**A6: Indexes, triggers, user functions and population**

This artefact contains the physical Postgres schema of our BidBay database. There is also included an analysis of the expected workload on the database and some indexes to help mitigate it. The complete database script is included at the end.

**1. Database workload**

**1.1. Estimate of tuples**

|  |  |  |  |
| --- | --- | --- | --- |
| **Relation reference** | **Relation name** | **Order of magnitude** | **Estimated growth** |
| R01 | users | hundreds | dozens per day |
| R02 | cities | hundreds | static |
| R03 | countries | hundreds | static |
| R04 | bans | dozens | units per day |
| R05 | auctions | thousands | dozens per day |
| R06 | categories | dozens | units per day |
| R07 | bids | thousands | hundreds per day |
| R08 | qas | hundreds | dozens per day |
| R09 | closed\_auctions | hundreds | units per day |
| R10 | won\_auctions | hundreds | dozens per day |
| R11 | reviews | hundreds | dozens per day |
| R12 | messages | thousands | hundreds per day |
| R13 | reports | hundreds | units per day |
| R14 | emails | thousands | hundreds per day |
| R15 | wishlists | thousands | hundreds per day |

**1.2. Most frequent queries**

|  |  |
| --- | --- |
| **Query reference** | SELECT01 |
| **Query description** | User's profile displayed on product page |
| **Query frequency** | tens of thousands per day |

|  |
| --- |
| SELECT username, rating  FROM users WHERE users.id = $ownerId; |

|  |  |
| --- | --- |
| **Query reference** | SELECT02 |
| **Query description** | User's profile page |
| **Query frequency** | thousands per day |

|  |
| --- |
| SELECT username, email, rating, country  FROM (users LEFT JOIN cities ON users.location = cities.id) LEFT JOIN countries ON cities.country\_id = countries.id WHERE users.id = $userId; |
|  |

|  |  |
| --- | --- |
| **Query reference** | SELECT03 |
| **Query description** | User login |
| **Query frequency** | hundreds per day |

|  |
| --- |
| SELECT username  FROM users WHERE users.username = $username AND users.password = $password; |

|  |  |
| --- | --- |
| **Query reference** | SELECT04 |
| **Query description** | Search auctions |
| **Query frequency** | tens of thousands per day |

|  |
| --- |
| SELECT auctions.id  FROM auctions LEFT JOIN closed\_auctions ON auctions.id = closed\_auctions.id WHERE closed\_auctions.id is NULL AND search @@ plainto\_tsquery('english', $searchString)  ORDER BY ts\_rank(search, plainto\_tsquery('english', $searchString)) DESC  LIMIT $resultPerPage; |

|  |  |
| --- | --- |
| **Query reference** | SELECT05 |
| **Query description** | Search auctions by category |
| **Query frequency** | thousands per day |

|  |
| --- |
| SELECT auctions.id  FROM auctions LEFT JOIN closed\_auctions ON auctions.id = closed\_auctions.id WHERE closed\_auctions.id is NULL AND auctions.category\_id = $categoryId AND search @@ plainto\_tsquery('english', $searchString)  ORDER BY ts\_rank(search, plainto\_tsquery('english', $searchString)) DESC  LIMIT $resultPerPage; |

|  |  |
| --- | --- |
| **Query reference** | SELECT06 |
| **Query description** | Auction item on product page |
| **Query frequency** | tens of thousands per day |

|  |
| --- |
| SELECT item\_name, description, current\_price, condition, publication\_date, end\_date, images\_folder, owner\_id, category\_id, city\_id FROM auctions WHERE auctions.id = $auctionId; |

|  |  |
| --- | --- |
| **Query reference** | SELECT07 |
| **Query description** | Auction QA |
| **Query frequency** | tens of thousands per day |

|  |
| --- |
| SELECT question, answer FROM qas JOIN users ON qas.questioner\_id = users.id WHERE auction\_id = $auctionId; |

|  |  |
| --- | --- |
| **Query reference** | SELECT08 |
| **Query description** | Email inbox for user |
| **Query frequency** | hundreds per day |

|  |
| --- |
| SELECT subject, message, send\_date, has\_been\_opened  FROM emails WHERE receiver\_id = $userId  ORDER BY send\_date DESC |

|  |  |
| --- | --- |
| **Query reference** | SELECT09 |
| **Query description** | Recent Auctions |
| **Query frequency** | thousands per day |

|  |
| --- |
| SELECT auctions.id  FROM auctions LEFT JOIN closed\_auctions ON auctions.id = closed\_auctions.id WHERE closed\_auctions.id is NULL ORDER BY auctions.publication\_date DESC  LIMIT $numResults; |

|  |  |
| --- | --- |
| **Query reference** | SELECT10 |
| **Query description** | Auctions Ending Soon |
| **Query frequency** | thousands per day |

|  |
| --- |
| SELECT auctions.id  FROM auctions LEFT JOIN closed\_auctions ON auctions.id = closed\_auctions.id WHERE closed\_auctions.id is NULL ORDER BY auctions.end\_date ASC  LIMIT $numResults; |

|  |  |
| --- | --- |
| **Query reference** | SELECT11 |
| **Query description** | Items auctioned by user |
| **Query frequency** | thousands per day |

|  |
| --- |
| SELECT auctions.id FROM auctions  WHERE auctions.owner\_id = $usedId; |

|  |  |
| --- | --- |
| **Query reference** | SELECT12 |
| **Query description** | Reviews of user |
| **Query frequency** | thousands per day |

|  |
| --- |
| SELECT rating, description FROM reviews JOIN auctions ON reviews.id = auctions.id WHERE auctions.owner\_id = $usedId; |

|  |  |
| --- | --- |
| **Query reference** | SELECT13 |
| **Query description** | Check if user is banned |
| **Query frequency** | thousands per day |

|  |
| --- |
| -- returns nothing if user isn't banned SELECT id FROM bans WHERE ban\_expiration\_date > CURRENT\_TIMESTAMP AND banned\_id = $userID; |

|  |  |
| --- | --- |
| **Query reference** | SELECT14 |
| **Query description** | Admin User Reports |
| **Query frequency** | dozens per day |

|  |
| --- |
| SELECT subject, message  FROM messages JOIN reports ON messages.id = reports.id ORDER BY send\_date DESC |

|  |  |
| --- | --- |
| **Query reference** | SELECT15 |
| **Query description** | Get Auction Winner |
| **Query frequency** | dozens per day |

|  |
| --- |
| -- if returns empty the auctions hasn't been closed yet SELECT bids.bidder\_id  FROM (auctions JOIN won\_auctions ON auctions.id = won\_auctions.id) AS auction JOIN bids ON auction.id = bids.id WHERE auctions.id = $auctionId ORDER BY bid\_amount DESC LIMIT 1 |

