A5: Relational schema, validation and schema refinement

5. Relational Schema

R01	users(id **UK**, name **NN**, email **UK NN**, photo, address, contact, country,password **NN**, typeUser **NN**)
R02	admin(id **UK,** id_user → user**)**
R03	auction(id **UK**, dateEnd **NN**, dateBegin **NN**, name **NN**, description, buyNow **NN**, active **NN**, id_user → user)
R04	reportUser(id, reason **NN,**id_user → userReporting, id_user → userReported)
R05	comment(like, dislike, date **NN**, comment **NN,** id_user → user, id_auction → auction)
R06	bid(id,status **NN**, price, bidDate **NN**, id_user → user, id_auction → auction)
R07	wishList(date **NN**, follow **NN,** id_user → user, id_auction → auction)
R08	reportAuction(reason **NN**, id_user → user, id_auction → auction)
R09	banAuction(isBanned **NN**, id_user → user, id_auction → auction)
R10	banUser(isBanned **NN**, id_user → user)
R11	owner(id_user → user, id_auction → auction)
R12	buyNow(id_user → user, id_auction → auction)

6. Domains

Today	DATE DEFAULT CURRENT_DATE
Category	ENUM('Electronics','Fashion', 'Home & Garden', 'Motors', 'Music', 'Toys', 'Daily Deals', 'Sporting', 'Others')
Type	ENUM('RegularUser', 'Administrator')

7. Functional Dependencies and schema validation

Table R01 (users)	
Keys: {id, email}	
Functional Dependencies	
FD0101	{id} → {email, name, photo, address, country, contact, password, type}
FD0102	{email} → {id, name, photo, address, country, contact, password, type}
NORMAL FORM	BCNF

Table R02 (admin)	
Keys: {id}	
Functional Dependencies	
FD0201	{id} → {id_user}
NORMAL FORM	BCNF

Table R03 (auction)	
Keys: {id}	
Functional Dependencies	
FD0301	{id} → { dateEnd, dateBegin, name, description, buyNow, active, id_user}
NORMAL FORM	BCNF

Table R04 (reportUser)	
Keys: {id}	
Functional Dependencies	
FD0401	{id} → {reason,id_user,id_admin}
NORMAL FORM	BCNF

Table R05 (comment)	
Keys: {id}	
Functional Dependencies	
FD0501	{id} → {like, dislike, date, comment, id_user, id_auction}
NORMAL FORM	BCNF

Table R06 (bid)	
Keys: {id}	
Functional Dependencies	
FD0601	{id} → {status, price, bidDate, id_user, id_auction}
NORMAL FORM	BCNF

Table R07 (wishList)	
Keys: {id_user, id_auction}	
Functional Dependencies	
FD0701	{id_user,id_auction} → {date, follow}
NORMAL FORM	BCNF

Table R08 (reportAuction)	
Keys: {id_user, id_auction}	
Functional Dependencies	
FD0801	{id_user,id_auction} → {reason}
NORMAL FORM	BCNF

Table R09 (banAuction)	
Keys: {id_user, id_auction}	
Functional Dependencies	
FD0901	{id_user,id_auction} → {isBanned}
NORMAL FORM	BCNF

Table R10 (banUser)	
Keys: {id_user}	
Functional Dependencies	
FD1001	{id_user} → {isBanned}
NORMAL FORM	BCNF

Table R11 (owner)	
Keys: {id_user, id_auction}	
Functional Dependencies	
(none)	
NORMAL FORM	BCNF

Table R12 (buyNow)	
Keys: {id_user, id_auction}	
Functional Dependencies	
(none)	
NORMAL FORM	BCNF

