

Projektarbete

Web API (20p)

EC Utbildning

Beskrivning	3
Inlämning	3
Betygsättning	3
Instruktioner	4
Fördefinierat projekt	4
Godkänd (G)	4
Logga in	4
Radera spel	5
Dokumentation	6
Muntligt förhör	6
Väl Godkänd (VG)	7
Behörighetskontroll	7
Uppdatera spel	8
Versionera API	9
Rapport	9
Eget projekt	10
Godkänd (G)	10
Väl Godkänd (VG)	11
FAQ	12
Får man använda scaffolding?	12
Får man jobba i grupp?	12

Beskrivning

I detta obligatoriska projektarbetet visar du att du besitter kunskap inom koncept som tagits upp under kursen.

Inlämning

När du är klar med uppgiften och **verifierat att samtliga kriterier för Godkänd (G) alternativt Väl Godkänd (VG) är uppfyllda**, skapa en Zip- eller RAR-fil innehållandes samtliga projektfiler och ladda upp denna till Ping Pong (Examinering > [Projektarbete](#)).

Därefter sätter du upp dig för [redovisning](#).

Inlämning sker senast **fredag 20/5 kl 8:30**. Missar du inlämningstiden kontaktar du utbildningsledaren.

Betygsättning

Du kan erhålla betyg Väl Godkänd (VG) eller Godkänd (G) på projektarbetet.

Instruktioner

Du har möjlighet att välja ett fördefinierat projekt, alternativt ta fram ett eget. Valde du eget projekt under kursen "ASP.NET Core" bygger du förslagsvis vidare på detta genom att lägga till ett webb-API enligt instruktioner nedan.

Fördefinierat projekt

Godkänd (G)

Logga in

Man måste logga in för att komma åt menyn i HighscoreConsole. När man startar applikationen visas följande vy:

Användare:

Lösenord:

Uppgifterna skickas till Highscore API, där kontroll av användarnamn och lösenord görs. Om dessa är felaktigt visas följande i 2 sekunder:

Felaktiga uppgifter, försök igen

Därefter visas inloggningsvyn på nytt.

Om användarnamn och lösenord är korrekt genereras en JWT (JSON Web Token) och skickas tillbaka till applikationen, varpå huvudmenyn visas.

Applikationen skickar nu med denna vid varje anrop som görs till API:et.

NOTERA: Det ska inte gå att använda Highscore API om man inte har en giltig token.

Radera spel

Man ska kunna radera spel, vilket görs via menyalternativ "2. Sök spel". Exempelvis, söker man efter "Te" skulle det kunna se ut så här:

```
ID  Titel
-----
1   Tetris
2   Extreme Tetris
```

Som tidigare, trycker man på Esc kommer man tillbaka till huvudmenyn.

Vi har nu ett meny-alternativ här för att radera spel:

```
ID  Titel
-----
1   Tetris
2   Extreme Tetris

(R)adera
```

Trycker man "R" visas följande:

```
ID  Titel
-----
1   Tetris
2   Extreme Tetris

ID:
```

"(R)adera" byts ut mot "ID:" med en prompt som låter användaren ange ID på spelet som ska raderas. Ett anrop skickas till Highscore API för att radera spelet och därefter visas följande i 2 sekunder:

```
Spel raderat
```

Därefter visas huvudmenyn på nytt.

Dokumentation

Highscore API ska förse utvecklaren med automatiskt genererad dokumentation, vilket görs med [Open API](#). Gör nödvändiga ändringar i Highscore API för att få detta på plats.

Muntligt förhör

Du förväntas kunna svara på följande frågor:

- Beskriv vad REST är
- Versionera API (alltså underhålla flera versioner av ett API)
 - Förklara varför det kan vara nödvändigt att versionera API
 - Ge exempel på hur man kan versionera API

Väl Godkänd (VG)

Börja med att implementera det som krävs för att uppnå Godkänd (G). Instruktioner för detta hittar du ovan.

