

Student Service System

Aplikacja ułatwiająca zarządzanie studentami

Filip Sokół

Arkadiusz Kędziora

Mateusz Łuczyński

Kierunek: Informatyka

Forma studiów: stacjonarne

Spis treści

WYMAGANIA FUNKCJONALNE 3 WYMAGANIA NIEFUNKCJONALNE 3 DIAGRAM KLAS 3 ENDPOINTY 4 DEPARTMENTS: 4 Wyświetl wszystkie obiekty typu Department: 4 Wyświetl obiekt typu Department o podanym Id 4 Dodaj obiekt typu Department 5 Edytuj obiekt typu Department 5 Edytuj obiekt typu Department 6 STUDIENTS: 5 Wyświetl wszystkie obiekty typu Group: 5 Wyświetl wszystkie obiekty typu Group: 5 Wyświetl wszystkie obiekty typu Group: 6 Edytuj obiekt typu Group 6 Edytuj obiekt typu Group 7 Edytuj obiekt typu Group 8 Edytuj obiekt typu Group 9 Usuń obiekt typu Group 9 Usuń obiekt typu Group 0 Edytuj obiekt typu Student 7 Edytuj obiekt typu Student 8 Edytuj obiekt typu Student 9 Edytuj obiekt typu Wark 0 Dodaj obiekt typu Mark 0 Dodaj obiekt typu Mark 0 Edytuj obiekt typu Mark 0 Edyświetl obiekt typu Mark 0 Edyświetl obiekt typu Mark 0 Edytuj obiekt typu Mark 0 Edytwi obiekt typu Mark 0 Edytuj obiekt typu Wark 0 Edytuj obiekt typu Subject 0 Edytuj obiekt 0 Edytuj obiekt 0 Edytuj obiekt 0 Edytuj obiekt 0 Edy	WYKORZYSTANE TECHNOLOGIE	3
DIAGRAM KLAS	WYMAGANIA FUNKCJONALNE	3
DIAGRAM KLAS	WYMAGANIA NIEFUNKCIONALNE	3
ENDPOINTY		
DEPARTMENTS:		
Wyświetl wszystkie obiekty typu Department: Wyświetl obiekt typu Department o podanym Id		
Wyświetl obiekt typu Department o podanym Id		
Dodaj obiekt typu Department		
Edytuj obiekt typu Department		
Usuń obiekt typu Department o podanym Id	, , ,	
GROUPS:		
Wyświetl wszystkie obiekty typu Group:		
Wyświetl obiekt typu Group o podanym Id		
Dodaj obiekt typu Group 6 Edytuj obiekt typu Group 7 Usuń obiekt typu Group o podanym Id 6 STUDENTS: 7 Dodaj obiekt typu Student 7 Edytuj obiekt typu Student 0 podanym Id 8 MARKS: 8 Wyświetl wszystkie obiekty typu Mark: 8 Wyświetl obiekt typu Mark o podanym Id 8 Dodaj obiekt typu Mark o podanym Id 8 SUBJECTS: 9 Wyświetl wszystkie obiekty typu Subject: 9 Wyświetl obiekt typu Subject o podanym Id 9 Edytuj obiekt typu Subject 10 Edytuj obiekt typu Subject 10	Wyświetl wszystkie obiekty typu Group:	5
Edytuj obiekt typu Group	Wyświetl obiekt typu Group o podanym IdId.	5
Usuń obiekt typu Group o podanym Id 6 STUDENTS: 7 Dodaj obiekt typu Student 7 Edytuj obiekt typu Student 6 Usuń obiekt typu Student 0 podanym Id 8 MARKS: 8 Wyświetl wszystkie obiekty typu Mark: 8 Wyświetl obiekt typu Mark o podanym Id 8 Dodaj obiekt typu Mark o podanym Id 8 Usuń obiekt typu Mark o podanym Id 8 Usuń obiekt typu Mark o podanym Id 9 SUBJECTS: 9 Wyświetl wszystkie obiekty typu Subject: 9 Wyświetl obiekt typu Subject o podanym Id 9 Edytuj obiekt typu Subject . 10	Dodaj obiekt typu Group	ε
STUDENTS: Dodaj obiekt typu Student Edytuj obiekt typu Student Usuń obiekt typu Student o podanym Id MARKS: Wyświetl wszystkie obiekty typu Mark: Wyświetl obiekt typu Mark o podanym Id Dodaj obiekt typu Mark o podanym Id SUBJECTS: Wyświetl wszystkie obiekty typu Subject: Wyświetl obiekt typu Subject o podanym Id SUBJECTS: SUBJE	Edytuj obiekt typu Group	<i>6</i>
Dodaj obiekt typu Student	Usuń obiekt typu Group o podanym IdId in wie wie wie wie wie de la company de l	<i>6</i>
Edytuj obiekt typu Student	STUDENTS:	7
Edytuj obiekt typu Student	Dodaj obiekt typu Student	
Usuń obiekt typu Student o podanym Id	, , , , , , , , , , , , , , , , , , , ,	
MARKS:		
Wyświetl wszystkie obiekty typu Mark:		
Wyświetl obiekt typu Mark o podanym Id		
Dodaj obiekt typu Mark		
Usuń obiekt typu Mark o podanym Id		
SUBJECTS:		
Wyświetl wszystkie obiekty typu Subject:9 Wyświetl obiekt typu Subject o podanym Id9 Edytuj obiekt typu Subject		
Wyświetl obiekt typu Subject o podanym Id9 Edytuj obiekt typu Subject		
Edytuj obiekt typu Subject		

