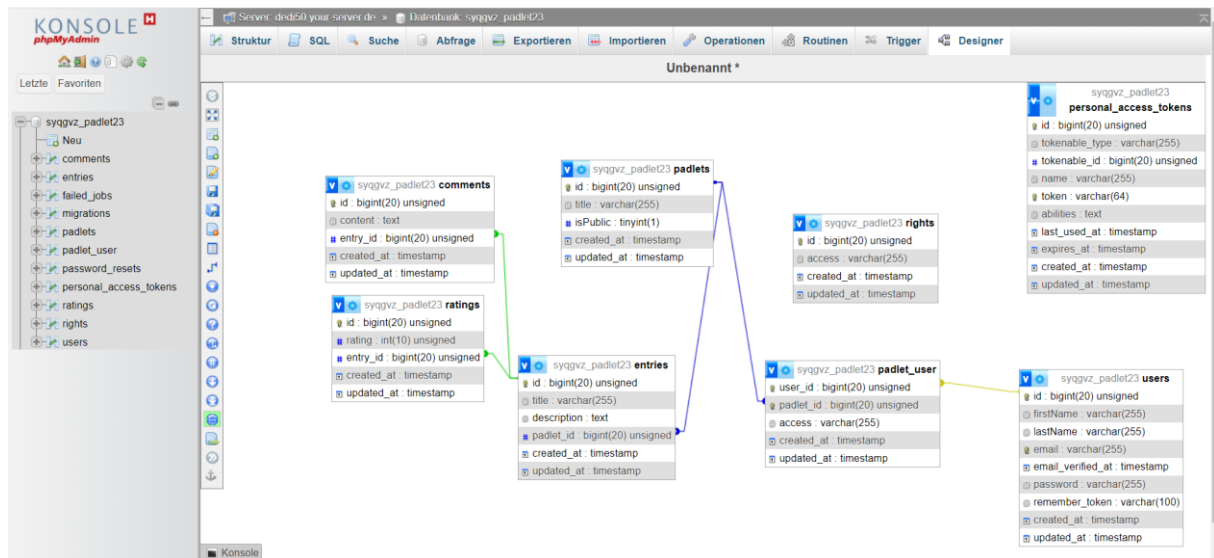


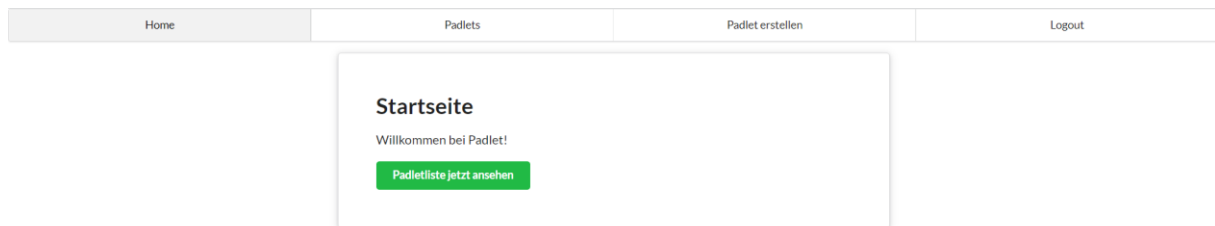
Datenbankschema:



