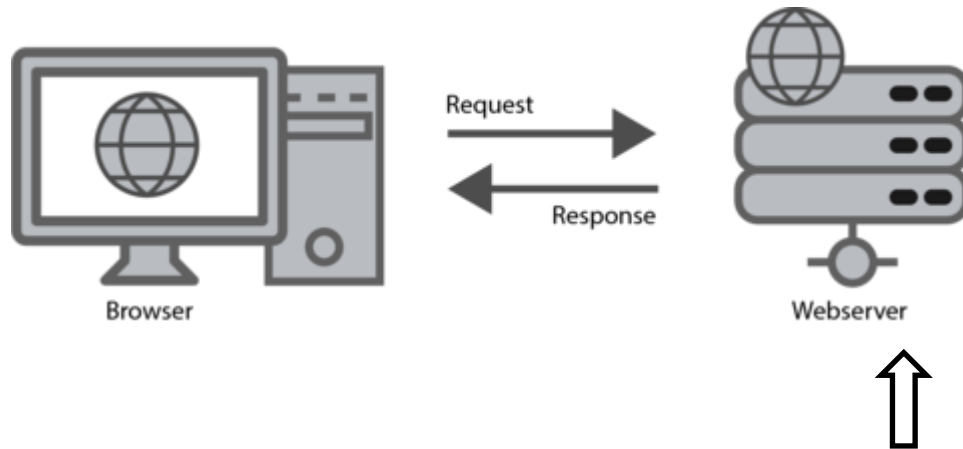


Trelagsarkitektur

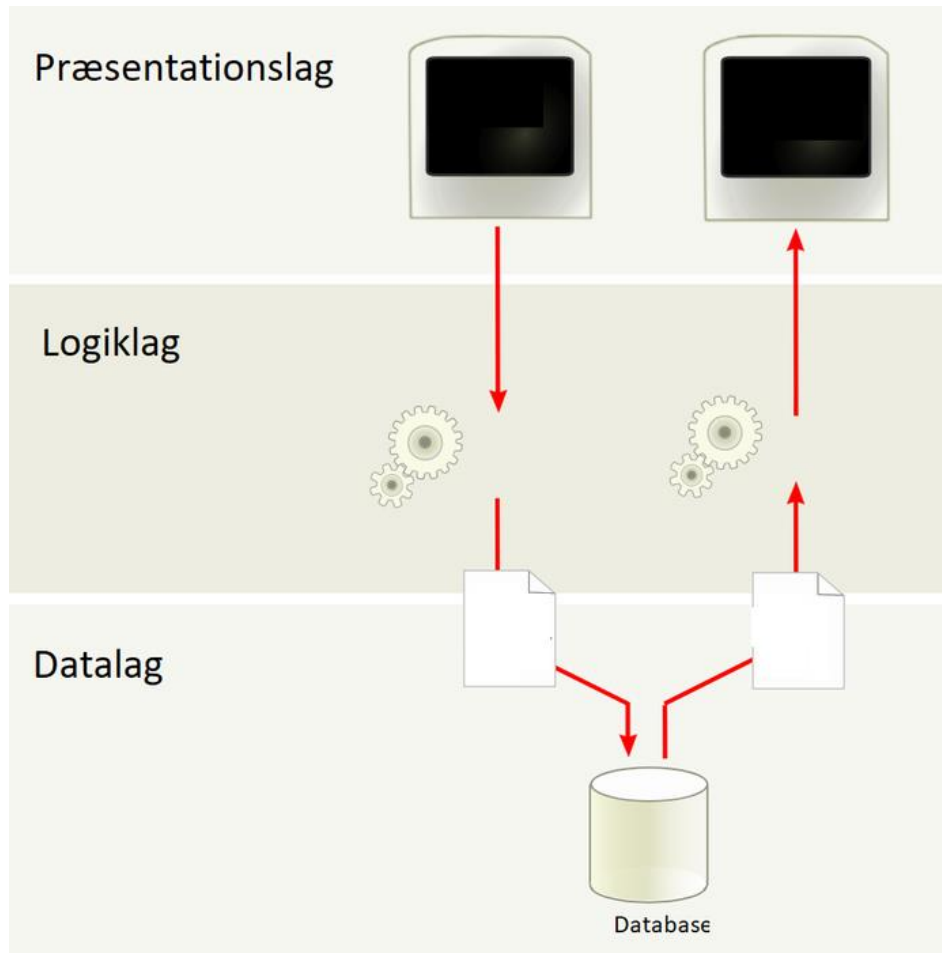
Klient-server arkitektur



Hvis man har mange forespørgsler på sine data, laver man typisk en videreopdeling af serveren i et:

- 1) logiklag
- 2) datalag

Trelagsarkitektur

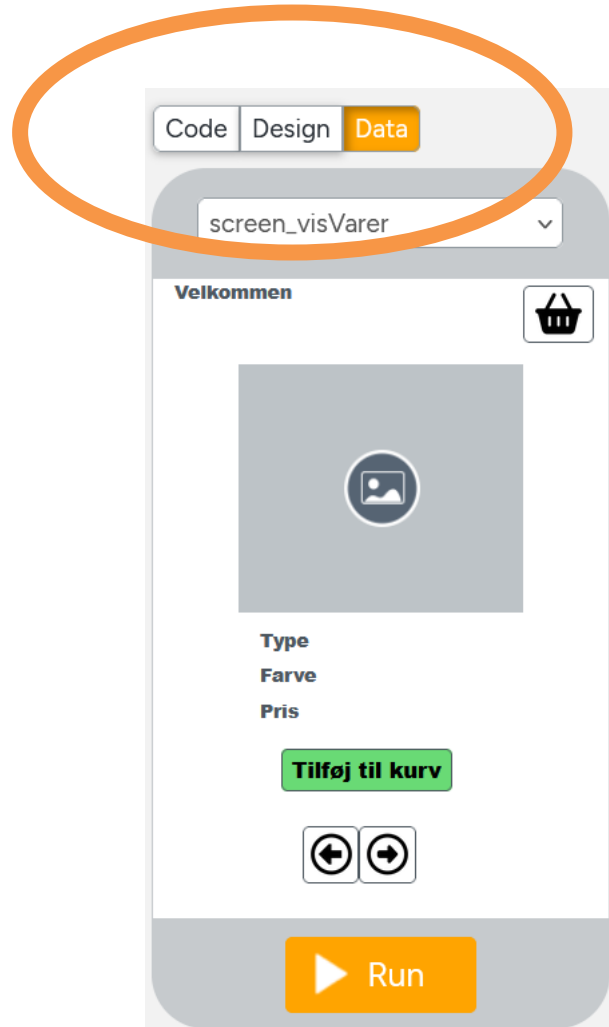


← Tættest på brugeren.
Modtager input fra brugeren.
Præsenterer output til brugeren.

← Foretager beregninger.
Udveksler data mellem præsentationslaget og datalaget

← Opbevarer data.
Søger i data.
Sender information tilbage til logiklaget.

I kender det allerede fra AppLab



Hvor kommunikerer lagene med hinanden?

```
5 // Indlæs varedata fra tabellen
6 readRecords("vare", {}, function(records) {
7     for (var i =0; i < records.length; i++) {
8         addItem(varer, records[i]);
9     }
10     visVarer();
11 });
12
13 // Funktion til at vise vare
14 function visVarer() {
15     setImageURL("image2", varer[nr].billede);
16     setText("label4", "Type: " + varer[nr].type);
17     setText("label5", "Farve: " + varer[nr].farve);
18     setText("label6", "Pris: " + varer[nr].pris + " kroner");
19 }
```

Hvor kommunikerer lagene med hinanden?

```
65 // Køb varer
66 ◀ onEvent("button6", "click", function( ) {
67     var kundeid;
68     var kunde = {};
69     var ordreid;
70     kunde.navn = getText("text_input1");
71     kunde.email = getText("text_input2");
72 ◀ createRecord("kunde", kunde, function(record) {
73     kundeid = record.id;
74     var date = new Date();
75     var day = date.getDate();
```

Øvelse

- Gå ind på din egen app – hvor kommunikerer lagene med hinanden?
- Tilføj på de steder en kommentarer til koden

```
// Dette er en kommentarer
```

```
/* Det er det her også */
```