Prosjekt bibliotek

**Tanker før jeg har startet med arbeidet:**

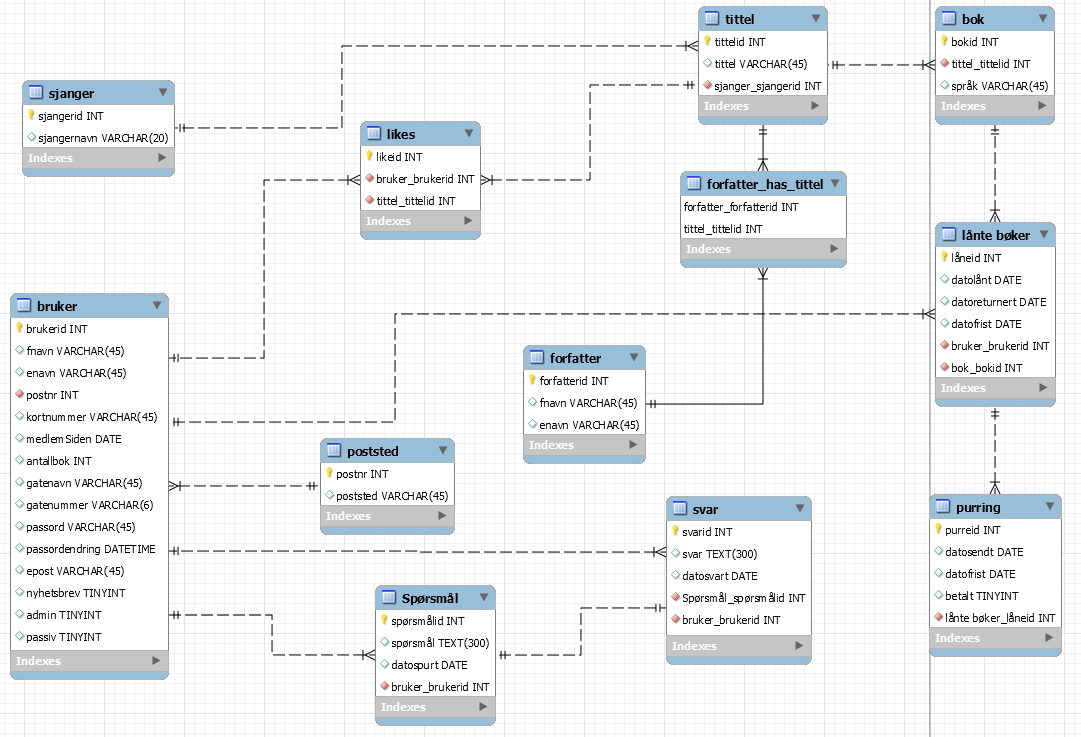
Jeg har lyst til å lage et nettsted for online-bibliotek. Ideen bak er å kunne «låne» ebøker, men også å bestille bøker som man kan få levert på døren. På hovedsiden ser jeg for meg at man kan søke etter bøker og forfattere og filtrere etter sjanger, språk og vanskelighetsgrad. Det må også være en side med kontaktinformasjon. Jeg vil også at man kan opprette «lånekort» på nett, samt en brukerside tilknyttet lånekortet der man kan se hvilke bøker du har lånt og når du må levere dem tilbake. Jeg vil også ha en side der man kan legge inn spørsmål. Administratorene kan deretter legge inn svar på administratorsiden. Man må selvfølgelig også kunne legge inn nye bøker og merke boksider som inneholder feilinformasjon. Våre anbefalinger, hent ut brukerdata og sorter etter sjanger. Deretter de fire bøkene med flest likes innen de sjangrene.

**Problemer jeg ser før jeg har startet:**

Dette blir et relativt stort og ambisiøst prosjekt, men jeg har jo ikke så mye bedre å gjøre for tiden så tid bør ikke være det største problemet. Jeg ser derimot for meg at det kan bli vanskelig å legge inn mange bøker, men jeg går ut ifra at jeg finner en løsning på dette. Om det blir at jeg får venner og familie til å manuelt legge inn bøker gjennom admin-siden eller om jeg finner en database med bøker som er open-source som jeg kan kopiere vet jeg ikke enda. Evt må jeg bare akseptere at det ikke blir så veldig mange bøker. Jeg skal i alle fall minst ha 50. Design (css og layout) blir nok også et lite problem for meg, men det så ikke ut til at det telte så veldig mye på dette prosjektet.

