

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, seleccionar – 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.