Wykorzystane technologie

Client: React, SCSS, Axios, Vite

Server: C#, .NET 6, TSQL

Mobile: React Native, TypeScript, Expo

Wymagania Funkcjonalne

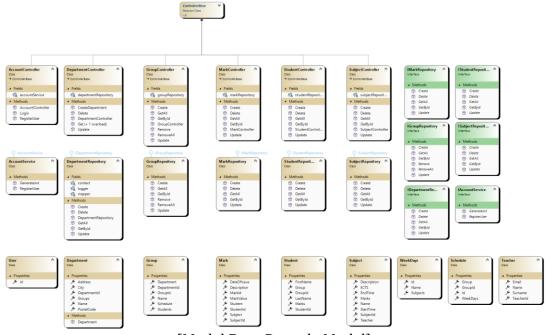
- Użytkownik może zarejestrować się oraz zalogować do aplikacji. - Użytkownik może wyświetlić swoje dane i w zależności od roli wykonywać adekwatne akcje na stronie. - Użytkownik z rolą Nauczyciela może wystawiać studentom oceny. - Użytkownik z rolą Admin może przypisywać studentów do grup.

Wymagania Niefunkcjonalne

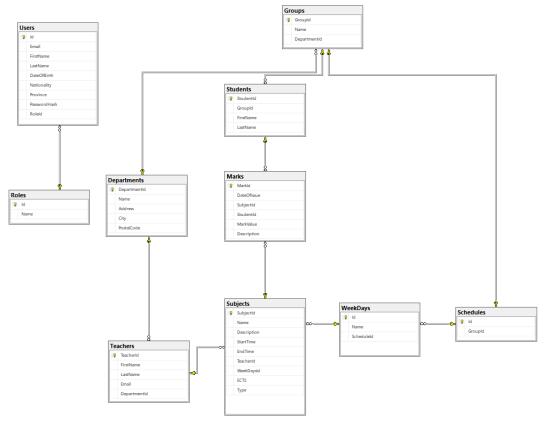
- Aplikacja powinna być cross-platformowa.
 - Wykorzystanie technologii ASP.NET Core 6.0
 / MS SQL / React Native 0.70 / ReactJS 18.2.0
 - Dokumentacja techniczna oparta na Swashbuckle Swagger.

Diagram Klas

Poniżej zamieszczone są główne klasy wykorzystane w części Backendu.



[Model Bazy Danych Model]



[Model Architektury]

Endpointy

Każdy z obiektów biznesowych posiada własny endpoint umożliwiający tworzenie, odczytywanie, modyfikację oraz usuwanie obiektu z bazy danych.

