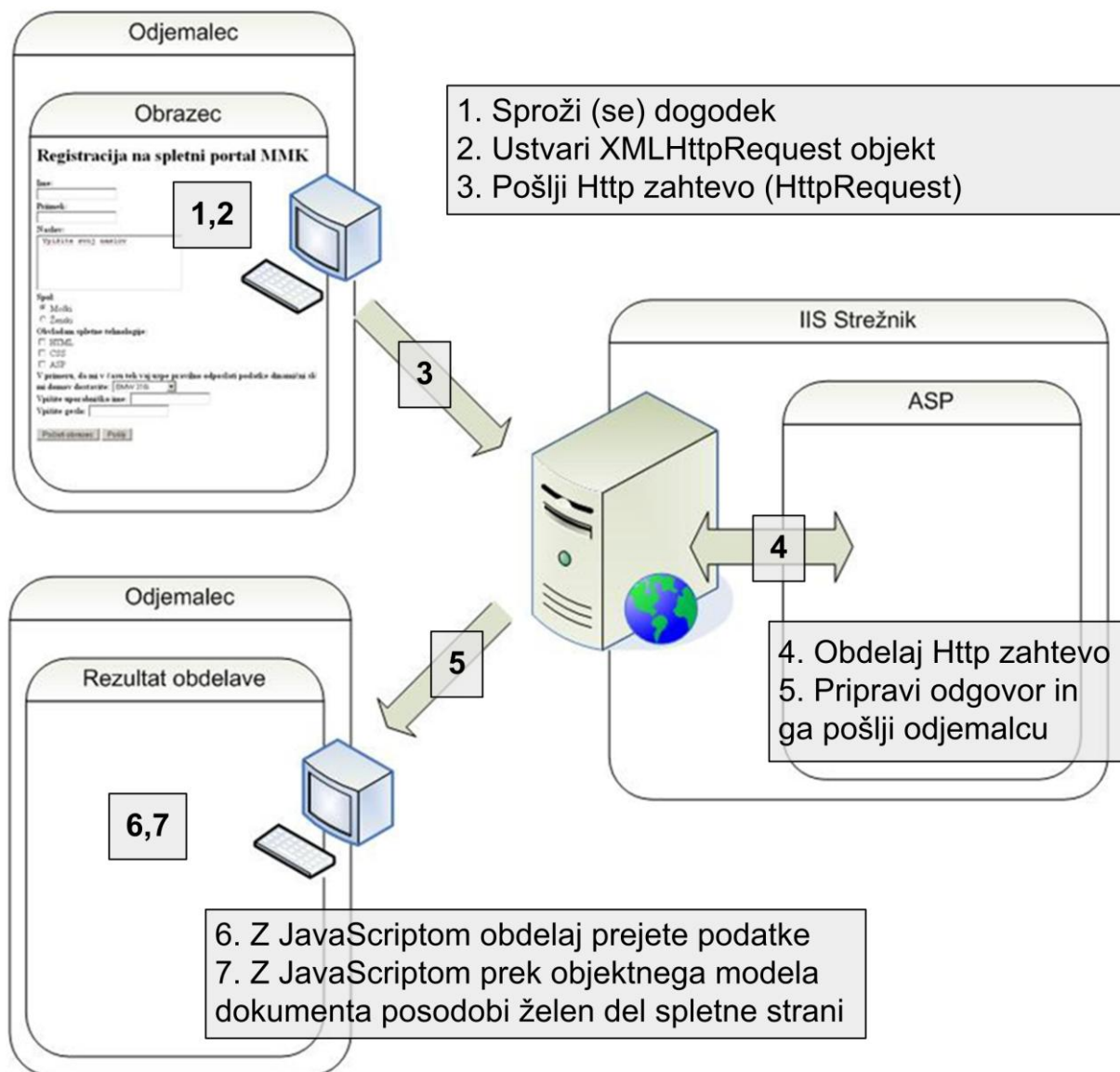


## Vaja 10: Asinhrona komunikacija med odjemalcem in strežnikom

Ajax (*Asynchronous JavaScript and XML*) je skupek tehnologij, ki se uporablja za izdelavo hitrih dinamičnih spletnih strani. Bistvo tehnologije Ajax je asinhrona komunikacija med odjemalcem in strežnikom, ki ne zahteva ponovnega nalaganja celotne spletne strani ampak le osveževanje posameznih delov strani.

