# Snackis

Jonas svedmyr

email: [jonas@svedmyr.nu](mailto:jonas@svedmyr.nu)

## Beskrivning

Snackis är ett diskussionsforum där admin kan skapa ämnen och användare kan skapa poster eller kommentera andras poster.

## Tidsöversikt

2021-05-17 - 2021-06-17

Det har fallit bort tid på grund av felbedömningar om hur lång tid vissa saker tog att bygga. Pandemin har även påverkat med att allt arbete har varit på distans och lett till svårigheter att hålla fokus.

Mycket tid i slutet försvann på grund av att få projektet att funka på Azure.

## Tekniska krav

Windows

Ramverk: .NET CORE 5

SQL

Azure

Snackis NuGet packages:

* FluentAssertions
* Newtonsoft.Json
* System.IdentityModel.Tokens.jwt

Snackis Api NuGet packages:

* Microsoft.AspNetCore.Authentication.JwtBearer
* Microsoft.AspNetCore.Identity.EntityFrameworkCore
* Microsoft.EntityFrameworkCore
* Microsoft.EntityFrameworkCore.Design
* Microsoft.EntityFrameworkCore.SqlServer
* Microsoft.EntityFrameworkCore.Tools
* Swashbuckle.AspNetCore

## Applikationsstruktur

Projektet är uppdelat i två delar. Snackis är frontend och är byggt i razor. Snackis är en hemsida som kommunicerar med ett api, Snackis api. Snackis api är byggt i mvc och är till för att data och databasen för att spara och skicka data till frontend.

Snackis api använder sig av autentisering med JWT token som konfigureras i startup.cs. Databasen är konfigurerad i IdentityHostingStartup.cs som ligger i SnackisAPI/Data.

## Installation

för att få igång projektet behövs update-database köras med snackis api som startup project. Alla roller på hemsidan och admin skapas automatiskt när update-database körs.

användarnamnet för admin är admin och lösenordet är AdminPass1! vilken bör bytas efter installationen. När hemsidan är installerad behöver admin skapa ämnen för att användare ska kunna börja använda hemsidan och skapa inlägg.

för att publicera sidan behövs snackis api och snackis publiceras separat till två olika sidor med publish via visual studio.

Connectionstring finns i snackis api, i filen appsettings.json.

I snackis appsettings.json finns bas url för api.

## Endpoints

(Det är 34 endpoints)

### Auth/Register

HttpPost.

Skapar ett konto.

{

"username": "string",

"email": "string",

"agreedToTerms": "string",

"password": "string"

}

Returnerar en sträng om hur det gick.

### Auth/Login

HttpPost

Loggar in en användare och skapar en token.

{

"user": "string",

"password": "string"

}

Returnerar token som sträng.

### Title/Set

HttpPost

Autentiserad kräver token.

Ändrar titeln på hemsidan.

{

"title": "string"

}

Returnerar statuskod.

### Title/Get

HttpGet

hämtar sidans titel

Returnerar

{

"title": "string"

}

### User/Picture/Set

HttpPost

Autentiserad kräver token.

Ändrar profil bildens uri

{

"imageUrl": "string"

}

Returnerar statuskod

### User/Picture/Get

HttpGet

Autentiserad kräver token.

Returnerar

{

"ImageUrl": "string"

}

### Subject/Add

HttpPost

Autentiserad kräver token.

{

"title": "string",

"description": "string"

}

Returnerar statuskod

### Subject/Get

HttpGet

Returnerar alla ämnen

[

{

"id": "string",

"title": "string",

"description": "string"

}

]

### Subject/Get/{Id}

HttpGet

Returnerar ett ämne

{

"id": "string",

"title": "string",

"description": "string"

}

### Subject/Edit

HttpPost

Autentiserad kräver token.

{

"id": "string",

"title": "string",

"description": "string"

}

Returnerar statuskod

### Subject/Delete/{Id}

HttpDelete

Autentiserad kräver token.

Tar bort ett ämne

Returnerar statuskod

### Chat/Message/Create

HttpPost

Autentiserad kräver token.

Skapar ett meddelande och skickar det till användaren som skickats med.

{

"userId": "string",

"message": "string"

}

Returnerar statuskod

### Chat/Get/{UserId}

HttpGet

Autentiserad kräver token.

hämtar chatten som användaren har med användaren med det id som skickas med eller skapar en ny chatt om det inte finns någon.

Returnerar

{

"id": "string",

"messages":

[

{

"id": "string",

"text": "string",

"sent": ”datetime",

"userId": "string",

"username": "string"

},

]

}

### Chat/Get/All

HttpGet

Autentiserad kräver token.

hämtar alla användare som användaren har en chat med

Returnerar

[

{

"userid": "string",

"user": "string",

}

]

### Post/Create

HttpPost

Autentiserad kräver token.

Skapar ett nytt inlägg under det ämne som skickats med

{

"subjectId": "string",

"title": "string",

"description": "string"

}

Returnerar statuskod

### Post/Edit

HttpPost

Autentiserad kräver token.

redigerar ett inlägg

{

"id": "string",

"description": "string",

"title": "string"

}

Returnerar statuskod

### Post/Get/{Id}

HttpGet

hämtar ett inlägg

Returnerar

{

"id": "string",

"title": "string",

"description": "string"

}

### Post/Get/All/{SubjectId}

HttpGet

hämtar alla inlägg under ett ämne

[

{

"id": "string",

"title": "string",

"description": "string"

}

]

Returnerar sidans titel

### Post/Delete/{Id}

HttpDelete

Autentiserad kräver token.

