



I DO

Aplicativo de serviço

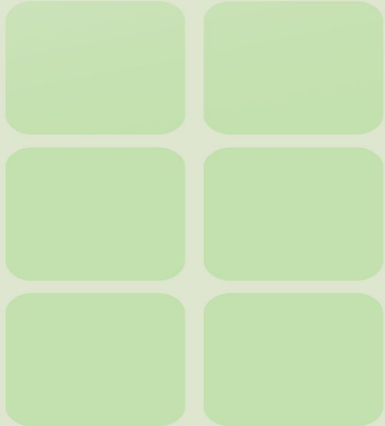
Propostas do i do

- Aplicativo onde pode se encontrado diversos serviços, oferecido pelos usuário;
- Encontra serviços de qualidade e avaliações boa;
- Contato direto com o fornecedor de serviço, sem máscara de preço;

O trabalho

Durante o semestre tivemos que passar a ideia de um banco de dados do nosso aplicativo, onde foi feito diversas coisa como:

- Criação de modelo conceitual, físico e lógico do nosso banco;
- Inserção, Update, deletes de dados
- Exclusão e restauração do banco;



Tabelão Inicial não normalizado

nome	e-mail	celular	telefone	cpf	rg	user	senha	rua	cidade
Tadeu Junior	tadeu@gmail.com	998765432	33480939	1111111	223344	111111		Rua Café Filho	Cariacica
Yan de Paula	yan@gmail.com	987654321	22228876	2222222	887766	22222		Rua Diamante	Vila Velha
Ewerson Vieira	ewerson@gmail.com	912345678	33398726	3333333	323456	33333		Rua Pedra Bonita,	Serra
Lucas Gomes Irinel	luca-irinel@gmail.com	903625347	33448765	4444444	265260	44444		Rua da Paz	Guarapari
Icaro Duarte	icaro@gmail.com	999876543	33876529	5555555	223344	55555		Rua da Guerra	Alegre
David Vilaca	david@gmail.com	988765432	34567839	6666666	445566	66666		Rua dos Aprovados	Domingos Martins
Leandro Goias	leandro@gmail.com	977865432	32987654	7777777	778899	77777		Rua Goiás	Venda Nova
Luiz Melodia	luiz@gmail.com	988766543	34568976	8888888	345234	88888		Rua Sem Nome	Viana
Elimar Lolzin	elimar@gmail.com	999988877	34567892	9999999	567533	99999		Rua do Beco	Fundao
Julio Faker	julio@gmail.com	990473645	34987654	123456	456789	12345		Rua Isabel	Aracruz

tabela completa em: <https://github.com/BD1-IDO-Higo-Icaro-Tadeu/trab01/blob/master/campos%20%20do.ods>

Tabelão criado no banco

Banco_IDo2 on postgres@PostgreSQL 9.6

```
1 select pessoa.nome,pessoa.email,pessoa.cpf_id,pessoa.rg,pessoa.senha,servico.servico_oferecido,
2 contato.contato,cep.rua as rua,bairro.bairro,cidade.cidade,estado.sigla as estado, pessoa.num_casa, cep.cep_id from pessoa
3 inner join avaliacao on (pessoa.cpf_id = avaliacao.cliente_fid)
4 inner join servico_contratado on (pessoa.cpf_id = servico_contratado.usuario_fid)
5 inner join servico on (servico.servico_id = pessoa.servico_fid)
6 inner join servico_descricao on (pessoa.cpf_id = servico_descricao.usuario_fid)
7 inner join contato on (pessoa.cpf_id = contato.usuario_fid)
8 inner join cartao on (pessoa.cpf_id = cartao.cliente_fid)
9 inner join endereco on (pessoa.cpf_id= endereco.usuario_fid)
10 inner join cep on (endereco.cep_fid = cep.cep_id)
11 inner join bairro on (bairro.bairro_id = cep.bairro_fid)
12 inner join cidade on (cidade.cidade_id = bairro.cidade_fid)
13 inner join estado on (cidade.cidade_id = estado.estado_id) where contato.tipo_contato ='telefone'
```

Data Output Explain Messages History

	nome character varying (50)	email character varying (50)	cpf_id integer	rg integer	senha character varying (40)	servico_oferecido character varying (100)	co ch
<input type="checkbox"/>	Luiz Melodia	luiz@gmail.com	888888	345234	88888	motorista	^
<input type="checkbox"/>	Tadeu Junior	tadeu_junior@hotmail.com	111111	223344	11111	pintor	
<input type="checkbox"/>	Ewerson Vieira	ewerson@gmail.com	333333	323456	33333	cozinheiro	
<input type="checkbox"/>	Lucas Gomes Irinel	luca-irinel@gmail.com	444444	265260	44444	pedreiro	
<input type="checkbox"/>	Icaro Duarte	icaro@gmail.com	555555	223344	55555	cliente	
<input type="checkbox"/>	David Vilaca	david@gmail.com	666666	445566	66666	segurança	
<input type="checkbox"/>	Leandro Goias	leandro@gmail.com	777777	778899	77777	motorista	v

query no endereço: <https://github.com/BD1-IDO-Higo-Icaro-Tadeu/trab01/blob/master/tabela.sql>

Conclusão

A metodologia usada neste trabalho, propôs ao grupo conhecer um pouco sobre o sql, e como a criação de um banco de dados demanda planejamento para que tenha necessidade de mudanças no mesmo. Acreditamos que tal metodologia no trás mais conhecimentos do que provas aplicadas em sala, onde as vezes so decoramos para passar.