UNIVERSITY OF TARTU
Institute of Computer Science
Software Engineering Curriculum

Magnar Aruoja

TEKI: an Online Learning Platform for Upper Primary Students

Master's Thesis (30 ECTS)

Supervisor(s): Kuldar Taveter, Ph.D

Tartu 2021

TEKI: an Online Learning Platform for Upper Primary Students

Abstract:

Online learning has become a mandatory part in everyday education, especially considering the current situation caused by COVID-19 pandemic. Despite there are many online education platforms available, there are still areas of improvement in making online learning available and seamless for everybody. Governments, for example in Estonia, have already declared strategies to address the problem. This paper gives an overview of the shortcomings of the existing platforms, including the comparison of popular online self-education platforms, and the motivation of developing, the development process, and the outcome of a new web based self-education platform designed for upper primary students acquiring basic education in Estonia.

Keywords:

Microservices, MVP, education, online learning, e-learning, self-education, Spring Boot, Angular, Keycloak, MongoDB

CERCS: P175 – Informatics, systems theory; S281 – Computer-assisted education

TEKI: veebipõhine õppeplatvorm II kooliastme õpilastele

Lühikokkuvõte:

Veebiõpe on saanud tähtsaks osaks igapäeva hariduses, eriti arvestades praegust COVID-19 pandeemia tõttu tekkinud olukorda. Hoolimata sellest, et veebis on kättesaadaval pajud erinevad õppeplatvormid, on nende kasutamise ja kättesaadavuse osas kõigi jaoks veel arenguruumi. Ka valitsused, näiteks Eestis, on probleemi märganud ning selle lahendamiseks strateegiaid loonud. Selles magistritöös antakse ülevaade olemasolevatest populaarsematest veebipõhistest õppeplatvormidest ning uue platvormi loomisvajadusest, selle arendamisest ning kasutajatesti tulemustest käsitledes Eestis põhiharidust omandavate II kooliastme õpilaste jaoks suunatud teemasid.

Võtmesõnad:

Mikroteenused, MVP, haridus, internetiõpe, e-õpe, iseõppimine, Spring Boot, Angular, Keycloak, MongoDB

CERCS: P175 – Informaatika, süsteemiteooria; S281 – Arvuti õpiprogrammide kasutamise metoodika ja pedagoogika

Table of Contents

List of Figures

List of Tables

1 Introduction

1.1 Motivation

As the web-based education has become a mandatory part in the Estonian education system, it is important that both teachers and students have access to the right tools. The Estonian Lifelong Learning Strategy 2020 [1] states that education should be an integral part of life and available to any individual.

One of the goals of this strategy was to implement a new approach to support each learner individually and at the same time reducing teachers' workload and increase cooperation between them.

The Estonian Lifelong Learning Strategy 2020 included Digital focus [2] as a part of the strategy. The focus was to apply digital technology in learning more efficiently, thus enabling access to digital learning resources in schools.

The purpose of this thesis is to analyse existing digital learning environments and map the shortcomings of the environments daily used by teachers and students. Using the provided information, the minimum viable product of a new online learning platform will be developed and evaluated.

1.2 Problem statement

Online learning (or e-learning) has been available for students for over more than two decades by now. For example, Miksike [3], an Estonian online learning platform has been developed since 1994. Although the world and technology are evolving day by day, not all problems have been solved.

Through the time, many different platforms have been developed, each one of them trying to solve different problems or make teaching and learning easier. There are platforms that support learners with materials for different subjects and topics, and there are platforms that allow teachers to create quizzes as simple test assignments for students. However, having many different platforms to choose from, with each one supporting different kinds of functionalities, makes it both hard and time consuming for teacher and students to learn, use and switch between.

What is more, these online learning platforms are not always available for everyone. One of the reasons is that the environments lack support for multiple languages, especially for minority groups such as Estonian, meaning that young students cannot take the advantage of these platforms because of a language barrier. Secondly, many platforms require a paid licence to use them, and, unfortunately, not all schools can make this possible for all the teachers and students.

In addition to that, many existing learning platforms provide a fixed set of exercises. It means that for any topic, there is usually a fixed number of different exercises that can be solved to ensure an individual will understand the topic. Still, sometimes this might not be enough. To consolidate the new topic learned, it is important to provide the student with as many exercises as possible regarding to the specific topic. Historically, this has been a teacher's job to generate new equations and exercises for students. With the rise of the artificial intelligence (AI) and computer technologies, it is possible to overcome both problems. By generating new exercises with the help of a computer, it is easy to provide students with new exercises as soon as possible and reduce teachers' workload at the same time.

Finally, some of the existing environments have become legacy systems, meaning they are not in active development anymore or they are only usable on desktop computers. This means there is no support available for the users encountering any issues and no new features or bugfixes will be implemented. Often, these platforms lack the support for current curriculum related topics and exercises. Moreover, as the mobile devices such as tablets and smartphones are becoming more and more popular as an alternative to the *old-fashioned* desktop computers, online platforms should also support mobile devices including responsive user interface.

1.3 Objectives

The main objective of this thesis is to develop a minimum viable product of a future platform of online learning for students in Estonia. The system should be easy-to-use for both teachers and students and on both mobile and desktop computers. The MVP is focused on the most important subjects and topics of the upper primary students in Estonia, thus will be available in Estonian but is not limited to support any other language in the future.

To reduce teachers' workload, the MVP should support automatic exercise generation. Students can solve the generated exercises in a practice mode and will get an instant feedback on the result. In addition to that, the system should allow teachers to create new exercises manually. The created exercises will be available for all students, establishing a shared exercise bank. This will help students to consolidate newly learned topics more easily by solving different exercises of same type, regardless of whether they are generated automatically or created manually. The system initially includes but is not limited exercises and assessments to some specific subjects, such as Mathematics and Estonian, and topics.

Another purpose of implementing a new system is to make it easily adaptable for any curriculum related topic. This would reduce the need of switching between different platforms to study different subjects or topics, being beneficial for both teachers and students as everybody will get used to the same user interface, thus less time is spent on finding the required materials and exercises.

The platform is developed following the architecture of microservices. This makes the platform scalable also for future developments including support for different subjects' and topics' related exercises for students of any stage.

1.4 Structure of the Thesis

The thesis is structured into five main chapters, future work, summary, and appendices. The chapters are ordered and described as follows:

- **Requirements Elicitation** – teacher survey analysis about shortcomings in teaching and using the existing online education environments; eliciting functional and non-functional requirements.
- **Related Work** – an in-depth comparison (including pros and cons) of the most popular educational environments in Estonian schools.
- **Software Design and Architecture** – a detailed description based on functional and non-functional requirements of creating a scalable, fully automated online platform of self-education.
- **Software Implementation** – development processes, chosen technologies and third-party dependencies described in-depth.
- **Software Evaluation** – the results and output of the MVP of the platform including the feedback survey analysis.

- **Future Work** – description of the possible future work and areas of improvement.
- **Summary**
- **Appendices** – hardware requirements for the software; application deployment instructions.

2 Requirements Elicitation

To find out the shortcomings in teaching and understanding the hard parts for students, a survey was conducted. Because of the COVID-19 pandemic during the writing of this thesis, conducting an online survey was the only way of mapping the current situation. The survey was conducted using the online environment Google Forms. The link to the survey was first sent on the 11th of December 2020 directly to the teachers of Elva Gümnaasium, Luunja Keskkool, Tartu Hansa Kool, Järve Gümnaasium, Ülenurme Kool, Võru Kreutzwaldi Kool, Tartu Lõuna TERA, Põlva Kool, and Viimsi Kool – nine (9) schools in total. As there was a Christmas break for schools at that time, a reminder to fill in the survey was sent on the 12th of January 2021. Apart from receiving no feedback from Ülenurme Kool and Järve Gümnaasium, 20 responses were received.

Due to the relatively small number of responses, the link to the survey was shared publicly on the 10th of February 2021 in the Facebook group "Eesti Õpetajate Liit". This resulted in getting 8 more responses from five (5) different schools including Puiga Põhikool, Salme Põhikool, Tallinna Humanitaargümnaasium, Viljandi Kaare Kool and Kortepohjan Väistökoulu Voionmaa (Estonian school located in Finland).

2.1 Survey Results

In total, 28 responses from 12 schools were received from the survey. The highest number of responses per school were received from Luunja Keskkool (5), followed by Põlva Kool (4) and Võru Kreutzwaldi Kool (4). The responses including undetermined answers, such as "*don't know*" or "*all subjects are important*", were excluded from the aggregated results. The related survey questions and results are available in Appendix II.

2.1.1 Subjects requiring independent work the most

In total, 57.7% of the teachers taking part in the survey stated that *mathematics* was the subject where independent work was most needed. The second most important subject deemed to be *Estonian,* which was marked as important by 30.8% of the respondents. *Biology* and *English* achieved 7.7% and 3.8% of votes respectively. The results are shown in Figure 1.

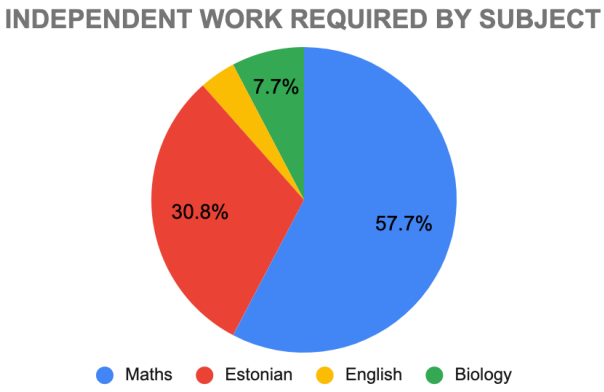

Figure 1. Independent work required by subject

2.1.2 Difficult topics in mathematics

Teachers reported that the most difficult topic in mathematics for upper primary students was related to *geometry* (31.9%) including the calculation of the perimeter and area of two-dimensional shapes, such as square, rectangle and triangle. It was followed by *common fraction calculation* with 24.6% and *decimal fraction calculation* with 20.3%. *Columnar calculation* and *mental calculation* received 13.0% and 10.1% of the votes, respectively. The results are aggregated in Figure 2.

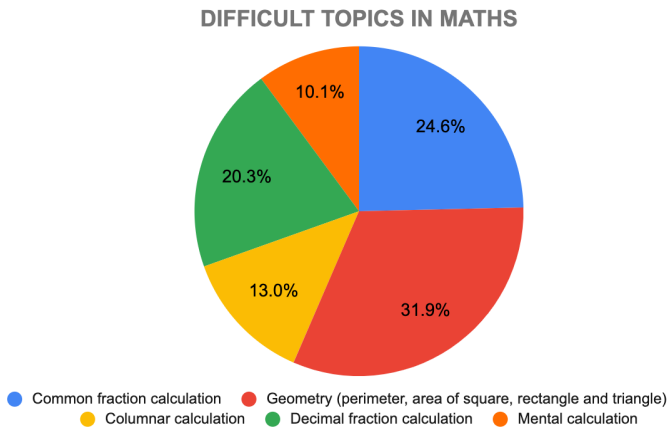

Figure 2. Difficult topics in mathematics

2.1.3 Difficult topics in Estonian

Similarly, to finding out difficult topics in mathematics, a question regarding Estonian and its grammar was included in the survey. Teachers thought that the most difficult topic for children was using commas in sentences which made up 26.6% of all the answers. *Comma rules* were followed by topics regarding *foreign words* and *consonant cluster*s with 23.4% and 20.3% of the votes, respectively. It was also mentioned by 15.6% of the respondents that learners struggle with *letter casing* (using respective case of the first letter in words) and 7.8% of the respondents considered *vowel clusters* as difficult. A few teachers also

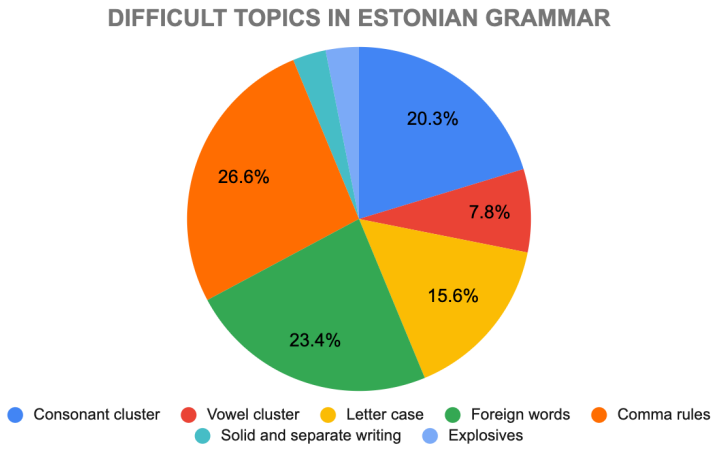

Figure 3. Difficult topics in Estonian grammar

mentioned *solid and separate writing* and *using plosives in words* is difficult for children, both constituting 3.1% of all the answers. The results are shown in Figure 3.

2.1.4 Usage of similar environments

All participating teachers reported that they used web environments for teaching children. In total, 21 different environments were mentioned. The most popular web environment used by teachers turned out to be Opiq[1], which made up 20.3% of all the answers. Learning Apps[2] and Kahoot![3] were respectively the second and the third, making up 15.6% and 14.8% of all the answers. The top three web environments were followed by 10monkeys[4] (9.4%), Miksike[5] (7.8%) and Quizizz[6] (7.8%). In addition to that, the environments mentioned contained titles such as Quizlet, TEBO (Õpiveeb), Eduten, Nutisport, 99math, Moodle, Google Classroom. The category "Other" included Matific, Baamboozle, Taskutark, Padlet, Socrates, KhanAcademy, LiveWorksheets, ThatQuiz which all received only one vote. The details are presented in Figure 4.

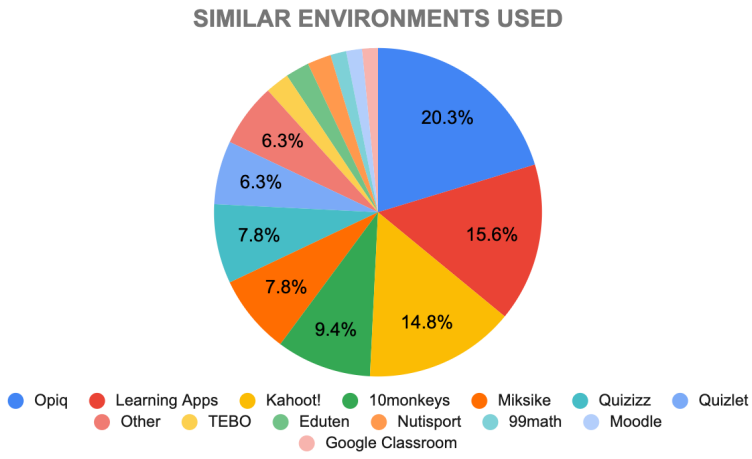

Figure 4. Similar environments used by teachers

2.1.5 Disadvantages of the existing environments

Although there are many existing e-learning web environments available, they are neither perfect nor flawless. One of the biggest disadvantages of the existing environments is that the best environments are not available in Estonian, which makes it hard for lower grade students to understand and use them (19.0% of the votes). Secondly, teachers pointed out that quite many environments lacked the possibility to check students' progress (17.9% of the votes). Another disadvantage of the existing environments that was mentioned was finding curriculum related topics covered by online environments (17.9%). Teachers also mentioned that the existing environments had paid features (14.3%), required a lot of preliminary work (11.9%) and usually required manual checking of answers by students (10.7%). Additionally, some environments forced time limit on exercises (4.8%) and were unattractive for children (2.4%). It was also stated by one teacher that because of having so many

[1] https://www.opiq.ee/
[2] https://learningapps.org/
[3] https://kahoot.com/
[4] https://www.10monkeys.com/
[5] https://miksike.ee/
[6] https://quizizz.com/

environments, it is hard to keep track of all the credentials (1.2%). Detailed information about disadvantages of the existing environments is presented in Figure 5.

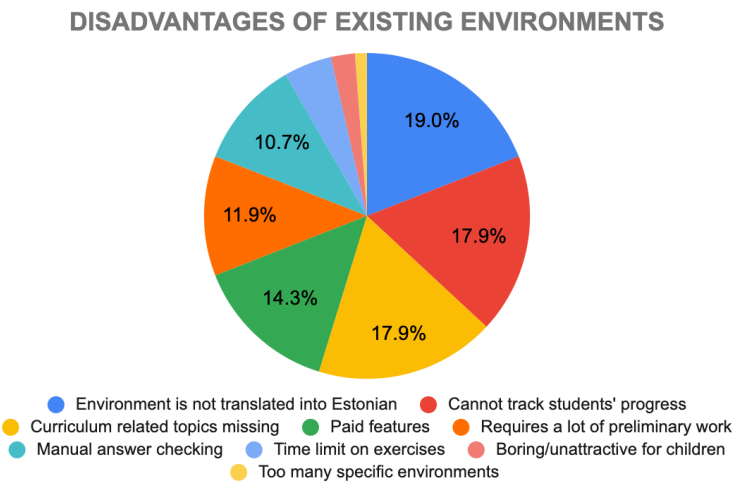

Figure 5. Disadvantages of existing environments

2.1.6 Using online environments for teaching and self-education

26 out of 28 (92.9%) respondents answered that they would like to solve the beforementioned exercises, such as comma rules, foreign words, consonant clusters in Estonian and geometry and common and decimal fractions in Mathematics, in an online environment. Also, 96.4% of teachers taking part of the survey insisted that these environments should be easy to use on mobile devices as well as on desktop computers.

7.1% of the respondents thought that online environments of this kind should be able to generate and automatically validate the exercises and submissions. Also, 7.1% stated that they would appreciate a functionality that would enable teachers to manually add exercises with valid answers. 89.3% of the respondents believed that the platform should support both automatic exercise generation and manual adding of exercises.

All but one (96.4%) of the respondents thought that online environments meant for studying should include gamification. By gamification students can, for example, gain some experience points and advance in progress levels, so that their profiles would stand out from the others.

23 out of 28 (82.1%) teachers surveyed said they would be interested in testing new online environment with their students, if possible.

2.1.7 Additional suggestions on motivating students and improving online learning

In addition to functional requirements related to teaching and learning using online environments, teachers shared their ideas on how the motivation by students could be improved. The suggestions included:

- Specific topic-related tutorial videos or animations.
- Different levels for exercises on the same topic (starter, experienced etc.)
- A possibility to retake tests – better result counts.
- Instant feedback.
- Statistics.

- Environment's attractive appearance and intuitive user experience.
- Gamification/competition/challenges.
- Motivational quotes.
- Individual or team competitions.
- Students can create and assign exercises to each other.
- Integration with third-party environments (for example, eKool or Stuudium).

2.2 Functional Requirements

Based on the survey results, an initial list of core functional requirements was observed. The functional requirements are described in Table 1.

Table 1. Functional requirements

ID	ROLE(S)	DESCRIPTION
FR1	STUDENT, TEACHER	Student/teacher can log in.
FR2	STUDENT, TEACHER	Student/teacher can log out.
FR3	TEACHER	Teacher can add vowel cluster exercises.
FR4	STUDENT	Student can practice solving vowel cluster exercises.
FR5	TEACHER	Teacher can add consonant cluster exercises.
FR6	STUDENT	Student can practice solving consonant cluster exercises.
FR7	SYSTEM	System generates natural number calculations.
FR8	STUDENT	Student can practice natural number calculations.
FR9	SYSTEM	System generates decimal fraction calculations.
FR10	STUDENT	Student can practice decimal fraction calculations.
FR11	SYSTEM	System generates exercises related to rectangle solving.
FR12	STUDENT	Student can practice rectangle solving.
FR13	SYSTEM	System generates common fraction calculations.
FR14	STUDENT	Student can practice common fraction calculations.

