The UX of AI-Augmented Question Generation

Hanna Tydén & Inger Gundersen

DEPARTMENT OF DESIGN SCIENCES | FACULTY OF ENGINEERING LTH
LUND UNIVERSITY 2020

MASTER THESIS

Sana Labs

The UX of AI-Augmented Question Generation

Hanna Tydén & Inger Gundersen

LUND
UNIVERSITY

The UX of AI-Augmented Question Generation

Published by
Department of Design Sciences
Faculty of Engineering LTH, Lund University
P.O. Box 118, SE-221 00 Lund, Sweden

Subject:	Interaction Design (MAMM01)
Supervisor:	Günter Alce
Co-supervisor:	Viktor Qvarfordt at Sana Labs AB
Examiner:	Joakim Eriksson

Abstract

The demand for improved digital learning tools has increased over the recent years, and got an additional boost from the Covid-19-pandemic. Many companies use artificial intelligence to improve their systems, one use case being question generation. However, artificial intelligence technology has advanced so quickly that other research fields such as user experience has not been able to keep up.

This thesis - which was done in collaboration with Sana Labs AB - therefore aims to develop and implement a user experience in the form of a tool that can generate questions. This was carried out through a design process in four phases. Initially, research was conducted in order to discover the needs of potential users of a question generation feature. Thereafter, prototypes were developed in an iterative process, and then the final prototype was implemented in code. Lastly, the implementation was evaluated through usability tests and resulted in a tool for question generation that was well received by the participants.

Keywords: Interaction Design, User Experience, Question Generation, Usability Testing, E-learning

Sammanfattning

Efterfrågan av bättre digitala läroverktyg har ökat de senaste åren, och fick en ytterligare höjning av Covid-19-pandemin. Många företag använder artificiell intelligens för att förbättra sina system, där ett användarfall är frågegenerering. Teknologin inom artificiell intelligens har dock avancerat så snabbt att andra forskningsområden inte har hunnit ikapp.

Målet med denna uppsats - vilken är genomförd i sammarbete med Sana Labs AB - är därför att utveckla och implementera en användarupplevelse i form av ett verktyg som kan generera frågor. Detta gjordes genom en designprocess i fyra faser. Till att börja med studerades vilka möjliga behov som potentiella användare hade, som ett frågegenereringsverktyg skulle kunna uppfylla. Därefter skapades prototyper i en iterativ process, där den slutgiltiga versionen implementerades i kod. Till sist utvärderades den implementerade versionen genom användbarhetstester vilket resulterade i ett verktyg som var väl uppskattat av deltagarna.

Nyckelord: Interaktionsdesign, Användarupplevelse, Frågegenerering, Användbarhetstestning, E-lärande

Acknowledgements

We would like to thank Günter Alce, our supervisor at the Department of Design Sciences for your continuous support, valuable feedback and cheerful attitude.

Thank you also to the whole team at Sana Labs for welcoming us and facilitating the project. A special thanks to our supervisor there, Viktor Qvarfordt for always making time for us and guiding us along the process. Oscar Täckström and Christopher Nilsson from the Sana-team also deserve a special mention for their valuable input.

Lastly our gratitude goes out to everyone who participated in our interviews and usability tests, your input was highly appreciated.

Stockholm, December 2020
Hanna Tydén & Inger Gundersen

Contents

Chapter 1
Introduction

In this chapter, the background research of online learning, learning management systems and AI in combination with these is presented. This is followed by a presentation of Sana Labs AB as well as the question generation project that this thesis will be based on. Furthermore, the purpose and aim is presented followed by related work as well as the limitations of this thesis.

1.1 Background

Online learning - or *e-learning* - is an area that has undergone great improvements over recent years. As a consequence, many learning management systems (LMS) have emerged on the market, and they have become trusted and valuable tools for learners and teachers worldwide. A learning management system can be used for a variety of activities, including administrating and delivering courses and training material online. They provide great flexibility; anyone with Internet access can now get an education. The Covid-19-pandemic further strengthened the position of e-learning in our societies as schools and education institutes migrated parts or all of their on-site learning to online systems [39]. However, as teachers and lecturers are replaced with screens, the learning becomes less personal. One way to maintain the human touch and accommodate learners on a personal level is to make the systems *adaptive/intelligent*. This means that the systems can among other things provide personalized learning paths for each student and give feedback accordingly [10]. This benefit not only the students who will feel understood and supported [18], but such systems can also give teachers and course administrators better reporting- and tracking tools [10] which in turn helps them save time and further improve their courses. Many of these intelligent LMSs are powered by Artificial Intelligence (AI)-technology which gives them exciting prospects for the future as the field of AI is rapidly evolving. A concrete example of where AI can be applied in an LMS is in the area of content creation : it can help course administrators by automatically producing test/quiz questions and answers, thus greatly reducing the time they spend on preparing questions.

A downside to the fast pace of AI-development is that other fields have not had time to adjust. For example, User Experience (UX)-designers are facing new challenges as they have to adapt to these new technologies in their work [7]. This thesis will therefore explore how to create the best user experience when implementing AI-technology to support question generation in an online learning management system.

1.1.1 Sana Labs

This master thesis is done in collaboration with Sana Labs AB. Sana Labs is a Stockholm-based company working to personalize education by applying state-of-the-art AI techniques [41]. The company was founded in 2016 and has at the time of writing about 20 employees. Their product, the Sana platform, is a platform for personalized learning using AI. It identifies where skill gaps exist, and what learning is needed for the user to reach mastery within that area.

1.1.2 Question Generation Project

The underlying project for this master thesis was proposed by the Sana Labs team and is part of a bigger project to create AI-augmented content. The aim of the project is to improve upon the already existing Sana Platform, which is a platform developed by Sana Labs for personalized learning. The target audience of the platform consists of companies, institutions and other corporations who use the platform for onboarding and upskilling their employees. The project will focus on the admin-side of the platform, which is where courses are created. The authors and course administrators - henceforth called *users* - who use this part of the platform often have learning material in the form of texts without associated questions. The possibility of generating associated questions with the help of a context-aware language model would give the users a better user experience as they do not have to manually write appropriate questions. To further enhance the experience, suitable answer alternatives should also be generated along with the questions. The purpose of the project is hence to create the user interface (UI) for the feature where the users can automatically generate questions and answers matching the content of a given text using underlying AI technology.

1.2 Purpose & Research Questions

The aim of this thesis is to design and implement a tool that both solves the issue of AI-augmented question generation for the English language, and that is possible to integrate into the current Sana platform.

These are the research questions that will be answered throughout this thesis project :

- Which user needs should be fulfilled through the use of a question generation tool?
- Which UX principles are important when designing a solution for AI-augmented question generation?
- How should this type of solution be developed and implemented to be compatible with the current design of the Sana platform while still complying with UX best practices?

1.3 Related Work

There is not conducted much work specifically on the UX aspect of question generation in combination with AI. However there are many articles and surveys on what characterizes a good user experience in learning platforms as a whole, and also AI/Machine Learning (ML) in combination with UX design which will be described in the following subsections.

1.3.1 UX in learning platforms

Zaharias and Pappas [46] carried out a study of LMSs with focus on UX where 446 users participated. This resulted in a new UX-oriented evaluation model for LMS platforms with four different dimensions: pragmatic quality, authentic learning, motivation and engagement, and autonomy and relatedness.

1.3.2 ML in UX Design

A recent study shows that UX designers are not accustomed to working with AI/ML and are therefore facing challenges when trying to incorporate the technology into their designs [7]. Efforts have therefore been made to pinpoint challenges and help designers release the potential and applications of the technology. Yang et al. [45] interviewed 13 designers who had worked for several years with designing the UX of ML-enhanced products and services. Their investigations resulted in three findings:

Firstly the subjects revealed that they had limited ML understanding and they did not believe that acquiring more knowledge would make them better designers. Instead they reason about ML in abstractions and exemplars, and they work continuously to expand on this collection in order to gain a better understanding of the capabilities of the technology.

Secondly the designers deemed that they are the most successful when they cooperate actively with data scientists to help envision what to make. The interviews uncovered that working with ML usually took longer than traditional projects, and the participants identified three phases of the ML design process:

1. **Concept Development**, where designers and data scientists decide on a worthy design goal.

2. **Interaction Design and Assessment** which is when the designers start to design the interactions. The subjects described that they used a combination of traditional UX methods such as user studies, sketching, and usability testing in this phase. They also had continuous negotiations with scientists throughout the process.

3. **Release and Continued Refinement**, where the product is further assessed, tested and refined.

All the subjects said that they went through phase two, whereas phase one only happened at larger, AI-focused companies. Lastly they said that only a small selection of projects go into phase three.

The last finding presented in the article is that the best results were achieved when the designers embraced the data-driven culture and used the data proactively in their design practice. How they did this specifically was to 1) use new techniques to gather telemetry data, which is data remotely collected from current products and services, 2) use and create data visualizations in order to understand their data and 3) make sure that data scientists and domain experts share the role as a user advocate so that this does not fall solely on the designer as it often does in traditional projects.

1.3.3 Guidelines for the UX of AI

Google has its own team dedicated to researching machine learning and AI related design called People + AI Research (PAIR) [16]. Their collective knowledge is presented in various articles and videos. One particularly useful resource for UX designers is the People + AI Guidebook [15], which gives recommendations on how to follow a human-centered approach to AI. The tips are based on data and observations from over a hundred people in the Google corporation, industry experts, and research. The guidebook is divided into six chapters which can be used as a reference.

Another useful resource when working with AI in UX design, is the guide *The UX of AI* [31]. It presents core principles and provides further reading. The principles are summarized below:

- **Start with the user** - think about the user experience and the user needs before you start thinking about the underlying algorithms.
- **Set the right expectations** - be clear about what your AI can do, and express its limitations as well.
- **Explain the results** - the user should be able to track the result back to the original data, and if that is not possible an explanation of the algorithm should be present in the user flow.
- **Communicate your confidence** - in order for the users to trust the AI they have to know about the quality of it. The confidence of each result should therefore be stated.
- **Degrade gracefully** - design for several outcomes so that results can be presented according to their confidence. Less confident results can for example be presented in a more toned down user interface.
- **Know what not to automate** - not all tasks are appropriate for AI, such as tasks that require uniquely human abilities.
- **Keep the user in control** - the AI should work with, and not for the user. This means that the user should ideally have the power to intervene, reverse and reward actions and provide feedback.
- **Build trust over time** - start slow and give the user time to get to know the AI. Start for example by making suggestions instead of decisions and then gradually automate more.
- **Help your users grow** - make sure that your AI can adapt to new behaviours, so that it does not get stuck on what it has learned in the past.

- **Balance predictability and serendipity** - AI that is used for personalization takes on the user's bias. This is great when predictability is needed, but it will limit the curiosity. A mix of both is the most desirable.
- **Escape the personality cult** - the AI should match the rest of the product, hence it should be designed with the brand values in mind instead of trying to give it another personality.
- **Forget chatbots** - chatbots can create uncertainty which in turn can frustrate and alienate the users.
- **Prototype with real data and fake AI** - for early prototypes it is better to work with real user data, so that the AI gets built off of the correct basis.
- **Work with everyone** - work with a diverse team when building AI so that different experiences and values can help shape it.
- **Share your process and intentions** - be transparent about how the user's data is gathered, stored and used, as well as any design and implementation choices that were made.
- **Avoid collecting user data** - do not collect user data unless you need to, and ask permission if you do.

1.4 Delimitations

As this is a master thesis in interaction design, the implementation of the underlying AI technology and algorithms is not included in the scope. An open source application programming interface (API) was therefore used to generate questions. This API only supports the English language at the time of writing, hence our solution only supports English as well. Furthermore, the Sana platform does not currently focus on accessibility, so our solution will not do so either. Lastly we decided to only implement support for one type of question, namely "choose the correct alternative", opposed to "select all that apply". These are the two question types that are currently supported by the Sana Platform, and we chose the first type partly to limit the scope of this project and also because the API was better fitted to this type.

Chapter 2
Theory & Methods

In this chapter the theory that has been used during this thesis project is presented. Since this thesis is based on interaction design theory and takes on the user experience perspective, a couple of theories on the topic are described to give the reader a deeper understanding of the subject.

2.1 User Experience

User experience (UX) is a huge part of interaction design and sums up how well a product is perceived by the users as well as the emotional effects the product may have on people. According to the international standard on ergonomics of human-system interaction from 2019 [13], UX is defined as : "User's perceptions and responses that result from the use and/or anticipated use of a product, system or service". In other words, this could be explained as the feeling or satisfaction users get when interacting with a product in all different ways possible [36].

Usability is also an important aspect of UX, which involves how easy a user interface is to use [29]. Usability is usually defined by five quality components which are :

1. *learnability* - which asks how easy is it for users to accomplish basic tasks the first time they encounter the design?
2. *efficiency* - which addresses how quickly can users perform tasks once they have learned the design?
3. *memorability* - when users return to the design after a period of not using it, how easily can they reestablish proficiency?
4. *errors* - how many errors do users make, how severe are these errors, and how easily can they recover from the errors?
5. *satisfaction* - how pleasant is it to use the design?

User Interface (UI) is all the components that make up the interactive system and gives the user the possibility to achieve their goals. It is important that the UI is based on appropriate design so that a good UX can be achieved [24].

2.2 User-centered design

In Norman's book *The Design of Everyday Things* [34], the importance of user-centered design is presented. It is explained as a philosophy based on the users needs and interests. Hence, to be able to maintain this philosophy there are three main principles to follow, which are: early focus on the users, empirical measurements and an iterative design process [17]. These three principles are obtained through continuous user testing and research, and by doing so, finding faults and correcting them as the process progresses.

2.3 Nielsen's 10 usability heuristics

To be able to create a good user interface design, Nielsen has concluded 10 general principles to follow [26]. They are used as a broad rule of thumb, and are helpful to get a deeper understanding of a system and hence identifying possible usability problems in a user interface. The 10 principles are:

1. **Visibility of system status**
   It is important that the system always provides needed information and appropriate feedback for its users within a reasonable time so that the user gains trust in the brand/the responsible company. This principle is about creating a good customer relation and trust through good communication.

2. **Match between system and the real world**
   The system should use a language that is familiar to the user and also use real-world conventions so that the user instinctively knows what to do and what way to operate the system, rather than using system-oriented terms.

3. **User control and freedom**
   This principle is about letting the user be able to undo and redo if necessary. The system should support that there can be made mistakes by a user unintentionally, and it should not be made complex to undo such a mistake.

4. **Consistency and standards**
   The users should not have to wonder whether different actions, words or such mean the same thing. To avoid this, the system should therefore follow platform conventions.

5. **Error prevention**
   The system should support some sort of error prevention such as error messages. Even better is if the system is designed in a way that prevents errors from occurring in the first place.

6. **Recognition rather than recall**
   It should be easy for the user to recognize what should be done and not having to recall what was shown or presented on an earlier page. This can be achieved through having instructions visible or at least easily retrievable whenever appropriate.

7. **Flexibility and efficiency of use**
   The system should be able to cater to both inexperienced and experienced users through having accelerators that may speed up the interaction for the expert user if necessary.

8. **Aesthetic and minimalist design**
   There should not be any unnecessary information or too much text presented at any time. This may cause confusion for the user and diminishes the relevant information's relative visibility.

9. **Help users recognize, diagnose, and recover from errors**
   When errors occur there should be presented a clear explanation in plain text understandable for the user as well as a suggested constructive solution.

10. **Help and documentation**
    The system should not need any documentation in order to be used, but nevertheless it can be helpful for the user to provide some information about how to perform different tasks or other helpful information in different scenarios. Any such information should be easy to search for and not be too large and tedious.

2.4 System Usability Scale

A System Usability Scale (SUS) is a quick and relatively easy way of gathering data surrounding usability of a system or product [4]. It consists of 10 standardized statements with options ranging from "I Strongly Disagree" to "I Strongly Agree" on a scale of one to five respectively. The 10 statements that are presented to the user are the following:

1. I think that I would like to use this system frequently.
2. I found the system unnecessarily complex.
3. I thought the system was easy to use.
4. I think that I would need the support of a technical person to be able to use this system.
5. I found the various functions in this system were well integrated.
6. I thought there was too much inconsistency in this system.
7. I would imagine that most people would learn to use this system very quickly.
8. I found the system very cumbersome to use.
9. I felt very confident using the system.
10. I needed to learn a lot of things before I could get going with this system.

2.4.1 Calculation of SUS Score

In order to calculate the overall SUS score of a system, each respondent's answers should be calculated according to the following rules[3]:

- For questions 1, 3, 5, 7 and 9: take the scale position and subtract 1.
- For questions 2, 4, 6, 8 and 10: use 5 and subtract the scale position.

Thereafter, the sum of the values found shall be multiplied by 2.5 in order to get the overall value of system usability (SU) according to that specific respondent. The final value should be a number between 0 to 100 and it is important to know that it is not a percentage scale. Lastly, the mean of all the SUS scores should be calculated in order to know what the average SU of the system corresponds to.

In order to interpret the final SUS score in a simpler way, Bangor et al. added an adjective rating scale to the existing SUS [2]. The scale is presented in figure 2.1. According to Sauro, based on an average from 500 studies, a SUS score above 68 is more than average and therefore considered good SU [42].

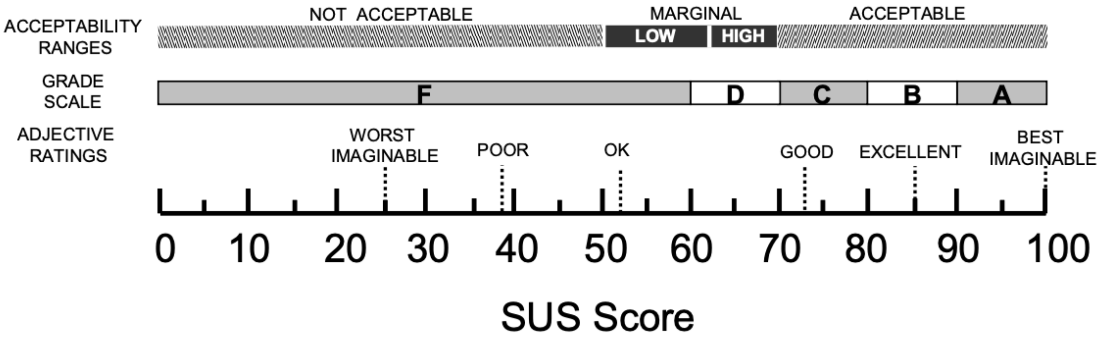


Figure 2.1: The comparison of the acceptability scores, school grading scales and adjective ratings, in relation to the average SUS score [2].

2.5 Prototyping

Prototyping is a way of showcasing how the final product, or a part of it, will function. The goal of the prototype is to get an idea of how the product or solution will be interacted with, without having the finalized product ready [1]. This is often considered a good way to test the functionality and by doing so realizing what has to be redone or changed before the final product is produced. This saves both time and money since many problems can be avoided before too much effort is put into building the finalized solution.

There are two types of prototypes: low-fidelity (Lo-Fi) prototypes and high-fidelity (Hi-Fi) prototypes [23]. Lo-Fi prototypes are a quick and cheap way to test at early stages. This is usually done by drawing pictures of a process and therefore does not have any interactive functionality. Hi-Fi prototypes on the other hand are usually more complex and realistic.

They are usually clickable in some way or have other types of intractable features. Hence, Hi-Fi prototypes are often more expensive and more time consuming to create. Lo-Fi prototypes are therefore more common in early stages of a project to test and evaluate before moving on to creating more complex Hi-Fi prototypes.

2.6 Brainstorming

Brainstorming is an effective idea generating activity. It is made under controlled conditions in an environment that is free-thinking and open to all ideas possible [22]. This is the core principle of brainstorming since the goal is to produce quantity rather than quality. Before the session starts, there are some conditions that are agreed upon so that everyone stays on topic. These conditions are for example a time limit, a target problem or area and allowing for everyone to get to speak their minds.

2.7 User Interviews

User interviews are useful in a variety of situations. One of them is in the beginning of a design process, before you have a design or testable prototype, in order to create personas or user stories and feature ideas.

There are three major types of interviews: structured, semi-structured and unstructured [30]. These three types all have their own area of use. During a structured interview all questions are asked to all participants in the exact same order. The questions are also asked in a way so that the interviewee needs to chose one of the possible options. This gives more quantitative data rather than qualitative. This type of interview is often used in the last stages of the design process. The semi-structured form on the other hand, has more of an open discussion form but still has pre-written questions. This allows the interviewer to steer the discussion through follow-up questions depending on what the interviewee answers. An unstructured interview is more of a discussion around a given theme. It could also be surrounding one single open question and provides the interviewer with many new insights. This type of interview is usually used in the early stages of the design process.

2.8 Used technologies

In order to implement and integrate a question generation feature in the Sana Platform, several technologies were used and are described below.

React

React is an open-source JavaScript library developed by Facebook [9]. It is used for building fast and interactive user interfaces for web and mobile applications. React is component-based, which means that code is encapsulated into components with independent states which can be used together to create complex UIs.

TypeScript

TypeScript is an open-source programming language developed by Microsoft that builds on JavaScript [32]. This is accomplished by adding static type definitions to the language. These definitions provide not only better documentation for objects, but it can also validate that the code is working correctly.

Next.js

Next.js is an open-source React framework created by Vercel [44]. A big advantage of Next.js is that it has two options for pre-rendering; static generation and server-side rendering. This hybrid solution gives developers more freedom as they can choose the strategy that best fit the task at hand. Additionally Next.js has built-in support for both CSS and TypeScript, and a powerful routing system is included as well.

PostgreSQL

PostgreSQL is an open-source object-relational database system [43]. It dates back all the way to 1986 and the POSTGRES project at the University of California at Berkeley. It supports both SQL and JSON querying, and is used as a primary database for many web-, mobile- and analytics applications.

2.8.1 Artificial Intelligence

Artificial intelligence (AI) is a collective word for all technologies that enables computers to imitate human intelligence [35]. By using logic and mimic decision making based on large datasets, it can be used for automation and problem solving amongst many other things. Machine learning (ML) is a subset of AI that enables machines to improve at tasks based on experience. Within machine learning there is further a subset called deep learning. Deep learning deal with allowing software to train itself to perform tasks such as image recognition and speech.

OpenAI API

OpenAI API is an API for accessing new AI models developed by OpenAI [19]. The API needs a pattern to base its result on and the person using the API has to provide some examples of patterns in order for it to work properly. The API can then take any text prompt and return a text completion. It does so by attempting to match the given pattern and its success generally varies depending on how complex the task given is. The API can also train on a dataset of examples that is provided for it, or by learning from human feedback provided by end-users. The API has several settings that can be used to influence the output text. One of these settings are temperature, which determines the randomness of the generated text. If the temperature is set to zero, the process will be deterministic and will always generate the same output for a given input. If it is set to one the process will use a lot of creativity and take many risks.

2.9 The Design Process

As described by Preece et al. [36], the four basic activities of interaction design that lay as foundation for this thesis' design process are:

1. identifying needs and establishing requirements,
2. developing alternative designs that meet those requirements,
3. building interactive versions so that they can be communicated and assessed,
4. and evaluating them, i.e., measuring their acceptability.

These four activities have been customized to better fit the work process of this specific project. The first activity is an important part of interaction design. This is where we find out more about our target users, what their needs are and what kind of support the interactive product should provide for them. The second activity is the core activity of designing. This is the development phase where prototypes are made based on the previous user research. Prototypes at this stage are often Lo-Fi prototypes which can iteratively be developed into Hi-Fi prototypes. The third activity will focus on the actual implementation of the chosen design. The fourth and last activity is when the implementation will be tested and evaluated to see whether the usability and acceptability etc. reaches the requirements that the users are aiming for.

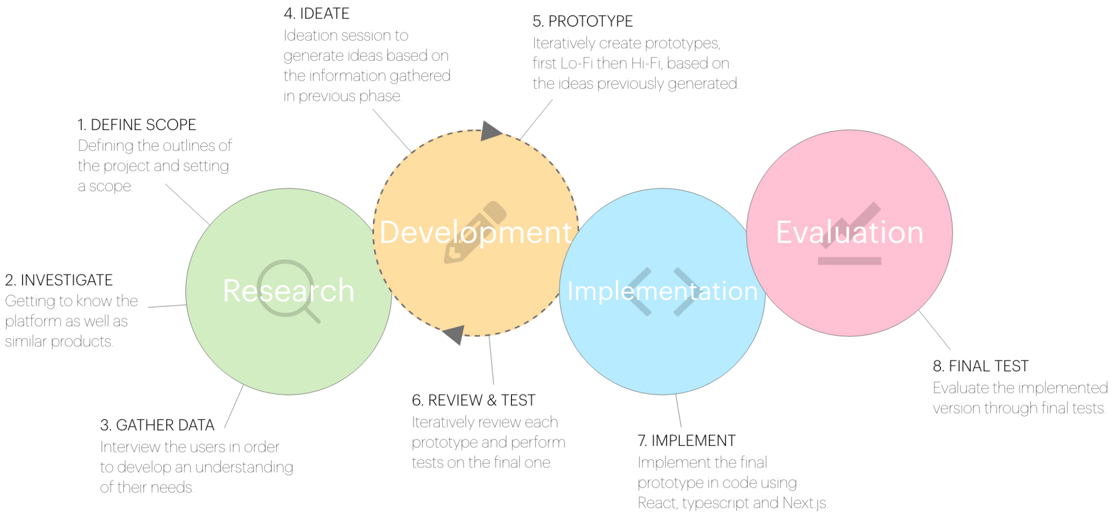

4. IDEATE
Ideation session to generate ideas based on the information gathered in previous phase.
5. PROTOTYPE
Iteratively create prototypes, first Lo-Fi then Hi-Fi, based on the ideas previously generated.
1. DEFINE SCOPE
Defining the outlines of the project and setting a scope.
Development
Evaluation
2. INVESTIGATE
Getting to know the platform as well as similar products.
Research
Implementation
8. FINAL TEST
Evaluate the implemented version through final tests.
6. REVIEW & TEST
Iteratively review each prototype and perform tests on the final one.
3. GATHER DATA
Interview the users in order to develop an understanding of their needs.
7. IMPLEMENT
Implement the final prototype in code using React, typescript and Next.js.

Figure 2.2: Our Design Process

Chapter 3

Research & Data Gathering

In this chapter, the research phase of the project is introduced. This is the first of the four activities in the design process, and presented below are the steps taken in order to getting to know the product better, gather relevant information, plan and carrying out the interviews and lastly the result of these.

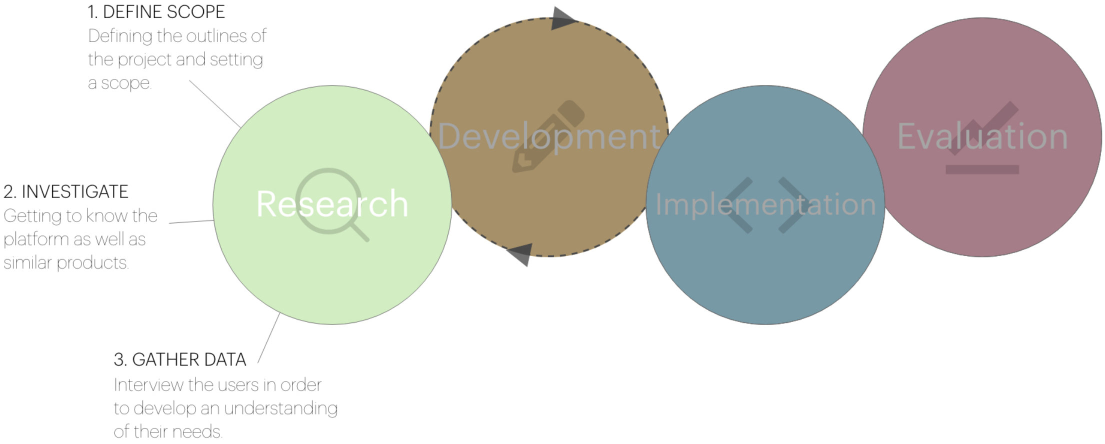

1. DEFINE SCOPE
Defining the outlines of the project and setting a scope
2. INVESTIGATE
Getting to know the platform as well as similar products
3. GATHER DATA
Interview the users in order to develop an understanding of their needs
Research
Development
Implementation
Evaluation

Figure 3.1: Our design process - the research phase

3.1 Literature study

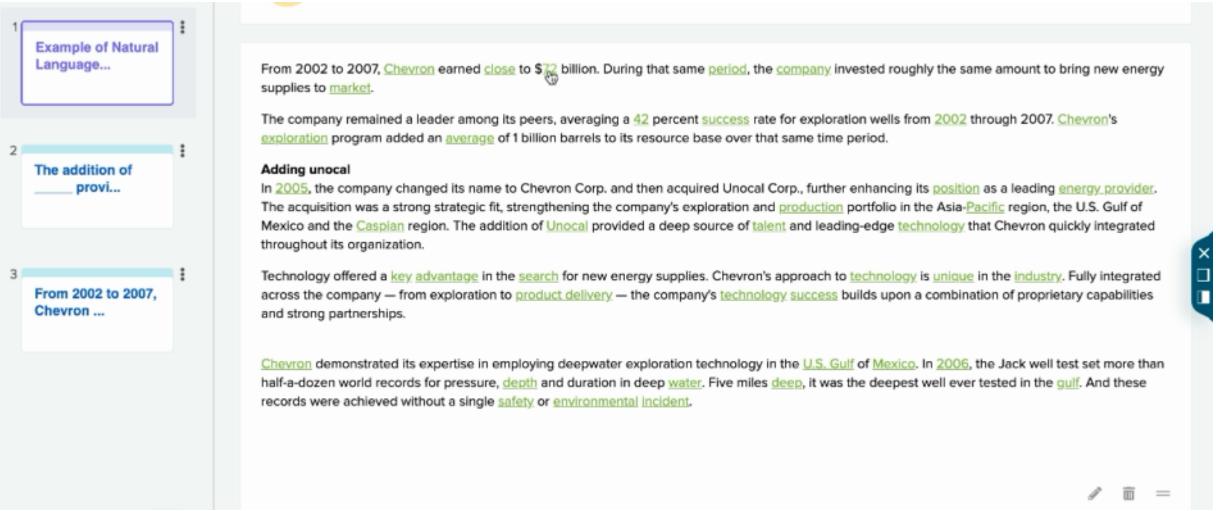
We started our research phase by conducting a literature study in order to get an overview of the work and developments that have already been done on the topic. The information that was gathered was mostly related to AI/ML in combination with UX as this is the aspect that this thesis focuses on. This part of the literature study is presented in section 1.3. Furthermore we wanted to explore other products and systems that are similar to the Sana-platform in order to get inspiration and see what they have considered when it comes to the design. The result is presented in the following subsections.

