

TALLINN UNIVERSITY OF TECHNOLOGY

School of Information Technologies

Andree Uuetoa 213236IADB

**OpusPortal: Study affairs manager
for music students**

ICD0021 Home Project

Supervisor: Andres Käver

Tallinn 2023

Author declaration of originality

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Author: Andree Uuetoa

7th of July 2023

Contents

Introduction	4
The project scope	5
Screen sketches	7
The application's architecture	12
Further improvements.....	13
References	14
Appendix – Entity Relationship Diagram	15

Introduction

The students Tallinn University of Technology have access to an application where they can check their timetable, current curriculum information, their current average grade throughout the years, books lent from the school library, mail and news. This application is called TalTechi Tudengiportaal. Although I am uncertain, how much students actually use it in their daily lives, I can see the potential of such a project to be utilized in other schools as well.

I have a friend who is currently studying at MUBA (Tallinn School of Music and Ballet). When I asked him about whether they had anything similar to Tudengiportaal, he replied that they used to have intranet for their school, where they could see their future duties. At the moment, however, they don't have an application for those instances and he would see quite large benefit of having one.

The goal of this project is to create an application similar to the Tudengiportaal, but designed to be used by MUBA students and teachers. Through the system, the students and teachers can see their school affairs through a dashboard-like UI and such affairs can be modified by either the administrator users or, in some cases, the teachers.

Project Scope - SMALL

Currently, there are dedicated email addresses for MUBA students and teachers. Those email addresses could be used for credentials when registering new users or signing in to the application.

The application would be divided between three different views for three types of users – students, teachers and administrators. The view for the first two would not differ much from one another. Both would include:

- Current subjects declared by a student (or subjects being taught by a teacher) with a real-time timetable
- Active changes to the timetable in case a class has been cancelled or is taking place in a different classroom (distinguishable from the regular timetable)
- The books lent from the school library with the future return dates
- Future concerts and competitions where the student would participate in
- Additional information about the school affairs, e.g. contacts to the personnel of the school

Additionally, the teacher's would have an overview of his/her students' performances and the ability to add the students to perform at a concert or take part in a competition somewhere in the future.

The functionality for the administrators would include:

- The possibility to add concerts and competitions and add students to participate in them
- The ability to add or remove books to the school library
- Creation, management and deletion of users (students and teachers)

At the moment, the application is designed for the administrators to lend books out to the students and teachers, but later this feature would be implemented automatically, so that when the library administrator lends out a book, a corresponding record would be inserted to the database.

Project Scope - LARGE

In addition to the concerts the student would attend, the student's view of the dashboard would include:

- Current subjects declared by a student (or subjects being taught by a teacher) with a real-time timetable
- Active changes to the timetable in case a class has been cancelled or is taking place in a different classroom (distinguishable from the regular timetable)
- The subjects finished by a student, currently running average weighted grade for the student

In order to accommodate the need to insert data to the timetables, the view for the administrators would also include:

- The ability to configure the timetable
- Management of curriculums and subjects taught at school

Since there may be foreign students studying at MUBA, it would be beneficial if the application included multilingual support. Initially, the application would support Estonian, but later it would also support English.

