing its further development by third parties.

The source code is publicly available on the GitHub platform, Borkowski (2025a), which facilitates transparency of work and allows submitting corrections, bug reports, and suggestions through the pull request system. The program can be downloaded in several ways. First, it is available on the project website. In the Releases section, Borkowski (2025b), one can download an apk file and manually install it on a mobile device.

To emphasize the open-source nature of the project and to meet the expectations of privacy-conscious users, the application was also made available through the F-Droid repository – a fully free software distribution platform, devoid of tracking mechanisms and advertisements, Borkowski (2025c).

It should be noted that installation of the application from this source requires first downloading and installing the F-Droid client, which is not available on the Google Play Store. The installer can be downloaded directly from the project's main website and then run on a mobile device to access the catalog of applications offered in this repository.

Additionally, to increase accessibility, simplify installation for ordinary users, and enhance the application's popularity, it is planned to also publish it on the Google Play platform.

Moreover, the application has been translated into over 10 languages, significantly increasing its usability in an international context and potential adoption by educational institutions in various countries. Multilingual support also aligns with the project's openness and inclusivity strategy.

Use in university promotional strategy

Contemporary educational institutions, in addition to providing high-quality education, increasingly build their recognition through innovative, social, and technological activities. The developed mathematics application, distributed as free software, fits this trend as a tool supporting the promotion of the university. The application’s name, visual identity, and embedded links to the university and faculty websites indicate its academic origin (Figure 4). As a result, the application also serves as a carrier of the institution’s image as a modern, child- and family-friendly environment, open to social needs in the field of education.

Although the application has not yet been formally used in promotional activities, its integration into informational campaigns is planned – for example, during Open Days, popular science events, or educational fairs.

According to its creators, the project has the potential to strengthen the university’s position as an innovator and social partner, which in the long term may positively influence recruitment and the establishment of lasting relationships with the community.

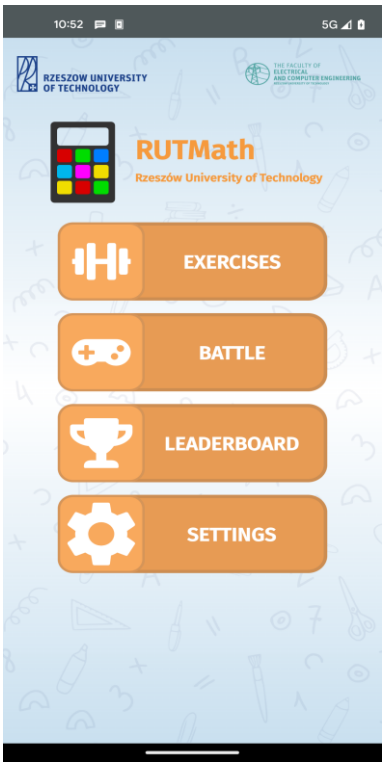

Fig 4. Main application window containing elements of the university's visual identity

Conclusions and directions for further development

The educational application described in this article represents an example of a modern, functional, and open tool supporting mathematics learning for early school-age children. Its development by students, the use of Android SDK technology in combination with Kotlin, as well as the deliberate adoption of the GPLv3 license – are elements that combine educational value with openness and transparency in line with the free software philosophy. The application features numerous advantages: offline availability, absence of advertisements, an intuitive interface, and a subdued visual design. Its availability in over 10 languages ensures that its impact potential extends far beyond the local context.

In the near future, further development work is planned, including both the expansion of the application's functionality (e.g., with additional types of exercises) and the creation of complementary tools supporting learning in other fields such as chemistry or physics. According to the authors' intentions, the project may serve as a foundation for a broader initiative in open digital education.

References

- Borkowski, M., (2025a). RUTMath Github Webpage. URL: https://github.com/przemarbor/RUTMath.
- Borkowski, M., (2025b). RUTMath Releases. URL: https://github.com/przemarbor/RUTMath/releases.
- Borkowski, M., (2025c). RUTMath Webpage on F-Droid. URL:https://f-droid.org/en/packages/com.hexbit.rutmath/.
- Brainjogging Team, (2025). brainjogging Webpage on F-Droid. URL: https://f-droid.org/pl/packages/de.telefongarten.brainjogging/.
- F-Droid Team, (2025). Primary Webpage on F-Droid. URL: https://f-droid.org/pl/packages/com.quaap.primary/.
- Google LLC, (2025). Google Play. URL: https://play.google.com/.
- Koin Team, (2025). Koin Webpage. URL: https://insert-koin.io.
- Phillips, B., Stewart, C., Marsicano, K., (2019). Android Programming: The Big Nerd Ranch Guide. 4th ed., Big Nerd Ranch / Pearson Education.
- Skeen, J., Greenhalgh, D., (2018). Kotlin Programming: The Big Nerd Ranch Guide. Big Nerd Ranch / Pearson Education.

- Subramaniam, V., (2019). Programming Kotlin: Create Elegant, Expressive, and Performant JVM and Android Applications. Pragmatic Bookshelf.
- Tsai, J., (2008). Note, For Better or Worse: Introducing the GNU General Public License Version 3. Berkeley Technology Law Journal 23, 547–550.
- TuxMath Team, (2025). TuX Math Webpage on F-Droid. URL: https://f-droid.org/pl/packages/org.afrikalan.tuxmath/.