STUDENT – Has no special privileges.

TEACER – Has special role to access teacher-only features.

SYSTEM – Responsible for performing automated actions.

2.3 Non-functional Requirements

In addition to functional requirements, non-functional requirements were elicited from the survey results. The requirements are presented in Table 2.

Table 2. Non-functional requirements

ID	DESCRIPTION
NFR1	The application can be used on mobile devices using modern browser.
NFR2	The application can be used on desktop computers using modern browser.
NFR3	The application user interface supports Estonian.
NFR4	Requests to the backend server should be served under 5 seconds each.
NFR5	Students should have no access to teachers-only features.

3 Related Work

There are many different online environments for teaching children. From the conducted survey, Estonian schoolteachers mentioned 21 different environments that they had been using. In this chapter, the top six (6) of the most popular online e-learning environments are compared in depth that were mentioned by the teachers who participated in the survey.

3.1 Opiq

Opiq is an online environment offering both teachers and students access to digitalized books and workbooks (v. Figure 6).

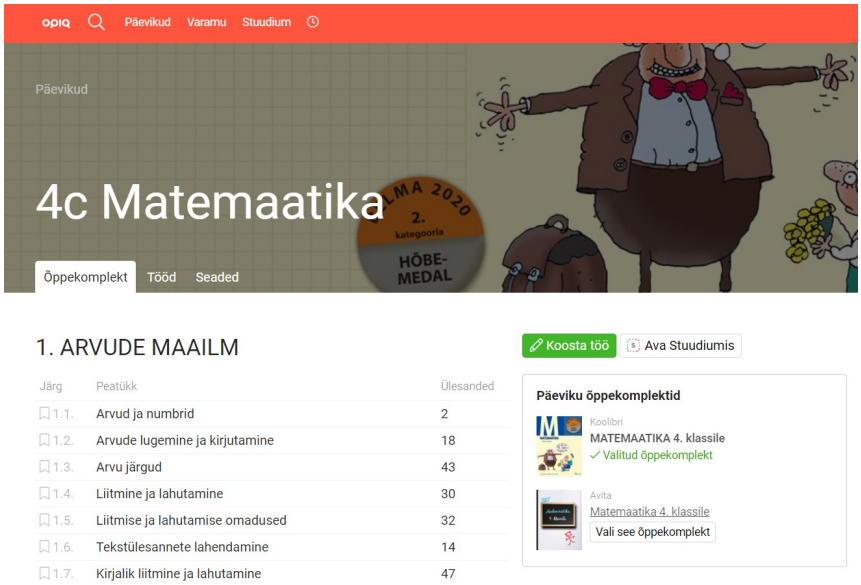

Figure 6. 4th Grade mathematics textbook in Opiq

The environment allows students to solve and submit the workbook exercises online (v. Figure 7) and teachers can also check them online. In some cases, automatic validation by the system works but in some cases, automatic validation is not functional. For example, when the user input contains spaces in wrong positions, the answer is marked as invalid, and

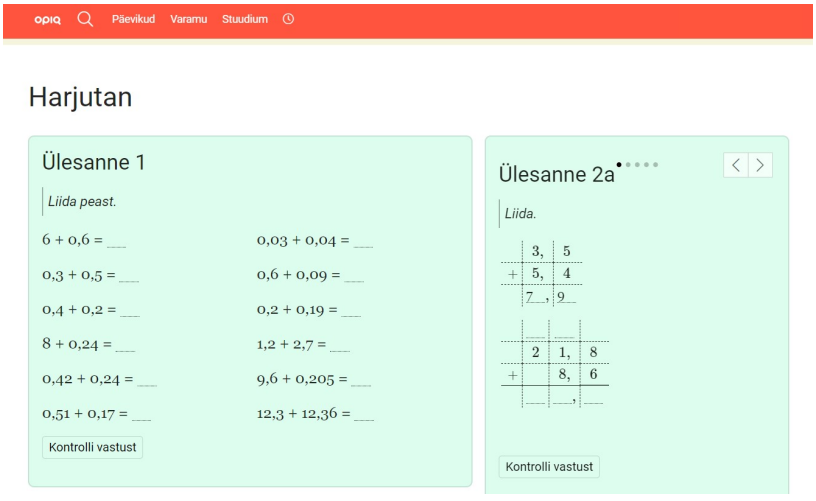

Figure 7. Mathematics exercise in Opiq

the teacher must recheck it later. Opiq does not include exercise generation functionality for any subject which means that there is a fixed number of exercises in Opiq.

However, for specific cases and subjects, Opiq offers a feature for underachievers to listen to long texts instead of reading them.

For creating online assessments, teachers can combine them using only the existing exercises provided by the environment. The assessment functionality has a feature enabling to assign the tests personally. An example of this is presented in Figure 8.

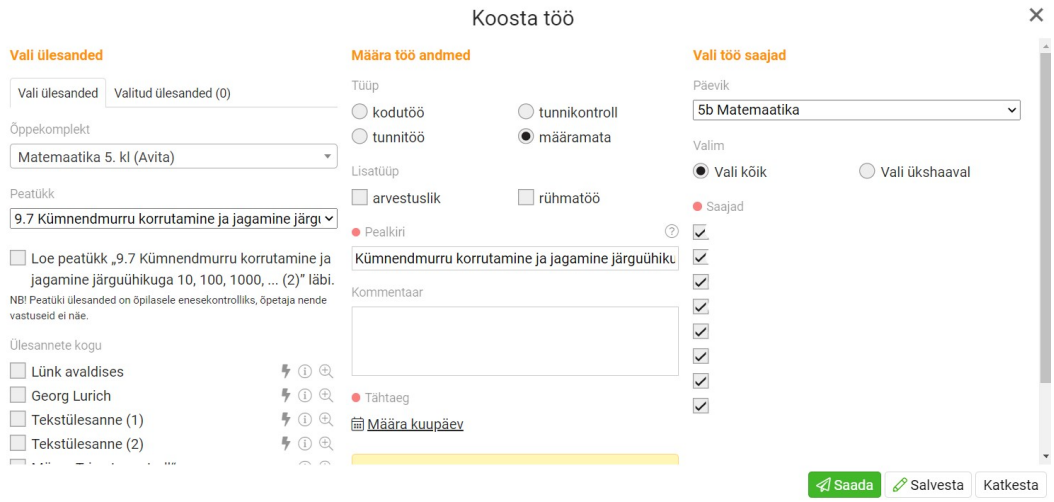

Figure 8. Assessment creation in Opiq

After the students have completed the assessments, teacher must review them and insert a grade for each student. Only after that can students receive feedback for their submissions. For each test, the teacher can see an aggregated view of the results at any given time.

Opiq also supports integration with the most popular Estonian online school diaries – eKool[7] and Stuudium[8]. This makes it easy for teachers to directly link homework exercises.

In addition, Opiq also offers a paid feature for students to buy a licence for specific workbooks. It means that all students have a free package which includes access to all the curriculum related mandatory books, but Opiq provides some additional workbooks which do not form parts of that, and therefore can be accessed only by acquiring an appropriate licence.

In Opiq, different user roles, such as student, teacher, parent, school administrator and private user, are distinguished. All the roles require verification before they can access the environment.

Despite having lots of features and a modern user interface, using Opiq requires a lot of time and manual work by teachers to get the most out of it. Teachers will have to do the same work they would do on paper and using the environment does not reduce their workload as much as it could.

3.2 LearningApps

LearningApps is a free web-based environment for creating small applications (these are called apps in LearningApps) – interactive exercises and games – for teaching and learning. The user can create an application and share it with others. Applications can be created by

[7] https://www.ekool.eu/
[8] https://stuudium.com/

choosing from among 22 available templates. An application can be built using any desired content. An example of choices between application templates is shown in Figure 9.

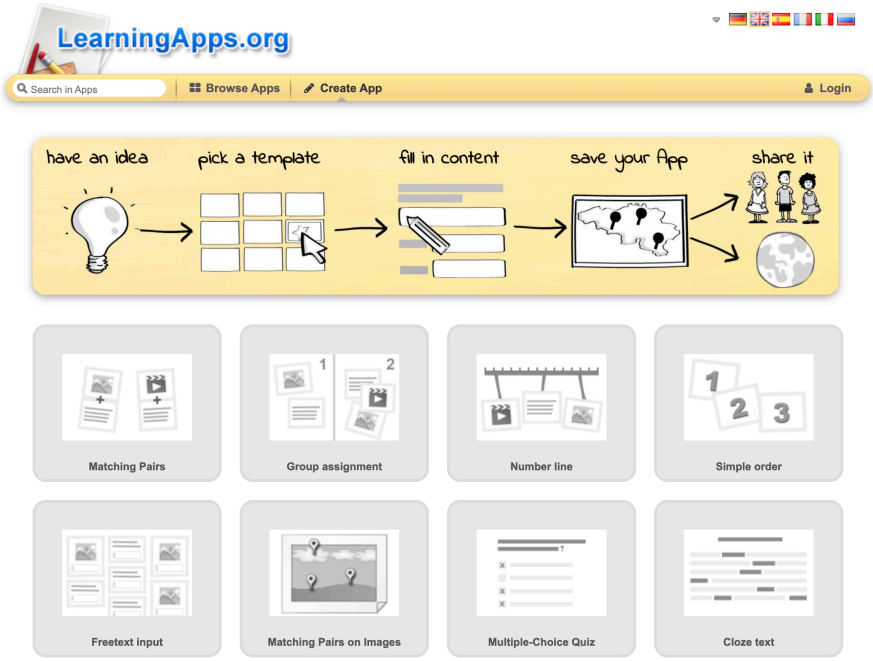

Figure 9. App template choices in LearningApps

Any created application is available for anyone with the direct link. Alternatively, an application can be found them by means of the “Browse Apps” functionality. For example, when the user searches for “decimal fraction”, a list of applications matching the search criteria are shown. Clicking on any of the matches will take the user to the application. A screenshot of a mathematics related application available on LearningApps is presented in Figure 10.

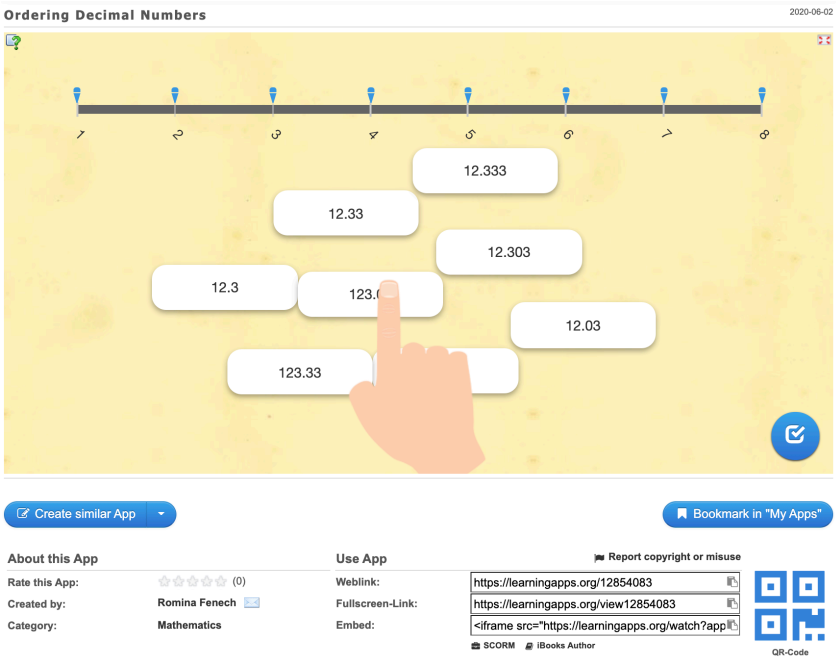

Figure 10. "Ordering Decimal Numbers" app based on "Number line" template

An answer by the user is validated instantly upon submission and the result is displayed to the user. Applications of LearningApps are easy to use on desktop computers but as the website lacks responsive design, the users might struggle using them on mobile devices.

3.3 Kahoot!

Kahoot! is a real-time environment for checking knowledge. In Kahoot!, the users can create quizzes or choose between the existing quizzes and create new Kahoot! game rooms. When a game room is created, any user with the room code can access the room and join the game. Usually, players are expected to join games using their smartphones, but Kahoot! is not limited to smartphones. The demonstration view of the Kahoot! game is represented in Figure 11. The questions are usually projected on a TV screen or any other screen that is visible to all players at the same time.

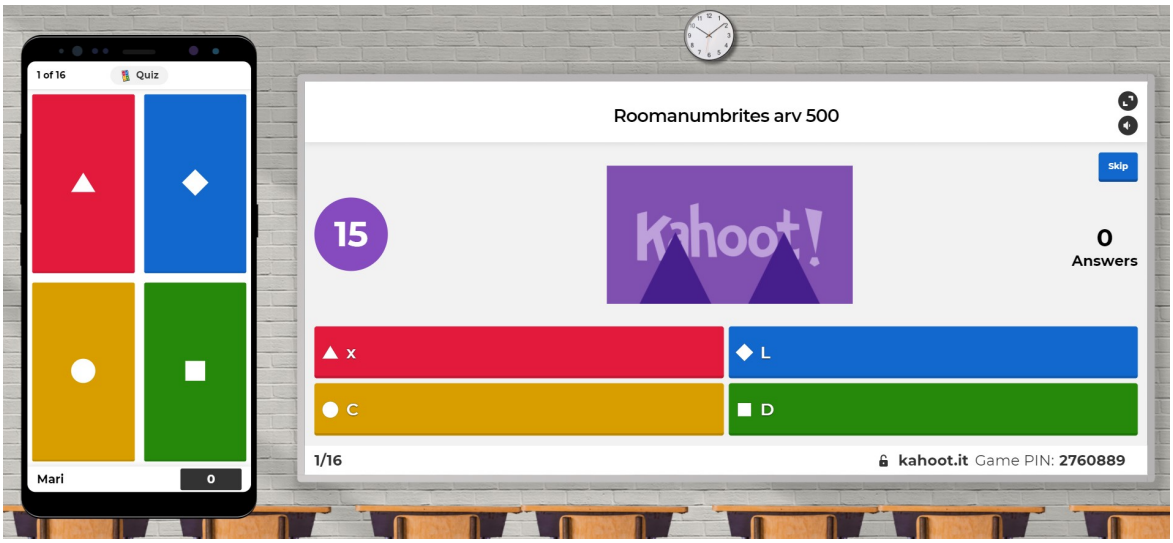

Figure 11. Demo view of Kahoot! game - smartphone view on the left, desktop view on the right

The time to answer each question is limited. The user playing a game scores points for getting the right answer to each question. The fastest player gets the maximum number of points for the given question. At the end of the game, all the points are summarized and the player with the maximum number of points wins the game.

In Kahoot!, all quizzes are created by hand and can be shared with anyone. Thus, there are no limits on the quiz topics. The users can also modify the existing quizzes into new ones, which is represented in Figure 12. At the end of a game, the teacher can request the statistics of the game and check the progress of each player.

The core functionalities of Kahoot! are free, but there are also Pro, Premium and Premium+ plans [4] available. These plans include extended functionalities that enable to take quizzes to the next level by adding new types of questions, creating bigger game rooms, conducting interactive lessons and much more.

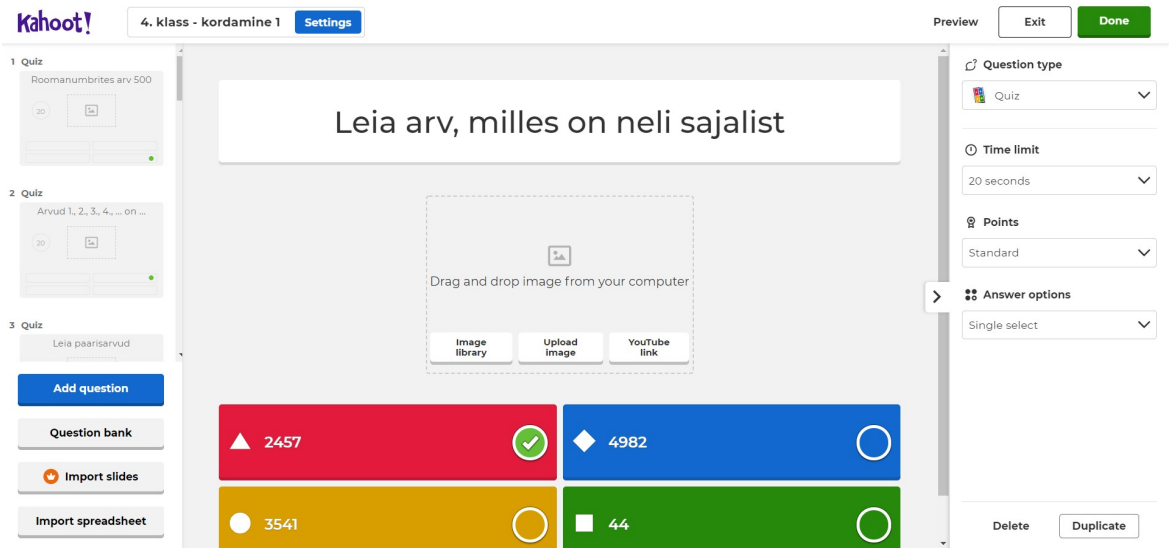

Figure 12. Edit view of existing quiz in Kahoot!

3.4 10Monkeys

10Monkeys (or 10Monkeys Math World) [5] is an online web-based environment for practicing mathematics. It offers online math programs for grades 1 to 4 including more than 300 different types of activities. A screenshot of 10Monkeys is shown in Figure 13.

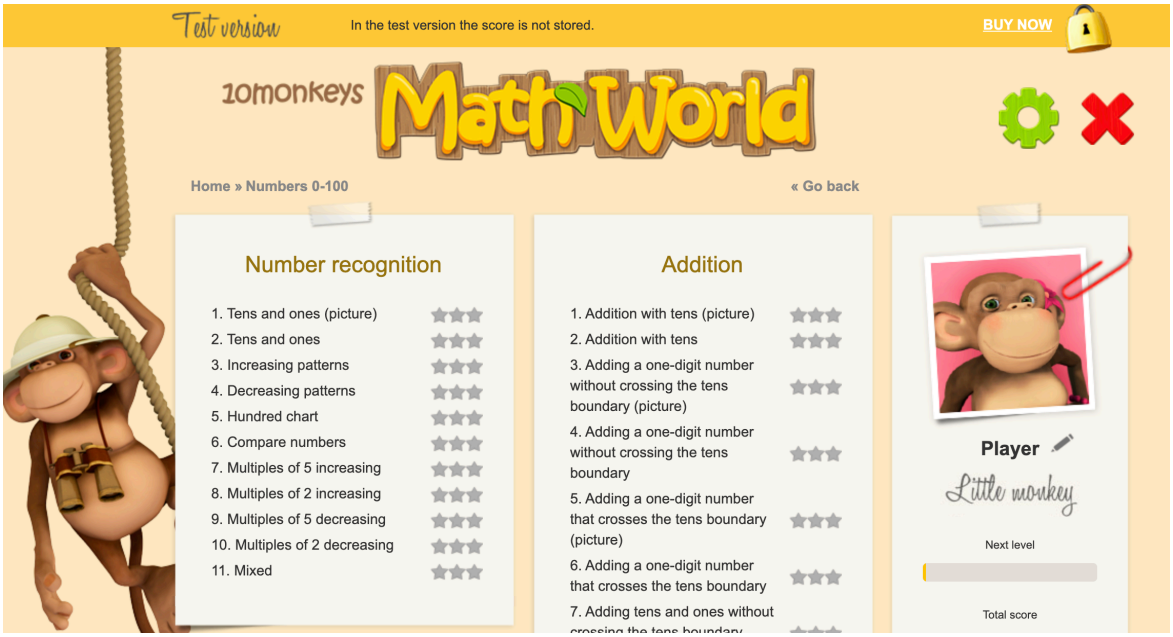

Figure 13. Activities with numbers 0-100 in 10Monkeys

The activities available on 10Monkeys contain comparisons of numbers and arithmetical base operations, such as addition, subtraction, multiplication, and division. There is also a separate activity for geometry. The free usage plan of 10Monkeys is limited to just a few practices per day. However, during the time of writing this thesis in the situation of the COVID-19 pandemics, the platform supported distance learning and made every teacher eligible to obtain a free licence for their students. To make learning and practicing fun, on

10Monkeys children are motivated with virtual prizes, awards and motivational quotes. The total scope of a student logged into 10Monkeys is persistent and a student can keep track of the progress made by him or her. An example of solving an exercise in 10Monkeys is shown in Figure 14.