3.1.1 Similar products

Today there are a plethora of different LMSs available online. In order to limit our search, we focused on products and systems that have implemented some sort of question generation for learning purposes.

Cerego

One such product is Cerego's platform [5]. It is an online learning platform that is adaptive and uses AI to create personalized experiences for the user. It uses a Natural Language Processing AI-tool to generate questions and answers from given learning material. The tool scans a text for teachable concepts, underlines these and the user can then choose to autogenerate a corresponding question by clicking the underlined word as can be seen in figure 3.2. Answer alternatives are generated along with the question.


1
Example of Natural Language...
2
The addition of ______ provi...
3
From 2002 to 2007, Chevron ...

From 2002 to 2007, Chevron earned close to $72 billion. During that same period, the company invested roughly the same amount to bring new energy supplies to market.

The company remained a leader among its peers, averaging a 42 percent success rate for exploration wells from 2002 through 2007. Chevron's exploration program added an average of 1 billion barrels to its resource base over that same time period.

Adding unocal

In 2005, the company changed its name to Chevron Corp. and then acquired Unocal Corp., further enhancing its position as a leading energy provider. The acquisition was a strong strategic fit, strengthening the company's exploration and production portfolio in the Asia-Pacific region, the U.S. Gulf of Mexico and the Caspian region. The addition of Unocal provided a deep source of talent and leading-edge technology that Chevron quickly integrated throughout its organization.

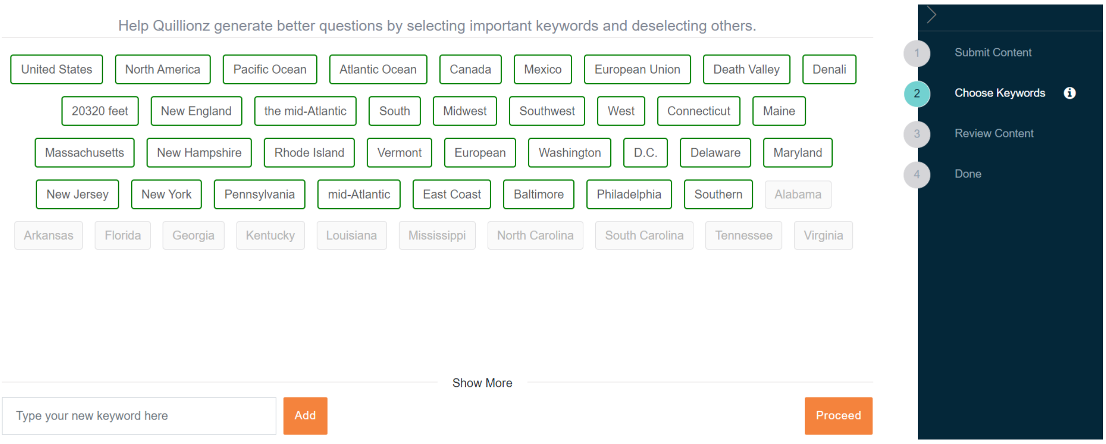
Technology offered a key advantage in the search for new energy supplies. Chevron's approach to technology is unique in the industry. Fully integrated across the company — from exploration to product delivery — the company's technology success builds upon a combination of proprietary capabilities and strong partnerships.

Chevron demonstrated its expertise in employing deepwater exploration technology in the U.S. Gulf of Mexico. In 2006, the Jack well test set more than half-a-dozen world records for pressure, depth and duration in deep water. Five miles deep, it was the deepest well ever tested in the gulf. And these records were achieved without a single safety or environmental incident.

Figure 3.2: Question generation in Cerego

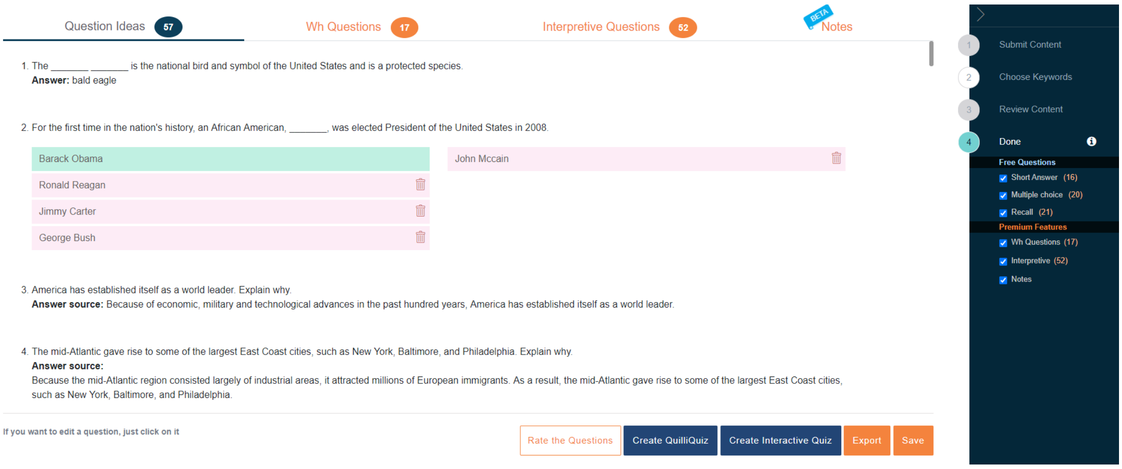
Quillionz

Quillionz is another platform for creating questions, assessments and quizzes [38]. It also supports question generation, but their approach differs slightly from that of Cerego. In the Quillionz platform it is the user itself who chooses which concepts and keywords from a given text that are important, see figure 3.3. The system then generates questions and answers that cover these keywords as seen in figure 3.4.



Help Quillionz generate better questions by selecting important keywords and deselecting others.

Figure 3.3: Question generation in Quillionz



1. The _______ _______ is the national bird and symbol of the United States and is a protected species.
   Answer: bald eagle

2. For the first time in the nation's history, an African American, _______, was elected President of the United States in 2008.

   Barack Obama
   Ronald Reagan
   Jimmy Carter
   George Bush
   John Mccain

3. America has established itself as a world leader. Explain why.
   Answer source: Because of economic, military and technological advances in the past hundred years, America has established itself as a world leader.

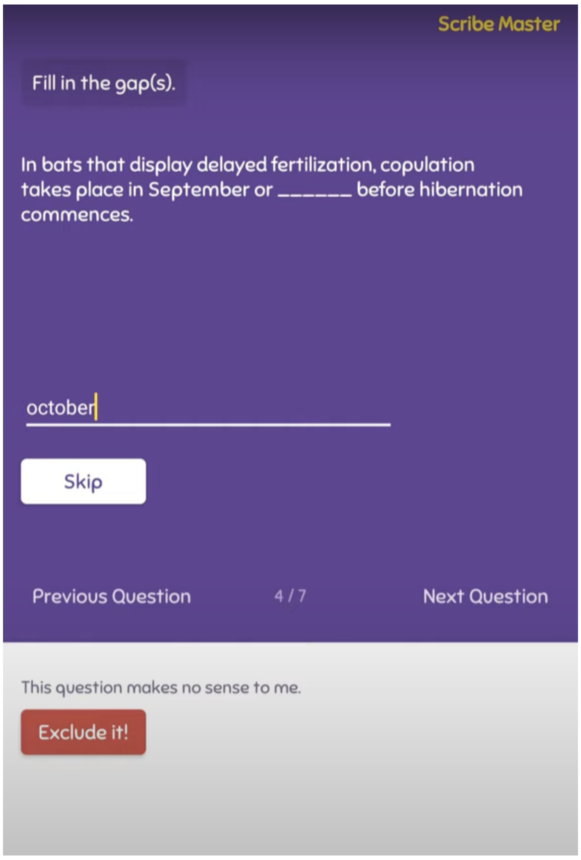
4. The mid-Atlantic gave rise to some of the largest East Coast cities, such as New York, Baltimore, and Philadelphia. Explain why.
   Answer source:
   Because the mid-Atlantic region consisted largely of industrial areas, it attracted millions of European immigrants. As a result, the mid-Atlantic gave rise to some of the largest East Coast cities, such as New York, Baltimore, and Philadelphia.

If you want to edit a question, just click on it

Figure 3.4: Question generation in Quillionz

Flexudy

Lastly, Flexudy is an education app that wants to help its users learn faster and easier by using AI [12]. The app can be used to generate quizzes from text files that you upload to the app. In Flexudy, the user has no say in which concepts are more important. Instead the app generates fill-in-the-blank-questions from the whole text. No answer alternatives are provided, and the user is supposed to guess for themselves before the right answer is revealed as seen in figure 3.5.



Fill in the gap(s).

In bats that display delayed fertilization, copulation takes place in September or ______ before hibernation commences.

october

This question makes no sense to me.

Figure 3.5: Question generation in Flexudy

Findings

The main takeaway from this research was that for automatic question generation to be useful it is not enough simply to generate a large amount of questions. The overall experience from a user perspective was better in the systems like Cerego and Quillionz where the user can influence the process by indicating in the text which concepts or sentences are especially important, since the resulting questions were more relevant. Therefore Flexudy generated more questions that did not make sense compared to the two others.

The system must also be quick and easy to use in order to keep the user satisfied. Quillionz did not do well in this aspect, as it often forces the user to perform a - usually tedious - review of the text, before the system can start generating questions.

Lastly, fill in the blank seems to be the easiest question type to generate as it was the most common type. Only Quillionz supported other types like true/false, "explain the following" and select the correct answer in addition to fill in the blank. This is an obvious advantage as it gives the content creator more freedom and flexibility. Regarding the answer(s) to the generated questions, Flexudy was the only one that did not generate answer alternatives. This is because their system is aimed mostly towards students and is supposed to work as a "flash-card"-generator. Both Cerego and Quillionz can on the other hand generate answer alternatives to their questions.

3.2 Getting to know the product

In order to get to know the product - the Sana-platform - and get a deeper understanding of how the platform works, we decided to do a heuristic evaluation using Nielsen's 10 usability heuristics described in section 2.3. This was done as a usability inspection method that helped us to identify usability problems in the UI design.

The layout of the Sana platform's authoring interface at the time of this research is presented in figures 3.6-3.8.

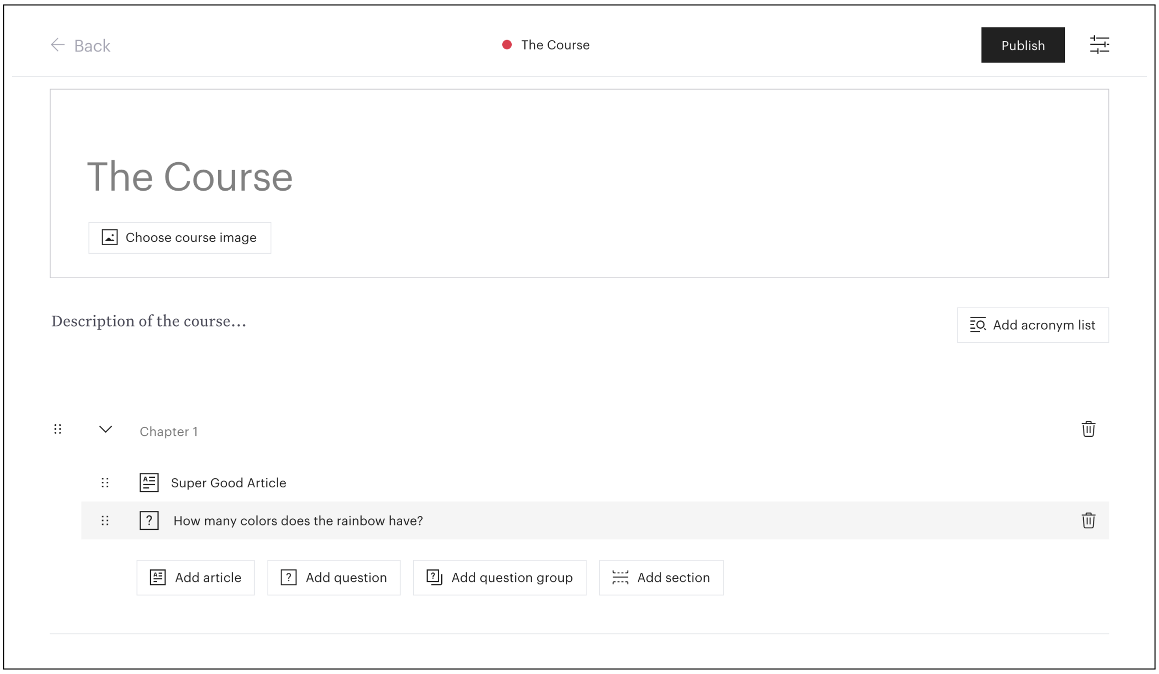
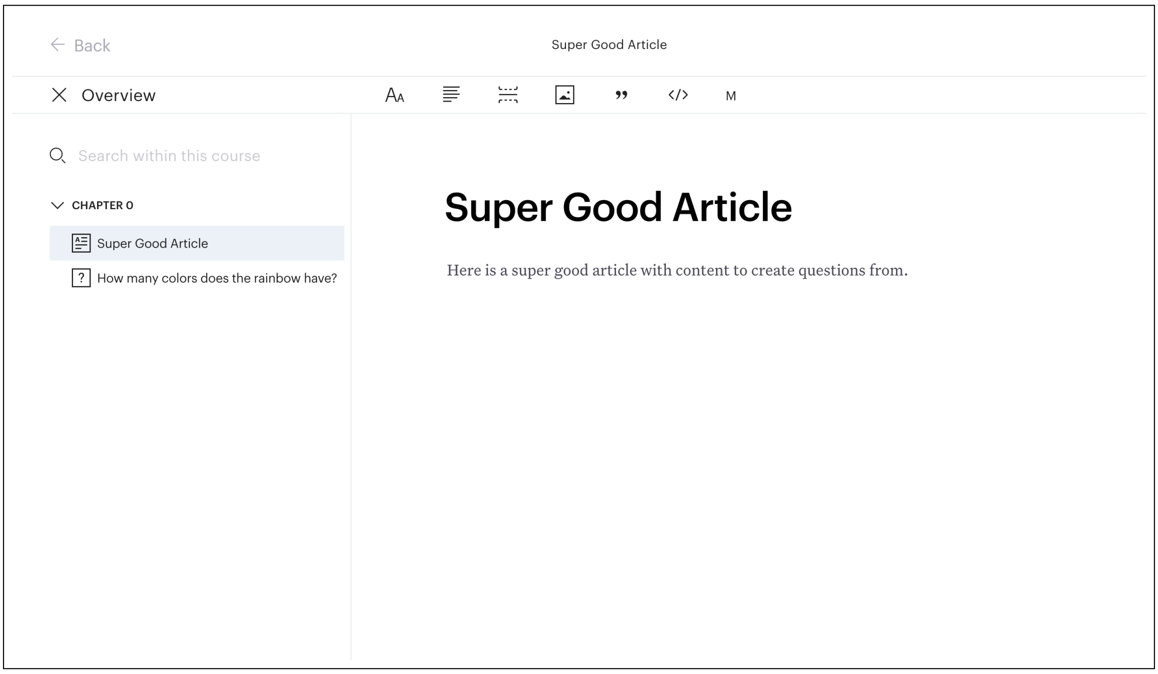


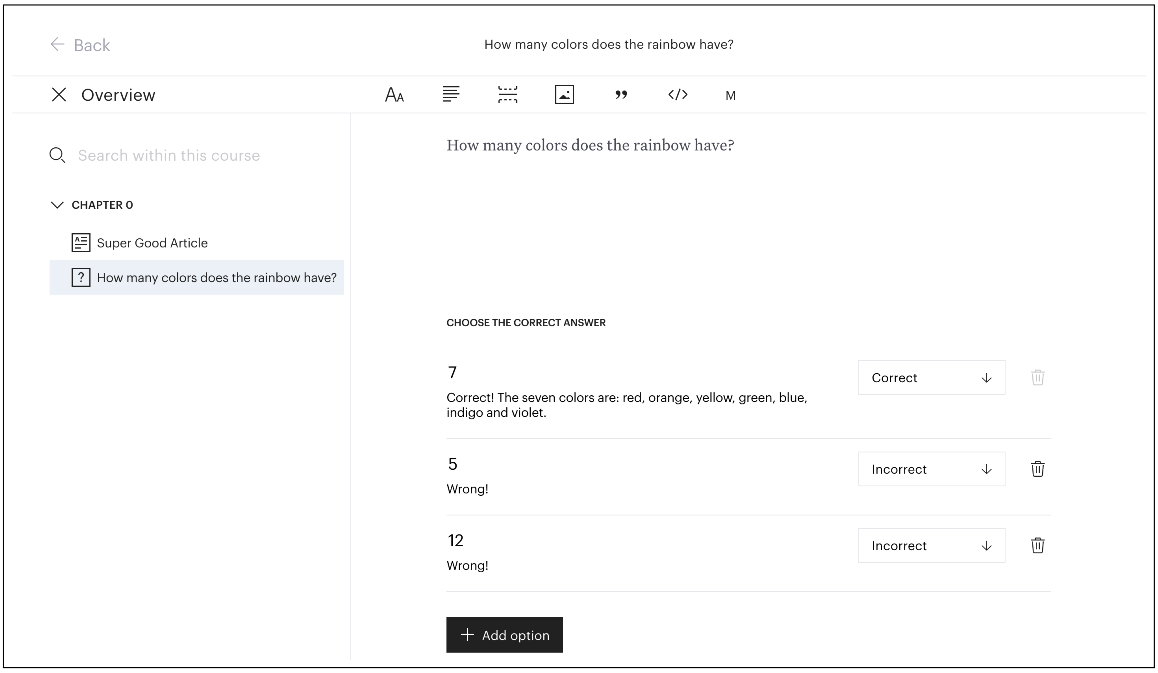
Figure 3.6: Course overview


Back
Super Good Article
Overview
AA
M
Search within this course
CHAPTER 0
Super Good Article
How many colors does the rainbow have?

Super Good Article

Here is a super good article with content to create questions from.

Figure 3.7: Article editor


Back
How many colors does the rainbow have?
Overview
AA
M
Search within this course
CHAPTER 0
Super Good Article
How many colors does the rainbow have?
Add option

How many colors does the rainbow have?

CHOOSE THE CORRECT ANSWER

7
Correct! The seven colors are: red, orange, yellow, green, blue, indigo and violet.
Correct

5
Wrong!
Incorrect

12
Wrong!
Incorrect

Figure 3.8: Question editor

3.2.1 Findings

The overall impression of the design was good as it was easy to use and understand, so it did well regarding the **aesthetic and minimalist design**-heuristic. When it comes to **visibility of system status** there is some room for improvement related to feedback. All buttons should give the user visual confirmation that they have been clicked and show that they are being hovered over. There were helpful texts in many places when hovering over icons, but these should be present everywhere so as to not contradict the **consistency and standards**-heuristic. Also, it has an auto-saving mechanism that sometimes made it unclear whether changes were actually saved, and it also made it difficult to discard changes. This could be improved upon to give the users more control and reassurance in accordance with the **visibility of system status**- and **user control and freedom**-heuristics. Lastly a tutorial of some kind could be added to better comply with the **help and documentation**-heuristic. It is however worth noting that the Sana Platform is a product that is rapidly developing and many of the things mentioned here were fixed/improved during the duration of this thesis.

3.3 Brainstorming Session

The two of us also had a brainstorming session according to the theory in section 2.6. This session was held in order to gather our thoughts and our initial assumptions regarding the feature. This was also a good way to start preparing for the interviews since it gave us a broad picture of what questions might be interesting to ask. Additionally it was useful in order to prepare use cases for the interviews since we got to better understand what user flows might be interesting. Before we started the actual brainstorming we agreed on some ground rules for the session. We decided that the topic would be the whole user flow of the feature, which is broad, but good since the goal of this session was to gather all our initial thoughts. The time was set to 10 minutes and each of us got a pen and post-it notes. After the 10 minutes, we wrote "before the question generation" and "after the question generation" on a whiteboard in order to create a timeline for the user flow. One by one we presented our post-it notes and placed these on the board wherever they fit in the most on the timeline. Lastly, these were documented and discussed.

3.3.1 Findings

During the session some main topics were brought up, which were:

- System requirements/restrictions
- The question generation interface - what components it should contain
- Marking of articles/parts of articles to generate questions from
- The displaying of generated questions
- Answer alternatives to the generated questions
- Editing of generated questions
- Integrate with the already existing option to write own questions

3.4 Interview process

Semi-structured interviews were then conducted in order to get a better understanding of the user flow in the current system and what needs or problems the users have. This was a part of the user research phase and in this section the participants' background, the procedure and the findings from the conducted interviews are presented.

3.4.1 Participants

The interviews were conducted with six people in total; four employees at Sana Labs, and two external users. All the interviewees were in some way familiar with the authoring tool which is why we chose them as participants. Out of the four employees that were interviewed, one was working with developing the authoring tool as a software engineer and the rest have created courses on behalf of external users of the system and were on a daily basis working with business development. The external interviewees had used the authoring tool for creating courses for their students. The interviewees were between 20 and 46 years old ($M = 26.8$). All the participants had proficient computer skills and were all familiar with the authoring tool on the Sana platform since it is involved in their everyday working tasks. With that said, they had only been using the tool one-five months and they had each created two-six courses with roughly two questions per article, except for one of the external interviewees who said they wrote about 30 questions depending on the article. Over all, the interviewees had used similar platforms in the form of other learning systems such as Enki [8], Moodle [20], Blackboard [21], Its Learning [25], etc. Their overall impression of the Sana platform, compared to these systems, was that the Sana platform is more restrictive and therefore easier to use and more straight forward.

3.4.2 Procedure

The interviews were conducted via Google Meet [14] and the audio was recorded throughout the whole process. During the use cases, the participants were asked to share their screens and screen recordings were then made. The "think aloud protocol" was used in order to understand what the participants thought while using the platform. This is a common method used to gather data during usability tests [28]. We took notes during the process, so the recordings were mainly used as a supplement when we were concluding our findings afterwards. The interviewees were informed of this and what the project was about, and they were asked to sign an informed consent form (see appendix A) before we started. Each interview took about 30 minutes.

Since the interviews were on a semi-structured form, the questions that were asked were a combination of open and closed questions with some follow up questions. As previously mentioned, the questions asked were based on the brainstorming session described in the previous section and aimed to uncover as many user needs and issues as possible. First, a couple of demographic questions were asked followed by these questions and use cases:

Pre-questions

- How often do you use the authoring-tool on the Sana Platform?

- How long have you been using the system?
- Roughly, how many courses have you created?

    - On average, how many questions do you write per article?
- Have you used similar products before? If yes, which ones?

    - What pros and cons did these systems have?

Use cases

- What difficulties do you see with the current system when it comes to creating questions for a course? Please click around in the platform and think out loud as you go along.
- What does the work flow usually look like when you are creating a course? Starting from clicking on “Create course” to when you are ready to publish. Please click around and think out loud as you go along.
- Describe your reasoning when creating questions from a given article, please click around and think out loud as you go along.
- Imagine a function that, when you click on a button would generate a number of questions that are linked to a given article. In the existing system, where do you feel that a good place for this button would be? Please click around and think out loud as you go along.

    - Then when you click this button, what would you want it to do?
    - How many questions should ideally be generated?

3.4.3 Findings

The interviews gave us essential information regarding the current platform as well as some problems the users face when using it. A summary of the findings from these interviews are presented in this section.

Workflows

Overall, the use cases gave great insights into how the platform is being used by both internal and external users to create courses and their contents. Most of the interviewees showed that they usually start by making a general outline for their course, i.e. adding chapters, articles and questions so that they get the items in the correct order before adding text and pictures to them. An opposing view was however also presented; that when adding a new article/question the user should automatically be redirected to the editor so that they are ready to start adding content instead of having to click on an article or question in order to get to the editor. This approach is obviously not favored by those who stated that they like to do an outline first as this would interrupt their workflow by sending them into the editor every time they add a new item.

Current issues

From the use cases some issues with the current platform became apparent. The main complaint from the internal interviewees was the inflexibility in regards to their workflow when adding new content. If the user is editing/creating an article or a question in the editor and would like to add another, they will have to navigate back to the course overview in order to do so, and then re-enter the editor. Another pain-point that was highlighted by both internal and external interviewees during the use cases is that when writing questions it would be useful to see the corresponding article/material at the same time. The user does not get any support from the platform to do this, instead some of the interviewees explained that they usually split their screen or switch between different tabs in their browser, which can be tedious. Also, some interviewees voiced wishes about being able to change the order of questions within question groups and being able to select several items at the same time to move around/delete in the overview. Lastly, one interviewee would like some sort of guide or help from the platform when creating courses, at least the first time they enter the system so that it is easier to get started.

Question content and related articles

Most of the internal interviewees pressed that they usually create and publish courses on behalf of clients. In these cases their material is already given so their work mainly consists of making the material fit into the Sana layout. This is sometimes difficult when a question in the given material is written in a way that does not fit into one of the two question types that the Sana Platform provides. In these cases the content creator will have to try to transform the question into another format. The external interviewees highlighted some other concerns regarding the relation between question and articles; as soon as you connect a question to a specific article, you create a focus on content. It is beneficial to also have some questions that test broader concepts and deeper understanding, i.e. questions that are not related to specific articles but to a topic in general. The external interviewees therefore explained that they write some more general questions as well that they do not relate to any articles. However, not relating a question to any articles will not utilize the full power of the AI-technology behind the Sana Platform as the learner will not receive feedback on which article to read up on should they need more revision.

Question generation

Four main ideas regarding where the question generation should be initiated from were gathered:

- Button/icon in the header of the article editor. This would automatically connect the generated question(s) to the article that the user was editing when clicking the button.

- Button in the course overview page next to the other "add <content>"-buttons with an explaining text akin to "Generate question(s)". Upon clicking this button the user should be prompted to choose which article(s) the question(s) should be related to before the questions are generated. It was also proposed to have this button next to each article, instead of next to the other buttons. That way the question(s) would automatically be related to the correct article.

- No new button, but the user should be presented with an extra option when clicking the "add question"- or "add question group"-buttons where the user can choose whether they want to write the question(s) themselves or if they want the system to generate the question(s).

- Questions automatically appear after articles are created, so that they become an integrated part of the work flow. The user hence does not have to explicitly ask for questions to be generated. An option to disable this feature should however exist. A related idea was to have some sort of auto completion when writing questions.

When asked to imagine what should happen after initiating the question generation, most of the interviewees imagined some sort of accept/reject-option on the generated questions. All the interviewees concluded that the user must be able to edit the generated questions in case they need adjustments. Some proposed an interface where the user can see the questions and the relevant article(s) side by side, as this would make it easier to reject/edit questions and also addresses the problem stated earlier in this section about having to jump back and forth between question and material.

We received scattered responses on the question regarding the number of questions that should be generated at a time. Many of the interviewees argued both for and against generating one at a time and generating several at a time. The main argument in favor of generating one at a time was that the user has to manually check each question anyway, so it is easier to just receive one at a time. The main argument in favor of getting several was that the user usually wants several questions per article, so by generating several right away the user gets to see some alternatives and get an overview. In many of the interviews this discussion led to the conclusion that an option to choose the number of questions to generate would be beneficial. The underlying technology should in that case scan the chosen article to see how many questions it is possible to generate for the given text. Thereafter the user can choose how many questions to generate, the upper limit being the number calculated by the system. The interviewee who stated that they used to write about 30 questions per article was especially enthusiastic about being able to choose the number of questions to be generated, and concluded that it would save a lot of time.

Chapter 4

Development

The second step of the design process - the development phase - is presented in this chapter. In this step, ideas were summarized and made into prototypes in an iterative process which is described below. Furthermore, the process of testing the final prototype in order to review and further develop our idea is presented.

