Titel der Diplomarbeit

Dein Name

April 9, 2025

Abstract

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Kurzfassung

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Contents

Chapter 1

Introduction

Link Gliederung: https://www.diplomarbeiten-bbs.at/durchfuehrung/gliederung-der-diplomarbeit-und-formale-vorgaben

1.1 Short description

The topic of this diploma thesis is creating a platform which supports different voting options like single, multiple or weighted choice. Additionally there should be a Login system with different roles to administer and create or delete polls and one where the user can simply vote for the polls he's included in. Furthermore there an option to disclose the results and who voted for which answers. The database should run on a remote server and be accessed by an API.
The reason we chose this topic is because our supervisor is part of the LMP party and they couldn't find an appropriate platform to vote on party intern problems and topics. Hence he approached us and suggested we write our diploma thesis on a voting platform.

1.2 Description of performed work

Our aim is to provide a website where different organizations can create and publish polls for their members. Since our finished work will be open source, everyone who wants to create polls will benefit from our work.
We chose to accept the LMP as our partner, because they brought up that there isn't a platform that supports all the features they need. Moreover can they give us feedback of the real life application so we can adjust the features to a user organization. During the development of our work we had monthly meetings with the LMP to discuss the progress. Because we decided

to develop our software in an agile way the discussions we had with them also helped so we could focus on the more important features first and implement elements of lesser importance later.

1.3 Methodology of the thesis

At first we had to decide on a tech stack. After careful consideration we decided upon a PostgreSQL database, a backend of node.js, sequelize to perform database operations and express to write APIs so we can connect with our frontend. Our frontend is based on React and we also included a PWA. After this decision we began with a simple input and output from front- to backend so ensure we all understood how each part is connected to each other. The next step was implementing the first features. We split the elements in different components so we could work separately and efficiently, e.g the single choice is split in create the poll, display the poll, vote, and show the results. Reasons we chose this tech stack and a thorough description of each function our work has will be in the main part.

Figure 1.1: Hier ist ein Bier

Chapter 2

Tech Stack

2.1 PostgreSQL

2.2 node.js

2.3 Sequelize

2.4 express

2.5 React

2.6 PWA

Chapter 3

Features

```
useEffect(() => {
  const linkParam = window.location.search.substring(1);
  if (linkParam) {
    const unhashed = atob(decodeURIComponent(linkParam));
    const params = new URLSearchParams(unhashed);
    const token = params.get('token');
    if (token) {
      setNewUserRegistration(1);
      setNewUserToken(token);
    } else {
      const publicValue = params.get('public');
      if (publicValue === "true") {
        setIsPublic(1);
      } else {
        setIsPublic(0);
      }
    }
  }
}, []);
```

Figure 3.1: Hier ist ein Beispielcode

```
    useEffect(() => {
      const linkParam = window.location.search.substring(1);
      if (linkParam) {
        const unhashed = atob(decodeURIComponent(linkParam));
        const params = new URLSearchParams(unhashed);
        const token = params.get('token');
        if (token) {
          setNewUserRegistration(1);
          setNewUserToken(token);
        } else {
          const publicValue = params.get('public');
          if (publicValue === "true") {
            setIsPublic(1);
          } else {
            setIsPublic(0);
          }
        }
      }
    }, []);
```

Figure 3.2: Hier ist ein Beispielcode

3.1 Login

3.2 Registration

3.2.1 Roles

Implementing a role-based system with three distinct roles - "Admin," "Poweruser," and "Normal" - is crucial for te functionality and security of the application. By assigning permissions flexibly, a clear hierarchy is established, enhancing both user experience and data integrity. Admins are grated full control over the application, while Poweruser enjoy extended privileges for managing polls. Normal users can seamlessly participate in polls and view results without jeopardizing sensitive functionalities. This structure facilitates efficient task delegation an scalability, allowing the application to be easily expanded with additional roles in the future. The role system thus significantly contributes to the security, organization, and user-friendliness of the polling application.

3.3 Group System

3.4 Create Polls

3.4.1 Start-/ Endtime

3.4.2 Questions

Single Choice

Multiple Choice

Weighted Choice

3.4.3 Demographic Questions

To gather the data of our public voters, the implementation of demographic questions was crucial. This feature is only available for public polls since the created user would be part of the organization using our project and therefore have the data already. If the users is not part of the organization there is still the option to contact them via the e-mail used for the registration. Since most of these questions are similar for every poll, a modular system where questions can be created, added, removed and changed is the best solution. Figure 3.3 shows the demographic question in create polls. The options and functionality of these questions are the like ones described in the previous sections.

