**PROJETO DE BASES DE DADOS**

**PARTE 3**

Grupo 20

Turno: 3ª-feira 8h30

Docente: Paulo Carreira

|  |  |  |
| --- | --- | --- |
|  | Horas | % de trabalho realizado |
| Joana Mendonça | 83597 | 18 | ~33% |
| Gonçalo Guerra | 83899 | 18 | ~33% |
| Filipe Colaço | 84715 | 18 | ~33% |

**Criação da Base de Dados (**schema.sql**)**

drop table local\_publico cascade;

drop table item cascade;

drop table anomalia cascade;

drop table anomalia\_traducao cascade;

drop table duplicado cascade;

drop table utilizador cascade;

drop table utilizador\_qualificado cascade;

drop table utilizador\_regular cascade;

drop table incidencia cascade;

drop table proposta\_de\_correcao cascade;

drop table correcao cascade;

create table local\_publico (

latitude numeric(8,6) not null,

longitude numeric(9,6) not null,

nome varchar(80) not null,

constraint pk\_local\_publico primary key (latitude, longitude)

);

create unique index coordenadas on local\_publico(latitude, longitude);

create table item (

item\_id int not null unique,

item\_descricao text not null,

localizacao text not null,

latitude numeric(8,6) not null,

longitude numeric(9,6) not null,

constraint pk\_item primary key (item\_id),

constraint fk\_item\_coordenadas foreign key (latitude, longitude)

references local\_publico(latitude, longitude) on delete cascade

);

create table anomalia (

anomalia\_id int not null unique,

zona box not null,

imagem text not null,

lingua varchar(25) not null,

ts timestamp not null,

anomalia\_descricao text not null,

tem\_anomalia\_redacao boolean not null,

constraint pk\_anomalia primary key (anomalia\_id)

);

create table anomalia\_traducao (

anomalia\_id int not null unique,

zona2 box not null,

lingua2 varchar(25) not null,

constraint pk\_anomalia\_traducao primary key (anomalia\_id),

constraint fk\_anomalia\_traducao\_id foreign key (anomalia\_id)

references anomalia(anomalia\_id) on delete cascade

);

create table duplicado (

item1\_id int not null unique,

item2\_id int not null unique,

constraint pk\_duplicado primary key (item1\_id, item2\_id),

constraint fk\_duplicado\_item1\_id foreign key (item1\_id)

references item(item\_id) on delete cascade,

constraint fk\_duplicado\_item2\_id foreign key (item2\_id)

references item(item\_id) on delete cascade,

constraint RI3 check (item1\_id < item2\_id)

);

create table utilizador (

email varchar(255) not null unique,

password varchar(255) not null,

type char(1) not null,

constraint pk\_utilizador primary key (email)

);

create unique index uuk on utilizador(email, type);

create table utilizador\_qualificado (

email varchar(255) not null,

type char(1) default 'Q',

constraint fk\_utilizador\_qualificado\_uuk foreign key (email, type)

references utilizador(email, type) on delete cascade,

constraint pk\_utilizador\_qualificado primary key (email)

);

create table utilizador\_regular (

email varchar(255) not null,

type char(1) default 'R',

constraint fk\_utilizador\_regular\_uuk foreign key (email, type)

references utilizador(email, type) on delete cascade,

constraint pk\_utilizador\_regular primary key (email)

);

create table incidencia (

anomalia\_id int not null unique,

item\_id int not null,

incidencia\_email varchar(255) not null,

constraint pk\_incidencia primary key (anomalia\_id),

constraint fk\_incidencia\_anomalia\_id foreign key (anomalia\_id)

references anomalia(anomalia\_id) on delete cascade,

constraint fk\_incidencia\_item\_id foreign key (item\_id)

references item(item\_id) on delete cascade,

constraint fk\_incidencia\_email foreign key (incidencia\_email)

references utilizador(email) on delete cascade

);

