

Grupo: Flávio, Gleice Kelly, João Coelho, José Rodolfo, Lucas Arruda

Feature: Costs

Custos

Model Custos{

- id: String
- valor: double
- dataInicio: string
- dataFim string
- descricao: string
- deletedBy: string
- deletedDate: string

}

@get

/api/custos/

Retorna lista de custos cadastrados.

Parâmetros:

descricao (query) - filtrar custos baseado na descrição

qtdItens (query) - quantidade de itens exibidos por página

pagina (query) - página exibida

/api/custos/{id}

Retorna o custo relacionado com o id.

Parâmetros:

id: **string** | id do custo a ser pesquisado

/api/custos/{dataInicio}/{dataFim}

Retorna todos os custos presentes entre duas datas.

Parâmetros:

dataInicio: string | data de início da pesquisa no formato YYYY-MM-DD

dataFim: string | data final da pesquisa no formato YYYY-MM-DD

/api/custos/projeto/{id}

Retorna lista de custos de determinado projeto

Parâmetros:

idProjeto: string | id do projeto solicitado

/api/custos/cargo/{id}

Retorna todos os custos referentes ao cargo especificado.

Parâmetros:

id: string | id do cargo

@post

api/costs/

Cria o custo, com seus respectivos parâmetros.

Parâmetros:

id: string | id do custo

valor: double | valor do custo

descricao: string | descricao do custo

json entrada

```
{
  "id": "id",
  "valor": 100,
  "dataInicio": "2005-05-24",
  "dataFim": "2006-07-30",
  "descricao": "desc",
}
```

json saída

```
{
  "id": "id",
  "valor": 100,
  "dataInicio": "2005-05-24",
  "dataFim": "2006-07-30",
  "descricao": "desc",
}
```

@put

/api/costs/{id}

Edita informações sobre o custo

Parâmetros:

id: string | id do custo

valor: double | valor do custo

descricao: string | descricao do custo

json entrada

```
{
  "id": "id",
  "descricao": "newDesc",
  "valor": "100"
}
```

json saída

```
{
  "id": "id"
}
```

@delete

/api/costs/{id}

Marca a flag de deleção da tupla e preenche os campos correspondentes.

Parâmetros:

id: **string** | id do custo a ser pesquisado e removido

Feature: Skills and Jobs

Cargos

Model Cargos {

- id: String
- descricao: String
- nome: String
- funcao: String

}

@get

/api/cargo/

Retorna uma lista de cargos cadastrados.

Parâmetros:

descricao (query) - filtrar cargos baseado na descrição

nome (query) - filtrar cargos baseado no nome

funcao (query) - filtrar cargos baseado na função

qtditens(query) - quantidade de itens exibidos por página

pagina(query) - página exibida

/api/cargo/{id}

Retorna um cargo em específico pelo id

Parâmetros:

id: **string** | Id do cargo a ser procurado

/api/cargo/{nome}

Retorna um cargo em específico pelo nome

Parâmetros:

nome: **String** | nome do cargo a ser procurado

/api/cargo/squad/{id}

Retorna os cargos presentes em uma equipe

Parâmetros:

squad: **String** | equipe a qual será a base para retornar a lista de cargos

/api/cargo/skill/{id}

Retorna uma lista de cargos que possuem a skill

Parâmetros:

id: **string** | Id da skill desejada

api/cargo

Retorna todos os cargos cadastrados.

