

# Webbteknik för Ingenjörer

## 5TF019

### OU3 - RAPPORT (G-NIVÅ)

JAVASCRIPT

Länk till hemsida: [Kajutan](#)

#### Lärare

Keni Ren  
Stefan Berglund

#### Student

Simon Lindgren (sili0125)

23 februari 2024

Umeå Universitet

# 1 Inledning

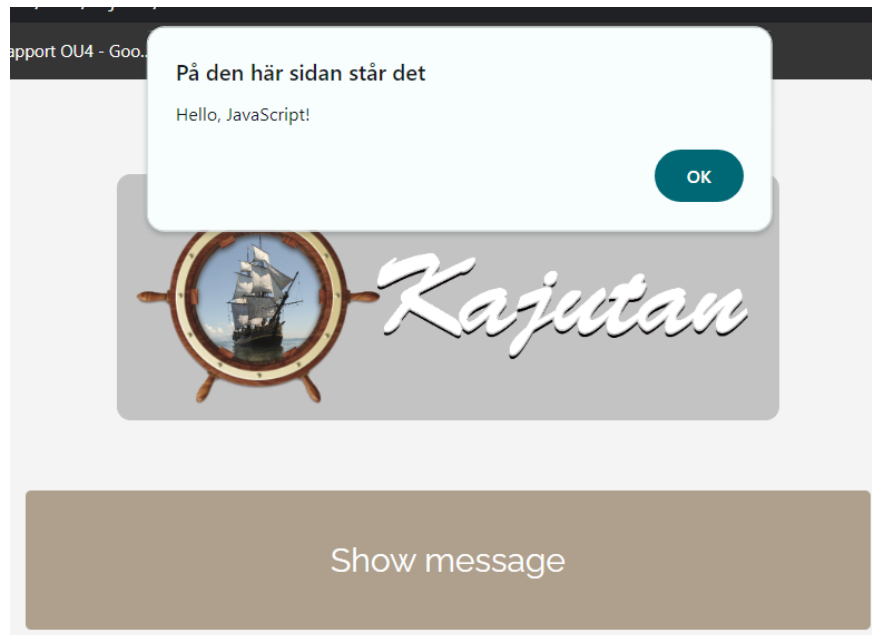
Denna uppgift gick ut på att implementera JavaScript funtionalitet till webbsidan som designades i labb 1 och implementerades med HTML och CSS i labb 2. Hemsidan det handlar om är Kajutan ([www.kajjan.se](http://www.kajjan.se)), som är en restaurang/pizzeria som ligger i Holmsund. Länk till hemsidorna (originalet och nya) finns längst ner i rapporten.

I den här rapporten kommer vi gå igenom genomförandet av uppgiften, vad som har lagts till och titta på hur slutresultatet blev. Som avslutning kommer vi diskutera och relektera på hur uppgiften har gått, vilka problem som har stötts på, samt vad som kunde förbättras.

# 2 Resultat

Uppgiften var uppdelad i fyra mindre uppgifter. Första uppgiften gick ut på att implementera en knapp på hemsidan som aktiverade en popup-ruta med texten "Hello, JavaScript!", resultatet visas i sektion **Uppgift 1**. Andra uppgiften handlade om att implementera en JavaScript funktion som påverkar utseendet på webbsidan, resultatet kan ses i sektion **Uppgift 2**. Tredje uppgiften gick ut på att implementera två olika JavaScript funktioner som påverkar DOM-trädet då användaren utför en interaktion, resultat i sektion **Uppgift 3**. Fjärde och sista uppgiften handlade om jQuery, där man skulle implementera funktioner med hjälp av jQuery, resultatet kan ses i sektion **Uppgift 4**.

## 2.1 Uppgift 1 - Hello, JavaScript!



Figur 1: Popup-ruta då man trycker på knappen "Show message"

Resultatet av första uppgiften visas i figur 1 ovan, där en popup-ruta med texten "Hello, JavaScript!" visas. Rutan poppar upp då man klickar på knappen "Show message". Figurerna 2 och 3 under visar koden i HTML dokumentet och i JavaScript dokumentet.

```
<button onclick="display()" id="javabutton">Show message</button>
```

Figur 2: HTML koden för knappen med texten "Show message"

