



ISEL
INSTITUTO SUPERIOR DE
ENGENHARIA DE LISBOA

2015/16 INV

**Departamento de Engenharia de Electrónica e
Telecomunicações e
de Computadores
Licenciatura em Engenharia Informática e de Computadores**

Sistemas de Informação I

Trabalho prático

(Fases 1, 2 e 3)

Grupo 12

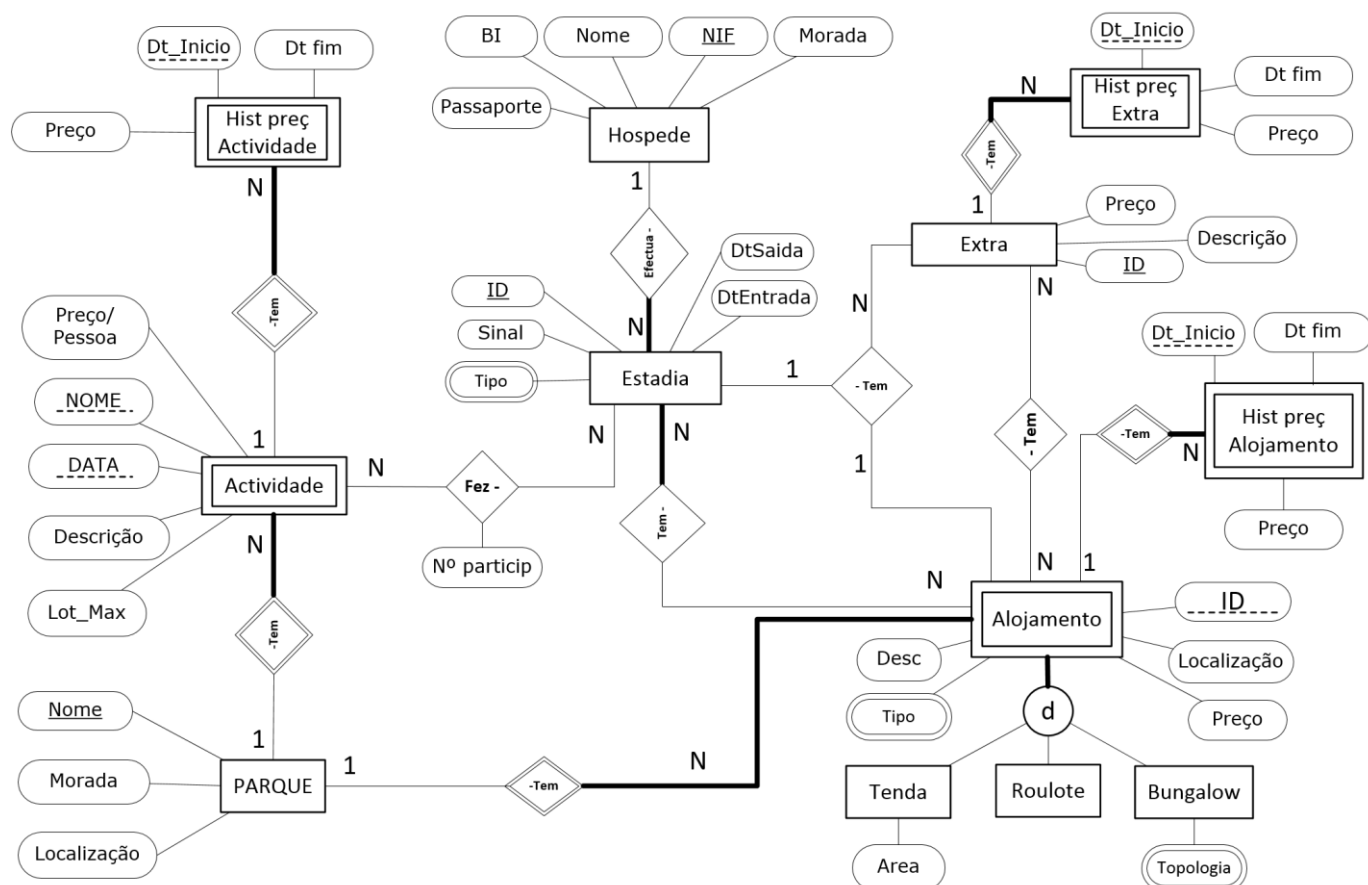
Ricardo Martins A21600

Hugo Reis A38652

Daniel Miranda A40911



Modelo Entidade Associação (MEA)





ISEL
INSTITUTO SUPERIOR DE
ENGENHARIA DE LISBOA

**Departamento de Engenharia de Electrónica e
Telecomunicações e
de Computadores
Licenciatura em Engenharia Informática e de Computadores**

Restrições de Integridade:

RI1: PARQUE: As coordenadas devem ser guardadas numa única cadeia de caracteres, especificando a latitude e longitude em formato decimal, e.g. "38.713889, -9.139444"

RI2: Alojamento: Localização é única em Parque.

