



FACULDADE DE INFORMÁTICA E ADMINISTRAÇÃO PAULISTA
(FIAP)

CURSO DE ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

AdOptimize

JAISY CIBELE ALVES - RM552269

LUIZ FELIPE CAMARGO PRENDIN - RM552475

TOMAZ DE OLIVEIRA PECORARO - RM98499

RENNAN FERREIRA DA CRUZ - RM99364

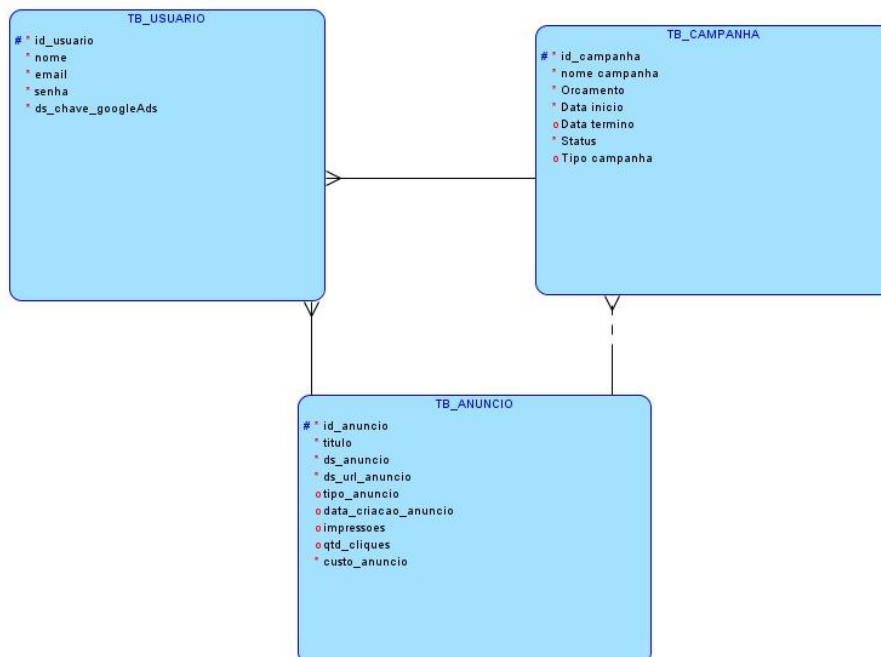
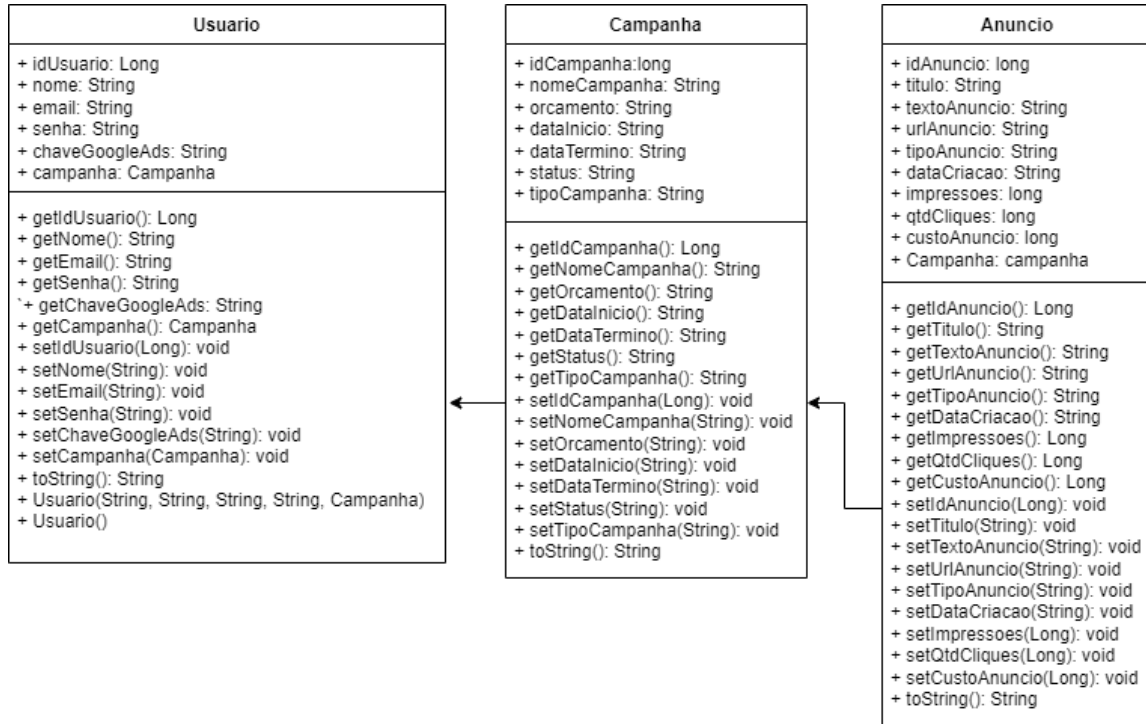
SÃO PAULO
ABRIL / 2023