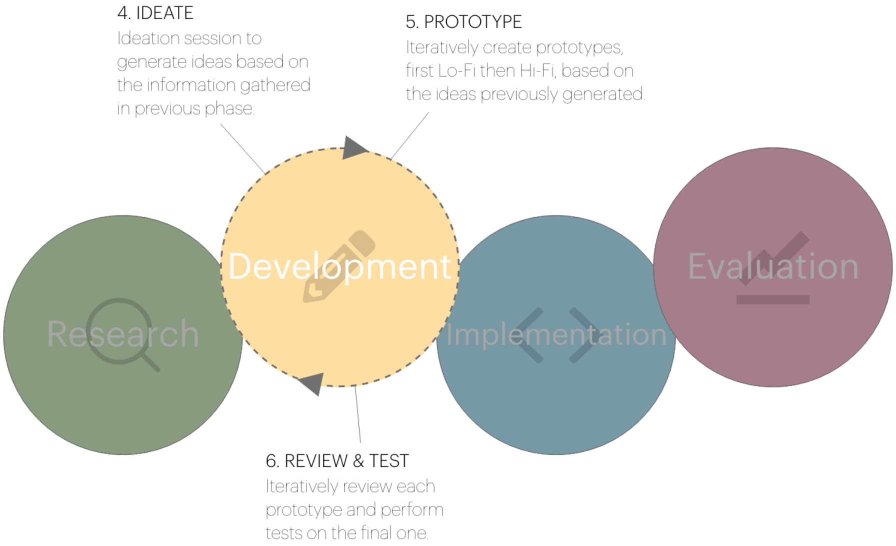

4. IDEATE
Ideation session to generate ideas based on the information gathered in previous phase
5. PROTOTYPE
Iteratively create prototypes, first Lo-Fi then Hi-Fi, based on the ideas previously generated.
Research
Development
Implementation
Evaluation
6. REVIEW & TEST
Iteratively review each prototype and perform tests on the final one.

Figure 4.1: Our design process - the development phase

4.1 Ideation session

Based on the user research that was gathered in the first phase of this project, we had an ideation session. During this session we summarized the ideas from the interviewees on how to have a question generation feature in the current platform. We also came up with some solutions on our own based on the interviewees' current problems or issues with the platform and the way it looks today. We also kept the UX of AI guidelines described in section 1.3.3 in mind when coming up with these ideas. By writing them all down we realized some advantages and disadvantages with each of them. We thoroughly went through the ideas one by one and thought of possible solutions to the potential disadvantages we could see. In the following section the ideas that we chose to move forward with are presented in more depth.

4.2 Sketching Lo-Fi

After the ideation session we moved forward with the six ideas that were the most promising based on the user research. We sketched these with pen and paper in order to quickly get physical models to work with and present during meetings with co-workers at Sana Labs. The ideas are summarized below:

- **Idea one: A new button next to the existing buttons on the course overview page**
  The button should have explanatory text, for example "Generate questions", see figure 4.2. A modal will appear once the button is clicked, see figure 4.3. In the modal the user must first choose which article(s) they want the question(s) to relate to. Thereafter they can choose the number of questions they would like to generate. We chose to use a modal since it facilitates user control, which is an important principle according to the heuristics by Nielsen listed in section 2.3. After closing the modal by clicking the "Generate"-button, the user is again presented the course overview page, but now new, generated questions have appeared, along with a small label indicating that the question has been automatically generated, see figure 4.4. The label will remain there until the user has opened the question at least once, in order to make sure that all questions are manually quality-checked before publishing a course. If the user chooses to generate more than one question, these will be generated as a question group instead.

- **Idea two: A button that appears when hovering over an article on the course overview page**
  This idea means that the user can click the button next to the article of their choice - see figure 4.5 - hence not needing to specify which article to generate questions from in the modal described in idea one. Otherwise the two ideas are identical.

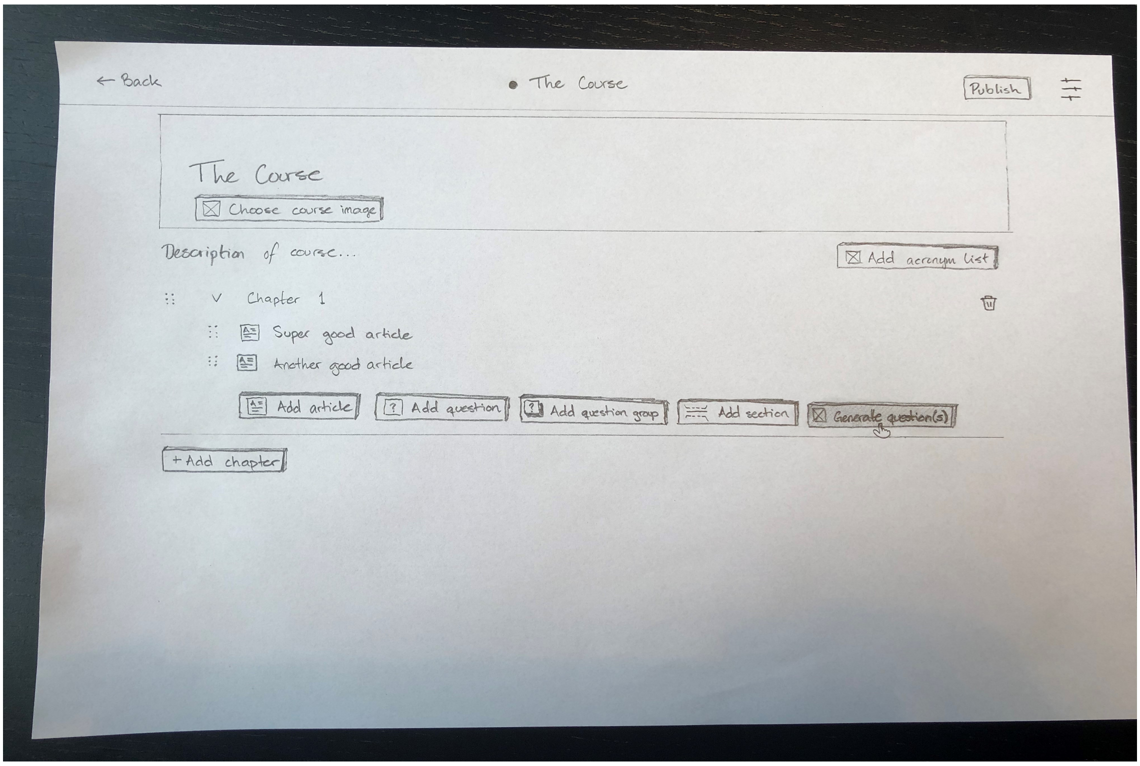
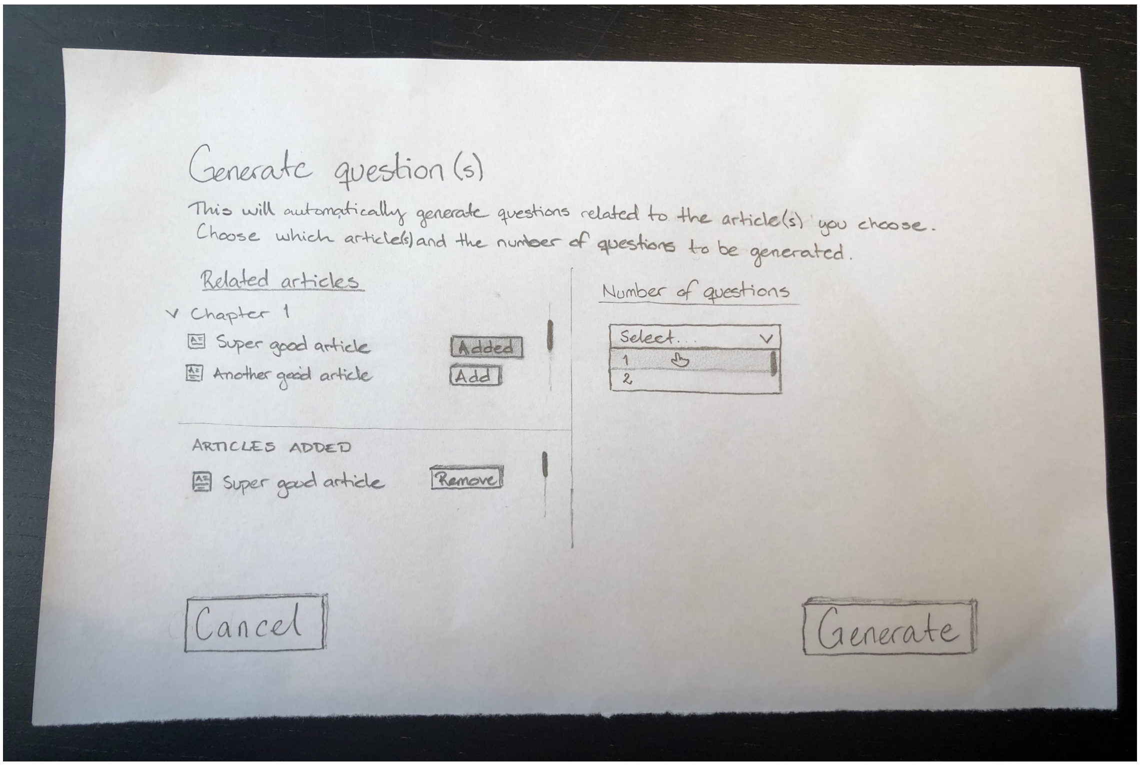


Figure 4.2: New button on course overview page



This will automatically generate questions related to the article(s) you choose. Choose which article(s) and the number of questions to be generated.

Figure 4.3: Modal for generating questions

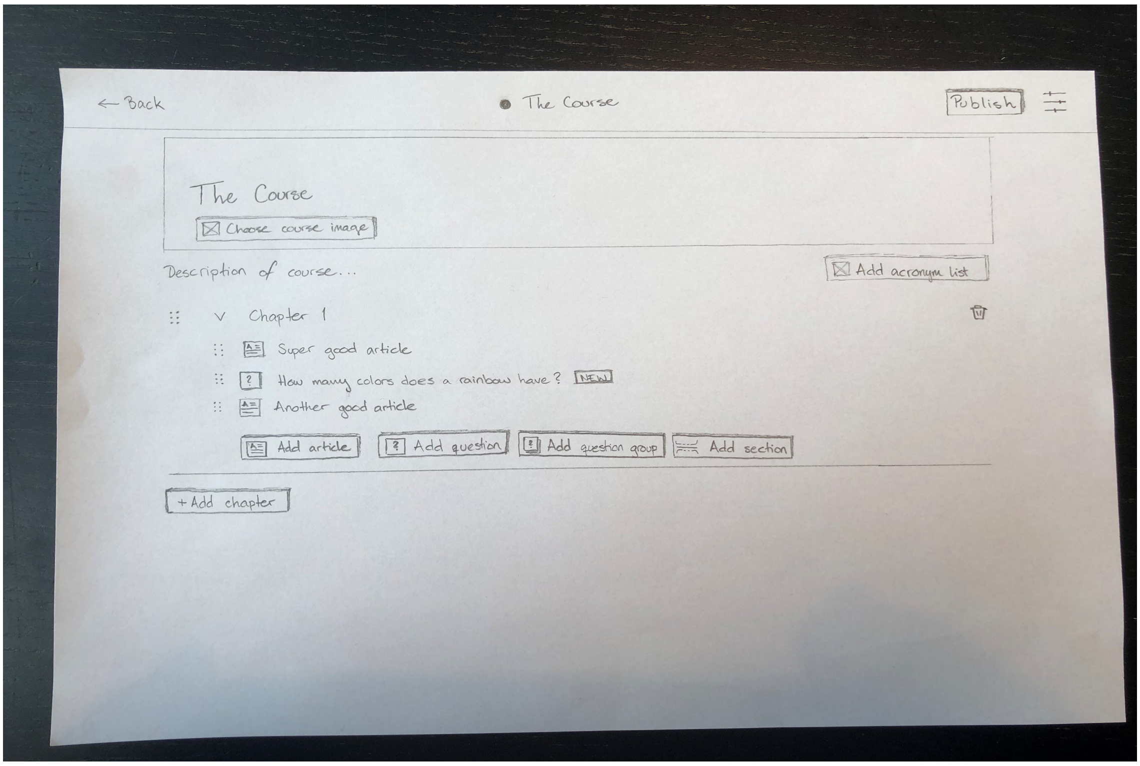


Figure 4.4: A generated question with a "new"-label beside it

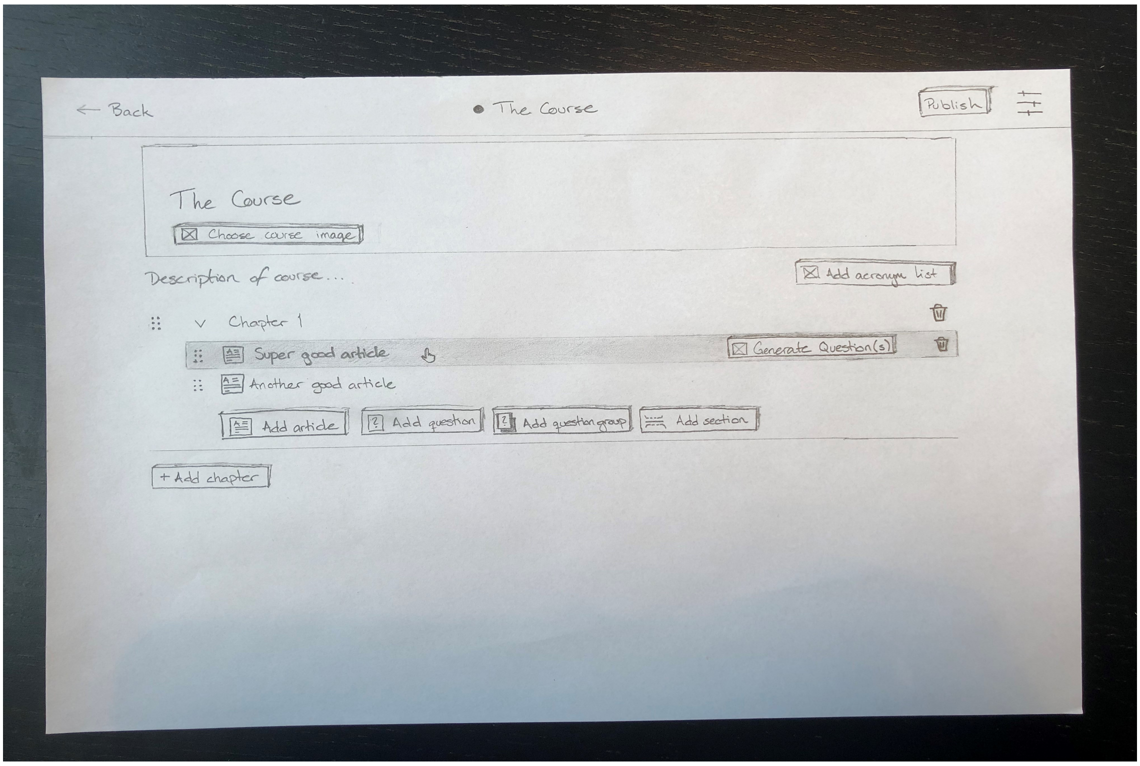


Figure 4.5: New button when hovering over an article

- **Idea three: "Generate question"-button when hovering over "Add question"/"Add question group" buttons**
  This idea means that when the user is hovering over the current "Add question" or "Add question group" buttons, two new buttons will appear below the button - see figure 4.6. One lets the user write their own question(s) and the other will automatically generate question(s). After clicking the generate-alternative, a modal will appear and then work exactly like presented in idea one.

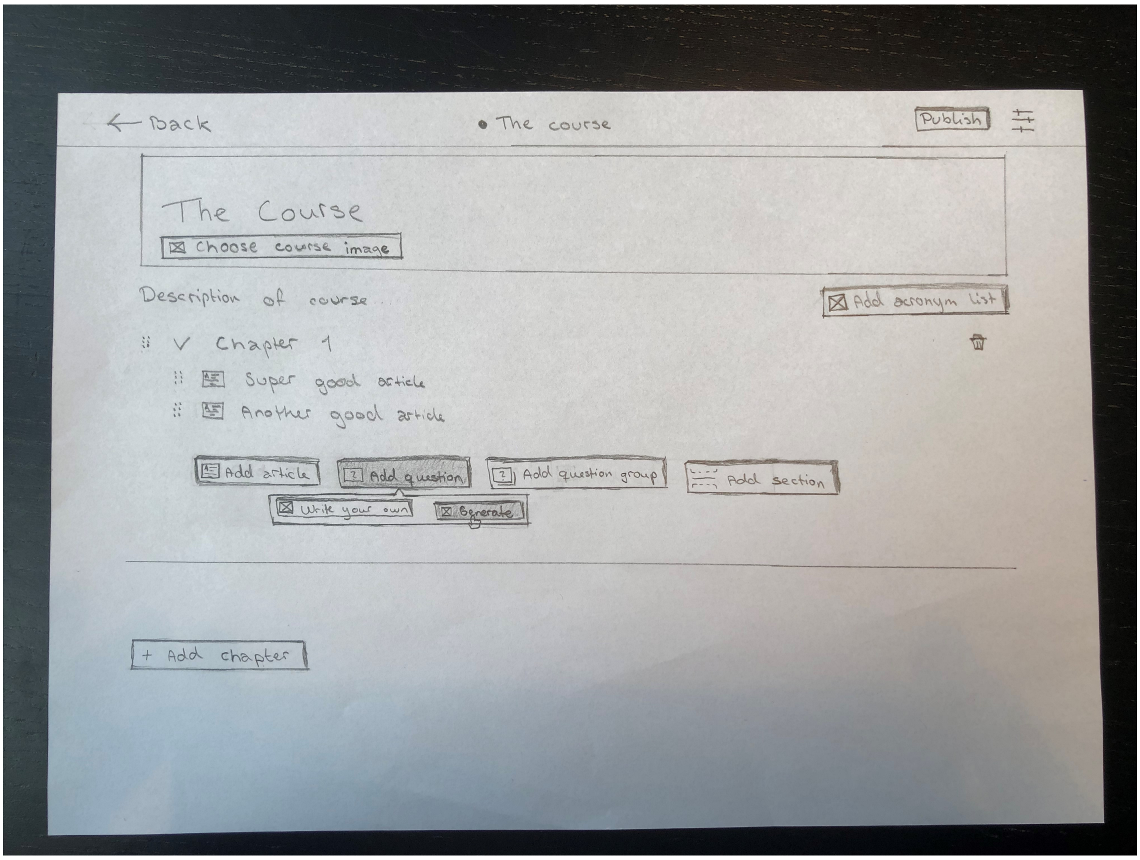


Figure 4.6: New buttons when hovering over the add question-button

- **Idea four : A button that appears when hovering over a question on the course overview** The placement of the button will be the same as shown in figure 4.5, except next to a question. Apart from where the button is located, this idea also works exactly like idea one. However when clicking on the button next to a single question the system will default to generate a single question. It must be clicked next to a question group for the user to be able to specify the number of questions to generate. This action will cause the existing content (if any) in the question to be overwritten by the generated question(s).
- **Idea five : An icon in the header of the question-editor** The last two ideas differ from the others in that the question generation is not initiated from the course overview. Instead in idea five the user must first have created a question or question group, and then clicked on it so that they enter the editor. In the header of the editor a new icon will be placed to the right of the other icons, see figure 4.7, and clicking it will prompt the modal to appear. Once again the user must choose article(s) to relate to the question(s). Also, as in idea four the system will by default generate one question if the user has entered a single question, whereas they will have the option to choose a number if they entered a question group. Once generated the question will appear directly in the opened editor as well as in the course overview to the left of the editor. In this case, no "new"-label is needed since the user will see the question immediately in their editor after generation. Again, any existing content will be overwritten by the generated question(s).

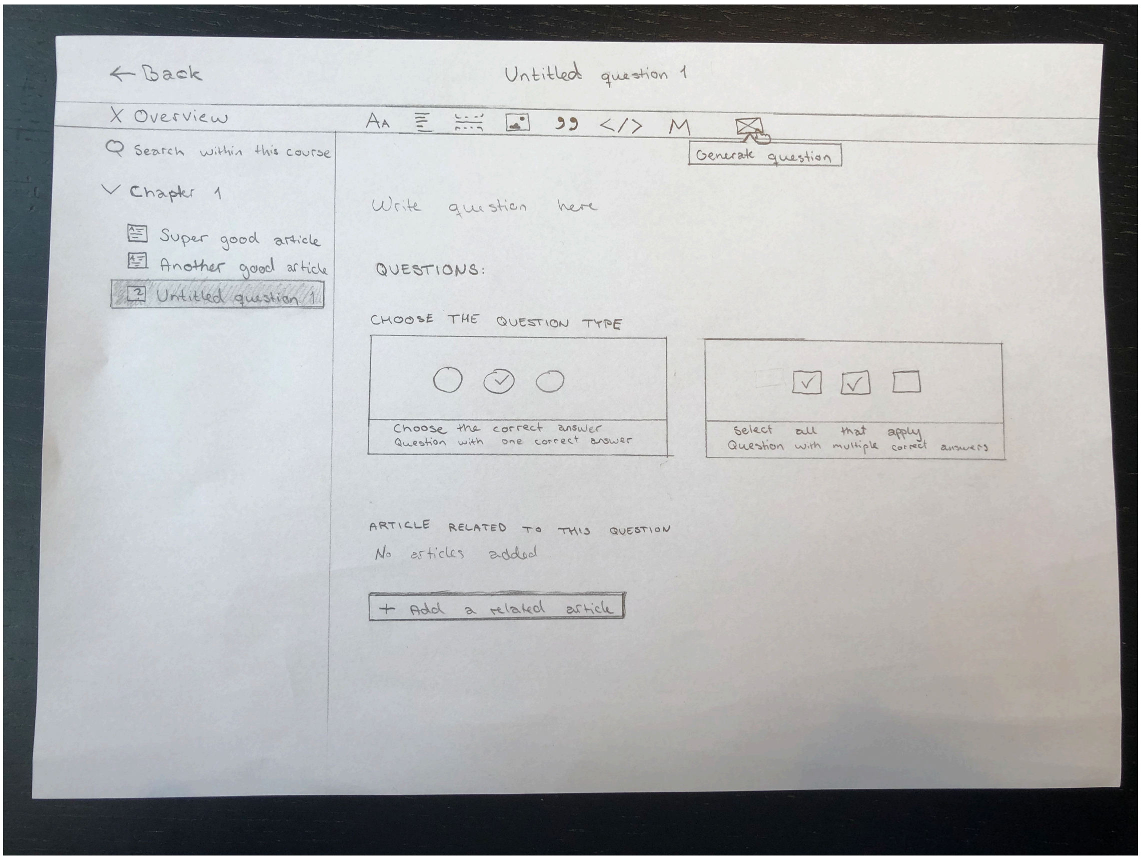
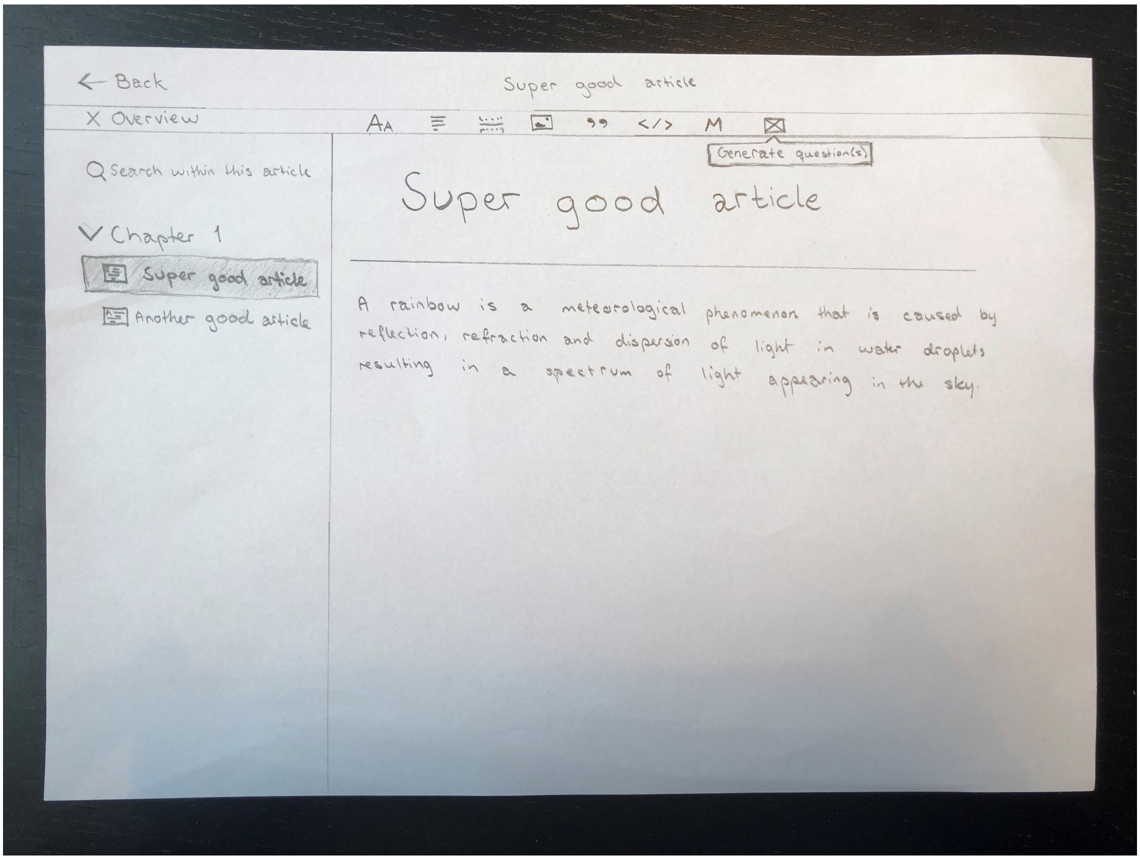


Figure 4.7: New icon in the header of the question editor

- **Idea six: An icon in the header of the article-editor**
  This idea requires that the user has created an article and has opened it in their editor. Then, as in idea five, a new icon to the right of the other icons in the editor header can be clicked in order to generate question(s), see figure 4.8. In this case, the user will only have to specify the number of wanted questions, not which articles to relate them to, since they will be automatically related to the article currently being viewed in the editor. This is done in the modal that appears upon clicking the icon. Thereafter the generated question or question group will appear in the course overview to the left of the editor with a "new"-label.



A rainbow is a meteorological phenomenon that is caused by reflection, refraction and dispersion of light in water droplets resulting in a spectrum of light appearing in the sky.

Figure 4.8: New icon in the header of the article editor

Some of these ideas work well together, for example idea one and six, and can therefore be implemented together - if needed in the future - in order to solve potential problems which are presented in section 4.2.1 below.

4.2.1 Potential issues to consider

We discussed potential advantages and disadvantages of the ideas above, as well as how the disadvantages could be solved and improved upon.

The advantages that we can see with the first idea is that it matches the current platform well, since we are just adding a new button for generating questions next to the other current buttons on the platform. This will also be intuitive for the users who are already familiar with the system and know where the tools for creating questions and articles are located. Hence it would be natural to have the automatic generation of questions tool located at the same place. A disadvantage with this idea is that many button clicks are necessary to generate a question since the user first needs to choose which article(s) to generate the question(s) from and also how many questions they would like to generate.

The second and fifth idea would create a more obvious connection to the chosen article. On the other hand this design might not be intuitive for the users as the placement of the tool does not follow the current platform design. This is also the case for the sixth idea.

Ideas three and four would solve the issue regarding how many questions that should be generated since if the user uses the "Add question" only one question will be generated and if "Add question group" is chosen, the user then needs to select a number. The disadvantages of these ideas would instead be that they deviate a lot from how the current authoring tool works. However this does not necessarily mean that it will be an issue, if the design is intuitive enough it will be easy and fast to learn.

4.2.2 Feedback on Lo-Fi prototypes

We presented our paper prototypes to Sana Labs' head of design, the director of research and our supervisor to get their input. We went through each idea and discussed advantages and disadvantages, as well as possible improvements. Also, during this meeting our understanding of the AI technology and the existing platform increased. For example, we realised that we had misunderstood the use of the question groups: We thought that all questions related to the same article should be placed in the same question group, but in reality question groups are used when you want to group some questions together under the same description. We therefore realised that it does not make sense for our project and scope to accommodate question groups at all, since the AI technology is not yet at a point where it is able to generate a common description needed for a question group or even group questions together based on a common topic within an article. From this point on we therefore focused on generating single questions only.

The meeting lastly resulted in two main ideas that we chose to move forward with. These two ideas were refined versions of our initial ideas one and six. The revisions we made were on the overall design and functionality of the modal, otherwise the placement and functionality of the button/icon that prompts the modal were kept the same. The changes that were made in the design of the modal are the following: We decided to split the modal into steps/pages instead of having all the information and functionality in the same view. On the first page the user must choose article(s) and then on the next page a question will appear. In this step the user can choose to reject/accept the question, as well as generate new questions as many times as they want. This change was decided upon in order to de-clutter the modal and improve the user's feel of control. The first page of the modal will not be shown if the user clicks on the icon in the article editor (idea six) since the related article will then automatically be the article open in the editor.

