

# Projeto de Bases de Dados: Parte 3

Nome	Número	Contribuição	Horas de trabalho
Daniel Castro	ist187644	33,33%	12h
Daniela Mendes	ist187646	33,33%	12h
Miguel Oliveira	ist187689	33,33%	12h

**Grupo Nº1**

Turno: BD817957L08

Professor: André Vasconcelos

### **Comandos de criação da base de dados**

**create table camara (**

numCamara int not null unique,  
constraint pk\_numCamara primary key(numCamara));

**create table video (**

dataHoraInicio timestamp not null,  
dataHoraFim timestamp not null,  
numCamara int not null,  
constraint pk\_video primary key(dataHoraInicio, numCamara),  
constraint fk\_video\_camara foreign key(numCamara) references camara(numCamara));

**create table segmentoVideo (**

numSegmento int not null,  
duracao interval not null,  
dataHoraInicio timestamp not null,  
numCamara int not null,  
constraint pk\_segmentoVideo primary key(numSegmento, dataHoraInicio, numCamara),  
constraint fk\_segmentoVideo\_video foreign key(dataHoraInicio, numCamara) references video(dataHoraInicio, numCamara));

**create table localidade (**

moradaLocal varchar(80) not null,  
constraint pk\_localidade primary key(moradaLocal));

**create table vigia (**

moradaLocal varchar(80) not null,  
numCamara int not null,  
constraint pk\_vigia primary key(moradaLocal, numCamara),  
constraint fk\_vigia\_localidade foreign key(moradaLocal) references localidade(moradaLocal) on delete cascade,  
constraint fk\_vigia\_camara foreign key(numCamara) references camara(numCamara) on delete cascade);

**create table processoSocorro (**

numProcessoSocorro int not null unique,  
constraint pk\_processoSocorro primary key(numProcessoSocorro));

**create table entidadeMeio (**

nomeEntidade varchar(80) not null unique,  
constraint pk\_entidadeMeio primary key(nomeEntidade));

**create table eventoEmergencia (**

numTelefone int not null,  
instanteChamada timestamp not null,  
nomePessoa varchar(80) not null,  
moradaLocal varchar(80) not null,  
numProcessoSocorro int,  
constraint pk\_eventoEmergencia primary key(numTelefone, instanteChamada),

```
constraint fk_eventoEmergencia_localidade foreign key(moradaLocal) references
localidade(moradaLocal) on delete cascade,
constraint fk_eventoEmergencia_processoSocorro foreign key(numProcessoSocorro) references
processoSocorro(numProcessoSocorro) on delete cascade,
UNIQUE(numTelefone, nomePessoa));
```

```
create table meio (
    numMeio int not null,
    nomeMeio varchar(80) not null,
    nomeEntidade varchar(80) not null,
    constraint pk_meio primary key(numMeio, nomeEntidade),
    constraint fk_meio_entidadeMeio foreign key(nomeEntidade) references entidadeMeio(nomeEntidade)
on delete cascade);
```

```
create table meioCombate (
    numMeio int not null,
    nomeEntidade varchar(80) not null,
    constraint pk_meioCombate primary key(numMeio, nomeEntidade),
    constraint fk_meioCombate_meio foreign key(numMeio, nomeEntidade) references meio(numMeio,
nomeEntidade) on delete cascade);
```

```
create table meioApoio (
    numMeio int not null,
    nomeEntidade varchar(80) not null,
    constraint pk_meioApoio primary key(numMeio, nomeEntidade),
    constraint fk_meioApoio_meio foreign key(numMeio, nomeEntidade) references meio(numMeio,
nomeEntidade) on delete cascade);
```

```
create table meioSocorro (
    numMeio int not null,
    nomeEntidade varchar(80) not null,
    constraint pk_meioSocorro primary key(numMeio, nomeEntidade),
    constraint fk_meioSocorro_meio foreign key(numMeio, nomeEntidade) references meio(numMeio,
nomeEntidade) on delete cascade);
```

```
create table transporta (
    numMeio int not null,
    nomeEntidade varchar(80) not null,
    numVitimas int not null default 0,
    numProcessoSocorro int not null,
    constraint pk_transporta primary key(numMeio, nomeEntidade, numProcessoSocorro),
    constraint fk_transporta_meioSocorro foreign key(numMeio, nomeEntidade) references
meioSocorro(numMeio, nomeEntidade) on delete cascade,
constraint fk_eventoEmergencia_processoSocorro foreign key(numProcessoSocorro) references
processoSocorro(numProcessoSocorro) on delete cascade);
```

```
create table alugado (
    numMeio int not null,
    nomeEntidade varchar(80) not null,
    numHoras int not null,
    numProcessoSocorro int not null,
```

```
constraint pk_alocado primary key(numMeio, nomeEntidade, numProcessoSocorro),
constraint fk_alocado_meioApoio foreign key(numMeio, nomeEntidade) references
meioApoio(numMeio, nomeEntidade) on delete cascade,
constraint fk_alocado_processoSocorro foreign key(numProcessoSocorro) references
processoSocorro(numProcessoSocorro) on delete cascade);
```

```
create table acciona (
    numMeio int not null,
    nomeEntidade varchar(80) not null,
    numProcessoSocorro int not null,
    constraint pk_acciona primary key(numMeio, nomeEntidade, numProcessoSocorro),
    constraint fk_acciona_meio foreign key(numMeio, nomeEntidade) references meio(numMeio,
nomeEntidade) on delete cascade,
    constraint fk_acciona_processoSocorro foreign key(numProcessoSocorro) references
processoSocorro(numProcessoSocorro) on delete cascade);
```

```
create table coordenador (
    idCoordenador int not null unique,
    constraint pk_coordenador primary key(idCoordenador));
```

```
create table audita (
    idCoordenador int not null,
    numMeio int not null,
    nomeEntidade varchar(80) not null,
    numProcessoSocorro int not null,
    datahoraInicio timestamp not null,
    datahoraFim timestamp not null,
    dataAuditoria timestamp not null,
    texto text,
    constraint pk_audita primary key(idCoordenador, numMeio, nomeEntidade, numProcessoSocorro),
    constraint fk_audita_acciona foreign key(numMeio, nomeEntidade, numProcessoSocorro) references
acciona(numMeio, nomeEntidade, numProcessoSocorro) on delete cascade,
    constraint fk_audita_coordenador foreign key(idCoordenador) references coordenador(idCoordenador)
on delete cascade,
    check (datahoraInicio < datahoraFim),
    check (datahoraFim < dataAuditoria));
```

```
create table solicita (
    idCoordenador int not null,
    dataHoraInicioVideo timestamp not null,
    numCamara int not null,
    dataHoraInicio timestamp not null,
    dataHoraFim timestamp not null,
    constraint pk_solicita primary key(idCoordenador, dataHoraInicioVideo, numCamara),
    constraint fk_solicita_coordenador foreign key(idCoordenador) references coordenador(idCoordenador)
on delete cascade,
    constraint fk_solicita_video foreign key(dataHoraInicioVideo, numCamara) references
video(dataHoraInicio, numCamara) on delete cascade,
    check (datahoraInicio < datahoraFim));
```