8. SQL Code

```sql
-- Tables

CREATE TABLE users(
 id SERIAL NOT NULL,
 email text NOT NULL UNIQUE,
 password text NOT NULL,
 name text,
 photo text,
 adress text,
 country text,
 contact NUMERIC(17,14),
 type text NOT NULL
);

CREATE TABLE admin(
 id SERIAL NOT NULL,
 id_user INTEGER NOT NULL
);

CREATE TABLE bid(
 id SERIAL NOT NULL,
 status BOOLEAN NOT NULL,
 price FLOAT,
 id_auction INTEGER NOT NULL,
 id_user INTEGER NOT NULL
);

CREATE TABLE auction(
 id SERIAL NOT NULL,
 "date" TIMESTAMP WITH TIME zone DEFAULT now() NOT NULL,
 name text NOT NULL,
 description text,
 buyNow  DOUBLE PRECISION NOT NULL,
 active BOOLEAN NOT NULL
);
```

```sql
CREATE TABLE comment(
 "like" INTEGER,
 dislike INTEGER,
 "date" TIMESTAMP WITH TIME zone DEFAULT now() NOT NULL,
 comment text NOT NULL,
 id_user INTEGER NOT NULL,
 id_auction INTEGER NOT NULL
);

CREATE TABLE reportUser(
 id SERIAL NOT NULL,
 reason text NOT NULL,
 id_user INTEGER NOT NULL,
 id_admin INTEGER NOT NULL
);

CREATE TABLE reportAuction(
 id_user INTEGER NOT NULL,
 id_auction INTEGER NOT NULL,
 reason text NOT NULL,
);


CREATE TABLE banUser(
 id_user INTEGER NOT NULL,
 isBanned BOOLEAN NOT NULL
);

CREATE TABLE banAuction(
 id_user INTEGER NOT NULL,
 id_auction INTEGER NOT NULL,
 isBanned BOOLEAN NOT NULL
);

CREATE TABLE "owner"(
 id_user INTEGER NOT NULL,
 id_auction INTEGER NOT NULL
);

CREATE TABLE wishList(
 id_user INTEGER NOT NULL,
 id_auction INTEGER NOT NULL,
 "date" TIMESTAMP WITH TIME zone DEFAULT now() NOT NULL,
 follow BOOLEAN NOT NULL
);

CREATE TABLE category(
   id_auction INTEGER NOT NULL,
   CATEGORY text NOT NULL,
   CONSTRAINT TYPE CHECK ((CATEGORY = ANY (ARRAY['Electronics'::text, 'Fashion'::text, 'Home &
Garden'::text, 'Motors'::text, 'Music'::text, 'Toys'::text, 'Daily Deals'::text, 'Sporting'::text, 'Others'::text]))));
```

```sql
CREATE TABLE buyNow(
 id_user INTEGER NOT NULL,
 id_auction INTEGER NOT NULL
);

-- Primary Keys and Uniques

ALTER TABLE ONLY users
 ADD CONSTRAINT user_pkey PRIMARY KEY (id);

ALTER TABLE ONLY admin
 ADD CONSTRAINT user_pkey PRIMARY KEY (id,id_user);

ALTER TABLE ONLY auction
 ADD CONSTRAINT auction_pkey PRIMARY KEY (id);

ALTER TABLE ONLY bid
 ADD CONSTRAINT bid_pkey PRIMARY KEY (id);

ALTER TABLE ONLY comment
 ADD CONSTRAINT comment_pkey PRIMARY KEY (id_user, id_auction);

ALTER TABLE ONLY reportUser
 ADD CONSTRAINT reportUser_pkey PRIMARY KEY (id);

ALTER TABLE ONLY reportAuction
 ADD CONSTRAINT reportAuction_pkey PRIMARY KEY (id_user,id_auction);

ALTER TABLE ONLY banUser
 ADD CONSTRAINT banUser_pkey PRIMARY KEY (id_user);

ALTER TABLE ONLY banAuction
 ADD CONSTRAINT banAuction_pkey PRIMARY KEY (id_user,id_auction);
ALTER TABLE ONLY owner
 ADD CONSTRAINT owner_pkey PRIMARY KEY (id_user, id_auction);

ALTER TABLE ONLY wishList
 ADD CONSTRAINT wishList_pkey PRIMARY KEY (id_user, id_auction);

ALTER TABLE ONLY buyNow
 ADD CONSTRAINT buynow_pkey PRIMARY KEY (id_user, id_auction);

-- Foreign Keys

ALTER TABLE ONLY admin
  ADD CONSTRAINT admin_id_auction_fkey FOREIGN KEY (id_user) REFERENCES users(id);

ALTER TABLE ONLY banauction
  ADD CONSTRAINT banauction_id_auction_fkey FOREIGN KEY (id_auction) REFERENCES auction(id);

ALTER TABLE ONLY banauction
  ADD CONSTRAINT banauction_id_user_fkey FOREIGN KEY (id_user) REFERENCES users(id);
```

```sql
ALTER TABLE ONLY banuser
    ADD CONSTRAINT banuser_id_user_fkey FOREIGN KEY (id_user) REFERENCES users(id);

ALTER TABLE ONLY bid
    ADD CONSTRAINT bid_id_auction_fkey FOREIGN KEY (id_auction) REFERENCES auction(id);

ALTER TABLE ONLY bid
    ADD CONSTRAINT bid_id_user_fkey FOREIGN KEY (id_user) REFERENCES users(id);

ALTER TABLE ONLY category
    ADD CONSTRAINT category_id_auction_fkey FOREIGN KEY (id_auction) REFERENCES auction(id);

ALTER TABLE ONLY comment
    ADD CONSTRAINT comment_id_auction_fkey FOREIGN KEY (id_auction) REFERENCES auction(id);

ALTER TABLE ONLY comment
    ADD CONSTRAINT comment_id_user_fkey FOREIGN KEY (id_user) REFERENCES users(id);

ALTER TABLE ONLY owner
    ADD CONSTRAINT owner_id_auction_fkey FOREIGN KEY (id_auction) REFERENCES auction(id);

ALTER TABLE ONLY owner
    ADD CONSTRAINT owner_id_user_fkey FOREIGN KEY (id_user) REFERENCES users(id);

ALTER TABLE ONLY reportauction
    ADD CONSTRAINT reportauction_id_auction_fkey FOREIGN KEY (id_auction) REFERENCES auction(id);

ALTER TABLE ONLY reportauction
    ADD CONSTRAINT reportauction_id_user_fkey FOREIGN KEY (id_user) REFERENCES users(id);

ALTER TABLE ONLY reportuser
    ADD CONSTRAINT reportuser_id_user_fkey FOREIGN KEY (id_user) REFERENCES users(id);

ALTER TABLE ONLY wishlist
    ADD CONSTRAINT wishlist_id_auction_fkey FOREIGN KEY (id_auction) REFERENCES auction(id);

ALTER TABLE ONLY wishlist
    ADD CONSTRAINT wishlist_id_user_fkey FOREIGN KEY (id_user) REFERENCES users(id);

ALTER TABLE ONLY buynow
    ADD CONSTRAINT buynow_id_auction_fkey FOREIGN KEY (id_auction) REFERENCES auction(id);

ALTER TABLE ONLY buynow
    ADD CONSTRAINT buynow_id_user_fkey FOREIGN KEY (id_user) REFERENCES users(id);
```