Additionally we discussed an extension to idea six: While in the article editor, the user should be able to highlight a portion of the text and get the option to generate question(s) related to this section specifically. We did however decide not to focus on this extension due to time restrictions, instead this will be an idea for future work on the project.

4.3 Hi-Fi Prototype in Figma

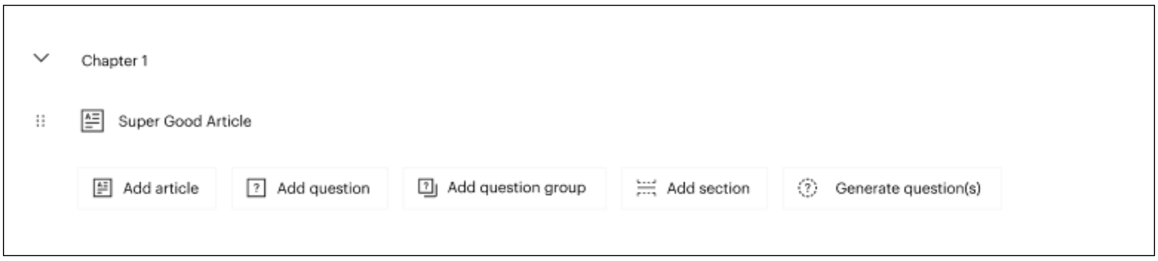
Figma is a browser-based UI and UX design application, with tools for designing, prototyping and code-generation [11]. We chose to use it for our Hi-Fi prototype as it is easy and convenient to collaborate on projects in real-time in it, and it supports interactive prototyping. Also, Sana Labs has a graphical profile in Figma that we could use, which included color schemes, fonts and buttons. The prototype that we made underwent several iterations, and the changes between each iteration are described below.

First iteration

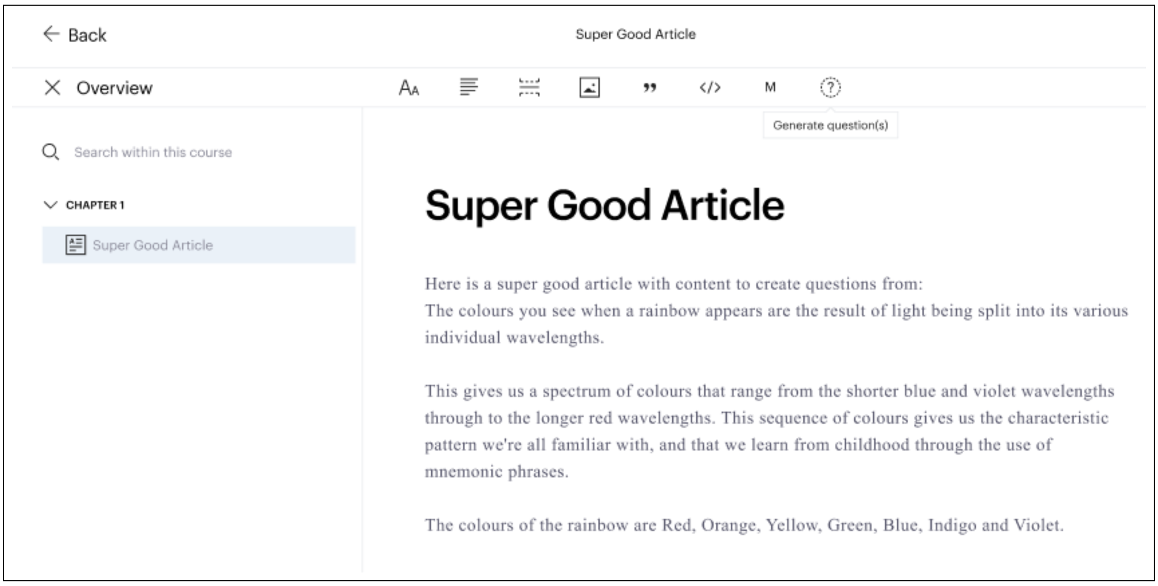
The first prototype that we built in Figma incorporated the revised versions of ideas one and six as presented above and is shown in figure 4.9. This included a button for generating questions on the course overview page (4.9a), as well as an icon with the same functionality in the article editor header (4.9b). The modal was split into two pages, letting the user choose article(s) in the first page (4.9c) and presenting generated questions in the second (4.9d). We decided to display one generated question at a time, and added a button named "New question" that when clicked generates a new question. The user can choose which questions they want to add by clicking a little check mark next to the question. Likewise, a "bad" question can be discarded by clicking the trash can next to the question. Clicking the check mark will move the question into the "Questions Added"-column. This column was added to help the users remember which questions they have added, which complies with Nielsen's sixth heuristic - recognition rather than recall - from section 2.3. Lastly the user can click on "Save changes" in order to save the questions from this column to the chapter. No answer alternatives were shown in the modal, instead the user will have to open the question in the editor as usual after having closed (and saved) the modal in order to see and edit them. The generated questions will appear in the overview with new labels next to the question title (4.9e) to improve the visibility of system status which is the first of Nielsen's heuristics from section 2.3.

Figma limitation

We used Figma's prototyping tool in order to make the prototype clickable, which facilitated testing it. Our idea includes the possibility for users to edit the question (and answer alternatives in later iterations) directly in the modal, but this was not possible to simulate in Figma. This applies to the rest of the iterations below as well.



(a) Overview page

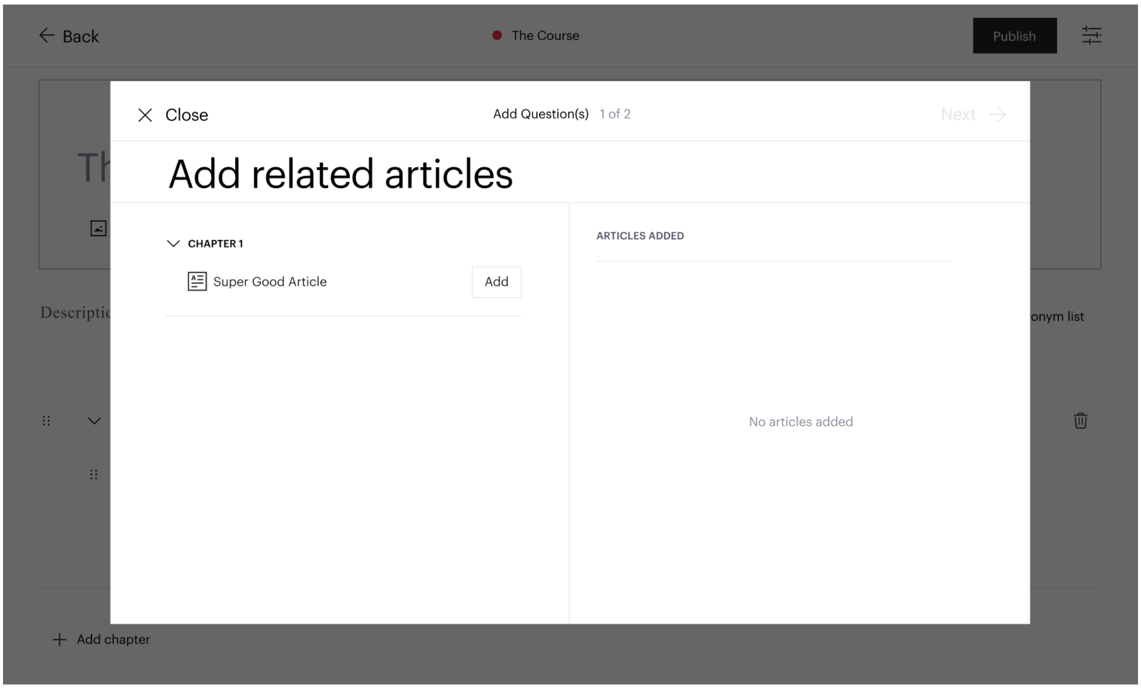


Here is a super good article with content to create questions from:
The colours you see when a rainbow appears are the result of light being split into its various individual wavelengths.

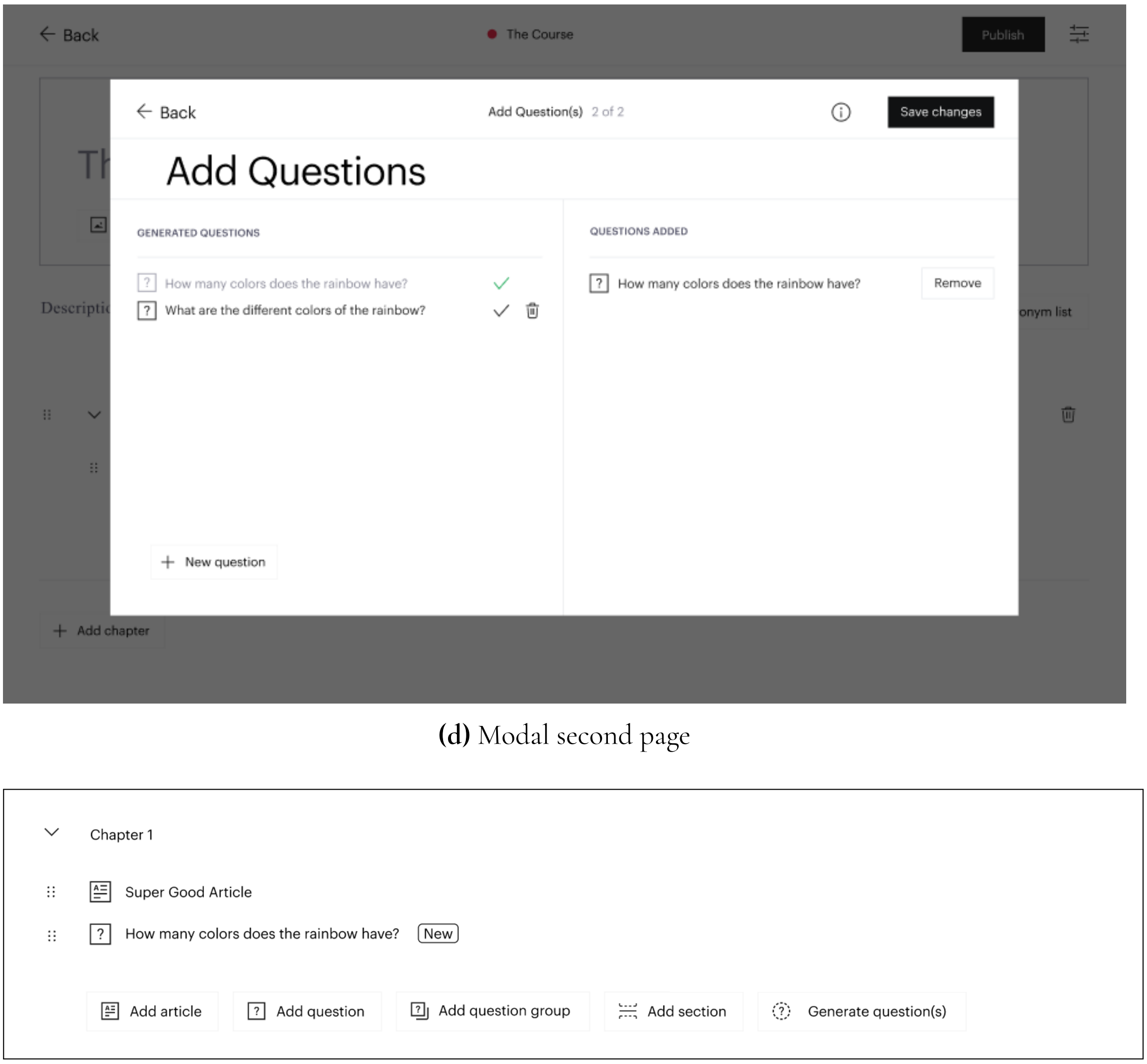
This gives us a spectrum of colours that range from the shorter blue and violet wavelengths through to the longer red wavelengths. This sequence of colours gives us the characteristic pattern we're all familiar with, and that we learn from childhood through the use of mnemonic phrases.

The colours of the rainbow are Red, Orange, Yellow, Green, Blue, Indigo and Violet.

(b) Editor page



(c) Modal first page



(d) Modal second page

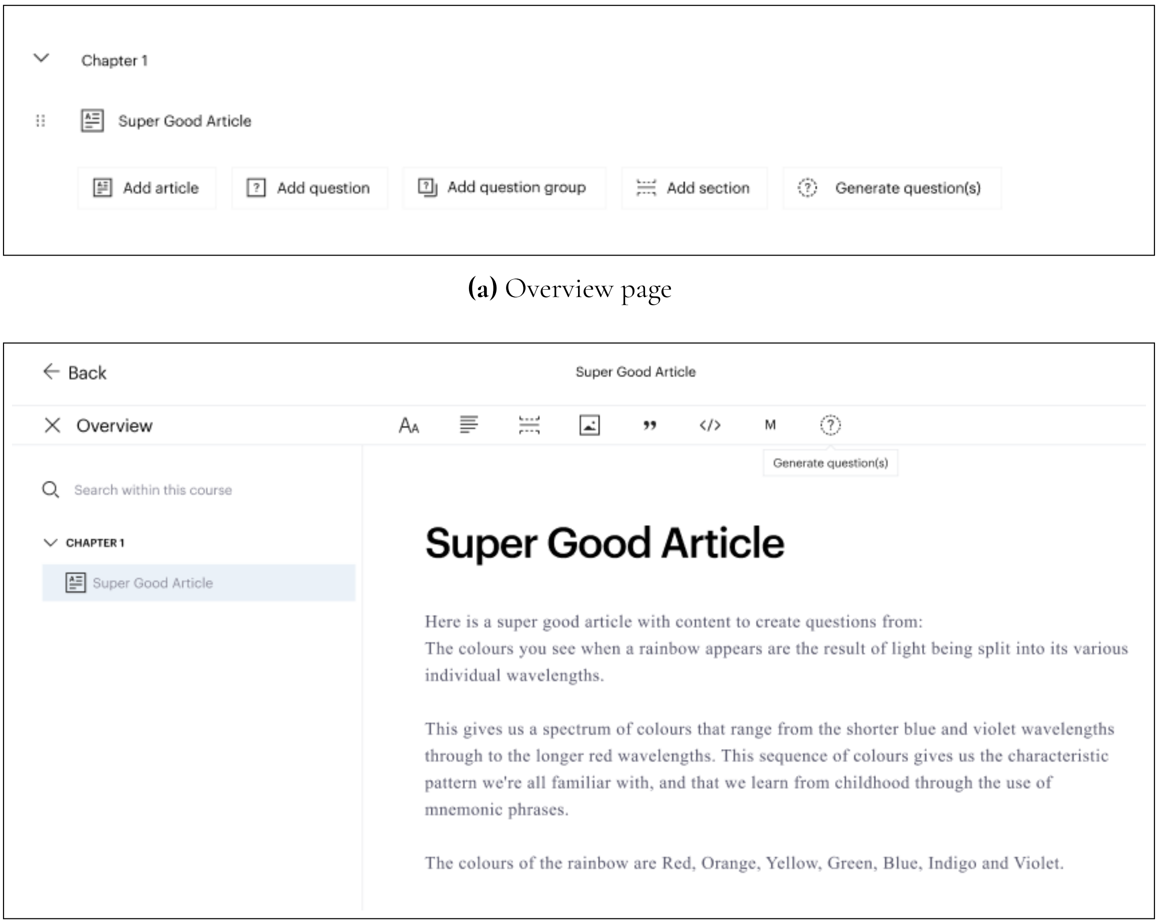
(e) Overview after question generation

Figure 4.9: First iteration

Second iteration

After showing the first iteration to our supervisors at Sana Labs we were informed that the AI technology is not yet powerful enough to generate questions that are based on several articles at the same time. This led to the second iteration which is shown in figure 4.10. The overview- and editor page stayed the same (4.10a and 4.10b), but the first page of the modal was simplified so that generated questions can only be related to one article (4.10c). Therefore, when clicking on the "generate questions"-icon in the editor, the user will be shown the second page of the modal directly. This is because the user will not have the option to choose additional articles, instead the one they have open in their editor at that point will be selected for them by default. On the other hand, if the user clicks the "generate questions"-button on the course overview, they will still need to choose an article in the first page and then click next. However they will no longer need a list showing which articles they have selected as it is only possible to have one article selected at a time.

In the second page we replaced the display of the generated questions and all buttons with a mini version of the question editor (4.10d). This means that only one question at a time will be displayed, but now the user can see the answer alternatives at the same time. Additionally, the trash can icon from the first iteration was replaced by a button which when clicked both discards the current question and generates a new one. The check mark was also replaced by a button which when clicked saves the question to the corresponding chapter. In order to inform the user that the question was successfully saved, the user is directed to a new page with a text saying that the question was added, along with a button that lets the user generate a new question (4.10e). The save button from the first iteration was removed all together, as the user will now save each question individually. Instead a "Close"-icon was introduced, which users can click at any time to exit the modal. As before, the generated questions will appear in the overview with new labels next to the question title (4.10f).



(a) Overview page

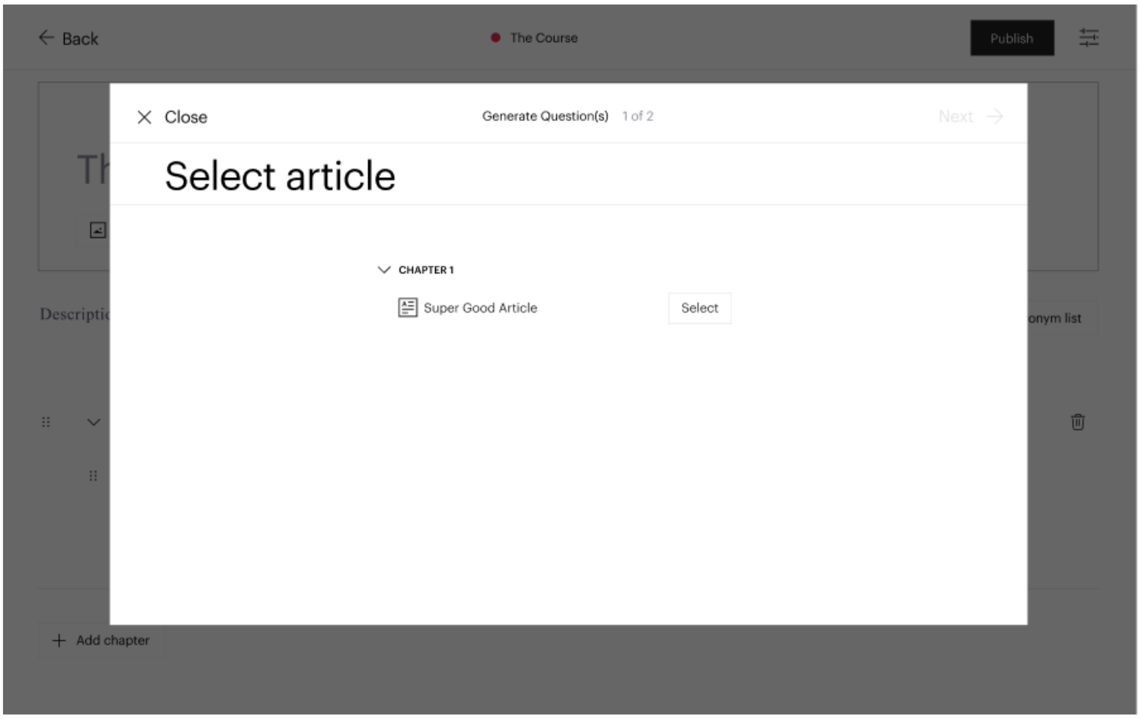
Super Good Article

Here is a super good article with content to create questions from:
The colours you see when a rainbow appears are the result of light being split into its various individual wavelengths.

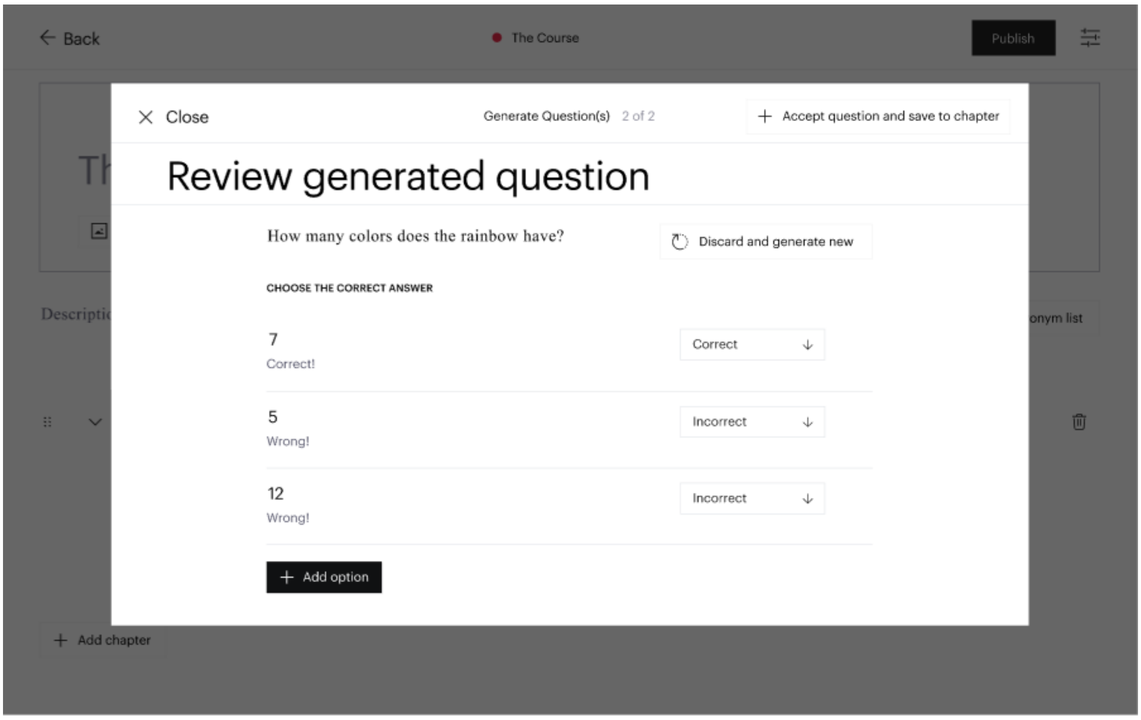
This gives us a spectrum of colours that range from the shorter blue and violet wavelengths through to the longer red wavelengths. This sequence of colours gives us the characteristic pattern we're all familiar with, and that we learn from childhood through the use of mnemonic phrases.

The colours of the rainbow are Red, Orange, Yellow, Green, Blue, Indigo and Violet.

(b) Editor page



(c) Modal first page



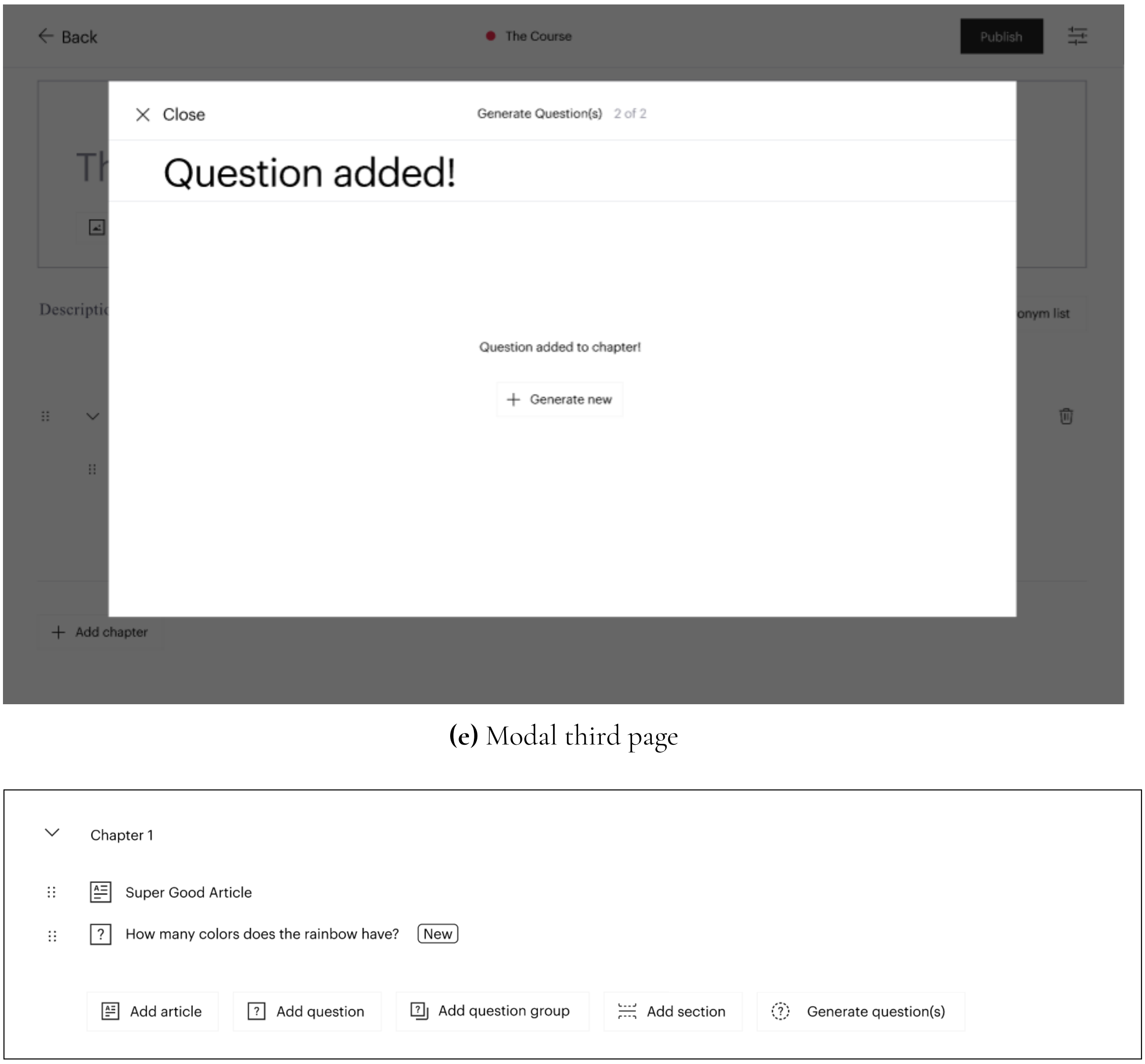
How many colors does the rainbow have?

7
Correct!

5
Wrong!

12
Wrong!

(d) Modal second page



Question added!

Question added to chapter!

(e) Modal third page

Super Good Article

How many colors does the rainbow have? New

(f) Overview after question generation

Figure 4.10: Second iteration

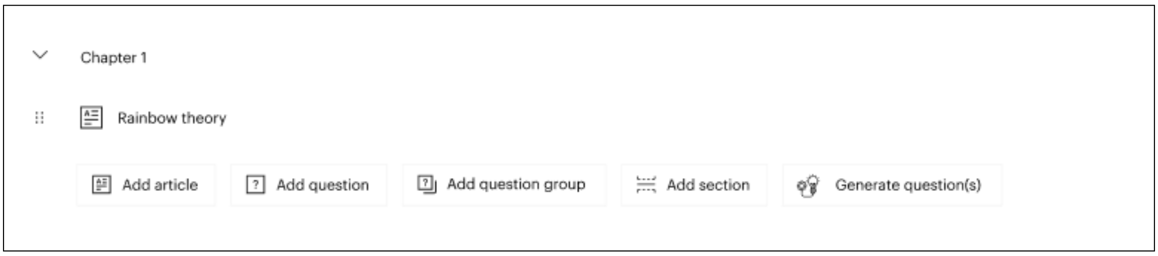
Third iteration

When discussing the second iteration with our supervisor at Sana Labs we saw that we could incorporate even more of the ideas that were voiced during the interviews (described in section 3.4) into our prototype, such as having the article text displayed while creating questions. This resulted in a third iteration which is displayed in figure 4.11. In order to prepare the prototype for testing we gave the fictional articles more professional titles. Additionally we designed a new icon for the "Generate question(s)"-button on the overview- and editor page (4.11a and 4.11b).