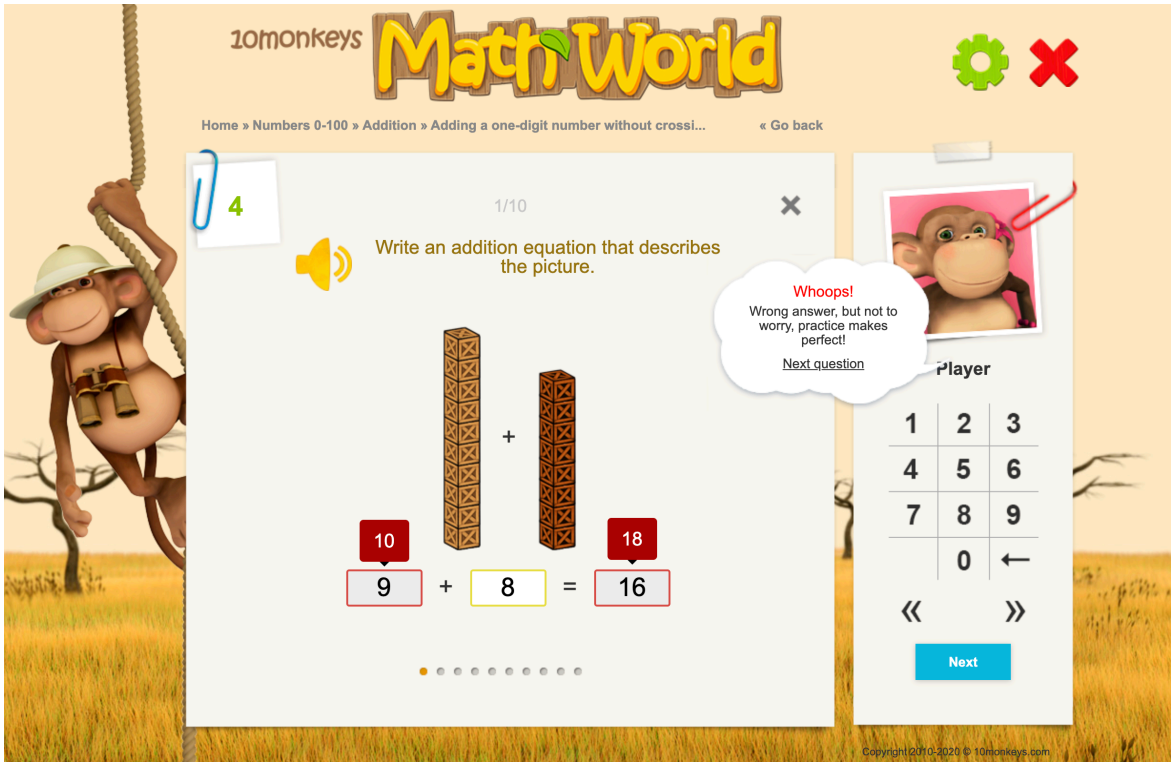

Figure 14. Adding one-digit number in 10Monkeys, an error is shown on invalid input

The user interface of 10Monkeys is child-friendly with a cartoon-like design and animations, but as it lacks responsive design, an approach to render web pages well on a variety of devices and screen sizes, the environment should be used on desktop computers or on tablets in the landscape mode.

3.5 Miksike

Miksike [3] is one of the oldest online learning environments for Estonian students the development of which started already in 1994. The idea is similar to that of LearningApps: teachers can create exercises and share them publicly. However, creating new exercises is only limited to verified teachers. Students can browse different types of exercises from the exercise banks. Performing any exercise has a time limit, and the faster the student is, the more points can be scored. A screenshot of solving a mathematics exercise with Miksike is shown in Figure 15.

Performing some exercises on Miksike is restricted to authenticated users only, which requires Estonian schools to create and connect student accounts to the platform. Despite many exercises being available on Miksike, most of them have been created more than 10 years ago, meaning that they might be outdated and not related to the curricula of today's school.

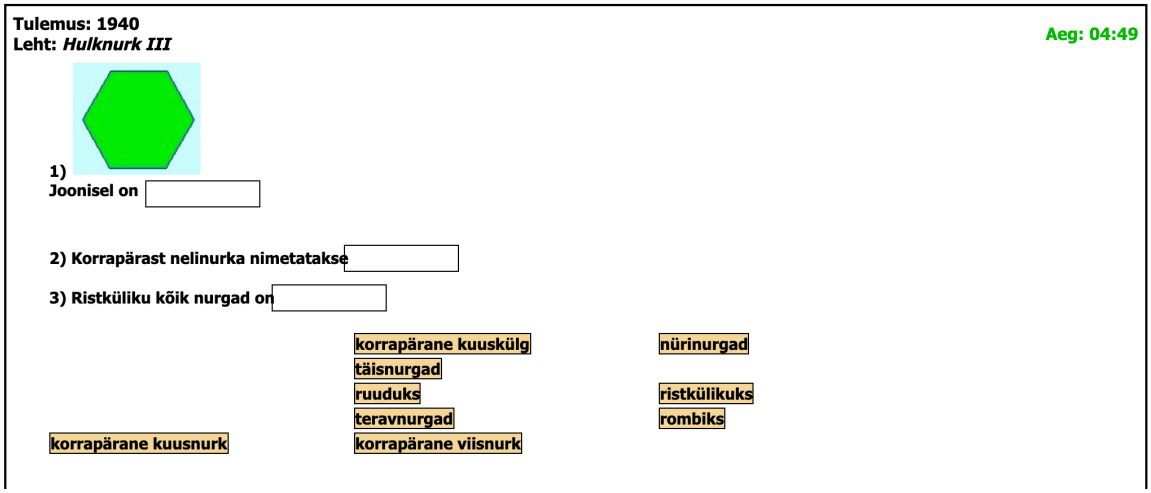

Figure 15. Solving a mathematics exercise on Miksike

In addition to exercises, Miksike has a special practice mode for mental calculations, called MentalMath [6] (in Estonian *pranglimine)*. This is a time-constrained challenge mode where students can choose between calculating with natural numbers, integers, or decimal fractions. At first, the equations are easy to solve, but they get harder over time as the student progresses by providing right answers and climbing to subsequent levels. There are also annual national contests held for MentalMath where students from the same level grade compete against each other. A screenshot of practicing mental calculations with Miksike is shown in Figure 16.

Figure 16. MentalMath practice mode

As the Miksike environment is old and visually not updated, its overall look is outdated, and the exercises can be only performed on desktop computers.

3.6 Quizizz

Another online learning platform is Quizizz [7]. Just like Kahoot!, it supports creating one's own quizzes and reusing the existing public quizzes created by others. Quizzes can be created and answered for free. When creating a new quiz, the user of Quizizz can choose to

import questions from any of the previously created public quiz or add a new question. The creation of a quiz on Quizizz is shown in Figure 17.

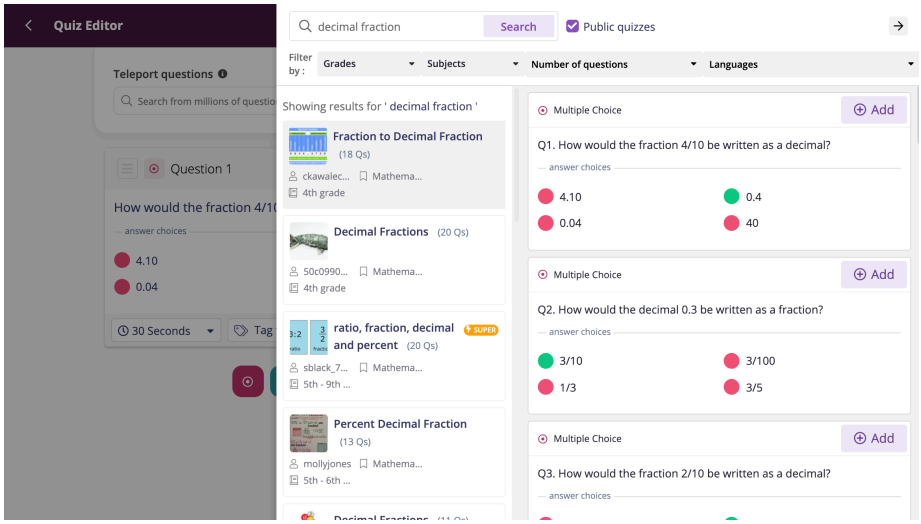

Figure 17. Quiz creation by Quizizz

The user of Quizizz can start the quiz synchronously, meaning that other players can join the room in real time by inserting a code given to them, or asynchronously in the homework mode, where the players can answer the quiz until the given deadline. A quiz can contain different types of questions, including multiple choice, checkbox, fill-in-the-blank, poll and open-ended questions. For the first four types of questions, the user can get instant feedback to the answers provided by him or her. Figure 18 shows a Quizizz quiz in progress.

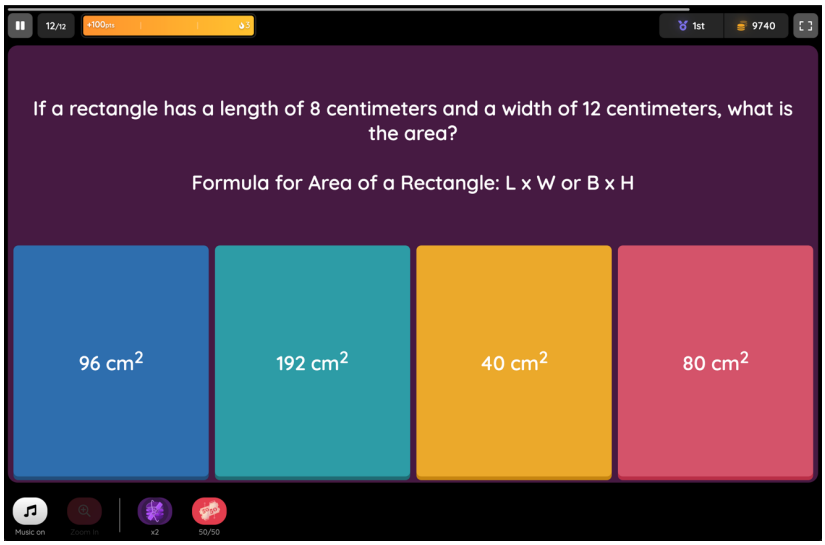

Figure 18. Quizizz quiz in progress

The Quizizz platform also has the Super plan, which costs 5 USD per month. The Super plan provides the Quizizz user with access to new tools for asynchronous learning, premium quizzes and lessons, interactive videos, and audio clips, and does not include advertisements.

3.7 Comparison

We can conclude, based on sub-sections 3.1-3.6, that online learning platforms can be divided into three categories – platforms of digitalized study materials, quiz platforms, and platforms for knowledge consolidation. The features of all the online environments described in sub-sections 3.1-3.6 are compared in Table 3.

Table 3. Existing popular environments comparison

	Opiq	LearningApps	Kahoot!	10Monkeys	Miksike	Quizizz
Environment is available in Estonian	•	•			•	
Responsive design	•		•			•
Public custom content		•	•		•	•
Teacher can add custom exercises		•	•		•	•
Supports exercise generation				•	•	
Time limit on exercises			•		•	
Free to use*		•	•		•	•

* - core features available for free.

Firstly, the most popular platform in the category of platforms of digitalized study materials was Opiq. It gives access to digitalized books and workbooks and provides the opportunity to solve online exercises included by these books. Despite that, Opiq lacks the functionalities of creating customized exercises and generating random exercises. Also, a school must pay for the licence of using Opiq.

Secondly, as the results of the survey conducted by us indicated, quiz platforms are popular among teachers. Such platforms include LearningApps, Kahoot! and Quizizz. For each of these quiz platforms, there exists quite a big community where lots of content is shared. Although the mentioned quiz platforms are international and there is a lot of content available, it might be hard to find the most accurate and helpful quizzes. Another problem is that although a perfect quiz for the given purpose might exist, it is probably not available in the required language. This means that additional work by the teacher who is trying to employ a quiz platform might be required. From the knowledge consolidation perspective, this requires the teacher to put in even more effort, because using the same quiz repeatedly is not helpful. Therefore, teachers must generate new exercises by themselves.

For the last category – platforms for knowledge consolidation – 10Monkeys is an online platform for practicing mathematics. It contains a wide variety of topics with automatically generated exercises. Despite being a good platform for consolidating mathematical knowledge, it lacks support for other subjects that could also benefit from automatically

generated exercises. The platform looks child-friendly and includes the feature of virtual awards and prizes, but the school must buy a licence for its pupils so that they could benefit from the platform.

Miksike is an online learning platform like LearningApps, containing user-created exercises for any subject which can be solved within the respective time ranges. Therefore, Miksike can be categorized into the quiz platforms' category. However, the MentalMath mode of Miksike is similar to the same feature of 10Monkeys – students can practice mental calculations by competitively solving automatically generated equations. Considering this, Miksike can also be categorized into the platforms of knowledge consolidation category. Although the Miksike platform looks outdated, MentalMath is still popular among Estonian schoolteachers and pupils, but the exercises part of Miksike is seldomly used these days.

In conclusion, the comparison of the existing popular platforms confirmed the shortcomings found through the survey conducted among the teachers. This addresses the non-functional requirements described in Section 2.3, such as the lack of mobile-friendly platforms including responsive design (NFR 2) and support for the Estonian language (NFR 3). The analysis of the existing popular platforms showed that quiz platforms are the most popular ones. They provide a convenient way to easily create questionnaires and test students. However, if no suitable quiz is found, a lot of time-consuming manual work by the teacher is required.

Using digitalized books and workbooks is another good way to teach and learn independent of one's location. But, due to the lack of unique exercises, it is difficult for students to consolidate knowledge through a fixed number of exercises. For knowledge consolidation, there are only a few platforms available supporting automatic generation of exercises but unfortunately, they only cover topics related to mathematics and do not support tracking progress by the students. This highlights the importance of functional requirements (FR 7 – 14) described in Section 2.2.

4 Software Design and Architecture

Based on the elicited functional requirements (FR1 – FR14) in Section 2.2., and non-functional requirements (NFR1 – NFR5) in Section 2.3, an MVP of a new product called TEKI was developed. The acronym TEKI comes from the first letters of both words in a phrase *"teadmiste kinnistamine"* (*"te" + "ki"*) in Estonian, meaning *"knowledge consolidation"* in English. The application's architecture, features and system requirements are covered in this chapter.

4.1 The Architecture

A web application consists of a client and a server. The client can be any device that has access to internet and is able to perform HTTP requests. For the TEKI platform, a mobile device and desktop computer with a modern web browser are considered as typical user devices.

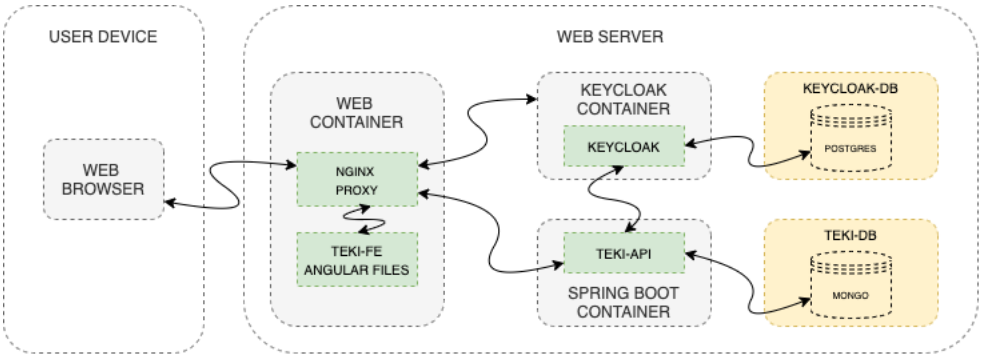

Figure 19. Platform architecture

The server (or web server) is a physical (or virtual) instance that is responsible for handling client requests. The web server consists of five (5) deployed and running containers. The architecture of the TEKI platform is described in Figure 19. Authentication and authorization are solved by using a token-based authentication protocol. This approach follows the architecture of microservices which ensures the platform can be scaled with future developments and thus retain the high performance with minimal costs. This will satisfy the non-functional requirement of the system responsiveness (NFR 4).

The *Web container* contains a proxy application that is responsible for proxying HTTP requests to the correct containers such as backend services, or serving static HTML, CSS, and JS resources for client-side application. This is the only container that is available for the public web. Access to other containers is managed by the Nginx proxy. Separating the client-side application from backend application will ease the development of client-specific features, such as improving the internationalization by adding support for multiple languages like Estonian (NFR 3) and English, without the need for introducing changes to the backend application.

The *Keycloak container* contains an application that is responsible for users, roles and authentication management. The authentication layer is important to distinguish between teachers and students (NFR 5) using the platform as there are different set of features available for both roles. The authentication service issues an authentication token (JWT) for the user, which is then used by the frontend client for making requests to the core service of the platform. JWT (JSON Web Token) is a digitally signed structure (payload) for representing claims between two parties, specified by RFC 7519 standard [8].

The *Keycloak-DB container* contains a database for persisting data about users and authentication related data.

The *Spring Boot (TEKI-API) container* contains an application with core business logic for the TEKI platform. The core service authorizes each request by validating the provided JWT against the token issuer, which in this case is the Keycloak.

The *TEKI-DB container* contains a database for persisting platform related data.

All the technologies used in developing and maintaining the platform are explained in detail in Chapter 5 of this thesis.

4.2 Software Features

On the TEKI platform, different topics on different subjects have been implemented to help students consolidate knowledge and reduce teachers' workload. Estonian and mathematics are the subjects covered in this MVP version of the platform. The platform supports authentication and authorization of users, and generation and manual creation of exercises and assessments. Furthermore, for several exercises, a statistical view as a bonus feature has also been implemented. The topics covered and features implemented in the MVP, considering the requirements specified in Section 2.2, are described in subsections 4.2.1 – 4.2.5. For each exercise, both valid and invalid answers are persisted on the TEKI platform.

4.2.1 The Home Page

The first view presented to the teacher or student is a home page. The user is greeted with the welcome text and a big button leading to start practicing is shown. The navigation bar is displayed on top of the page with navigation links to the home page and subjects' pages on the left, and the language choice dropdown menu with the login button on the right (Figure 20).