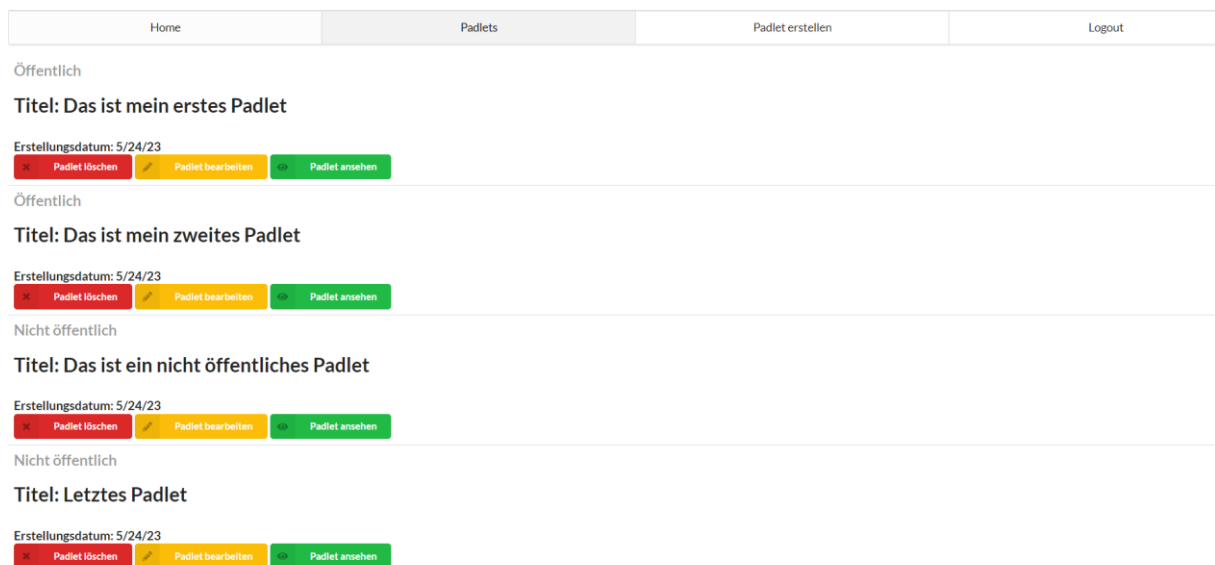
Mein Datenbankschema beinhaltet die Tabellen

- Users
 - Jeder User hat eine id, einen firstName, lastName, email und ein password
- Padlet_user vereint die padlet_id und die user_id gemeinsam mit dem access
 - Die Rechte wurden aus Zeigründen nicht vollständig umgesetzt
- Rights
 - Wie bereits erwähnt nicht vollständig umgesetzt aus Zeitgründen
 - Beinhaltet eine id und den access, welcher bestimmt, was getan werden darf (schreiben, lesen, etc.)
- Entries
 - Beinhaltet die id, den title, die description und bekommt die padlet_id als Fremdschlüssel
- Padlets
 - Beinhaltet die id, den title des padlets und sagt, ob es öffentlich oder nicht öffentlich ist
- Comments
 - Jeder Kommentar gehört zu einem Entry und beinhaltet dementsprechend auch die entry_id als Fremdschlüssel
 - Zudem hat jeder Kommentar eine id und den content
- Ratings
 - Jedes Rating gehört ebenfalls zu einem Entry und beinhaltet zusätzlich eine id und ein rating

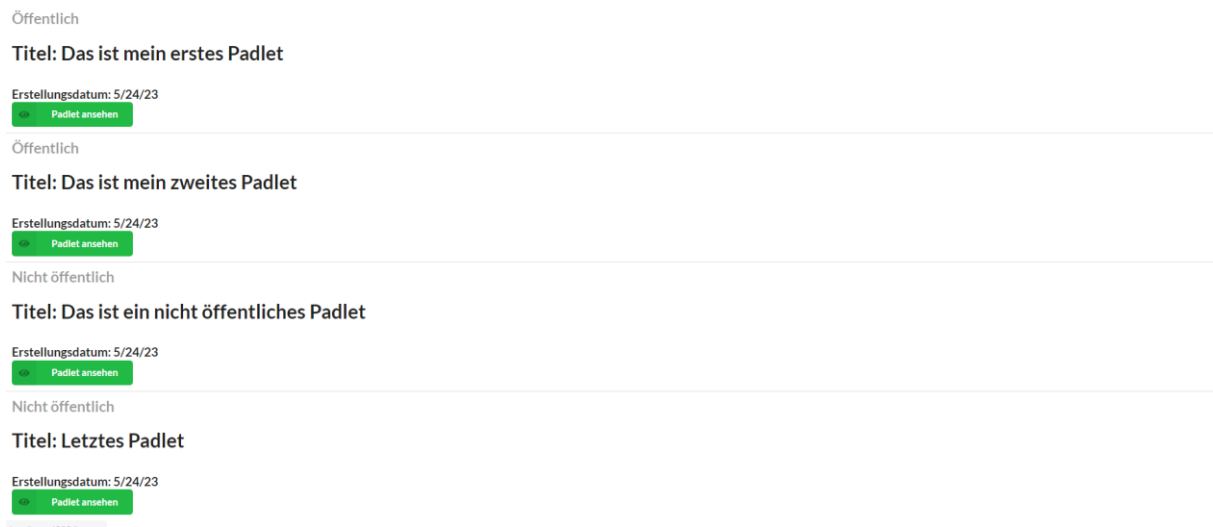
Jeder User kommt zu Beginn auf die Startseite, wo er zu den Padlets navigieren kann. Dies funktioniert sowohl über den Button in der Mitte, als auch über die Reiter am oberen Bildschirmrand.



