

**ДВАДЕСЕТ И ПЕТА
УЧЕНИЧЕСКА КОНФЕРЕНЦИЯ
УК'25**

ТЕМА НА ПРОЕКТА

Мобилно образователно приложение MathKiddo



Автори:

**Снежана Иванова Минчева, Професионална гимназия по електротехника и
електроника - гр. Пловдив,
Даниел Даниелов Учкунов, Професионална гимназия по електротехника и
електроника - гр. Пловдив,**

Научен ръководител (консултант):

**Катя Семкова, старши учител по теоретично обучение, Професионална гимназия по
електротехника и електроника - гр. Пловдив**

РЕЗЮМЕ:

Проектът MathKiddo е разработка в областта на информационните технологии и представлява мобилно приложение за изграждане и развиване мотивация за работа по математиката на деца в начална училищна възраст. То показва математиката по интересен начин, като задачите са представени от животинки, а при затруднения децата получават асистенция от самото приложение и не е нужна допълнителна помощ на родител.

Мобилното приложение има различни видове профили (учител, родител и ученик). Създадено е за да подпомогне образователния процес, като заинтригува всички малчугани в начална училищна възраст и изгради у тях интерес към математиката.

SUMMARY:

The MathKiddo project is a development in the field of information technology and represents a mobile application designed to build and develop motivation for working on mathematics among primary school children. It presents mathematics in an interesting way, with tasks introduced by animals. When children encounter difficulties, the application itself provides assistance, eliminating the need for additional help from parents.

The mobile application offers various types of profiles (teacher, parent, and student). It is created to support the educational process by engaging all young children in primary school and fostering their interest in mathematics.

УВОД

Основна цел на настоящата разработка е създаването на MathKiddo – мобилно приложение, подпомагащо обучението по математика в начален етап. Приложението е с опростен, но атрактивен интерфейс. Разработката предоставя задачи с различни нива на сложност спрямо изучавания материал.

Допълнителни цели на разработката са:

- Да подпомогне образователния процес в начален етап, както и родителите с по-забавен и мотивиращ за децата им начин да учат.
- Да заинтригува подрастващите ученици.
- Да направи математиката забавна за учащите, както и да повиши резултатите им чрез задачите си, подбрани с надграждане на сложността.

Въпреки, че вече съществуват подобни разработки – напр. **HOMER**, или **Elephant Learning Math Academy**, настоящата разработка има преимущество да използва български език, което е изключително важно за деца от начална училищна възраст.

За да се постигне всичко това е нужно да се обърне особено внимание на потребителския интерфейс, като се използват специфично разработени, ненатоварващи очите цветови комбинации. Важен е подборът на подходящи задачи по математика, приветливата комуникация от страна на приложението, както и лесен за използване интерфейс. По тази причина разработката трябва да разреши следните задачи:

1. Да се изберат подходящи софтуерни технологии;
2. Да се създадат интерфейсите компоненти (изображения, цветови схеми) за приложението;
3. Да се изградят софтуерните компоненти;
4. Да се тества разработката;
5. Да се разпространи приложението.

За решаване на тези задачи, разработката съдържа четири глави. В първа глава са разгледани основните стъпки в реализирането на проекта. Втора глава разглежда накратко основните технологии, използвани за решаване на поставените задачи. В трета глава накратко е разкрита структурата на разработката и реализацията на програмните компоненти. Четвърта глава разкрива начина на работа с приложението.

ГЛАВА 1. ОСНОВНИ ЕТАПИ В РЕАЛИЗИРАНЕТО НА ПРОЕКТА

Работата по създаването на настоящата разработка започва с избор на подходящи технологии за изработка на приложението. За създаването му е избран програмния език Dart, Flutter - комплект за разработка на UI софтуер с отворен код, създаден от Google, както и Firebase за базата от данни и част от функциите.