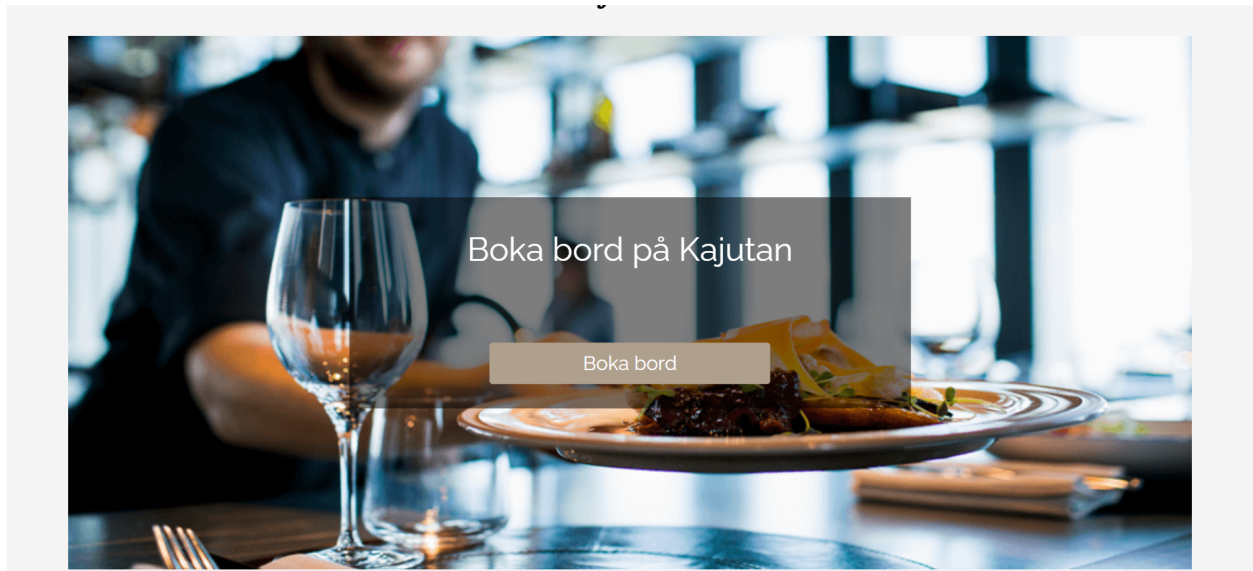
HTML koden för knappen visas i figur 2 ovan. Där den har värdet/texten "Show message", och anropar funktionen `display` (beskrivs nedan i figur 3) vid interaktionen `onclick`.

```
function display() {  
    alert('Hello, JavaScript!');  
}
```

Figur 3: JavaScript koden/funktionen som aktiverar popup-rutan

I figur 3 ovan visas koden för JavaScript funktionen som aktiverar popup-rutan via knapptrycket. En kort funktion `display` som innehåller en alertbox med texten "Hello, JavaScript!", som poppar upp och visas vid anrop.

## 2.2 Uppgift 2 - JavaScript into your HTML



Figur 4: Automatiskt bildspel på Kajutan hemsidan

Resultatet av andra uppgiften visas i figur 4 ovan, där ett bildspel är implementerat. Bildspelet byter automatiskt bild var 10 sekund, och roterar mellan sex stycken olika bilder. JavaScript koden/funktionen över bildspelet visas nedan i figur 5.

```

let index = 0;
slideshowImages();

function slideshowImages() {

    let i;
    const images = document.getElementsByClassName("image");

    for (i = 0; i < images.length; i++) {
        images[i].style.display = "none";
    }

    index++;

    if (index > images.length) {
        index = 1;
    }

    images[index-1].style.display = "block";

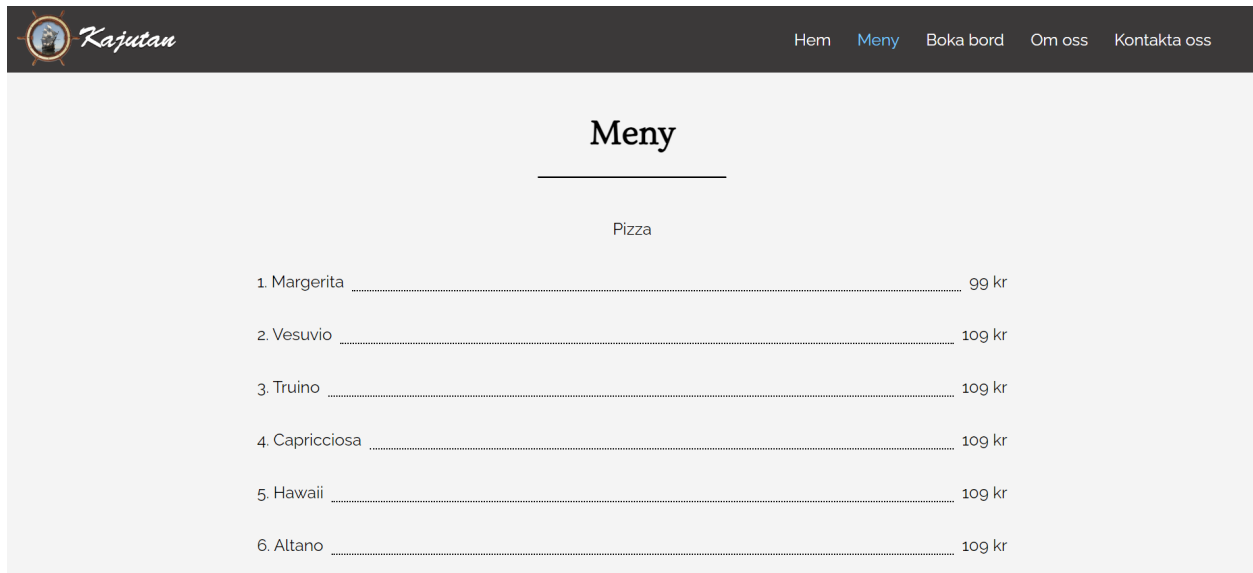
    // 10 sec timer
    setTimeout(slideshowImages, 10000);
}