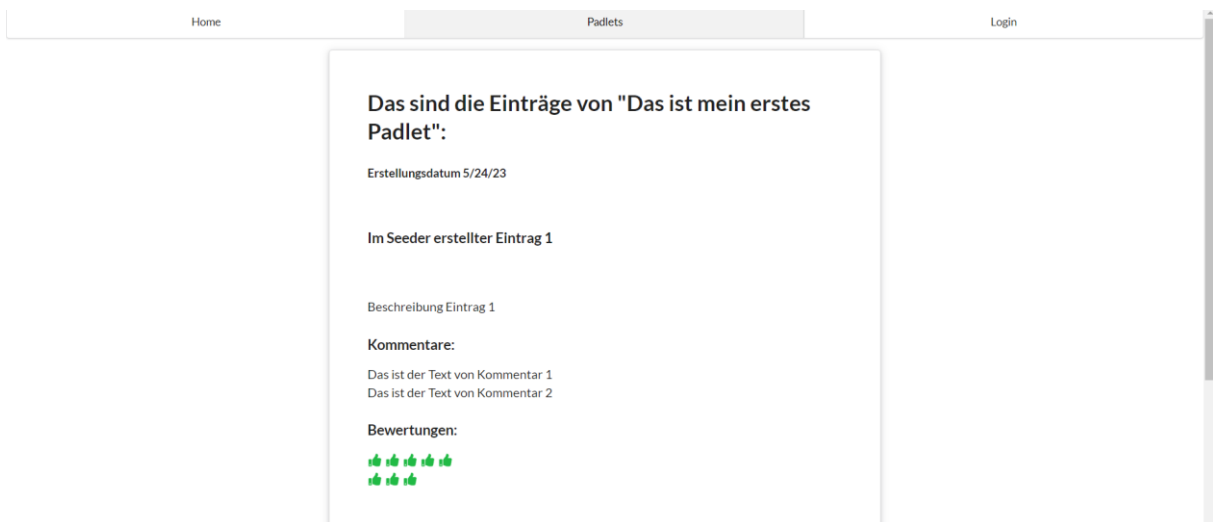
Danach kommt man zur Listenansicht der Padlets, in welchen die unterschiedlichen Padlets mit dem Status und dem Erstellungsdatum und Titel zu sehen sind.



Wenn man nicht eingeloggt ist, kann man auch nicht bearbeiten oder löschen, sondern nur das Padlet ansehen.



In den einzelnen Padlets sieht man die einzelnen Daten des Padlets, sowie die Bewertungen und Kommentare.



Als eingeloggtter User hat man zusätzlich die Möglichkeit die Einträge zu löschen und auch neue Einträge zu erstellen.

Beschreibung Eintrag 1

Kommentare:

Das ist der Text von Kommentar 1

Das ist der Text von Kommentar 2

Bewertungen:



✖ Entry löschen

Das ist ein zweiter Eintrag

Dieser zweite Eintrag passt gut

Kommentare:

Bewertungen:

✖ Entry löschen

+ Entry erstellen

Zurück zur Padletliste

Auch das Updaten von Einträgen ist möglich, indem man den Titel oder die Öffentlichkeit ändert.

Home	Padlets	Padlet erstellen	Logout
Öffentlich			
Titel: Das ist mein erstes Padlet			
Erstellungsdatum: 5/24/23			
✖ Padlet löschen	✎ Padlet bearbeiten	👁 Padlet ansehen	
Öffentlich			
Titel: Das ist mein zweites Padlet			
Erstellungsdatum: 5/24/23			
✖ Padlet löschen	✎ Padlet bearbeiten	👁 Padlet ansehen	
Nicht öffentlich			
Titel: Das ist ein nicht öffentliches Padlet - mit Update			
Erstellungsdatum: 5/24/23			
✖ Padlet löschen	✎ Padlet bearbeiten	👁 Padlet ansehen	
Nicht öffentlich			
Titel: Letztes Padlet			
Erstellungsdatum: 5/24/23			
✖ Padlet löschen	✎ Padlet bearbeiten	👁 Padlet ansehen	

Unter dem Reiter „Padlet erstellen“ ist es möglich Padlets zu erstellen und anzugeben, welchen Titel es hat und ob es öffentlich oder privat ist. Wenn der User keine vollständige Eingabe tätigt, oder eine die Fehler beinhaltet, ist ein speichern nicht möglich. Man gelangt auch hier hin, wenn ein Padlet geupdatet werden soll.

