Extra: Product Features Overview

Topicus Team 1

Umair Aamir Mirza (s2928558)
Martin Demirev (s2965046)
Condu Alexandru-Stefan (s2769549)
Narendra Setty (s2944200)
Alexandru Lungu(s3006301)
Teodor Pintilie (s2920344)

Introduction:

This brief report serves to overview the required features for the application that were implemented, and more importantly, serves to indicate all additional features added to the project (both on client-side and development-side).

Required Features (MUST HAVES):

This list of requirements was derived from the original list provided by Topicus:

☑ The system should be able to handle the administration of multiple schools.
- The Topicus Administrator is able to view, add, and remove schools from the system.
- Additionally, the Topicus Administrator is able to view, add, and remove School Administrators for a School.

☑ Fill in a registration form as a parent.
- Two user groups can fill a registration form:
  - Guardian (with account).
  - Guardian (without account).
- Both user groups are able to successfully create registrations.

☑ Create an account as a parent, after having filled out a form.
- Guardians (parents) can create an account after filling in the Registration Form, but they are not forced to perform this functionality as the extra feature of accountless registration management is functional.

☑ See your own registrations and their statuses as a parent, if you have an account.

    - This is enabled in the dashboard, where registrations can be viewed for their status and details, and can be edited if an editing status is allowed.

- ☑ Login as a school administrator.
    - The log-in for a school administrator is possible in the same fashion, which directs school administrators to be able to view registrations for a school on their dashboard (with applied filters).
- ☑ See, update, and delete available registrations as a school admin.
    - The dashboard enables the view of all registrations. The update of registration statuses is possible when an administrator edits the registration. The deletion of registrations is not directly permissible for a school administrator, instead, they can reject the registration directly.
        - We would implement Deletion in the following way: include a status for the registration called DELETE_PENDING, and as soon as the server receives the registration in this state, it will immediately process the deletion.
- ☑ Change the registration form, both in content and styling wise, on a per-school basis.
    - The RegistrationFormBuilder on the front-end enables the creation of unique registration forms and their meta-information in an appropriate manner. These forms are stored with the appropriate styling that they require. The styling includes:
        - Section Header Color.
        - Component Input Field Color.
        - Background Color.
        - Profile Photo (half-implemented).

<u>Functional Additional Features:</u>

The additional features for the application are large in number, and are shown in specific groups as below. As the expectations of the client changed over the course of development, we referred back to the original list of MUST HAVES for the section above, and thus, anything extra was considered as an extra feature. Each element of the group described below is considered an extra feature.

Topicus Administrator:

- Login.
- View All Schools.
- View All School Administrators for a School.
- Create School.
- Create a School Administrator for School.
- Delete School.
- Delete School Administrator for School.

Accountless Registration Management:

- Access existing registrations without an account.
- View registration forms for a school.
- Create registration without an account.
- Creating children without an account.
    - Including choosing previous children.
- Form preview page for details after submission.

Downloadable Elements:

- All registrations for a particular form can be exported to CSV.
- A registration can be downloaded by the parent.

Registration Form Builder:

- View a form that is already existing (inside builder mode).
- Edit a form that is already existing (AND UNRELEASED) inside builder mode.
- Preview styling of a form inside builder mode.

Contact Elements:

- School Administrators leave notes.
- These notes, either public or private (private means to other school administrators in the system for their school, or public to be shown to parents as well) regarding Registration Status Updates.

Registration Form Usability:

- Registration Forms can be deactivated and also controlled with their release permissions to enable greater control over data.

- Forms are displayed in the appropriate styling.

Registration Form Usage:

- Data Type Selection for School Admin.
- Mandatory Type Selection for School Admin.
- Support for Parents to See Data Type and Mandatory Status.

Auto-Complete School Search Bar:

- Search bar for navigating websites and finding the school.
- On click, view the registration forms for a school.

School Admin Dashboard:

- Provides in-built filtering for:
  - Year.
  - Form Name.
  - Registration Status.

Profiles:

- Users can view their profiles.
- Users can manage and edit their profiles.
- Parents can view their children on the profile.

School Detail Page for School Admin:

- Can view the details for the school (not editable).

Server Logger:

- An additional feature was the ServerLogger, which produces access logs that can be viewed by any maintainer to enable greater control over the performance of the server and if it is performing correctly.

JWT Token Validation:

- Authentication was completed with the use of JWT Tokens.

Conclusion:

This report summarizes all the features that were implemented to the application. Over the course of the development, changing perceptions of the application functionality by the client, and also healthy competitiveness between groups pushed for a greater focus on feature implementation. However, on reflection, perhaps some features being omitted would have enabled greater organization of the project, and allowed refactoring of code to make the source code more maintainable for future developers.