TAL TECH

Infotehnoloogia teaduskond

Simo Savila 185579IADB

Veebipood

Juhendaja: Andres Käver

Tallinn 2020

Contents

Introduction 3
1. Scope of work 4
Extra 1. Entity Relationship Diagram 6

Introduction

The goal of this project is to create a homepage and web store to family business. This business main focus is cutting children's furniture out of plywood and selling it. Business produces four types of furniture – beds in different sizes, shelfs, chairs and tables. They also take custom orders, where they usually offer full service packet, designing according to client's wish, blueprints and cutting. Business also sells textbook named "Häälikuline teadlikkus". It talks how to teach your pre-school aged kids to read through games and puzzles.

1. Scope of work

The web environment must be easy to understand, usable and nicely designed, so it can be easily used by clients. The finished project also must be easily managed by site administrator. It means that whoever is managing the site can easily add, delete or edit the products.

Regular user or client can add products to cart, remove them from cart and also change the quantity of the products. If client adds item to the cart then he will see the expected delivery time and the price of whole cart.

Expected delivery time is calculated by the product availability in warehouse, material availability and expected post service time.

When client puts in an order then automatically the material or product quantity changes in the warehouse.

When product has a discount then it will be shown in the discounted products section and when client adds this product to the cart then the price will be calculated according to the discount price.

User can also comment and give feedback about the products.

Site administrator can add new product and add pictures about them. He can also mark products categories, add products and materials to warehouse and if needed also remove them from the warehouse. He can also add discount to products or make a campaign where selected products have the same discount. He can also highlight the products and point them out on the main page. He can also relate products, these products will be shown together.

2. Database Soft update/delete logic

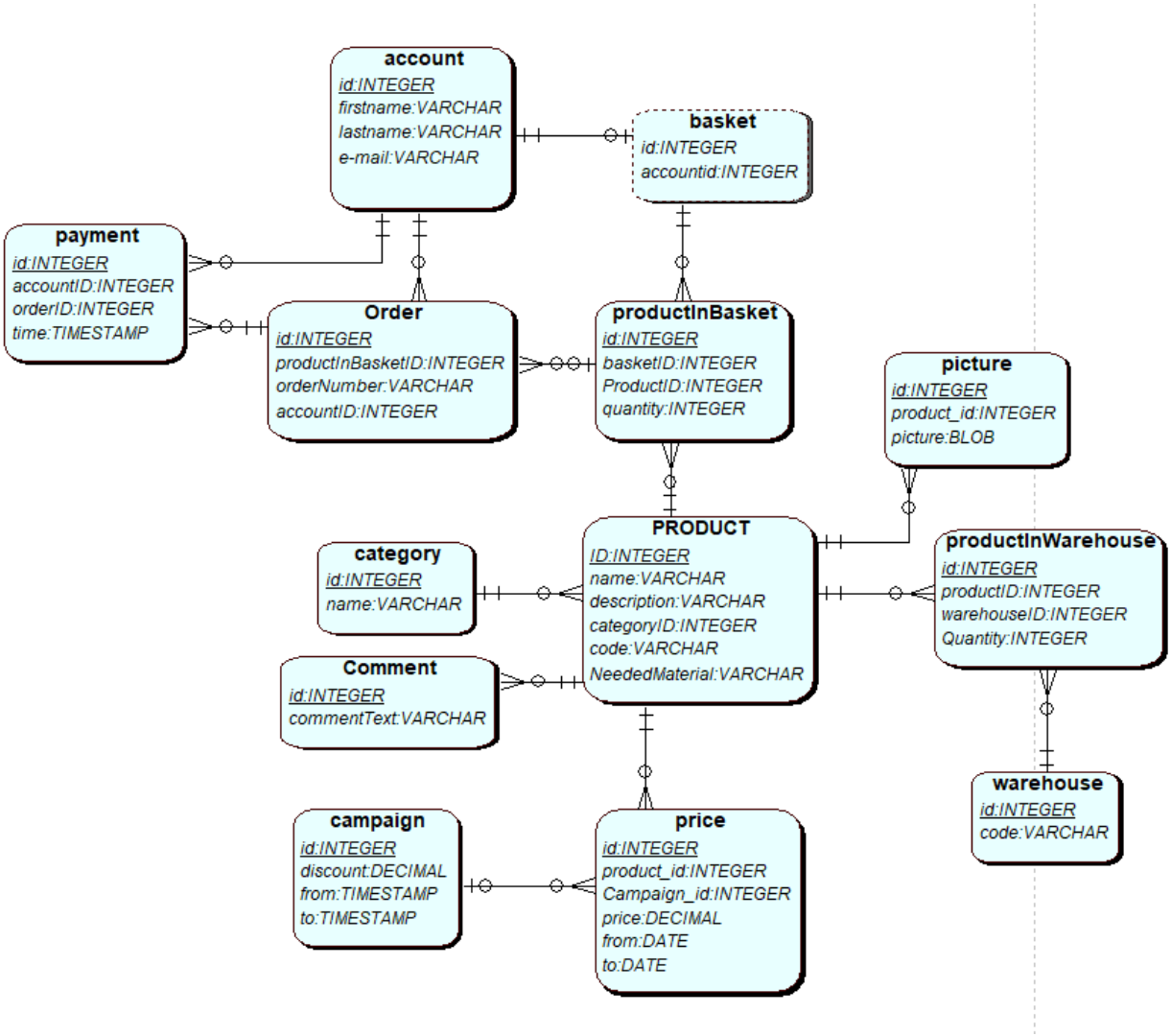
To make database easily auditable and reversible there must be some kind of logic how to see done changes and if needed reverse those changes or make a change in the future.