Изработката на приложението започва със създаването на страници за идентификация на потребителя (създаване на профил, вход в профил и забравена парола). След това е избрана подходяща за потребителите цветова гама и палитра - пъстра, за да привлече вниманието на децата, както и ненаситени цветове, за ненатоварване на очите им. Впоследствие се изработват графиките за приложението - лого, както и четирите животни в интерфейса - авторски нарисувани. За този процес са използвани програмите Affinity Designer 2 и Krita.

След изграждането на интерфейса се пристъпва към разработка функционалността на началната страница, както и модулите за различните класове в нея. След избиране на модул се отваря страница за избор на ниво със задачи спрямо класа. Във всяка страница със задачи има бутон за помощ, обясняващ на разбираем език от децата проблема. Необходимо е да се създадат страница за потребителя и информация за неговия профил, като има 3 различни вида потребители – учител, родител и ученик с различно ниво на достъп.

Работата по проекта продължава с изработка на документацията, както и брошури за потенциални потребители.

ГЛАВА 2 ИЗПОЛЗВАНИ ТЕХНОЛОГИИ

Разработването на мобилни приложения понастоящем е дейност, при която разработчиците имат възможност да избират между множество технологични решения. В настоящата разработка Flutter е предпочетен пред опции като MAUI или Xamarin, защото той е по-прегледен, с повече готови функции за мобилните приложения, както и защото авторите проявяват желание за усвояване на програмния език Dart. Flutter е сравнително нов и бързоразвиващ се фреймуърк за създаване на крос-платформени мобилни приложения, разработен от Google¹. За работа с него са нужни умения за програмиране с Dart. Той е изключително удобен и интуитивен, като работата с него е изключително приятна.

Плюсовете на Flutter са :

- Крос-платформеност: Flutter позволява на разработчиците да създават еднокодово крос-платформени мобилни приложения, които работят както на iOS, така и на Android. Това намалява разходите и времето за разработка, тъй като не е необходимо да се създава отделно приложение за всяка от платформите.
- Бързина и производителност: Flutter използва свой собствен двигател за рисуване на интерфейса, наречен "Skia", който позволява бързо и плавно изпълнение на приложенията. Това допринася за по-гладко преживяване за потребителите.
- Гъвкав UI и виджети: Flutter предоставя богато разнообразие от вградени виджети и инструменти за създаване на красив и интерактивен потребителски интерфейс. Разработчиците имат голяма свобода при създаването на UI, като могат да персонализират външния вид и поведението на приложението.
- Бързо обновяване и гъвкавост: Flutter предоставя бърз механизъм за обновяване на приложенията, като позволява на разработчиците да въвеждат промени в реално време и да ги виждат незабавно в приложението, без необходимост от рестартиране.

¹ Flutter. Build for any screen. Достъпно към 21.11.2024 от <https://flutter.dev/>

- Активна общност и поддръжка от Google: Flutter има активна общност от разработчици, която предоставя поддръжка, ръководства, библиотеки и други полезни ресурси. Освен това Google осигурява постоянна поддръжка и актуализации на фреймуърка.

Изграждането на достъп посредством различни профили изисква използване на съвременна система за управление на база данни. В разработката е избрана Firebase поради множество нейни характеристики. Тя е синхронизирана в реално време и предоставя възможност за съхранение и синхронизиране на данни между потребители в реално време. Тази база данни е JSON базирана и лесно мащабируема. Особено важно за проекта е, че: Firebase предлага автентикация и управление на потребителски акаунти чрез електронна поща/парола, чрез социални мрежи (Facebook, Google, Twitter и други), анонимна аутентикация и администраторски контрол над акаунтите. Не е за пренебрегване и фактът, че Firebase е безплатен и достъпен вариант².