Screen sketches

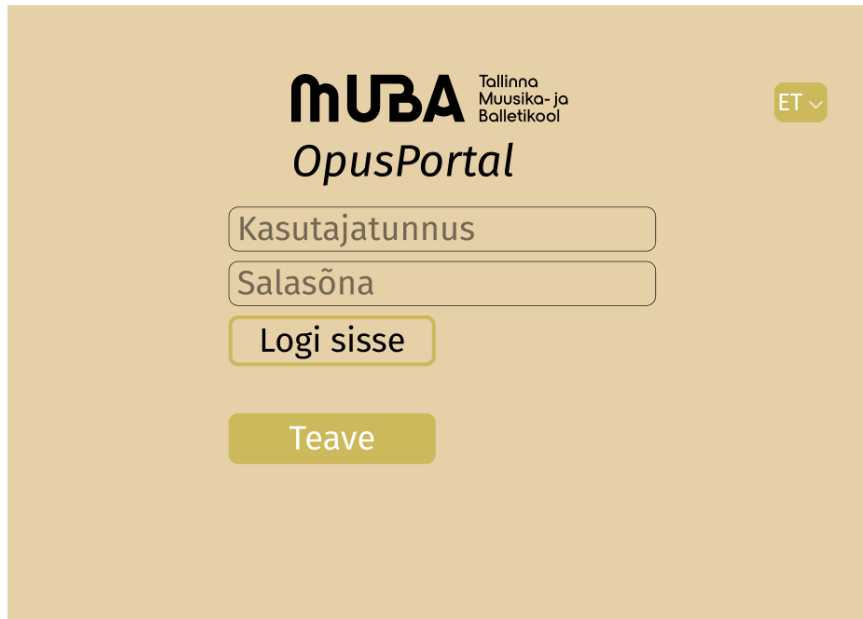


Fig 1. The opening (login) page

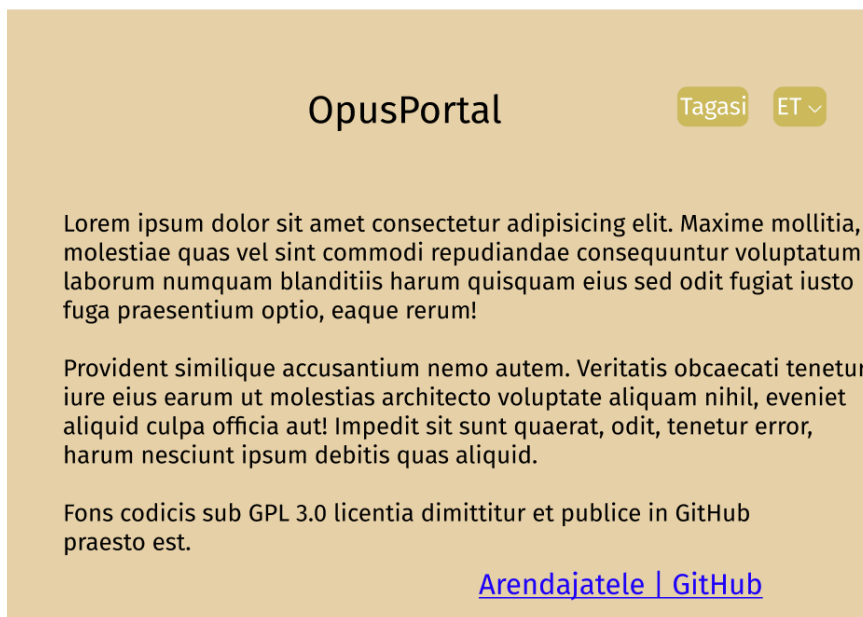


Fig 2. Clicking „Teave“ („About“) from the opening page: The „About“ page

Clicking the „Tagasi“ („Back“) button will return the user to the opening page shown in Figure 1.

After logging in, a dashboard according to the user's role will be displayed:

OpusPortal

Tere, [nimi]!
ET ▼ Kasutaja sätted | Logi välja

Tunniplaan

P, 15.10	E, 16.10	T, 17.10	K, 18.10	N, 19.10
	09:00-10:30 Itaalia keel D314	11:00-13:00 Ansambel Black-box	12:00-13:30 Klaver D217	10:00-11:30 Muusikaajalugu A123
	10:45-11:30 Solfedžo D222			12:00-14:00 Ansambel Black-box
	12:00-13:30 Klaver D217			

Esinemised

Lisa/Eemalda (kooliväline) esinemine? Eksamid/arvestused? Konkursid?

Nimetus	Aeg	Asukoht
Beethoveni 253. sünniaastapäeva kontsert	17. detsember 2023, 20:00	Estonia kontserdisaal
In Corpore 2023	10. märts 2024, 18:30	Tallinna Muusika- ja Balletikool

Toimuvad õppeained

Nimetus	Õpetaja(d)	EAP
Solfedžo II	Taavi Talviste	6
Sissejuhatus ansamblimängu	Riina Raja	6
Itaalia keel muusikutele	Ursula Ratas	6
Muusikaajalugu II	Kalle Mets	6
Eriala II	Olga Pekareva	6

Laenutatud raamatud

Nimi	Autor	Tähtaeg
Das wohltemperierte Klavier I	Johann Sebastian Bach	23. oktoober 2023
Fryderyk Chopin: A Life and Times	Alan Walker	20. oktoober 2023
Mozart: The Reign of Love	Jan Swafford	7. oktoober 2023

Õppekava

Õppekava: klaver, kutseõpe
Kaalutud keskmine hinne: 4,5
Omandatud EAP: 24
Semester: 2

Kontaktid

Õppematerjalid
[Kohustuslikud materjalid/kirjandus](#)

Fig 3. Student's dashboard

OpusPortal
Tere, [nimi]!
ET Kasutaja sätted | Logi välja

Tunniplaan

Muuda ruumi, kus tund toimub?