Figura 1: Sql code

Revision History:

5. Relational Schema

R03	auction(<u>id</u> **UK**, dateEnd **NN**, dateBegin **NN**, name **NN UK**, description, actualPrice **NN,** photo **NN**, buyNow **NN**, active **NN**, <u>id_user</u> → user)
R05	comment(<u>id</u> **UK,** like, dislike, date **NN**, comment **NN,** <u>id_user</u> → user, <u>id_auction</u> → auction)
R09	banAuction(isBanned **NN**, <u>id_user</u> → user, <u>id_auction</u> → auction, dateBegin **NN**)
R10	banUser(<u>id</u> **UK,** isBanned **NN**, <u>id_user</u> → user, <u>id_admin</u>→ admin,dateBegin **NN,** dateEnd)

7. Functional Dependencies and schema validation

Table R03 (auction)	
Keys: {id}	
Functional Dependencies	
FD0301	{id} → {dateEnd, dateBegin, name, description, actualPrice**,** photo, buyNow, active, id_user}
NORMAL FORM	BCNF

Table R09 (banAuction)	
Keys: {id_user, id_auction}	
Functional Dependencies	
FD0901	{id_user,id_auction} → {isBanned,dateBegin}
NORMAL FORM	BCNF

Table R10 (banUser)	
Keys: {id}	
Functional Dependencies	
FD1001	{id} → {isBanned, dateEnd, dateBegin,id_user,id_admin}
NORMAL FORM	BCNF

8. SQL code

```sql
CREATE TABLE auction (
 id integer NOT NULL,
 dateBegin  TIMESTAMP WITH TIME zone DEFAULT now() NOT NULL,
 dateEnd  TIMESTAMP WITH TIME zone DEFAULT now() NOT NULL,
 name text NOT NULL,
 description text,
 actualPrice double precision NOT NULL,
 photo text NOT NULL,
 buynow double precision NOT NULL,
 active boolean NOT NULL
);

CREATE TABLE comment(
 id SERIAL NOT NULL,
 "like" INTEGER,
 dislike INTEGER,
 "date" TIMESTAMP WITH TIME zone DEFAULT now() NOT NULL,
 comment text NOT NULL,
 id_user INTEGER NOT NULL,
 id_auction INTEGER NOT NULL
);

CREATE TABLE banUser(
 id SERIAL NOT NULL,
 id_user INTEGER NOT NULL,
 id_admin  INTEGER NOT NULL,
 isBanned BOOLEAN NOT NULL,
 dateBegin  TIMESTAMP WITH TIME zone DEFAULT now() NOT NULL,
 dateEnd  TIMESTAMP WITH TIME zone DEFAULT now() NOT NULL

);
```

```sql
CREATE TABLE banAuction(
 id_user INTEGER NOT NULL,
 id_auction INTEGER NOT NULL,
 isBanned BOOLEAN NOT NULL,
 dateBegin  TIMESTAMP WITH TIME zone DEFAULT now() NOT NULL
);

ALTER TABLE ONLY comment
 ADD CONSTRAINT comment_pkey PRIMARY KEY (id);
```

Figura 1: Sql code

Group1756 , 19/03/2018
- Diogo Silva, up201405742@fe.up.pt
- Francisca Cerquinho, up201505791@fe.up.pt
- José Manuel Azevedo, up201506448@fe.up.pt
- Pedro Miranda, up201506574@fe.up.pt