**1.3. Frequent Updates**

|  |  |
| --- | --- |
| **Query reference** | UPDATE01 |
| **Query description** | Update user authentication fields |
| **Query frequency** | hundreds per day |

|  |
| --- |
| UPDATE "users"  SET email = $email, username = $username,  first\_name = $first\_name, last\_name = $last\_name,  zip\_code = $zip\_code, address = $address,  location = $location, profile\_picture\_path = $profile\_picture\_path   WHERE id = $id; |
|  |

|  |  |
| --- | --- |
| **Query reference** | UPDATE02 |
| **Query description** | Update auction |
| **Query frequency** | hundreds per day |

|  |
| --- |
| UPDATE "auctions"  SET item\_name = $item\_name, description = $description,  category\_id= $category\_id, condition= $condition,  images\_folder = $images\_folder, starting\_price = $starting\_price,  end\_date = $end\_date, payment\_type = $payment\_type,  shipping\_options = $shipping\_options, shipping\_cost = $shipping\_cost  city\_id = $city\_id,   WHERE id = $id; |
|  |

|  |  |
| --- | --- |
| **Query reference** | UPDATE03 |
| **Query description** | Promote user to admin |
| **Query frequency** | dozens per day |

|  |
| --- |
| UPDATE "users"  SET is\_administrator = TRUE  WHERE id = $id; |
| |  |  | | --- | --- | | **Query reference** | INSERT01 | | **Query description** | Regístrate a new user | | **Query frequency** | dozens per day |  |  | | --- | | INSERT INTO "users" (id, username, first\_name, last\_name, password, email, zip\_code, address, registration\_date, profile\_picture\_path, location, rating, is\_administrator)  VALUES ($id, $username, $first\_name, $last\_name, $password, $email, $zip\_code, $address, $registration\_date, $profile\_picture\_path, $location, $rating, $is\_administrator); | |  |  |  |  | | --- | --- | | **Query reference** | INSERT02 | | **Query description** | Create a new auction | | **Query frequency** | dozens per day |  |  | | --- | | INSERT INTO "auctions" (id, item\_name, description, starting\_price, current\_price, condition, publication\_date, end\_date, payment\_type, shipping\_options, shipping\_cost, images\_folder, owner\_id, category\_id, city\_id)  VALUES ($id, $item\_name, $description, $starting\_price, $current\_price, $condition, $publication\_date, $end\_date, $payment\_type, $shipping\_options, $shipping\_cost, $images\_folder, $owner\_id, $category\_id, $city\_id); | |  |  |  |  | | --- | --- | | **Query reference** | INSERT03 | | **Query description** | Place a bid | | **Query frequency** | hundreds per day |  |  | | --- | | INSERT INTO "bids" (id, bidder\_id, bid\_amount)  VALUES ($id, $bidder\_id, $bid\_amount); | |  |  |  |  | | --- | --- | | **Query reference** | INSERT04 | | **Query description** | Create a new review | | **Query frequency** | dozens per day |  |  | | --- | | INSERT INTO "reviews" (id, rating, description)  VALUES ($id, $rating, $description); | | |  |  | | --- | --- | | **Query reference** | INSERT05 | | **Query description** | Create a new question | | **Query frequency** | dozens per day |  |  | | --- | | INSERT INTO "qas" (id, question, answer, auction\_id, questioner\_id)  VALUES ($id, $question, $answer, $auction\_id, $questioner\_id); | |  |  |  |  | | --- | --- | | **Query reference** | INSERT06 | | **Query description** | Send a message | | **Query frequency** | hundreds per day |  |  | | --- | | INSERT INTO "messages" (id, subject, message, send\_date)  VALUES ($id, $subject, $message, $send\_date); | |  |  |  |  | | --- | --- | | **Query reference** | INSERT07 | | **Query description** | Add to wishlist | | **Query frequency** | dozens per day |  |  | | --- | | INSERT INTO "wishlists" (auction\_id, id)  VALUES ($auction\_id, $id); | |  |  |  |  | | --- | --- | | **Query reference** | INSERT08 | | **Query description** | Ban a user | | **Query frequency** | dozens per day |  |  | | --- | | INSERT INTO "bans" (id, banned\_id, admin, ban\_start\_date, ban\_expiration\_date, ban\_reason)  VALUES ($id, $banned\_id, $admin, $ban\_start\_date, $ban\_expiration\_date, $ban\_reason); | |  |  |  |  | | --- | --- | | **Query reference** | DELETE01 | | **Query description** | Remove an auction | | **Query frequency** | dozens per day |  |  | | --- | | DELETE FROM "auctions"  WHERE id = $id; | |  |  |  |  | | --- | --- | | **Query reference** | DELETE02 | | **Query description** | Delete a message | | **Query frequency** | dozens per day |  |  | | --- | | DELETE FROM "message"  WHERE id = $id; | |  | | |

**2. Proposed Indexes**

**2.1. Performance Indexes**

|  |  |
| --- | --- |
| **Index reference** | IDX01 |
| **Related queries** | SELECT05 |
| **Index relation** | auctions |
| **Index attribute** | category\_id |
| **Index type** | btree |
| **Cardinality** | low |
| **Clustering** | No |
| **Justification** | Query SELECT05 has to be fast since the the auctions table is big and search latency needs to be low; since cardinality is low it’s a bad candidate for clustering. |
| CREATE INDEX category\_auctions\_idx ON auctions USING btree(category\_id); | |

|  |  |
| --- | --- |
| **Index reference** | IDX02 |
| **Related queries** | SELECT08 |
| **Index relation** | emails |
| **Index attribute** | receiver\_id |
| **Index type** | B-tree |
| **Cardinality** | Medium |
| **Clustering** | Yes |
| **Justification** | Query SELECT08 has to be fast since the emails table is big and latency needs to be low; it's a good candidate for clustering since multiple emails with the same receiver\_id are retrieved and cardinality is medium. |
| CREATE INDEX receiver\_emails\_idx ON emails USING btree(receiver\_id); | |

|  |  |
| --- | --- |
| **Index reference** | IDX03 |
| **Related queries** | SELECT09 |
| **Index relation** | auctions |
| **Index attribute** | publication\_date |
| **Index type** | B-tree |
| **Cardinality** | low |
| **Clustering** | No |
| **Justification** | The auctions table is big so searching the whole table will be slow. Multiple results are returned by the query at a time so it’s a good candidate for clustering. |
| CREATE INDEX recent\_auctions\_idx ON auctions USING btree(publication\_date); | |

|  |  |
| --- | --- |
| **Index reference** | IDX04 |
| **Related queries** | SELECT10 |
| **Index relation** | auctions |
| **Index attribute** | end\_date |
| **Index type** | B-tree |
| **Cardinality** | low |
| **Clustering** | No |
| **Justification** | auctions table is big so searching the whole table will be slow. Multiple results are returned by the query at a time so it’s a good candidate for clustering. |
| CREATE INDEX ending\_soon\_auctions\_idx ON auctions USING btree(end\_date); | |

|  |  |
| --- | --- |
| **Index reference** | IDX05 |
| **Related queries** | SELECT07 |
| **Index relation** | qas |
| **Index attribute** | auction\_id |
| **Index type** | B-tree |
| **Cardinality** | medium |
| **Clustering** | Yes |
| **Justification** | Query SELECT07 is executed many times, so it needs to be fast. All Tuples which have the desired id are returned at the same time, so it’s a good candidate for clustering. |
| CREATE INDEX ending\_soon\_auctions\_idx ON auctions USING btree(end\_date); | |

|  |  |
| --- | --- |
| **Index reference** | IDX06 |
| **Related queries** | SELECT11, SELECT12 |
| **Index relation** | auctions |
| **Index attribute** | owner\_id |
| **Index type** | hash |
| **Cardinality** | medium |
| **Clustering** | No |
| **Justification** | Both queries are executed many times, so they need to be fast. Multiple tuples are returned at a time, so it’s a good candidate for clustering. |
| CREATE INDEX owner\_auction\_idx ON auctions USING hash(owner\_id); | |

|  |  |
| --- | --- |
| **Index reference** | IDX07 |
| **Related queries** | SELECT13 |
| **Index relation** | bans |
| **Index attribute** | banned\_id |
| **Index type** | hash |
| **Cardinality** | medium |
| **Clustering** | Yes |
| **Justification** | The query is executed many times and latency needs to be low. Tuples with the same owner\_id are used by the query. |
| CREATE INDEX bannee\_bans\_idx ON bans USING hash(banned\_id); | |

|  |  |
| --- | --- |
| **Index reference** | IDX08 |
| **Related queries** | SELECT08, SELECT14 |
| **Index relation** | messages |
| **Index attribute** | send\_date |
| **Index type** | B-tree |
| **Cardinality** | high |
| **Clustering** | Yes |
| **Justification** | Query SELECT08 and SELECT14 has to be fast since the messages table is big and latency needs to be low; it's a good candidate for clustering since messages are retrieved by arrival data. |
| CREATE INDEX date\_messages\_idx ON messages USING btree(send\_date); | |