Всички графични файлове в приложението са авторска разработка. Те са създадени посредством специализираната програма за създаване на изображения Krita. Това е мощен и безплатен софтуер за дигитално рисуване, който предлага множество полезни функции³, например:

- Разнообразие от четки: Krita предоставя различни видове четки, подходящи за различни стилове на изкуство;
- Работа със слоеве: Програмата поддържа работа със слоеве, което улеснява създаването на сложни изображения
- Инструменти за анимация: Krita включва инструменти за създаване на анимации, което позволява на художниците да оживяват своите произведения
- Асистенти за рисуване и стабилизатори: Тези функции помагат за по-прецизно рисуване и коригиране на линиите

² Lido. What is Google Firebase? Everything you Need to Know in 2024. Достъпно към 20.11.2024 от <https://www.lido.app/firebase/what-is-google-firebase>

³ Creative Freedom. Digital Painting. Достъпно към 20.11.2024 от <https://krita.org/en/>

- Поддръжка на различни файлови формати: Krita поддържа множество файлови формати, включително BMP, JPEG, PNG, TIFF и други

В допълнение на възможностите на Krita, в дизайна на приложението и редактирането на графичните файлове са използвани възможностите на Affinity Designer, който е мощен софтуер за графичен дизайн и илюстрации и предлага множество функции⁴:

- Векторни и растерни работни пространства: Позволява работа както с векторни, така и с растерни изображения, което дава голяма творческа свобода
- Инструменти за прецизност: Включва инструменти за създаване на пикселно перфектни дизайни, като режим за преглед на пиксели и възможност за мащабиране до един милион процента.

⁴ 2024 Serif (Europe) Ltd. Illustration & graphic design software at its best. Достъпно към 20.11.2024 от <https://affinity.serif.com/en-us/designer/>

ГЛАВА 3. СТРУКТУРА И РЕАЛИЗАЦИЯ НА РАЗРАБОТКАТА

Структурата на мобилно приложение, изградено с помощта на тези технологии, обикновено се състои от няколко основни компонента. В кореновата си директория приложението съдържа файловете и ресурсите, които дефинират неговата функционалност и визуален стил. Вътре в директорията 'lib', се намират източниците на приложението, като например файлът 'main.dart', който служи като входна точка за приложението. Архитектурата на настоящия проект следва типичната структура на Flutter приложение с Firebase интеграция и включва следните компоненти:

- Главна входна точка – файлът `main.dart` инициализира Firebase приложението и стартира Flutter приложението чрез `MyApp`.

- UI компоненти – класът `HomePage` в `home_page.dart` дефинира основния екран на приложението, включително app bar, съдържанието на тялото и навигационното меню. Класовете `FirstGrade` и други класове за различни нива в `first_grade.dart` дефинират UI за различните нива на въпроси.

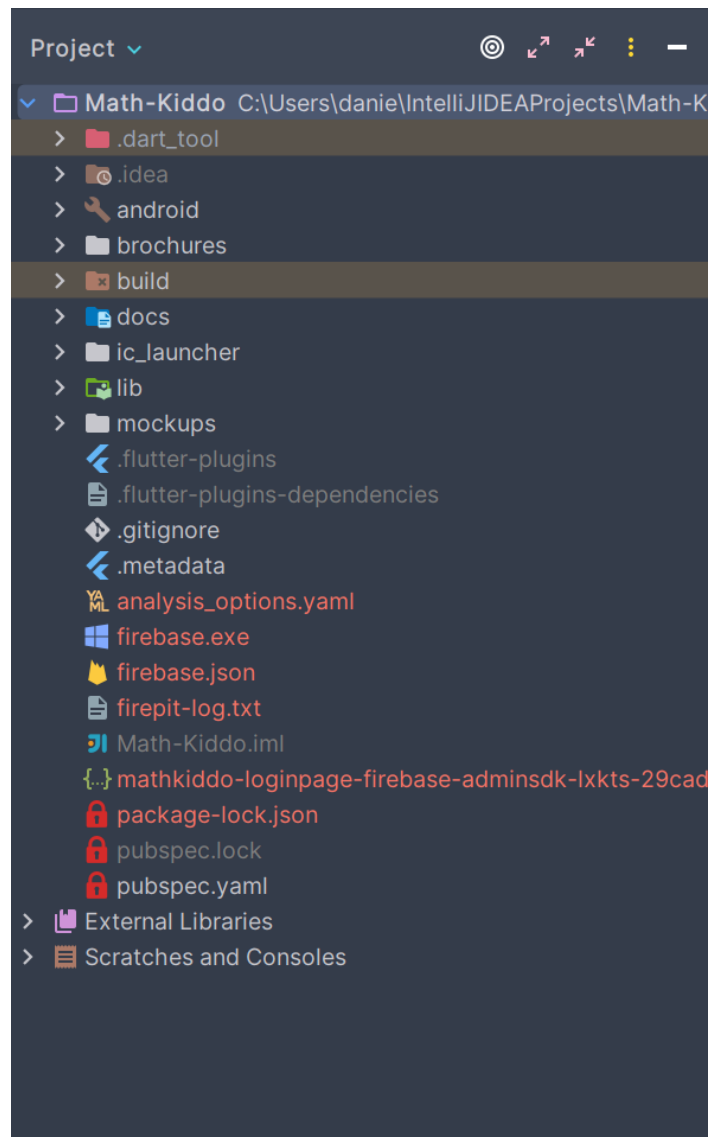
- Навигация – навигацията между различните екрани се осъществява чрез `Navigator.push` и `MaterialPageRoute`.

