DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Databases Project – Spring 2019

Team No: 22

Names: Gerald Sula, Georgios Fotiadis, Ridha Chahed

Contents

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

Deliverable 1

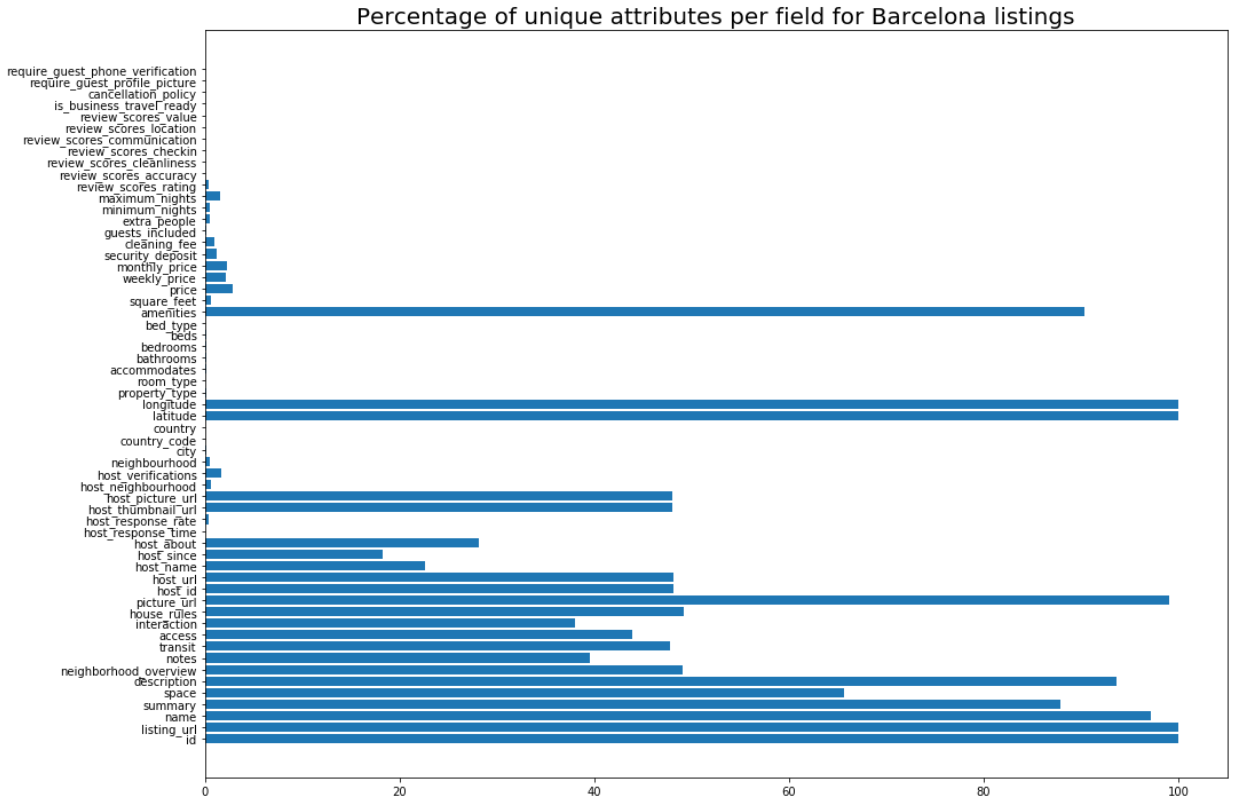
Assumptions

We are using as a central reference point the Listing entity (which has only 3 attributes unique to each listing) because almost all of the data could be linked to it in a very intuitive way.

All the entities have been selected judging on the principle that: if the attributes found in one instance of an entity are repeated in the data, those attributes must be included in one entity. We have made sure to "package" contextually similar attributes into one entity, as to not increase the complexity of the schema. For the attributes which can be different from one instance to the other, we have set them as relationship attributes (to avoid having a new entity all together, and also for not violating the aforementioned principle).

This is an overview of the percentage of unique attributes per field of the Barcelona listings dataset. Note that the plot is exactly the same for every city.