Behörighetskontroll

Vi har ofta olika typer av användare i ett system som har olika behörigheter. När man startar HighscoreConsole och loggar in som en vanlig användare (som inte har rollen "Administrator"), så visas följande meny:

1. Sök spel
2. Avsluta

När man söker efter spel listas sökresultatet - alternativ för att radera spel visas inte:

ID	Titel

1	Tetris
2	Extreme Tetris

Loggar man däremot in som en användare som har rollen "Administrator" kan man även göra följande:

- Registrera highscore
- Lägga till spel
- Ta bort spel

Uppdatera spel

Man ska kunna uppdatera ett spel, vilket sker via "Sök spel". När man söker efter ett spel visas samtliga träffar. Längst ner har vi nu ett alternativ för att uppdatera ett spel:

```
ID    Titel
-----
1     Tetris
2     Extreme Tetris

(R)adera (U)ppdatera
```

När man väljer "U" visas en vy som låter oss välja spel baserat på dess ID:

```
ID>
```

Därefter kan man ange nya detaljer för spelet:

```
ID: 1 (denna sätter användaren inte själv - skriv ut spelets ID)
Titel:
Beskrivning:
Lanseringsår:
Genre:
Bild URL:

Är detta korrekt? (J/N)
```

Notera att "ID" inte är ett värde som kan sättas av användaren - vi skriver enbart ut värdet här för att visa att detta är ett existerande spel som vi uppdaterar.

Använd förslagsvis PUT (finns även PATCH) - och i detta fallet är det så att vi behöver ange och skicka med samtlig information för spelet, oavsett om denna har förändrats eller ej.

Alltså, om vi enbart vill ändra spelets titel, behöver vi ändå ange resten av informationen på nytt, då PUT ersätter spelet i sin helhet (med undantag för ID).

Det går även bra att använda PATCH, dock kommer koden bli mer komplex.

Versionera API

Implementera en versioneringslösning - du behöver ha denna på plats i API:et, men behöver inte göra någon ändring i HighscoreConsole för att använda dig av olika versioner av API:et. Du kan använda Insomnia för att visa att versionering fungerar i API:et.

Rapport

- Lämna in rapport som "Skriftligen reflektera över, och analysera de lösningar som valts i projektet gällande behörighetskontroll, versioneringsstrategi och dokumentation" (tagit från Kursplanen).

Eget projekt

Väljer du eget projekt, boka tid med läraren för säkerställa att du fått med allt för eftersträvad betygsnivå.

Godkänd (G)

För betygsnivån krävs följande:

- Via webb-API:et ska man kunna
 - Söka resurs
 - Skapa resurs
 - Radera resurs
 - Logga in (generera token)
- Man måste logga in i konsol-applikationen (token) för att kunna använda den och samtliga endpoints ska vara skyddade.
- Dokumentation för webb-API genereras automatiskt (Open API)
- Svara på följande frågor
 - Beskriv vad REST är
 - Versionera API (alltså underhålla flera versioner av ett API)
 - Förklara varför det kan vara nödvändigt att versionera API
 - Ge exempel på hur man kan versionera API

Väl Godkänd (VG)

För betygsnivån krävs följande:

- Samtliga kriterier för Godkänd (G) har uppfyllts (se ovan).
- Via webb-API:et ska man kunna
 - Uppdatera resurs
- Behörighetskontroll
 - En användare som INTE har rollen "Administrator" kan
 - Söka resurs
 - En användare som har rollen "Administrator" kan
 - Söka resurs
 - Skapa resurs
 - Uppdatera resurs (PUT eller PATCH)
 - Radera resurs
- Ska finnas versionering på plats i API:et - du behöver dock inte använda dig av denna i din konsolapplikation, men du behöver visa att den fungerar via t.ex. Insomnia
- Lämna in rapport som "Skriftligen reflektera över, och analysera de lösningar som valts i projektet gällande behörighetskontroll, versioneringsstrategi och dokumentation" (tagit från Kursplanen).

FAQ

Får man använda scaffolding?

Ja det går bra.

Får man jobba i grupp?

Detta är ett individuellt projektarbete, dvs. det förväntas att man tar sig igenom samtliga moment på egen hand i största möjliga mån och presenterar en fungerande lösning för eftersträvad betygsnivå.

Däremot får ni samarbeta och hjälpa varandra. Se dock till att inte bara kopiera varandra rakt av - du behöver förstå vad du gör.