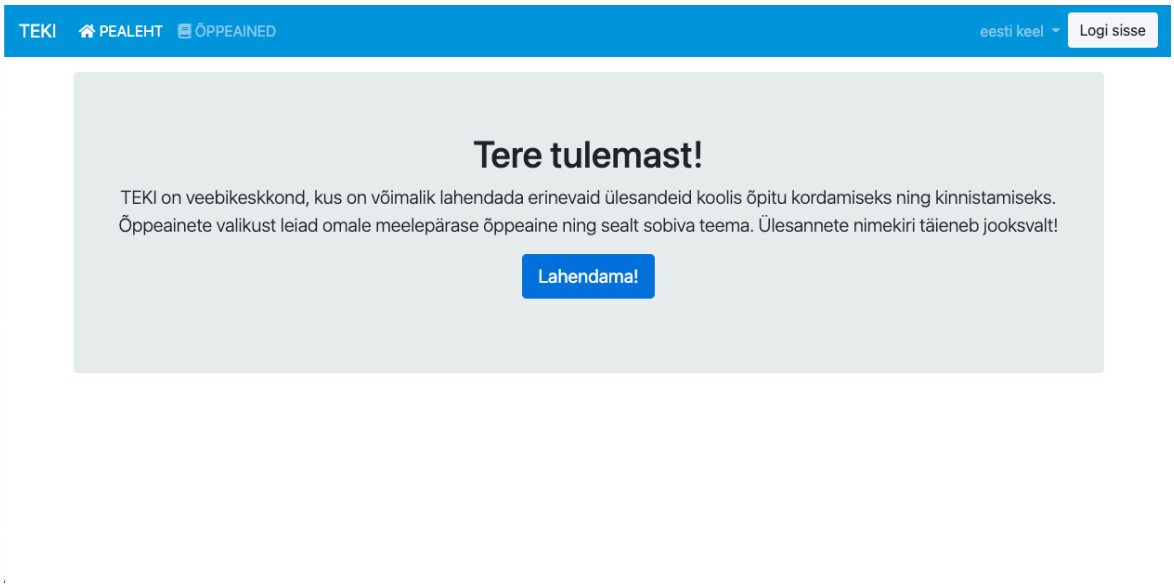

Figure 20. TEKI's Home page

4.2.2 Authentication and authorization

As the MVP is used for the evaluation of the platform, only verified users can access the content provided by the platform. To authenticate, both teachers and students can click on the login button in the top right corner of the navigation bar. If any unauthenticated user tries to access TEKI's features, he or she will be redirected to the login page for authentication. The login page is represented in Figure 21.

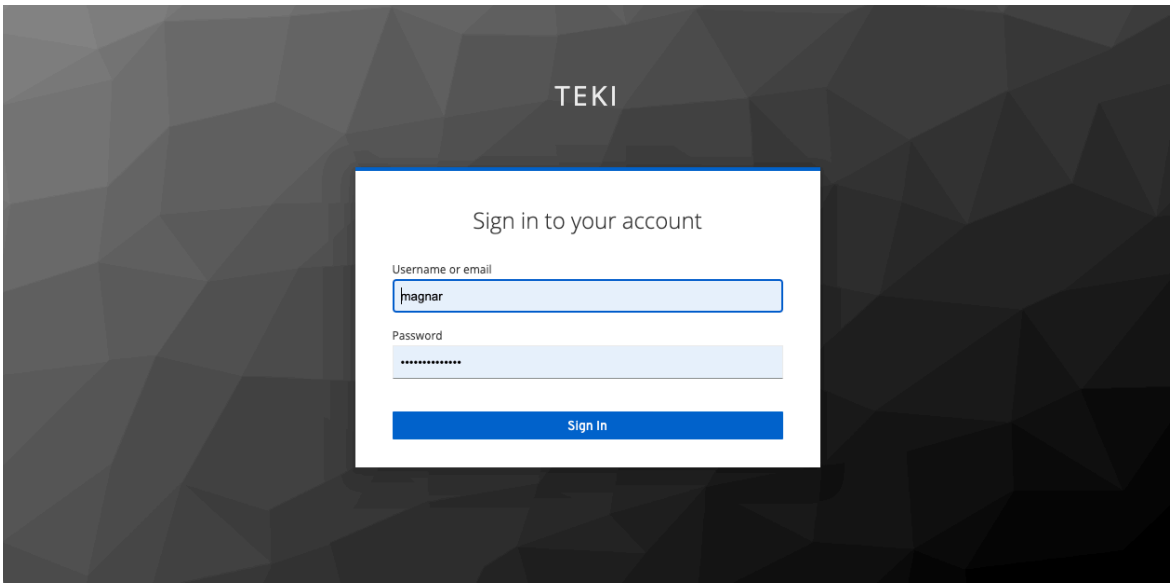

Figure 21. TEKI's Login page

Authentication requires using valid credentials: username and password. For the evaluation, TEKI users with corresponding roles for both teachers and students were provided with pre-generated users credentials.

On successful authentication (FR1), the user is redirected back to the page where the authentication process was started. For example, when the user clicked on the login button provided on the home page, the user will be redirected back to the home page.

For an authenticated user, the username will be shown instead of the login button together with the dropdown menu containing the log out button for performing the logout action (FR2).

4.2.3 The Subjects Page

Regardless of whether a student or teacher has logged into the TEKI platform, the user can navigate to the Subjects page. The Subjects page (Figure 22) contains a list of available subjects with topics available on the platform. The subjects are shown as cards, each con-

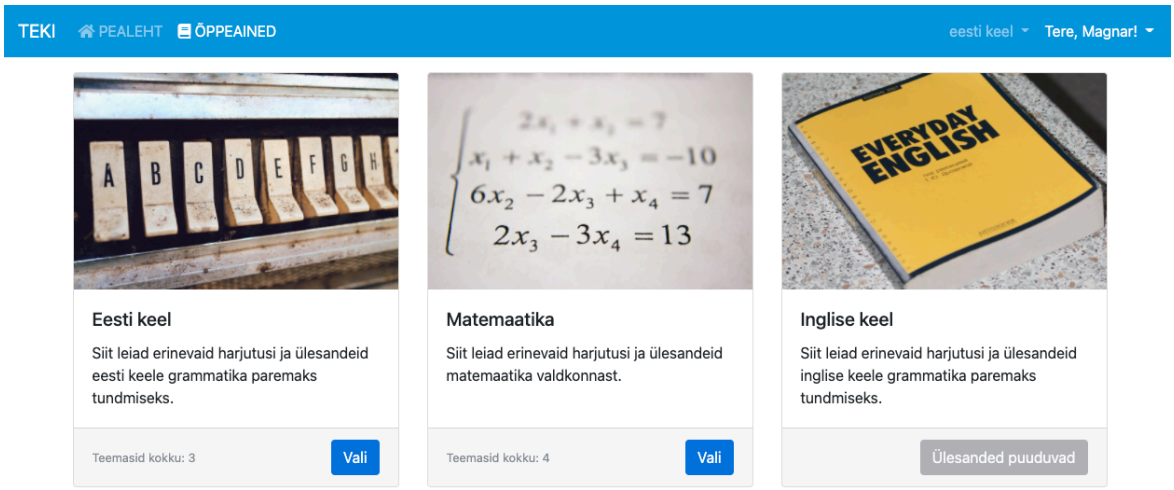

Figure 22. TEKI's Subjects page

taining the subject title, description, and the number of topics available under the subject. If the subject has at least one topic available, a button to choose the topic is enabled. In case there are no available topics for the subject, a disabled button with the corresponding label is shown.

4.2.4 Estonian Subject

In TEKI, three (3) topics from the Estonian subject have been implemented. These include *consonant cluster*, *vowel cluster* and *comma rules*. The practicable exercises in these topics can be added only by teachers. To add an exercise to a particular topic, a button with the pencil icon is shown to the authorized user. An example of the topics for the Estonian subject is shown in Figure 23.

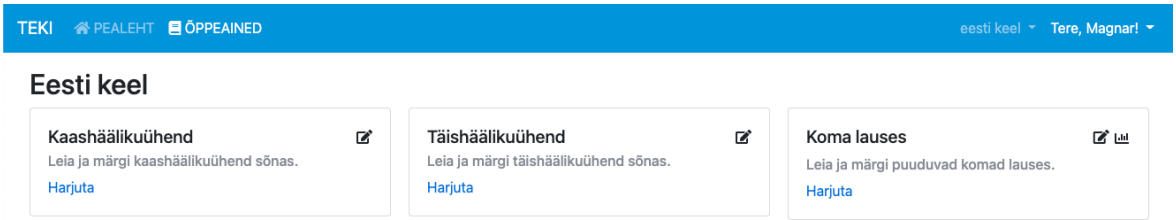

Figure 23. Estonian related topics available in TEKI

Consonant and Vowel Cluster

Consonant and vowel cluster exercises are of the same type. In the practice mode, the student is shown a word and is required to mark a consonant or vowel cluster in the word (FR4 and FR6). Additionally, the view (v. Figure 24) contains a counter for showing the number of right answers in the current practice session. After the student has marked the answer, the answer is validated by the TEKI platform. In case of a valid answer, the counter is incremented, and a success message is shown (v. Figure 24). Also, in this case, the student can choose to get the next random word for practicing. If the answer is invalid, the student will have to retry solving the same exercise again.

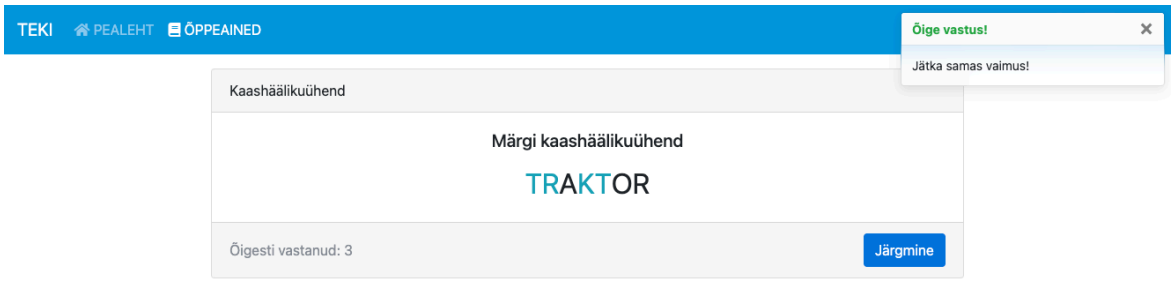

Figure 24. Consonant cluster word dialog in TEKI

Words and the corresponding clusters are first added to the TEKI platform by teachers (FR3 and FR5). To add new words for any of the cluster exercises, a dialog is opened by clicking the pencil icon on the corresponding topic card (v. Figure 23). The dialog contains a list view of all the available words and clusters added to the platform. An example of consonant cluster words is presented in Figure 25. The cluster is marked using a different colour – in this case using the teal colour[9]. The dialog contains an input field to find matching words. If a word contains the input string, the word is matched. Only the matching results are shown to the user. If the input field is empty, all words are shown.

[9] Hexadecimal colour reference value is `#17a2b8`.

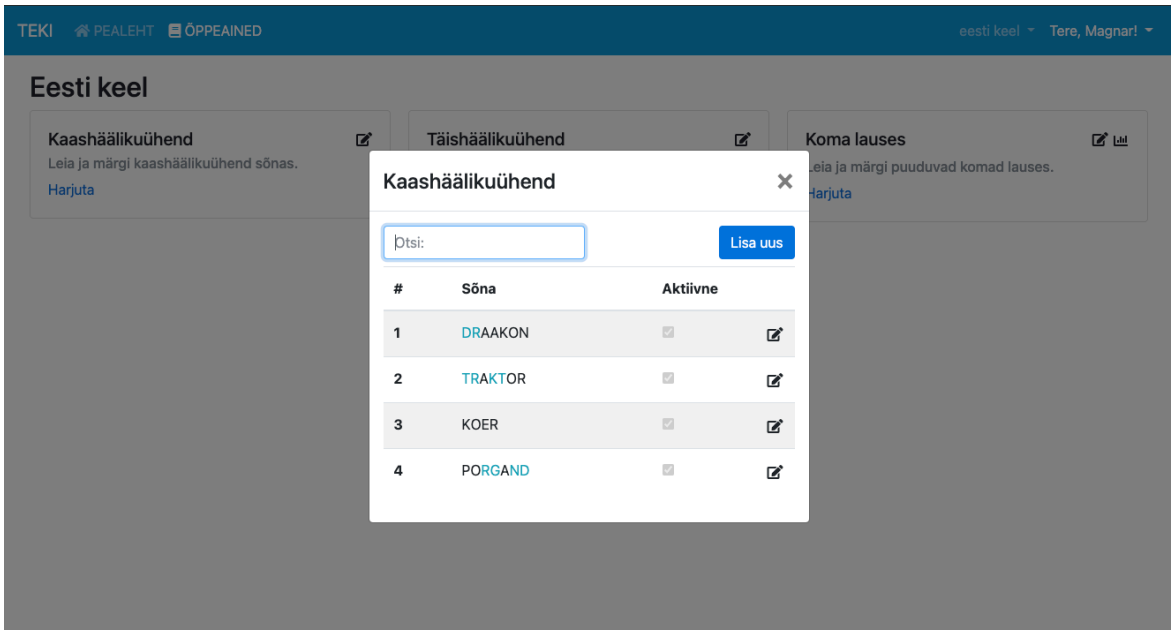

Figure 25. Consonant cluster word dialog in TEKI

For each word, four (4) columns are specified in the table (v. Figure 25). The first column contains the table row index. The second column is used to show the word and its cluster(s), if any. The third column indicates if the word is currently active or inactive. The last column contains an edit button. If the button is clicked, the word's state and clusters can be changed by clicking on the checkbox and choosing the characters that make up a cluster by clicking on them.

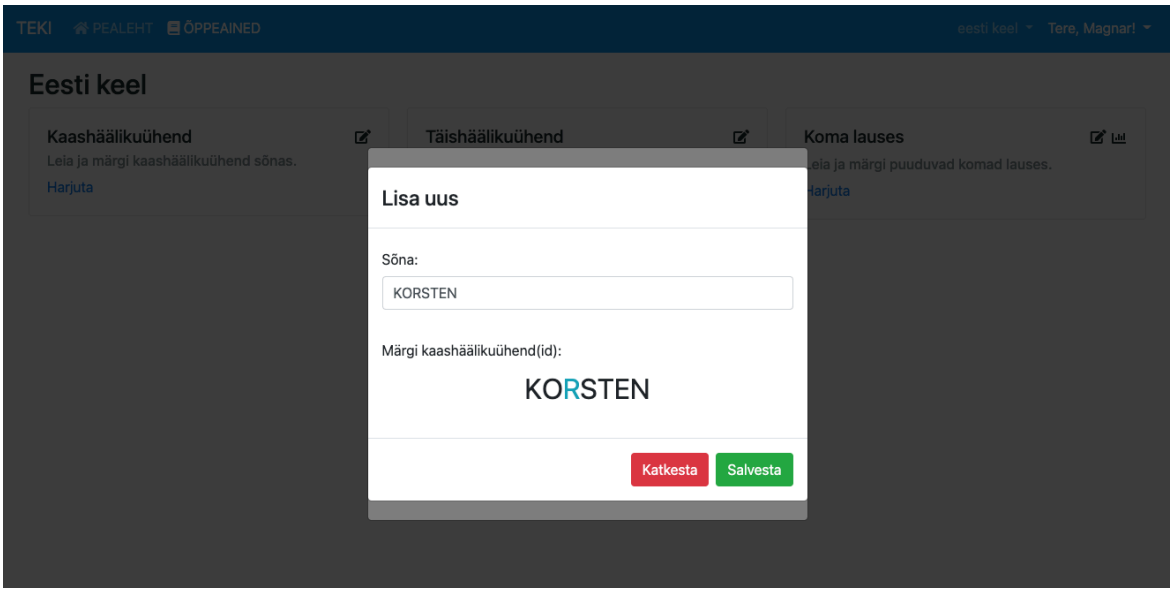

Figure 26. Add new word dialog for consonant cluster exercises in TEKI

Next to the filter input field, there is a button to add new words for the particular topic into the TEKI platform. When the button is clicked, a new dialog is opened. The dialog contains an input field for the new word. As the word is inserted into the input field, the selectable characters for cluster are shown. By clicking the selectable characters, the teacher can mark

the cluster in the word. An example of adding a new word to the consonant cluster exercises is shown in Figure 26. In this example, the first character of the consonant cluster of the word to be added is marked. The dialog has two buttons in the footer, one for cancelling the insertion and the other one to save the word with marked cluster characters into the platform. On any of the user action, the word insertion dialog is closed. If the user saves the new word, the previous dialog is refreshed, and the inserted word can be seen in the list of words.

Comma Rules

To easily practice using commas in sentences and consolidate the comma rules, the comma rules' practice feature is available. In the practice mode, the student is shown a sentence and must set comma to the right place between two words. Additionally, the user interface of the comma rules' feature contains a counter for showing the number of right answers in the current practice session. When commas are set, the answer is validated by the platform. In case of a valid answer, the counter is incremented, and a success message is shown (Figure 27). Also, in this case, the learner can choose to get the next random sentence for practicing. If the answer is invalid, the learner will have to retry solving the same exercise again.

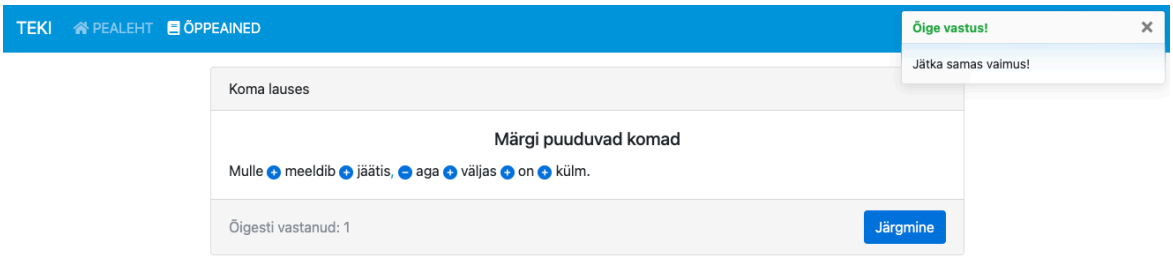

Figure 27. Comma rule practicing in TEKI

Similarly, like in cluster-related exercises, a teacher can add new sentences into the platform for students practicing the rules. To add a new sentence, a dialog is opened by clicking the pencil icon on the corresponding topic card (Figure 23). The dialog contains a list view of all the sentences available on the platform. An example of the comma rule sentences view dialog is shown in Figure 28. The dialog contains an input field for finding matching sentences. If a sentence contains the input string, the sentence is matched. Only the matching results are shown to the teacher. If the input field is empty, all sentences are shown.

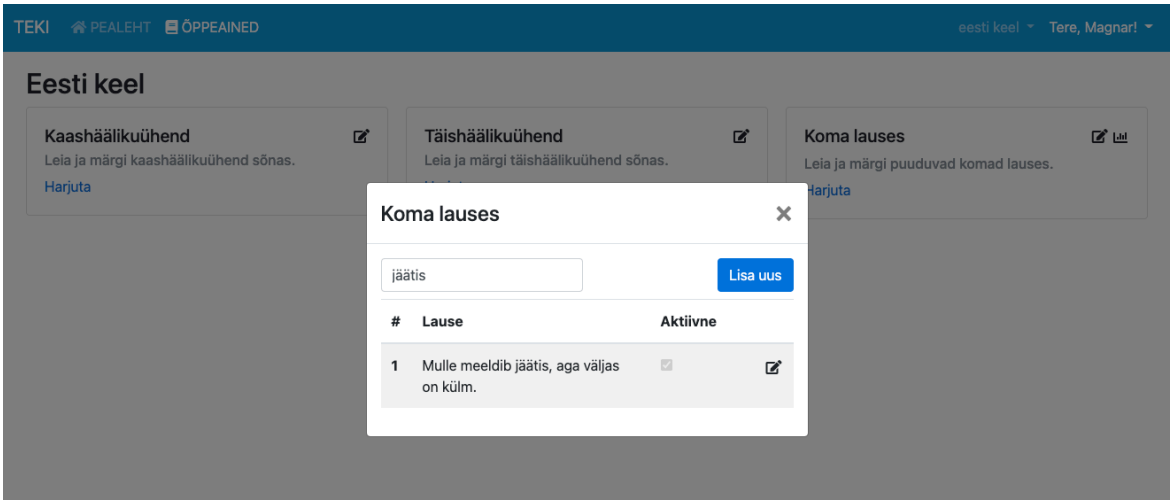

Figure 28. Comma rule sentences view dialog in TEKI, sentences are filtered by "jäätis" keyword

For each sentence, four (4) columns are specified in the dialog's table (v. Figure 28). The first column contains the sentence index. The second column is used to show the sentence value as a string. The third column indicates if the sentence is currently set as active or inactive state. The last column contains an edit icon. If the icon button is clicked, the active state of the sentence can be changed using the checkbox. Only the active sentences are used in random exercise generation for students.