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

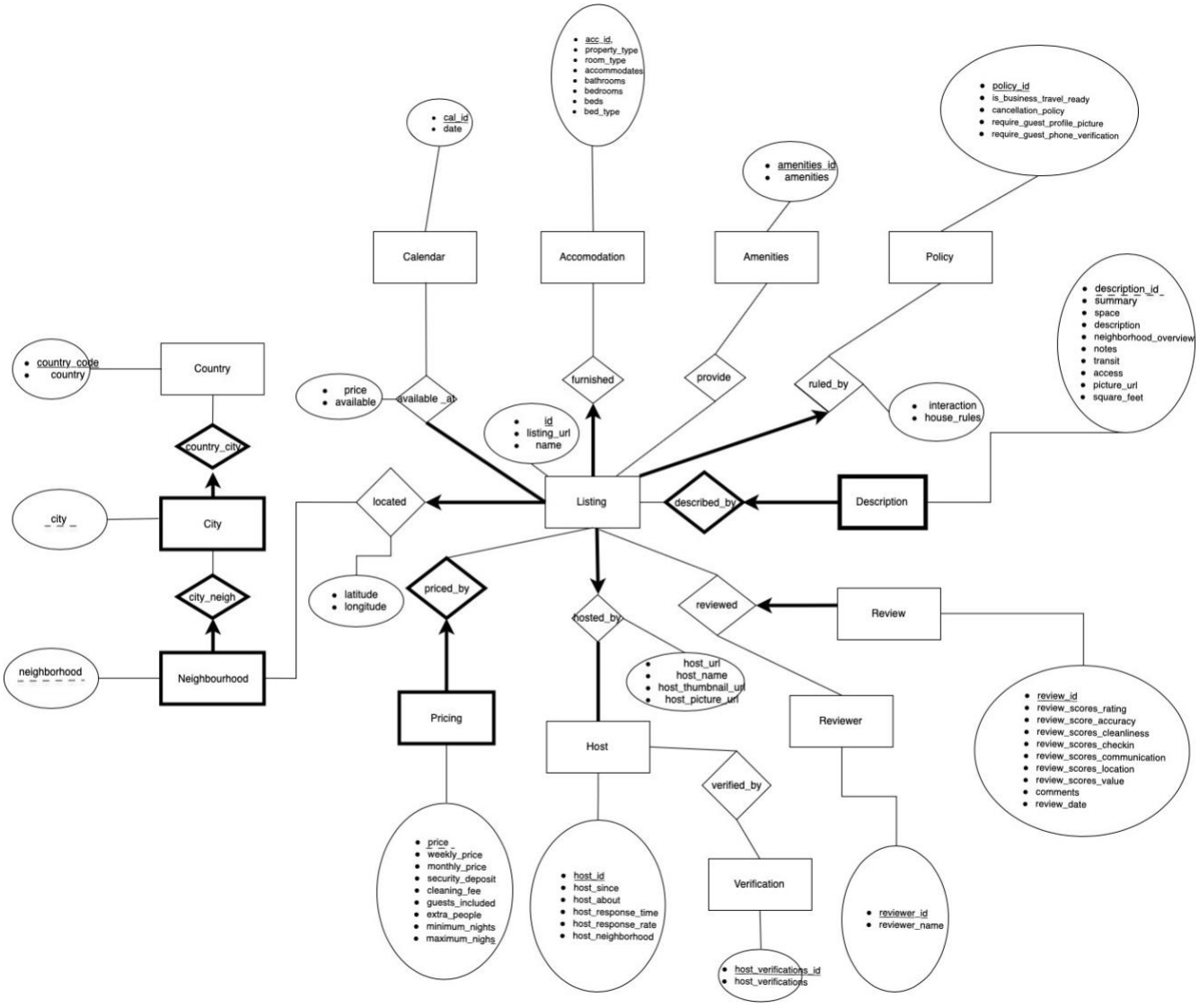


Entity Relationship Schema

Schema

Note: We included all the attributes of an entity in a big bubble as to conserve space in the diagram.

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/



Description

Calendar: there is a 'one to many' relation with listing because one listing will have at least one date of availability. The 'price' and 'available' attributes were set as relational attributes, because they were different for different listings (as they should be).

Accommodation: This entity describes the interior of the listing's apartment. All of these attributes will change from listing to listing so getting them together in one entity seems to be the right choice. Every listing will have exactly one set of those attributes, hence we used the 'exactly one' relation. It can happen that there are more than one listing that have the same set of attributes of this kind, that's why we opted out on having a weak relation here.

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

Amenities: The data contained here would make sense to be included in the accommodation entity, but since listings have more than exactly one amenity, we decided to create a separate simple entity.

Policy: Similar to the accommodation entity, we grouped together attributes which are unique to the listing (and also describe the same thing, in this case the rules of the listing) into one place. There is again an 'exactly one' relationship for the same reasons. One difference is that there are two relationship attributes here (interaction and house_rules). We decided to put them here because when looking at the data we saw that the policy was similar in most of the listings, with the exception of these two fields, as you can see in the plot above.