O problema que estamos enfrentando é a dificuldade que muitas empresas têm em otimizar suas campanhas de anúncios no Google Ads para maximizar o retorno sobre o investimento. A complexidade e a quantidade de dados envolvidos tornam desafiador para os profissionais de marketing identificar as melhores estratégias e tomar decisões informadas. Nosso público-alvo são empresas de todos os tamanhos que utilizam o Google Ads como parte de sua estratégia de marketing digital. Isso inclui desde pequenas empresas até grandes corporações que buscam melhorar o desempenho de suas campanhas de anúncios online.

2 EXPLICAÇÃO E JUSTIFICATIVA/ OBJETIVOS DO PROJETO

Nossa solução visa oferecer insights valiosos e recomendações personalizadas para otimizar campanhas de anúncios no Google Ads. Utilizamos inteligência artificial e análise avançada de dados para coletar, analisar e interpretar os dados das campanhas, oferecendo recomendações claras e acionáveis para melhorar o desempenho e maximizar o ROI. Ao colocar o controle nas mãos dos usuários, nossa solução permite que as empresas tomem decisões mais inteligentes com base nos insights fornecidos, garantindo uma maior flexibilidade e personalização para atender às necessidades específicas de cada cliente.

3 MODELAGEM DAS CLASSES DAS ENTIDADES (UML) E DIAGRAMA DE ENTIDADE - RELACIONAMENTO



4 LINK DO REPOSITÓRIO DO PROJETO NO GITHUB

<https://github.com/jaisycibele/Challenge-2024.git>

5 LINK DO VIDEO DA SOLUÇÃO PROPOSTA

<https://www.youtube.com/watch?v=xVgYOcJJJFU>

6 CRONOGRAMA DE DESENVOLVIMENTO

TAREFA	DATA PRAZO	RESPONSÁVEL
Criação das Entidades	22/03/2024	Jaisy Cibeles Alves
Criação dos DTO's	29/04/2024	Jaisy Cibeles Alves
Criação dos services	03/04/204	Jaisy Cibeles Alves
Criação dos repositories e API	10/04/2024	Jaisy Cibeles Alves
Criação do Bean Validation e testes da API	14/04/2024	Jaisy Cibeles Alves

7 LISTA DOS ENDPOINTS

URL	VERBO
http://localhost:1521/adoptimize/usuarios/buscar/{email}	GET
http://localhost:1521/adoptimize/usuarios/cadastrar	POST

Exemplo de requisição do método POST:

```
{
  "nome": "Charles",
  "email": "charles@gmail.com",
  "senha": "1234578",
  "chaveGoogleAds": "chave101"
}
```

8 BANCO DE DADOS E SCRIPT

Usuario: rm552269
Senha: 150504
Host: oracle.fiap.com.br
Porta: 1521
SID: orcl

DROP TABLE tb_anuncio CASCADE CONSTRAINTS;

DROP TABLE tb_campanha CASCADE CONSTRAINTS;

DROP TABLE tb_usuario CASCADE CONSTRAINTS;

```

CREATE TABLE tb_anuncio (
    id_anuncio      NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY
PRIMARY KEY,
    titulo          VARCHAR2(80) NOT NULL,
    ds_anuncio      VARCHAR2(250) NOT NULL,
    ds_url_anuncio  VARCHAR2(450) NOT NULL,
    tipo_anuncio    VARCHAR2(60),
    data_criacao_anuncio  TIMESTAMP WITH LOCAL TIME ZONE,
    impressoes      NUMBER(10,0), -- Alterado para permitir números inteiros de até 10
dígitos
    qtd_cliques     NUMBER(10,0), -- Alterado para permitir números inteiros de até 10
dígitos
    custo_anuncio   NUMBER NOT NULL
);