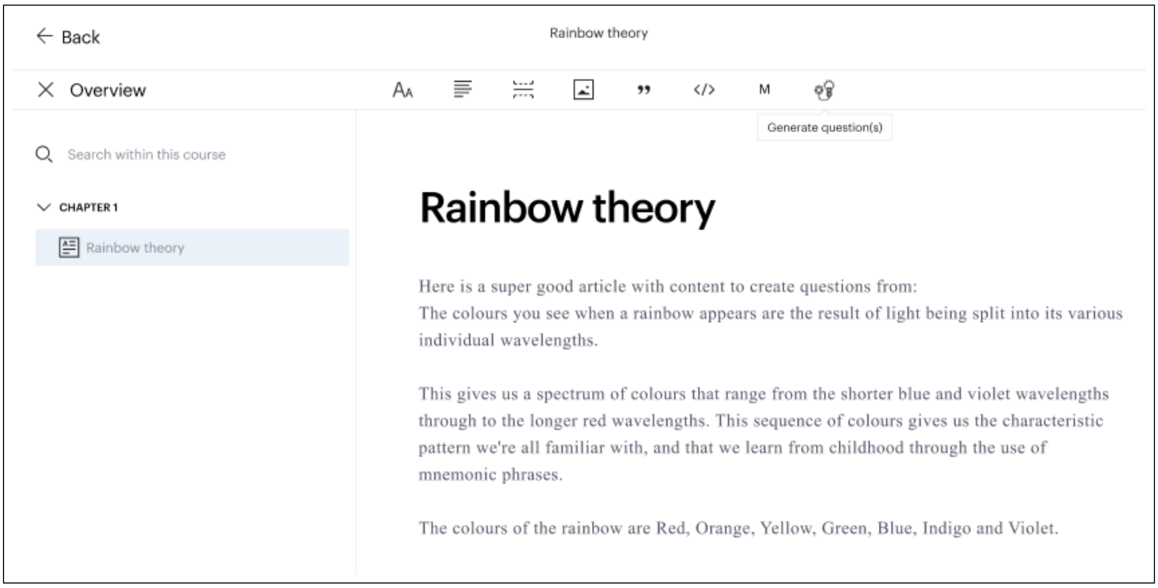
The first page of the modal was further simplified: when selecting an article the user is now redirected to the next page of the modal directly, without having to manually click next. Furthermore, articles will appear greyed-out if they do not contain enough content to generate questions from. This means that the user will not be able to select these articles, and an info-icon will be next to the article which informs the user of the issue when hovered over (4.11c).

In the second page we also made several changes. First, we made the discard button red so that users will be more hesitant to use it unless necessary, in order to comply with Nielsen's fifth heuristic - error prevention - from section 2.3. We also made the save button black and placed both the buttons on the same line, just below the heading (4.11d). Then we included a blue text box at the bottom of the modal containing an extract from the article of which the question was generated, so that the user can easily see where the question originated from. This was done in order to comply with the wish of seeing the article text when writing a related question as well as with Nielsen's sixth heuristic - recognition rather than recall - from section 2.3. Lastly we made the modal scroll-able since the new content would otherwise not fit (4.11e).

Lastly we removed the large headings on all pages of the modal, since we realised that they did not give any new information to the user. Instead we made the small headings at the top of the modal more specific to the action the user is currently performing (see for example figure 4.11f). As before, the generated questions will appear in the overview with new labels next to the question title (4.11g).



(a) Overview page



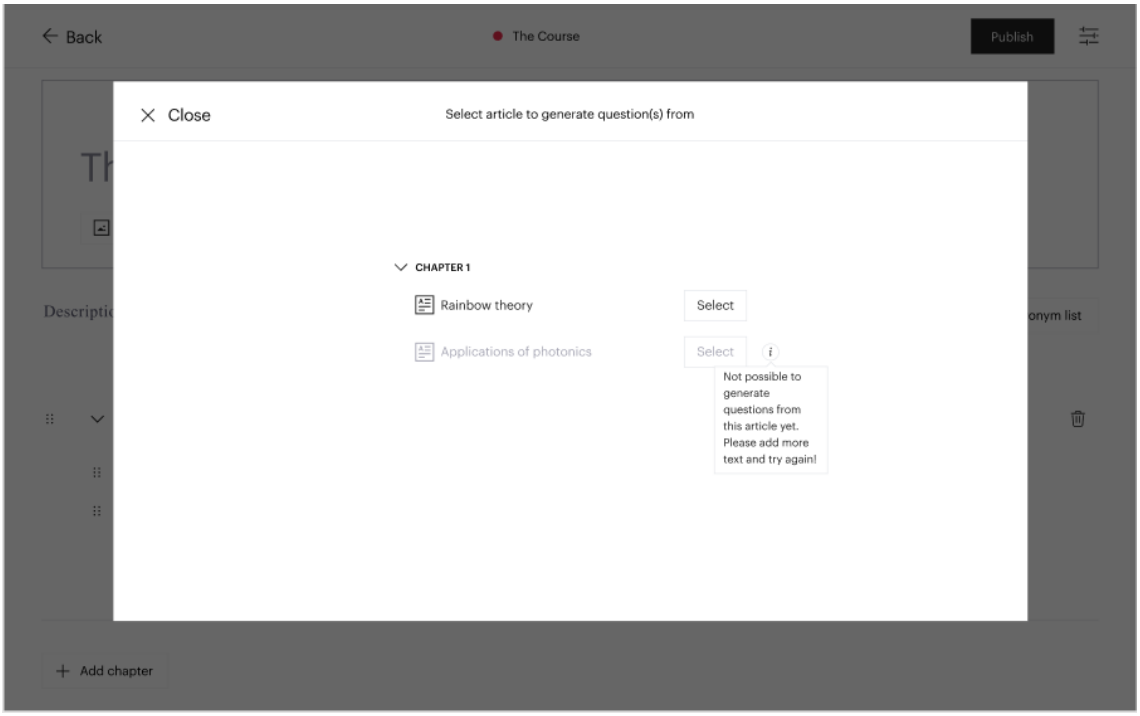
Rainbow theory

Here is a super good article with content to create questions from:
The colours you see when a rainbow appears are the result of light being split into its various individual wavelengths.

This gives us a spectrum of colours that range from the shorter blue and violet wavelengths through to the longer red wavelengths. This sequence of colours gives us the characteristic pattern we're all familiar with, and that we learn from childhood through the use of mnemonic phrases.

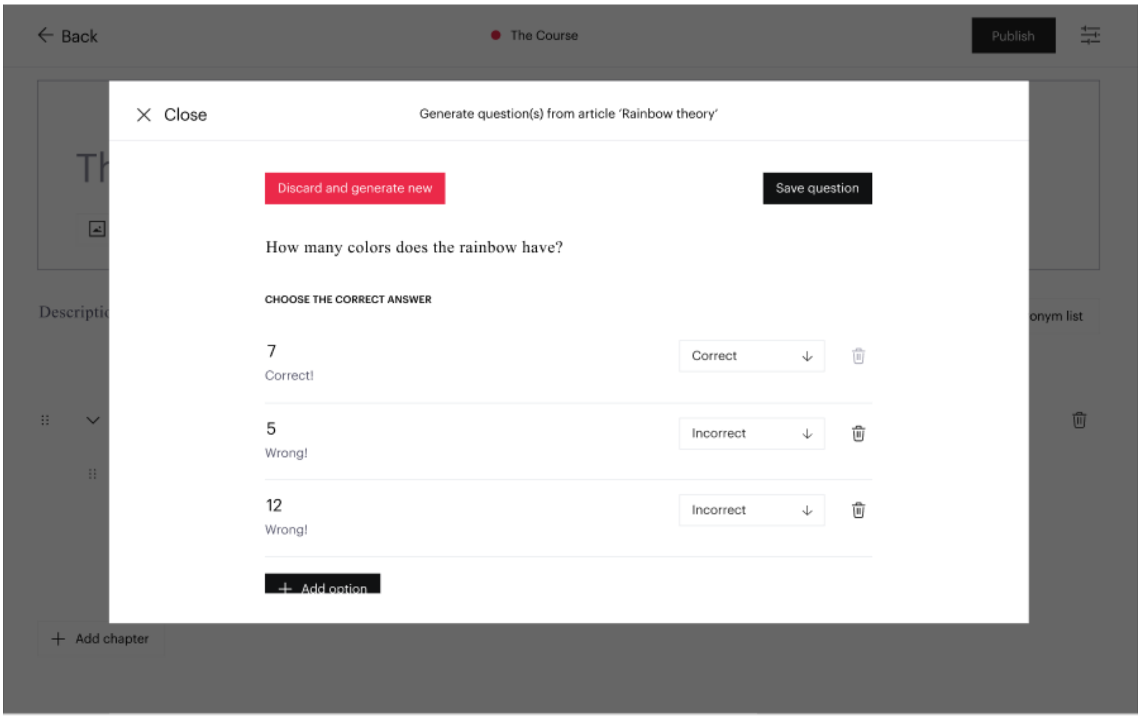
The colours of the rainbow are Red, Orange, Yellow, Green, Blue, Indigo and Violet.

(b) Editor page



Not possible to generate questions from this article yet. Please add more text and try again!

(c) Modal first page



Generate question(s) from article 'Rainbow theory'

How many colors does the rainbow have?

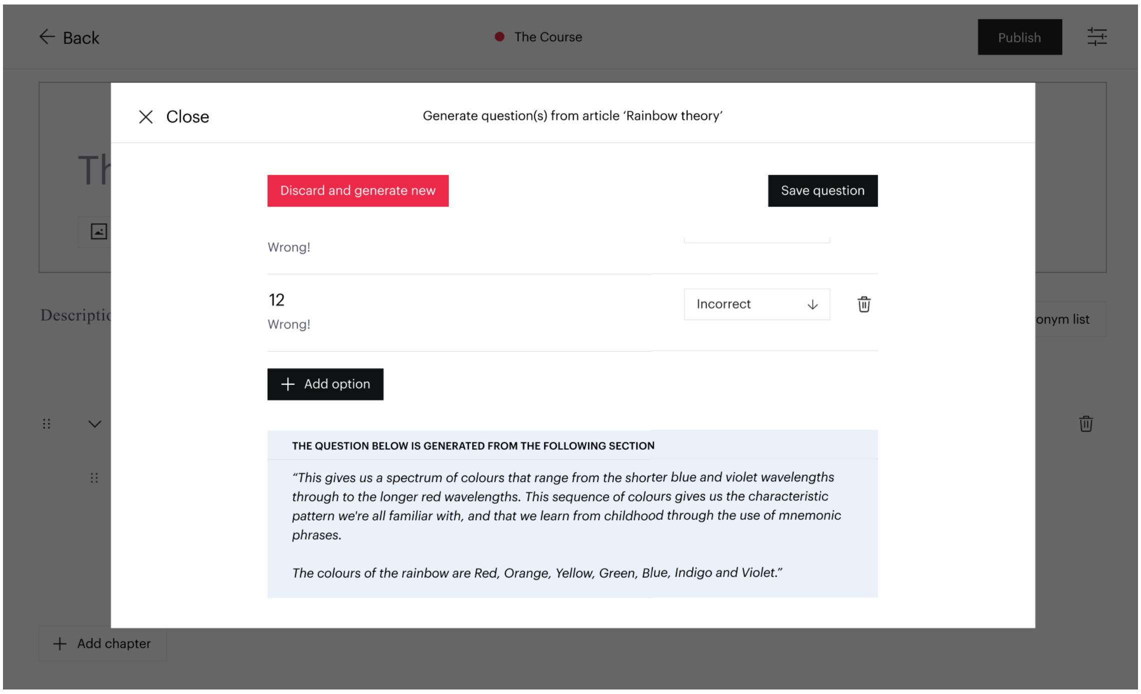
CHOOSE THE CORRECT ANSWER

7
Correct!

5
Wrong!

12
Wrong!

(d) Modal second page



Generate question(s) from article 'Rainbow theory'

Wrong!

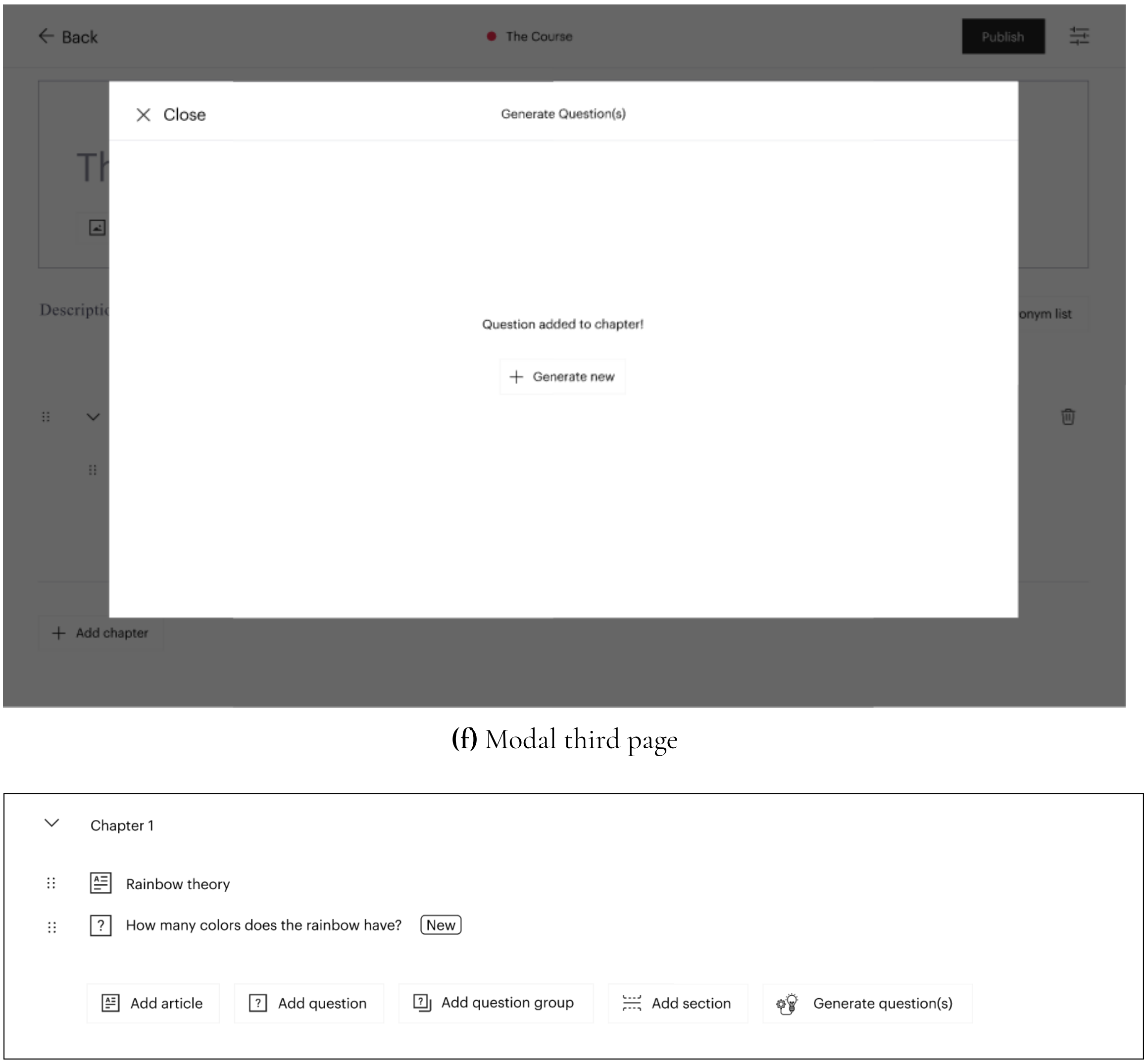
12
Wrong!

THE QUESTION BELOW IS GENERATED FROM THE FOLLOWING SECTION

"This gives us a spectrum of colours that range from the shorter blue and violet wavelengths through to the longer red wavelengths. This sequence of colours gives us the characteristic pattern we're all familiar with, and that we learn from childhood through the use of mnemonic phrases.

The colours of the rainbow are Red, Orange, Yellow, Green, Blue, Indigo and Violet."

(e) Modal second page scrolled down



Question added to chapter!

(f) Modal third page

Chapter 1

Rainbow theory

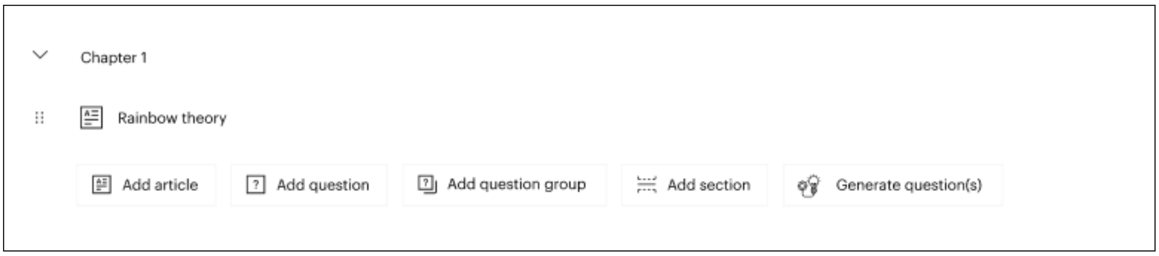
How many colors does the rainbow have? New

(g) Overview after question generation

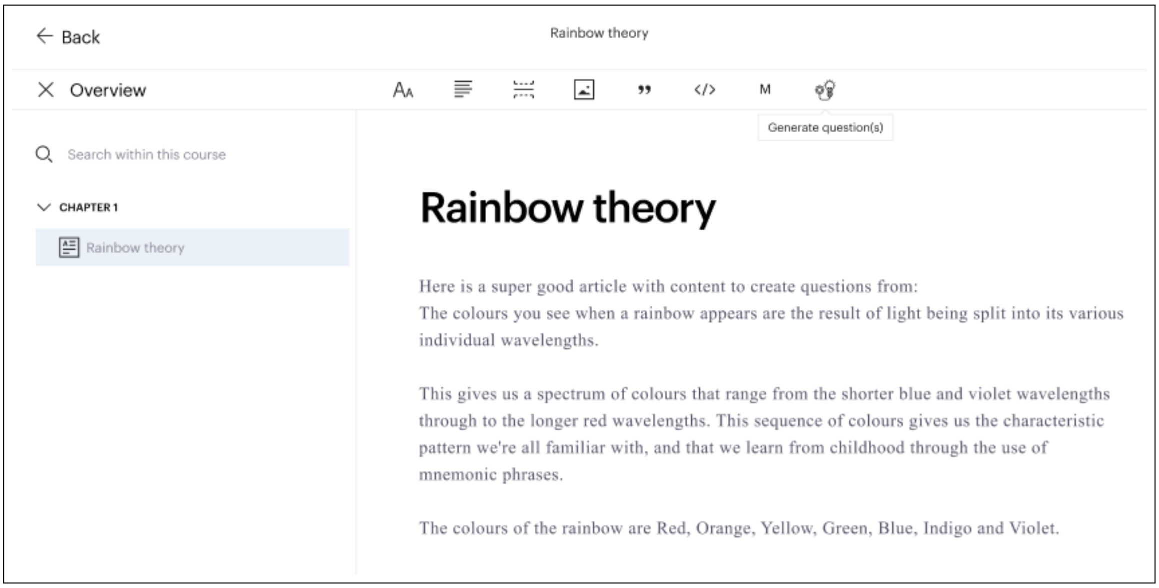
Figure 4.11: Third iteration

Fourth iteration

We decided to do yet another iteration on the prototype in order to test a slightly different design for having the article text visible on the screen at the time of question creation. This iteration can be seen in figure 4.12. The overview-, editor- and first page stayed the same (4.12a, 4.12b and 4.12c). However we split the second page of the modal into two steps. The first of these now shows only the generated, editable question and the relevant paragraph from the related article directly below in the blue text box (4.12d). This way we make sure that the user does not have to scroll down to see the text box or misses it altogether. Then, the "save"-button was replaced by a "configure question"-button which upon click directs the user to the next step where also the editable question alternatives are shown (4.12e) and this step is therefore identical to the second page of iteration three (4.11d). There were no changes made to the third page in this iteration (4.12f). As before, the generated questions will appear in the overview with new labels next to the question title (4.12g).



(a) Overview page



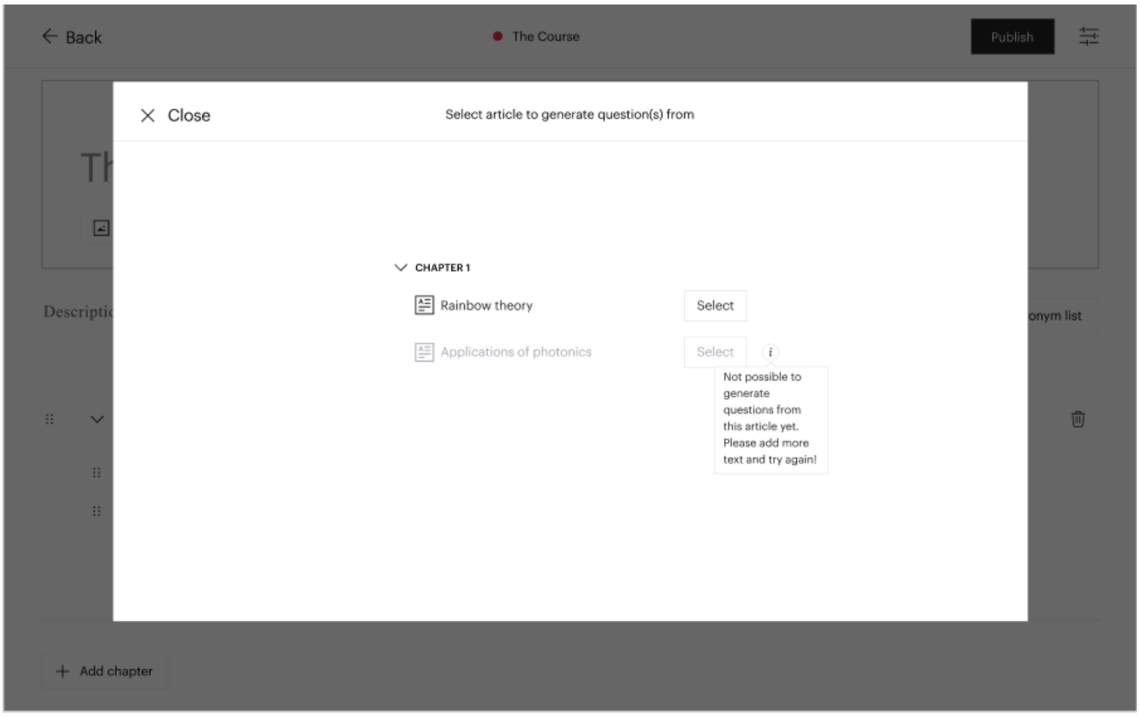
Rainbow theory

Here is a super good article with content to create questions from:
The colours you see when a rainbow appears are the result of light being split into its various individual wavelengths.

This gives us a spectrum of colours that range from the shorter blue and violet wavelengths through to the longer red wavelengths. This sequence of colours gives us the characteristic pattern we're all familiar with, and that we learn from childhood through the use of mnemonic phrases.

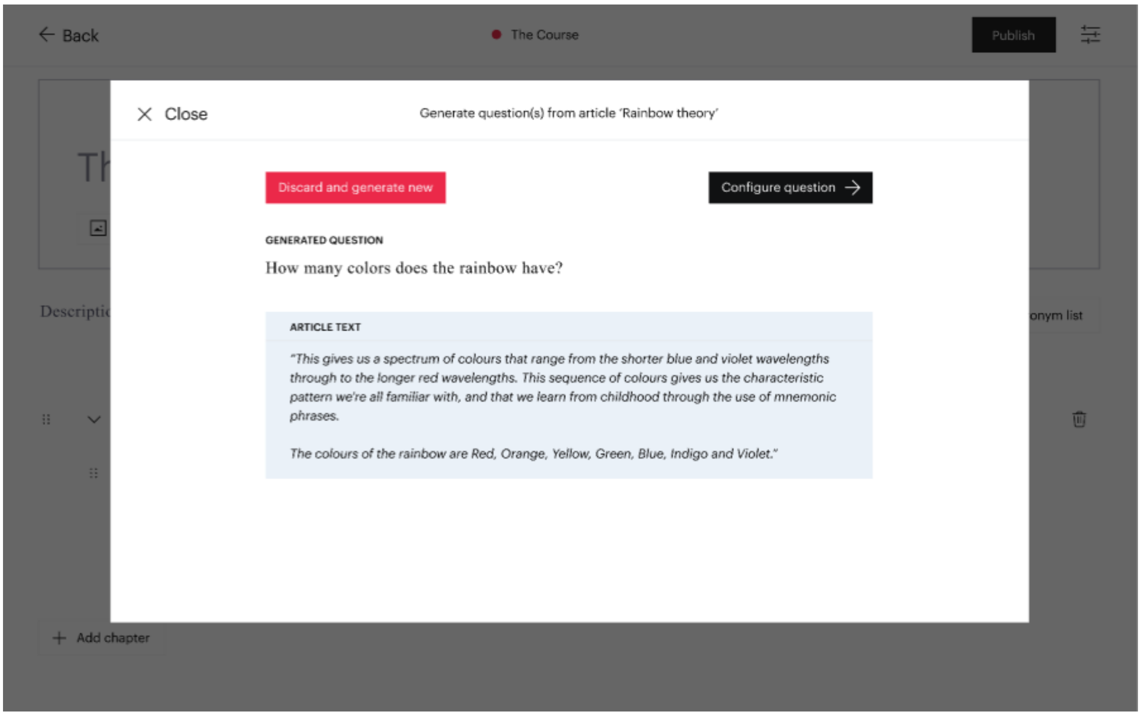
The colours of the rainbow are Red, Orange, Yellow, Green, Blue, Indigo and Violet.

(b) Editor page



Not possible to generate questions from this article yet. Please add more text and try again!

(c) Modal first page



GENERATED QUESTION

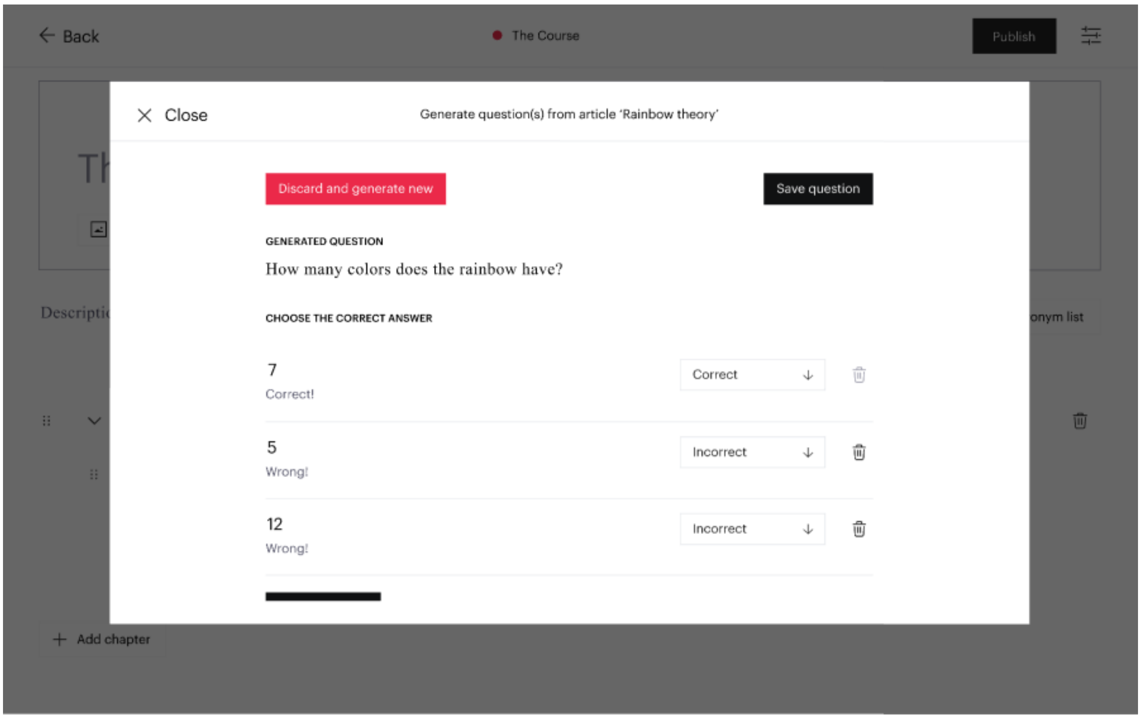
How many colors does the rainbow have?

ARTICLE TEXT

"This gives us a spectrum of colours that range from the shorter blue and violet wavelengths through to the longer red wavelengths. This sequence of colours gives us the characteristic pattern we're all familiar with, and that we learn from childhood through the use of mnemonic phrases.

The colours of the rainbow are Red, Orange, Yellow, Green, Blue, Indigo and Violet."