```

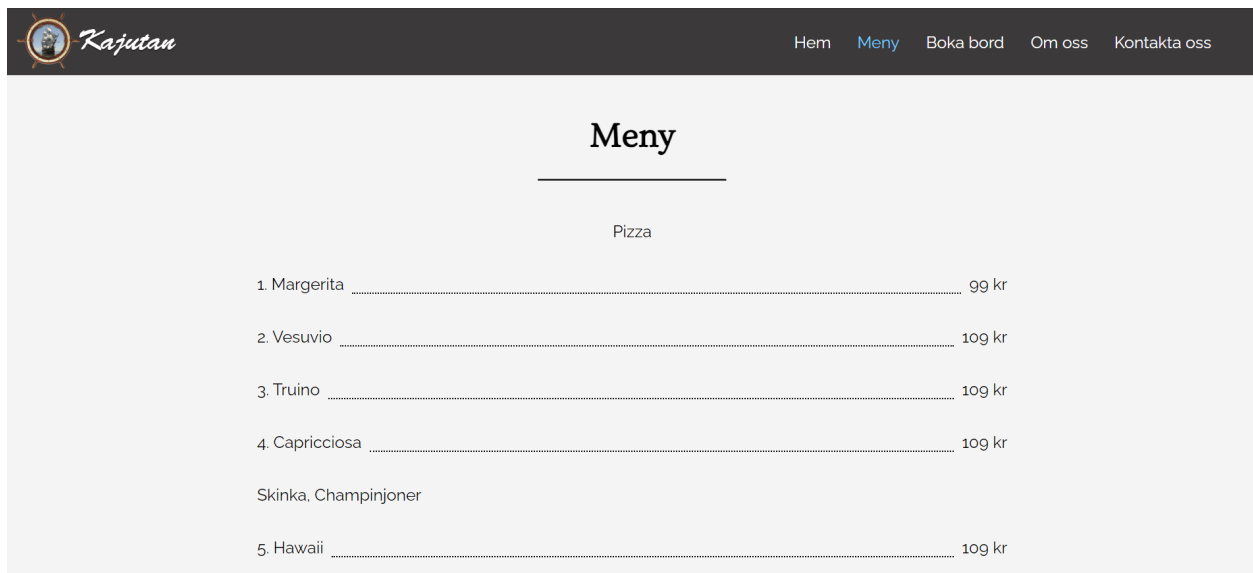
Figur 5: JavaScript koden över bildspelet

I figur 5 ovan så visas JavaScript koden över det automatiska bildspelet. Koden börjar med att sätta ett index värde till 0, sedan starta funktionen `slideshowImages`. Funktionen hämtar ut alla bilder (med klassnamnet 'images') och sätter `display none` på alla bilder, vilket resulterar i att bilderna göms undan. Där efter ökar `index` med 1 och visar upp bilden som är på plats `index - 1`. Ifall `index` skulle bli högre än antalet bilder, så sätts `index` till 1 igen, vilket resulterar i att bilderna loopar runt och runt. Funktionen avslutar med att sätta en timer som ska anropa på funktionen igen då timern gått ut, tiden som är satt är 10000 ms, vilket motsvarar 10 sekunder.