Parâmetros:

Nenhum

/api/cargo/colaborador/{id}

Retorna a lista de cargos que possuem colaboradores

Parâmetros:

id: **string** | id do cargo

@post**api/cargo/create**

Cria um novo cargo

Parâmetros:

id: **string** | id do cargo

nome: **string** | nome do cargo

funcao: **string** | funcao do cargo

json entrada

```
{
  "nome": "name",
  "funcao": "teste"
}
```

json saída

```
{
  "id": "id",
  "nome": "name",
  "funcao": "teste"
}
```

@put**/api/cargo/edit/**

Edita parâmetros de algum cargo existente

Parâmetros:

id: **string** | id do cargo que terá os dados editados

nome: **string** | nome do cargo

funcao: **string** | funcao do cargo

json entrada

```
{
  "id": "id",
  "nome": "newName",
  "funcao": "newTeste"
}
```

json saída

```
{
  "id": "id",
  "nome": "newName",
  "funcao": "newTeste"
}
```

@delete

/api/cargo/delete/{id}

Remove um cargo cadastrado

Parâmetros:

id: string | Id do cargo a ser removido

Skills

Model Skills{

- id: String
- descricao: String

}

@get

/api/skills/

Retorna lista de skills cadastradas.

Parâmetros:

descricao(query) - filtrar skills baseado na descrição

qtditens(query) - quantidade de itens exibidos por página

pagina(query) - página exibida

/api/skills/{id}

Retorna a skill relacionado com o id.

Parâmetros:

id: string | Id da skill a ser procurada

/api/skills/colaborador/{id}

Retorna a lista de skills que o colaborador possui

Parâmetros:

id: string | id do colaborador

@post

/api/skill/createskill/

Cria uma nova skill e adiciona à lista de skills existentes

Parâmetros:

descricao: string | descricao da skill

```
json entrada
{
    "descricao": "desc"
}
json saida
{
    "id": "id",
    "descricao": "desc"
}
```

@put

/api/skill/edit/

Edita uma skill já existente

Parâmetros:

id: string | id da skill que terá os dados editados

descricao: string | descrição da skill que terá os dados editados

@delete

api/skill/{id}

Deleta uma skill cadastrada

Parâmetros:

id: string | Id da skill que será deletada

Colaborador

Model Colaborador {

- id: String
- cpf: String
- nome: String
- matricula: String
- telefone: String
- emailPessoal: String
- emailInstitucional: String
- lattes: String

}

@get

/api/colaborador/

Retorna todos os colaboradores cadastrados

Parâmetros:

email (query) - filtrar colaborador baseado no email

nome (query) - filtrar colaborador baseado no nome

qtditens(query) - quantidade de itens exibidos por página
pagina(query) - página exibida

/api/colaborador/{id}/

Retorna o colaborador relacionado com o id.

Parâmetros:

id: string | Id do colaborador a ser procurado

/api/colaborador/skills/{id}

Retorna a lista de colaboradores que possuem determinada skill

Parâmetros:

id: string | id da skill

/api/colaborador/cargo/{id}

Retorna lista de colaboradores referentes ao cargo especificado.

Parâmetros:

id: string | id do cargo

@post

/api/colaborador/

Cria um novo colaborador

Parâmetros:

name: string | nome do colaborador

matricula: string | matrícula do colaborador

cpf: string | cpf do colaborador

id: string | id do colaborador

json entrada

```
{
  "name": "newName",
  "matricula": "newTeste",
  "cpf": "555555555"
}
```

}

json saída

```
{
  "id": "id",
  "name": "newName",
  "matricula": "newTeste",
  "cpf": "555555555"
}
```

}

@put

/api/colaborador/{id}

Edita parâmetros de um colaborador já existente

Parâmetros:

id:string | id do colaborador que terá os dados editados

name: string | nome do colaborador que terá os dados editados

matricula: string | matricula do colaborador que terá os dados editados

cpf: string | cpf do colaborador que terá os dados editados

json entrada

```
{
  "id": "id",
  "name": "newName",
  "matricula": "newTeste",
  "cpf": "555555555"
}
```

json saída

```
{
  "id": "id",
  "name": "newName",
  "matricula": "newTeste",
  "cpf": "555555555"
}
```

@delete

/api/colaborador/{id}

Remove um colaborador da lista de colaboradores cadastrados

Parâmetros:

id: string | id do colaborador que será deletada