Next to the filter input field, there is a button for adding new sentences to the platform. When the button is clicked, a new dialog is opened. The dialog contains a text area field for a new sentence. The example of adding a new sentence to the comma rules' exercise is shown in Figure 29.

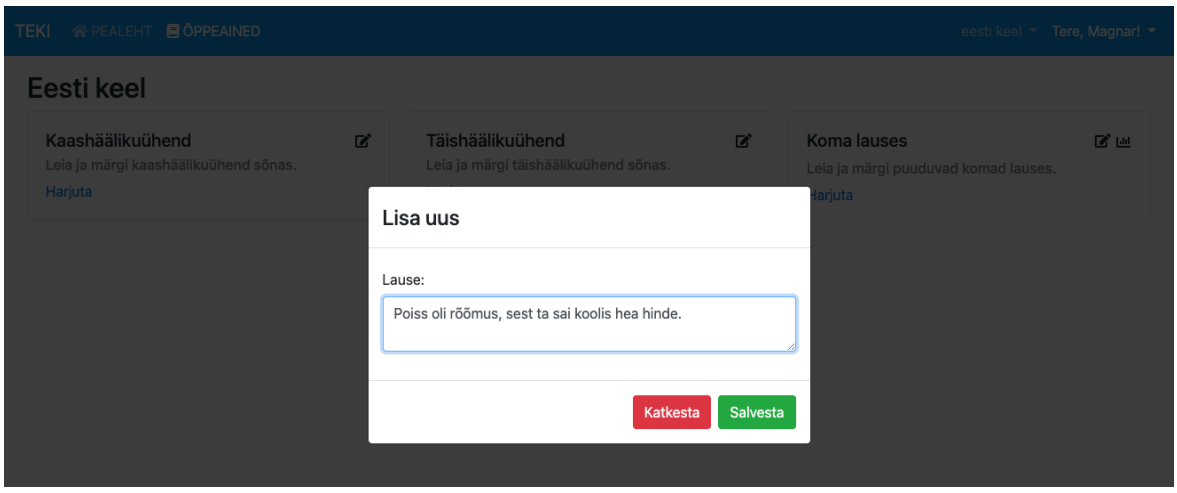

Figure 29. Add new sentence dialog for comma rules exercises in TEKI

The dialog contains two buttons in the footer: one for cancelling the insertion and the other one for saving the sentence into the platform. On any action by the teacher, the dialog is closed. If the teacher saves the new sentence, the previous dialog is refreshed, and the inserted sentence can be seen in the list of sentences. The platform automatically recognizes commas in sentences and will exclude commas when generating an exercise for a student.

As an additional feature, there is also the statistics view available for comma rules. This view can be reached by clicking the graph icon on the comma rules card (v. Figure 23). A dialog is shown that consists of the table of students and the valid result count for each student. The list can be filtered by means of the input field on top of the table. Only the students with names matching the search value in the input field are shown. If the search field is blank, all entries will be shown. Only the students who have at least one valid practice result will be shown in the dialog's table. An example of the results is shown in Figure 30.

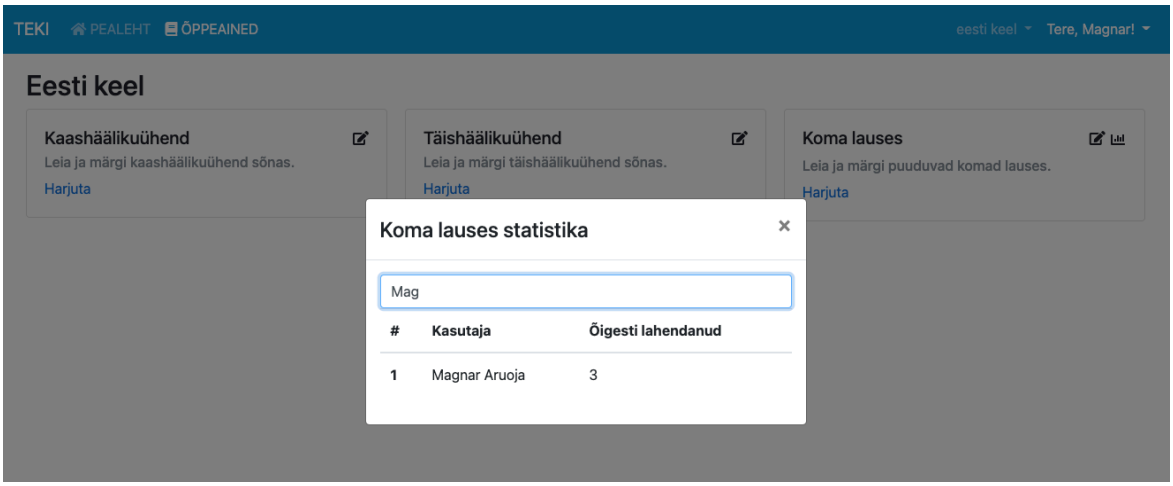

Figure 30. Comma rules' statistical view in TEKI, names containing "Mag" are filtered out

4.2.5 Mathematics Subject

On the TEKI platform, four (4) topics of mathematics have been implemented. These topics are calculations with *natural numbers*, *decimal fractions*, *common fractions* and *solving a rectangle*. The exercises for each topic are generated automatically by the platform. As a bonus feature, a button leading to the statistics dialog is shown for some topics, containing the total number of successfully completed exercises for the given learner. The opening page of the mathematics-related topics available on the TEKI platform is depicted in Figure 31.

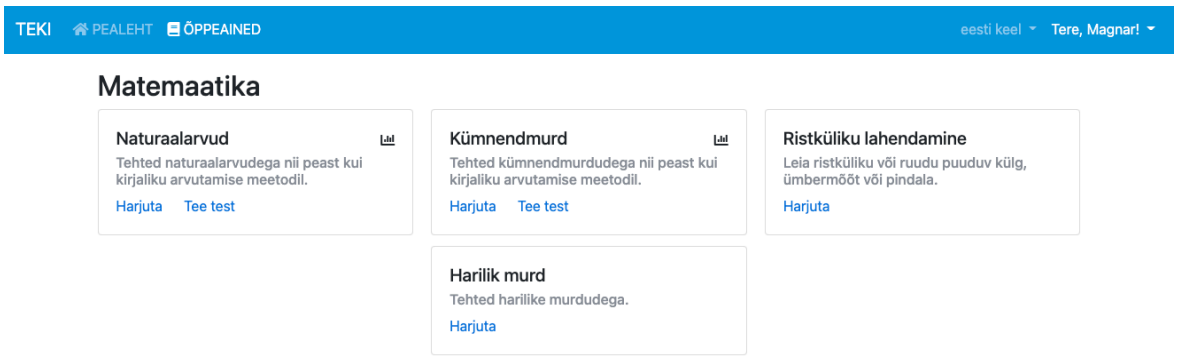

Figure 31. Mathematics related topics available in TEKI

Natural Numbers and Decimal Fractions

Exercises on natural numbers and decimal fractions belong to the same type. For these topics, the assessment mode and statistics have been implemented, in addition to the practice mode.

In the practice mode, when the student selects any of the topics, natural numbers or decimal fractions (FR8 and FR10), a dialog with exercise options is shown. The student can choose between the calculation methods and operations. Two calculation methods are available – mental calculation and columnar calculation. Three types of calculation operations are supported – addition, subtraction, and multiplication. An example of choosing practice options for calculations with natural numbers is shown in Figure 32. The student will be redirected back to the mathematics topics' page when he or she decides to discard the selection between the options.

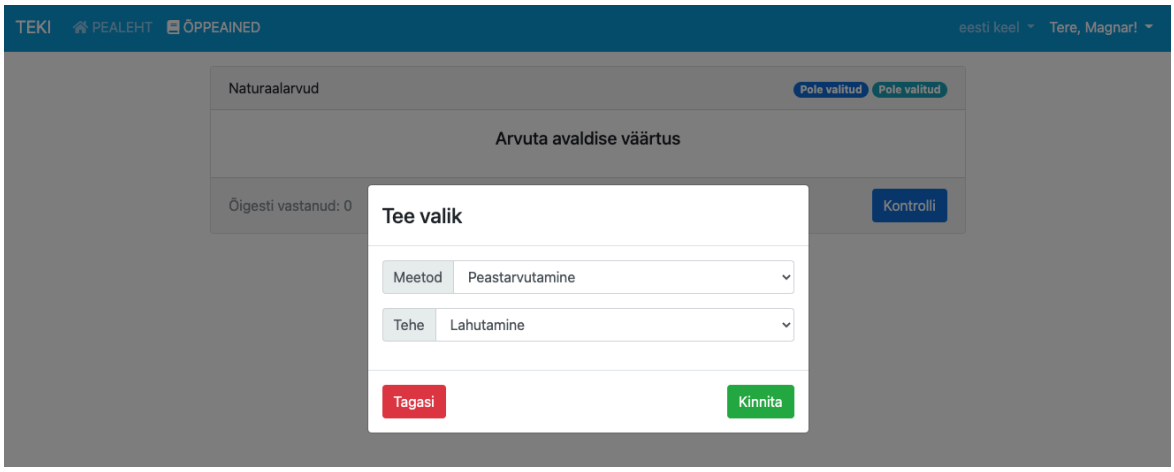

Figure 32. Natural numbers' practice options selection in TEKI

Independently of the calculation methods, the exercise view contains a counter showing the number of right answers by the learner in the current practice session. If the student has marked the answer, the answer is validated by the platform. In case of a valid answer, the counter is incremented, and a success message is shown. Also, in this case, the student can choose to get the next equation generated for practicing (FR7 and FR9). If the answer is invalid, the student will have to retry solving the same exercise again.

In the top right corner of the exercise card header, the tags for the chosen practice mode calculation method and calculation operation are shown. If the student clicks on any of the tags, the practice options dialog with the available options is opened and the student can change the practice options. If the student confirms the change, an equation making use of the new options will be generated.

For a mental calculation, an exercise with the selected operation is generated. The student is shown the generated equation and an input field for the answer. A mental calculation with natural numbers is exemplified by Figure 33.

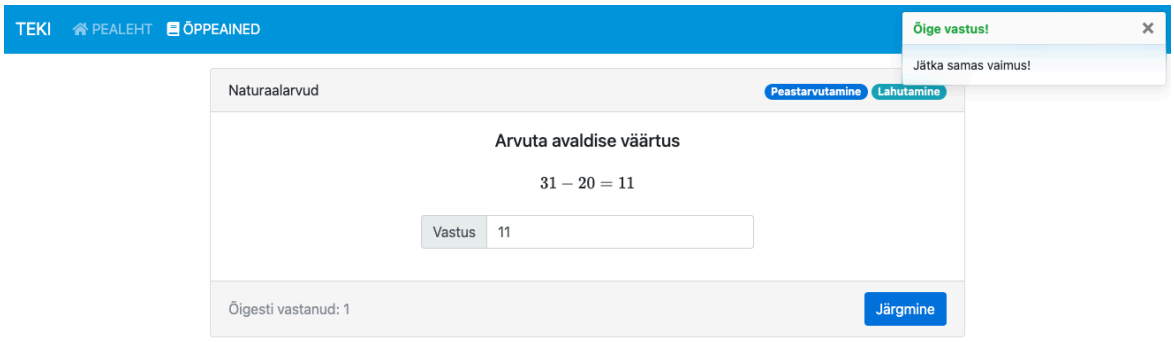

Figure 33. Natural numbers' mental calculation subtraction practice example in TEKI

For columnar calculation, similarly, an exercise with the selected operation is generated. The student will be shown the equation and columnar calculation cells with input fields forming the answer. In the columnar calculation mode, in addition to the result message in the top right corner, invalid rows of the calculation result will be marked as erroneous with red background. Also, for a decimal fraction equation answer, the system expects the decimal separator to be inserted in the same field with the number that is followed by the separator. For example, if the answer is *"18,24"*, the separator is expected to be inserted in the same field with number 8 as *"8,"*. An example of the columnar calculation practice view is shown in Figure 34.

Figure 34. Decimal fractions' columnar calculation multiplication practice invalid answer example in TEKI

All equations are generated using similar algorithm – two numbers are generated between the given range, from 1 to 100 for natural numbers and from 0,1 to 10,0 for decimal fractions. The number generated having higher value will be assigned to the left side of the operation, and the one with smaller value to the right.

In addition to the practice mode, the assessment mode was implemented for calculations with both natural numbers and decimal fractions. If the assessment mode is selected on the topic selection page, the dialog window with the table containing a list of available assessments will be shown to the student. The table consists of four (4) columns. The first and second column respectively contains the number and name of the assessment. The third

column shows the creation time of the assessment. The fourth column has an action button to start the assessment. When clicking the button, an assessment with the chosen assessment configuration is generated for the student. It means that for assessments, there are configurations created in the system, and when an assessment is started, assessment tasks are generated based on the configuration meaning that each student will get unique exercises for the identical assessment configurations (v. Figure 35).

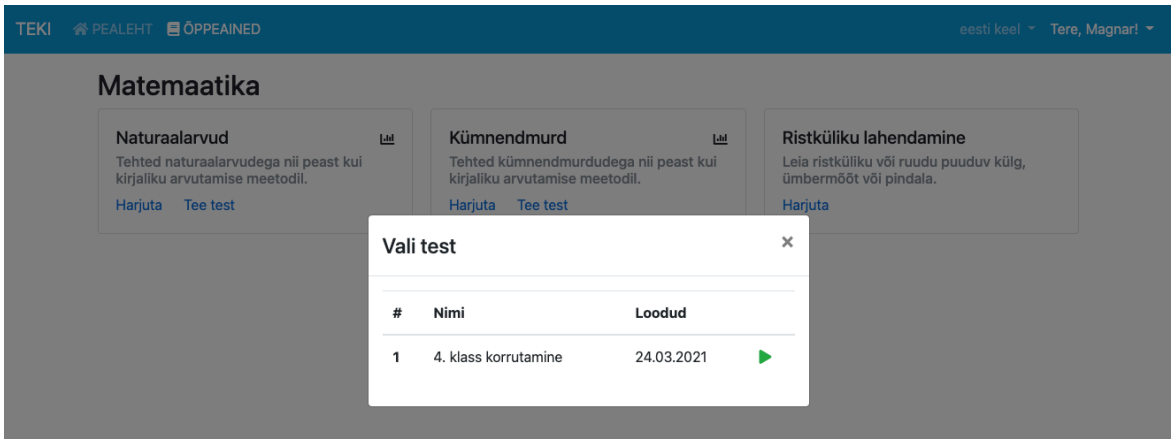

Figure 35. Natural numbers' assessment dialog in TEKI

In the assessment mode, the student will be provided with all the tasks in the beginning of the assessment. The visual of this mode is similar to the visual of the practice mode, but instead of showing a valid exercise count, a student is shown the assessment progress in the top right corner of the card. This includes showing the current task and the task count in total. In addition to that, the student is allowed to navigate between the assessment tasks to recheck them before the final submission. The user interface of the assessment mode is represented in Figure 36. Once the student has filled in all the answers, the answers will be sent to the system where the assessment validation result is stored to the database.

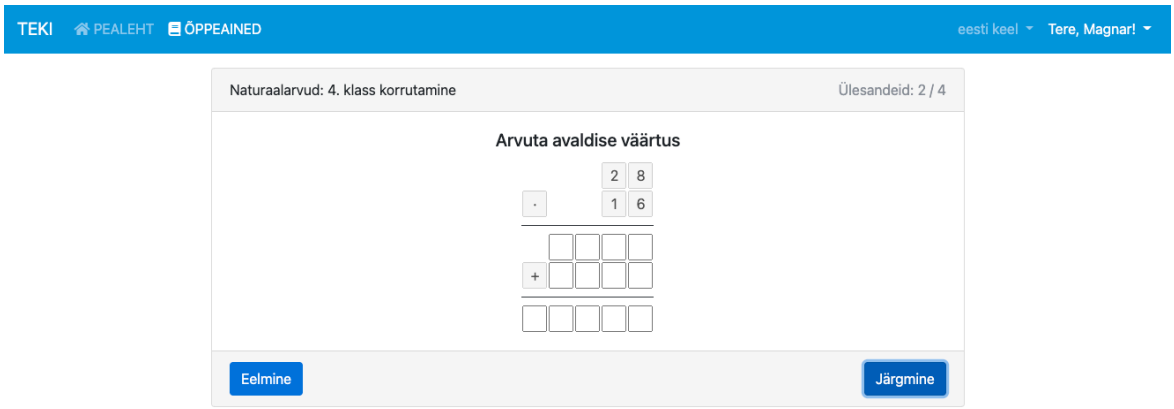

Figure 36. Natural numbers' assessment view in TEKI

Solving Rectangle

The practice mode of solving a rectangle includes the following six (6) types of tasks:

- find rectangle width by area;
- find rectangle width by perimeter;
- find rectangle height by area;

- find rectangle height by perimeter;
- find rectangle area;
- find rectangle perimeter.

In general, the practice mode of solving a rectangle works like any other practice mode on the topics of Estonian or mathematics. In the MVP, a task of random type is generated for practicing (FR11). The student is shown an exercise of solving a rectangle (v. Figure 37). The exercise has some information provided and the missing part is expected to be input by

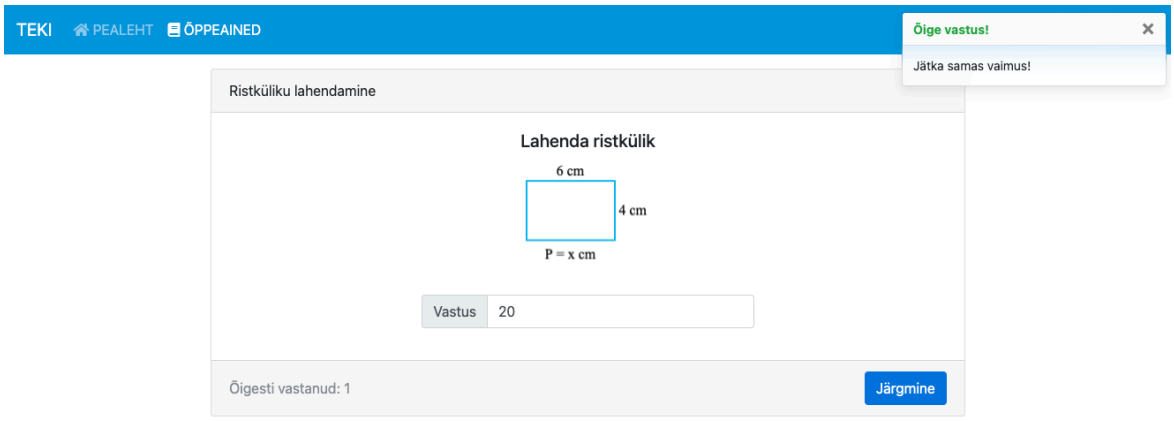

Figure 37. Solving rectangle practice mode in TEKI

the learner as an answer to the exercise (FR12). In the example shown in Figure 37, the student is provided the height and width of a rectangle, and the learner is expected to calculate the perimeter of the rectangle.