## 2.3 Uppgift 3 - JavaScript and DOM



Figur 6: Matmenyn innan ingrediens expandering



Figur 7: Matmenyn efter ingrediens expandering på pizzan nr 4. Capricciosa

Figurerna 6 och 7 ovan visar JavaScript funktionen över ingrediens expanderings information för maträtter. Som figur 6 visar finns endast information om maträtternas namn, men då man klickar på en maträtt (exempelvis pizza nr 4. Capricciosa) så poppar det up vilka ingredienser maträtten innehåller. Där vi ser i figur 7, att pizzan **Capricciosa** innehåller ingredienserna skinka och champinjoner. Man kan klicka upp flera ingrediens informationer

som visas samtidigt, samt kan man stänga informationen genom att klicka på maträttens namn igen. Implementationsmässigt är det endast pizzorna 2–10 som expanderar ingrediens information, pga enkelhet, men är implementerat så det enkelt kan läggas till information till övriga maträtter. JavaScript-koden visas nedan i figur 8

```
function vesuvio(string, number, id) {  
  
    var info = document.getElementById(id);  
  
    if(!info) {  
        var newElement = document.createElement("p");  
        newElement.id = id;  
        var text = document.createTextNode(string);  
        newElement.setAttribute("style", "color: black; font-size: 20px;");  
        newElement.appendChild(text);  
        var divElement = document.getElementsByClassName("showmore")[number];  
        divElement.appendChild(newElement);  
    }  
  
    else {  
        var textRemove = document.getElementById(id);  
        textRemove?.remove();  
    }  
}
```

Figur 8: JavaScript koden över expanderingen av ingrediens informationen

Figur 8 visar JavaScript koden för ingrediens informations expanderingen, som visades i figur 7. Funktionen tar in en sträng, som representerar ingredienserna som ska skrivas ut, ett nummer, som representerar i vilken div-kontainer p-taggen ska placeras, sen ett id, som representerar id:t på p-taggen. Först kontrolleras ifall id:t existerar, ifall det gör det, så tas elementet bort (ifall man vill ta bort ingrediens informationen), annars skapas en ny p-tagga med det id:et. Strängen som skickas in sätts som p-taggens innehåll, sedan placeras p-taggen innanför det valda div-nummret.

Implementationen använder sig av DOM, då den skapar en p-tagga och placerar den som barn till en div. Då man tar bort informationen, raderas p-taggen, via ett id den fått vid initialisering.



# Boka bord

---

Boka bord på Kajutan

Namn \*

Telefonnummer & Antal personer \*

Antal personer (max 20)

Datum och tid \*

åååå-mm-dd --:--

Meddelande (200 tecken)

Boka

Figur 9: Boka bord sidan innan försök av bokning

# Boka bord

---

Boka bord på Kajutan

Namn \*

Du måste ange ett namn

Telefonnummer & Antal personer \*

09040903

1

Datum och tid \*

2024-02-23 17:00

Meddelande (188 tecken)

Hello World!

Boka

Figur 10: Boka bord sidan efter försök av bokning

En annan funktion visas i figurerna 9 och 10 ovan som innehåller validering vid bokning av bord. Där användaren måste skriva in namn, telefonnummer, antalet personer och datum för att bokningen ska skickas. Som exempel ser vi att i figur 10 så dyker det upp en röd text under namninmatningsrutan som säger "Du måste ange ett namn", och detta för att namnrutan är tom. Ifall användaren då fyller i ett namn också så är bokningen godkänd. Samma sak händer då användaren inte fyller i eller anger ett ogiltigt telefonnummer, då dyker det upp en röd text under telefonnummerrutan som säger "Du måste ange ett giltigt telefonnummer". Kontrollen av det giltiga telefonnumret kontrolleras med hjälp av ett regex format. Antalet personer kontrolleras också, så det är ifyllt och är mer än 0 personer och max 20 personer, annars dyker det upp en röd text som felmeddelande. Datumet fungerar likadant som de tidigare fälten, måste ange ett datum (går inte att ange ett passerat datum) annars poppar felmeddelande upp. En annan kontroll är i meddelanderutan, som max får innehålla 200 tecken, och som uppdateras efter varje tecken man fyller i eller tar bort, för att hjälpa användaren att hålla koll. Ifall användaren skulle försöka mata in mer än 200 tecken så går det inte, då det tar stopp vid 200. Även kontaktformuläret på sidan "Kontakta oss" innehåller validering som är i princip exakt likadan, bara att den har namn, email och meddelande. Där namn fungerar likadant, email kontrollerar att det är ett giltigt format med ett annat regex format, samt meddelande är likadant med max 200 tecken.

