



PastaBin

TAlIB Semestr 6

Wojciech Kubowicz, Damian Barczak, Maciej Wielgosz

Opis projektu

Celem projektu jest napisanie aplikacji internetowej umożliwiającej dzielenie się tekstem oraz zdjęciami za pomocą krótkiego wygenerowanego kodu dostępu. Ma ona możliwość zalogowania się w celu dostępu przez użytkownika do jego utworzonych „Past” i zarządzania nimi.

Diagram ERD

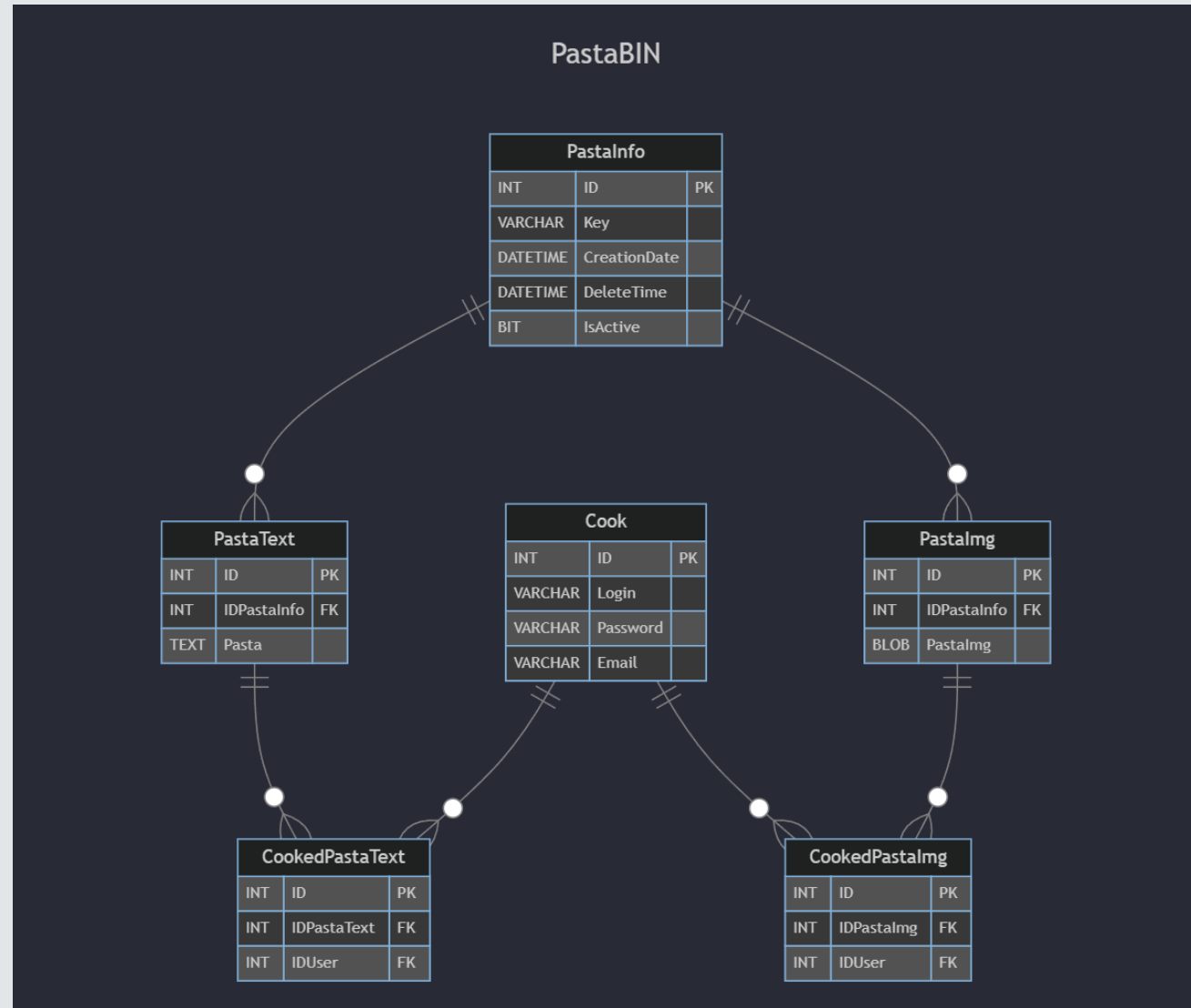
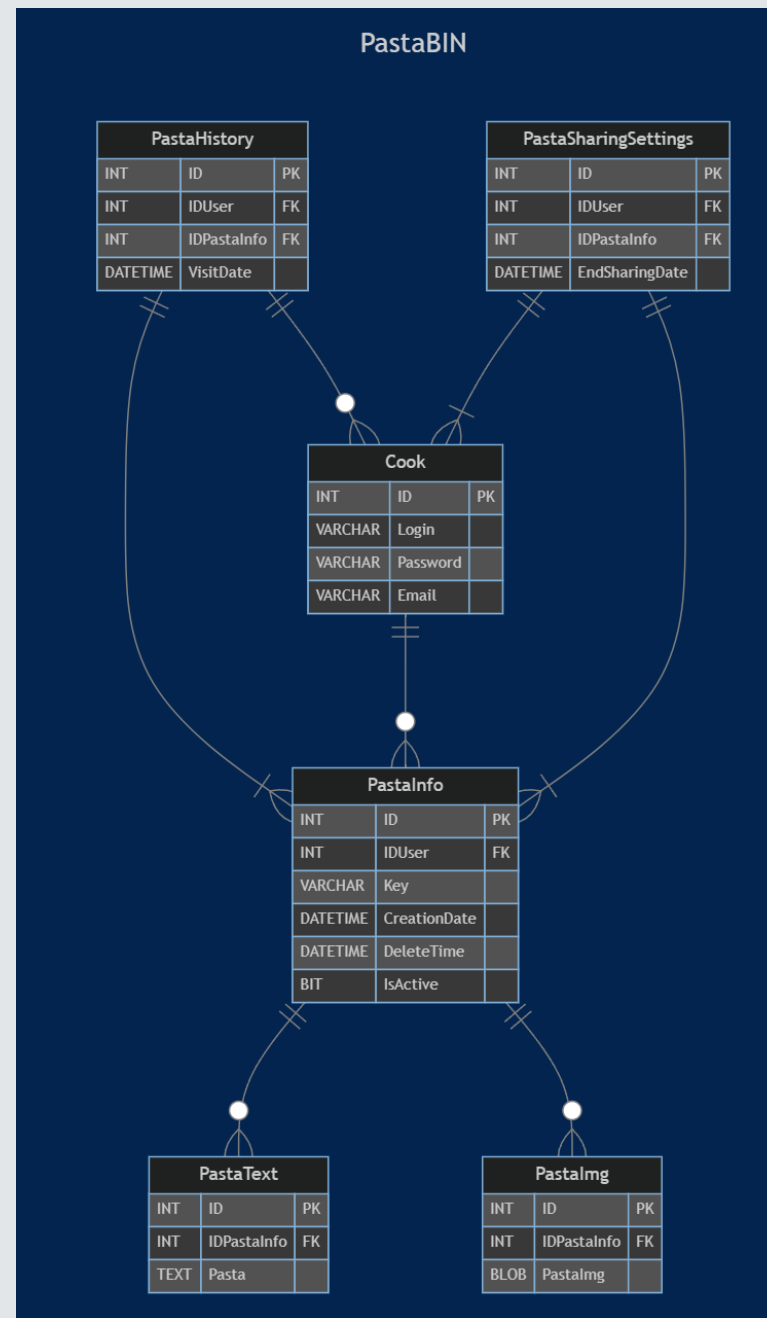