Common Fractions

The practice mode of calculations with common fractions is similar to the calculations with natural numbers and decimal fractions. For calculations with common fraction, the platform generates random equations (FR13). The practice mode of calculations with common fractions supports all basic operations, such as addition, subtraction, multiplication, and division. In this practice mode, the student must provide both numerator and denominator for the answer (FR14). The exercise view (v. Figure 38) contains the counter, action buttons and messages in the top right corner indicating the exercise results.

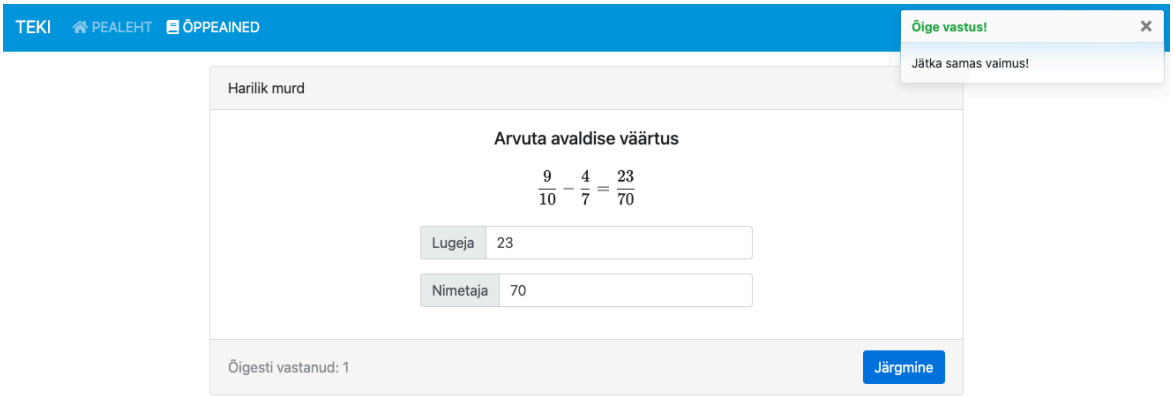

Figure 38. Common fraction practice view in TEKI

5 Software Implementation

This chapter gives an overview of the software implementation process of the platform including examples and a justified choice of each technology applied.

5.1 System Requirements

TEKI is a platform that consists of multiple microservices. To run TEKI, the following requirements must be met:

- An HTTP server serving TEKI front-end application, such as Apache or Nginx;
- A JDK[10] 11+ running TEKI back-end Keycloak applications;
- A MongoDB server for business data persistence;
- A Postgres database server for Keycloak data persistence;
- At least 1 CPU core with a clock rate of 1 GHz;
- At least 2 GB memory available in the system (to run all the applications and database servers);

To facilitate the platform management, the applications and databases should be divided into containers. For this, also a Docker is required. Usage of Docker images and possible division of the platform into containers is addressed by Section 5.5 of this thesis.

5.2 Development methodology

Software development is usually managed by project manager. The project manager's job is to keep track of the project and make sure the team is not losing track. To ease project management, numerous methodologies have been developed for software development. Depending on the product or project, the team and the deadlines, the most appropriate methodology should be used. The methodologies are split into two [9] – lightweight and heavyweight methodologies – respectively following the agile and waterfall models. Each development methodology is characterized by a fixed set of roles and activities.

For the platform addressed by this thesis, the author of this thesis has undertaken multiple roles such as Software Developer, DevOps Engineer, Quality Assurance Specialist, Project Manager, and Product Owner. Despite the fixed set of requirements for the platform and a fixed deadline for the thesis, a combination of multiple agile methodologies, instead of waterfall methodologies, was applied, as agile methodologies emphasize iterative and incremental software development.

The agile methodology applied was Rapid Application Development. Rapid Application Development (RAD) [10] is a development methodology that puts a lot of emphasis on development and user feedback instead of strict planning. This is useful for building new features from scratch, allowing users to interact with the evolving system in short cycles, and introducing new feature requests or changes in requirements. In addition to RAD, some elements of Kanban [11] were applied such as prioritizing the work and viewing the whole progress of the platform being built. Mixing such elements from Kanban made creating short-cycle iterations for RAD easier.

The biggest challenge in building a new platform was to develop an intuitive and responsive user interface that would be easy to use for both teachers and students. To achieve this, direct communication and teamwork with teachers was required. To enable that, the author

[10] Java Development Kit

of this thesis and a teacher from Luunja Keskkool agreed to collaborate in building the new platform.

Together with the teacher, the functional requirements were discussed and prioritized depending on the curriculum and the study plan of the mathematics teacher's students. This allowed to prepare for the evaluation of the platform, as it was important that the topics implemented in the platform would be the ones taught at school in the same year. For example, as the students of grades 4 and 5 learning mathematics were known to be taught columnar calculation later that year, implementing the columnar calculation feature in TEKI had a top priority.

In addition to that, working with the particular stakeholder – teacher – aroused more interest among other teachers at Luunja Keskkool. This resulted in new feature requests from the teachers. For example, during the implementation of the columnar calculation, teachers requested to have the statistics' view, to see how many times each student has practiced. Moreover, they also requested to have an assessment mode for columnar calculations, meaning that for each student, for example, 6 equations of multiplication would be generated and no instant feedback would be given. Instead of the instant feedback, the student would submit all the answers at once and the teacher would later check the results and provide the student with the feedback.

Overall, tight cooperation with teachers, according to the RAD and Kanban methodologies, made the development process very effective and this way proved to be useful in developing a new platform from scratch.

5.3 Frontend technologies used

Frontend (or frontend application) is a client application used to interact with the system's backend server. In web applications, the users of the system use the frontend application through any available web browser on their device. A typical web application client application consists of templates (HTML), styles (CSS) and scripts (JS) creating a seamless user experience for the end-user.

In recent years, single-page applications (SPA) have become popular in the field. Instead of rendering entire pages all over again, in SPAs, the content is dynamically reloaded making the user experience smoother. The frontend technologies considered for building the TEKI's client-side application were Angular, React, and Vue.js.

In Sections 5.2.1-5.2.2, an overview is given of the choices of frontend technologies applied for developing the TEKI platform.

Because of the author's prior experience with Angular and its support for building both mobile and desktop applications, Angular was chosen as the framework to be used. In addition to that, Angular has already many useful features implemented, unlike, for example, React which requires custom implementation or combining with other frontend libraries.

In Sections 5.2.1-5.2.2, an overview is given of the frontend technologies chosen for developing the TEKI platform.

5.3.1 Angular

Angular [12] is a development platform for building single-page applications in TypeScript. Angular applications can be initialized using *angular-cli*[11]. TEKI's frontend application

[11] https://angular.io/cli

base was generated using ngX-Rocket[12]. ngX-Rocket is an open-source Angular project generator based on the *angular-cli*. It allows to generate an Angular application using ready-to-use Bootstrap[13] user interface components including Font Awesome[14] icons and a support for internalization (*i18n*) using *ngx-translate*[15].

In addition to Angular (version 11.0.9) and its related core libraries, a set of open-source third-party libraries were used for developing the frontend application. The list of used libraries is described in Table 4.

Table 4. Third-party frontend libraries used

Library	Version	Description
keycloak-angular	8.1.0	An Angular wrapper for keycloak-js.
keycloak-js	12.0.4	Provides a set of useful functions to easily connect frontend application with Keycloak
lodash	4.17.20	A JavaScript utility library to ease working with arrays, numbers, objects, strings, etc. For example, provides functions to iterate and manipulate arrays using functional style.
ng-katex	2.0.3	An Angular wrapper for KaTeX library used to write mathematical expressions (TeX rendering).

5.3.2 Bootstrap

Bootstrap [13] is an open-source front-end toolkit for designing responsive mobile-first webpages. Bootstrap includes extensive prebuilt components and powerful plugins to enrich user interface. In addition to that, Bootstrap supports customizing the existing components and styles through Sass variables and mixins. All of this facilitates the development of a web application that is used on both mobile and desktop devices (NFR 1 – 2).

Bootstrap version 4.5.0 was used to create responsive user interface for TEKI.

5.4 Backend technologies used

In web application development, backend refers to the parts and applications of the web server. TEKI's backend consists of multiple applications and databases, such as an application server, an operational database, an authentication manager, and a database for the authentication manager.

For all the parts of the backend, there are many programming languages, frameworks, and tools to choose from. For example, there are Java web frameworks, such as Spring Boot, Apache Spark and Micronaut, and Python web frameworks, such as Django and Flask, or JavaScript web frameworks, such as Express.js and Next.js.

[12] https://ngx-rocket.com/home
[13] https://getbootstrap.com/
[14] https://fontawesome.com/
[15] https://github.com/ngx-translate/core

For data persistence, two types of databases are widely used – relational (SQL) databases and document-oriented (NoSQL) databases. The relational databases include, for example, Microsoft SQL Server, Oracle, PostgreSQL, and MySQL. The document-oriented databases include, for example, MongoDB and DynamoDB.

To develop the backend for TEKI, a Spring Boot was chosen as the framework to implement the core application logic, and MongoDB was chosen for data persistence. The author of this thesis has had prior experience with both technologies, and in addition, both the Spring Boot and MongoDB tools are free and available as open source. Keycloak was used for implementing the authentication manager as it supports easy setup and integration with both Spring and Angular application, and the author of this thesis has also had prior experience with Keycloak.

In Sections 5.3.1-5.3.3, an overview is given of the backend technologies chosen for developing the TEKI platform.

5.4.1 Spring Boot

Spring Boot [14] is a framework built on top of the Spring framework that includes a combination of the Spring framework tools, third-party libraries, and configurations. Spring Boot enables to get a new production-grade web application server up and running with a minimal overhead.

Spring Boot applications are stand-alone web applications that do not require any external web server to run Java applications. Instead, Spring Boot applications are bundled with Jetty or Tomcat and are run as *fat jar*s[16]. In addition to that, Spring Boot also provides production-ready features such as metrics and support externalized configuration.

For TEKI, the Spring Boot project was generated using Spring Initializr[17]. Spring Initializr is a simple web application used to generate base for Spring Boot projects. While generating the project, a developer can choose between Maven or Gradle project, the version of Spring Boot used, the programming language used (Java, Kotlin or Groovy), and add project related metadata, such as project's group, artifact name, and description. In addition to that, the developer can select a list of common Spring and third-party libraries that should be added to the project. TEKI is a Spring Boot (version 2.4.1) project, written in Java 11, and managed by Gradle[18] (version 6.8) build tool. The list of TEKI's Spring Boot application dependencies is presented in Table 5.

[16] *Fat jar* is a type of Java application containing all the required classes and resource files to run the application, including dependencies, and for web application, also a web server such as Tomcat or Jetty.
[17] https://start.spring.io/
[18] https://gradle.org/

Table 5. TEKI's Spring Boot application dependencies

Library	Version	Description
commons-collections4	4.4	A Java library containing helper utility classes and methods to work with collections, such as *Set* and *List*, in Java.
commons-lang3	3.11	A Java library containing helper utility classes and methods to work with classes in *java.lang* package, such as *String* for example.
commons-math3	3.6.1	A Java library containing helper utility classes related to mathematics and statistics, for example, an implementation to greatest common divisor (gcd).
keycloak-admin-client	12.0.4	A library containing helper classes and resource information for integration with Keycloak Admin's REST API.
lombok	1.18.16	A Java library and annotation processor to reduce the need of writing Java boilerplate code such as getters and setters.
sbs*-actuator	2.4.1	An extension for Spring Boot application including additional features to monitor and manage the application.
sbs*-data-mongodb	2.4.1	An extension for Spring Boot application including the required drivers and classes to connect Spring Boot application with MongoDB database.
sbs*-security	2.4.1	An extension for Spring Boot application including the configuration classes to add security to the application, such as authentication and authorization for web endpoints and integration with Keycloak.
sbs*-validation	2.4.1	An extension for Spring Boot application to add resource validation support using Hibernate Validators, for example.

* spring-boot-starter

5.4.2 MongoDB

MongoDB [15] is a free, open-source, cross-platform, NoSQL database using JSON-like documents for data storage. As the data is stored in documents, it is easy to map the objects

with the application code. Unlike relational databases, MongoDB does not require schemas or support transactions. MongoDB is a good choice for systems that require high availability, as it supports horizontal scaling and has a distributed database at its core. MongoDB provides a nice and easy aggregation pipeline JSON-like API for real-time data aggregation.

Due to not making up of schemas, MongoDB is also a good choice for creating prototypes and MVPs where the data and its schema frequently change. As the database operates in-memory whenever possible, the speed and performance of the database are very high compared to other databases.

5.4.3 Keycloak

Keycloak [16] is an open-source identity and authentication manager. Keycloak provides centralized management for users and their roles. User management can be performed using the Keycloak's web interface. The user management view allows a Keycloak administrator to add, modify or delete any user. Furthermore, user attributes like roles and credentials can be configured. An example of the user management page is represented in Figure 39.

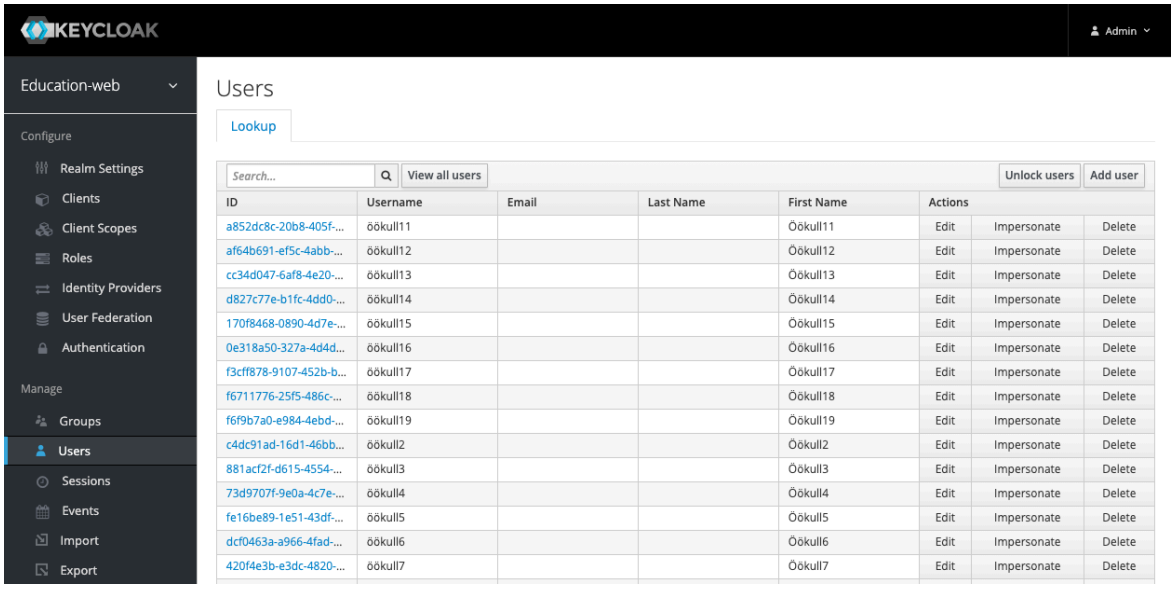

Figure 39. Keycloak's user management view

Keycloak provides a special console enabling administrators to configure the Keycloak realms and clients, enable or disable various features of Keycloak, and define fine-grained authorization policies. In addition to that, Keycloak supports integration with different identity providers such as OpenId, Github, Facebook, and Google, for authentication using the accounts of social networking services, and user federation through LDAP[19] or Kerberos[20].

To integrate Keycloak with applications, a Keycloak's realm and clients must be created. This can be easily done by following the Keycloak's documentation. For example, *education-web* realm and *education-web-client* client could be created accordingly. The Keycloak application requires a relational database. For authentication management of TEKI, the PostgreSQL database was set up and configured.

[19] The Lightweight Directory Access Protocol
[20] A Network Authentication Protocol (https://web.mit.edu/kerberos/)

5.5 Deployment

To enable the usage of TEKI by teachers and students, the platform must be made available to be accessed from the public web. For deployment, all the applications forming parts of TEKI were divided into containers for easier management in the production environment. In addition to that, it was important to configure a reverse proxy server for HTTP traffic routing and balancing between the corresponding services. The policy of not exposing the services directly improves the platform's security, as this way together with the firewall, unnecessary ports and paths can be hidden from the public. The platform, managed by Docker, is deployed on the virtual machine in the cloud.

5.5.1 Docker

TEKI is a platform combined of multiple services that are managed as Docker containers. Docker [17] is supported on all popular operating systems, such as Linux, MacOS (Docker Desktop for Mac), and Windows (Docker Desktop for Windows). On Linux, Docker is available for a variety of Linux platforms, such as Centos, Debian, Fedora, Raspbian and Ubuntu.

To run TEKI frontend and backend in a container, Docker images for respective applications must be built. For both applications, a Dockerfile[21] is included in the repository.

The TEKI frontend image is built using Node.js[22] and Nginx images. The TEKI frontend container is thus a Nginx container configured to serve Angular application files and proxy the requests to respectively the TEKI backend and Keycloak.

The TEKI backend container requires a Docker image with Java support and a packaged *jar* file of the given application. The *jar* file can be built using Gradle.

In Sections 5.4.1-5.4.4, an overview is given of the technologies chosen for deployment of the TEKI platform.

In addition to that, the TEKI backend requires a MongoDB database for data persistence and an integration with the Keycloak application for user management. All the required Docker images are publicly available and are listed in Table 6.

[21] https://docs.docker.com/engine/reference/builder/
[22] https://nodejs.org/en/

Table 6. Third-party Docker images required for dockerizing TEKI

Image	Description
adoptopenjdk/openjdk11	A Docker image for creating container with JDK 11 support and running Spring Boot application in it.
mongo	A Docker image for creating MongoDB container.
node	A Docker image for creating container with Node.js support.
nginx	A Docker image for creating Nginx server container for reverse proxy and serving Angular application files.
quay.io/keycloak/keycloak	A Docker image for creating Keycloak application container.
postgres	A Docker image for creating PostgreSQL container.

5.5.2 Nginx

Nginx [18] is a popular, open-source HTTP and reverse proxy server. Nginx can be used, for example, for serving static files, adding SSL support for a web server, handling redirection on different error codes, and load balancing. Nginx has been used by many large platforms all over the world, such as Dropbox and Netflix.

In the local development mode, using a separate reverse proxy server is rather uncommon because the webpack server used for local Angular application development already supports http-proxy [19].

In production environments, Nginx is mostly used for load balancing the incoming HTTP traffic. In an architecture that is based on microservices, a reverse proxy is used to proxy requests from different routes to different services.

For TEKI, Nginx is used in the production environment for serving Angular application resources and for routing traffic to the TEKI Spring Boot application.

5.5.3 Gitlab and Pipelines

Gitlab [20] is an open-source DevOps platform that includes a built-in version control system, issue tracking, code review, continuous integration (CI) and delivery (CD), and many other features. All these features make Gitlab a single place to manage everything related to the project and product development. The Gitlab platform is available in two editions – Community Edition, which is freely available under the MIT license, and Enterprise Edition, including additional features for enterprises and other organizations.

Gitlab supports both public and private repositories and includes a container registry for each repository. All source code, configurations and deployments of the TEKI platform are managed by the version control of Gitlab. For example, the following private repositories are available under *education-web* group (v. Figure 40):

- **education-api** – a repository for TEKI's Spring Boot application source and related configurations for deployment using Gitlab CI/CD features;
- **education-client** – a repository for TEKI's Angular application source and related configurations for deployment using Gitlab CI/CD features;
- **keycloak** – a repository for managing Keycloak's (and its Postgres database) version and automated deployment using Gitlab CI/CD features;
- **mongo-db** – a repository for managing MongoDB version and automated deployment using Gitlab CI/CD features.