**Backlog:**

|  |  |  |
| --- | --- | --- |
| **DATO** | **Gjøremål** | **Logg** |
| 16.04.2020 | Planlegge hva jeg skal gjøre, lage en backlog, se på eventuelle problemer som kan dukke opp. | Fikk tenkt ferdig hva jeg har lyst til å gjøre med nettsiden min. Tenkt over problemer som kan dukke opp så jeg har det i bakhodet. Dette brukte jeg 2 timer på. |
| 17.04.2020 | Lage en grundig skisse for hvordan hovedsiden skal se ut i paint. Hvis jeg får tid begynne på HTML-koden. | Gjorde ferdig skisse for navigasjonsbar og footer, samt innholdet på hovedsiden. Startet også å strukturere div’er i html/css. Brukte 2 timer. |
| 20.04.2020 | Jobbe ekstra | Jobbet 8 timer med IT. Fikk laget ferdig hovedsiden sin html og css. Begynte også på side nummer to sånn smått. |
| 21.04.2020 | Lage skisser for resten av sidene. Lage HTML-struktur for de. | Laget ferdig HTML-struktur for de resterende sidene. |
| 24.04.2020 | Gjøre datamodellering | Fikk gjort ferdig datamodellering og tenkt litt mer gjennom det praktiske ved siden. |
| 25.04.2020 | Jobbe med innhold til sidene |  |
| 28.04.2020 | Jobbe med innhold til sidene |  |
| ~~01.04.2020~~ | Jobbe med innhold til sidene |  |
| 05.05.2020 | Jobbe med innhold til sidene |  |
| 08.05.2020 | Jobbe med innhold til sidene |  |
| 12.05.2020 | Jobbe med innhold til sidene |  |
| 15.05.2020 | Jobbe med innhold til sidene |  |

Datamodellering: 

**Forklaring på datamodelleringen:**

*-Tabeller og kolonner*

Det er mye informasjon som lagres om brukeren, det meste sier seg selv, de som har datasett TINYINT er rett og slett boolske variabler. Boolske variabler er lagret som true eller false, det vil si at på kolonnen «admin», så er den enten lagret som sann, eller usann. Dette er gjort slik at vi kan skile mellom administratorer og brukere på en enkel måte. Brukerne kan stille spørsmål hvis den boolske variabelen «admin» er satt til usann, og «brukerid» blir koblet opp mot spørsmålet. På samme måte kan administratorer skrive svar som igjen kobles opp til et spørsmål. Et spørsmål kan kun ha ett svar. Når en bruker velger å låne en bok, blir det lagret i tabellen «lånte bøker». Da opprettes en unik låneid som definerer akkurat det lånet. Tabellen «låneid» lagrer alle tidligere og aktive lån. Den er igjen koblet opp mot tabellen «bok», som er koblet opp mot tabellen «tittel», for å vite hvilken bok som har blitt lånt. VI vil også vite hvem som har skrevet boken, men det kan være flere forfattere som har skrevet på samme bok, og en forfatter kan ha skrevet flere bøker, derfor er det en mange-til-mange kobling mellom tittel og forfatter. For at biblioteket skal kunne overleve, må det også være en purrefunksjon, den funker slik at hvis det ikke opprettes en verdi for «datoreturnert» innen fristen som er lagret i «datofrist», så opprettes en purring. Denne purringen har en egen id som jeg har valgt å kalle «purreid». Kolonnen «betalt» er en boolsk variabel som sier om purringen har blitt betalt eller ikke. Dersom purringen ikke betales innen fristen som har blitt laget, blir det opprettet en ny purring, med ny purreid og ny dato. Dersom dette hadde vært en faktisk purretjeneste ville jeg ha lagt til en egen kobling slik at purrekostnaden var forskjellig for forskjellige titler. Nå har jeg det bare sån nat det er en grunnkostnad, og for hver ny gang en purring blir sendt ut øker kostnaden for purringen.

-*Koblinger*

En sjanger kan være knyttet til flere titler, men en tittel er bare knyttet til en sjanger. Dette er selvfølgelig diskutabelt, men jeg velger å holde det «enkelt» og kun ha en sjanger. Det er videre flere eksemplarer av samme tittel, men et eksempler kan bare ha en tittel, derfor er det en en-til-mange kobling mellom tittel og bok. En bok kan ha blitt lånt ut flere ganger, men en låneid kan bare være knyttet opp til en bok. Igjen så kan det være flere purringer på samme låneid, men en purreid tilsvarer bare en låneid. Mellom tittel og forfatter er det en mange til mange kobling fordi flere titler kan være skrevet av en forfatter, og flere forfattere kan ha skrevet en bok. En brukerid kan svare til flere spørsmål, ett spørsmål kan bare ha ett svar. Igjen så kan et spørsmål bare være besvart av en administrator, og en administrator kan svare på flere spørsmål. Derfor en til mange mellom bruker og spørsmål, en til en mellom spørsmål og svar, og en til mange mellom bruker og svar. Til slutt så kan en bruker bare være knyttet til et postnummer, og et postnummer kan være knyttet til flere brukere. Poststed er i samme tabell som postnummer fordi dersom man har postnummeret så kan man finne poststedet i registrene til posten.