**2.2. Full-text Search indexes**

|  |  |
| --- | --- |
| **Index reference** | IDX09 |
| **Related queries** | SELECT04 |
| **Index relation** | auctions |
| **Index attribute** | search |
| **Index type** | GIN |
| **Clustering** | Yes |
| **Justification** | To improve the performance of full text searches for aution’s title and description; GIN because it’s the fastest one. Each query returns multiple results, so it’s a good candidate for clustering. |
| CREATE INDEX search\_auctions\_idx ON auctions USING GIN (search); | |

**3. Triggers**

|  |  |
| --- | --- |
| **Trigger reference** | TRIGGER01 |
| **Trigger description** | Pre-Calculates Auctions search ts\_vector |
| CREATE OR REPLACE FUNCTION auctions\_search\_update() RETURNS TRIGGER AS $$ BEGIN  IF TG\_OP = 'INSERT' THEN  NEW.search = setweight(to\_tsvector('english', NEW.item\_name), 'A') || setweight(to\_tsvector('english', NEW.description), 'D');  END IF;  IF TG\_OP = 'UPDATE' THEN  IF NEW.item\_name <> OLD.item\_name OR NEW.description <> OLD.description THEN  NEW.search = setweight(to\_tsvector('english', NEW.item\_name), 'A') || setweight(to\_tsvector('english', NEW.description), 'D');  END IF;  END IF;  RETURN NEW; END $$ LANGUAGE 'plpgsql';  CREATE TRIGGER auctions\_search\_tsvector BEFORE INSERT OR UPDATE ON auctions FOR EACH ROW EXECUTE PROCEDURE auctions\_search\_update(); | |

|  |  |
| --- | --- |
| **Trigger reference** | TRIGGER02 |
| **Trigger description** | Update current price on auctions after insert on bids |
| DROP TRIGGER IF EXISTS update\_current\_price ON bids; CREATE OR REPLACE FUNCTION fn\_update\_current\_price() RETURNS TRIGGER AS $$  BEGIN  IF TG\_OP = 'INSERT' THEN  IF NEW.bid\_amount> (SELECT current\_price FROM auctions WHERE id=NEW.id) THEN  UPDATE auctions  SET current\_price = NEW.bid\_amount  WHERE id = NEW.id;  END IF;  END IF;  RETURN NEW;  END  $$ LANGUAGE 'plpgsql';   CREATE TRIGGER update\_current\_price BEFORE INSERT ON bids FOR EACH ROW EXECUTE PROCEDURE fn\_update\_current\_price(); | |

|  |  |
| --- | --- |
| **Trigger reference** | TRIGGER03 |
| **Trigger description** | Check if who is banning is an administrator |
| DROP TRIGGER IF EXISTS check\_if\_is\_admin ON bans; CREATE OR REPLACE FUNCTION fn\_check\_if\_is\_admin() RETURNS TRIGGER AS $$ BEGIN  IF TG\_OP = 'INSERT' THEN  IF NEW.admin IN (SELECT users.id FROM users, bans WHERE users.id=bans.admin AND users.is\_administrator is FALSE) THEN  RAISE EXCEPTION 'You are not administrator! You dont have permissions to ban!';  END IF;   END IF;  RETURN NEW; END $$ LANGUAGE 'plpgsql';   CREATE TRIGGER check\_if\_is\_admin BEFORE INSERT ON bans FOR EACH ROW EXECUTE PROCEDURE fn\_check\_if\_is\_admin(); | |

|  |  |
| --- | --- |
| **Trigger reference** | TRIGGER04 |
| **Trigger description** | Check if who is bidding is not an administrator |
| DROP TRIGGER IF EXISTS fn\_add\_bid\_only\_if\_not\_admin\_banned ON bids; CREATE OR REPLACE FUNCTION fn\_add\_bid\_only\_if\_not\_admin\_banned() RETURNS TRIGGER AS $$ BEGIN  IF TG\_OP = 'INSERT' THEN  IF NEW.bidder\_id IN (SELECT users.id FROM users, bids WHERE users.id= NEW.bidder\_id AND users.is\_administrator = true) THEN  RAISE EXCEPTION 'You are an administrator! You can not bid';  END IF;  END IF;  RETURN NEW; END  $$ LANGUAGE 'plpgsql';   CREATE TRIGGER fn\_add\_bid\_only\_if\_not\_admin\_banned BEFORE INSERT ON bids FOR EACH ROW EXECUTE PROCEDURE fn\_add\_bid\_only\_if\_not\_admin\_banned(); | |

|  |  |
| --- | --- |
| **Trigger reference** | TRIGGER05 |
| **Trigger description** | Check if who is bidding is not an auctioneer |
| DROP TRIGGER IF EXISTS fn\_check\_if\_bidder\_is\_not\_auctioneer ON bids; CREATE OR REPLACE FUNCTION fn\_check\_if\_bidder\_is\_not\_auctioneer() RETURNS TRIGGER AS $$ BEGIN  IF TG\_OP = 'INSERT' THEN  IF NEW.bidder\_id = (SELECT owner\_id FROM auctions WHERE id = NEW.id) THEN  RAISE EXCEPTION 'You cannot bid your own product!';  END IF;   END IF;  RETURN NEW; END  $$ LANGUAGE 'plpgsql';   CREATE TRIGGER check\_if\_bidder\_is\_not\_auctioneer BEFORE INSERT ON bids FOR EACH ROW EXECUTE PROCEDURE fn\_check\_if\_bidder\_is\_not\_auctioneer(); | |

|  |  |
| --- | --- |
| **Trigger reference** | TRIGGER06 |
| **Trigger description** | Update users rating after insert a review |
| DROP TRIGGER IF EXISTS fn\_update\_user\_rating\_after\_review ON reviews; CREATE OR REPLACE FUNCTION fn\_update\_user\_rating\_after\_review() RETURNS TRIGGER AS $$ BEGIN  IF TG\_OP = 'INSERT' THEN  UPDATE users  SET rating = (SELECT avg(reviews.rating) FROM reviews WHERE id in ( SELECT id FROM auctions WHERE owner\_id = (SELECT owner\_id FROM auctions WHERE auctions.id = NEW.id)))  WHERE id = (SELECT owner\_id from auctions WHERE id = NEW.id);  END IF;  RETURN NEW; END  $$ LANGUAGE 'plpgsql';   CREATE TRIGGER update\_user\_rating\_after\_review AFTER INSERT ON reviews FOR EACH ROW EXECUTE PROCEDURE fn\_update\_user\_rating\_after\_review(); | |