Diagram ERD

Nowa wersja



BLL

▼ BLL

▼ DTO

- 🔗 CookRequestDTO.cs
- 🔗 CookResponseDTO.cs
- 🔗 PastaHistoryRequestDTO.cs
- 🔗 PastaHistoryResponseDTO.cs
- 🔗 PastalngRequestDTO.cs
- 🔗 PastalngResponseDTO.cs
- 🔗 PastalngRequestDTO.cs
- 🔗 PastalngResponseDTO.cs
- 🔗 PastaInfoRequestDTO.cs
- 🔗 PastaInfoResponseDTO.cs
- 🔗 PastaSharingSettingsRequestDTO.cs
- 🔗 PastaSharingSettingsResponseDTO.cs
- 🔗 PastaTextRequestDTO.cs
- 🔗 PastaTextResponseDTO.cs

▼ Interface

- 🔗 ICook.cs
- 🔗 IPastaHistory.cs
- 🔗 IPastalng.cs
- 🔗 IPastaInfo.cs
- 🔗 IPastaSharingSettings.cs
- 🔗 IPastaText.cs

🔗 BLL.csproj

🔗 CookRequestDTO.cs

BLL > DTO > 🔗 CookRequestDTO.cs > {} BLL

```
1 namespace BLL
2 {
3     public class CookRequestDTO
4     {
5         public string Login { get; set; }
6         public string Password { get; set; }
7         public string Email { get; set; }
8     }
9 }
10
```

🔗 CookResponseDTO.cs

BLL > DTO > 🔗 CookResponseDTO.cs > ...

```
1
2 namespace BLL
3 {
4     public class CookResponseDTO
5     {
6         public int ID { get; set; }
7         public string Login { get; set; }
8         public string Password { get; set; }
9         public string Email { get; set; }
10    }
11
12
13
14 }
```

🔗 ICook.cs

BLL > Interface > 🔗 ICook.cs > ...

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace BLL
8 {
9     public interface ICook
10    {
11        public IEnumerable<CookResponseDTO> GetCooks();
12        public CookResponseDTO GetCook(int id);
13        public void DeleteCook(int id);
14        public void PutCook(int id, CookRequestDTO cookRequest);
15        public void PostCook(CookRequestDTO cookRequest);
16    }
17 }
18
```

BLL-EF

▼ BLL-EF

- 🔥 BLL-EF.csproj
- 🔗 CookService.cs
- 🔗 PastaHistoryService.cs
- 🔗 PastaImgService.cs
- 🔗 PastaInfoService.cs
- 🔗 PastaSharingSettingsService.cs
- 🔗 PastaTextService.cs

```
CookService.cs X
BLL-EF > CookService.cs > ...
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6  using BLL;
7  using DAL;
8  using Model;
9
10 namespace BLL_EF
11 {
12     public class CookService : ICook
13     {
14         private readonly PastaBINContext _context;
15
16         public CookService(PastaBINContext context)
17         {
18             _context = context;
19         }
20
21         public IEnumerable<CookResponseDTO> GetCooks()
22         {
23             return _context.Cooks
24                 .Select(c => new CookResponseDTO
25                 {
26                     ID = c.ID,
27                     Login = c.Login,
28                     Password = c.Password,
29                     Email = c.Email
30                 }).ToList();
31         }
32
33         public CookResponseDTO GetCook(int id)
34         {
35             var cook = _context.Cooks
36                 .Where(c => c.ID == id)
37                 .Select(c => new CookResponseDTO
38                 {
39                     ID = c.ID,
40                     Login = c.Login,
41                     Password = c.Password,
42                     Email = c.Email
43                 }).FirstOrDefault();
44
45             return cook;
46         }
47     }
48 }
```

```
47
48     public void DeleteCook(int id)
49     {
50         var cook = _context.Cooks.Find(id);
51         if (cook != null)
52         {
53             _context.Cooks.Remove(cook);
54             _context.SaveChanges();
55         }
56     }
57
58     public void PutCook(int id, CookRequestDTO cookRequest)
59     {
60         var cook = _context.Cooks.Find(id);
61         if (cook != null)
62         {
63             cook.Login = cookRequest.Login;
64             cook.Password = cookRequest.Password;
65             cook.Email = cookRequest.Email;
66             _context.SaveChanges();
67         }
68     }
69
70     public void PostCook(CookRequestDTO cookRequest)
71     {
72         var cook = new Cook
73         {
74             Login = cookRequest.Login,
75             Password = cookRequest.Password,
76             Email = cookRequest.Email
77         };
78         _context.Cooks.Add(cook);
79         _context.SaveChanges();
80     }
81 }
82
83 }
```

DAL

▼ DAL
 > Migrations
 🔗 DAL.csproj
 📄 PastaBINContext.cs

