**IP Adresse**IP står for “Internet Protocol”. Det er en logisk numerisk unik adresse som bliver tildelt til alle enheder som er forbundet til et netværk der bruger Internet Protokollen til at kommunikere.

**MAC Adresse**MAC står for “Media Access Control”. MAC adressen kaldes også for ethernet-adresse. Fysisk udstyr som f.eks routere, switches, hubs, netkort og andre computer komponenter som kommunikere med hinanden via internettet har en MAC adresse.

**Localhost**Localhost referere til den computer programmet køre på, f.eks med localhost i en web browser. Localhost bruges blandt andet til at teste ens Web Applikation og dets funktionalitet.  
 **HTTP**HTTP står for “HyperText Transfer Protocol” og er en protokol brugt af World Wide Web. Protokollen bliver brugt til at forbinde til web servere og sende HTML sider tilbage til brugerens browser.

**OSI Model**OSI står for “Open Systems Interconnection”. OSI er en abstrakt lagdelt beskrivelse af kommunikation og netværksprotokoller der findes på de syv lag.

**Netværk**Et netværk kan beskrives som et system af sammenkoblet enheder der kommunikere ved at bruge overensstemmende protokoller.

**Router**En router er en enhed på et computernetværk som forbinder et antal logiske eller fysiske netværk ved at videresende packets fra et netværk til deres destination på et andet netværk

**Sliding window protocol**

det er mængden af data som klient og server kan sende til hinanden inden der skal være en bekræftelse.

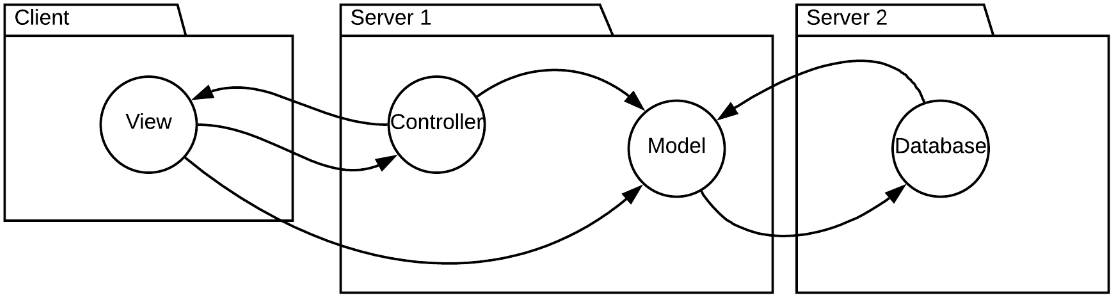
**TCP**

TCP er en connection-oriented protocol som sørger for at din client opretter forbindelse til din server inden den overfører data, det gør den med et 3-way handsshake: 1. sørger for at der er oprettet forbindelse til Server. 2. bekræftelse på at der er oprettet en forbindelse. 3. bekræftelse på at der er oprettet en sikker forbindelse.

***Listen over de emner/begreber fra faget Teknologi I anvender i jeres løsning skal opdateres.***

*Giv en kort beskrivelse af hvorledes jeres MVC løsning kunne implementeres på flere (adskilte) servere. I skal altså ikke (nødvendigvis) beskrive den aktuelle implementering, men fremstille en tænkt løsning, hvor forskellige elementer af jeres løsning er placeret (implementeret) på flere individuelle servere der er fysisk helt adskilte. Overvej hvilke fordele og hvilke ulemper det kan have ud fra et rent netværksperspektiv.*

Der er mange forskellige måde at implementere MVC mellem en klient og en server. Vi ville implementere vores løsning således at vores View var client-side. Det ville sige at brugeren kun kan interagere med vores brugergrænseflade/GUI. Det gør også at vores Controller ikke kan blive “manipuleret” med. Vores Controller, Model og database ville være server-side.



**Sikkerhed:**I det at vi har med brugernes personlige oplysninger at gøre (de skal have en profil på vores hjemmeside), så er vi nødt til at sørge for der en sikkerhed for dem, så de ikke får misbrugt deres følsomme data, nogle af de måder vi kan gøre det på er ved at bruge “Hashing” og tilføje “Salt” til deres passwords, så det gør det nært umuligt for hackere at få fingrene i vores brugeres oplysninger.

**Hashing:**Når brugeren laver et password så kan vi “Hashe” det, når det bliver gjort så krypteres deres password, så det bliver lavet til en string med matematiske funktioner.

**Salt:**Salt er en tilføjelse til Hashing det er en ekstra string som bliver lagt ovenpå vores hashede password. på den måde fungere det som en ekstra tabel som det er hackere også skal igennem for at kunne lave deres “dictonary” (liste med kendte tegn i hashen)