Poniżej przedstawiono endpointy wraz z metodami:

Parametr	Тур	Opis
TBD JWT Bearer	string	Wymagany. Dla wszystkich operacji wymagany jest token autentykacyjny

Departments:

Wyświetl wszystkie obiekty typu Department:

GET /api/departments

Wyświetl obiekt typu Department o podanym Id

GET /api/departments/\${id}

```
Parametr Typ Opis
id
                Wymagane. Id wydziału
Dodaj obiekt typu Department
  POST /api/departments
                                     Opis
Body
                      Typ
CreateDepartmentDto serialized json Wymagane. Id wydziału
Przykład body CreateDepartmentDto
   "Name": "Wydział Inżynierii",
   "Address": "Zielona 3",
   "City": "Częstochowa",
   "PostalCode": "42-700"
}
Edytuj obiekt typu Department
  PUT /api/departments/{id}
                                     Opis
Body
                      Typ
UpdateDepartmentDto serialized json Wymagane. Id wydziału
Przykład body UpdateDepartmentDto
   "Name": "Wydział Inżynierii",
   "Address": "Zielona 3",
   "City": "Częstochowa",
   "PostalCode": "42-700"
}
Usuń obiekt typu Department o podanym Id
  DELETE /api/departments/${id}
Parametr Typ Opis
id
           int Wymagane. Id wydziału
Groups:
Wyświetl wszystkie obiekty typu Group:
  GET /api/departments/{departmentId}/groups
Parametr
               Typ Opis
departmentId int
                    Wymagane. Id wydziału
Wyświetl obiekt typu Group o podanym Id
  GET /api/departments/{departmentId}/groups/{groupId}
```

Parametr	Тур	Opis
departmentId	int	Wymagane. Id wydziału
groupId	int	Wymagane. Id grupy

Dodaj obiekt typu Group

POST /api/departments/{departmentId}/groups

Parametr	Тур	Opis
departmentId	int	Wymagane. Id wydziału
Body	Тур	Opis
CreateGroupDto	serialized json	Wymagane. Dane tworzonego Deparamentu
<pre>Przykład body CreateDepartmentDto { "Name": "group_2022_2", }</pre>		

Edytuj obiekt typu Group

PUT /api/departments/{departmentId}/groups/{groupId}

Parametr	Тур	Opis
departmentId	int	Wymagane. Id wydziału
groupId	int	Wymagane. Id grupy
Body	Тур	Opis
UndataGnoupDto	comiolized icon	Warmagana Dana turanganaga Danananantu

UpdateGroupDto serialized json Wymagane. Dane tworzonego Deparamentu

```
Przykład body UpdateGroupDto
{
    "Name": "group_2022_2",
}
```

Usuń obiekt typu Group o podanym Id

DELETE /api/departments/{departmentId}/groups/{groupId}

Parametr	Тур	Opis
departmentId	int	Wymagane. Id wydziału
groupId	int	Wymagane. Id grupy

Usuń wszystkie obiekty typu Group dla danego wydziału DELETE /api/departments/{departmentId}/groups

Parametr	Тур	Opis
departmentId	int	Wymagane. Id wydziału

Students:

Wyświetl wszystkie obiekty typu Student:

GET /api/departments/{departmentId}/groups/{groupId}/students

Parametr	Тур	Opis
departmentId	int	Wymagane. Id wydziału
groupId	int	Wymagane. Id grupy

Wyświetl obiekt typu Student o podanym Id GET /api/students/{id}

Parametr	Тур	Opis
id	int	Wymagane. Id studenta

Dodaj obiekt typu Student

POST /api/departments/{departmentId}/groups/{groupId}/students

Parametr	Тур	Opis
departmentId	int	Wymagane. Id wydziału
groupId	int	Wymagane. Id grupy
Body	Тур	Opis
CneateStudentDto	coriolized icon	Wymagana Dana twarzanai grupy