(d) Modal second page (1/2)



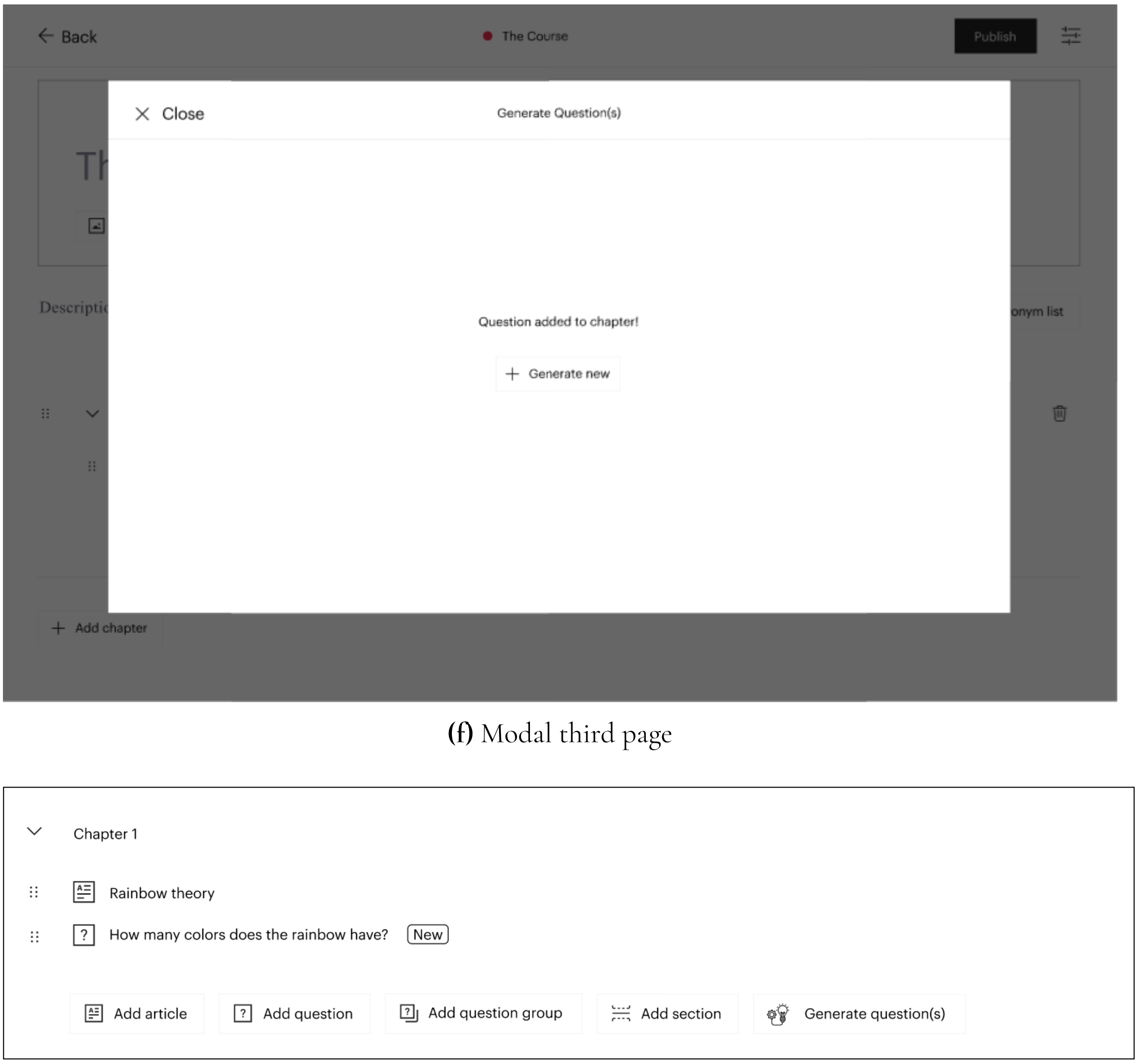
GENERATED QUESTION

How many colors does the rainbow have?

CHOOSE THE CORRECT ANSWER

7 Correct!	Correct
5 Wrong!	Incorrect
12 Wrong!	Incorrect

(e) Modal second page (2/2)



Generate Question(s)

Question added to chapter!

Chapter 1

Rainbow theory

How many colors does the rainbow have? New

(f) Modal third page

(g) Overview after question generation

Figure 4.12: Fourth iteration

4.4 Testing

In order to get feedback on our prototype we conducted user tests on the fourth iteration. This was done to see how our design choices were received and whether any big issues would arise that we would have to address before starting to implement the feature. Participants, process and findings are presented in the following subsections.

4.4.1 Participants

We had five employees at Sana Labs and one external user test our prototype since they were familiar with the current platform and our project. The testers were between 20 and 33 years old (mean 26.5 years old), and they all have proficient computer skills. The testers all have different roles and responsibilities at their job, but common for all of them is that they either develop or use different parts of the platform.

4.4.2 Procedure

The tests were conducted via Google Meet and the audio was recorded throughout the whole test. For the scenarios, the participants were sent a link to the prototype and asked to share their screen as they interacted with it so that we could make screen recordings. We also took notes throughout the whole test, so the recordings were only used to help summarize our findings afterwards. The interviewees were informed of this and what the purpose of the test was, and they were asked to sign an informed consent (see appendix A) form before we started. The tests lasted approximately 30 minutes.

Before the participants gained access to the prototype they were also informed of what a Hi-Fi prototype is and which limitations it has. This was important so that the participants are aware that not all of the functionality that they are used to from the real product will work in the prototype. This helps them stay focused on the functionality of the new feature. We also explained that in the implemented version of the prototype it will be possible to edit questions and answers directly in the modal, but that it was not possible to simulate in Figma.

We constructed two scenarios that would allow us to see how the testers interacted with the new functionality. Each scenario covers one of the two possible user flows present in the prototype, and are described below. Before the scenarios we also gathered some demographic information in order to get an understanding of the test subjects' background. Afterwards we asked some follow-up questions which were inspired by standardized usability questions [37]. The purpose of asking these questions were to gather feedback that might not have been communicated to us during the scenarios.

- **First scenario : Generate question from course overview**
  You have just created a new course, called "The Course", and added a new article called "Rainbow theory" with some content in it. Now you think "Hmm I need some questions for this article" but you don't have any ideas for questions yourself yet. So therefore you decide to let the platform generate a question for you. How do you go about this?

- **Second scenario : Generate question from editor**
  You have clicked on the article "Rainbow theory" to open it in the editor. You are happy with the text, and would like to create some questions to go along with the article. You would like some help to come up with good questions. Without exiting the editor, you would therefore like to have the platform generate a question for you. How do you go about this?

Follow up questions :

- Was there anything that surprised you? If yes, what?
- Was there anything you expected to find that was not there?
- What was difficult or strange about this experience, if anything?

- Was there anything that did not look the way you expected? If so, what was it?
- Was anything out of place? If so, what was it?
- If you had a magic wand and could change anything you would like, what would you change about this experience?
- Did any of the tasks feel tedious; would you rather have just created your own questions or do you feel like this tool could be helpful?

4.4.3 Findings

What we found from the testing of our Hi-Fi prototype was that overall the participants felt at ease with the design. They all expressed that the design made the decision making straight forward since it would guide you through the entire process in a clear way.

Two of the participants that are actual users of the authoring tool today, expressed that it was especially good that the paragraph from the article where the question was generated from is shown at the same time as the generated question. It made them feel in control, and allowed them to check that the question is correct and well worded. They especially liked that it was just a paragraph and not the entire article. They also liked that you could discard a question directly when it shows, since they thought that they often could decide whether it was a good question pretty fast when first seeing it.

Some of the participants said that they thought they where going to be able to choose how many questions that should be generated. However when they thought more about it, they realised that our solution was actually more practical since it keeps the user in control as they either way would have to see each question before deciding whether to keep it. One participant also mentioned that the "generate question"-button should be placed to the right of "add question group"-button, instead of to the far right, so that all the buttons related to questions will be grouped together.

When asked what other functionality they wanted to have, one participant wanted to be able to choose some conditions for the questions before generating them. For example, if the generated question should be easy or hard or if it should be a true or false answer or more of a case type of question. Also some of the participants expressed that they would want to be able to select a specific part of the article to generate a question from and not only from the entire article text. Another participant said it would be better if the already existing "Add question"-button could be used instead and then the "generate question" tool should be inside the question editor, essentially our idea five from section 4.2. This way would make the design be more in line with the existing system.

Issues

The biggest concern that the participants expressed was that they felt that the step where the generated question was shown and the next step which was to configure the question should just be combined into one step. They expressed that it would be more intuitive if it was all shown at once, because that would give the user a better overview before you have to decide whether to keep the question or generate a new one.

Another issue that was brought up, was the fact that the "generate question"-icon inside the article editor was out of place, since it was placed next to the content editing tools. An idea to make the placement more intuitive was that all of the "add..."-buttons that today exist in the course overview page should also be inside the article editor. For example in the overview on the left side in the article editor view. Otherwise, it was given as a suggestion to at least have a separating line between the editing tools and the generate questions icon, so that it would be clearer that this icon has nothing to do with the other editing tools.

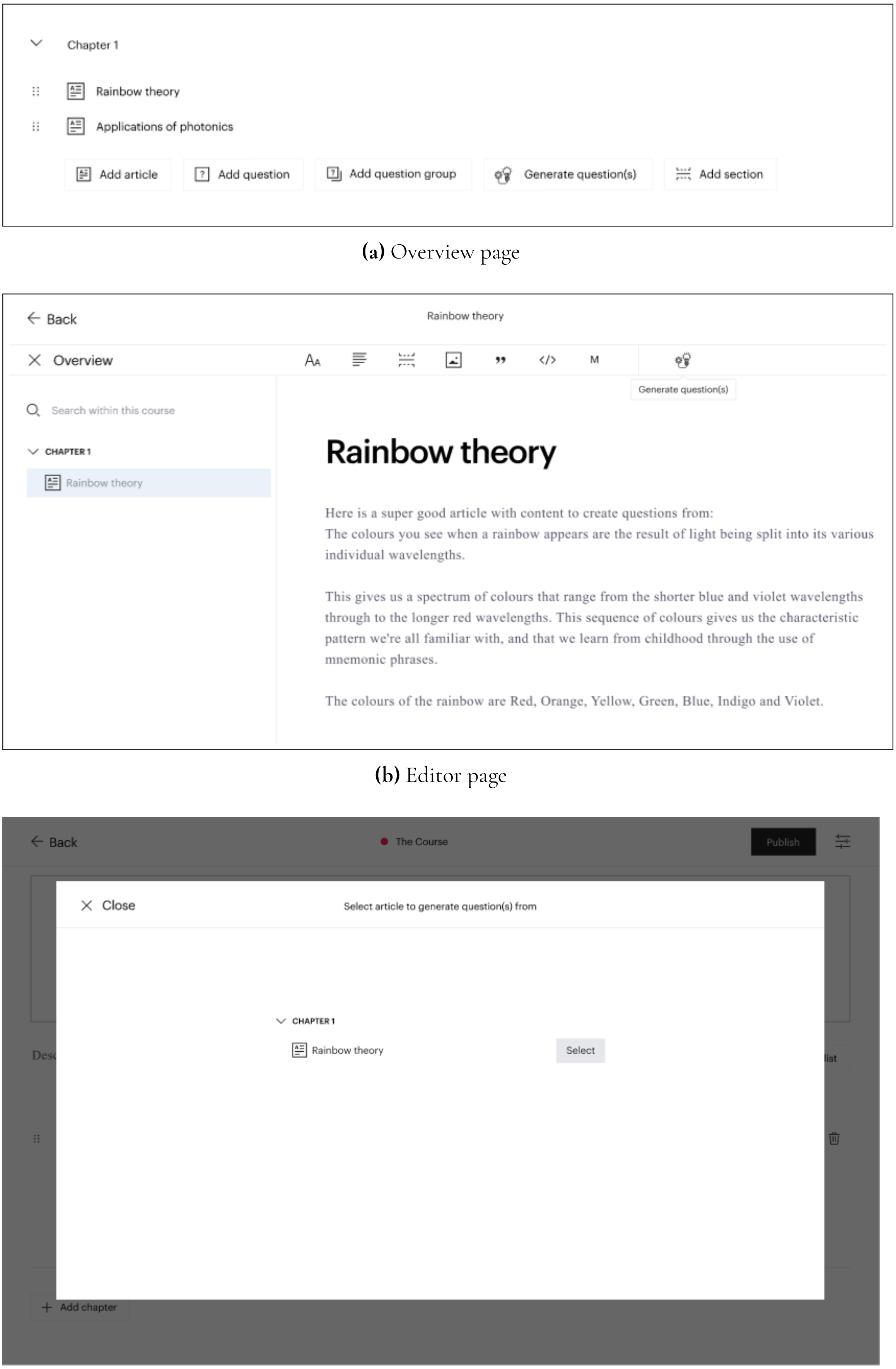
Two of the internal participants expressed that the flow of the feature was a bit slow and tedious to go through each time a new question was generated and did not see why the user could not just pick a number of questions to be generated and then save the ones they liked. Also, they felt like the first step of having to chose which article to generate the questions from was a bit odd, since it was unclear if it would be possible to choose more than one article. One of them also thought it was a little tedious to not be able to just go into the article if it was not select-able due to lack of content, in order to add more text straight away.

Fifth iteration

After having tested the prototype we decided that we would have to do another iteration on the prototype in order to address the concerns that were voiced. This iteration is shown in figure 4.13. We started by reverting the prototype to the third iteration, because splitting up the second page of the modal was not received well by the testers. Then we moved the "Generate question(s)"-button on the overview page so that it is situated between the "Add question group"- and "Add section"-buttons instead of being to the far right (4.13a) as was pointed out to us during the tests. We also placed a line between the other icons and the generate questions icon in the editor page to better display that our icon differs in functionality from the other icons, which was a concern voiced by several of our testers (4.13b).

We also removed the info icon next to the empty articles on the first page of the modal (4.13c), since we received feedback that it was confusing to use info icons in this setting. We then came up with a new design for the second page to solve the problem of having the article text visible on the screen at the same time as the question/answers: We increased the size of the modal slightly, and split page two into two columns. The left column contains the question with the answer alternatives below, and the right column contains the text box with the article paragraph. All the buttons were also moved into the left column to make it clear that they are connected to the question (4.13d). Both columns are scroll-able, but independently of each other.

Lastly we transferred this column-design to the last page of the modal: The text confirming that the question was saved and the button to generate a new question was moved to the left column, and in the right column we inserted a list of all the questions that the user has chosen to add up to this point (4.13e). This list was added because many testers felt that the page was too empty and additionally they would like to be reminded of how many and which questions they have previously added before generating a new question. We placed trash can icons next to the questions in the list as well, so that users can remove questions without having to exit the modal, should they change their mind throughout the process. As before, the generated questions will appear in the overview with new labels next to the question title (4.13f).



(a) Overview page

Rainbow theory

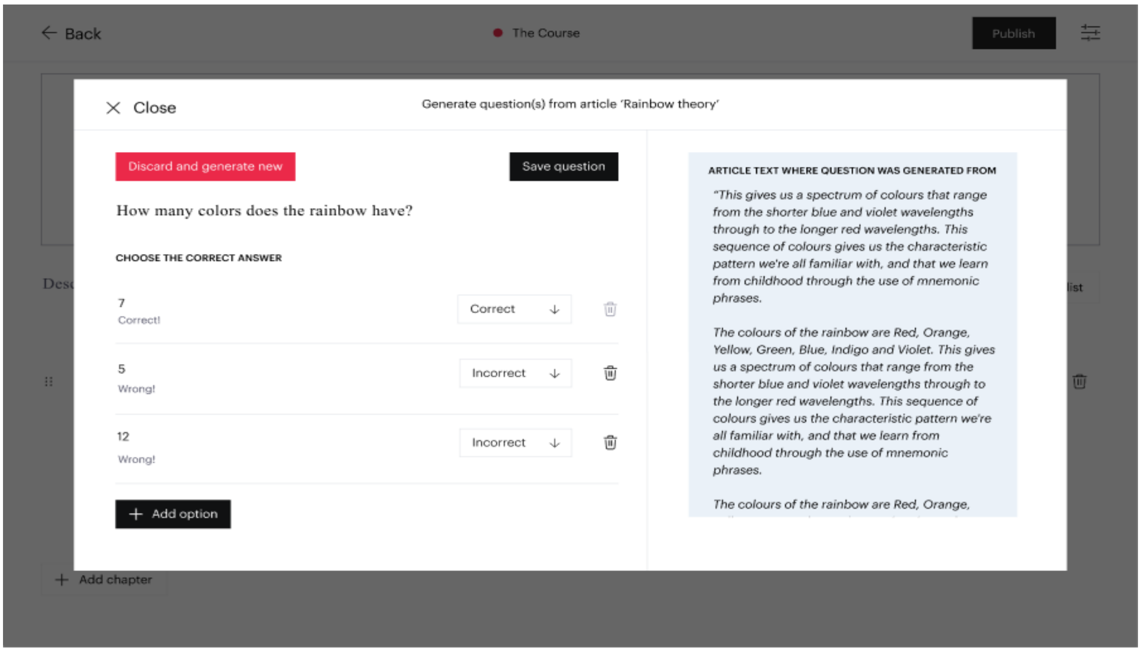
Here is a super good article with content to create questions from:
The colours you see when a rainbow appears are the result of light being split into its various individual wavelengths.

This gives us a spectrum of colours that range from the shorter blue and violet wavelengths through to the longer red wavelengths. This sequence of colours gives us the characteristic pattern we're all familiar with, and that we learn from childhood through the use of mnemonic phrases.

The colours of the rainbow are Red, Orange, Yellow, Green, Blue, Indigo and Violet.

(b) Editor page

(c) Modal first page



Generate question(s) from article 'Rainbow theory'

How many colors does the rainbow have?

CHOOSE THE CORRECT ANSWER

7
Correct!
Correct

5
Wrong!
Incorrect

12
Wrong!
Incorrect

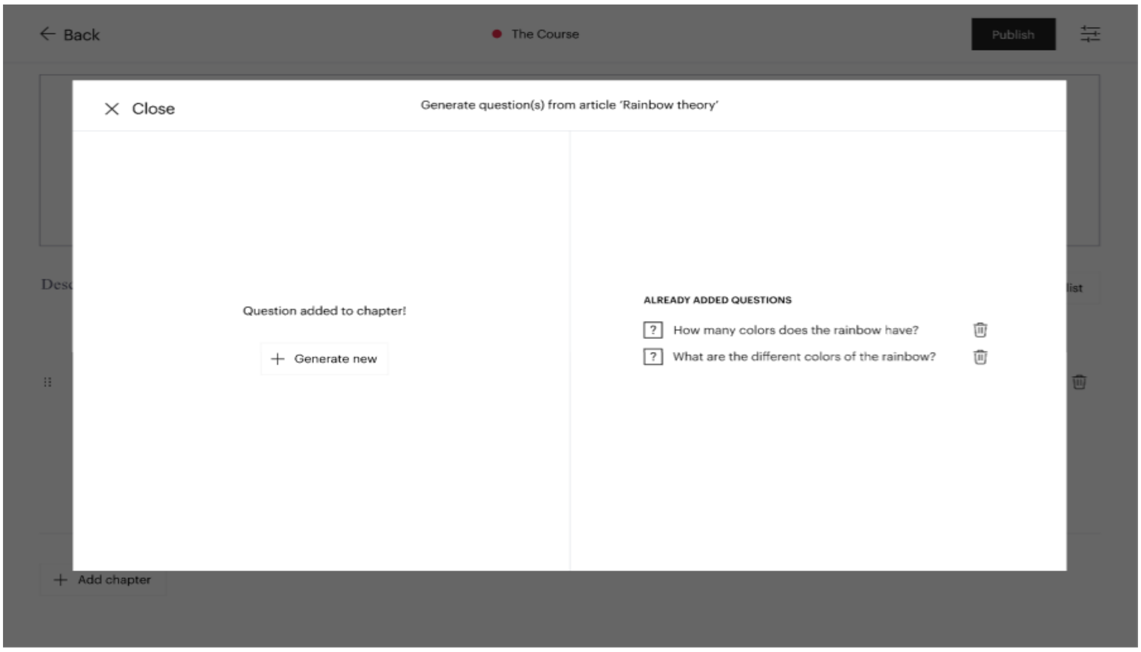
ARTICLE TEXT WHERE QUESTION WAS GENERATED FROM

"This gives us a spectrum of colours that range from the shorter blue and violet wavelengths through to the longer red wavelengths. This sequence of colours gives us the characteristic pattern we're all familiar with, and that we learn from childhood through the use of mnemonic phrases.

The colours of the rainbow are Red, Orange, Yellow, Green, Blue, Indigo and Violet. This gives us a spectrum of colours that range from the shorter blue and violet wavelengths through to the longer red wavelengths. This sequence of colours gives us the characteristic pattern we're all familiar with, and that we learn from childhood through the use of mnemonic phrases.

The colours of the rainbow are Red, Orange,

(d) Modal second page



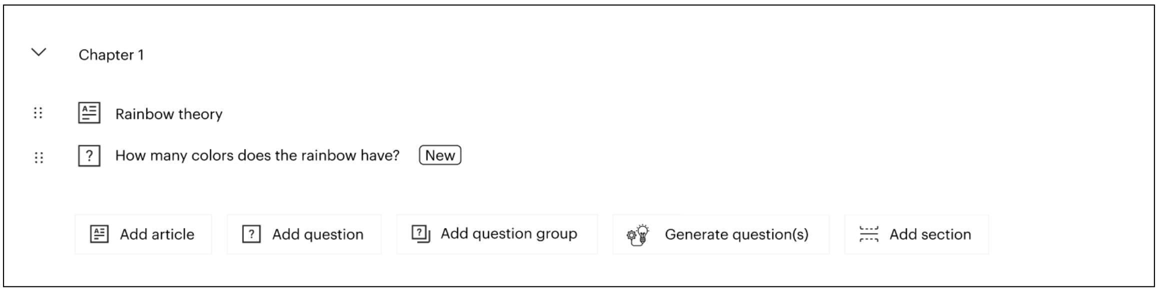
Generate question(s) from article 'Rainbow theory'

Question added to chapter!

ALREADY ADDED QUESTIONS

- How many colors does the rainbow have?
- What are the different colors of the rainbow?

(e) Modal third page



Chapter 1

- Rainbow theory
- How many colors does the rainbow have? New

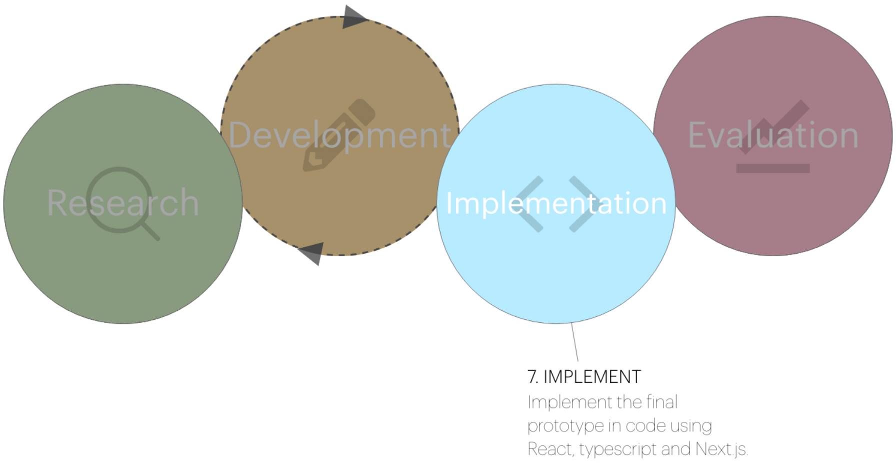
(f) Overview after question generation

Figure 4.13: Fifth iteration

Chapter 5

Implementation

In this chapter, the implementation phase - the third phase in the design process - is presented. The architecture and technology used for the final implementation is briefly described followed by the result of the implementation.



Research
Development
Implementation
Evaluation
7. IMPLEMENT
Implement the final prototype in code using React, typescript and Next.js

Figure 5.1: Our design process - the implementation phase

5.1 Architecture and technology

The implemented prototype is based on the fifth Hi-Fi prototype iteration. The architecture is therefore identical to the one presented in section 4.4.3. The prototype was implemented using TypeScript, React and Next.js, as well as a PostgreSQL database for storing and fetching data which are all presented in section 2.8. We used the code for the already existing

Sana Platform and integrated our prototype into it. Therefore some functionality such as saving questions to the chosen chapter and editing questions/alternatives was already built and could be reused by us.

5.2 Question Generation API

In order to implement the question generation we initially used the OpenAI API which is described in section 2.8.1. As input for the API we used whole article texts without formatting, and output was a question and accompanying answer alternatives; one correct and three incorrect. This format fits the "Choose the correct answer" - question type on the Sana Platform well, and was easily displayed in the question editor part of our modal. The API could be prompted for a new question as many times as needed, but depending on the temperature (see section 2.8.1) used it produced either the same question multiple times (when the temperature was low) or questions that addressed the correct topic but were not derived from the article itself (when the temperature was high).

5.2.1 Mocked API

We experimented with different values for the temperature, but we ultimately decided to mock the API before the usability tests. We decided on this because we were not satisfied with the generated questions, and we feared that our test subjects might get too fixed on the poor quality of the questions instead of focusing on the usability. For example the API generated the same question many times in a row but with slightly different phrasing, or it generated questions that had no correlation to the input text. We therefore chose an article text that would be appropriate for our tests, and wrote questions and answers related to it. The article can be seen in appendix B and the questions in appendix C. Thereafter we tweaked the tool so that every time it would normally use the OpenAI API to generate a question, it fetched one of our questions instead. This gave us less flexibility in terms of testing since the tool then only generates questions for the article we chose, but at the same time we could control the quality of the questions. The speed of the generation also increased greatly, from the process taking seconds (depending on the length of the input texts) to being instantaneous.

5.3 Result

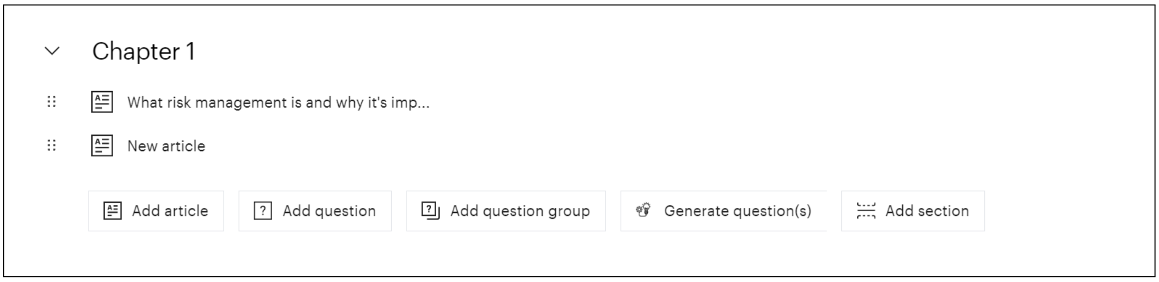
The result of the implementation is shown below in figure 5.2. Some details on the platform in the background differ between the Hi-Fi prototype and the implemented prototype. This is because we continuously merged the work of the development team at Sana into our project during implementation in order to keep our feature up to date with their changes. These changes did not affect our implementation in any way that is relevant to this thesis.

In order to limit the scope of the project we chose to not implement the user flow starting from the article editor (idea six) which was included in the Hi-Fi prototypes presented in section 4.3. Therefore the only way to open the tool in our implementation is to click on the "Generate question(s)"-button in the overview page (5.2a). We favored this flow over the other because it seemed more intuitive, since the button is visible on the overview page.

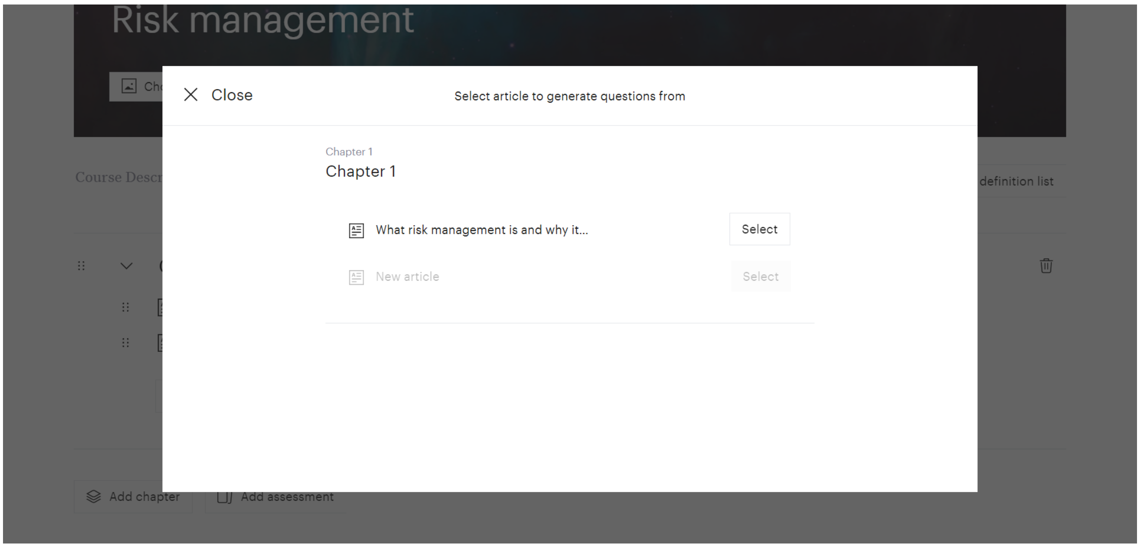
The first page of the implemented modal only differs from the iteration five prototype by a line and an additional chapter heading (5.2b), which were added to make the modal aesthetically compatible with the platform. The second page follows the prototype closely (5.2c), the only addition being a spinner with an explaining text that appears in the modal header as well as the editor area of the modal being greyed out whenever a question is being generated (5.2d). We included this feature to give the user more feedback on what is happening, and also because it matches the design of the platform.

The third page has several differences from the prototype (5.2e). Firstly, some of the headings were rewritten for better readability. Also, because of time constraints we chose not to implement functionality to delete questions directly from the modal. Therefore the trash can icons next to the questions as shown in iteration five (4.13e) were not added in the implementation.