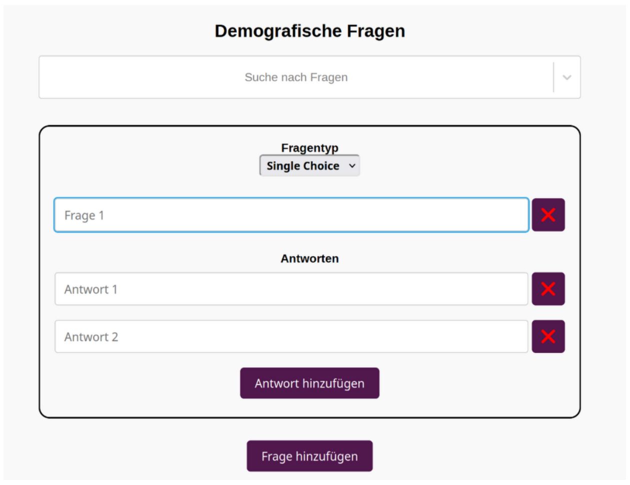

Figure 3.3: Create Demographic Question

The new part for this feature is the search bar. For this the "Select" component of "react-select" is used. The components' controllable state props and modular architecture allows "isMulti" to select multiple options or "isSearchable" to search. These features, allow easy implementation and an already styled search bar in the project. [4]

The database structure for the demographic questions is similar to the standard ones. The table "PublicQuestions" is used to store the question specific data like name and type. To enable the reuse of questions on different polls "PublicQuestions" is in an many-to-many relationship with "Polls" through "PublicPollQuestions". Answers are stored within the table "PublicAnswers", which is connected to "PublicQuestions" via "PublicQuestionAnswers. This relation is also many-to-many since the options yes or no for example would be used in multiple questions.

For the select every existing question with its answers is fetched from the database and stored within an array. The options are then mapped with the value being id and the label as name of each element. To display the selected options they are mapped through and use the same functions as the standard ones used for the poll. When saving these questions a problem arises, as the selected ones are already stored in the database and possibly changed. To handle this an findOrCreate 3.4 is used to get the existing questions and answers or create new ones. This function also returns each instance found or the created one. With this the question and answer ids can be used for the many-to-many relationship. [1]

```
let [createdQuestion, created] = await
↪  PublicQuestions.findOrCreate({
  where: {
    name: question.name,
    typeId: questionType.id,
  }
});
```

Figure 3.4: findOrCreate PublicQuestions

3.5 Edit Polls

3.6 Voting

3.6.1 Disclosed Voting

3.6.2 Anonymous Voting

3.6.3 Public Voting

The public voting allows users without an account to vote. With this feature a wide range of people can be questioned via street surveys or through a shared link. This poll type has two section, the normal and demographic questions. The order of these play a major role. Römermann [3] mentions trust, benefits of the demographic data and the ability to abstain. All of these factors have to be taken into ones account when creating these questions. The article also mentions, that at the beginning of a survey the motivation is high and the demographic data are answered, but the trust in full anonymity is decreased. Therefore the author states it is best to put these questions at the end.

Vote integrity

A major problem when having anonymity and no accounts is the data integrity. Without the ability to store some information about a voter to check for multiple votes, it is important to prevent them from voting multiple times.

what to write about: how to prevent multiple votes: - captcha against bots - cookies to prevent non techie users - users with knowledge almost impossible to prevent without storing ip or device fingerprints, problem with ip and device fp is probably legal reasons

- userData: which information is important for polls (gender, age, job,)

3.7 Results

3.7.1 CSV-Export

3.8 MyPolls

3.8.1 Polllink

3.8.2 Delete Polls

3.9 Accessibility

3.9.1 Tooltips

3.9.2 Screenreader

3.10 Styling

Chapter 4

Summary

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Hier ein Zitat aus einer Quelle [2].

Bibliography

[1] Sequelize Contributors. *Sequelize Documentation.* URL: `https://sequelize.org/docs/v6/core-concepts/model-querying-finders/` (visited on 04/09/2025).

[2] Autor Name. *Beispieltitel des Buches.* Verlag, 2023.

[3] Carina Römermann. *Was ist bei soziodemografischen Angaben in Befragungen zu beachten?* URL: `https://www.rogator.de/soziodemografischen-angaben-mitarbeiterbefragungen/` (visited on 04/09/2025).

[4] Jed Watson and contributors. *React-Select.* URL: `https://www.npmjs.com/package/react-select` (visited on 04/09/2025).

List of Figures