## 2.4 Uppgift 4 - jQuery



Figur 11: Lunchmatsedeln över nuvarande vecka (Vecka 6)



Figur 12: Lunchmatsedeln över nästkommande vecka (Vecka 7)

Figureerna 11 och 12 ovan visar JQuery funktionen som byter lunchmatsedel. Då man befinner sig på den nuvarande veckan så kan man klicka på knappen som har texten "Nästa vecka" för att få upp lunchmatsedeln över nästkommande vecka. Befinner man sig på nästkommande vecka trycker man på samma knapp som då har texten "Nuvarande vecka" för att ta sig tillbaka till lunchmatsedeln över den nuvarande veckan.

```

$(document).on('click', '#nextweek', () => {

    if(clicked === 0) {

        $("#menuweek").animate({right: '2000px'}, () => {

            $("#nextweek").text("Nuvarande vecka");
            $("#week").html("Veckans Lunchmatsedel Vecka 7 <br> 10:00 - 13:00");
            $("#monday").text("Löksoppa");
            $("#tuesday").text("Torsk med potatis");
            $("#wednesday").text("Kebabtallrik");
            $("#thursday").text("Raggmunk med lingonsylt");
            $("#friday").text("Hawaii pizza");
            $("#lunchprice").text("Erbjudande! Pris 90 kr");
            $("#lunchprice").css({ 'color': 'red'});
            clicked = 1;

            $("#menuweek").animate({left: '0px'});

        });
    }
});

```

Figur 13: JQuery koden som byter lunchmatsedel

Figur 13 visar JQuery koden/funktionen som byter lunchmatsedel. Då knappen som har id:t 'nextweek' används så anropas funktionen. Funktionen börjar med att kontrollera ifall värdet clicked har värdet 0, vilket indikerar att man befinner sig på den nuvarande veckan och har klickat på knappen med texten "Nästa vecka". Där all text byts ut till nästa veckas lunchmatsedel, där även priset ändras till rött, eftersom det är extra pris, erbjudande den veckan, och clicked värdet sätt till 1. Lunchmatsedeln glidar ut från bild och tillbaka då matsedeln ändras, detta för att ge användaren en mer fysisk upplevelse, av känslan att byta matsedel. Befinner användaren sig på nästa vecka och vill komma tillbaka till nuvarande vecka, så fungerar den funktionen liknande. Clicked värdet bör vara 1 vid det fallet och hoppar då över if-satsen i figuren ovan, och kommer in i else statsen, som ser princip likadan ut, bara att all text blir matchade nuvarande vecka, pris texten sätts till svart, och clicked värdet sätt till 0.

### 3 Diskussion/Reflektion

Uppgiften som helhet har gått bra och varit rolig. Några problem jag stötte på under uppgiftens gång var att uppgift 4, där matsedeln ska slida ut och slida in igen då den byts, fungerar endast en gång. Då man klickar nästa gång på knappen så byts matsedeln men utan slide ut och in. Försökte fixa det, men fick inte till det inom tiden. Ett annat problem är att då man väljer datum vid boka bord sidan, så kan man välja tider på nuvarande dag som har passerat. Något som kan förbättras är som sagt att matsedeln slidar in och ut varje gång, och att inte kunna boka passerade tider på nuvarande dag på boka bord sidan. Angående uppgift 2, att lägga till en användarinteraktion som ska påverka utseendet av webbsidan, så valde jag att göra ett automatiskt bildspel istället för ett klickbart bildspel. Detta gjorde pga det passade bättre med automatisk bildspel som då ska ”visa upp restaurangen” och tillföra ett välkommet bemötande till användaren. Samt tyckte jag att det skulle bli konstigt och onödigt ifall användaren skulle klicka manuellt för att iterera igenom bilderna. Så hoppas på att det är okej för uppgift 2. Annars är jag nöjd med slutresultatet och tyckte det var en kul och lärorik uppgift.

### 4 Länk till hemsidorna

Originalet: [Gamla Kajutan hemsidan](#)

Nya: [Nya Kajutan hemsidan](#)