|  |  |
| --- | --- |
| **Trigger reference** | TRIGGER07 |
| **Trigger description** | Update users rating after delete a review |
| DROP TRIGGER IF EXISTS update\_user\_rating\_after\_delete\_review ON reviews; CREATE OR REPLACE FUNCTION fn\_update\_user\_rating\_after\_delete\_review() RETURNS TRIGGER AS $$ BEGIN  IF TG\_OP = 'DELETE' THEN  UPDATE users  SET rating = (SELECT avg(reviews.rating) FROM reviews WHERE id in (  SELECT id FROM auctions WHERE owner\_id = (SELECT owner\_id FROM auctions WHERE auctions.id = OLD.id)))  WHERE id = (SELECT owner\_id from auctions WHERE id = OLD.id);  END IF;  RETURN OLD;  END  $$ LANGUAGE 'plpgsql';   CREATE TRIGGER update\_user\_rating\_after\_delete\_review AFTER DELETE ON reviews FOR EACH ROW EXECUTE PROCEDURE fn\_update\_user\_rating\_after\_delete\_review(); | |

**4. Código SQL**

Link for the GitHub repository code: <https://github.com/afonsobspinto/lbaw1764/blob/master/databases/database.sql>

|  |
| --- |
| DROP TABLE IF EXISTS auctions CASCADE; DROP TABLE IF EXISTS bans CASCADE; DROP TABLE IF EXISTS bids CASCADE; DROP TABLE IF EXISTS categories CASCADE; DROP TABLE IF EXISTS cities CASCADE; DROP TABLE IF EXISTS countries CASCADE; DROP TABLE IF EXISTS "users" CASCADE; DROP TABLE IF EXISTS closed\_auctions CASCADE; DROP TABLE IF EXISTS emails CASCADE; DROP TABLE IF EXISTS messages CASCADE; DROP TABLE IF EXISTS qas CASCADE; DROP TABLE IF EXISTS reports CASCADE; DROP TABLE IF EXISTS reviews CASCADE; DROP TABLE IF EXISTS wishlists CASCADE; DROP TABLE IF EXISTS won\_auctions CASCADE;  CREATE TABLE categories (  id integer NOT NULL PRIMARY KEY,  name character(50) );  CREATE TABLE countries (  id integer NOT NULL PRIMARY KEY,  country character(50) );  CREATE TABLE cities (  id integer NOT NULL PRIMARY KEY,  city character(50) NOT NULL,  country\_id integer NOT NULL REFERENCES countries(id) ON UPDATE CASCADE );   CREATE TABLE "users" (  id integer NOT NULL PRIMARY KEY,  username character(50),  first\_name character(50),  last\_name character(50),  password character(50) NOT NULL,  email character(50) NOT NULL UNIQUE,  zip\_code character(25),  address character(50),  "registration\_date" timestamp DEFAULT CURRENT\_TIMESTAMP,  profile\_picture\_path character(250),  location integer REFERENCES cities(id) ON UPDATE CASCADE,  rating real CONSTRAINT rating\_ck CHECK (((rating > 1.0) AND (rating <= 5.0))),  is\_administrator boolean DEFAULT false );  CREATE TABLE auctions (  id serial PRIMARY KEY,  item\_name text NOT NULL,  description text,  starting\_price real NOT NULL CONSTRAINT starting\_price\_ck CHECK (starting\_price >0.0),  current\_price real DEFAULT null,  condition character(50) NOT NULL,  CONSTRAINT condition CHECK ((condition = ANY (ARRAY['New'::text, 'Used'::text, 'Not Working'::text]))),  "publication\_date" timestamp DEFAULT CURRENT\_TIMESTAMP,  end\_date timestamp NOT NULL CONSTRAINT end\_date\_ck CHECK (end\_date > "publication\_date"),  payment\_type character(25) NOT NULL,  shipping\_options character(25) NOT NULL,  CONSTRAINT payment\_type CHECK ((payment\_type = ANY (ARRAY['PayPal'::text, 'Credit Card'::text, 'Bank Transfer'::text, 'Other'::text]))),  CONSTRAINT shipping\_options CHECK ((shipping\_options = ANY (ARRAY['Domestic Shipping'::text, 'International Shipping'::text, 'No shipping'::text]))),  shipping\_cost real CONSTRAINT shipping\_cost\_ck CHECK (shipping\_cost >0.0),  images\_folder character(250),  owner\_id integer NOT NULL REFERENCES "users"(id) ON UPDATE CASCADE,  category\_id integer NOT NULL REFERENCES categories(id) ON UPDATE CASCADE,  city\_id integer NOT NULL REFERENCES cities(id) ON UPDATE CASCADE,  search tsvector );     CREATE TABLE bans (  id serial PRIMARY KEY,  banned\_id integer NOT NULL REFERENCES "users"(id) ON UPDATE CASCADE,  admin integer NOT NULL REFERENCES "users"(id) ON UPDATE CASCADE,  "ban\_start\_date" timestamp DEFAULT CURRENT\_TIMESTAMP,  ban\_expiration\_date timestamp CONSTRAINT banExpiration\_ck CHECK (ban\_expiration\_date>"ban\_start\_date"),  ban\_reason text NOT NULL );   CREATE TABLE bids (  id integer NOT NULL PRIMARY KEY REFERENCES auctions(id) ON UPDATE CASCADE,  bidder\_id integer REFERENCES "users"(id) ON UPDATE CASCADE, --CONSTRAINT bidder\_id\_ck CHECK (bidder\_id <> id.owner\_id),  bid\_amount integer NOT NULL );    CREATE TABLE messages (  id serial PRIMARY KEY,  subject text NOT NULL,  message text NOT NULL,  "send\_date" timestamp DEFAULT CURRENT\_TIMESTAMP );   CREATE TABLE closed\_auctions (  id integer NOT NULL PRIMARY KEY REFERENCES auctions(id) ON UPDATE CASCADE );  CREATE TABLE emails (  id integer NOT NULL PRIMARY KEY REFERENCES messages(id) ON UPDATE CASCADE,  "has\_been\_opened" boolean DEFAULT false NOT NULL,  receiver\_id integer NOT NULL REFERENCES "users"(id) ON UPDATE CASCADE,  sender\_id integer NOT NULL REFERENCES "users"(id) ON UPDATE CASCADE );    CREATE TABLE qas (  id serial PRIMARY KEY REFERENCES auctions(id) ON UPDATE CASCADE,  question text NOT NULL,  answer text,  auction\_id integer NOT NULL,  questioner\_id integer NOT NULL REFERENCES "users"(id) ON UPDATE CASCADE );  CREATE TABLE reports (  id integer NOT NULL PRIMARY KEY REFERENCES messages(id) ON UPDATE CASCADE );  CREATE TABLE reviews (  id integer NOT NULL PRIMARY KEY REFERENCES auctions(id) ON UPDATE CASCADE,  rating integer NOT NULL CONSTRAINT rating\_ck CHECK (((rating >= 0) OR (rating <= 5))),  description text NOT NULL );  CREATE TABLE wishlists (  auction\_id integer NOT NULL PRIMARY KEY REFERENCES auctions(id) ON UPDATE CASCADE,  id integer NOT NULL REFERENCES "users"(id) ON UPDATE CASCADE );  CREATE TABLE won\_auctions (  id integer NOT NULL PRIMARY KEY REFERENCES auctions(id) ON UPDATE CASCADE,  "is\_successful\_transaction" boolean DEFAULT false NOT NULL,  "has\_winner\_complained" boolean DEFAULT false NOT NULL,  winner\_id integer NOT NULL REFERENCES "users"(id) ON UPDATE CASCADE ); |