```

-- Defina a coluna id_anuncio como chave primária usando uma constraint

```

ALTER TABLE tb_anuncio
ADD CONSTRAINT tb_anuncio_pk PRIMARY KEY (id_anuncio);

```

```

CREATE TABLE tb_campanha (
    id_campanha     NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY
PRIMARY KEY,
    nome_campanha   VARCHAR2(140) NOT NULL,
    orcamento      VARCHAR2(20 CHAR) NOT NULL,
    data_inicio     TIMESTAMP WITH LOCAL TIME ZONE NOT NULL,
    data_termino    TIMESTAMP,
    status          CHAR(1) NOT NULL,
    tipo_campanha   VARCHAR2(120),
    id_anuncio      NUMBER(10,0) -- Alterado para permitir números inteiros de até 10 dígitos
);

```

```

ALTER TABLE tb_campanha ADD CONSTRAINT tb_campanha_pk PRIMARY KEY (
id_campanha );

```

```

CREATE TABLE tb_usuario (
    id_usuario      NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY
PRIMARY KEY,
    nome            VARCHAR2(100 CHAR) NOT NULL,
    email           VARCHAR2(160 CHAR) NOT NULL,
    senha           VARCHAR2(90 CHAR) NOT NULL,
    ds_chave_googleads  VARCHAR2(200) NOT NULL,
    fk_id_anuncio   NUMBER(10,0), -- Alterado para permitir números inteiros de até 10
dígitos
    fk_id_campanha  NUMBER(10,0)-- Alterado para permitir números inteiros de até 10
dígitos
);

```

```
ALTER TABLE tb_usuario ADD CONSTRAINT tb_usuario_pk PRIMARY KEY (
id_usuario );
```

```
ALTER TABLE tb_usuario
ADD CONSTRAINT relation_4 FOREIGN KEY ( fk_id_campanha )
REFERENCES tb_campanha ( id_campanha );
```

```
ALTER TABLE tb_campanha
ADD CONSTRAINT relation_5 FOREIGN KEY ( id_anuncio )
REFERENCES tb_anuncio ( id_anuncio );
```

```
ALTER TABLE tb_usuario
ADD CONSTRAINT tb_usuariov1_tb_anuncio_fk FOREIGN KEY ( fk_id_anuncio )
REFERENCES tb_anuncio ( id_anuncio );
```

--INSERTS TABELA ANUNCIO

```
INSERT INTO tb_anuncio (id_anuncio, titulo, ds_anuncio, ds_url_anuncio, tipo_anuncio,
data_criacao_anuncio, impressoes, qtd_cliques, custo_anuncio)
VALUES (1, 'Anúncio 1', 'Descrição do Anúncio 1', 'http://www.anuncio1.com', 'Texto',
SYSTIMESTAMP, 100, 10, 50.00);
```

```
INSERT INTO tb_anuncio (id_anuncio, titulo, ds_anuncio, ds_url_anuncio, tipo_anuncio,
data_criacao_anuncio, impressoes, qtd_cliques, custo_anuncio)
VALUES (2, 'Anúncio 2', 'Descrição do Anúncio 2', 'http://www.anuncio2.com', 'Imagem',
SYSTIMESTAMP, 150, 20, 70.00);
```

```
INSERT INTO tb_anuncio (id_anuncio, titulo, ds_anuncio, ds_url_anuncio, tipo_anuncio,
data_criacao_anuncio, impressoes, qtd_cliques, custo_anuncio)
VALUES (3, 'Anúncio 3', 'Descrição do Anúncio 3', 'http://www.anuncio3.com', 'Vídeo',
SYSTIMESTAMP, 200, 30, 100.00);
```

```
INSERT INTO tb_anuncio (id_anuncio, titulo, ds_anuncio, ds_url_anuncio, tipo_anuncio,
data_criacao_anuncio, impressoes, qtd_cliques, custo_anuncio)
VALUES (4, 'Anúncio 4', 'Descrição do Anúncio 4', 'http://www.anuncio4.com', 'Texto',
SYSTIMESTAMP, 120, 15, 60.00);
```

```
INSERT INTO tb_anuncio (id_anuncio, titulo, ds_anuncio, ds_url_anuncio, tipo_anuncio,
data_criacao_anuncio, impressoes, qtd_cliques, custo_anuncio)
VALUES (5, 'Anúncio 5', 'Descrição do Anúncio 5', 'http://www.anuncio5.com', 'Imagem',
SYSTIMESTAMP, 180, 25, 80.00);
```