P, 15.10	E, 16.10	T, 17.10	K, 18.10	N, 19.10
	<div>10:45-11:30 Solfedžo D222</div> <div>12:00-13:30 Solfedžo D222</div>		12:00-13:30 Muusikaajalugu A221	10:00-11:30 Muusikaajalugu A123

Õpilaste esinemised

Lisa/Eemalda esinemine, eksamid/arvestused, konkursid

Nimetus	Aeg	Asukoht	Esineja
Erialaeksam	9. detsember 2023, 18:30	Tallinna Muusika- ja Balletikool	Ville Ümera
Beethoveni 253. sünniaastapäeva kontsert	17. detsember 2023, 20:00	Estonia kontserdisaal	Luisa Viks

Laenutatud raamatud

Nimi	Autor	Tähtaeg
Das wohltemperierte Klavier I	Johann Sebastian Bach	23. oktoober 2023
Fryderyk Chopin: A Life and Times	Alan Walker	20. oktoober 2023
Mozart: The Reign of Love	Jan Swafford	7. oktoober 2023

Kontaktid

[Õppematerjalid](#)
[Kohustuslikud materjalid/kirjandus](#)

Fig 4. The teacher's dashboard

The screenshot shows the 'Lisa kontsert' form in the OpusPortal application. The header includes the site name 'OpusPortal', the user status 'Tere, administraator!', and navigation links 'ET', 'Kasutaja sätted', and 'Logi välja'. The main navigation bar contains 'Kontserdid', 'Konkursid', 'Raamatukogu', 'Õppimine', and 'Kasutajad'. The form itself has input fields for 'Nimi' and 'Asukoht', a text area for 'Asutused, mille nimed on sarnased otsitavaga, klikitavad', and calendar pickers for 'Algusaeg (kalender)' and 'Lõppaeg (kalender)'. A yellow 'Lisa kontsert' button is at the bottom right.

Fig 7. Adding a concert

After clicking „Lisa kontsert“, the user will be redirected back to the page showing all the concerts

The screenshot shows the concerts list page with a confirmation pop-up. The header and navigation bar are identical to Figure 7. The main content area has a table with columns 'Nimi', 'Asukoht', 'Algusaeg', 'Lõppaeg', and 'Lisa kontsert'. A pop-up dialog is displayed over the table, asking 'Kas oled kindel, et soovid kustutada kontserdi [Nimi]?' with 'Jah' and 'Ei' buttons. Behind the pop-up, buttons for 'Lisada kontserti' and 'Kustuta kontsert' are visible.

Fig 8. A pop-up will show up on the concerts list page when clicking „Kustuta kontsert“ („Delete concert“)

After clicking „Logi välja“ („Sign out“) button located in the header, the user will be returned to the login page in Figure 1

The other sections of the application (managing users, concerts and books) would look analogous to the concerts' section.

The application's architecture

The application will be written in the .NET Core framework with a frontend done in React and using a PostgreSQL database. Later, the application will be hosted on Azure cloud server with the application image built in Docker. The source code for both the backend and the frontend will be publicly available on my GitHub page under the GNU General Public License.

Although the aim of this project is to have the data stored securely in a database in the cloud, the application will be designed so that the database engine could be switched to some other standard SQL database, a NoSQL database, a distributed database, a file-based repository or any other method of storing data. Similarly, the application needs to be designed so that the user interface can be implemented in many different technologies, such as a web application for the desktop or a mobile application.

For such purposes, I could implement an architecture popularized by an American software developer Robert C. Martin (Uncle Bob). His vision of an application architecture describes that the business rules of an application must be changed only in crucial circumstances, indicated by being in the inner circle shown on the diagram, and the elements in the outer circle, such as the database, front-end elements and external interfaces, should be easily appendable and replacable. In addition, any changes made to the elements in the outer circle must not cause any changes to the business rules in the inner circle.