CreateStudentDto serialized json Wymagane. Dane tworzonej grupy

```
Przykład body CreateStudentDto
{
    "FirstName": "Jan",
    "LastName": "Kowalski"
}
```

Edytuj obiekt typu Student

PUT /api/departments/{departmentId}/groups/{groupId}/students/{id}

Parametr	Тур	Opis
departmentId	int	Wymagane. Id wydziału
groupId	int	Wymagane. Id grupy
id	int	Wymagane. Id studenta
Body	Тур	Opis
UndataStudantDta	agriclized icon	Warmagana Dana turanganai anuny

UpdateStudentDto serialized json Wymagane. Dane tworzonej grupy

```
Przykład body UpdateStudentDto
{
    "FirstName": "Jan",
    "LastName": "Kowalski"
}
```

Usuń obiekt typu Student o podanym Id

DELETE /api/students/{id}

Parametr	Тур	Opis
id	int	Wymagane. Id studenta

Marks:

Wyświetl wszystkie obiekty typu Mark:

GET /api/students/{studentId}/marks

Parametr	Тур	Opis
studentId	int	Wymagane. Id studenta

Wyświetl obiekt typu Mark o podanym Id

GET /api/students/{studentId}/marks/{id}

Parametr	Тур	Opis
studentId	int	Wymagane. Id studenta
id	int	Wymagane. Id oceny

Dodaj obiekt typu Mark

POST /api/students/{studentId}/marks

Parametr	Тур	Opis
studentId	int	Wymagane. Id studenta
Body	Тур	Opis
CreateMarkDto	serialized json	Wymagane. Dane tworzonej oceny

Przykład body CreateMarkDto

```
{
    "DateOfIssue": "2022-01-01:14:42:34",
    "SubjectId": 41,
    "Description": "This is a mark!",
    "StudentId": 132412,
    "MarkValue": 5
}
```

Edytuj obiekt typu Mark

PUT /api/students/{studentId}/marks/{id}

Parametr	Typ	Opis
studentId	int	Wymagane. Id studenta
id	int	Wymagane. Id oceny

```
Body
                               Opis
                Typ
UpdateMarkDto
                serialized
                               Wymagane. Dane tworzonej
                ison
                               oceny
Przykład body UpdateMarkDto
   "DateOfIssue": "2022-01-01:14:42:34",
   "SubjectId": 41,
   "Description": "This is a mark!",
   "StudentId": 132412,
   "MarkValue": 5
}
Usuń obiekt typu Mark o podanym Id
  DELETE /api/students/{studentId}/marks/{id}
Parametr
            Typ Opis
 studentId int Wymagane. Id studenta
id
            int Wymagane. Id oceny
Subjects:
Wyświetl wszystkie obiekty typu Subject:
  GET /api/subjects
Wyświetl obiekt typu Subject o podanym Id
  GET /api/subjects/{id}
Parametr Typ Opis
id
                Wymagane. Id przedmiotu
Dodaj obiekt typu Subject
  POST /api/subjects
Body
                    Typ
                                  0pis
CreateSubjectDto serialized json Wymagane. Dane tworzonego przedmiotu
Przykład body CreateSubjectDto
{
   "Name": "Projekt zespołowy",
   "Description": "This is a subject",
   "StartTime": "2022-01-01:14:42:34",
   "EndTime": "2022-01-01:15:42:34",
   "WeekDaysId": 3,
   "ECTS": 4,
   "TeacherId": 5
}
```

Edytuj obiekt typu Subject

```
PUT /api/subjects
```

```
Body Typ Opis

UpdateSubjectDto serialized json Wymagane. Dane tworzonego przedmiotu

Przykład body UpdateSubjectDto

{
    "Name": "Projekt zespołowy",
    "Description": "This is a subject",
    "StartTime": "2022-01-01:14:42:34",
    "EndTime": "2022-01-01:15:42:34",
    "WeekDaysId": 3,
    "ECTS": 4,
    "TeacherId": 5
}
```

Usuń obiekt typu Subject o podanym Id

DELETE /api/subjects/{id}

Parametr Typ Opis

id int Wymagane. Id przedmiotu