In order to keep the database auditable and reversible I can not use only ID as primary key in the tables because it would make database tables hard to read after someone modifies the data and it would be almost impossible to find previous data version using ID because the ID has to be unique.

The updated or changed data must reference to the data that was modified. For example if product code is changed it must also show what the old code was, because maybe someone have added it to their basket.

To make keeping track of changes easier I use composite Primary key. It consists of two parts - Id and date when it was deleted. By deleted I mean that it is no longer in use, for example if Product data is updated it is no longer in use and this item deleted at will be marked that timestamp when it was modified. If data is not deleted and is in use the deleted at value will be assigned automatically as 3000-01-01. No data will be deleted from the database and when it is updated it will be added as new row and old version of data will be there also. The Id never changes but Deleted At will change, so it is easier to see what changes has been made and you will still be able to track down the correct item using ID.

Extra 1. Entity Relationship Diagram



Extra 2. Updated Product code

	ProductId	ProductDelAt	AddedAt	Productname
1	2	3000-01-01 00:00:00.0000000	2020-02-02 00:00:00.0000000	Tool

	ProductId	ProductDelAt	AddedAt	ProductName
1	2	2020-03-08 00:00:00.0000000	2020-02-02 00:00:00.0000000	Tool

	(No column name)
1	Full Data after soft update

	ProductId	DelAt	CreatedAt	ProductName	ProductDescription	ProductCode
1	1	2020-03-07 00:00:00.0000000	2020-03-06 20:14:38.6200000	Voodi	Puit voodi vahatatud vineeri…	VV0001
2	2	2020-03-08 00:00:00.0000000	2020-03-06 20:17:37.2433333	Tool	Vineerist toolide komplekt 1…	VT0001
3	2	3000-01-01 00:00:00.0000000	2020-03-06 20:17:37.2433333	Tool	Vineerist toolide komplekt 1…	VTAAA2
4	3	3000-01-01 00:00:00.0000000	2020-03-06 20:17:37.2466667	Laud	Vineerist Laud värvitud ja v…	V10001