Tar bort en post

Returnerar statuskod

### Post/Report/Create

HttpPost

Autentiserad kräver token.

skapar en rapport om den posten som skickats med

{

"postId": "string",

"reason": "string"

}

Returnerar statuskod

### Post/Report/Get/{Id}

HttpGet

Autentiserad kräver token.

Hämtar en rapport

Returnerar

{

"id": "string",

"reason": "string",

"userid": "string",

"username": "string",

"reportedPost":

{

“id": "string",

"title": "string",

"description": "string",

"username": "string",

},

}

### Post/Report/Get/All

HttpGet

Autentiserad kräver token.

hämtar alla rapporterade poster

Returnerar

[

{

"id": "string",

"reason": "string",

"userid": "string",

"username": "string",

"reportedPost":

{

“id": "string",

"title": "string",

"description": "string",

"username": "string",

},

}

]

### Post/Report/Remove/{Id}

HttpDelete

Autentiserad kräver token.

Tar bort en rapport

Returnerar statuskod

### Post/Report/Remove/Post/{id}

HttpDelete

Autentiserad kräver token.

Tar bort en rapporterad post

Returnerar statuskod

### Comments/Create

HttpPost

Autentiserad kräver token.

Skapar en kommentar till det inlägg som skickats med

{

"postId": "string",

"comment": "string"

}

Returnerar statuskod

### Comments/Edit

HttpPost

Autentiserad kräver token.

Redigerar en kommentar

{

"commentId": "string",

"comment": "string"

}

Returnerar statuskod

### Comments/Get/{id}

HttpGet

Hämtar en kommentar

Returnerar

{

"id": "string",

"comment": "string",

"userid": "string",

"username": "string",

"posted": "string"

}

### Comments/Get/All/{postId}

HttpGet

Hämtar alla kommentarer som ett inlägg har

Returnerar

[

{

"id": "string",

"comment": "string",

"userid": "string",

"username": "string",

"imageUri": "string",

"posted": "string"

}

]

### Comments/Delete/{Id}

HttpDelete

Autentiserad kräver token.

Tar bort en kommentar

Returnerar statuskod

### Comments/Report/Create

HttpPost

Autentiserad kräver token.

skapar en rapport

{

"commentId": "string",

"reason": "string"

}

Returnerar statuskod

### Comments/Report/Get/{id}

HttpGet

Autentiserad kräver token.

Hämtar en rapport för en kommentar

Returnerar

{

"id": "string",

"reason": "string",

"userid": "string",

"username": "string",

"reportedComment":

{

“id": "string",

"text": "string",

"username": "string",

},

}

### Comments/Report/Get/All

HttpGet

Autentiserad kräver token.

Hämtar alla rapporter för rapporterade kommentarer

Returnerar

[

{

"id": "string",

"reason": "string",

"userid": "string",

"username": "string",

"reportedPost":

{

“id": "string",

"text": "string",

"username": "string",

},

}

]

### Comments/Report/Remove{id}

HttpDelete

Autentiserad kräver token.

Tar bort en rapport

Returnerar sidans titel

### Comments/Report/Remove/Comment/{id}

HttpDelete

Tar bort en rapporterad kommentar

Returnerar sidans titel

## Funktionalitet

### Inloggning

* Microsoft Identity och JWT

### Ämnen

* Lägga till
* Editera
* Ta bort

### Registrering

* Användarnamn, Email

### Poster

* Lägga till
* Editera
* Ta bort
* Rapportera

### Profil

* Byta profilbild

### kommentarer

* Lägga till
* Editera
* Ta bort
* Rapportera

### Titel

* Ändra titel på sidan

### Chat

* Skapa
* Skicka meddelanden

### Rapportering

* Ta bort rapporterade poster och kommentarer
* Ta bort rapporter

## Brister

* Token för inloggning ligger i session just nu. Det är inte säkert eftersom det går enkelt att läsa av session.
* Efter att Snackis Api har autentiserat token, autentiseras användaren bara genom att kolla om användaren finns i databasen.
* Går inte att byta lösenord
* Går inte att ta bort poster om den är rapporterad eller om en kommentar är rapporterad
* Går inte att ta bort kommentarer om dom är rapporterade
* Går inte att ta bort ämnen om dom har poster

## Förbättringsförslag

* Verifiera användarens email
* Användarflöde
* Responsiv chat
* att hämta en användares profil bild borde inte vara autentiserad

## Summering

Projektet är uppdelat i två delar Snackis och SnackisAPI. De båda delarna behöver publiceras separat. Admin skapas automatiskt och har ett lösenord som bör bytas. Ämnen skapas inte automatisk utan admin behöver skapa nya ämnen för att användare ska kunna skapa inlägg.

Under arbetet har det gått bra men fokus har varit ett problem då allt arbete har varit på distans. Jag stötte inte på några större problem förrän jag skulle ladda upp projektet på Azure. Anledningen till att Azure var problematisk var att autentiserade post endpoints inte funkade. Anledningen var att identity server hade en roll som hette RoleManager i en tidigare version som inte längre finns. När jag laddade upp projektet och kör projektet försöker den använda RoleManager. Lösningen var att skriva en web.config fil för att ta bort RoleManager. När jag hade löst det insåg jag att jag glömt att tänka på alla relationer när jag tog bort objekt ur databasen. Jag spenderade den sista tiden på att fixa det men jag fick problem och han inte fixa det.

Det jag har lärt mig under projektet är vikten av att planera ordentligt. Det som var planerat bra gick fort att göra men allt annat tog lång tid att göra. En till sak som jag lärde mig var att saker tar alltid längre att göra än planerat.