**Introduction**

**Contacts, Big deals, Warehouse (ContactsBDW)**

**Introduction**

The application **Contacts, Big deals** will help organize your work with information of products, warehouse, suppliers and customers.

Information about suppliers and customers includes the following data: name (first, middle and last name), type of business, phone number (mobile and stationary), Tax identification number (TIN – ИНН), document details (ID, passport), address and etc.

Information about deals includes the following data: supplier and customer, type of business, name of product, price, quantity, sum, details of operation.

Information about warehouse includes the following data: brand, type of products, stock current status.

**Permission groups:**

**- Admin** - direct access to DBMS. Check and fix internal errors application and database. Communication with software developer: critical and system errors, improving, new functionalities and etc.

**- All Permissions users -** read, insert, update and delete all data.

**- Read And insert/update/delete Suppliers data -**  read all data and insert/update/delete suppliers data.

**- Read And insert/update/delete Customers data -**  read all data and insert/update/delete customers data.

**- Read And insert/update/delete Warehouse data -**  read all data and insert/update/delete stock data.

**- Read Only users** – read all data.

**Fitches**

**Minimum Requirements**

For the full functioning of the program, you only need: a browser, access to the Internet or a local network with a dedicated server.

**User interface**

User interface: simple and clear. This will allow you to optimize the use of working time. App do all operations (create, save, delete data) without reload page.

**Safety**

High data security is provided by the database server. Blowfish-based encryption algorithm, variation 2a provides high security for users' personal passwords.

**Logging**

Application don't delete records. App move records from active dataset to archive dataset. App also save last time and user for create, update or delete records.

**Requirements**

Requirements:

- Working on computers.

- Storing, creating, editing, deleting data for warehouse, suppliers, customers and information on big deals with them.

Suppliers, customers: name (first, middle and last name), phone number (mobile and stationary), Tax identification number (TIN – ИНН), document details (ID, passport), address.

Big deals: name, price, quantity, sum, details of deal.

- Warehouse:

- Types of products.

- Brands of products.

- Stock: products, quantity, price.

- Reports for analyze:

TOP-10 suppliers by biggest sum.

TOP-10 supplier deals by biggest sum.

TOP-10 customers by biggest sum.

TOP-10 customer deals by biggest sum.

Report: Stock Current Status.

**Developer tools**

- Java (1.8): basic programmer language. IDE: Eclipse IDE for Java Developers.

- PostgreSQL (11.16): basic database. Database administration tools: HeidiSQL, Dbeaver.

- Spring Boot (2.4.1): main framework.

- Hibernate (5.2.6.Final): framework for working with database.

- JSP, HTML, JavaScript (D3.js): front-end display and manipulate data.

- Maven (4.0.0): project management and building tool.

- Git/GitHub: distributed version control and store code.

**DBMS**

**Delete record – dataset 1 -> 0 (and add <#id> to end of fields with key constrait for bypass unique index). Application read/update/insert only dataset = 1.**

**Permission:**

**- Admin** - direct access to DBMS.

**- All Permissions users -** read, insert, update and delete all data.

**- Read And insert/update/delete Suppliers data -**  read all data and insert/update/delete suppliers data.

**- Read And insert/update/delete Customers data -**  read all data and insert/update/delete customers data.

**- Read Only users** – read all data.

**Algorithm**

**Window 1:**

- enter users.

- enter password.

- check login.

**Window 2:**

- Creating, editing, deleting data for suppliers.

**Window 3:**

- Creating, editing, deleting data for suppliers big deals.

**Window 4:**

- Creating, editing, deleting data for customers.

**Window 5:**

- Creating, editing, deleting data for customers big deals.

.......

**Window X:**

- Creating, editing, deleting data for X.

**Structure DB**

-- types of goods, services

CREATE TABLE types (

id smallserial,

type\_name varchar(30) PRIMARY KEY, -- types of goods or services

description text,

date\_create date, -- date creating record

dataset smallint, -- 1 - live, 0 – archive/deleted, -1/-2/-3 - reserv

last\_user varchar(20), -- last user

last\_changed timestamp, -- date and time last changed

CONSTRAINT types\_c UNIQUE (dataset, type\_name)

)

**\*\*\*\*\***

**-- 14 fields**

CREATE TABLE suppliers (

id serial PRIMARY KEY,

type\_name varchar(30) references types(type\_name), -- types of goods or services

first\_name varchar(20), --

middle\_name varchar(20), --

last\_name varchar(20), --

TIN varchar(30) ,-- INN

mobile\_p varchar(20), -- mobile phone number

station\_p varchar(20), -- stationary phone number

address varchar(50), -- address of supplier

date\_create date, -- date creating record

description text,

dataset smallint, -- 1 - live, 0 – archive/deleted, -1/-2/-3 - reserv

last\_user varchar(20), -- last user

last\_changed timestamp, -- date and time last changed

--CONSTRAINT type\_TIN\_s UNIQUE (dataset, type\_name, TIN),

--CONSTRAINT type\_mobile\_s UNIQUE (dataset, type\_name, mobile\_p),

CONSTRAINT type\_user\_s UNIQUE (dataset, type\_name, first\_name, middle\_name, last\_name)