Link for the GitHub repository code: <https://github.com/afonsobspinto/lbaw1764/blob/master/databases/populate.sql>

|  |
| --- |
| INSERT INTO "public"."categories" ("id","name") VALUES (1,'mE2j0rkp7DHY3i47jCtmXs1o');     INSERT INTO "public"."categories" ("id","name") VALUES (2,NULL);     INSERT INTO "public"."categories" ("id","name") VALUES (3,'h8Fpi80yKkrtZXXazmm3xiFOVcyl4bUGAZk');     INSERT INTO "public"."categories" ("id","name") VALUES (4,'5yOECwAD3iXtgKtkjKZG5xVNr5k0');     INSERT INTO "public"."categories" ("id","name") VALUES (5,'mboqS');     INSERT INTO "public"."categories" ("id","name") VALUES (6,'ZJvromwckSamMDUVIl5v48');     INSERT INTO "public"."categories" ("id","name") VALUES (7,'51gO3Reap4');     INSERT INTO "public"."categories" ("id","name") VALUES (8,NULL);     INSERT INTO "public"."categories" ("id","name") VALUES (9,'OFre4');     INSERT INTO "public"."categories" ("id","name") VALUES (10,'fsD8WWVwRHBqkiZ');        INSERT INTO "public"."countries" ("id","country") VALUES (1,'Kiribati');     INSERT INTO "public"."countries" ("id","country") VALUES (2,'French Southern Territories');     INSERT INTO "public"."countries" ("id","country") VALUES (3,'Ghana');     INSERT INTO "public"."countries" ("id","country") VALUES (4,'Belize');     INSERT INTO "public"."countries" ("id","country") VALUES (5,'Ethiopia');     INSERT INTO "public"."countries" ("id","country") VALUES (6,NULL);     INSERT INTO "public"."countries" ("id","country") VALUES (7,'United Arab Emirates');     INSERT INTO "public"."countries" ("id","country") VALUES (8,'Nigeria');     INSERT INTO "public"."countries" ("id","country") VALUES (9,'Suriname');     INSERT INTO "public"."countries" ("id","country") VALUES (10,'S. Georgia and S. Sandwich Isls.');     INSERT INTO "public"."countries" ("id","country") VALUES (11,'Iran');     INSERT INTO "public"."countries" ("id","country") VALUES (12,'Malta');     INSERT INTO "public"."countries" ("id","country") VALUES (13,'Gambia');     INSERT INTO "public"."countries" ("id","country") VALUES (14,'Portugal');     INSERT INTO "public"."countries" ("id","country") VALUES (15,'Lebanon');     INSERT INTO "public"."countries" ("id","country") VALUES (16,'Netherlands');     INSERT INTO "public"."countries" ("id","country") VALUES (17,'Kiribati');     INSERT INTO "public"."countries" ("id","country") VALUES (18,'Italy');     INSERT INTO "public"."countries" ("id","country") VALUES (19,'Mozambique');     INSERT INTO "public"."countries" ("id","country") VALUES (20,'Heard and McDonald Islands');     INSERT INTO "public"."countries" ("id","country") VALUES (21,'Brazil');     INSERT INTO "public"."countries" ("id","country") VALUES (22,'Egypt');     INSERT INTO "public"."countries" ("id","country") VALUES (23,NULL);     INSERT INTO "public"."countries" ("id","country") VALUES (24,NULL);     INSERT INTO "public"."countries" ("id","country") VALUES (25,'Monaco');     INSERT INTO "public"."countries" ("id","country") VALUES (26,'French Southern Territories');     INSERT INTO "public"."countries" ("id","country") VALUES (27,'United Arab Emirates');     INSERT INTO "public"."countries" ("id","country") VALUES (28,'Syria');     INSERT INTO "public"."countries" ("id","country") VALUES (29,NULL);     INSERT INTO "public"."countries" ("id","country") VALUES (30,'Nicaragua');     INSERT INTO "public"."countries" ("id","country") VALUES (31,NULL);     INSERT INTO "public"."countries" ("id","country") VALUES (32,'US Minor Outlying Islands');     INSERT INTO "public"."countries" ("id","country") VALUES (33,NULL);     INSERT INTO "public"."countries" ("id","country") VALUES (34,'Mali');     INSERT INTO "public"."countries" ("id","country") VALUES (35,NULL);     INSERT INTO "public"."countries" ("id","country") VALUES (36,'Finland');     INSERT INTO "public"."countries" ("id","country") VALUES (37,NULL);     INSERT INTO "public"."countries" ("id","country") VALUES (38,'Cameroon');     INSERT INTO "public"."countries" ("id","country") VALUES (39,'Virgin Islands (U.S.)');     INSERT INTO "public"."countries" ("id","country") VALUES (40,'Russian Federation');     INSERT INTO "public"."countries" ("id","country") VALUES (41,'Ghana');     INSERT INTO "public"."countries" ("id","country") VALUES (42,'Honduras');     INSERT INTO "public"."countries" ("id","country") VALUES (43,'Aruba');     INSERT INTO "public"."countries" ("id","country") VALUES (44,'Botswana');     INSERT INTO "public"."countries" ("id","country") VALUES (45,'Zimbabwe');     INSERT INTO "public"."countries" ("id","country") VALUES (46,'Cote D''Ivoire (Ivory Coast)');     INSERT INTO "public"."countries" ("id","country") VALUES (47,'Bahrain');     INSERT INTO "public"."countries" ("id","country") VALUES (48,'Guatemala');     INSERT INTO "public"."countries" ("id","country") VALUES (49,NULL);     INSERT INTO "public"."countries" ("id","country") VALUES (50,NULL);        INSERT INTO "public"."messages" ("id","subject","message","send\_date") VALUES (1,'czGzXwEEjqjx6gyk1YnaVAQw1Ds5j4NfynInIyrTGZeBimKeWzrU4Iy4YD0IYEUfID5An2WMYmebnw1xsQCc7YUv1BKbX0OiddibtqJQNZdUEluZWR5QEBK7ArRed1o8XS','Qj8aixfPmTeMcNnAN1','11/26/2000 10:21:00');     