



Kursnamn: Frontendutveckling- Server
Klass: Frontendutvecklare webbsäkerhet
Termin: Termin 1 HT23

REST API (ONLINE MARKET APPLICATION)

INLEDNING

Bakgrundsbeskrivning, frågeställning, avgränsning och mål	<p>Bakgrund: Vår kund har precis lanserat en webbshop för sitt nya varumärke "Drezztore" och har handlat upp ett leverantörsavtal med ett kinesiskt varuhus. Kunden vill att vi ska hjälpa hen med att sätta upp ett API så att produkthanteringen ska bli enklare. I denna uppgift ska ni skapa en REST server för att applicera det vi har lärt oss om API:er, serverprogrammering och JSON</p> <p>Mål: En REST server som kan ta emot GET, POST, PUT, och DELETE request för att hantera produkter i en ehandel-applikation. Applikationen ska också ha en route för att rendera en html sida med alla produkter.</p> <p>Avgränsning: Ni får implementera detta med vilken metod (paket) som helst.</p>
Varför ska ni utföra detta arbete?	<p>Syftet:</p> <ul style="list-style-type: none">• Studenten ska få utökad förståelse för hur hen skapar en REST applikation med Node och NPM• Studenten ska få arbeta med problemlösning och design av JavaScript kod enligt god sed.
Vad ska ni leverera?	<p>Följande ska levereras:</p> <ol style="list-style-type: none">1. Följande ska levereras: 1. En mapp med ett NPM projekt (package.json etc...)<ol style="list-style-type: none">a. Inkludera inte /node_modules i projekt2. Videoinspelning på genomgång av applikation med Postman eller Insomnia (max 10 min) på samtliga routes3. Skriftlig rapport om användning och olika val vid utformning av API:et



ER PROJEKTUPPGIFT

Vad ska ni göra?	<p>Skapa ett REST-API för Node.js med Express som interagerar med en MongoDB-databas för en ehandel-applikation (Obs: applikation behöver inte frontend men dock ska rot-routen (/) rendera html. För detta behövs ingen css).</p> <p>Skriv en skriftlig rapport med en beskrivning av användningen av API:et</p>
------------------	---



Hur ska ni lösa uppgiften?

1. Skapa en mapp med ert projekt (ex: inlämningsuppgift) och lägg på ert skrivbord
2. Öppna terminalen och navigera till mappen (`cd ~/desktop/inlämningsuppgift`)
3. Kör kommandot `"npm init"` i mappen för att initialisera ett npm projekt
4. Installera nu följande npm paket (`cors`, `express`, `mongoose`) och/eller andra paket som du vill använda i projektet
5. Lägg till en fil (`.gitignore`) i mappen med följande innehåll
`node_modules`
6. Nu kan vi initialisera ett git repo och koppla det till github. Stå kvar i mappen i terminalen och följ instruktionerna i tutorialen (<https://www.digitalocean.com/community/tutorials/how-to-push-an-existing-project-to-github>). OBS: se till att repot på github är privat så att ingen kan se eventuella MongoDB credentials osv.

- ☐  **Public**
Anyone on the internet can see this repository. You choose who can commit.
- ☒  **Private**
You choose who can see and commit to this repository.

7. Nu har du en mall för att börja arbeta med ditt REST API. När du lägger in ny kod i projektet glöm inte att committa och pusha till github (du måste stå i mappen i terminalen)
 - a. `git add .`
 - b. `git commit -m "mitt commit meddelande"`
 - c. `git push`
8. För att komma igång börja med att lägg in en route som ger följande meddelande när du loggar in på `http://localhost:3000`

```
← → ↻ localhost:3000
{"message": "Welcome to DressStore application."}
```

Följande CRUD metoder ska stödjas av API:et

Methods	Urls	Actions
GET	<code>api/products</code>	Hämtar alla produkter
GET	<code>api/products/:id</code>	Hämtar produkt med id
POST	<code>api/products</code>	Skapar en ny produkt
PUT	<code>api/products/:id</code>	Ändrar produkt med id
DELETE	<code>api/products/:id</code>	Tar bort produkt med id
DELETE	<code>api/products</code>	Tar bort alla produkter
GET	<code>api/products?name=[kw]</code>	Hämtar produkt vars namn innehåller "kw"
GET	<code>/</code>	Sänder tillbaka html med



			alla produkter i en tabell med all information om produkt
	Varje produkt har <ul style="list-style-type: none">- name (String)- description (String)- price (Number)- quantity (Number)- category (String) Använd Postman eller Insomnia för att testa ert API under utvecklingen.		

INLÄMNING OCH REDOVISNING

Inlämning	Inlämning sker via Learn Point datum 10/12 senast kl. 23:59 <ul style="list-style-type: none">• Video med redovisning (max 10min) laddas upp med mp4 eller länk till zoom (alternativt redovisar ni live för kursansvarig lärare)• Zip fil med alla projekt filer läggs upp på learnpoint plattformen• Skriftlig rapport över användning av API:et
-----------	---

BEDÖMNING OCH ÅTERKOPPLING

Bedömning sker mot följande betygskriterier:	<p>Betygskriterierna för Godkänd respektive Väl godkänd är:</p> <p>Godkänd</p> <ul style="list-style-type: none">• Studenten har uppvisat kunskap och färdigheter i serverprogrammering med Node.js• Studenten har visat kunskap och färdighet i att strukturera ett REST-API• Studenten använder en array i javascript som behållare för alla produkter <p>Väl godkänd</p> <ul style="list-style-type: none">• Studenten har visat kunskap och färdighet i Backendutveckling (koppling med Databas)• Studenten använder MongoDB (i "cloud-atlas" miljö) som behållare för alla produkter.• Studenten har skrivit en rapport där hen förklarar de olika REST-API routerna och hur dessa ska användas, studenten förklarar med hjälp av olika testfall (exempel av användning) och ger nyanserade resonemang för olika utkomster vid anrop mot olika routes, dvs om vad som ska hända t.ex om man skickar fel data när man skickar rätt data osv.
--	--



Återkoppling

I och med redovisningstillfället ges muntlig återkoppling till grupperna.

Grupperna får skriftlig återkoppling via lärplattformen LearnPoint **senast den datum 20/12.**