```
PastaBINContext.cs M X
DAL > PastaBINContext.cs > ...
1  using Microsoft.EntityFrameworkCore;
2  using Model;
3
4  namespace DAL
5  {
6      public class PastaBINContext : DbContext
7      {
8          public DbSet<Cook> Cooks { get; set; }
9          public DbSet<PastaInfo> PastaInfos { get; set; }
10         public DbSet<PastaText> PastaTexts { get; set; }
11         public DbSet<PastaImg> PastaImgs { get; set; }
12         public DbSet<PastaHistory> PastaHistories { get; set; }
13         public DbSet<PastaSharingSettings> PastaSharingSettings { get; set; }
14
15         protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)
16         {
17             //optionsBuilder.UseSqlServer("Data Source=(localdb)\\MSSQLLocalDB;Initial Cat
18             optionsBuilder.UseSqlServer("Data Source=MACIEJ;Initial Catalog=PastaBin;Integ
19         }
20
21     }
22 }
23
24
```

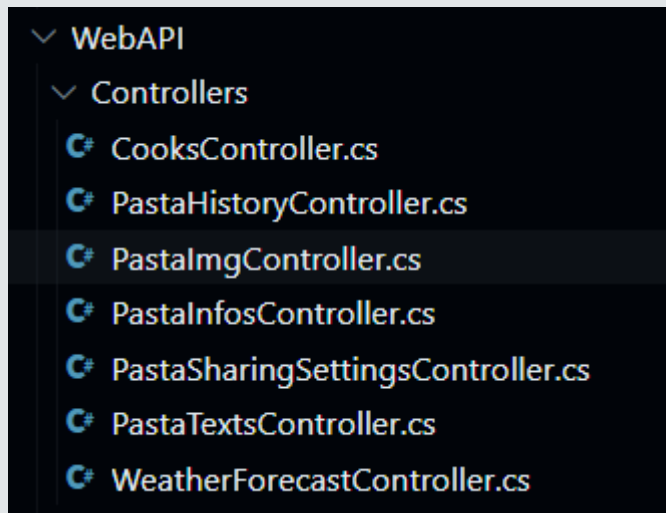
Model

Model

- Cook.cs
- Model.csproj
- PastaHistory.cs
- PastaImg.cs
- PastaInfo.cs
- PastaSharingSettings.cs
- PastaText.cs

```
Cook.cs x
Model > Cook.cs > ...
1 using System.ComponentModel.DataAnnotations;
2 using System.ComponentModel.DataAnnotations.Schema;
3 using Microsoft.EntityFrameworkCore;
4 using Microsoft.EntityFrameworkCore.Metadata.Builders;
5
6
7 namespace Model
8 {
9     [Table("Cook")]
10    public class Cook : IEntityTypeConfiguration<Cook>
11    {
12        [Key]
13        public int ID { get; set; }
14
15        [Required]
16        [MaxLength(255)]
17        public string Login { get; set; }
18
19        [Required]
20        [MaxLength(255)]
21        public string Password { get; set; }
22
23        [MaxLength(255)]
24        public string Email { get; set; }
25
26        public ICollection<PastaInfo> PastaInfos { get; set; }
27        public ICollection<PastaHistory> PastaHistories { get; set; }
28        public ICollection<PastaSharingSettings> PastaSharingSettings { get; set; }
29
30        public void Configure(EntityTypeBuilder<Cook> builder)
31        {
32            builder.HasMany(p => p.PastaInfos)
33                .WithOne(p => p.Cook)
34                .HasForeignKey(p => p.IDUser)
35                .OnDelete(DeleteBehavior.Cascade);
36
37            builder.HasMany(p => p.PastaHistories)
38                .WithOne(p => p.User)
39                .HasForeignKey(p => p.IDUser)
40                .OnDelete(DeleteBehavior.Cascade);
41
42            builder.HasMany(p => p.PastaSharingSettings)
43                .WithOne(p => p.User)
44                .HasForeignKey(p => p.IDUser)
45                .OnDelete(DeleteBehavior.Cascade);
46        }
47    }
48 }
49
```


Kontrolery