INSERT INTO "public"."messages" ("id","subject","message","send\_date") VALUES (2,'U0ncfZbDVOHZThv2dGnDfmmZKeerlzQU6XcaMLlebWbtDw3UVuwUZpJSI8rAlfCqsr43puJ61U1NKUHAq5CENUby3zUmK7nORM3Q6Hi0eJuLOGktujJloZyqvLb55ILlAVqrkSJWK5mvXrYFMBFcSmenT81UsVc6BoVSRbjAZcOvAghHTNmvWmuY48ZRQn8r2','ZziNgQ7tHjsIIj1dGPo8VVfFwyv0YZa2f0ttLgkzkIvoOdkc6FnJk2e8fORR35WHEnQAQTIQlGC8F7OlZLgbcGwiaytQzgtAFIAsZ6JCweB5hXwFvv4lQaNdiViHe32BVcf7GRSNp8SS8ROIrA88SGoTZZmrT6jqMlfmNfhuV0yTPKUX44gkS','10/26/2009 06:19:00');     INSERT INTO "public"."messages" ("id","subject","message","send\_date") VALUES (3,'huyRPnXkUEOhn3625CNS8kLmCHqiWUCUqZZyZwlKJLT7MzD4','G0qarV2iZwcXlgnS7al3eOu','07/03/2005 08:02:00');     INSERT INTO "public"."messages" ("id","subject","message","send\_date") VALUES (4,'OsuIQhtwOk0bPfaX3rDXiPVYjcWshH5uPURx63GkDYz0Enw6qWsTpprnj267SXcDAA5pdKIJPCAfLjDdEy2BP0eLUeP','irYQr5kTno5GjAbgpgCG2vGJp1mjN21TMnWVgzexxOYMMdILDeogZMiFdjcyfgPc78tCfPQanscoBmsTuMnAhsPGG7nEiRDwLQ','07/04/2000 01:55:00');     INSERT INTO "public"."messages" ("id","subject","message","send\_date") VALUES (5,'deX8ZPbDdpdUhyjqBD7jHaSWL2Op8UmjPPpUkgglr8hTqvTzYTELIUdUhLlWOdmKkQr3NM2xLmL2IM4fJ8e1bj6ECYRg4WOIuIQ4Xzj652ckpHA70RrWdp6VR67XD85rZkUKiZ','IB1FysKucmBpRPinp3LmtmkotBE2ZDWLUNL2IdEIXSzhvBxQkqnftuRjqZbkBP4','08/19/2012 04:51:00');     INSERT INTO "public"."messages" ("id","subject","message","send\_date") VALUES (6,'fOoHZTVXph5iIOTLm716XXw624ZhltQ82jeDkzB8pew6Hv2nGvgHVDNcjIdDOnEIalOWDmCGywRGjxDbWijyoRqeaRIrZmvSbMQ82XcA5mlLL54oCnJg0pG24JOq8dRWSP38SOAj2RFrQmNah3NwYXfQrfplbDwmSHNt0xBCkRF4KZaflNIliHWMkeldqxqw01DZyMKMZJAeNKGu51xCuIm5Ofe','NKz6TbzWfyj0wnNWqNMB7lgrnjIa8BYHM4A8Q5REGvN0i3ZDEbeW000DxeAD503H42nygNk8fheRfI0QlEQ','10/29/2013 08:47:00');     INSERT INTO "public"."messages" ("id","subject","message","send\_date") VALUES (7,'De5EUi6zelaVEufO2rKGZyF1wLD3Lo3JeVZKZsXGK2gocCvvJYdSegTfGz8EyTYjO2kL7Ey','bauHM21euQRZR3208LizEtZfh7LSL4HL2J',NULL);     INSERT INTO "public"."messages" ("id","subject","message","send\_date") VALUES (8,'nsnstrOE1YX2YCeCvVgtRvnIF3df3tYttlwJ3gFsu1qgUWWH2RrO7q2Hcsku2y6iYVxLfRolppRUfNKbK3ZZCQ5aF3YHpH8Z5ylH86Fhe6qsYs5CtGj4g3UfzPQq03VkuRngRm8R2jxnh16gVjyFthVrzoY257lHnsOvzzdR4QsoeWhJlo','Zlqo0A2d4dgyzXzohQKEKb','10/18/2000 02:05:00');     INSERT INTO "public"."messages" ("id","subject","message","send\_date") VALUES (9,'OIAxjji3gO3WjaToLM5nyzhGFBtZWqgLpJQXGMTQbf6PAmUAbGjU2Tp0pJzvEXQKdGMFTokJjcGdEL2DnUNHZfqBbME5c27WmnY1ufFqa0k77wwLriBy4i0t6Iqj1D1QiXSFoV6GR','wkrce1a7Mx8R5fUUb2Sv0GaTXxSdB6FbwxR2XAaIwsMfJVM26iwAvFwh8GMQqhfiQeNqL2ep1ZX1ZmKouex7TQKI0PEuI1VSkoIipzcwbchF6GroyEGkiINBlkREAsrKDyM2q5hkwPhvva0aaw1EXtPcXkaWAPBDmiNe2kW4Hz6YhOa3OtZPFOLcyxsPiQ07PjbEaRkMJK4feYinbXN0LoS4FQRr8aaZqjgf2iSM1oednH4tXqYFHnpe',NULL);     INSERT INTO "public"."messages" ("id","subject","message","send\_date") VALUES (10,'cJffKxYJ7WVl2OTjR3sZ1VXEeWBW0O6fm7qqPU2N6TzHsSheBxRmvCvOW2s6VjGjDqCq4R2LQBFbyQzG0aUbBFPYFZOrSBnugUoGqVshm7SyVvlPMZNnZnkkl5mWOLBt145TibQ8CLV7yFkYjqedVWbS6U2kWW3XWgaqHSgkg7WLzmCR7PeOSPH7qJ1lGldygouHidizzJUPnJCLoG2g866FDB3JpYFaN5','ywlD5byLg0nJVjs3eZs5GNzjdTsGQuP3Cw','08/24/2007 02:48:00');        INSERT INTO "public"."reports" ("id") VALUES (1);     INSERT INTO "public"."reports" ("id") VALUES (2);     INSERT INTO "public"."reports" ("id") VALUES (3);     INSERT INTO "public"."reports" ("id") VALUES (4);     INSERT INTO "public"."reports" ("id") VALUES (5);           INSERT INTO "public"."cities" ("id","city","country\_id") VALUES (1,'Nizhny Novgorod', 1);     INSERT INTO "public"."cities" ("id","city","country\_id") VALUES (2,'Weifang', 1);     INSERT INTO "public"."cities" ("id","city","country\_id") VALUES (3,'Taiyuan',2);     INSERT INTO "public"."cities" ("id","city","country\_id") VALUES (4,'Dallas (TX)',3);     INSERT INTO "public"."cities" ("id","city","country\_id") VALUES (5,'TEHRAN',4);     INSERT INTO "public"."cities" ("id","city","country\_id") VALUES (6,'Pueblade Zaragoza',4);     INSERT INTO "public"."cities" ("id","city","country\_id") VALUES (7,'Quezon City',5);     INSERT INTO "public"."cities" ("id","city","country\_id") VALUES (8,'Fulin',5);     INSERT INTO "public"."cities" ("id","city","country\_id") VALUES (9,'Kunming',5);     INSERT INTO "public"."cities" ("id","city","country\_id") VALUES (10,'BUDAPEST',6);        INSERT INTO "public"."users" ("id","username","first\_name","last\_name","password","email","zip\_code","address","registration\_date","profile\_picture\_path","location","rating","is\_administrator") VALUES (1,'Frederik804','Ross',NULL,'BSS8Q1EBkaFYrQdhWOtudgYeVTeFpWW1qnfPhaD','E.Young@libero.cn','94167','31 Stewart Ave','04/04/2011 06:48:00','C:\DOCS\DATA46',1,1.58,True);     INSERT INTO "public"."users" ("id","username","first\_name","last\_name","password","email","zip\_code","address","registration\_date","profile\_picture\_path","location","rating","is\_administrator") VALUES (2,'Mads9','Marie','Katsekes','KoXWfF0CzM6UzCuKOA8p8GbH','Mick.Byrnes5@telfort.cc','42767','8 Brooke Valley Drive','06/25/2013 03:26:00','C:\PROGRAM FILES\TEMP98', 1,1.1,True);    INSERT INTO "public"."users" ("id","username","first\_name","last\_name","password","email","zip\_code","address","registration\_date","profile\_picture\_path","location","rating","is\_administrator") VALUES (3,'Richard889',NULL,'Morton','h2CwXXCEBU8KHcBrnsf0DvPLRwCqkuqGQGQrKZczLwmTG','Dave.Petterson@dolfijn.dk','84063',NULL,NULL,'C:\DOCUMENTS\TEMP26', 1,4.01,True);    INSERT INTO "public"."users" ("id","username","first\_name","last\_name","password","email","zip\_code","address","registration\_date","profile\_picture\_path","location","rating","is\_administrator") VALUES (4,'Dave3','Isabel','Bloom','uIzryBfxvW0I4QGAZjU3Qba5tx5JLa','JohanDepew@libero.be','39927','4235 Serang Place','12/04/2006 00:22:00','pathy', 2,3.73,True);     INSERT INTO "public"."users" ("id","username","first\_name","last\_name","password","email","zip\_code","address","registration\_date","profile\_picture\_path","location","rating","is\_administrator") VALUES (5,'Magnus420','Olivia','Stewart','wgD3eJPKOWH4B6fev32IW2q1gWTckCT0YCzug3','E.King3@dolfijn.fr','73108','20 Devonshire Rd','12/06/2016 06:23:00','C:\TEMPORARY\DATA76', 2,3.21,False);     INSERT INTO "public"."users" ("id","username","first\_name","last\_name","password","email","zip\_code","address","registration\_date","profile\_picture\_path","location","rating","is\_administrator") VALUES (6,'Sophia812','Olivia','Swaine','btF7nhhTFQ47Ky6UfGaOqsIIFe','YLamere@excite.us',NULL,'0 Melrose Ave',NULL,'C:\TEMP\DOCS03', 3,1.22,NULL);     INSERT INTO "public"."users" ("id","username","first\_name","last\_name","password","email","zip\_code","address","registration\_date","profile\_picture\_path","location","rating","is\_administrator") VALUES (7,'Tim16',NULL,'Bernstein','AntqmXxP126y7GdcM','Trees.King@hotmail.gov','94588','991 East MacArthur','11/29/2006 04:42:00','C:\TEMPORARY\DOC87', 3,4.91,True);     INSERT INTO "public"."users" ("id","username","first\_name","last\_name","password","email","zip\_code","address","registration\_date","profile\_picture\_path","location","rating","is\_administrator") VALUES (8,'Louise00','Cloe','Nahay','AVICgUWfmHgEI0z8Yj0UBsFtieg8qya5ZL1rwAulDrlcQsQzm','LindsyBrown@weboffice.no','71654','2024 Melrose Ave','11/30/2011 01:07:00','C:\TEMP\TEMP57', 3,3.48,False);     INSERT INTO "public"."users" ("id","username","first\_name","last\_name","password","email","zip\_code","address","registration\_date","profile\_picture\_path","location","rating","is\_administrator") VALUES (9,'Henk241','Oscar','Poissant','6deWU7sUNR','E.Cramer@dolfijn.us',NULL,'8 N. Jackson Road','08/09/2003 10:01:00','C:\DATA\DOCS91', 4,2.21,NULL);     INSERT INTO "public"."users" ("id","username","first\_name","last\_name","password","email","zip\_code","address","registration\_date","profile\_picture\_path","location","rating","is\_administrator") VALUES (10,NULL,'Peggy',NULL,'6BbknOftvtVyJWWCNu3zMCQY1ORPcfiT7IvJrFW75vzcY','JohanTudisco5@dolfijn.com',NULL,'0 W. Broadway','01/29/2014 04:25:00','dwad', 5,1.78,False);        INSERT INTO "public"."bans" ("id","banned\_id","admin","ban\_start\_date","ban\_expiration\_date","ban\_reason") VALUES (1, 1, 5,'07/02/2001 04:15:00','07/11/2016 09:14:00','6LgRbEbX3RBWeGZsNIhVgaA1bXEdcfTVaNrgkjzYo3wnu8DhLgWSy4faOWd1Y8IuGLCQ2BmbfNdqZNIjwV3DyhvnkPZRlMZXfUykHWN1wY33Eb7zE86hASKPo2tIX0Cpugm1kgarlD3wvr');     INSERT INTO "public"."bans" ("id","banned\_id","admin","ban\_start\_date","ban\_expiration\_date","ban\_reason") VALUES (2, 1, 5,'06/05/2012 06:13:00',NULL,'bxIYx5OjvL5V6rR4AAKxxDsRuj2hB6wCe11uwNwPgwKo');        INSERT INTO "public"."emails" ("id","has\_been\_opened","receiver\_id","sender\_id") VALUES (6,False, 1, 2);     INSERT INTO "public"."emails" ("id","has\_been\_opened","receiver\_id","sender\_id") VALUES (7,False, 1, 5);     INSERT INTO "public"."emails" ("id","has\_been\_opened","receiver\_id","sender\_id") VALUES (8,False, 2, 3);     INSERT INTO "public"."emails" ("id","has\_been\_opened","receiver\_id","sender\_id") VALUES (9,True, 3, 4);     INSERT INTO "public"."emails" ("id","has\_been\_opened","receiver\_id","sender\_id") VALUES (10,True, 5, 1);         INSERT INTO "public"."auctions" ("id","item\_name","description","starting\_price","current\_price","condition","publication\_date","end\_date","payment\_type","shipping\_options","shipping\_cost","images\_folder","owner\_id","category\_id","city\_id","search") VALUES (1,'Lv5QAwM3aBBQ70mN0DgN8J1EH06AA7c6F6vbwMygPV575I1DwXlRU6oduc1bHOP8Iw7mdHv8HhDsgq55YrCHTRkr3CdUBT0lCi5uOcw0bPdEhtr8JiqeaTZtXkfawXTdpvjUK8tXAgcpovMGmpdItYPUxoIny',NULL,8049.15,NULL,'Used','01/07/2000 00:50:00','01/21/2012 04:30:00','PayPal','No shipping',494.59,'C:\DATA\DOCS98',1, 1,1,'3AuhETChCHqwAeAx0ohvhEqVsnWKGBUyoR2jnJrZ6mdgPh2Fw3xSM5rXr5CUyTrD8Iadi4MkZTfvtAtZMj1vIkZuKN1ZVEda6JIQmWgJk');     INSERT INTO "public"."auctions" ("id","item\_name","description","starting\_price","current\_price","condition","publication\_date","end\_date","payment\_type","shipping\_options","shipping\_cost","images\_folder","owner\_id","category\_id","city\_id","search") VALUES (2,'aWpjpRqelVcL7MdJ12vjtDPfFABtL0p0Vl6cBo8oaSCoJtArjCkbGhTUH2HbNfXkqeCmCgP0IonNPJwunDBxYAZ4o7kGxDVluqGO6fYGdtr84','This is a description 5384726',4337.40,35.21,'Used','11/07/2007 04:32:00','04/12/2012 05:08:00','PayPal','No shipping',4.63,NULL, 1, 