--INSERTS TABELA CAMPANHA

```
INSERT INTO tb_campanha (id_campanha, nome_campanha, orcamento, data_inicio,
data_termino, status, tipo_campanha, id_anuncio)
VALUES (1, 'Campanha 1', '1000.00', SYSTIMESTAMP, SYSTIMESTAMP + INTERVAL
'30' DAY, 'A', 'Pesquisa', 1);
```

```
INSERT INTO tb_campanha (id_campanha, nome_campanha, orcamento, data_inicio,
data_termino, status, tipo_campanha, id_anuncio)
VALUES (2, 'Campanha 2', '2000.00', SYSTIMESTAMP, SYSTIMESTAMP + INTERVAL
'45' DAY, 'A', 'Display', 2);
```

```
INSERT INTO tb_campanha (id_campanha, nome_campanha, orcamento, data_inicio,
data_termino, status, tipo_campanha, id_anuncio)
VALUES (3, 'Campanha 3', '1500.00', SYSTIMESTAMP, SYSTIMESTAMP + INTERVAL
'60' DAY, 'A', 'VÍdeo', 3);
```

```
INSERT INTO tb_campanha (id_campanha, nome_campanha, orcamento, data_inicio,
data_termino, status, tipo_campanha, id_anuncio)
VALUES (4, 'Campanha 4', '1800.00', SYSTIMESTAMP, SYSTIMESTAMP + INTERVAL
'90' DAY, 'A', 'Pesquisa', 4);
```

```
INSERT INTO tb_campanha (id_campanha, nome_campanha, orcamento, data_inicio,
data_termino, status, tipo_campanha, id_anuncio)
VALUES (5, 'Campanha 5', '2500.00', SYSTIMESTAMP, SYSTIMESTAMP + INTERVAL
'120' HOUR, 'A', 'Display', 5);
```

--INSERTS TABELA USUARIO

```
INSERT INTO tb_usuario (id_usuario, nome, email, senha, ds_chave_googleads,
fk_id_anuncio, fk_id_campanha)
VALUES (1, 'João', 'joao@example.com', 'senha123', 'chave123', 1, 1);
```

```
INSERT INTO tb_usuario (id_usuario, nome, email, senha, ds_chave_googleads,
fk_id_anuncio, fk_id_campanha)
VALUES (2, 'Maria', 'maria@example.com', 'senha456', 'chave456', 2, 2);
```

```
INSERT INTO tb_usuario (id_usuario, nome, email, senha, ds_chave_googleads,
fk_id_anuncio, fk_id_campanha)
VALUES (3, 'Pedro', 'pedro@example.com', 'senha789', 'chave789', 3, 3);
```

```
INSERT INTO tb_usuario (id_usuario, nome, email, senha, ds_chave_googleads,
fk_id_anuncio, fk_id_campanha)
VALUES (4, 'Ana', 'ana@example.com', 'senhaabc', 'chaveabc', 4, 4);
```

```
INSERT INTO tb_usuario (id_usuario, nome, email, senha, ds_chave_googleads,
fk_id_anuncio, fk_id_campanha)
VALUES (5, 'Carlos', 'carlos@example.com', 'senhaxyz', 'chavexyz', 5, 5);
```

```

--BLOCOS ANONIMOS
SET SERVEROUTPUT ON
-- Mostrar os anúncios (tb_anuncio) com os detalhes da campanha (tb_campanha) a que
pertencem
DECLARE
    v_id_anuncio tb_anuncio.id_anuncio%TYPE;
    v_titulo tb_anuncio.titulo%TYPE;
    v_nome_campanha tb_campanha.nome_campanha%TYPE;
BEGIN
    FOR anuncio_rec IN (SELECT a.id_anuncio, a.titulo, c.nome_campanha
                        FROM tb_anuncio a
                        INNER JOIN tb_campanha c ON a.id_anuncio = c.id_anuncio
                        ORDER BY a.id_anuncio)
    LOOP
        v_id_anuncio := anuncio_rec.id_anuncio;
        v_titulo := anuncio_rec.titulo;
        v_nome_campanha := anuncio_rec.nome_campanha;
        DBMS_OUTPUT.PUT_LINE('ID do Anúncio: ' || v_id_anuncio || ', Título: ' || v_titulo ||
        ', Campanha: ' || v_nome_campanha);
    END LOOP;
END;