);

CREATE TABLE suppliers\_deals (

id bigserial PRIMARY KEY ,

id\_supplier integer references suppliers(id), --

name\_good varchar(30), -- name of goods or services

price bigint, -- unit price

quantity integer,

sum bigint, -- usual price \* quantity, m.b. discount

details text,

date\_deal date,

dataset smallint, -- 1 - live, 0 – archive/deleted, -1/-2/-3 - reserv

last\_user varchar(20), -- last user

last\_changed timestamp, -- date and time last changed

CONSTRAINT suppliers\_deals\_s UNIQUE (dataset, id\_supplier, name\_good, price, quantity, sum, date\_deal)

);

\*\*\*\*\*\*\*\*

CREATE TABLE customers (

id serial PRIMARY KEY,

type\_name varchar(30) references types(type\_name),

first\_name varchar(20),

middle\_name varchar(20),

last\_name varchar(20),

TIN varchar(30), --INN

mobile\_p varchar(20),

station\_p varchar(20),

address varchar(50),

date\_create date,

description text,

dataset smallint, -- 1 - live, 0 – archive/deleted, -1/-2/-3 - reserv

last\_user varchar(20), -- last user

last\_changed timestamp, -- date and time last changed

--CONSTRAINT type\_TIN\_c UNIQUE (dataset, type\_name, TIN),

--CONSTRAINT type\_mobile\_c UNIQUE (dataset, type\_name, mobile\_p),

CONSTRAINT type\_user\_c UNIQUE (dataset, type\_name, first\_name, middle\_name, last\_name)

);

\*\*\*\*\*\*\*\*

CREATE TABLE customers\_deals (

id bigserial PRIMARY KEY,

id\_customer integer references customers(id),

name\_good varchar(30),

price bigint,

quantity integer,

sum bigint,

date\_deal date,

details text,

dataset smallint, -- 1 - live, 0 – archive/deleted, -1/-2/-3 - reserv

last\_user varchar(20), -- last user

last\_changed timestamp, -- date and time last changed

CONSTRAINT customers\_deals\_c UNIQUE(dataset, id\_customer, name\_good, price, quantity, sum, date\_deal)

);

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

CREATE TABLE brands (

id serial,

brand\_name varchar(30) PRIMARY KEY,

description text,

date\_create date,

dataset smallint, -- 1 - live, 0 – archive/deleted, -1/-2/-3 - reserv

last\_user varchar(20), -- last user

last\_changed timestamp, -- date and time last changed

CONSTRAINT brands\_c UNIQUE (dataset, brand\_name)

);

\*\*\*\*\*\*\*\*

CREATE TYPE measure\_type AS ENUM

('Единица','Комплект','Метр','М\_погон','Кг','Коробка','Прочее','');

CREATE TABLE products (

id serial,

type\_name varchar(30) references types(type\_name), -- new record = Прочее

brand\_name varchar(30) references brands(brand\_name), -- new record = NoName

product\_name varchar(30) PRIMARY KEY,

measure varchar(10), -- new record = Прочее measure\_type,

price bigint,

description text,

date\_create date,

dataset smallint, -- 1 - live, 0 – archive/deleted, -1/-2/-3 - reserv

last\_user varchar(20), -- last user

last\_changed timestamp, -- date and time last changed

CONSTRAINT products\_c UNIQUE (dataset, type\_name, brand\_name, product\_name, measure)

);

\*\*\*\*\*\*\*\*

CREATE TABLE stock (

id serial PRIMARY KEY,

product\_name varchar(30) references products(product\_name), -- new record = Goods

measure varchar(10), -- measure\_type,

price bigint, -- average price

quantity bigint, -- quantity on the stock

amount bigint, -- price \* quantity

description text, -- type\_name+brand\_name + other

date\_create date,

dataset smallint, -- 1 - live, 0 – archive+**id**/deleted, -1/-2/-3 - reserv

last\_user varchar(20), -- last user

last\_changed timestamp, -- date and time last changed

CONSTRAINT stock\_c UNIQUE (dataset, product\_name, measure)

);

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- **change only directly in DB**, because these changes depend changes in application code

CREATE TABLE permission\_level (

id smallserial,

permit\_level varchar(20) PRIMARY KEY, -- user permit level name - 'Admin','AllPermissions','ReadAndSuppliers','ReadAndCustomers','ReadAndStock','ReadOnly'

description text, -- Admin, All permissions, Read and insert, Read only.

date\_create date,

dataset smallint, -- 1 - live, 0 – archive/deleted, -1/-2/-3 - reserv

last\_user varchar(20), -- last user

last\_changed timestamp, -- date and time last changed

CONSTRAINT permission\_c UNIQUE (dataset, permit\_level)