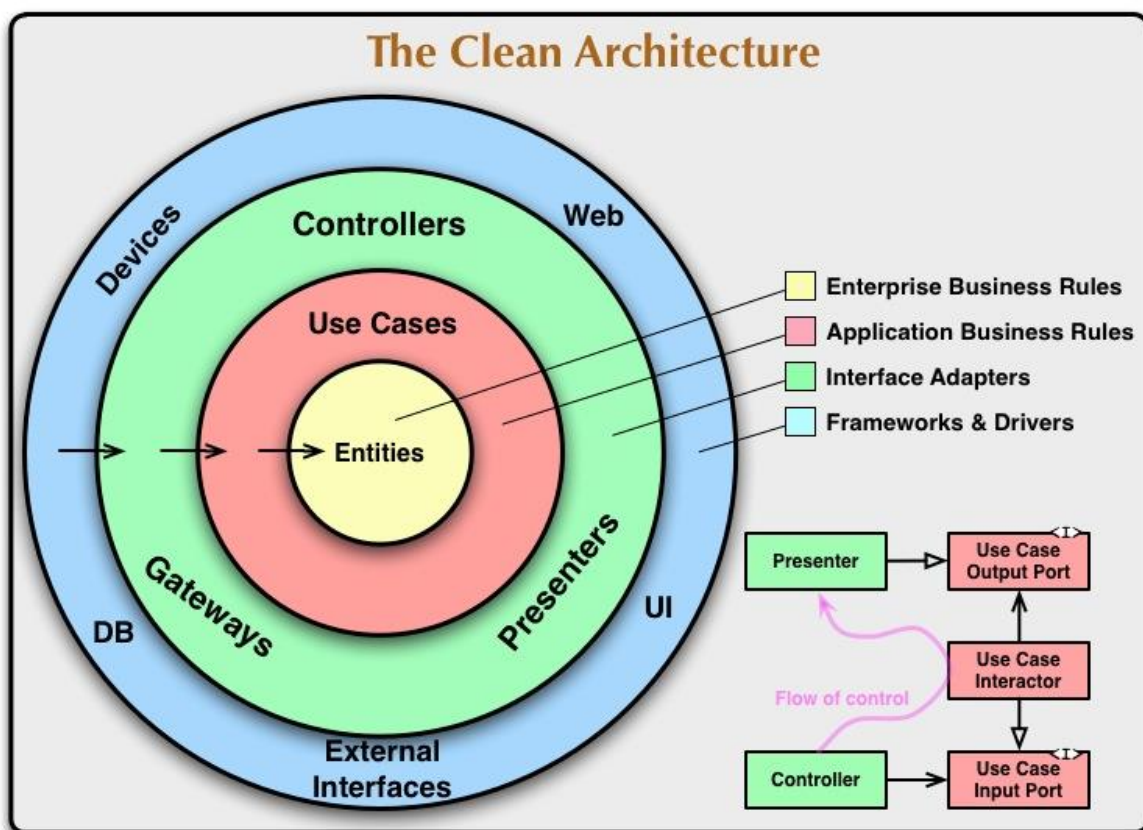


Fig 9. „The Clean Architecture“, as proposed by Robert C. Martin^[1]