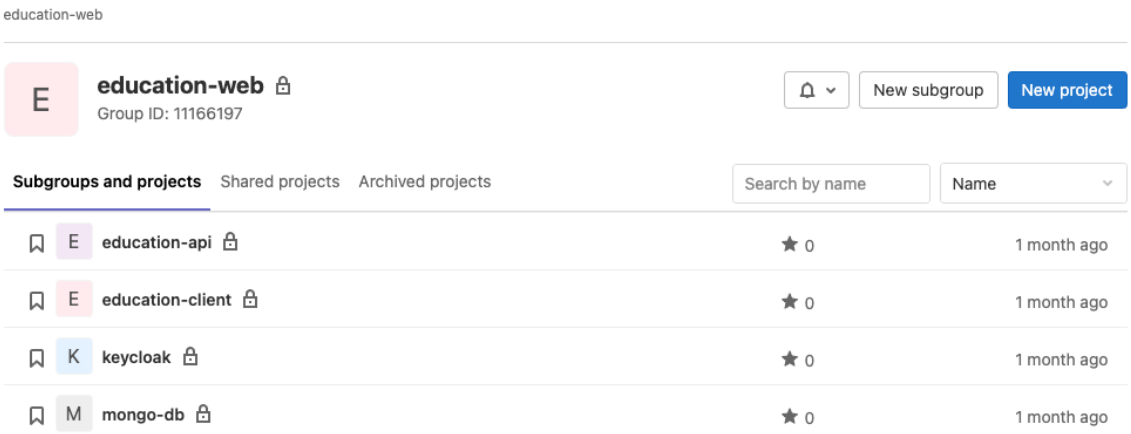

Figure 40. Education-web group in Gitlab, containing all repositories of the TEKI platform

For each repository, there is a Gitlab CI pipeline configured using the *gitlab-ci.yml* file in the project's root repository (for example, see Appendix III). Gitlab CI pipelines support splitting the continuous integration into different stages and jobs. For example, the build stage includes building jobs, where the source code is compiled and verified by means of unit and integration tests. Similarly, the publish stage includes publishing jobs, where Docker images are created and pushed to the Gitlab's registry. Finally, at the deployment stage, a connection to the remote server is established and a new version of the application is deployed.

Gitlab supports deployments in multiple environments. These can be configured using Gitlab's Environments feature. For TEKI, a *production* environment has been configured using the external URL of the web server.

Also, Gitlab supports the usage of secrets in pipelines so that they are not leaked into the repository. For each repository, or globally for the group of repositories, CI/CD variables can be set. This is useful for storing secrets and credentials relevant for the production environment of a specific application. Viewing and modifying the CI/CD variables is only available to authorized users. The owner of the group or the owner of the specific repository can add or remove permissions.

All component applications of TEKI are built and deployed to the production environment on every commit following a successful CI pipeline, meaning that the application integration and delivery are fully automated. A snapshot of a successful Gitlab CI pipeline is depicted in Figure 41.

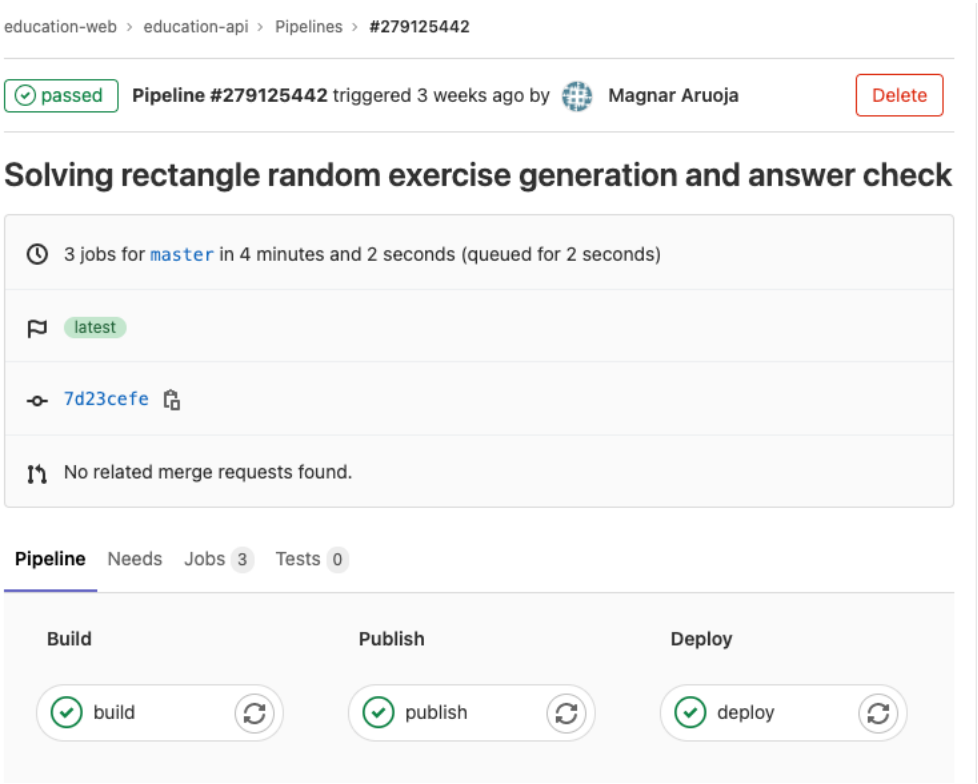

Figure 41. A snapshot of successful education-api CI pipeline in Gitlab, all stages have completed successfully in 4 minutes

5.5.4 DigitalOcean

DigitalOcean [21] is a cloud platform for hosting and managing applications. DigitalOcean products and services are categorized into the following five (5) categories:

- **Compute** – including Droplets, which are on-demand Linux virtual machines, Kubernetes for cluster management, and App Platform, which is a fully managed solution providing all the infrastructure based on the application source code;
- **Storage** – for data management in cloud;
- **Networking** – support for cloud firewalls, load balancing, floating IP-s, and DNS;
- **Databases** – fully managed PostgreSQL, MySQL or Redis instances;
- **Tools** – a set of development and management tools, such as command-line interface, RESTful API for managing DigitalOcean related infrastructure, monitoring applications and managing projects.

Due to the features supported and a beneficial pricing model, DigitalOcean was chosen as the platform to host applications of the TEKI platform. In particular, a Droplet with Ubuntu image, including Docker support, were selected.

6 Software Evaluation

This chapter gives an overview of the software evaluation process for the TEKI platform. The platform's functionality was evaluated through multiple stages throughout the development of the platform. The evaluation process consisted of automated unit and integration tests for both frontend and backend applications and a manual User Acceptance Testing (UAT) in the production environment in cooperation with the teachers from Luunja Keskkool. In addition, the platform was evaluated by students from Luunja Keskkool, including practicing topics of mathematics and performing e-assessments by the TEKI platform. At the end of the evaluation period, both students and teachers were asked to fill in the feedback survey.

6.1 Automated tests

Most of the features of the software developed were covered by means of automated tests. As the business logic for the features was mainly embedded in the backend, a lot of emphasis was put to cover with tests the backend application. The tests that were embedded in the backend included unit, component, and integration tests.

Unit and component tests were created to cover a logic of a single component and isolating the dependencies. This was the best way to cover the logic of generating exercises and checking answers by the students. For unit tests, the JUnit 5[23] framework was applied in combination with AssertJ[24].

In addition to unit tests, integration tests were implemented. Integration tests were used to test the key integration parts of the application, such as integration with the MongoDB database and the API presentation layer (REST). The tests were created by using the Spring Boot Test[25] framework. A snapshot of successful automated tests is shown in Figure 42.

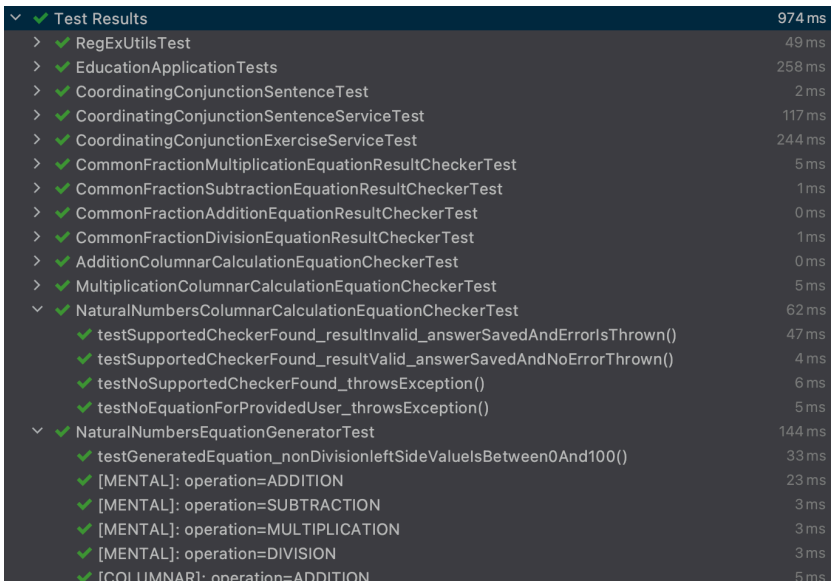

Figure 42. A snapshot of a successful TEKI backend application test run

[23] https://junit.org/junit5/
[24] https://assertj.github.io/doc/
[25] https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/#boot-features-testing

6.2 Production environment

The production environment was configured using the DigitalOcean platform. For TEKI, a Droplet with Unix and Docker support was created. The Droplet configuration ($10/month) selected for TEKI platform included (v. Figure 43):

- 1 vCPU;
- 2 GB RAM;
- 25 GB disk space;
- Static IP-address.

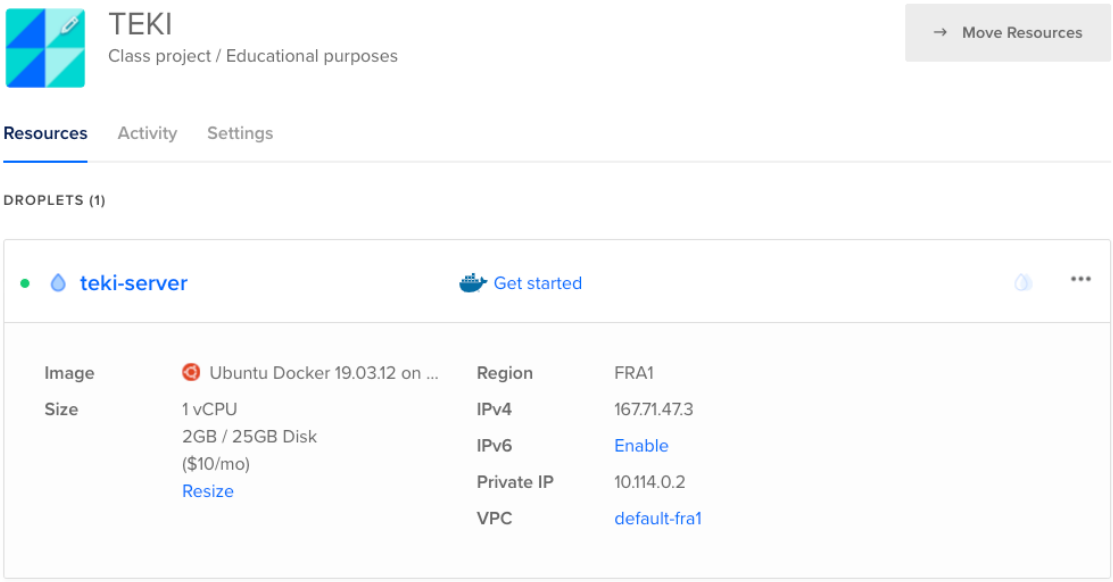

Figure 43. The Droplet server for the TEKI platform hosted on DigitalOcean

The Droplet server was assigned the IP-address 167.71.47.3. To make the platform easily accessible, the domain name *teki.ee* was ordered and configured using Zone[26], an Estonian web-hosting platform, to point to the server hosted on DigitalOcean. As DigitalOcean supports free cloud-based firewall services, a firewall was configured for TEKI allowing inbound traffic only through ports 22 (SSH), 80 (HTTP), and 443 (HTTPS). The outbound traffic was enabled for all TCP and UDP connections.

[26] https://www.zone.ee/et/

The Nginx reverse proxy was configured to always redirect HTTP (port 80) requests to HTTPS (port 443) to make the connection between the client and server secure (see Appendix IV for the Nginx configuration sample file). In addition to that, the SSL certificate (v. Figure 44), using certbot[27], was configured to support HTTPS for TEKI.

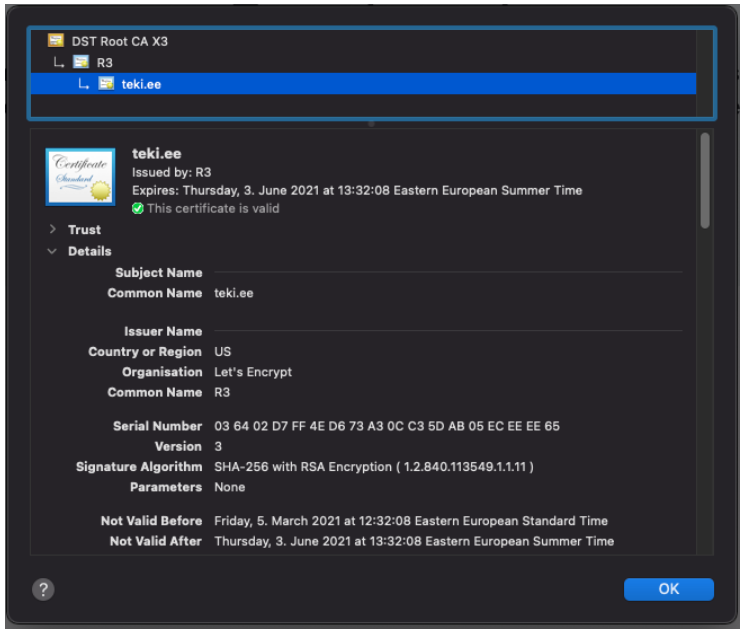

Figure 44. An SSL certificate issued by Let's Encrypt for teki.ee

6.3 Evaluation by end users

The platform was evaluated by 68 students (from four different classes) and 3 teachers in the period between the 1st of March 2021 and the 18th of April 2021. All the participants received the credentials generated for their anonymous TEKI accounts. In the beginning of the evaluation, a short introduction of the TEKI platform was presented to the participating teachers. Thereafter, the teachers taught their students how the platform should be used.

The students were from two different grades of Luunja Keskkool – 37 students from Grade 4 and 31 students from Grade 5. Depending on the grade, different topics of mathematics were evaluated. Students from Grade 4 focused on practicing columnar multiplication of natural numbers and students from Grade 5 focused on columnar addition, subtraction, and multiplication of decimal fractions.

In total, 51 of 68 (75%) students who had access to the TEKI platform, participated in the evaluation. In total, 710 exercises concerned with natural numbers were performed by the students, and 347 exercises concerned with decimal fractions were done. In addition to the exercises in mathematics, one class also had an opportunity to try out the comma rules practice feature. In total, they practiced using commas in 364 sentences. The results are represented in Figure 45.

[27] https://certbot.eff.org/

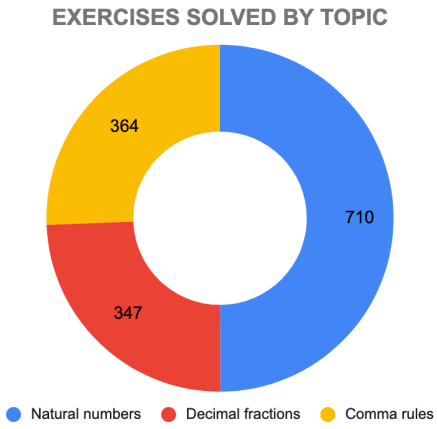

Figure 45. Exercises solved by topics

In addition to practicing performing exercises of different types, online test assessments were conducted by one teacher. To enable that, the following three tests were added to the TEKI platform:

- a test of columnar addition and subtraction of decimal fractions – three exercises of each calculation type for Grade 5;
- a test of columnar multiplication of decimal fractions – 4 exercises for students of Grade 5;
- a test of columnar multiplication of natural numbers – 4 exercises for students of Grade 4.

In total, 33 of 68 students were asked to take the test on the TEKI platform. The participation rate was 86.7%, 80%, and 100% for each test respectively. As the minimum viable product that was used for the evaluation by end users did not include a user interface enabling teachers to easily check the results of the assessments, the results to be checked were retrieved directly from the database with the help of the author of this thesis.

6.4 Feedback

Following the evaluation period, both the participating teachers and students were asked for the feedback. For this purpose, two online surveys were conducted using the Google Forms environment among the participating teachers and students, respectively. The outcomes of the reviews are summarised in the respective Sections 6.4.1 and 6.4.2, with the surveys' questions and results available in Appendices V and VI, respectively.

6.4.1 Teachers

All the teachers participating in the evaluation process were asked for the feedback. All of them (100%) agreed that the platform was easy to use, and they were also willing to use the platform in the future. They appreciated that the environment automatically generated and validated mathematical equations and the answers to solve these given by students. In addition to that, the teachers liked the intuitive user interface that enabled them to easily choose between the subjects and topics.

The teachers also mentioned some minor problems that they encountered during the evaluation of the TEKI platform. For example, some students experienced technical problems in the authentication process. However, it was not clear whether the problems were caused by

the platform or were rather due to a particular computer or internet connection. One teacher also commented upon the user interface for columnar calculation, saying that the length of the answer could be predicted by the number of input boxes in the calculation rows. To alleviate the problem, a possible improvement in the user interface would be to make the answers' cells dynamically addible and removable.

All the respondents agreed that the feature of automatic exercise generation for the topics of mathematics reduced their workload, and they all emphasized the importance of being able to see for each topic statistics about how well students had performed. Moreover, they stated that solving the exercises without the time limit and being able to see the statistics is something that is not provided by the existing environments they have been using so far.

In addition to the implemented features, the teachers mentioned some topics that should be addressed by the TEKI platform. The desired features included:

- Synonyms and antonyms in Estonian;
- Division of natural numbers and decimal fractions in mathematics;
- Number rounding and comparison in mathematics;
- Calculating the volumes of three-dimensional shapes in mathematics.

In parallel with introducing new features, the teachers would also like the platform to be integrated with the existing e-school platforms, such as Stuudium or eKool. This would allow them to mark more easily the homework assignments on the topics available on the TEKI platform.

All the desired features addressed by teachers were considered essential, thus should be implemented within the future development of the platform.

On the scale between 1 to 5, the TEKI platform received the maximum possible score from the teachers – 5 out of 5.

6.4.2 Students

All the students participating in the evaluation process were asked for the feedback. In total, 34 out of 68 students responded to the feedback survey. Most of the responding students (91.2%) liked practicing and consolidating knowledge by means of the TEKI platform. A slightly smaller number of the respondents, 26 out of 34 (76.5%), stated that they would like to use TEKI also in the future. On the other hand, the reasons for disliking the platform included the following ones:

- *"Don't know"*;
- *"I don't like performing exercises, however, the platform looked normal"*;
- The user interface was incomprehensible or confusing, for example, it was not intuitive how many zeros were expected in a columnar calculation in case of a multiplication with zero.

Despite of that, all but one respondent (97.1%) agreed that the platform was easy to use. In addition, 32.4% of the respondents admitted that they also practiced using TEKI without a specific request by the teacher. 38.2% of the respondents stated that they would like to see the progress by their classmates for comparison, and 26.5% would like to compete with classmates using the TEKI platform.

The feedback survey showed the ease of use of the TEKI platform. Despite that, the results showed that the TEKI platform is not yet appealing for students as most of them did not practice without the teacher's request, and neither were they interested in seeing classmates progress nor competing with them. This could be solved by analysing and implementing

gamification features, such as customizing personal character and related achievements, in the future.

On a scale from 1 to 5, the TEKI platform received from the participating students a fair average score of 4.24.

