Programming Project: Krusty Cookies

Johannes Jansson, F11
tfy11jja@student.lu.se

Victor Miller, F11
tfy11vmi@student.lu.se

Introduction

Requirements

Even though the requirements were a bit diffuse (as intended) we are fairly confident that we have met all of them.

Outline of the System

The system is constructed in a way much similar to the system in lab 4. The database manager MySQL is used to manage the database, located on puccini.cs.lth.se. The creation and population of the database is performed by sourcing the file **tables.sql**, the creation is described in listing 1.

The factory interface is a web page written in PHP. The php tool PDO was used to establish a connection to the database. The PHP class **database.inc.php** handles everything related to the SQL database, and the PHP class **pallet.inc.php** is used to transmit data from the database class to the webpage.

The interface consists of 8 views. The first view that the user ecounters is the login page. This is a simple view that only has as purporse to login to to the database. The view can be seen below

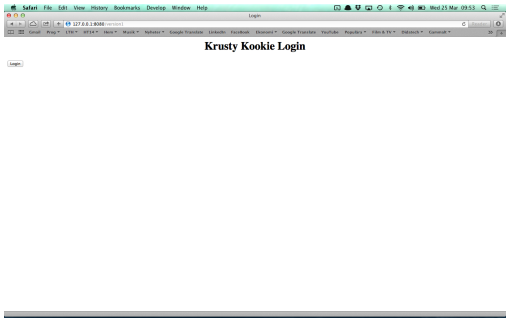
The user can then navigate on to one of the three options given, *Search, Block Pallet(s)* or *Simulate Production*. Each one of them has one input view where the user can provide the program with suitable input and the submit. The user is the provided with a result view to see what happend according to the given input.

Searching

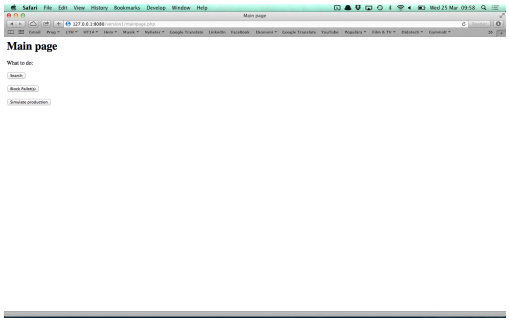
In the searching view the user can choose to search on different parameters. Each parameter can be left as *All* or set to a specific value. This gives the user the possibility to define thier search as they want. An example of the search view and the result is given below

Blocking

When the user wants to block pallets in a certain time intervall containing a certain product they can go to the block view. The user is the asked to choose which product they want to block and then a production date and time intervall to block all pallets with that product. The result of the request is the displayed in the next view. The user



(a) Login view for the interface



(b) View for the main page

Figure 1: First views

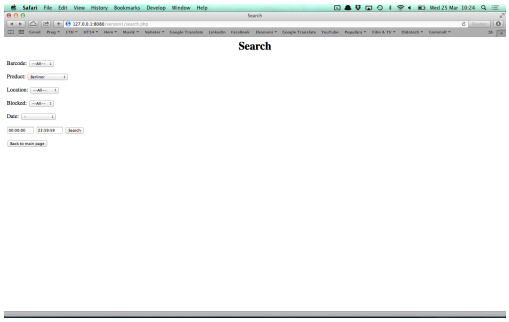
is provided with information on how many pallets that where blocked but also pallets in the intervall that are already blocked are displayed.

E/R Diagram

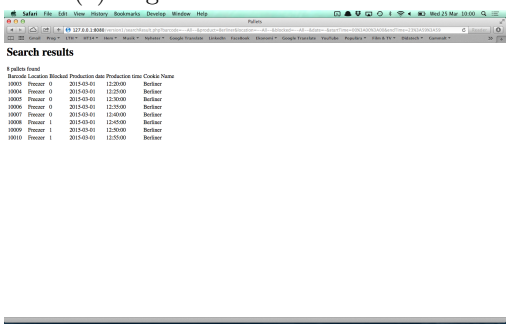
Relations

The following relations were the basis for creating the database:

```
CookieTypes(_name_);
(Primary key: name)

Ingredients(_name_, quantity, latestDeliveryDate,
    latestDeliverySize);
(Primary key: name)

Customers(_name_, address);
```