Extra 3. SQL code

```
CREATE TABLE Basket(
    BasketId INT not null ,
    DelAt DATETIME2 ,
    CreatedAt DATETIME2 NOT NULL,
    BasketName VARCHAR(50) NOT NULL,
    PRIMARY KEY (BasketId, DelAt)

);

CREATE TABLE Product(
    ProductId INT not null ,
    DelAt DATETIME2 ,
    CreatedAt DATETIME2 NOT NULL DEFAULT Current_timestamp,
    ProductName VARCHAR(50) NOT NULL,
    ProductDescription VARCHAR(128),
    ProductCode VARCHAR(64) NOT NULL,
    PRIMARY KEY (ProductId, DelAt)
);

CREATE TABLE ProductInBasket(
    ProductInBasketId int not NULL,
    BasketId int NOT NULL,
    BasketDelAt DATETIME2,
    ProductId int Not null,
    ProductDelAt DATETIME2,
    Quantity int not NULL,
    DelAt DATETIME2 NOT NULL ,
    AddedAt DATETIME2 NOT NULL,
    CONSTRAINT FK_Basket FOREIGN KEY (BasketId, BasketDelAt) REFERENCES Basket(BasketId, DelAt
) on update cascade,
    FOREIGN KEY (ProductId, ProductDelAt) REFERENCES Product(ProductId, DelAt) on update casca
de,
    PRIMARY KEY (ProductInBasketId, DelAt)

);
INSERT into Product(ProductId, DelAt, ProductName, ProductDescription, ProductCode, CreatedAt)
 VALUES ('1','3000-01-
01', 'Voodi', 'Puit voodi vahatatud vineerist', 'VV0001', Current_timestamp );
INSERT into Product(ProductId, DelAt, ProductName, ProductDescription, ProductCode, CreatedAt)
 VALUES ('2','3000-01-
01', 'Tool', 'Vineerist toolide komplekt lakitud', 'VT0001', Current_timestamp );
INSERT into Product(ProductId, DelAt, ProductName, ProductDescription, ProductCode, CreatedAt)
 VALUES ('3','3000-01-
01', 'Laud', 'Vineerist Laud värvitud ja vahatatud', 'Vl0001', Current_timestamp );

insert into Basket(BasketId, DelAt, CreatedAt, BasketName) VALUES ('1','3000-01-
01', Current_timestamp, 'MariOma' );
insert into Basket(BasketId, DelAt, CreatedAt, BasketName) VALUES ('2','3000-01-
01', Current_timestamp, 'KalleOma' );
insert into Basket(BasketId, DelAt, CreatedAt, BasketName) VALUES ('3','3000-01-
01', Current_timestamp, 'ErikaOstukorv' );
```

```
insert into ProductInBasket(ProductInBasketId, BasketId, BasketDelAt, ProductId, ProductDelAt,
 Quantity, DelAt, AddedAt) VALUES ('1', '2', '3000-01-01', '1', '3000-01-01', '2', '3000-01-
01', '2020-02-02');
insert into ProductInBasket(ProductInBasketId, BasketId, BasketDelAt, ProductId, ProductDelAt,
 Quantity, DelAt, AddedAt) VALUES ('2', '3', '3000-01-01', '3', '3000-01-01', '1', '3000-01-
01', '2020-02-02');
insert into ProductInBasket(ProductInBasketId, BasketId, BasketDelAt, ProductId, ProductDelAt,
 Quantity, DelAt, AddedAt) VALUES ('3', '1', '3000-01-01', '2', '3000-01-01', '1', '3000-01-
01', '2020-02-02');

-- Delete Voodi from Product.
DECLARE @Time1 DATETIME2
SELECT @Time1 = '2020-03-07'

UPDATE Product SET DelAt=@Time1 WHERE ProductId like '1'
SELECT 'Full Data after soft delete'
SELECT * FROM Product

DECLARE @TimeNow DATETIME2
SELECT @TimeNow = '2020-03-08'

SELECT 'Correct Data after soft delete'
SELECT * FROM Product WHERE  CreatedAt <= @TimeNow AND (DelAt > @TimeNow)

-- Select deleted Product
SELECT 'deleted Prodcut'
DECLARE @TimeNow2 DATETIME2
SELECT @TimeNow2 = '2020-03-08'
SELECT * FROM Product WHERE  CreatedAt <= @TimeNow2 AND (DelAt < @TimeNow2)

--Updated Product
SELECT ProductInBasket.ProductId, ProductInBasket.ProductDelAt, ProductInBasket.AddedAt, Produ
ctname FROM ProductInBasket INNER JOIN Product on Product.ProductId = ProductInBasket.ProductI
d AND Product.DelAt= ProductInBasket.ProductDelAt where Product.ProductId = '2';
Declare @ID INT
DECLARE @delTime DATETIME2
DECLARE @newCode VARCHAR(60)
SELECT @delTime = '2020-03-08'
SELECT @ID = ProductId FROM Product where ProductId = '2'
select @newCode = 'VTAAA2'
UPDATE Product set DelAt = @delTime WHERE ProductId = @ID
INSERT into Product (ProductId, DelAt, ProductName, ProductDescription, ProductCode, CreatedAt
) SELECT ProductId, '3000-01-
01', ProductName, ProductDescription, @newCode, CreatedAt from Product WHERE ProductId = @ID
SELECT ProductInBasket.ProductId, ProductInBasket.ProductDelAt, ProductInBasket.AddedAt, Produ
ctName FROM ProductInBasket INNER JOIN Product on Product.ProductId = ProductInBasket.ProductI
d AND Product.DelAt= ProductInBasket.ProductDelAt where Product.ProductId = '2';

DECLARE @Timenow5 DATETIME2
SELECT @TimeNow5 = '2020-03-05'
SELECT 'Full Data after soft update'
SELECT * FROM Product
```