7 Future Work

The MVP of TEKI that was designed and implemented in the context of this thesis included a set of features for different topics in Estonian and mathematics. Based on the results of the initial survey for eliciting the requirements and the feedback surveys conducted among both the teachers and students, as well as the feedback received after the evaluation period, directions for future work with the TEKI platform were decided. These directions for future work are described in the rest of this chapter.

As the scope of the MVP of the platform that was designed and implemented in the context of this thesis was fixed, not all the elicited requirements could be satisfied. Because of that, one of the next steps in further development of the TEKI platform will be to implement new features for new topics for students from a wider spectrum of grades.

Despite the fixed scope of the MVP, an additional assessment feature based on tests generated online was implemented. The next step will be to enable teachers to check the results and give feedback directly through the user interface of the TEKI platform.

Today, the homework assignments and grades for students are managed through e-school environments. Therefore, it is important to integrate online education systems with each other, this way creating a seamless experience for both teachers and students. Considering this, in the future, TEKI will be integrated with eKool and Stuudium, which are two of the most popular e-school platforms used in Estonia.

In addition to new features and integration with other systems, it appeared from both teachers' and students' feedback that it is important that the students should be motivated to use the platform and perform the exercises. To enable this, gamification will be added to the platform. For example, students could gather experience points by solving exercises and these points could be spent on upgrading the avatar attached to the TEKI account. Additionally, other students would be able to see each other's avatars and the progress made by them in each subject or topic. This set of features would bring a positive impact on the motivation by virtual competitive experience.

Finally, based on the feedback, the user interface of the platform should also be improved. For example, it would be beneficial if the developers, user interface designers and teachers would cooperate and improve the user interface by adding more colours and rework the layout for even more intuitive user experience to better engage the students.

The development work will be continued by applying agile methodologies, meaning more schools, teachers and students will be involved for immediate feedback. This will help the platform to grow fast and become a popular online education platform in Estonia.

8 Conclusion

This thesis focused on the analysis of the most popular existing online learning environments used by teachers and students in Estonian schools and developing a new online learning platform based on the results. The analysis included a survey conducted among the teachers to collect the shortcomings of the existing platforms and to elicit the requirements for developing the new online platform – TEKI. Because of the situation caused by COVID-19 pandemic during the writing of this thesis, it was quite demanding to conduct the survey, elicit requirements for and gather feedback about the new platform. One of the main reasons for this was the lack of direct contact with stakeholders which made the feedback cycle and collecting the survey results slow. Fortunately, the usage of the existing online platforms, such as Google Forms and Zoom, facilitated the communication between the stakeholders.

Instead of developing the new platform from the beginning to the end following the waterfall model and only considering the initial set of functional and non-functional requirements, a group of teachers from Luunja Keskkool were involved as stakeholders in the development process. Following a combination of agile methodologies, such as Rapid Application Development and Kanban, the new platform was developed from scratch in an iterative way. The short feedback cycles together with the possibility to test each new feature as soon as possible were indispensable for designing an intuitive, easy-to-use user interface for both teachers and students. Equally importantly, configuring the continuous integration (CI), continuous development (CD), and using state-of-the-art technologies made the development process even more efficient. As the teachers were involved during the whole development process, they requested additional features to be developed. Some of the requested features, such as statistical views for different topics of mathematics and an assessment mode, were developed in addition to the initial set of requirements.

At the end of the evaluation of the platform by both teachers and students, feedback was collected using online surveys. Constructive and useful feedback about the platform being developed was received by feedback surveys, especially from the involved teachers. Moreover, the results confirmed that there is clearly a need for the new platform and additionally included some new ideas for the future work.

9 Summary

Online learning has become a mandatory part in everyday education, especially considering the current situation caused by the COVID-19 pandemic. It is important that the appropriate tools for teaching and learning are available for everyone independently of the current situation or location.

The purpose of this thesis was to analyse the existing online learning environments, map the shortcomings of these environments, and based on the obtained information, design and develop a minimum viable product (MVP) of a new online learning platform. The MVP focused on the most important subjects and topics of the students of the upper primary school, especially from grades 4 to 6, in Estonia.

Based on the overview and comparison of the existing online environments and the feedback received from the survey conducted among teachers of a number of schools in Estonia, a set of functional and non-functional requirements was described. Most of the existing platforms lacked a responsive user interface, required a lot of manual work by teachers, such as creating exercises with both right and wrong answers. In addition, most of the platforms were not available in Estonian. All these deficiencies together made using the online environments difficult and time-consuming for both teachers and students. Based on the shortcomings of the existing platforms and the requirements, a new online learning platform – TEKI – was designed and developed. Design and development of the platform included the usage of the state-of-the-art technologies, such as the architecture of microservices, and agile methodologies, such as Rapid Application Development.

The new platform with all the implemented features was evaluated by a group of teachers and students from Luunja Keskkool. Based on the feedback received in the course of the evaluation process, the new platform was deemed to be capable of avoiding the shortcomings of the existing online environments. Considering this, the TEKI platform is expected to be developed further in the future as there is clearly a need for the new e-learning environment.

10 Kokkuvõte

E-õpe on saanud vältimatuks osaks meie igapäevaelust, eriti arvestades COVID-19 pandeemia poolt põhjustatud olukorda. Seetõttu on oluline, et õppimiseks ja õpetamiseks vajalikud tööriistad oleksid kättesaadaval kõigile, sõltumata olukorrast või paigast.

Selle magistritöö eesmärk oli analüüsida juba olemasolevaid veebipõhiseid õpikeskkondi, kaardistada nende puudujäägid ning saadud andmete põhjal kavandada ja valmis arendada minimaalne töötav toode (MTT) uuest veebipõhisest õppeplatvormist. MTT keskendus kõige olulisemate õppeainetega seotud teemadele lähtudes algkooli vanema astme, täpsemalt 4. – 6. klasside, õppekavast Eestis.

Olemasolevate õpikeskkondade analüüsi, nende omavahelise võrdluse ning mitmete Eesti õpetajate hulgas läbi viidud küsitluse tulemused olid oluliseks sisendiks uue platvormi funktsionaalsete ja mittefunktsionaalsete nõuete koostamisel. Enamikul olemasolevatest platvormidest puudus seadmetundlik disain ning nende kasutamine nõudis palju lisategevusi nagu näiteks ülesannete loomine nii õigete kui ka valede vastustega. Samuti polnud enamus platvorme kättesaadaval eesti keeles. Kõik nimetatud puudused kokku tegid olemasolevate platvormide kasutamise keeruliseks ja aeganõudvaks nii õpetajatele kui ka õpilastele. Tuginedes olemasolevate platvormide puudujääkidele ja nõuetele uue platvormi jaoks, sai loodud uus platvorm – TEKI. Platvormi disainimisel ja arendamisel kasutati teadaolevaid parimaid tehnoloogiaid ning praktikaid, sealhulgas lähtuti mikroteenuste arhitektuurist ning väledast arendusmudelist nagu näiteks väledast tarkvaraarenduse metoodikast RAD (*Rapid Application Development*).

Uut platvormi hinnati Luunja Keskkooli õpetajate ja õpilaste poolt. Tagasisidest selgus, et uus õppeplatvorm lahendas olulised puudujäägid võrreldes seni olemasolevate sarnaste platvormidega. Sellest tulenevalt on tulevikus oodata TEKI-platvormi täiendusi ning edasiarendusi, sest vajadus uueks õpikeskkonnaks on selgelt olemas.

11 References

[1] Ministry of Education and Research, “The Estonian Lifelong Learning Strategy 2020,” 2014. [Online]. Available: https://www.hm.ee/sites/default/files/estonian_lifelong_strategy.pdf. [Accessed 4 April 2021].

[2] Ministry of Education and Research, “Digipöörde programm 2016-2019,” 2016. [Online]. Available: https://www.hm.ee/sites/default/files/lisa_2_digipoorde_programm_2016.pdf. [Accessed 4 April 2021].

[3] Miksike, “Miksike,” [Online]. Available: https://miksike.eu/. [Accessed 4 April 2021].

[4] Kahoot!, “Kahoot! for schools - choose plan,” [Online]. Available: https://kahoot.com/schools/plans/. [Accessed 4 April 2021].

[5] 10monkeys, “Supporting distance learning,” [Online]. Available: https://www.10monkeys.com/us/free-math-games.html. [Accessed 4 April 2021].

[6] Miksike, “Pranglimine,” [Online]. Available: https://miksike.ee/#pranglimine. [Accessed 4 April 2021].

[7] Quizizz, “Quizizz,” [Online]. Available: https://quizizz.com/teachers. [Accessed 4 April 2021].

[8] IETF, “RFC 7519,” 2015. [Online]. Available: https://tools.ietf.org/html/rfc7519. [Accessed 5 May 2021].

[9] M. Liviu, “Comparative study on software development methodologies,” *Database Systems Journal,* vol. 5, no. 3, pp. 37-56, 2014.

[10] P. Beynon-Davies, C. Carne, H. Mackay and D. Tudhope, “Rapid application development (RAD): An empirical review,” *European Journal of Information Systems,* vol. 8, no. 3, 1999.

[11] M. Hammarberg and J. Sunden, Kanban in action, Manning Publications Co., 2014, p. 69.

[12] Angular, “Introduction to the Angular Docs,” [Online]. Available: https://angular.io/docs. [Accessed 19 April 2021].

[13] Bootstrap, “Bootstrap,” [Online]. Available: https://getbootstrap.com/. [Accessed 25 April 2021].

[14] VMWare, Inc., “Spring Boot,” [Online]. Available: https://spring.io/projects/spring-boot. [Accessed 20 April 2021].

[15] MongoDB, Inc., “What is MongoDB?,” [Online]. Available: https://www.mongodb.com/what-is-mongodb. [Accessed 20 April].

[16] Keycloak, “Keycloak on Docker,” [Online]. Available: https://www.keycloak.org/getting-started/getting-started-docker. [Accessed 13 April 2021].

[17] Docker, “Install Docker Engine,” [Online]. Available: https://docs.docker.com/engine/install/. [Accessed 13 April 2021].

[18] nginx, “nginx,” [Online]. Available: http://nginx.org/en/. [Accessed 20 April 2021].

[19] webpack.js, “DevServer Proxy,” [Online]. Available: https://webpack.js.org/configuration/dev-server/#devserverproxy. [Accessed 20 April 2021].

[20] Gitlab, “Gitlab,” [Online]. Available: https://about.gitlab.com/. [Accessed 21 April 2021].

[21] DigitalOcean, “Products,” [Online]. Available: https://www.digitalocean.com/products/. [Accessed 12 May 2021].

Appendix

I. Glossary

Upper primary students (Estonian *II (teise) kooliastme õpilased*) describes students studying in grades 4 to 6 in Estonian schools.

Columnar calculation (Estonian *kirjalik arvutamine*) – a calculation method where digits of both numbers are calculated separately.

Mental calculation (Estonian *peastarvutamine*) – a method of calculation where person is expected to do all calculations by heart without using any help.

Plosives (Estonian *sulghäälikud*) – used to describe specific Estonian consonants *k*, *p*, *t*, *g*, *b* and *d.*

Responsive design (Estonian *seadmetundlik disain*) – a web design approach to render web pages well on a variety of devices and screen sizes.

MVP (Estonian *minimaalne töötav toode*) – an abbreviation for minimum viable product.

II. Survey: Shortcomings in online learning for upper primary students

Questions:

1) Name of the school
2) What subject requires independent work by students the most?
3) Which mathematics related topics are the most challenging for students?
4) Which Estonian related topics are the most challenging for students?
5) Do you use any online learning environment(s) for teaching students?
6) If yes, then which ones?
7) What are the shortcomings of the existing platforms?
8) Would you like to have an online environment that could address these problems?
9) Should the online learning platforms be mobile-friendly?
10) Which types of exercises do you expect from online learning environments – automatically generated and verified, created by teachers, or both?
11) How do you motivate students for learning?
12) Do you think that implementing any kind of gamification into the online learning platform would motivate children?
13) Would you be interested in evaluating a new platform for online learning together with your students?
14) Please add any thoughts that could help improving the students' motivation and online learning.

Link to the survey results (tab *TEKI: Shortcomings in Online Learning Survey Responses*):

- https://docs.google.com/spreadsheets/d/1ECsjXHtiOp6xuJvUAkrnNmOA3McNhqOABw-Zae0nuxTY/edit?usp=sharing

III. Gitlab CI configuration sample file for TEKI backend

URL: <repository_root>/education-api/gitlab-ci.yml

```
variables:
  DOCKER_REGISTRY: registry.gitlab.com/education-web
  APP_NAME: education-api
  DOCKER_HOST: tcp://docker:2375
  TAG_LATEST: $CI_REGISTRY_IMAGE/$CI_COMMIT_REF_NAME:latest
  TAG_COMMIT: $CI_REGISTRY_IMAGE/$CI_COMMIT_REF_NAME:$CI_COMMIT_SHORT_SHA

cache:
  paths:
    - .gradle/wrapper
    - .gradle/caches

stages:
  - build
  - publish
  - deploy

before_script:
  - echo $CI_REGISTRY_USER
  - echo $CI_REGISTRY

build:
  stage: build
  image: amd64/gradle:6.8.3-jdk15
  script:
    - ./gradlew assemble
  artifacts:
    paths:
      - build/libs/*.jar
    expire_in: 1 week
  only:
    - master

publish:
  stage: publish
  image: docker:19.03.12
  services:
    - docker:19.03.12-dind
  before_script:
    - docker login -u $CI_REGISTRY_USER -p $CI_REGISTRY_PASSWORD $CI_REGISTRY
  script:
    - docker build --cache-from $TAG_LATEST --tag $TAG_COMMIT --tag $TAG_LATEST .
    - docker push $TAG_COMMIT
    - docker push $TAG_LATEST

deploy:
  image: alpine:latest
  stage: deploy
  tags:
    - deployment
  script:
    - chmod og= $ID_RSA
    - apk update && apk add openssh-client
    - ssh -i $ID_RSA -o StrictHostKeyChecking=no $SERVER_USER@$SERVER_IP "docker login -u
$CI_REGISTRY_USER -p $CI_BUILD_TOKEN $CI_REGISTRY"
    - ssh -i $ID_RSA -o StrictHostKeyChecking=no $SERVER_USER@$SERVER_IP "docker pull
$TAG_COMMIT"
    - ssh -i $ID_RSA -o StrictHostKeyChecking=no $SERVER_USER@$SERVER_IP "docker container
rm -f $APP_NAME || true"
    - ssh -i $ID_RSA -o StrictHostKeyChecking=no $SERVER_USER@$SERVER_IP "docker run -d -p
8080:8080 --env SPRING_PROFILES_ACTIVE=docker --env TEKI_ADMIN_CLIENT_SECRET=$TEKI_AD-
MIN_CLIENT_SECRET --net=education-backend --name $APP_NAME $TAG_COMMIT"
  environment:
    name: production
    url: http://$SERVER_IP
  only:
    - master
```

IV. Nginx configuration sample file for TEKI platform

URL: <repository_root>/education-client/nginx.conf

```
events{}
http {
    include /etc/nginx/mime.types;
    server {
        listen 80;
        listen [::]:80;
        server_name teki.ee;
        location ~ /.well-known/acme-challenge {
            allow all;
            root /var/www/certbot;
        }
        location / {
            rewrite ^ https://$host$request_uri? permanent;
        }
    }

    server {
        listen 443 ssl;
        listen [::]:443 ssl;

        server_name teki.ee;
        server_tokens off;
        ## SSL start
        ssl_certificate /etc/letsencrypt/live/teki.ee/fullchain.pem;
        ssl_certificate_key /etc/letsencrypt/live/teki.ee/privkey.pem;
        ssl_buffer_size 8k;
        ssl_dhparam /etc/ssl/certs/dhparam-2048.pem;
        ssl_protocols TLSv1.2 TLSv1.1 TLSv1;
        ssl_prefer_server_ciphers on;
        ssl_ciphers ECDH+AESGCM:ECDH+AES256:ECDH+AES128:DH+3DES:!ADH:!AECDH:!MD5;
        ssl_ecdh_curve secp384r1;
        ssl_session_tickets off;
        ssl_stapling on;
        ssl_stapling_verify on;
        resolver 8.8.8.8;
        ## SSL end

        root /usr/share/nginx/html;
        index index.html;

        proxy_buffers         8 16k;  # Buffer pool = 8 buffers of 16k
        proxy_buffer_size     16k;    # 16k of buffers from pool used for headers

        location / {
            try_files $uri $uri/ /index.html;
        }

        location /api {
            proxy_pass http://education-api:8080;
            add_header X-Frame-Options "SAMEORIGIN" always;
            add_header X-XSS-Protection "1; mode=block" always;
            add_header X-Content-Type-Options "nosniff" always;
            add_header Referrer-Policy "no-referrer-when-downgrade" always;
            add_header Content-Security-Policy "default-src * data: 'unsafe-eval'
'unsafe-inline'" always;
            proxy_set_header Host $http_host;
            proxy_set_header X-Real-IP $remote_addr;
            proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        }

        location /auth {
            proxy_pass http://keycloak:8080;
            proxy_set_header Host $http_host;
            proxy_set_header X-Real-IP $remote_addr;
            proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
            proxy_set_header X-Forwarded-Proto $scheme;
        }
    }
}
```

V. Teacher's Feedback Survey

Questions:

15) Was the TEKI platform easy to use?
16) What did you like about the TEKI platform?
17) What did you not like about the TEKI platform?
18) Did the automatic exercise generation related to mathematics topics reduce the workload?
19) If no, why?
20) How important it is for you to see the statistical results of students' progress per exercise?
21) Would you consider using TEKI in the future?
22) If yes, then what would be the next new feature you would like to try out using TEKI?
23) Did TEKI solve any of the shortcomings of existing platforms?
24) If yes, then for which exactly?
25) Should TEKI be integrated with any of the e-school systems for marking student's grades directly, for example?
26) What score, from one to five, would you give for TEKI platform?

Link to the survey results (tab *TEKI: Teachers' Feedback Survey Responses*):

- https://docs.google.com/spreadsheets/d/1ECsjXHtiOp6xuJvUAkrnNmOA3McNhqOABw-Zae0nuxTY/edit?usp=sharing

VI. Students' Feedback Survey

Questions:

1) Did you enjoy solving exercises using TEKI platform?
2) If no, why?
3) Was the TEKI platform easy to use?
4) If no, why?
5) Would you consider solving exercises using TEKI in the future?
6) Did you practice anything on TEKI without the teacher's request?
7) Would you like to see how many exercises have your classmates solved using TEKI?
8) Would you like to compete in exercise solving together with your classmates?
9) What score, from one to five, would you give for TEKI platform?

Link to the survey results (tab *TEKI: Students' Feedback Survey Responses*):

- https://docs.google.com/spreadsheets/d/1ECsjXHtiOp6xuJvUAkrnNmOA3McNhqOABw-Zae0nuxTY/edit?usp=sharing

VII. License

Non-exclusive licence to reproduce thesis and make thesis public

I, **Magnar Aruoja**,

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to

reproduce, for the purpose of preservation, including for adding to the DSpace digital archives until the expiry of the term of copyright,

TEKI: an Online Learning Platform for Upper Primary Students,

supervised by **Kuldar Taveter**.

2. I grant the University of Tartu a permit to make the work specified in p. 1 available to the public via the web environment of the University of Tartu, including via the DSpace digital archives, under the Creative Commons licence CC BY NC ND 3.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work until the expiry of the term of copyright.

3. I am aware of the fact that the author retains the rights specified in p. 1 and 2.

4. I certify that granting the non-exclusive licence does not infringe other persons' intellectual property rights or rights arising from the personal data protection legislation.

Magnar Aruoja
14/05/2021