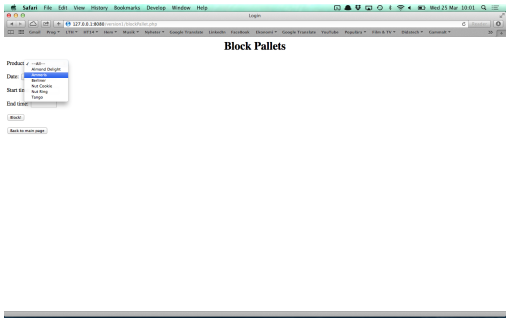
(a) Login view for the interface



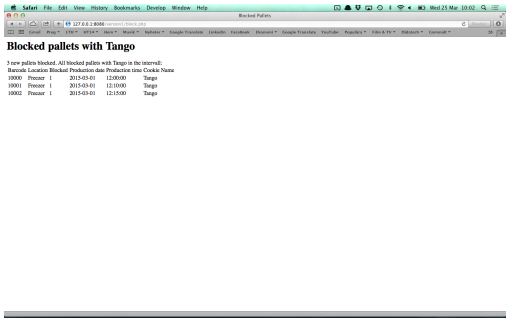
(b) View for the main page

Figure 2: First views

```
(Primary key: name)

Orders(_orderNbr_,deliveryDate,~customerName~);
(Primary key: orderNbr. Foreign key: customerName)

Locations(_location_);
(Primary key: location)

Pallets(_barcode_,~location~,blocked,producedDate,producedTime,
    ~cookieName~);
(Primary key: barcode. Foreign keys: location, cookieName)

Recipes(_~cookieName~,~ingredientName~_,quantity);
(Primary keys: cookieName, ingredientName.
    Foreign keys: cookieName, ingredientName)

CookieTypesInOrders(_~cookieName~,~orderNbr~_,quantity);
(Primary keys: cookieName, orderNbr.
```



(a) Login view for the interface



(b) View for the main page

Figure 3: First views

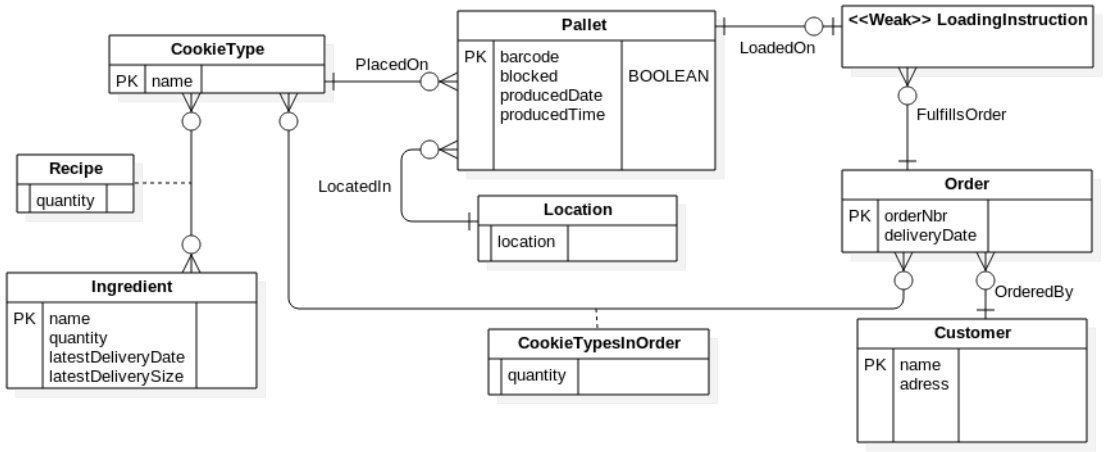


Figure 4: E/R–diagram for the system

Foreign keys: cookieName, orderNbr)

```
LoadingInstructions(~orderNbr~,_~barcode~_);
(Primary key: barcode. Foreign keys: orderNbr, barcode)
```

SQL Statements

The following SQL statements were used to create the database:

Listing 1: tables.sql

```
-- Delete the old tables
set foreign_key_checks = 0;
drop table if exists Pallets;
drop table if exists Locations;
drop table if exists CookieTypes;
drop table if exists Recipes;
drop table if exists Ingredients;
drop table if exists CookieTypesInOrders;
drop table if exists LoadingInstructions;
drop table if exists Orders;
drop table if exists Customers;
set foreign_key_checks = 1;

-- Create the new tables
create table CookieTypes (
  name            varchar(64),
  primary key (name)
);

create table Ingredients (
  name                varchar(64),
  quantity            integer check (quantity >= 0),
  latestDeliveryDate  date,
  latestDeliverySize  integer,
  primary key (name)
);

create table Customers (
  name    varchar(128),
  address varchar(256),
  primary key (name)
);

create table Orders (
```

```
  orderNbr      integer auto_increment,
  deliveryDate  date,
  customerName  varchar(128),
  primary key (orderNbr),
  foreign key (customerName) references Customers(name)
);

create table Locations (
  location     varchar(32),
  primary key (location)
);

create table Pallets (
  barcode       integer auto_increment,
  location      varchar(32) default 'Freezer',
  blocked       boolean default 0,
  producedDate  date,
  producedTime  time,
  cookieName    varchar(64),
  primary key (barcode),
  foreign key (cookieName) references CookieTypes(name),
  foreign key (location) references Locations(location)
);

create table Recipes (
  cookieName      varchar(64),
  ingredientName  varchar(64),
  quantity        integer check (quantity >= 0),
  primary key (cookieName, ingredientName),
  foreign key (cookieName) references CookieTypes(name),
  foreign key (ingredientName) references Ingredients(name)
);

create table CookieTypesInOrders (
  cookieName  varchar(64),
  orderNbr    integer,
  quantity    integer check (quantity >= 0),
  primary key (cookieName, orderNbr),
  foreign key (cookieName) references CookieTypes(name),
  foreign key (orderNbr) references Orders(orderNbr)
);

create table LoadingInstructions (
  orderNbr  integer,
```

```
  barcode    integer,
  primary key (barcode),
  foreign key (orderNbr) references Orders(orderNbr),
  foreign key (barcode) references Pallets(barcode)
);
```

User's manual

We consider the system self–explanatory enough not to require an user manual. The system has, in fact, been tested on a med–student with great success. The only things worth pointing out are that:

1. This thing
2. This thing
3. And this thing