Ajax uporablja kombinacijo različnih spletnih tehnologij:

- objekt **XMLHttpRequest** za asinhrono izmenjavo podatkov s spletnim strežnikom
- JavaScript in DOM za izpisovanje podatkov na poljubnem mestu v spletni strani
- CSS za oblikovanje prikaza podatkov
- XML kot običajni transportni format podatkov



Slika 1: Princip asinhrone komunikacije med odjemalcem in strežnikom

## Objekt XMLHttpRequest

XMLHttpRequest je najpomembnejši objekt pri delu z Ajaxom, saj skrbi za izmenjavo podatkov med odjemalcem in strežnikom, medtem ko uporabnik nemoteno naprej uporablja spletno stran. Objekt XMLHttpRequest podpirajo vsi moderni brskalniki. Spodnja vrstica prikazuje ustvarjanje novega XMLHttpRequest objekta.

```
xmlhttp=new XMLHttpRequest();
```

## Pošiljanje zahteve

Ko ustvarimo XMLHttpRequest objekt, ga lahko uporabimo za pošiljanje zahteve strežniku. Najpomembnejši metodi objekta sta metodi *open* in *send*.

```
xmlhttp.open("GET","ajax.php",true);  
xmlhttp.send();
```

Z metodo *open* določimo lastnosti zahteve, in sicer

- metodo zahteve,
- URL lokacijo, kamor se pošlje zahteva, in
- »zastavico«, ki pove, ali gre za asinhrono (*true*) ali sinhrono zahtevo (*false*).

Z metodo *send* pošljemo zahtevo na strežnik.



**Prednosti asinhrone komunikacije lahko izkoristite le, če nastavite tretji parameter funkcije *open* na »true«.**

## Posodabljanje strani

Za posodobitev strani z asinhrono pridobljenimi podatki s strežnika so najpogostejše uporabljane lastnosti objekta XMLHttpRequest sledeče:

- **onreadystatechange** – vsebuje funkcijo, ki je poklicana samodejno, ko pride do spremembe lastnosti **readyState**
- **readyState** – vsebuje stanje zahteve (4 pomeni, da je že prišel odgovor na zahtevo)
- **status** – vsebuje stanje iz Http odgovora (npr. 200: »OK«, 404: »Stran ni bila najdena«)
- **responseText** – vsebuje podatke iz odgovora v obliki znakovnega niza

Spodnja koda prikazuje uporabo omenjenih lastnosti za posodobitev vsebine elementa, ki ima id »mojDiv«.

### 3 MMK - Spletne tehnologije – Vaja 10: Asinhrona komunikacija med odjemalcem in strežnikom

```
xmlhttp=new XMLHttpRequest();

xmlhttp.onreadystatechange=function()
{
    if(xmlhttp.readyState==4 && xmlhttp.status==200)
    {

document.getElementById("mojDiv").innerHTML=xmlhttp.responseText;
    }
}

xmlhttp.open("GET","ajax.php",true);
xmlhttp.send();
```



1. Izdelajte dinamično strežniško skripto *ajax.php*, ki bo pripravila preprost odgovor strežnika odjemalcu.
2. Dopolnite poljubno spletno stran z asinhrono komunikacijo med odjemalcem in strežnikom.
  - a. Zgornjo »ajax« kodo vključite v JavaScript funkcijo.
  - b. Izberite si element na spletni strani, vsebino katerega bi radi posodobili na podlagi asinhrono komunikacije s strežnikom. Dodelite mu *id*, s pomočjo katerega boste kasneje do elementa dostopali prek modela DOM.
  - c. Izberite si dogodek, ki bo sprožil asinhrono komunikacijo.
  - d. Ob pojavu dogodka, naj se pokliče pripravljena JavaScript funkcija (uporabite znanje iz Vaj 8 in 9).

### **Literatura**

1. Spletni tečaj Ajaxa:
  - <http://www.w3schools.com/Ajax/Default.Asp>
2. Prosojnice s predavanj