

Rapport slutprojekt

Yasir Kakar

För min applikation började jag med att skapa min databas i mysql med hjälp av kommandotolken i min dator för att det kändes både smidigare och gick snabbare. Strukturen på databasen speglar anvisningarna för uppgiften.

URL

URLerna som används i projektet går hand i hand med vad applikationen gör. I applikationen har jag fyra olika sidor som man kan komma in på vilket är följande:

- Home
 - Home är första sidan för projektet och har inget extra i URLen.
 - Här kan man se alla tabeller och vad de innehåller.
- Add
 - Add lägger man till data i de olika tabellerna och har två olika URLer.
 - selectAdd: Detta är första URLen vart användaren väljer en tabell.
 - Add: i add så läggs datan till i databasen.
- Update
 - I Update uppdateras data i en column i en tabell och har två URLer.
 - selectUpdate: Användaren väljer en tabell.
 - Update: användaren uppdaterar data i en column.
- Remove
 - I remove raderas det data från en tabell och har två URLer.
 - selectRemove: Användaren väljer en tabell.
 - Remove: Användaren tar bort data med ett id.

WEBGUI

Applikationens WEBGUI är väldigt simpel och har endast fyra sidor. Det är tydliga namn på vad de olika sidorna är till för och vad användaren kan förvänta sig hur de ska användas.

Tanken bakom applikationens WEBGUI är att det ska vara så simpelt som möjligt och att användaren inte ska ha allt för många alternativ. På varje sida kan användaren se alla tabeller som finns på databasen och sedan manipulera data till sin egen önskan.

Användarens alternativ gällande varje tabell är att titta på data, lägga till data, uppdatera data och radera data. Användaren får en relativ tydlig vägledning genom alla sidor med hjälp av texten som står innan alla input och sedan knappen gällande vad användaren kan åstadkomma.

Upplevelse

Utmanande

Det som jag tyckte var utmanande var hantering av json objekt för att få ut rätt data. Min lösning var att omvandla det till en array och sedan använda foreach. Eftersom endpoints skulle visas som json objekt så har jag använt två olika pool objekt där ena är specifikt för WEBGUI och den andra är för endpoints.

Intressant

Det som var intressant var att jag kunde ha igång två servrar på localhost samtidigt. Orsaken till att jag byggde den funktionen var för att jag har delat upp endpoints och WEBGUI i två olika js filer.

Lätt

Hantering av mysql var nog det lättaste då genomgången var ganska tydlig och informationen var väldigt enkel att hitta.

Endpoints

Visar alla studenter:

<http://localhost:3001/students>

Visar alla Kurser:

<http://localhost:3001/courses>

Visar alla associationer:

http://localhost:3001/student_courses

Visar alla studenter med id:

<http://localhost:3001/students/id/:id>

Visar alla studenter med förnamn:

<http://localhost:3001/students/fname/:fname>

Visar alla studenter med efternamn:

<http://localhost:3001/students/lname/:lname>

Visar alla studenter med stad:

<http://localhost:3001/students/town/:town>

Visar alla kurser med id:

<http://localhost:3001/courses/id/:id>

Visar alla kurser med namn:

<http://localhost:3001/courses/name/:name>

Visar alla kurser med del av namnet:

<http://localhost:3001/courses/cname/:cname>

Visar alla kurser med del av beskrivningen:

<http://localhost:3001/courses/description/:description>

