

Duckvillage DBMS

DWP Database 2020

Semester Project

Duckvillage

Gruppemedlemmer:

Philip Thang

Vejleder

Søren Spangsberg Jørgensen

Antal sider:

4

Hjemmeside

<https://www.erss.dk/shopping/>
<https://github.com/sangboi2/DWP-Database2020>

Dato: 20/05/2020

Duckvillage DBMS

Indholdsfortegnelse

Indledning	1
Databasebeskrivelse:	1
Data attributter:	2
Database design proces:	2
Normalisering	3
Konklusion	4
Referencesliste:	4

Indledning

I dag bliver online-shoppings popularitet stigende og accepteret og brugt som et sted, hvor man laver ens alle indkøbsaktivitet på nettet. Man må sige, at processen med shopping på nettet er ved at blive almindelig. Dog for at kunne have en hjemmeside, der leverer funktionalitet til at udføre kommercielle transaktioner over nettet med høj sikkerhed og giver mest brugeroplevelse, skal man have en fungerende database.

Formålet med denne database er at designe en fundamental database system for Duckvillage online-shopping, der giver et overblik hvordan et databasesystem ser ud og vedligeholder.

Databasebeskrivelse:

Denne database består af 6 entiteter.

- En bruger:
 - en bruger består af både en almindelig bruger, en gæst og administrator. Som administrator bliver administratoroplysninger tilføjet til databasen med unikt id-baseret på deres roller.
- Shopping:
 - der vises og gemmes alle produktinformation og kategori i denne tabel.
- Bestilling:
 - alle kundens bestilte og købte produkter, status, leveringsoplysninger bliver gemt i denne tabel.
- Produkt:
 - Komplet produktinformation registreres i denne tabel.
- Kategori:
 - nogle af produktinformation, der tilhører en bestemt kategori registreres i denne tabel.
- Postnummer:
 - alle postnummer registreres i denne tabel.

Duckvillage DBMS

Data attributter:

User:

user_id, user_name, admin, guest, user, email, password, telephone, street, postcode, country

Shopping:

shopping_id, product, image, price, description, category

Order:

order_id, user_name, product_name, quantity, price, date_of_purchase, delivery_status

Product:

product_id, name, image, description, price, qty, stock

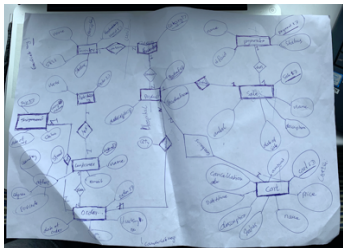
Category:

category_id, name

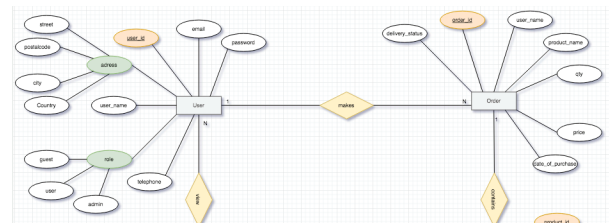
Database design proces:

Et konceptuelt skema

Der bliver udarbejdet et skelet i form af sketching (se figur nr. 1 og 2) for den nye hjemmeside database. Sketching, som er den første del af denne fase, er blot en enkel og hurtig papir skitse, der skal medvirke som grundstenen for det nye design.



Figur 1: Se billedet i fuld størrelse på bilag 1



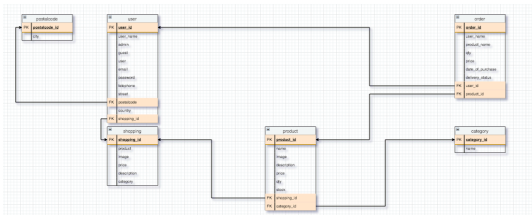
Figur 2: Se billedet i fuld størrelse på vedlagte bilag 2

Derefter bliver også et skelet udarbejdet i draw.io for at det bliver et bedre skelet at se på.

Databasetabeldiagram

I denne fase er der blevet udarbejdet med et diagram model (ER diagram, Relationel Model) for at give et henblik på hvordan database kan sættes op. (se vedlagte bilag 2).

Duckvillage DBMS



Figur 3 Relationelt Data Model. Se billedet i fuld størrelse på vedlagte bilag 1

Normalisering

Efter der er lavet er ER-diagram model, begyndte man at lave normalisering, for at sikre og at have kvalitet i database. Ved normaliseringen bliver der elimineret unødvendige data gjort database nemmere at forstå.

Meningen med normalisering er at man sørger for at beskytte data, fjerne unødvendige data og inkonsistent afhængighed, der giver besværslighed at få adgang til data. Det er også at beskytte data. Man skal overholdes visse regler, når man laver normalisering. Processen benyttes tit når man laver tabeller og tabellernes relationer.

I dette projekt bliver database normaliseret og optimeret, og det er mere fleksibelt.

Databasen er lavet i *tredje normal form* i dette projekt. *Første- og anden normal form* skal man have fulgt for at kunne lave tredje normal form. Man kan forkorte Tredje normal form med *3NF*.

- *Først normal form:* gentagende kolonner i de enkelte tabeller bliver fjernet. Der må ikke have multiple værdier i de enkelte kolonner. Man undgår disse problemer ved at lave flere kolonner.
- *Anden normal form:* hvis *først normal form* er fulgt, kan man forsat at arbejde videre med *anden normal form*. Deraf skal alle kolonner afhænge af den primære nøgle. De kolonner, som ikke afhænger af den primære nøgle, skal afhænge af en anden nøgle (Foreign Key).
- *Tredje normal form:* I denne form skal man fjerne de kolonner, som ikke afhænger af den primære nøgle. For eksempel: postalcode er fjernet og puttet i egen tabel. Kolonnerne, der har en primærnøgle må ikke fjernes.

Duckvillage DBMS

I dette projekts database er der oprettet en tabel: *product*. Figuren forneden er tabellen for *product*.

product_id	name	image	description	price	qty	stock	shopping_id	category_id
------------	------	-------	-------------	-------	-----	-------	-------------	-------------

Konklusion

Der kan konkluderes, at der er plads for forbedring på min database for potentielle brugernes behov. Rapporten inkluderer datamodel og procesmodel og illustration, og *script* hvordan databasen er bygget med forskellige tabeller. Opbygningen af database har givet mig en præcis viden om, hvordan *mysql* bruges til at udvikle en database, og hvordan det opretter forbindelse til online-shopping med *mysql* database. Det har også givet mig viden om hvor vigtigt at have database på plads og hvordan websiderne ændres for at give brugeren en oplevelse, tilfredshed og sikkerhed.

Jeg må også indrømme at pandemisk har påvirket mit arbejde på projektet. Tilslut kan der konkluderes at der har været et meget udfordrende projekt at arbejde med —især denne pandemisk periode.

References liste:

- https://www.youtube.com/watch?v=CZTkgMoqVss&list=PLJetLDY7yKup6URKdaK7rfuBMPLkx_-ao
- <https://www.w3schools.com/sql/default.asp>
- <https://app.diagrams.net/>

Duckvillage DBMS

Bilag nr. 1