```
CooksController.cs X
WebAPI > Controllers > CooksController.cs > ...
1  using Microsoft.AspNetCore.Mvc;
2  using BLL;
3
4  namespace WebAPI.Controllers
5  {
6      [ApiController]
7      [Route("[controller]")]
8      public class CooksController : ControllerBase
9      {
10         private readonly ICook _cookService;
11
12         public CooksController(ICook cookService)
13         {
14             _cookService = cookService;
15         }
16
17         [HttpGet]
18         public IEnumerable<CookResponseDTO> GetCooks()
19         {
20             return _cookService.GetCooks();
21         }
22
23         [HttpGet("{id}")]
24         public CookResponseDTO GetCook(int id)
25         {
26             var cook = _cookService.GetCook(id);
27             return cook;
28         }
29
30         [HttpPost]
31         public void PostCook([FromBody] CookRequestDTO cookRequest)
32         {
33             _cookService.PostCook(cookRequest);
34         }
35
36         [HttpPut("{id}")]
37         public void PutCook(int id, CookRequestDTO cookRequest)
38         {
39             _cookService.PutCook(id, cookRequest);
40         }
41
42         [HttpDelete("{id}")]
43         public void DeleteCook(int id)
44         {
45             _cookService.DeleteCook(id);
46         }
47     }
48 }
```

Swagger - 1

Cooks

GET /Cooks

POST /Cooks

GET /Cooks/{id}

PUT /Cooks/{id}

DELETE /Cooks/{id}

PastaHistories

GET /PastaHistories

POST /PastaHistories

GET /PastaHistories/{id}

PastaImgs

GET /PastaImgs

POST /PastaImgs

GET /PastaImgs/{id}

DELETE /PastaImgs/{id}

PUT /PastaImgs/{id}

Swagger - 2

PastaInfos

GET /PastaInfos

POST /PastaInfos

GET /PastaInfos/{id}

DELETE /PastaInfos/{id}

PUT /PastaInfos/{id}

PastaSharingSettings

GET /PastaSharingSettings

POST /PastaSharingSettings

GET /PastaSharingSettings/{id}

DELETE /PastaSharingSettings/{id}

PUT /PastaSharingSettings/{id}

PastaTexts

GET /PastaTexts

POST /PastaTexts

GET /PastaTexts/{id}

DELETE /PastaTexts/{id}

PUT /PastaTexts/{id}

Swagger - użycie

POST /Cooks

Parameters

No parameters

Request body

```
{
  "login": "test",
  "password": "ZAQ!1qaz",
  "email": "test@test.pl"
}
```

Execute

Responses

Curl

```
curl -X 'POST' \
  'https://localhost:7023/Cooks' \
  -H 'accept: */*' \
  -H 'Content-Type: application/json' \
  -d '{
    "login": "test",
    "password": "ZAQ!1qaz",
    "email": "test@test.pl"
  }'
```

Request URL

https://localhost:7023/Cooks

Server response

Code Details

200

Response headers

```
content-length: 0
date: Thu, 18 Apr 2024 12:31:44 GMT
server: Kestrel
```

Responses

Code Description

200

Success

dbo.Cook [Data]				
PastaBINContext.cs Program.cs CooksC				
Max Rows: 1000				
	ID	Login	Password	Email
	1	test	ZAQ!1qaz	test@test.pl
	NULL	NULL	NULL	NULL



Wojciech Kubowicz, Maciej Wielgosz, Damian Barczak