2,2,'rCsPR7Vab6e1KyVPFV60im3UTcE3cMe8WfC0Lh7SGsVefcgSIDuI6i2wXKhMkcWoNEXLEHNWhSEUCLnKchNJvNrnEkkDIKmO1XH3G28Bo4CLPqEVs');     INSERT INTO "public"."auctions" ("id","item\_name","description","starting\_price","current\_price","condition","publication\_date","end\_date","payment\_type","shipping\_options","shipping\_cost","images\_folder","owner\_id","category\_id","city\_id","search") VALUES (3,'OuRRXvuAXCY8uokM8C2OsamXVJaj350Iji1XqCPhi177P3Mc3SxCCjXsBrC32FUAE0iXHAHIx25brM23dppbii','This is a description 1341487',2866.59,5221.08,'Used','03/19/2001 10:06:00','11/15/2005 10:34:00','PayPal','No shipping',269.93,'C:\APPDATA\DOC81', 2, 2,3,'ryKAl0611N2KczhccdmIug6CNtGRT06hjjy1eVEUZrdQvzJ7ZTGeZWr4hqNhhbQJGMRl5G7yUCeHDZyOwV2fGxV0mBg2vwxgoEw8nBpjLemw2XsgCnmqcZn8x20UQgLEWOKsUijKvpZHoOBIUxvEwtDFW566lEKcnzxho7i40GHYYv0nXQKB');     INSERT INTO "public"."auctions" ("id","item\_name","description","starting\_price","current\_price","condition","publication\_date","end\_date","payment\_type","shipping\_options","shipping\_cost","images\_folder","owner\_id","category\_id","city\_id","search") VALUES (4,'cC5rBCtWsr4X1N7lif2sQQZDnrn5Tg7tAmeDxz5yThUjpLaAigPStUuSdNOHYObt4wthMAwO6DdTkavhNeHAh54XSGBQ1P4h31kE6zXWpYFWo3aJf5XWmkxgcX7ZqLGmQ1uMEPUwTQlFsPFSS7DUiDW7eWIS1xpDXAa7a2GbjekuWJl','Description number 7219436',0.85,NULL,'Used','05/07/2002 08:47:00','08/20/2010 10:51:00','PayPal','No shipping',95704.75,'C:\DOCUMENTS\DOCS28', 3, 1,1,NULL);     INSERT INTO "public"."auctions" ("id","item\_name","description","starting\_price","current\_price","condition","publication\_date","end\_date","payment\_type","shipping\_options","shipping\_cost","images\_folder","owner\_id","category\_id","city\_id","search") VALUES (5,'MddSsCpo13VJp8HBxqlwSTwzO','Description number 8637276',6.37,6155.35,'Used','11/20/2003 01:06:00','06/04/2007 03:07:00','PayPal','No shipping',NULL,NULL, 2, 5,2,NULL);     INSERT INTO "public"."auctions" ("id","item\_name","description","starting\_price","current\_price","condition","publication\_date","end\_date","payment\_type","shipping\_options","shipping\_cost","images\_folder","owner\_id","category\_id","city\_id","search") VALUES (6,'Zo2lhsYyEMylLoYwhrtspCAkN5GSDNbAFZW5yylNX1FUnAJ54LVq8mIii5aKJqa6hfpt68FDkBfygZeaAE666lIMZ7XGQwjOQjzFTD8j1oeOywGOQbiibWJjOJF3ybiwo7hbh1It5dVByUMwFqXcQrCHqJRdfhn6cViVYKsMQjF5K8tnOjxjyQk4ynnOHm5edPpuWkY43RscZCySIWhksrmcmp7fKxB3MSW','This is a description 8815846',37502.85,5.71,'Used',NULL,'11/16/2006 10:07:00','PayPal','No shipping',0.15,'C:\TEMPORARY\DATA22', 4, 3,3,'Yz');        INSERT INTO "public"."bids" ("id","bidder\_id","bid\_amount") VALUES (1 ,6,763895);     INSERT INTO "public"."bids" ("id","bidder\_id","bid\_amount") VALUES (2,7,353202);     INSERT INTO "public"."bids" ("id","bidder\_id","bid\_amount") VALUES (3,8,842809);      INSERT INTO "public"."reviews" ("id","rating","description") VALUES (5,642350,'Description number 3028653');     INSERT INTO "public"."reviews" ("id","rating","description") VALUES (6,688953,'Description number 3261390');     INSERT INTO "public"."reviews" ("id","rating","description") VALUES (1,901247,'This is a description 7010386');        INSERT INTO "public"."won\_auctions" ("id","is\_successful\_transaction","has\_winner\_complained","winner\_id") VALUES (6,True,False,1);     INSERT INTO "public"."won\_auctions" ("id","is\_successful\_transaction","has\_winner\_complained","winner\_id") VALUES (5,True,True,2);       INSERT INTO "public"."qas" ("id","question","answer","auction\_id","questioner\_id") VALUES (1,'EnfGW8NyO6OH0yuHlv4ihFMFi0LymeDTZZwgd5KikuCrbzPafURMscr6t06pXYTtewJ0vZTvvqjZQIZSILsJbzlXjounzSoDlkfHYIwkiErzDdH0lEIY2ikuNCRB64jxoeuKGMm8qcjKUYz0fCFwsSuHi2RVQXTW3xE2jxakZTyhHWnLLfuMg2oBHAAuYhgQSZqqhwtUfBCGUZYZ82aRQziRjwv42jKMex6','ojBOa',707318,3);     INSERT INTO "public"."qas" ("id","question","answer","auction\_id","questioner\_id") VALUES (2,'bEfgYqkriLX0aypAYrkaWAPxgAaEcWM0Nfga5u5jPWj','J23duGx40axM7tpzQyDVwdgfwRpKHSqafibQKITo0Tjkr5yyNRCrWxRB2oCWVw7Wz',502838,2);     INSERT INTO "public"."qas" ("id","question","answer","auction\_id","questioner\_id") VALUES (3,'OfHrDhswL0GR3NVQC51VFRgUjZFiGQEqfcBizHyxeCSfKjl4YVxNs6KiEyCVkJrQvhU01xUHRoorGftxlN4n5iYNpfCvB0x6cuGJEmRvm17iyDZeXHJuXTd6JzbC56hKDT62ykDaJ1lZSyybDb','lCtzqQXkCD6VKnanOEvWN1CIgNoCLcGuwCugSUYbGaVvsa5qIPCqMITmgKrS4bLmkRqOAQXMxUz',584031,7);     INSERT INTO "public"."closed\_auctions" ("id") VALUES (4);         INSERT INTO "public"."wishlists" ("auction\_id","id") VALUES (1,2);     INSERT INTO "public"."wishlists" ("auction\_id","id") VALUES (2,3);     INSERT INTO "public"."wishlists" ("auction\_id","id") VALUES (3,1);     INSERT INTO "public"."wishlists" ("auction\_id","id") VALUES (5,1); |

**Revision history**  
   
Changes made to the first submission:

1. Added database.sql annex
2. Added more “Frequent Updates” queries
3. Added more indexes
4. Changed the title
5. Fixed multiple Clustering on same table
6. Changed database.sql
7. Changed populate.sql

GROUP1764, 11/04/2018  
   
Afonso Bernardino da Silva Pinto, up201503316@fe.up.pt  
Cristiana Maria Monteiro Ribeiro, up201305188@fe.up.pt  
Rostyslav Khoptiy, up201506219@fe.up.pt  
Tomás Sousa Oliveira, up201504746@fe.up.pt