# **C:\Users\DELL\Documents\Sofware_Development\Standardbank_Learner_engineer_assessment\learner_engineer_assessment\src\assets\images\qr-code.png**

# **Arithmetic Drills for kids App Documentation**

## **Table of Contents**

1. Introduction
2. Features
3. [I](https://chatgpt.com/c/48161a2c-15d2-4f6f-9c26-1100456b8105#installation)nstallation
4. Usage
5. Components
6. Services
7. Routing
8. Styling
9. Future enhancements and build updates

10. Credits and Inspiration

Introduction

The Arithmetic Drills for kids App is an Angular application designed to provide arithmetic problem-solving drills for kids. It offers a user-friendly interface for practicing arithmetic operations such as addition, subtraction, multiplication, and division.

Features

• Ten (10) Dynamic and randomized generation of arithmetic problems per session.

• User input for answering problems (input field only allows numbers).

• Feedback on correctness of each answer.

• User input validations (when field is empty, and when a non-number value is inputted)

• Question numbering.

• Remaining questions tracking.

• Score tracking.

• Score presentation for the session presented by dynamic color themes (red, yellow and green)

• ‘Show Answers’ for displaying report card for the session.

• ‘Exit session’ for exiting the session.

• ‘Restart’ for restarting the session.

• Alert notification of page refresh during problem-solving session.

• Prevention of going back or forward using the browsers navigation.

• Alert notifications using sweet alert on certain actions (exit, restart and report card display).

• Wild-card routing once a wrong url path is inserted and display error 404.

## Installation

To install and run the app locally, follow these steps:

1. Clone the repository: ***git clone <https://github.com/PheletsoBear/Arithmetic-Drills-for-kids.git>***

2. Navigate to the project directory: ***cd Arithmetic-Drills-for-kids***

3. Install dependencies: ***npm install***

4. Start the development server: ***ng serve***

Usage

Once the app is running, users can access it through their web browser from all devices (desktop and mobile). They can then navigate through different components, solve arithmetic problems, track their scores and view report card.

* Read instructions and Navigate to the "Arithmetic Problem" component using ‘Begin’ button.

1. Read the arithmetic problem presented on the screen.

2. Enter your answer in the provided input field.

3. Click the "Submit" button to check your answer and move to next question.

4. Receive feedback on the correctness of your answer.

* Navigate to the ‘Score-display’ component by automatically completing all ten (10) questions.

1. Read your score together with the remark.

2. View report card by clicking ‘show answers’.

3.click ‘restart’ and confirm the alert to start new session.

4.click ‘exit’ and confirm alert to exit the session and navigate to the exit component.

* System navigates to the ‘ERROR 405: PAGE NOT FOUND’ if user inputs wrong Url path.

Preventing Data Loss

To prevent data loss during your problem-solving session, please avoid the following actions:

• **Refreshing the Page**: Refreshing the page will reset your current session and you will lose your progress.

**• Clicking Standard bank logo when on session**: clicking Standard bank logo navigating to the root page or closing the browser tab will also result in the loss of your session data.

• **Clicking Restart button**: Clicking Restart button takes you to a new session and the previous session data will be lost.

Components

The app consists of several components, including:

• **NavbarLikeComponent**: Navigation bar for easy navigation back to InstructionsComponents.

• **InstructionsComponent**: Serves as a root component displaying the rules of the application.

• **ArithmeticProblemComponent**: Displays arithmetic problems, displays correctness of each answer and handles. user input.

• **ScoreDisplayComponent**: Displays the user's score and provides report card for the session.

• **ExitComponent** : Thanks the use for participating in the drill.

• **NotFoundComponent** : Serves as error 404 wildcard routing.

Services

The following services are utilized in the app:

•**ProblemService**: Generates arithmetic problems randomly and facilitates communication between components.

Routing

The app uses Angular Router for navigation between different components. Routing configurations can be found in the app-routing.module.ts file (src/app/app-routing.module.ts).

Styling

Styling for the app is achieved using Bootstrap5 and CSS3. Additional styling can be done by modifying the corresponding CSS files in the ‘src/styles directory’.