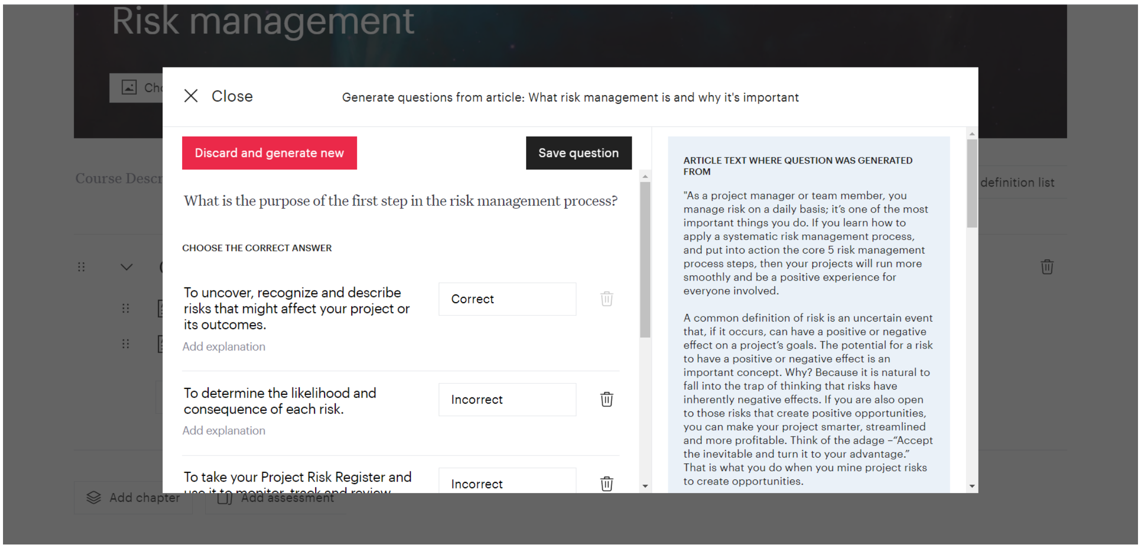
Lastly we chose not to implement new labels next to the generated questions in the overview page (5.2f). The technical complexity of this task proved to be greater than anticipated because of the structure of the platform itself. In order to stick to the project scope we therefore only implemented new labels in the third screen of the modal, and not in the overview page.



(a) Overview page



(b) Modal first page



Generate questions from article: What risk management is and why it's important

What is the purpose of the first step in the risk management process?

CHOOSE THE CORRECT ANSWER

To uncover, recognize and describe risks that might affect your project or its outcomes. — Correct

To determine the likelihood and consequence of each risk. — Incorrect

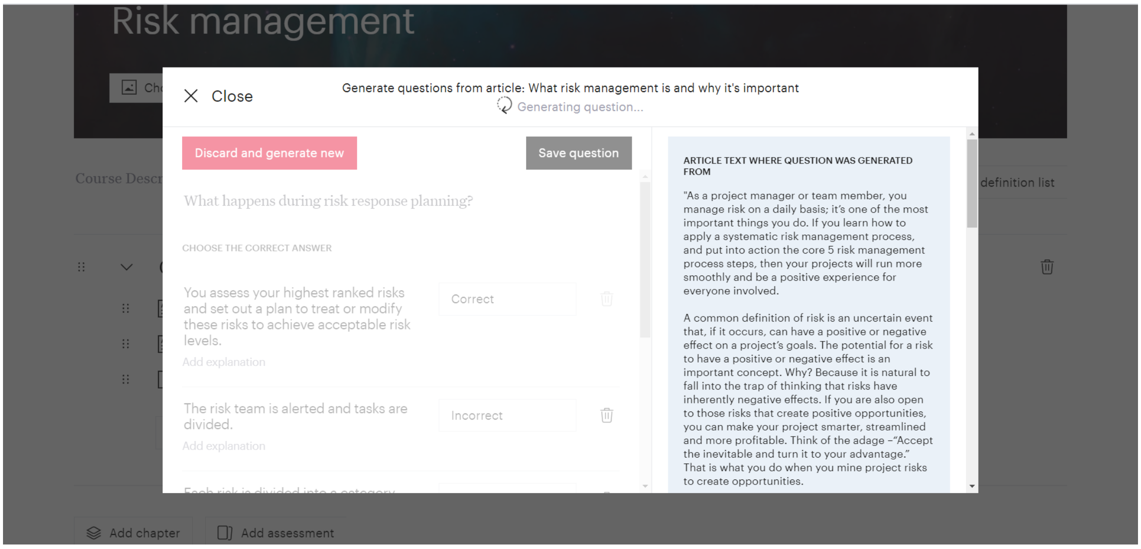
To take your Project Risk Register and — Incorrect

ARTICLE TEXT WHERE QUESTION WAS GENERATED FROM

"As a project manager or team member, you manage risk on a daily basis; it's one of the most important things you do. If you learn how to apply a systematic risk management process, and put into action the core 5 risk management process steps, then your projects will run more smoothly and be a positive experience for everyone involved.

A common definition of risk is an uncertain event that, if it occurs, can have a positive or negative effect on a project's goals. The potential for a risk to have a positive or negative effect is an important concept. Why? Because it is natural to fall into the trap of thinking that risks have inherently negative effects. If you are also open to those risks that create positive opportunities, you can make your project smarter, streamlined and more profitable. Think of the adage –"Accept the inevitable and turn it to your advantage." That is what you do when you mine project risks to create opportunities.

(c) Modal second page



Generate questions from article: What risk management is and why it's important

What happens during risk response planning?

CHOOSE THE CORRECT ANSWER

You assess your highest ranked risks and set out a plan to treat or modify these risks to achieve acceptable risk levels. — Correct

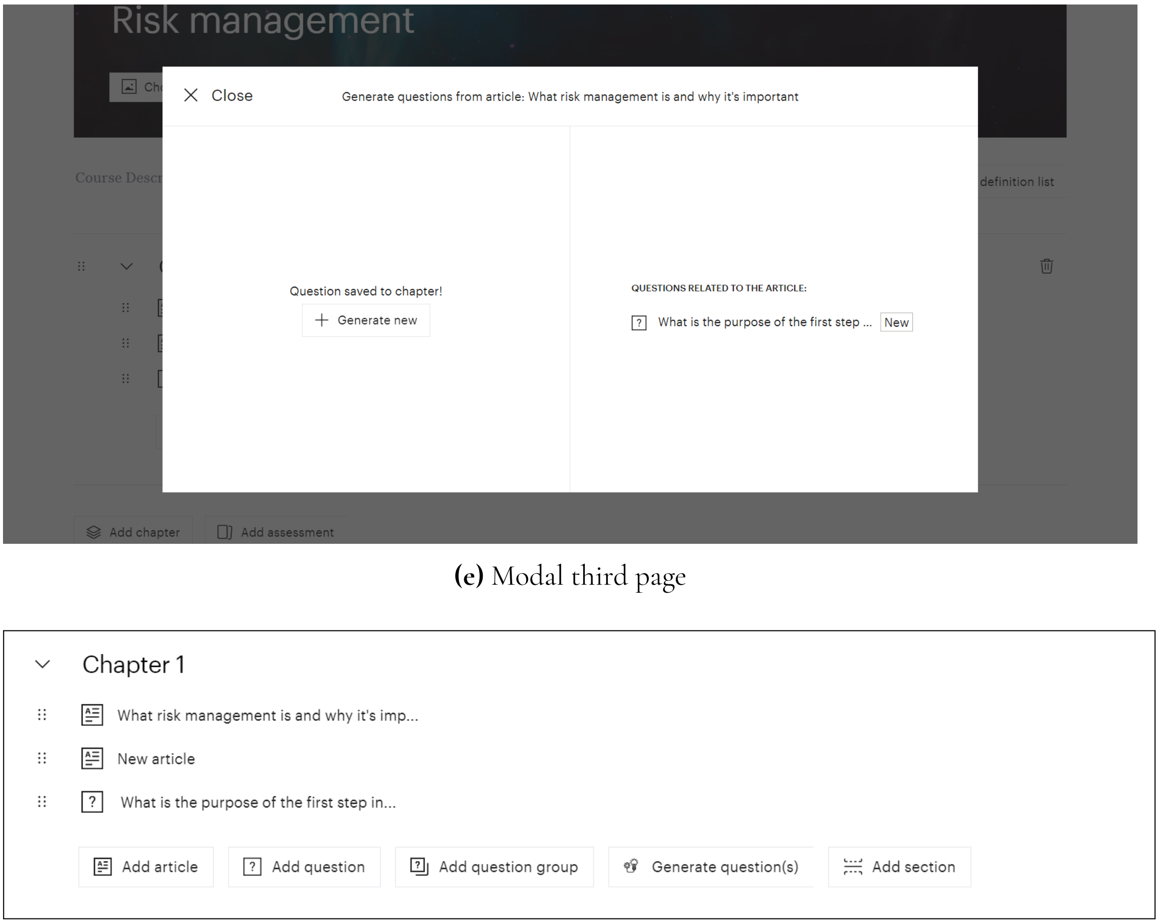
The risk team is alerted and tasks are divided. — Incorrect

ARTICLE TEXT WHERE QUESTION WAS GENERATED FROM

"As a project manager or team member, you manage risk on a daily basis; it's one of the most important things you do. If you learn how to apply a systematic risk management process, and put into action the core 5 risk management process steps, then your projects will run more smoothly and be a positive experience for everyone involved.

A common definition of risk is an uncertain event that, if it occurs, can have a positive or negative effect on a project's goals. The potential for a risk to have a positive or negative effect is an important concept. Why? Because it is natural to fall into the trap of thinking that risks have inherently negative effects. If you are also open to those risks that create positive opportunities, you can make your project smarter, streamlined and more profitable. Think of the adage –"Accept the inevitable and turn it to your advantage." That is what you do when you mine project risks to create opportunities.

(d) Modal second page while generating question



Risk management

Generate questions from article: What risk management is and why it's important

Question saved to chapter!

QUESTIONS RELATED TO THE ARTICLE:

? What is the purpose of the first step ... New

(e) Modal third page

Chapter 1

- What risk management is and why it's imp...
- New article
- ? What is the purpose of the first step in...

(f) Overview after question generation

Figure 5.2: Implemented prototype

Chapter 6

Evaluation

In this chapter, the fourth and last step in the design process is presented - the evaluation phase. In this chapter the evaluation of the implemented prototype is described through the planning, procedure and results of a usability study.

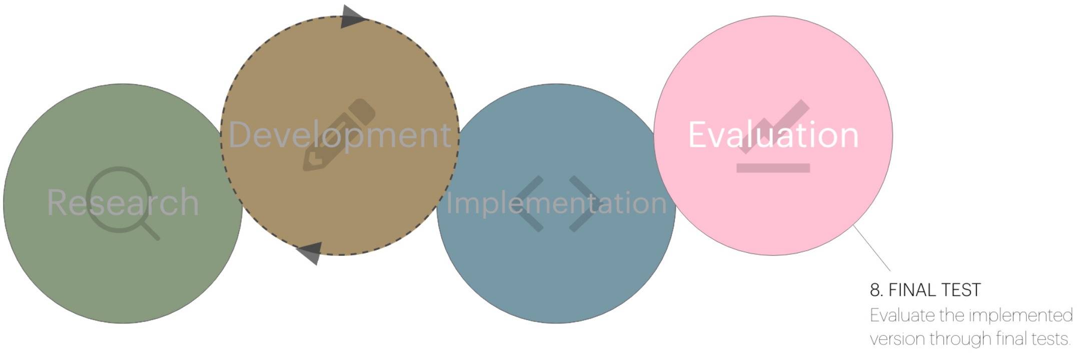


Figure 6.1: Our design process - the evaluation phase

6.1 Final Test

In this section the process of the final usability test is described. The first subsection describes who our participants were and some demographic data is presented. This is followed by a description of the procedure and which tasks and questions that where given to the participants. Lastly the findings are presented.

6.1.1 Participants

19 participants were recruited for the final tests. They were between 20-43 years old (*M* = 26.2), and they all had proficient computer skills. Four of them are employees at Sana Labs and work regularly with the authoring tool in different ways. Two of the participants were or had been users of the Sana platform's authoring tool. The rest of the participants were externals - henceforth called external testers - who had never seen or used the platform before. Five out of six of the Sana employees/users had created course content on the Sana platform, and four of the external testers had created some type of content on other, similar sites. When asked which gender the participants identified as, 14 answered woman, four man and one preferred to not disclose. This information is summarized in table 6.1 below.

Table 6.1: Demographic data for the participants in the usability tests

Age	Gender	Participant type
43	Woman	User
25	Woman	Sana employee
25	Woman	External tester
23	Prefer to not disclose	External tester
26	Man	External tester
24	Woman	External tester
26	Man	External tester
23	Woman	External tester
28	Man	Sana employee
24	Woman	External tester
20	Woman	User
26	Woman	External tester
25	Woman	External tester
23	Woman	Sana employee
26	Woman	External tester
35	Man	Sana employee
27	Woman	External tester
26	Woman	External tester
23	Woman	External tester

6.1.2 Procedure

When the participants had been recruited they were directly asked to fill in a form where we collected demographic data such as gender, age and if they had previously created content of some sort.

All of the tests were then conducted via Google Meet and were made as observation studies where the interviewee were given tasks to solve using the implemented prototype. This was followed by a questionnaire and a debriefing interview. The participants were asked to share their screens while solving the tasks so that we could observe their actions and we also screen recorded during the tests so that we could look at them again if needed afterwards.

The participants were therefore asked to sign an informed consent (see appendix A) before the tests started. The participants could then access the implemented prototype through a link to a local server being hosted on our computer [40]. We followed a script during the test sessions to ensure that the instructions were the same for all participants. Before they were given the actual tasks, they got told a short story to set them into a character. The story was as follows:

> *Imagine yourself working as a project manager at a company. You are going to train your team members in risk management and are therefore preparing some material for them. You have previously created a course called "Risk management" and added an article called "What risk management is and why it's important" that you thought was good for your team members to read. You can also see an untitled article, you can just leave that for now. Now you are ready to finalize the course and add some questions for the article so that you can really make sure that they learn the course material.*

Four tasks were given to the participants. The first two tasks were used as warm up to make the test subjects get used to the platform as a whole before testing our feature. The following tasks were given:

- **First task:** Since it was a while ago that you added the article to the course you want to start out by refreshing your memory on the content. Your first task is therefore to open up the article called "What risk management is and why it's important", skim through the text and then return to the overview page.

- **Second task:** Now you would like to add some questions relating to this article. Your second task is therefore to manually add the following question and relate it to the article:

  Question: How many steps are there in the risk management process?
  Correct answer: 5
  Wrong answers: 4, 3 and 2

- **Third task:** You now feel like it took too much time to do this manually so you decide to test out the question generation tool, which automatically generates questions for you. Your third task is therefore to use this tool to create and save two-three questions that relate to the article. Spend as much time as you'd like getting to know the tool, just tell us when you feel satisfied with the questions added.

- **Fourth task:** Upon seeing your course in the overview, you feel like there are too few questions. Your next task is therefore to again generate one-two questions. However this time you are not 100% satisfied with the generated questions and need to fix this before saving.

When the tasks were completed, a SUS questionnaire was sent to the participants for them to fill out in order to evaluate the usability of the question generation tool. They were reminded to stay in the character that was presented previously and also to specifically consider the question generation tool and not the entire platform, so that there would be no confusion. The questions that where asked and the layout of the questionnaire is presented in section 2.4. The only thing that was changed in the questions was the formulation. Instead of asking "...in the system..." the questions were re-written to "...in the tool..." to further emphasize the fact that they should evaluate the question generation tool and not the platform as a whole.

Lastly, a semi-structured interview was performed as a debriefing of the test in order to collect as much data as possible other than what we could see during the observation study at the beginning of the tests. These questions were as in the previous interviews inspired by standardized usability questions [37]. The questions that were asked were the following:

1. How was your overall impression of the question generation tool?
2. What was the best thing about the tool?
3. What was not as good about the tool?
4. Was there anything you expected to find that was not there?
5. Was anything difficult to understand about the tool?
6. How did you experience the overall feedback in the tool? (For example: feedback while loading or feedback when a question was added.)
7. (If they didn't use the discard-button and/or didn't edit the question) Did you notice the "Discard and generate new" button? If so, was there a reason why you did not try it? - OR - Did you notice that you could edit the generated question and answers? If so, was there a reason why you did not?

6.1.3 Findings

Both qualitative and quantitative data was collected during the usability tests, and in this subsection, the findings of each of the three components of the tests are presented.

Tasks

Since the first two tasks were used as warm up tasks and are not related to the usability of the question generation tool, the result of the observation study of these will not be presented in depth. However, the common thread that we observed during these two tasks was the fact that many of the external participants had issues with relating the article to the question in task two. The same participants had some trouble understanding whether the question had been saved or not since there is no save button.

While the third and fourth tasks were being performed by the participants we observed that one of the biggest issues was the fact that the platform overall had a slight learning curve for those who had not been in contact with it before. For example, it was not clear to them that the text field where the generated question was displayed was editable, while the regular users of the platform did not seem to have as much trouble understanding this. Many of our observations were brought up and discussed during the interview and debriefing session at the end of the tests. Therefore we will not present our observations further, these will instead be covered in the findings from the interview in section 6.1.3 below.

System Usability Scale Score

The mean SUS score for the implemented tool was calculated according to the procedure described in section 2.4.1. The result of this calculation is 94.1 which refers to above *excellent* according to the adjective scale described in the same section. This result is shown in figure 6.2. The standard deviation (SD) of the same score was calculated to 5.6, which is very low and tells us that almost all of the participants answered the same on each of the questions asked in the SUS questionnaire.

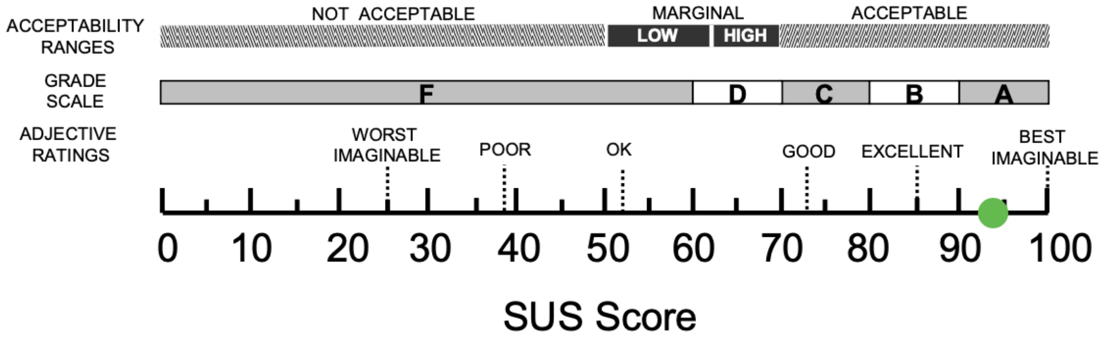


Figure 6.2: Our SUS score, 94.1, displayed as the green dot on the scale.

Interview

The following paragraphs are the participants' answers to the interview questions. Many participants had comments that are related to the design of the platform as a whole, and not specifically to our tool. We therefore chose to exclude these comments since they are not relevant for the result of this interview.

How was your overall impression of the question generation tool?
Overall the participants had a positive attitude towards the tool. Many expressed that it was intuitive, had a stylish design and was straightforward to use. Some particularly expressed that it was going to be a time saving tool and that they probably would not have to get to know the article content straight away since the tool shows the article text at the same time as the generated question. They also said it could be of great use since it is troublesome to manually write many questions for an article, and the tool can generate ideas on what type of questions would be suitable even if they decide to not use the question exactly like it was generated.

What was the best thing about the tool?
Apart from what was already told in the answers to the first question, many of the participants specifically mentioned the article text being shown in the right column as the best thing about the tool, since it made it easy to verify the validity of the generated question and also edit the question and alternatives accordingly if need be. One participant, that is a current user of the authoring tool, explicitly said that they really liked that the article text was presented in a blue box since this made the division of the two sides more obvious and the article text more apparent.

One participant appreciated the option to generate several questions, related to the same article, directly after one another, without having to exit to the main screen in between. Another person liked that they could edit the questions directly after generating them, instead of having to open them one by one to edit afterwards, as they had to do in task two when manually adding a question. Another person also mentioned that the discard-button felt responsive since it generates a new question straight away when clicked.

Two people mentioned that they liked how the modal was divided into three parts since it made the process straightforward, and because they got to focus on one question at a time the process was clear and precise. One of them also appreciated the fact that the clean design minimizes the risk of doing something wrong.

What was not as good about the tool?

Several participants had opinions regarding the article text in the right column: they thought it was too long and therefore that it was difficult to find the relevant section(s) when checking the question against the article. Several solutions to this problem were suggested by the participants: highlighting the relevant part(s), only showing the relevant part(s) or a search function. It was also mentioned that the article text could be collapsible, so that it is only shown when needed. One participant said that they did not like that the article text lost its original formatting. Another participant also pointed out that the design of the article box does not follow the design of the rest of the platform.

Another issue that was brought up by one participant was that saving is automatic on the platform, but must be done manually in our modal. The participant thought this was a little confusing. It was also mentioned by another participant that the modal felt a bit cramped so they had to scroll a lot. This participant did however perform the test on a large screen instead of on a laptop.

Was there anything you expected to find that was not there?

Some of the participants expressed that they would like to have a preview of how the questions would look for the end users. Another wish expressed by many of the participants was to have the option to open previously generated questions again without having to exit the modal. In other words they wanted to be able to click on a question listed in the third page of the modal and then have it displayed again in the second page. Another opinion expressed by one participant was to have the possibility to go back and forth between generated questions before they were saved to chapter, while still on the second step of the modal. To clarify, this means that they would like to move between the questions without having to discard or save, in order to get a couple of questions to choose from.

Several participants also expected to be able to configure the questions to a greater extent, such as choosing the difficulty of the question or picking a specific part of the article that the question should cover. One person also said that they expected the generator to generate additional alternatives when they clicked the “Add option”-button instead of just getting an empty text field. Lastly one participant said that they wanted to know how many questions the tool could generate and another would like to see the confidence level of the question.

Was anything difficult to understand about the tool?

Most of the participants did not find anything difficult but a couple of things were mentioned as being confusing. For example, one participant was a bit confused regarding the question type: in the platform you can choose between two different types whereas in the modal all the questions are automatically set to one type. There was also one participant who explicitly mentioned the "Choose the correct answer"-text field above the alternatives as being confusing since they did not understand the meaning of this text. They first thought it had popped up since they changed something in the question. As a solution to this, the same person said that they either would have wanted to not have this text at all, or that the text area instead could be used as a warning if the question was changed and then display something like: "Remember to check that the alternatives are still valid!". Lastly, one participant was a bit confused regarding why explanations for the alternatives were not generated but shown as an empty text field, and also why they could not choose the empty article in the first step of the modal.

How did you experience the overall feedback in the tool? (For example: feedback while loading or feedback when a question was added.)
The participants overall thought that feedback was presented from the system whenever needed. Many of them again expressed that the process of generating questions was even more clear and straightforward than they thought it would be. For example many participants noticed the spinning icon and text at the top header while loading, as well as the grayed out question area and also the new labels on the last step of the modal. This feedback was well received by many of the participants. Two people thought it was great that the correct answer alternative always was displayed at the top, since this made it clear that a correct answer had been generated. Many of the participants also mentioned that it was great that the tool was responsive, and that there were no delays without any feedback of what was happening. They also said it was nice to not have to confirm actions a second time, for example when clicking the discard button, because they thought that could have been annoying in the long run.

One thing that was mentioned as not being as clear was when a question was added to the list on the third step. It was not clear to everyone that the question was actually saved, as well as where in the chapter this question had been placed. They all seemed to notice the 'New'-label next to the questions, but thought it would be great with even more feedback. As an example that one of the participants thought of, the question that was newly added could be highlighted. Some people also mentioned that they would have wanted to 'save and close' when closing the modal because they were not entirely sure that the questions had actually been added to the chapter even though they had just saved a question at a time. It was described as: "the last step felt like a shopping cart, and I wanted to save it before closing the modal". On the other hand, one of the participants thought it was clearer that a question was saved when using the question generator tool rather than when using the regular 'add question' tool, where everything is saved automatically. One participant also wanted some signifier on the overview page to again get a better overview of which questions had been generated.

The "Add option"-button in step two was missing some feedback according to two of the participants, as they did not see that anything had happened until they clicked it a second time.

Lastly one participant suggested that some form of tutorial, pop-ups or tool-tips on various parts of the tool could be included to give the user more guidance, at least the first time they use the tool.

Did you notice the 'Discard and generate new' button? If so, was there a reason why you did not try it? - AND/OR - Did you notice that you could edit the generated question and answers? If so, was there a reason why you did not?

Overall the participants who had never used the platform before had issues understanding that the questions and alternatives were editable, whilst it was more intuitive for the users who had used the platform before.

Regarding the discard button, several people said that they chose not to use it because they did not know the article well enough to judge whether the questions were good or bad. Most of them did however understand the button and would have used it otherwise. Another participant did not understand that they should/were allowed to use the button for these tasks, but explained that they had understood what the button does. Three participants explicitly said that the red color of the button was a little scary to them, and one of these three also thought that the button would only remove the answer alternatives. The other two were unsure of what exactly was going to be removed, but explained that this might be because they thought the word "discard" was unfamiliar as well. They thought that "Change question" instead of "Discard..." would be an improvement.

Chapter 7

Discussion

In this chapter, the discussion is presented. The first section covers the limitations that arose during this project. Thereafter, a discussion and evaluation regarding the design process is presented and lastly a section where future work and possibilities are brought up and discussed.

7.1 Limitations

One unforeseen factor that limited our project to some extent was the API we used for generating questions. We had expected it to generate questions of higher quality. We made the decision to use a mocked API instead as described in section 5.2.1, in order to keep the testers focused on the usability and to have as much control over the tests as possible. Since this thesis is only focusing on the UX aspect of the question generation, we did not look into the API before the implementation phase as we assumed we could just use it as is. If we had tested the API and seen that the generated questions were unsatisfactory at an earlier stage, we might have had time to improve it or find another solution. That in turn could have led to better tests, where the testers could for example write their own article text to get a more complete feel of the tool.

Also, due to the Covid-19-pandemic all of our tests and interviews were conducted remotely via Google Meet. It worked well, and we discovered that it was easier to recruit people to this kind of test since they can participate from home. However when testing our final prototype our test subjects also had to access the prototype via a local tunnel to our local server being hosted on our computer. As a consequence, fetching data from the database and rendering information in the browser was slower than usual, and certain parts of the prototype therefore felt unresponsive to the users. It also caused unfortunate interruptions in some of our tests by displaying an error message and forcing the test subject to reload the browser and restart the task they were on. As a result we received a few complaints on the speed and responsivity of the prototype, but these problems would not have occurred in normal circumstances when the tests would have been conducted in person. We therefore chose to disregard this feedback.

7.2 The Design Process

In the following subsections, the design process and its four phases are discussed and critically evaluated in order to better understand what could have been done differently or improved.

7.2.1 Research Phase

Several techniques were used in the research phase to gather relevant data and background information needed to carry out this project. Looking at similar products proved to be valuable as we were able to see different solutions to the question generation problem and evaluate these against each other. We used the findings described in section 3.1.1 to make design decisions later on in the process.

Brainstorming

The brainstorming session was only performed by the two of us. As the goal of a brainstorming session is to produce quantity and not quality when it comes to ideas, we could have benefited from including more people in our session. Even so, we felt like the brainstorming session was a good opportunity for us to reach a common understanding of the project and also uncover which areas we needed to understand better in order to start developing design solutions.

Bias in interviewees

The mean age of the interviewees we recruited for our user interviews was 26.8 years which is quite young. People of different ages usually have different needs and different perceptions of what is difficult, especially when it comes to more technical applications. As was discovered in a study regarding web site content [6], demographics such as age influence user perceptions and preferences. By not having more age diversity we therefore might have missed out on important insights. Additionally, four out of six of the interviewees were employees at Sana Labs. These people naturally reason differently about the product than the users of it. The employees might be affected by their ambition for the product and can therefore have identified user needs and use cases that do not reflect the reality or the actual needs of the users of the product. We had this in mind during the next phases of the project, and therefore chose to put more focus on the input given by the users.

Too much focus on the current design

Since one of our research topics was to develop a feature that is compatible with the platform it was natural that we spent some time getting to know it, for example by going through Nielsen's 10 usability heuristics. However, we might have limited ourselves and the creative process by getting too familiar with the current design. One example is that on the platform the process of adding new content or components always starts with the click of a button on the overview page; add article, add question, add section and so on. This may have caused us to fixate on this type of work flow too early in the process, without exploring other alternatives. In our interviews we even asked the following question: *Imagine a function that, when you click on a* ***button*** *would generate a number of questions that are linked to a given article. In the existing system, where do you feel that a good place for this* ***button*** *would be?* This phrasing naturally led the interviewees to think about a solution involving a button. A few people did however give alternative solutions despite the question being specific about a button, but a more open question could perhaps have given us even more ideas.