Description: Once again we grouped together attributes which describe the same aspect of the listing. We made this a weak entity, because a description should not exist on its own, it should always be paired with a listing.

Review and Reviewer: We decided to go with a ternary relation here between review-reviewer-listing. This is because they will always be used together: A review is given by a reviewer for a listing. The attributes we assigned to each entity are self-explanatory and there is an 'exactly one" relation for review because that has to be unique to the listing-reviewer. There are 'many to many' relations for both listing and reviewer because a listing can have many reviews from different reviewers, and a reviewer can leave many reviews for different listings.

Host: The host entity has all the attributes which describe a host. There is an "exactly one" relation from listing, because a listing should have exactly one host (we are not allowing multiple host to manage the same listing, same as Airbnb does in real life). The host has an 'at least' relation with listing because it makes no sense to be a host when you have no listing to manage.

Verification: We decided to split the verification into a new entity because the data of the field 'host_verification' was reused multiple times for different hosts. There is an 'exactly one' relation with hos since every host has only one verification. Pricing: For the exact same reasons as with description we decided do have a weak entity of similar data here.

Country, City, Neighborhood: Here we are using the logic that a Country can have multiple Cities, and each City can have multiple Neighborhoods. City and Neighborhood are weak entities to the ones above them because, for example, there can be no city which has no country. Finally, we 'connect' the most descriptive of the three (Neighborhood) with Listing with an 'exactly one' relation, because each listing is located at exactly one neighborhood. The attributes of country, city and neighborhood are reused for different instances of listings, the ones that change (latitude and longitude), we put inside the relationship, as to be able to keep the property of reusability.

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Relational Schema

ER schema to Relational schema, DDL:

```
CREATE TABLE Calendar(cal_id CHAR(32),
                date DATE,
                PRIMARY KEY (cal_id))

CREATE TABLE available_at(id CHAR(32),
                  cal_id CHAR(32),
                  price FLOAT,
                  available CHAR(1),
                  PRIMARY KEY (id,cal_id),
                  FOREIGN KEY (cal_id) REFERENCES Calendar(cal_id),
                  FOREIGN KEY (id) REFERENCES Listing(id)
                  /*can't ensure participation constraint*/)

CREATE TABLE Accomodation(acc_id CHAR(32),
                  property_type CHAR(32),
                  room_type CHAR(32),
                  accomodates CHA(32),
                  bathrooms INTEGER,
                  bedrooms INTEGER,
                  beds INTEGER,
                  bed_type CHAR(32,
                  PRIMARY KEY (acc_id) )

CREATE TABLE Amenities(amenities_id CHAR(32),
                amenities CHAR(32))

CREATE TABLE Provide(id CHAR(32),
                amenities_id CHAR(32),
                PRIMARY KEY (id,amenities_id),
                FOREIGN KEY (amenities_id) REFERENCES Amenities(amenities_id),
                FOREIGN KEY (id) REFERENCES Listing(id) )

CREATE TABLE Policy(policy_id CHAR(32),
                is_buisness_travel_ready CHAR(1),
```

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

```
        cancellation_policy CHAR(32),
        require_guest_profile_picture CHAR(1),
        require_guest_phone_verification CHAR(1)
        PRIMARY KEY (policy_id) )

CREATE TABLE Described_Description(description_id CHAR(32),
                summary CHAR(1024),
                space CHAR(1024),
                description CHAR(1024),
                neighborhood_overview CHAR(1024),
                notes CHAR(1024),
                transit CHAR(1024),
                access CHAR(1024),
                picture URL CHAR(32),
                square_feet FLOAT,
                id CHAR(32) NOT NULL
                PRIMARY KEY (description_id,id)
                FOREIGN KEY (id) REFERENCES Listing(id)
                ON DELETE CASCADE) )

CREATE TABLE Review(review_id CHAR (32),
        review_scores_rating INTEGER,
        review_score_accuracy INTEGER,
        review_scores_cleanliness INTEGER,
        review_scores_checkin INTEGER,
        review_scores_communication INTEGER,
        review_scores_location INTEGER,
        review_scores_value INTEGER,
        comments CHAR(1024),
        review_date DATE,
        id CHAR(32) NOT NULL,
        reviewer_id CHAR(32) NOT NULL,
        PRIMARY KEY (review_id),
        FOREIGN KEY (id) REFERENCES Listing(id),
        FOREIGN KEY (reviewer_id) REFERENCES Reviewer(reviewer_id) )

CREATE TABLE Reviewer(reviewer_id CHAR(32),
```