RI3: Alojamento: Topologia pode tomar os valores: T0,T1,T2 e T3

RI4: Os preços apenas podem ser alterados 1 vez por dia

RI5: A alteração de um preço implica a alteração da data de validade do registo em Hist_Preço e criação de novo registo com Data de início do dia de alteração

RI6: Alojamento: Tipo só pode tomar os valores: T,R e B

RI7: Estadia: Tipo só pode tomar os valores: "Res", "Est", "Res+Est"



Modelo Relacional (MR)

Parque (Nome, morada, localização)

Alojamento (Parque, ID, localização, desc, preço, tipo,)

CE: {Parque Ref. Parque(Nome)}

OCC : { Parque, localização }

RI: Tipo só pode tomar os valores: T,R e B

RI: Sempre que seja alterado o valor é necessário preencher a tabela
Hist_preços_Alojamento

Tenda (Parque, Alojamento, area)

CE: { (Parque,Alojamento) Ref. Alojamento(Parque,ID) }

Bungalow (Parque, Alojamento, topologia)

CE: { (Parque,Alojamento) Ref. Alojamento(Parque,ID) }

RI: Topologia só pode ter os valores (T0,...,T3)

Roulote (Parque, Alojamento)

CE: { (Parque,Alojamento) Ref. Alojamento(Parque,ID) }

Extra (ID, desc, preço)

RI: Sempre que seja alterado o valor é necessário preencher a tabela
Hist_preços_Extra

Alojamento_Extras (Parque, Alojamento, Extra)

CE: { { Parque,Alojamento } Ref. Alojamento (Parque, ID) },
{ Extra } Ref. { Extra (ID) } }

Hospede (BI, Passaporte, nome, morada, NIF)

Estadia (ID, Tipo, Sinal, DT_Entrada, DT_Saida, Hospede, Parque, Alojamento)

CE: { Hospede Ref. Hospede(BI) }

RI: Tipo só pode ter os valores (Res, Est, Est+Res) e NOT NULL

RI: Só pode ser associado um Alojamento à reserva se não tiver nenhuma
reserva/marcação para as datas pretendidas

Estadia_Alojamentos (Estadia, Aloj_ID, Aloj_Parque)

CE: { { Estadia Ref. Estadia(ID)},

{ (Aloj_Parque, Aloj_ID) Ref. Alojamento(Parque, ID) } }

RI: Para 1 estadia os Alojamentos tem de pertencer ao mesmo Parque

RI: Estadia tem participação obrigatória em **Estadia_Alojamentos**



ISEL
INSTITUTO SUPERIOR DE
ENGENHARIA DE LISBOA

**Departamento de Engenharia de Electrónica e
Telecomunicações e
de Computadores
Licenciatura em Engenharia Informática e de Computadores**

Estadia_Extras_Por_Alojamento (Extra, Estadia, Aloj_ID, Aloj_Parque)

CE: { { Estadia Ref. Estadia(ID)},

{ (Aloj_Parque, Aloja_ID) Ref. Alojamento(Parque, ID) } ,

{ Extra Ref. Extra(ID) } }

CC: { Extra, Aloj_ID, Aloj_Parque }

RI: Para 1 estadia os Alojamentos e extras tem de pertencer ao mesmo Parque

RI: Os alojamentos tem de pertencer à estadia

Actividade (Parque, nome, data, descrição, lot_Max, preçoPessoa)

CE: { Parque Ref. Parque(Nome) }

RI: Sempre que seja alterado o valor é necessário preencher a tabela Hist_preços_Actividade

Estadia_Actividades (Estadia, Parque, Nome, Data, num_participantes)

CE: { { (Parque,Nome,Data) Ref. Actividade(Parque,Nome,Data)},

{ Estadia Ref. Estadia(ID) } }

Hist_preços_Actividade (Parque, Nome, Data, DT_Inicio, DT_fim, PrecoPessoa)

CE: { Parque, Nome, Data Ref. **Actividade** (Parque, Nome, Data)},

RI: Dt Fim é igual à data de Inicio do proximo registo ou null se não houver mais registos

Hist_preços_Extra(ExtraID, DT_Inicio, DT_Fim, Preço)

CE: { ExtraID Ref. Extra (ID) }

RI: Dt Fim é igual à data de Inicio do proximo registo ou null se não houver mais registos

Hist_preços_Alojamento(Parque, Alojamento, DT_Inicio, DT_Fim, Preco)

CE: { { Parque, Alojamento } Ref. {Tenda, Bungalow, Roulote (Parque, Alojamento) } }

RI: Dt Fim é igual à data de Inicio do proximo registo ou null se não houver mais registos



ISEL
INSTITUTO SUPERIOR DE
ENGENHARIA DE LISBOA

**Departamento de Engenharia de Electrónica e
Telecomunicações e
de Computadores
Licenciatura em Engenharia Informática e de Computadores**

SQL :

6a.

```
select nome, DT_entrada, tipo, Localização

from hospede as H
    inner join Estadia as E on (H.nif = E.hospede)
    inner join Estadia_Alojamento EA on (E.alojamento = EA.estadia)
    inner join Alojamento as A on (A.id = EA.aloj_id)
where nif = 123456789
```

6b

```
select top 1 nif, Nome from Hospede

inner join Estadia on Estadia.Hospede = Hospede.NIF

inner join Estadia_Alojamentos on Estadia_Alojamentos.Estadia = Estadia.ID

where Estadia_Alojamentos.Aloj_Parque = 'mirasol'

group by nif, nome

order by count(nif)
```

6c

```
select * from Hospede

inner join Estadia on Hospede.nif = Estadia.Hospede
where year(DT_entrada) = 2015 and
estadia.tipo = 'est'
or year(DT_entrada) = 2015 and
estadia.tipo = 'est+res'

and not exists ( select Hospede.nif
where Hospede.nif = Estadia.Hospede

and year( DT_entrada) > 2015 )
```



ISEL
INSTITUTO SUPERIOR DE
ENGENHARIA DE LISBOA

**Departamento de Engenharia de Electrónica e
Telecomunicações e
de Computadores
Licenciatura em Engenharia Informática e de Computadores**

6d

```
select nif
  from hospede as H
 where H.nif in(select Hospede
                from Estadia as E
                inner join ( select *
                           from Estadia_Extras_Por_Alojamento
                           group by Aloj_Parque, Aloj_ID
                           ) as A on (E.ID = A.Estadia)
```

6e

```
select distinct count(*) as C
  from Actividades as A
 group by Parque, Nome, Data

 having C = A.Lot_Max
```