7.2.2 Development Phase

In the beginning of the development phase we designed different solutions which we presented to two of Sana Labs' employees whom we consulted and made decisions together with. Some of these early design decisions later turned out to not be as thought through as they should have been. As an example, one of the participants during the first interview, gave us the idea that a question could be generated automatically when an article text was long enough to generate one from. This idea was something that we never went forward with, partly due to that we focused too much on the idea that it should be a button on the overview page that solves the problem. Also, we discussed the idea briefly and decided that this would be irritating for the user rather than helpful and could provoke impatience, anger, or displeasure using the tool. The project could have had different results if we had been more open minded towards different ways of solving the problem. What we instead should have done is to conduct smaller test sessions or have focus groups discuss each iteration before deciding on what to go forward with.

Another decision we made early on, between the first and second iteration of the Lo-Fi prototype, was to add the question editor as a second step in the modal and therefore also remove the selection of how many questions that should be generated. These decisions were taken after we got a better understanding of the limitations and capabilities of the API, for example that it would never run out of new questions to generate and therefore it does not make sense to calculate the maximum number of possible questions. This is something that we could have researched more during the first phase of the project, in order to have a better understanding of how the API would be able to function. However, since the implementation of this API was not in the scope of what our focus would be on, this is probably nothing that we could have done without extending the scope to also implement the API ourselves.

These decisions are also good examples of decisions that should have been based on some sort of data instead of just us making the decisions during a meeting. However, this decision turned out to be a good one, since we noticed that it was much appreciated during the Hi-Fi prototype test sessions.

Balancing the current design with new ideas

Another thing that turned out to be difficult, was to balance the current platform with the new design of our tool. Since we had not been a part of the decision making regarding the platform design from the beginning it was hard to know what design to follow strictly and what we could change while still making the two parts fit well together. As a consequence, some inconsistencies in relation to the platform design were introduced in our tool. One example is that we chose to have a modal displayed when clicking the "generate questions"-button on the overview page, while this is not coherent with the functionality of the other buttons on the overview page. Another example is that even though almost everything on the platform is saved automatically, we still decided to have a save button in our modal. This is a consequence of the fact that we chose to have the modal as a solution, since it would be difficult to make it clear that something is saved from a modal without having the user explicitly press a save button.

7.2.3 Implementation Phase

As described in section 5.3 we chose to only move forward with and implement one of the user flows from our prototype when transitioning from the development phase into the implementation phase. This helped us stick to the scope of the project. The decision to move forward with this specific flow was made by us and our supervisor at the company as we thought it was the most useful flow of the two. However this decision should have been backed up by some other means, for example a user test or survey to ensure that we did in fact pick the most valuable flow.

During implementation we tried to build a tool that resembled the final prototype as closely as possible. However, we decided to not implement certain components which is also described in section 5.3. These decisions were made purely because of time and scope restrictions. The only other major design choice that was made was to include the spinner, text and greyed out area while the generated question is loading. It might not be the best practice to change the design during implementation, but we deemed it necessary in this case. This is because it was not apparent that a new question was being generated - for example when clicking on the discard button - since several seconds could elapse before the question appeared. We therefore wanted to give the user confirmation that their request to generate a new question was being processed.

After the unforeseen change to a mocked API, the process of generating a new question became instantaneous. This removed the feedback problem described above, but created another; since the change to a new question is so fast with the mocked API, it was sometimes difficult to spot that a new question had been generated when clicking on the discard button. We therefore decided to keep the spinner, text and greyed out area and added a delay of 1.5 seconds before displaying the new question so that the user has enough time to register that something is happening when generating a new question. This improved the tool in regards to Nielsen's first heuristic - visibility of system status - from section 2.3.

7.2.4 Evaluation Phase

During the evaluation phase of this project, the tests that were conducted were based on a mocked API as previously explained. This could potentially have affected the result of these tests since the UX of the tool could have been perceived differently if the implemented API would have been used instead. It is important to have this in mind since this could be a contributing factor to the result we have concluded and further analyzed in this discussion.

Gender bias

The majority (73.7 %) of the recruited test subjects for the testing of the final prototype identified as women. We have previously discussed that demographics influence a user's perception and preferences, and experiments done by Colman and Moss[33] additionally found a significant tendency for respondents to favor designs made by members of their own sex. Since both of the authors and designers of this project identify as women we believe that we could have benefited from recruiting more participants identifying as men as we might have gotten more constructive feedback had we done so.

Differences between the test subjects

The test users that we recruited can be divided into three groups:

1. Employees at Sana Labs

2. Existing users of the Sana Platform

3. External testers

These three groups naturally had somewhat different perceptions of our prototype partly because of their familiarity (or lack thereof) with the product, but also due to several other factors. An employee at Sana Labs might for example appreciate a certain design solution because they are biased by their ambitions and/or goals for the company, whereas a user might find the same solution flawed as they see no use for it in their normal work flow. For this reason, we could have structured the testing of our final prototype differently. For example, it might have been interesting to compile the test results for the three groups independently and compared the results against each other. That way biases might have become more clear. However the standard deviation for our SUS-score presented in section 6.1.3 was only 5.6, meaning that there was not a big variance in the answers given by the participants. We can therefore assume that the users from the different groups had the overall same experience using our tool and that looking at the groups' results individually probably would not have provided any interesting discoveries.

Another strategy we could have opted for regarding our final tests was to only test on users of the Sana Platform. These are the people who would actually use our tool in their everyday work and one can argue that their feedback should be the only one that matters. However, this group was by far the most difficult to recruit from as we were reliant on Sana Labs to provide their contact information. Secondly, Sana Labs get new customers continually. Therefore, test subjects from the third group can also provide important information, for example regarding the learning curve of the tool and platform as they used them both for the first time during their tests. On the other hand, a drawback of using subjects from the third group is that several of them did not understand the distinction between the platform and our tool. This became evident during some of the interviews since they discussed components on the platform when for example asked what was good/not as good about the tool. We disregarded these comments during their interviews, but it might have influenced some of their answers during the SUS-questionnaire as well.

Editing content & article choice

During the tests it became apparent that most of the subjects from the third group did not realise that they could edit the generated question and answer alternatives whereas most of the subjects from the other two groups did. We assume that it is due to the subjects in group three's unfamiliarity with the platform, but nevertheless it is relevant feedback as we want new users to understand our tool as well. Some of the subjects who did not understand that they could edit did however explain during their interviews that they did not think to even try to edit since they did not know the content of the article well enough. This lead us to another potential flaw of our test setup: When we designed task three and four we assumed that our test subjects would spend more time reading and getting to know the article in task one than they actually did. Therefore, a common behaviour when executing task three - and task four when necessary - was that that the test subjects rushed through the process, trusting the tool blindly and not paying much attention to the wording of the question and answer alternatives. This might also have contributed to the fact that so many did not understand that they could edit the question and alternatives. Had they been more familiar with the article they might have been more inclined to change the wordings and therefore at least checked whether the generated content was editable. The article that was used in the tests was chosen because we thought it was academic enough for our purpose but still covering a topic that most people can relate to. In retrospect we see that we could have chosen a shorter and/or less complex article as it would have been easier for the test subjects to familiarize themselves with it. The optimal solution would of course have been to let the test subjects write the article themselves, that way we could ensure that they knew the content. This was not doable because we opted to conduct the tests with the mocked API as described in section 5.2.1.

7.3 Future Work

An interesting aspect to evaluate during future work is the comparison between the question generation tool using the mocked API and the OpenAI API. This could be evaluated by conducting a comparison test. We think that this could provide a clear view of what should be improved regarding the feedback of the UI when changing from the mocked API to the implemented API in the future. A/B testing could be used in order to conduct such tests, and could also provide insights on how to optimize the UX of the tool [27]. Another interesting aspect that could be evaluated using this type of comparison tests is to compare the user flow of our implemented tool with the user flow of a "generate question"-icon implemented inside the article editor. This was the sixth idea that was generated during the first part of the development phase and is described in more detail in section 4.2. This could be a great way to see which of these user flows is best received.

7.3.1 New/improved functionality

During the final tests of this project, issues and ideas for potential improvements were brought to our attention. These were both new ideas, but also things that we had already thought of during earlier stages of the project. As we have already mentioned previously in the report, these ideas were not implemented or further developed due to restrictions and limitations that had to be made for different reasons.

Responsivity

One detail that we noticed, was that the modal was not responsive to screen size, meaning that when the participants had larger screens (larger than 13 inch screens), the modal looked and felt much smaller and more restrictive in relation to their big screen than when viewed on a 13 inch screen which we used during the implementation of the tool. This should be fixed during future implementations since we believe that it is a small change that would make a huge difference in the UX for our tool.

Article Text

In our implementation the users must choose an entire article to generate questions from, but functionality allowing the users to also choose a specific part of the article to generate questions from should be explored and added. Based on feedback from the test participants as well as the UX of AI guidelines presented in section 1.3.3, such functionality would keep the users in control and make them feel that the tool is even more powerful.

It should also be explored whether only the specific part of the article where the question is generated from should be displayed in the second page of the modal or if the whole article text should be shown but with the specific part highlighted in some way. This would make it much easier for the user to quickly recognize whether the generated question and alternatives are well written without having to scroll through the entire article text. It would also contribute to explaining the results for the users, which is one of the UX of AI guidelines presented in section 1.3.3.

Another idea is to add the possibility to collapse the article text on the second step of the modal in order to make it feel less compact. It will also make a huge difference in clarity if the article text keeps its formatting when displayed.

Configuration of questions

To keep the user in control and improve the UX even further, functionality that would make it possible to generate different types of questions and also to choose the degree of difficulty of a question should be added. Another detail that could improve the experience for the users would be to generate another answer alternative when the "Add option"-button on the second step of the modal is pressed. This would ease the process for the users even more.

Displaying the confidence of the AI

It should also be explored whether it would be possible to display the level of certainty a question was generated with. The level of certainty is an indication of how good the question is and hence how much it can be trusted. This would express the limitations of the AI and that way set the right expectations as well as communicating the confidence of the results which are two of the UX of AI guidelines described in section 1.3.3. By showing the level of certainty one might also reduce the problem we had during our tests where the test subjects trusted the tool blindly and hardly read the questions. Hopefully users might be more inclined to proof-read the generated content if they see the limitations of the AI. This will also help decrease the probability of creating faulty or poor content.

Improved feedback

By adding tool-tips on all buttons in the tool, the overall feedback would be improved a lot and also the design would be more cohesive with the rest of the platform. The input fields where the generated question and alternatives are displayed should also be re-designed in order to make it more obvious that the fields are actually editable since this was not clear to many of the test participants.

If a user re-writes something in a generated question, a warning should be displayed telling the user to check the alternatives since they might no longer be accurate. This would probably make the user trust the tool more and also reduce the chance of making errors.

Another suggestion for improving the feedback would be to change the close-button, especially on the third step of the modal, to "Save and Close" or something similar in order to ensure the user that the added questions will been saved. This idea came from one of the test participants after they explained that the last step felt like a shopping cart as explained in the test results in section 6.1.3.

Lastly the new labels that is displayed next to a newly added question on the third step of the modal should also be displayed next to the questions on the course overview page as long as the question is not opened again. We believe that this would provide the users with even better feedback and clarify where the saved questions have been placed.

To have discard functionality or not?

When the implemented API for question generation is in place and it is known how it will actually function, we recommend that the "Discard and generate new"-button on the second step of the modal is re-evaluated. It should be obvious for the user whether this button discards the question fully or if it is possible to retrieve the question again if removed by accident. This would improve the error handling in the tool and thereby improve the UX. Therefore, a suggestion is to research and add functionality to be able to move back and forth between already discarded questions as well as display how many questions the tool is able to generate from the given text. This would make the tool even more useful for the user. Of course this idea is only applicable if the discarded questions are retrievable after being discarded, again depending on how the API will function.

Make better use of the last step of the modal

The possibility of opening already added questions directly from the third step in the modal should be explored further. We believe that this could make the users feel even more in control, which again is an important aspect of the UX of AI guidelines presented in section 1.3.3. Also, this would make the third step of the modal more interactive and by that more useful.

Chapter 8
Conclusion

In this chapter, the conclusion of this thesis is presented. The aim was to design and implement a tool that both solves the issue regarding question generation and that is possible to integrate into the current Sana platform. Based on the results from the various phases of this project we answer our research questions presented in section 1.2

Which user needs should be fulfilled through the use of a question generation tool?
Our research phase and final test results uncovered and confirmed several needs that users of a question generation tool have. The most important was to be in control of the configuration of the generated questions, meaning that the tool should provide example questions that the user gets to edit if needed. Another important need that was identified was to simultaneously see the generated question and the part of the content where it originated from.

Which UX principles are important when designing a solution for AI-augmented question generation?
Our final test results and future work discussion showed that the five most recurring principles of the ones listed in section 1.3.3 were : *start with the user*, *set the right expectations*, *explain the results*, *communicate your confidence* and *keep the user in control*. Thereby we can conclude that apart from the five quality components listed in section 2.1 - which should always be in place - these are the five most important UX of AI-principles for an AI-augmented question generation feature.

How should this type of solution be developed and implemented to be compatible with the current design of the Sana platform while still complying with UX best practices?
As previously discussed, one of the most challenging parts of this project was to balance the current platform design with the new feature design. We noticed that it was difficult not to introduce any inconsistencies into the new feature's design. However, based on the positive results of our final tests, we can conclude that some inconsistencies are not necessarily bad

for the new design. It can still be compatible with the current design if it is obvious why the inconsistency was necessary. Therefore the method used throughout this project proved to be adequate for developing and implementing a solution for AI-augmented question generation. Nevertheless we also acknowledge the importance of keeping an open mind at the beginning of the process so that other possible design solutions are not overlooked.

References

[1] Mattias Arvola. *Interaktionsdesign och UX: om att skapa en god användarupplevelse*. Studentlitteratur AB, 2014.

[2] Aaron Bangor, Philip Kortum, and James Miller. Determining what individual sus scores mean: Adding an adjective rating scale. *Journal of usability studies*, 4(3):114–123, 2009.

[3] John Brooke. Sus: A quick and dirty usability scale, 1996. https://hell.meiert.org/core/pdf/sus.pdf.

[4] John Brooke. Sus - a quick and dirty usability scale. 2006.

[5] Cerego. Cerego, 2020. https://www.cerego.com/.

[6] Ann Chadwick-Dias, Donna Tedesco, and Thomas Tullis. Demographic differences in preferred web site content. 01 2004.

[7] Graham Dove, Kim Halskov, Jodi Forlizzi, and John Zimmerman. Ux design innovation: Challenges for working with machine learning as a design material. In *Proceedings of the 2017 chi conference on human factors in computing systems*, pages 278–288, 2017.

[8] Enki Labs Inc. Enki, 2020. https://www.enki.com/.

[9] Facebook Inc. React, 2020. https://reactjs.org/.

[10] Ali Fardinpour, Mir Mohsen Pedram, and Martha Burkle. Intelligent learning management systems: Definition, features and measurement of intelligence. *International Journal of Distance Education Technologies (IJDET)*, 12(4):19–31, 2014.

[11] Figma. Figma, 2020. https://www.figma.com/.

[12] Flexudy. Flexudy, 2020. https://flexudy.com/.

[13] International Organization for Standardization. Ergonomics of human-system interaction — part 210: Human-centred design for interactive systems. Technical report, 2019.

[14] Google. How to video conference with google meet, 2020. https://apps.google.com/meet/how-it-works/.

[15] Google Design. People + ai guidebook, 2019. https://pair.withgoogle.com/guidebook/.

[16] Google Design. People + ai research, 2020. https://design.google/library/ai/.

[17] John D. Gould and Clayton Lewis. Designing for usability: Key principles and what designers think. *Commun. ACM*, 28(3):300–311, March 1985.

[18] Sabine Graf and K Kinshuk. Providing adaptive courses in learning management systems with respect to learning styles. In *E-Learn: World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education*, pages 2576–2583. Association for the Advancement of Computing in Education (AACE), 2007.

[19] Greg Brockman, Mira Murati, Peter Welinder and OpenAI. Openai api, 2020. https://openai.com/blog/openai-api/.

[20] Moodle HQ. Moodle, 2020. https://moodle.org/.

[21] Blackboard Inc. Blackboard, 2020. https://www.blackboard.com/.

[22] Interaction Design Foundation. Brainstorming, 2020. https://www.interaction-design.org/literature/topics/brainstorming.

[23] Interaction Design Foundation. Prototyping, 2020. https://www.interaction-design.org/literature/topics/prototyping.

[24] Interaction Design Foundation. User interface design, 2020. https://www.interaction-design.org/literature/topics/ui-design.

[25] itslearning. It's learning, 2020. https://itslearning.com/se/.

[26] Jakob Nielsen. 10 usability heuristics for user interface design, 1994. https://www.nngroup.com/articles/ten-usability-heuristics/.

[27] Jakob Nielsen. Putting a/b testing in its place, 2005. https://www.nngroup.com/articles/putting-ab-testing-in-its-place/.

[28] Jakob Nielsen. Thinking aloud: The #1 usability tool, 2012. https://www.nngroup.com/articles/thinking-aloud-the-1-usability-tool/.

[29] Jakob Nielsen. Usability 101: Introduction to usability, 2012. https://www.nngroup.com/articles/usability-101-introduction-to-usability/.

[30] Kara Pernice. User interviews: How, when, and why to conduct them, 2018. https://www.nngroup.com/articles/user-interviews/.

[31] Lennart Ziburski. The ux of ai, 2020. https://uxofai.com/.

[32] Microsoft. Typescript, 2020. https://www.typescriptlang.org/.

[33] Gloria Moss and Andrew Colman. Choices and preferences: Experiments on gender differences. *The Journal of Brand Management*, 9:89–98, 11 2001.

[34] Don Norman. *The design of everyday things*. Basic books, 2002.

[35] Ulrich Paschen, Christine Pitt, and Jan Kietzmann. Artificial intelligence: Building blocks and an innovation typology. *Business Horizons*, 63(2):147 – 155, 2020. ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING.

[36] Jennifer Preece, Yvonne Rogers, and Helen Sharp. *Interaktionsdesign: bortom människa-dator-interaktion*. Studentlitteratur, 2016.

[37] Qualaroo. 29 questions to consider asking users during prototype testing, 2019. https://qualaroo.com/blog/29-questions-to-ask-users-during-prototype-testing/.

[38] Quillionz. Quillionz, 2020. https://www.quillionz.com/.

[39] R Radha, K Mahalakshmi, V Sathiskumar, and Ar Saravanakumar. E-learning during lockdown of covid-19 pandemic: A global perspective. *International Journal of Control and Automation*, 13(4):1088–1099, 2020.

[40] Roman Shtylman. Expose yourself to the world, 2019. https://localtunnel.github.io/www/.

[41] Sana Labs AB. Sana labs homepage, 2016. https://www.sanalabs.com/.

[42] Jeff Sauro. *A practical guide to the system usability scale: Background, benchmarks & best practices*. Measuring Usability LLC, 2011.

[43] The PostgreSQL Global Development Group. Postgresql, 2020. https://www.postgresql.org/.

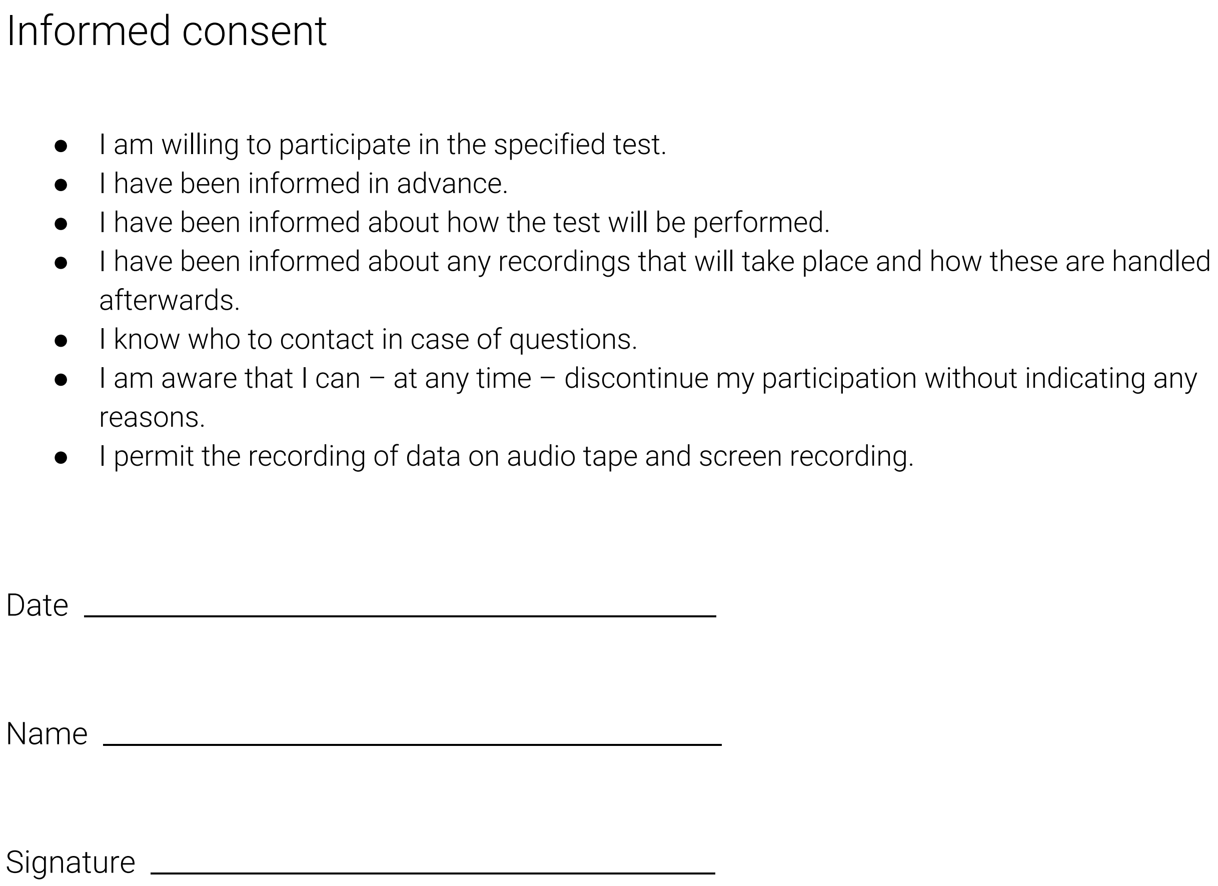
[44] Vercel, Inc. Next.js, 2020. https://nextjs.org/.

[45] Qian Yang, Alex Scuito, John Zimmerman, Jodi Forlizzi, and Aaron Steinfeld. Investigating how experienced ux designers effectively work with machine learning. In *Proceedings of the 2018 Designing Interactive Systems Conference*, pages 585–596, 2018.

[46] Panagiotis Zaharias and Christopher Pappas. Quality management of learning management systems: A user experience perspective. *Current Issues in Emerging eLearning*, 3(1):5, 2016.

Appendices

Appendix A
Informed consent



Informed consent

- I am willing to participate in the specified test.
- I have been informed in advance.
- I have been informed about how the test will be performed.
- I have been informed about any recordings that will take place and how these are handled afterwards.
- I know who to contact in case of questions.
- I am aware that I can – at any time – discontinue my participation without indicating any reasons.
- I permit the recording of data on audio tape and screen recording.

Date ______________________________

Name ______________________________

Signature ___________________________

Appendix B
Mocked article text

Source: https://continuingprofessionaldevelopment.org/risk-management-steps-in-risk-management-process/

As a project manager or team member, you manage risk on a daily basis; it's one of the most important things you do. If you learn how to apply a systematic risk management process, and put into action the core 5 risk management process steps, then your projects will run more smoothly and be a positive experience for everyone involved.

A common definition of risk is an uncertain event that, if it occurs, can have a positive or negative effect on a project's goals. The potential for a risk to have a positive or negative effect is an important concept. Why? Because it is natural to fall into the trap of thinking that risks have inherently negative effects. If you are also open to those risks that create positive opportunities, you can make your project smarter, streamlined and more profitable. Think of the adage –"Accept the inevitable and turn it to your advantage." That is what you do when you mine project risks to create opportunities.

Uncertainty is at the heart of risk. You may be unsure if an event is likely to occur or not. Also, you may be uncertain what its consequences would be if it did occur. Likelihood – the probability of an event occurring, and consequence – the impact or outcome of an event, are the two components that characterize the magnitude of the risk.

All risk management processes follow the same basic steps, although sometimes different jargon is used to describe these steps. Together these 5 risk management process steps combine to deliver a simple and effective risk management process.

Step 1: Identify the Risk. *You and your team uncover, recognize and describe risks that might affect your project or its outcomes. There are a number of techniques you can use to find project risks. During this step you start to prepare your Project Risk Register.*

Step 2: Analyze the risk. *Once risks are identified you determine the likelihood and consequence of each risk. You develop an understanding of the nature of the risk and its potential to affect project goals and objectives. This information is also input to your Project Risk Register.*

Step 3: Evaluate or Rank the Risk. *You evaluate or rank the risk by determining the risk magnitude, which is the combination of likelihood and consequence. You make decisions about whether the risk is acceptable or whether it is serious enough to warrant treatment. These risk rankings are also added to your Project Risk Register.*

Step 4: Treat the Risk. *This is also referred to as Risk Response Planning. During this step you assess your highest ranked risks and set out a plan to treat or modify these risks to achieve acceptable risk levels. How can you minimize the probability of the negative risks as well as enhancing the opportunities? You create risk mitigation strategies, preventive plans and contingency plans in this step. And you add the risk treatment measures for the highest ranking or most serious risks to your Project Risk Register.*

Step 5: Monitor and Review the risk. *This is the step where you take your Project Risk Register and use it to monitor, track and review risks.*

Risk is about uncertainty. If you put a framework around that uncertainty, then you effectively de-risk your project. And that means you can move much more confidently to achieve your project goals. By identifying and managing a comprehensive list of project risks, unpleasant surprises and barriers can be reduced and golden opportunities discovered. The risk management process also helps to resolve problems when they occur, because those problems have been envisaged, and plans to treat them have already been developed and agreed. You avoid impulsive reactions and going into "fire-fighting" mode to rectify problems that could have been anticipated. This makes for happier, less stressed project teams and stakeholders. The end result is that you minimize the impacts of project threats and capture the opportunities that occur.

Appendix C
Mocked generated questions

1. What are the different steps in the risk management process?
    - **a. Identify the risk, analyze the risk, evaluate/rank the risk, treat the risk and monitor & review the risk.**
    - b. Define scope, identify possible consequences, delegate resources & responsibilities, treat risk.
    - c. Gather information, treat risk and monitor & review the risk.
    - d. Report the risk, evaluate/rank the risk, treat the risk, write a risk report and assess the risk.

2. What is the purpose of the first step in the risk management process?
    - **a. To uncover, recognize and describe risks that might affect your project or its outcomes.**
    - b. To determine the likelihood and consequence of each risk.
    - c. To take your Project Risk Register and use it to monitor, track and review risks.
    - d. To assess your highest ranked risks and set out a plan to treat or modify these risks to achieve acceptable risk levels.

3. What are the two components that characterize the magnitude of the risk?
    - **a. Likelihood and consequence.**
    - b. Exposure and probability.
    - c. Probability and vulnerability.
    - d. Probability and impact.

4. What happens during risk response planning?
    - **a. You assess your highest ranked risks and set out a plan to treat or modify these risks to achieve acceptable risk levels.**
    - b. The risk team is alerted and tasks are divided
    - c. Each risk is divided into a category and prioritized according to their category and rank
    - d. You assess your highest ranked risks and take measures to ensure that these risks never occur again

5. Why is it important to identify project risks?
    - **a. To de-risk the project.**
    - b. To get rid of the project.
    - c. To make sure there are no risks.
    - d. To become risk-averse.

6. What are project risks?
    - **a. Unpleasant surprises and barriers.**
    - b. The list of "bad" things that can happen.
    - c. The list of good things that can happen.
    - d. An area of unexplored research.

7. What is outlining?
   a. **A framework around uncertainty.**
   b. A framework around problems.
   c. A framework around opportunity.
   d. A framework around risks.

8. What approach is used to identify risks?
   a. **The 5-Whys**
   b. The 5-How-To's
   c. The 5-Ways
   d. The 5-What's

9. What will you capture in the process of risk management?
   a. **Opportunities**
   b. Solutions
   c. Stakeholders
   d. Reactions

10. What is the project risk register used for?
    a. **Monitor, track and review risks**
    b. Assess, treat and remove risks
    c. Logging the time spent on each risk
    d. Archive treated risks

11. What is the purpose of the second step in the risk management process?
    a. **To determine the likelihood and consequence of each risk.**
    b. To uncover, recognize and describe risks that might affect your project or its outcomes.
    c. To take your Project Risk Register and use it to monitor, track and review risks.
    d. To assess your highest ranked risks and set out a plan to treat or modify these risks to achieve acceptable risk levels.

12. Why should you avoid going into “fire-fighting” mode?
    a. **To keep team members and stakeholders happier and less stressed**
    b. It gets too hot
    c. It is bad PR for your company
    d. Your boss might fire you