

# Inlämningsuppgift Klientutveckling

---

## Nackademiska Webbutik (Nackbutik)

Gunnars syster Gunhild driver en klädbutik som säljer exklusiva kläder i en butik på rådmansgatan. Gunhild har länge önskat sig en webbutik och har nu haft inne en konsult som har byggt ett REST API till henne som exponerar hennes klädbutiks databas över internet.

Din uppgift blir att bygga hennes webbutiks front-end med hjälp av AngularJS och använda det existerande REST API:et som finns på (<http://nackbutik.azurewebsites.net>).

## Funktionella krav (Godkänt)

- När en besökare kommer till sidan så skall samtliga produkter visas med namn, bild och pris. Produkterna skall kunna sorteras på kategori samt det skall även gå att filtrera/söka efter produkter. Även produkter som är slut i lager skall visas, dock ska det synas att de är slut i lager på något sätt, exempelvis att det står en text, eller att produkten är "utgråad" eller liknande.
- Besökaren skall också kunna klicka på en produkt och se mer detaljer om produkten, som den detaljerade beskrivningen.
- Besökaren kan lägga produkter i en varukorg som skall hanteras i din applikation. När besökaren är nöjd med sina varor så skall denna kunna trycka på en "check out" knapp och skall då komma till en sida där kundvagnens innehåll visas med totalbeloppet och man ska kunna trycka på "beställ" och då skall en order skapas och ett tack meddelande visas. Ingen betalning görs i webbutiken utan kunden får en faktura senare som hanteras utanför detta system.
- Besökaren måste vara inloggad för att kunna beställa, antingen loggar den in innan den checkar ut eller så måste besökaren logga in innan beställningen görs.
- Om besökaren inte har något konto än så skall denne kunna skapa ett nytt konto.

# Funktionella krav (Väl Godkänt)

- När en kund är inloggad så skall det finnas en “kundsida” för den inloggade kunden där denne kan se sina detaljer och ändra sin information.
- På kundsidan skall det också finnas en sammanställning av kundens alla ordrar samt totalbeloppet för alla ordrar.
- Om man klickar på en specifik order så skall denna visas enskilt med all information som finns om ordern samt även totalpriset för ordern.

## Icke funktionella krav

- Det skall finnas validering av alla fält som finns på webbsajten med tydliga feltexter.
- Du skall använda Bootstrap eller annat grafiskt bibliotek.
- Din webbapplikation skall vara en Single Page Application (SPA) där logic, vy och data är tydligt indelade i olika logiska delar och moduler.
  - Du skall ha egna services och/eller components och direktiv
  - Du skall använda dig av routing och routing parametrar
- Du skall använda dig av `npm` och `bower` för ditt projekt, samt projektet får inte innehålla några andra filer än de som används i projektet.

## Betygskriterier

För **godkänt (G)** krävs att samtliga krav förutom de stycken som är markerade med VG i texten ovan är implementerade.

För **väl godkänt (VG)** krävs att samtliga krav är implementerade. Applikationen ska ha god arkitektur och följa best practice när det gäller namngivning och struktur.

# Inlämning

Senaste inlämningstid för din projektuppgift är den 22 mars, 2017.

- Du skall lämna in ditt projekt med samtliga tillhörande filer i inlämningsmappen på kursportalen. OBS katalogerna `bower_components/` och `node_modules/` skall inte vara med i ditt inlämnade projekt, utan alla paket som din applikation använder skall finnas med i `bower.json` och `package.json` så att det går att starta ditt program genom att installera alla paket med `bower install` och `npm install`, sen starta med `npm run start`.
- Din applikation skall vara ZIP:ad till EN fil.
- Din applikation skall också vara publicerad på Microsoft Azure, skicka med en länk till din applikation som en kommentar till din inlämning i kursportalen.

Lycka till!