- Интеграция с Firebase – Firebase се инициализира в `main.dart`, а данните се извличат от `Firestore` чрез асинхронни функции.

- Тематизация – приложението използва персонализирани теми и стилове, дефинирани в `theme.dart`.

- Управление на въпроси – въпросите и отговорите се управляват в структурирана форма, а прогресът се следи чрез речници.

Тази архитектура осигурява модулна и организирана структура, което улеснява управлението и разширяването на приложението. На фигура 1 е демонстрирана файловата структура на разработката:



Фигура 1 Файлова структура

Изграждането на приложението стартира със създаването и свързването с базата данни на Firebase. Впоследствие е необходимо да се импортират необходимите за разработката библиотеки:

```
import 'package:firebase_core/firebase_core.dart';

import 'package:flutter/material.dart';

import
'package:mathkiddo/pages/others/firebase_options.dart';

import
'package:mathkiddo/pages/authPages/auth_page.dart';
```

```
import 'package:mathkiddo/pages/questions.dart';

import 'package:mathkiddo/pages/theme.dart';
```

Още при стартиране на приложението се отваря екран за създаване на потребителски профил. Чрез функциите на flutter и firebase се осъществява създаването на профил (проверка на e-mail):

```
//опит за създаване на потребител

try{

    //дали е потвърдена паролата

    if(passwordController.text ==
confirmPasswordController.text){

        await
FirebaseAuth.instance.createUserWithEmailAndPassword(

            email: emailController.text,

            password: passwordController.text,

            );

    } else {

        //грешка при еднаквостта на паролите

        showMessage("Паролите не са еднакви");

    }
}
```

Има и страница за влизане във вече създаден профил. Бутонът за създаване на нова парола отваря страница, която дава възможността за въвеждане на e – mail, след което чрез опциите на Firebase се създава e – mail съобщение. След натискане на линка се въвежда новата желана парола, потвърждава се и без затваряне на приложението се влиза в профила безпроблемно.

Преди отваряне на началната страница, приложението се допитва до потребителя дали е учител, родител или ученик, за да даде различни нива на достъп до приложението.

Началната страница при опция ученик ни приветства с избор на класа на ученика, както и с меню отстрани. Опция учител – отново менютата по подразбиране, които има и ученика плюс меню за добавяне на задачи. Опция родител – менютата по подразбиране и линия, която проследява напредъка на ученика.

Менюто по подразбиране показва двете страници - началната и за профила на потребителя.