## Consultas em SQL

### **1. Processo de socorro que envolveu maior número de meios distintos**

```
select distinct numProcessoSocorro
from acciona
group by numProcessoSocorro
having count(numProcessoSocorro) >= all (
    select count(numProcessoSocorro)
    from acciona
    group by numProcessoSocorro);
```

### **2. Entidade fornecedora de meios que participou em mais processos de socorro no Verão de 2018**

```
select nomeEntidade
from (
    select distinct nomeEntidade, numProcessoSocorro
    from acciona natural join eventoEmergencia
    where instanteChamada >= '2018-06-21 00:00:00'
    and instanteChamada <= '2018-09-21 23:59:59') as R1
group by nomeEntidade
having count(nomeEntidade) >= all (
    select count(nomeEntidade)
    from (
        select distinct nomeEntidade, numProcessoSocorro
        from acciona natural join eventoEmergencia
        where instanteChamada >= '2018-06-21 00:00:00'
        and instanteChamada <= '2018-09-21 23:59:59') as R2
    group by nomeEntidade);
```

### **3. Processos de socorro, referentes a eventos de emergencia em 2018 de Oliveira do Hospital, onde existe pelo menos um acionamento de meios que nao foi alvo de auditoria**

```
select distinct R1.numProcessoSocorro
from (
    select numProcessoSocorro, numMeio, nomeEntidade
    from eventoEmergencia natural join acciona
    where moradaLocal = 'Oliveira do Hospital'
    and instanteChamada >= '2018-01-01 00:00:00'
    and instanteChamada <= '2018-12-31 23:59:59'
) as R1 left join (
    select numProcessoSocorro, numMeio, nomeEntidade
    from audita) as R2
on R1.numProcessoSocorro = R2.numProcessoSocorro
and R1.numMeio = R2.numMeio
and R1.nomeEntidade = R2.nomeEntidade
where R2.numProcessoSocorro is null
and R2.numMeio is null
and R2.nomeEntidade is null;
```

**4. Número de segmentos de video com duração superior a 60 segundos, que foram gravados em camaras de vigilancia de Monchique durante o mes de Agosto de 2018**

```
select count(numSegmento)
from segmentoVideo natural join vigia
where duracao > '60sec'
    and moradaLocal = 'Monchique'
    and dataHoraInicio >= '2018-08-01 00:00:00'
    and dataHoraInicio <= '2018-08-31 23:59:59';
```

**5. Meios de combate que nao foram usados como Meios de Apoio em nenhum processo de socorro**

```
select numMeio, nomeEntidade
from meioCombate C
Where not exists (
    select numMeio, nomeEntidade
    from alocado A
    where C.numMeio = A.numMeio
    and C.nomeEntidade = A.nomeEntidade);
```

**6. Entidades que forneceram meios de combate a todos os Processos de socorro que acionaram meios**

```
select nomeEntidade
from meioCombate natural join acciona
group by nomeEntidade
having count(distinct numProcessoSocorro) = (
    select count(distinct numProcessoSocorro)
    from acciona);
```

## Explicação da arquitetura da aplicação PHP e das relações entre os diversos ficheiros