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

```
            reviewer_name CHAR(32)
            PRIMARY KEY (reviewer_id) )

CREATE TABLE Host(host_id CHAR(32),
        host_since DATE,
        host_about CHAR(1024),
        host_response_time FLOAT,
        host_response_rate FLOAT,
        host_neighborhood CHAR(32),
        PRIMARY KEY(host_id) )

CREATE TABLE Verification(host_verifications_id CHAR(32),
            host_verifications CHAR(32),
            PRIMARY KEY (host_verifications_id) )

CREATE TABLE Verified_by(host_id CHAR(32),
            host_verifications_id CHAR(32),
            PRIMARY KEY (host_id,host_verifications_id),
            FOREIGN KEY (host_id) REFERENCES Host(host_id),
            FOREIGN KEY (host_verifications_id) REFERENCES
            Verification(host_verifications_id) )

CREATE TABLE Pricing(price FLOAT,
        weekly_price FLOAT,
        monthly_price FLOAT,
        security_deposit FLOAT,
        cleaning_fee FLOAT,
        guests_included INTEGER,
        extra_people FLOAT,
        minimum_nights INTEGER,
        maximum_nighs INTEGER,
        id CHAR(32) NOT NULL,
        PRIMARY KEY(price,id),
        FOREIGN KEY (id) REFERENCES Listing(id) ON DELETE CASCADE )

CREATE TABLE Country(country_code CHAR(32),
        country CHAR(32),
```

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

```
            PRIMARY KEY (country_code) )

CREATE TABLE City(city CHAR(32),
            country_code CHAR(32),
            PRIMARY KEY (country_code,city),
            FOREIGN KEY (country_code) REFERENCES Country(country_code) )

CREATE TABLE Neighbourhood(neighborhood CHAR(32),
                  city CHAR(32),
                  country_code CHAR(32),
                  PRIMARY KEY (country_code,city,neighborhood),
                  FOREIGN KEY (country_code) REFERENCES Country(country_code),
                  FOREIGN KEY (city) REFERENCES City(city) )

CREATE TABLE Listing(id CHAR(32),
            lisitng_url CHAR(32),
            name CHAR(32),
            acc_id CHAR(32) NOT NULL,
            policy_id CHAR(32) NOT NULL,
            host_id CHAR(32) NOT NULL,
            host_url CHAR(32),
            host_name CHAR(32),
            host_thumbnail_url CHAR(32),
            host_picture_ur CHAR(32),
            neighborhood CHAR(32) NOT NULL
            PRIMARY KEY (id),
            FOREIGN KEY (acc_id) REFERENCES Accomodation(acc_id),
            FOREIGN KEY (policy_id) REFERENCES Policy(policy_id),
            FOREIGN KEY (host_id) REFERENCES Host(host_id),
            FOREIGN KEY (neighborhood) REFERENCES Neighbourhood(neighborhood) )
```

General Comments

We all worked on the design of the ER schema, for the rest we decided to split the workload like this:

- Ridha did the translation to the Relational Model
- George analysed the data to further optimize the ER schema for maximum data reusability and locality
- Gerald wrote the report explaining what each relation contains and justify our design choices

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Deliverable 2

Assumptions

<In this section write down the assumptions you made about the data. Write a sentence for each assumption you made>

Data Loading

Query Implementation

<For each query>

Query a:

Description of logic:

<What does the query do and how do I decide to solve it>

SQL statement

<The SQL statement>

Interface

Design logic Description

<Describe the general logic of your design as well as the technology you decided to use>

Screenshots

<Provide some initial screen shots of your interface>

General Comments

<In this section write general comments about your deliverable (comments and work allocation between team members>

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Deliverable 3

Assumptions

<In this section write down the assumptions you made about the data. Write a sentence for each assumption you made>

Query Implementation

<For each query>

Query a:

Description of logic:

<What does the query do and how do I decide to solve it>

SQL statement

<The SQL statement>

Query Analysis

Selected Queries (and why)

Query 1

<Initial Running time:
Optimized Running time:
Explain the improvement:
Initial plan
Improved plan>

Query 2

<Initial Running time:
Optimized Running time:
Explain the improvement:
Initial plan
Improved plan>

Query 3

<Initial Running time:
Optimized Running time:
Explain the improvement:
Initial plan
Improved plan>

DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

Interface

Design logic Description

<Describe the general logic of your design as well as the technology you decided to use>

Screenshots

<Provide some initial screen shots of your interface>

General Comments

<In this section write general comments about your deliverable (comments and work allocation between team members>