Home	Padlets	Padlet erstellen	Logout
------	---------	------------------	--------

Erstellen eines Padlets

Padlettitel

Soll dieses Padlet öffentlich sein? (0 = nicht öffentlich 1 = öffentlich)

Speichern

Unter dem letzten Reiter ist es möglich sich ein bzw. auszuloggen mit den richtigen Benutzerdaten. In diesem Fall ist es möglich sich mit den Daten Kothbauer@gmail.com und secret einzuloggen.

Home	Padlets	Login
------	---------	-------

Login

E-Mail

Passwort

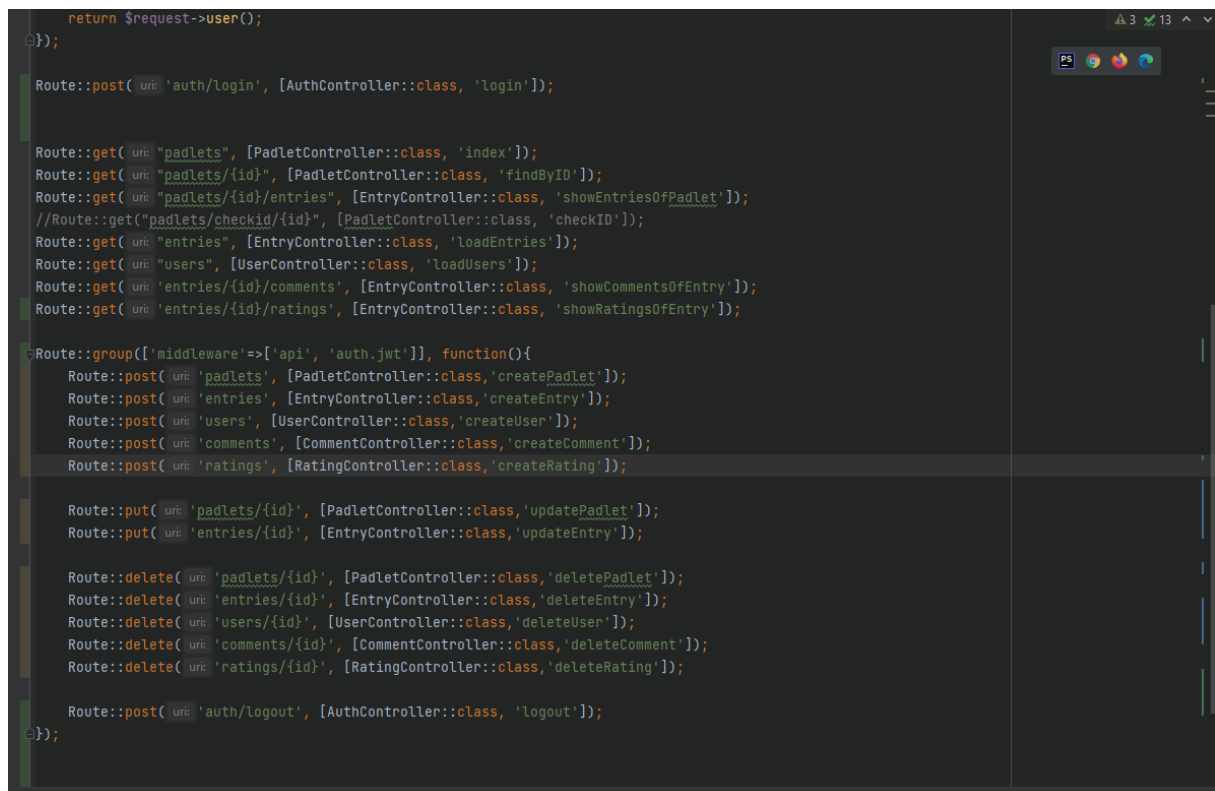
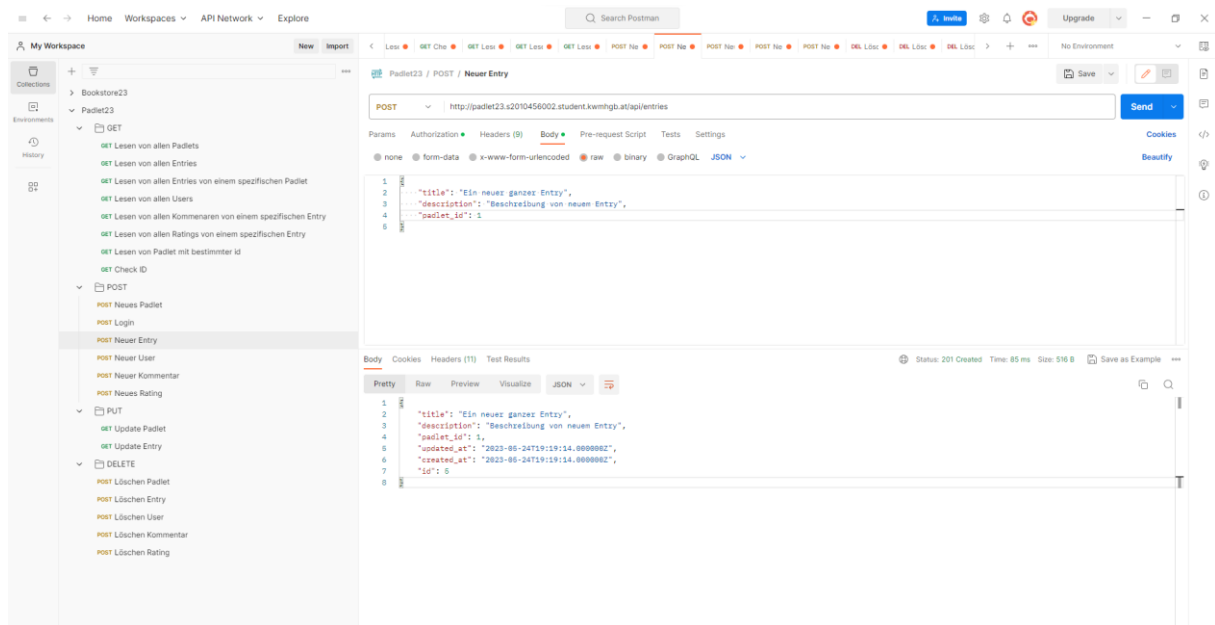
Login