);

\*\*\*\*\*\*\*

CREATE TABLE users (

id smallserial,

user\_id varchar(20) PRIMARY KEY, -- user login name

pswd text, -- password

permit\_level varchar(20) references permission\_level(permit\_level),

description text,-- user first, middle, last name

date\_create date,

dataset smallint, -- 1 - live, 0 – archive/deleted, -1/-2/-3 - reserv

last\_user varchar(20), -- last user

last\_changed timestamp --, date and time last changed

-- CONSTRAINT users\_c UNIQUE (dataset, user\_id)

);

DROP EXTENSION pgcrypto;

CREATE EXTENSION pgcrypto;

CREATE FUNCTION crypt(text, text)

RETURNS text

AS 'MODULE\_PATHNAME', 'pg\_crypt'

LANGUAGE C IMMUTABLE STRICT PARALLEL SAFE;

CREATE FUNCTION gen\_salt(text)

RETURNS text

AS 'MODULE\_PATHNAME', 'pg\_gen\_salt'

LANGUAGE C VOLATILE STRICT PARALLEL SAFE;

**INSERT** **INTO** users (user\_id, pswd,permit\_level,dataset)

**select** 'kanvladi',crypt('\*\*\*\*\*\*\*\*\*', gen\_salt('bf')),'Admin',1

SELECT count(\*) AS match

FROM users u

WHERE u.user\_id = 'johndoe2'

AND u.pswd = crypt('johnspassword', u.pswd);

**SELECT** \*

**FROM** users u

**WHERE** u.user\_id = 'johndoe2'

**AND** u.pswd = crypt('johnspassword', u.pswd);

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Types of goods, services**

Новые телефоны

БУ телефоны

Аксессуары моб.

Аксессуары комп.

Аксессуары проч. - ТВ, батарейки, калькуляторы,

Текстиль - Постель, Полотенца, Халаты

Сумки, Рюкзаки

Сантехник

Сайты

Прочее

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

String type\_name, String first\_name, String middle\_name, String last\_name, String TIN, String mobile\_p, String station\_p, String address, Date date\_create, String description

**Reports**

**CREATE OR REPLACE VIEW customers\_by\_sum AS**

**select** v.id, v.type\_name, v.first\_name, v.middle\_name, v.last\_name, v.dataset, v.customers\_sum

**from**

(**select** t.id, t.type\_name, t.first\_name, t.middle\_name, t.last\_name, t.dataset,

**sum**(**c**.**sum**) **as** customers\_sum

**from** customers t, customers\_deals **c**

**where** t.dataset=1 **and** **c**.id\_customer=t.id **and** **c**.dataset=t.dataset

**group** **by** t.id, t.type\_name, t.first\_name, t.middle\_name, t.last\_name, t.dataset

) v

**order** **by** v.customers\_sum desc

\*\*\*\*\*\*

**CREATE** **OR** **REPLACE** **VIEW** suppliers\_by\_sum **AS**

**select** v.id, v.type\_name, v.first\_name, v.middle\_name, v.last\_name, v.dataset, v.suppliers\_sum

**from**

(**select** t.id, t.type\_name, t.first\_name, t.middle\_name, t.last\_name, t.dataset, **sum**(**c**.**sum**) **AS** suppliers\_sum

**from** suppliers t, suppliers\_deals **c**

**where** t.dataset=1 **and** **c**.id\_supplier=t.id **and** **c**.dataset=t.dataset

**GROUP** **BY** t.id, t.type\_name, t.first\_name, t.middle\_name, t.last\_name, t.dataset)

v

**order** **by** v.suppliers\_sum desc

\*\*\*\*\*\*

**CREATE** **OR** **REPLACE** **VIEW** customers\_deals\_by\_sum **AS**

**select** t.id, t.type\_name, t.first\_name, t.middle\_name, t.last\_name, t.dataset,

**c**.**sum**

**from** customers t, customers\_deals **c**

**where** t.dataset=1 **and** **c**.id\_customer=t.id **and** **c**.dataset=t.dataset

**order** **by** **c**.**sum** **desc**

\*\*\*\*\*\*\*\*

**CREATE** **OR** **REPLACE** **VIEW** suppliers\_deals\_by\_sum **AS**

**select** t.id, t.type\_name, t.first\_name, t.middle\_name, t.last\_name, t.dataset, **c**.**sum**

**from** suppliers t, suppliers\_deals **c**

**where** t.dataset=1 **and** **c**.id\_supplier=t.id **and** **c**.dataset=t.dataset

**order** **by** **c**.**sum** **desc**

http://localhost:8080/contact\_style.css

**Comments**

// save records

// loop by exists element

// change element's color to red

// add new row and cells

// fill new cells by data

// send and fetch data to / from server without refreshing page

// send data to MvcController without refreshing page

// fetch data from server

//delete records

// clear cells of removed records

// delete buttons if users don't have permission for insert/update/delete records