Further improvements

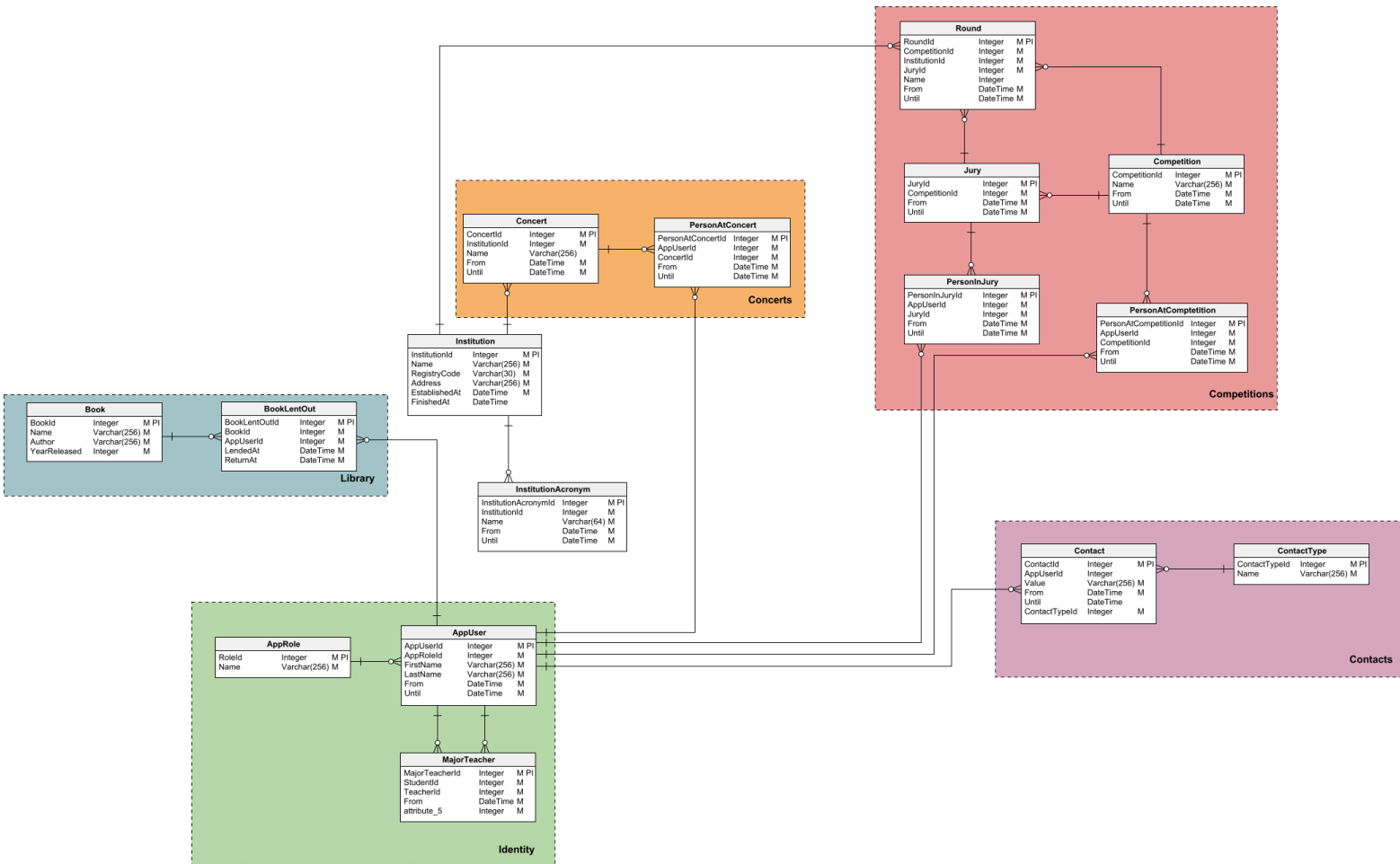
In an event where a teacher would assign his/her student to perform at a concert, the student could receive a notification of some kind. Since the users already are using their email addresses as sign-in credentials, the same addresses can also be used to notify them via mail, when needed. For this, it seems I would need to compose a special email service to handle the message sending, since, according to Microsoft, it is not recommended to use their own `SmtpClient` class in modern .NET Core framework.^[2]

Once the application is done and being hosted in a cloud server, I have a plan to implement it in real life and offer it to MUBA for use. If the application will be used broadly, I would also create a hybrid mobile application that uses the OpusPortal API as its backend. Although it is currently unclear in which technology it would be wise to write the mobile application in, it's possible that it will be written in one of the four technologies Andres Käver has spoken about in his hybrid mobile application lectures – React Native, Flutter, Xamarin or by writing a Progressive Web App. Provided, that the application is designed so that it adheres to the architecture shown previously, creating an OpusPortal mobile application will not oblige me to rewrite the business rules to my application.

References

- [1]: <https://blog.cleancoder.com/uncle-bob/2012/08/13/the-clean-architecture.html>
- [2]: <https://learn.microsoft.com/en-us/dotnet/api/system.net.mail.smtpclient?view=net-7.0>

Appendix A – Entity Relationship Diagram (Small scope)



Appendix B – Entity Relationship Diagram (Large scope)