Additional Resources

• **Angular Documentation**: https://angular.io/docs

• **Angular CLI Documentation**: https://angular.io/cli

• **RxJS Documentation**: https://rxjs.dev/guide/overview

• **Bootstrap Documentation**: https://getbootstrap.com/docs/4.1/getting-started/introduction/

• **SweetAlert2 Documentation** : <https://sweetalert2.github.io/>

Future Enhancements and build updates

Future enhancements planned for the app include:

• **User Accounts**: Implement user accounts to allow users to save their progress and track their performance over time.

• **Customizable Settings:** Add options for users to customize the difficulty level of arithmetic problems and adjust other settings.

• **Interactive Tutorials**: Develop interactive tutorials to help users improve their arithmetic skills and learn new problem-solving strategies.

• **Teacher Admin Panel**: Develop a teacher admin panel to set questions for their students and monitor their progress if the schools are keen to integrate app to their curriculum.

• **AI Chats**: Incorporate AI chat functionality for tutoring and mentoring users via private chats.

• **All Users Ranking**: Implement a feature to display all users' rankings.

• **Multi-player Mode**: Allow users to challenge each other in a multiplayer mode.

• **Tournament Mode**: Introduce a tournament mode inspired by math Olympiads, featuring knockout stages where users compete against each other in timed arithmetic challenges. Users progress through rounds based on their performance, leading to an ultimate winner. The tournament mode adds a competitive element to the app, fostering engagement and motivation among users.

• **Community Support**: Establish community support features for users to interact, share tips, and seek assistance.

• **Ranking Badges**: Introduce ranking badges where users move up in rank with each difficulty conquered.

**• Increased Difficulty**: Add more challenging arithmetic questions with multiple operators and operands.

• **Session Timer Feature**: Implement a timer feature for each session, where users must complete a set of arithmetic problems within a specified time limit. The session timer adds urgency and challenge, and completion times can be used to determine users' performance and eligibility for ranking badges.

• **Sound Effects**: Add sound effects to enhance user experience, such as sound cues for correct and incorrect answers, button clicks, and session completion. Sound effects can make the app more engaging and immersive, especially for younger users.

• **Animations**: Introduce animations throughout the app to enhance visual appeal and interactivity. Include Animations that can be used for transitions between screens, feedback on user actions, and highlighting important elements. Add Animations to contribute to a more polished and dynamic user experience, making the app more enjoyable to use.

• **Session Persistence**: Implement a feature that saves the user's current session state when they exit the app, allowing them to resume their session when they return. This enhances user convenience and ensures uninterrupted progress, even if the user leaves and returns to the app later.

• **Progressive Web App (PWA) Integration**: Convert the application into a Progressive Web App (PWA) to provide offline access, fast loading times, and an app-like experience for users on mobile and desktop devices. This includes enabling service workers, configuring a manifest file, and ensuring HTTPS compatibility for deployment.

• **Kid-Friendly Ads**: Incorporate advertisements into the app that promote products or services appealing to children, such as toys, games, educational resources, or children's entertainment. These ads should be carefully curated to ensure they align with the app's target audience and provide value to users while generating revenue for the app's developers. Ads should be displayed in a non-intrusive manner and adhere to relevant advertising guidelines and regulations to maintain a positive user experience.

Credits and Inspiration

**IKM Website**: The design inspiration for the arithmetic-problem component was drawn from the IKM website, providing insights into user interface elements and layout design.

**Standard Bank Website**: The theme and color palette inspiration were derived from the Standard Bank website, guiding the choice of colors and overall visual aesthetics of the application.

**ChatGPT-4**: User stories and project planning were facilitated with the assistance of ChatGPT-4, providing valuable insights and guidance throughout the development process.

**GitHub Repositories**: My personal GitHub repositories served as a valuable resource for accessing code snippets, logic implementations, and best practices, contributing to the development and enhancement of the application.

These sources have played a significant role in shaping the design, functionality, and development process of the Arithmetic Drills for Kids application. Their contributions are greatly appreciated.