В началната страница се избира класът на ученика, като след избор на класа се отваря страница с избор на ниво от зададените 5. Във всяко едно ниво има по 3 задачи, като първо ниво е най-лесно и с покачването на нивата се покачва и трудността. Задачите са избирани от учебник по математика, както и от НВО-то за 4ти клас:

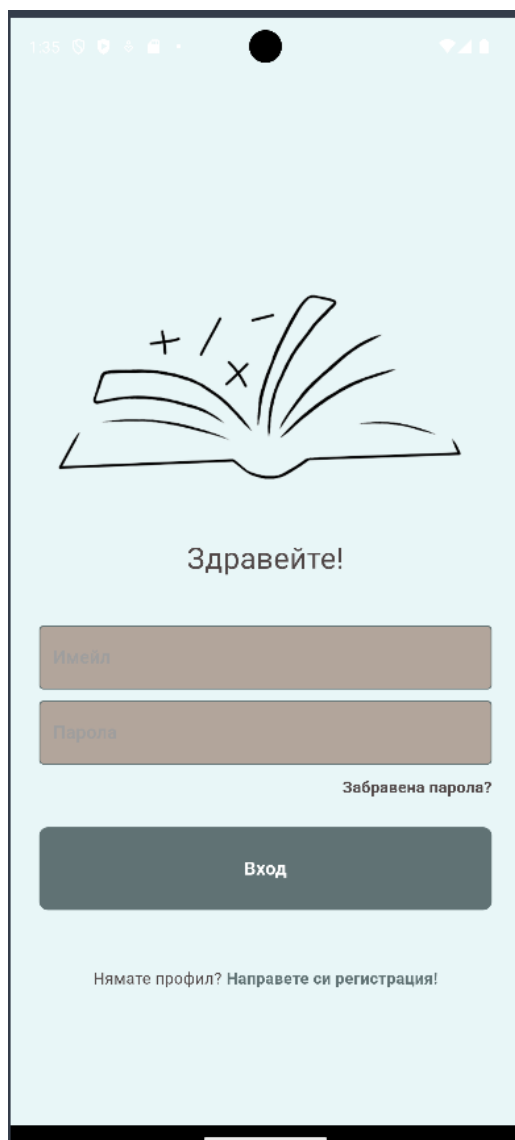
```
class Q1FourthGrade extends StatefulWidget {  
  
    const Q1FourthGrade({super.key});  
  
    @override  
  
    Q1FourthGradeState createState() =>  
    Q1FourthGradeState();  
  
}
```

В страницата със задачите има бутон за преминаване към следващата задача, като отваря другата страница автоматично вместо да се връща назад, както и има опция за затваряне на всички страници. Бутонът помощ показва насоки към дадената задача, а накрая на нивото се показва колко верни отговора има потребителя. Има опция за изработване на нивото отново след показването на

резултата, интерфейс и функционалност за избор на клас и отвеждането към страницата с различните нива.

ГЛАВА 4. ОПИСАНИЕ НА ПРИЛОЖЕНИЕТО

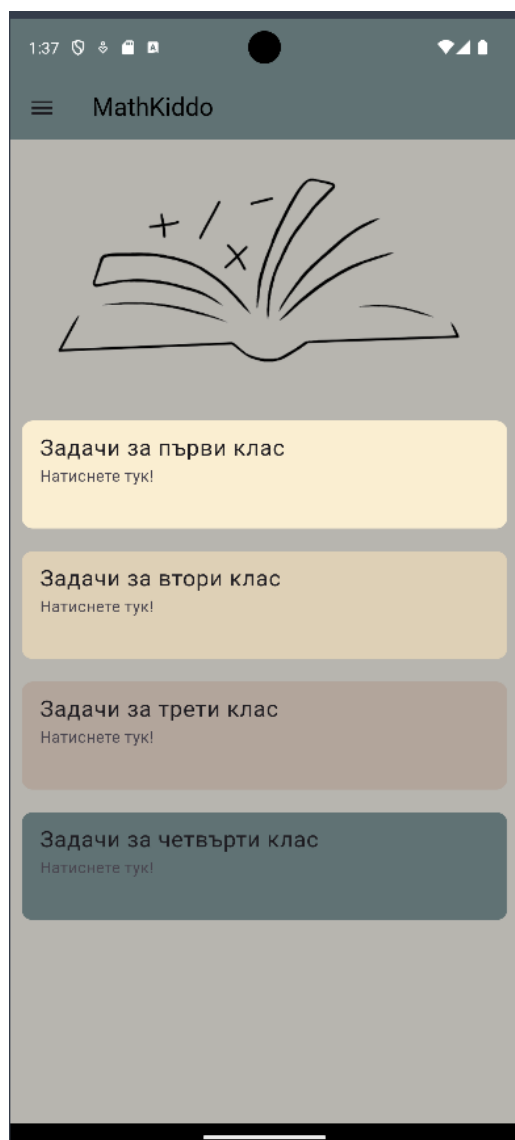
Работата с приложението започва с екран, в който потребителите имат възможност да създадат профил – чрез въвеждане на имейл, както и парола. На фигура 2 можете да се види как изглежда този екран:



Фигура 2 Създаване на потребителски профил

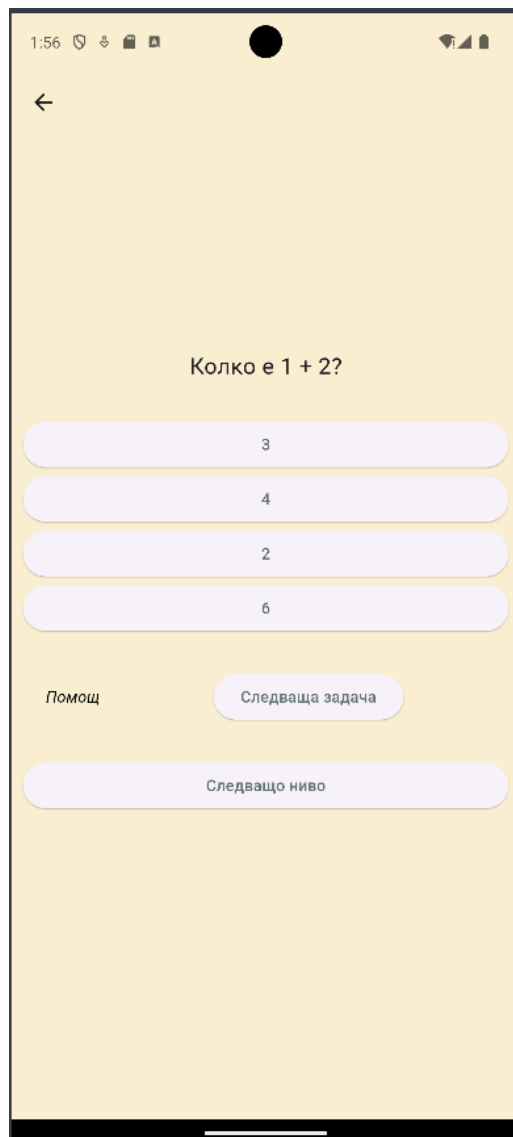
От този екран се осъществяват и функциите за влизане в профил – чрез използване на имейла, с който потребителят е регистриран, както и подходящата парола. Тук е и функционалността за създаване на нова парола – чрез натискане на бутон “Забравена парола”. Предоставя се опция за въвеждане на използвания имейл, в който се получава линк за въвеждане и възстановяване на паролата.

След включване в приложението се осъществява избор на ниво и клас посредством съответен бутон, изобразен на фигура 3:

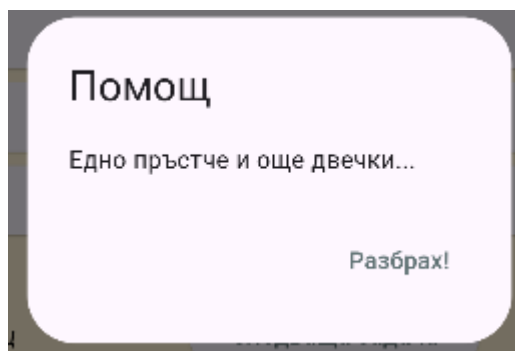


Фигура 3 Избор на клас

Преминаване на следващо ниво и получаване на помощ е предвидено да се осъществява с бутоната под въпроса. Изгледите са демонстрирани на фигури 4 и 5:

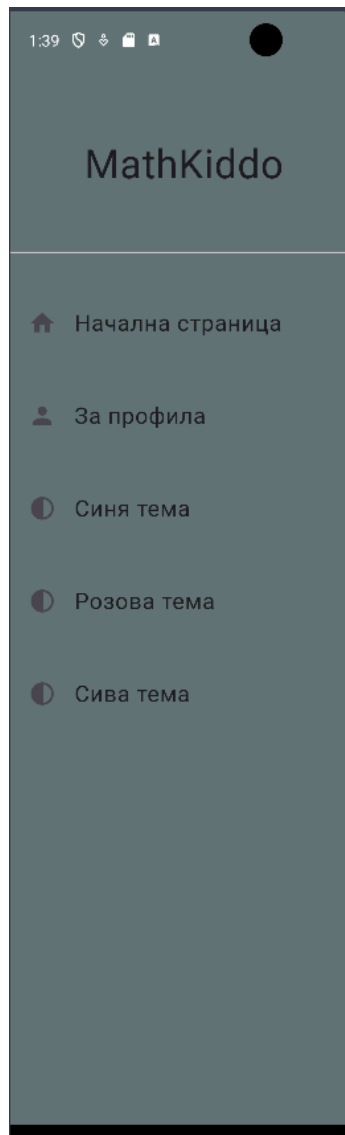


Фигура 4 Екран за задача



Фигура 5 Екран за помощ

При приключване на работа с приложението е достатъчно да се избере бутона Изход от профила - странично меню и избор на страница “За потребителя”, като в нея има бутон за изход. На фигура 6 е показан вида на изходния екран:



Фигура 6 Финален екран

ЗАКЛЮЧЕНИЕ

В настоящата разработка е създадено мобилното приложение MathKiddo, което успешно постига основната си цел – да подпомогне обучението по математика в начален етап. Приложението предлага опростен, но атрактивен интерфейс и предоставя задачи с различни нива на сложност, съобразени с изучавания материал.

Допълнителните цели също бяха постигнати. MathKiddo подпомага образователния процес, като предоставя на родителите и учителите инструмент, който прави ученето по-забавно и мотивиращо за децата. Приложението успява да заинтригува подрастващите ученици и да направи математиката по-достъпна и интересна за тях, като същевременно повишава техните резултати чрез задачи с нарастваща сложност.

За бъдещо развитие на MathKiddo могат да се предложат следните идеи:

- Добавяне на нови типове задачи и игри, които да обхващат по-широк спектър от математически концепции.
- Въвеждане на адаптивно обучение, което да персонализира задачите според индивидуалните нужди и напредък на всяко дете.
- Разширяване на функционалността за учители и родители, като се добавят инструменти за проследяване на напредъка и анализ на резултатите.
- Интеграция с други образователни платформи и ресурси, за да се предостави по-богат и разнообразен учебен материал.

MathKiddo има потенциала да се превърне в незаменим помощник в обучението по математика за началните класове, като продължава да се развива и подобрява в бъдеще.

ИЗПОЛЗВАНИ ИЗТОЧНИЦИ

1. 2024 Serif (Europe) Ltd. Illustration & graphic design software at its best.
Достъпно към 20.11.2024 от <https://affinity.serif.com/en-us/designer/>
2. Creative Freedom. Digital Painting. Достъпно към 20.11.2024 от
<https://krita.org/en/>
3. Flutter. Build for any screen. Достъпно към 21.11.2024 от <https://flutter.dev/>
4. Lido. What is Google Firebase? Everything you Need to Know in 2024.
Достъпно към 20.11.2024 от <https://www.lido.app/firebase/what-is-google-firebase>