Je nachdem, ob ein User eingeloggt ist, ändern sich die Reiter am oberen Bildschirmrand.

Home	Padlets	Login
------	---------	-------

Home	Padlets	Padlet erstellen	Logout
------	---------	------------------	--------

Im Postman wurden viele HTTP-Requests abgesetzt, um einen reibungslosen Ablauf während der Frontend-Programmierung zu gewährleisten.



Ein nicht eingeloggter User, kann bloß GET-Requests absetzen, ein eingeloggter User kann auch andere Requests absetzen.

```
1 import { NgModule } from '@angular/core';
2 import { RouterModule, Routes } from '@angular/router';
3 import { HomeComponent } from "../home/home.component";
4 import { PadletDetailsComponent } from "../padlet-details/padlet-details.component";
5 import { PadletListComponent } from "../padlet-list/padlet-list.component";
6 import { PadletFormComponent } from "../padlet-form/padlet-form.component";
7 import { LoginComponent } from "../login/login.component";
8
9 const routes: Routes = [
10   {path: '', redirectTo: 'home', pathMatch: 'full'},
11   {path: 'home', component: HomeComponent},
12   {path: 'padlets', component: PadletListComponent},
13   {path: 'padlets/:padlet_id', component: PadletDetailsComponent},
14   {path: 'padlets/:padlet_id/entries', component: PadletDetailsComponent},
15   {path: 'admin', component: PadletFormComponent},
16   {path: 'admin/:id', component: PadletFormComponent},
17   {path: 'login', component: LoginComponent},
18 ];
19
20 @NgModule({
21   imports: [RouterModule.forRoot(routes)],
22   exports: [RouterModule]
23 })
24 export class AppRoutingModule { }
25
```

Im Frontend habe ich diese Routen definiert. Die Route admin soll nur von eingeloggten Usern erreicht werden. Aus Zeitgründen wurde hier kein Guard hinzugefügt, wäre jedoch sinnvoll, um nicht authentifizierte User daran zu hindern, die URL zu benutzen.

Leider ist mir die Zeit ausgegangen, weshalb ich nicht mehr implementiert habe, dass man Kommentare und Ratings über die UI hinzufügen kann. Auch das Rechtesystem habe ich aus Zeitgründen nicht mehr umgesetzt, tut mir leid.