create table proposta\_de\_correcao (

nro int not null unique,

pdc\_email varchar(255) not null,

data\_hora timestamp not null,

texto text not null,

constraint pk\_proposta\_de\_correcao primary key (nro, pdc\_email),

constraint fk\_proposta\_de\_correcao foreign key (pdc\_email)

references utilizador\_qualificado(email) on delete cascade

);

create unique index pdcuk on proposta\_de\_correcao(nro, pdc\_email);

create table correcao (

nro int not null,

pdc\_email varchar(255) not null,

anomalia\_id int not null,

constraint pk\_correcao primary key (pdc\_email, nro, anomalia\_id),

constraint fk\_correcao\_pdcuk foreign key (nro, pdc\_email)

references proposta\_de\_correcao(nro, pdc\_email) on delete cascade,

constraint fk\_correcao\_anomalia\_id foreign key (anomalia\_id)

references incidencia(anomalia\_id) on delete cascade

);

***Queries* SQL (**queries.sql**)**

1. Qual o local público onde estão registadas mais anomalias?

with num\_anomalias\_utilizador as (

select nome, count(anomalia\_id)

from

incidencia natural join (

select item\_id, latitude, longitude

from item

) as item2 natural join

local\_publico

group by nome

),

maximum\_nau as (

select max(count) as count

from num\_anomalias\_utilizador

)

select nome, count

from num\_anomalias\_utilizador natural join maximum\_nau;

2. Qual o utilizador regular que registou mais anomalias de tradução no 1º semestre de 2019?

with num\_anomalias\_utilizador as (

select incidencia\_email, count(incidencia\_email)

from (

select anomalia\_id, incidencia\_email, type, ts

from

anomalia natural join

incidencia natural join

utilizador

where

tem\_anomalia\_redacao = FALSE and

type = 'R' and

ts between '2019-01-01 00:00:00' and '2019-06-30 23:59:59'

) as an2

group by incidencia\_email),

maximum\_nau as (

select max(count) as count

from num\_anomalias\_utilizador

)

select incidencia\_email, count

from num\_anomalias\_utilizador natural join maximum\_nau;

3. Quais são os utilizadores que registaram em 2019 incidências em todos os locais públicos

situados a norte de Rio Maior (Portugal)?

select incidencia\_email

from

anomalia natural join

incidencia natural join

item

where

ts between '2019-01-01 00:00:00' and '2019-12-31 23:59:59' and

latitude > 39.336775

group by incidencia\_email;

4. Quais são os utilizadores qualificados que não apresentaram uma proposta de correção para

cada uma das incidências por eles registadas em locais públicos a sul de Rio Maior no ano

corrente?

select incidencia\_email

from

incidencia natural join

correcao natural join

proposta\_de\_correcao natural join (

select anomalia\_id, ts

from anomalia

where extract(year from ts) = extract(year from now())

) as a1 natural join (

select item\_id, latitude

from item

where latitude < 39.336775

) as a2

where pdc\_email != incidencia\_email

group by incidencia\_email;**Arquitetura da Aplicação**

A aplicação desenvolvida corre sobre a base de dados PostgreSQL do IST do aluno Filipe Colaço (ist426015), onde estão carregadas as tabelas correspondentes ao sistema Translate Right.

Cada página .php contém uma tabela correspondente à funcionalidade pedida;

* anomalia.php, local\_publico.php e item.php permitem inserir e remover Anomalias, Locais Públicos e Itens, respetivamente
* proposta\_de\_correcao.php permite inserir, editar e remover Propostas de Correção
* utilizadores.php lista os Utilizadores registados no sistema
* anomalias\_entre\_locais.php permite ao Utilizador ver quais as Anomalias das Incidências registada na área compreendida entre dois Locais Públicos

Cada ficheiro contém uma tabela de resultados gerada a partir de uma *query* passada ao servidor de base de dados, e, internamente, trata de cada uma das operações a si designadas. Estas operações – eliminar, inserir, editar, selecionar – é tratada por um formulário HTML dentro do ficheiro, que submete os seus valores de volta para a própria página, que irá refrescar e tratar dos dados enviados pelo formulário.