```

```

-- Mostrar o número de anúncios por tipo de campanha (tb_campanha)
DECLARE
    v_tipo_campanha tb_campanha.tipo_campanha%TYPE;
    v_count NUMBER;
BEGIN
    FOR tipo_rec IN (SELECT c.tipo_campanha, COUNT(a.id_anuncio) AS num_anuncios
                    FROM tb_anuncio a
                    INNER JOIN tb_campanha c ON a.id_anuncio = c.id_anuncio
                    GROUP BY c.tipo_campanha
                    ORDER BY num_anuncios DESC)
    LOOP
        v_tipo_campanha := tipo_rec.tipo_campanha;
        v_count := tipo_rec.num_anuncios;
        DBMS_OUTPUT.PUT_LINE('Tipo de Campanha: ' || v_tipo_campanha || ', Número de
        Anúncios: ' || v_count);
    END LOOP;
END;
/

```

9 TESTES DA API

Para testar a API Rest, utilizamos o console e o Postman.

[illegible]

POST http://localhost:1521/adaptimize/usuarios/cadastrar

Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings

Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON

Beautify

```
1 {
2   "nome": "Maria",
3   "email": "maria@gmail.com",
4   "senha": "1234578",
5   "chaveGoogleAds": "chave234"
6 }
```

Body Cookies Headers (5) Test Results (1/1)

Status: 201 Created Time: 362 ms Size: 222 B Save as example

Pretty Raw Preview Visualize

JSON



```
1 {
2   "nome": "Maria",
3   "email": "maria@gmail.com",
4   "senha": ""
5 }
```

select * from tb_usuario;


Saída do Script


Resultado da Consulta



SQL Todas as Linhas Extraídas: 2 em 0,017 segundos

	ID_USUARIO	NOME	EMAIL	SENHA	DS_CHAVE_GOOGLEADS	FK_ID_ANUNCIO	FK_ID_CAMPANHA
1	1	Jaisy	jaisy@gmail.com	123456	chavel23	(null)	(null)
2	2	Maria	maria@gmail.com	1234578	chave234	(null)	(null)

```
Hibernate:
insert
into
  tb_usuario
(fk_id_campanha, ds_chave_googleads, email, nome, senha, id_usuario)
values
  (?, ?, ?, ?, ?, default)
2024-04-14T12:03:32.170-03:00 TRACE 12984 --- [challenge] [nio-1521-exec-1] org.hibernate.orm.jdbc.bind : binding parameter (1:BIGINT) <- [null]
2024-04-14T12:03:32.170-03:00 TRACE 12984 --- [challenge] [nio-1521-exec-1] org.hibernate.orm.jdbc.bind : binding parameter (2:VARCHAR) <- [chave234]
2024-04-14T12:03:32.170-03:00 TRACE 12984 --- [challenge] [nio-1521-exec-1] org.hibernate.orm.jdbc.bind : binding parameter (3:VARCHAR) <- [maria@gmail.com]
2024-04-14T12:03:32.170-03:00 TRACE 12984 --- [challenge] [nio-1521-exec-1] org.hibernate.orm.jdbc.bind : binding parameter (4:VARCHAR) <- [Maria]
2024-04-14T12:03:32.170-03:00 TRACE 12984 --- [challenge] [nio-1521-exec-1] org.hibernate.orm.jdbc.bind : binding parameter (5:VARCHAR) <- [1234578]
```

 REST API basics: CRUD, test & variable / **Get data**

 Save

GET

http://localhost:1521/adaptimize/usuarios/buscar/maria@gmail.com

Send


Params

Authorization

Headers (6)

Body

Pre-request Script

Tests 

Settings

Cookies

Query Params



	Key	Value	Description	...	Bulk Edit
	Key	Value	Description		

Body

Cookies

Headers (6)

Test Results (1/1)

 Status: 200 OK Time: 59 ms Size: 261 B  Save as example ...


Pretty



Raw

Preview

Visualize

JSON



1

{

2

"nome": "Maria",

3

"email": "maria@gmail.com",

4

"senha": ""

5

}

```
Hibernate:
select
  ul_0.id_usuario,
  ul_0.fk_id_campanha,
  ul_0.ds_chave_googleads,
  ul_0.email,
  ul_0.nome,
  ul_0.senha
from
  tb_usuario ul_0
where
  ul_0.email=?

2024-04-14T12:06:35.354-03:00 TRACE 12984 --- [challenge] [nio-1521-exec-7] org.hibernate.orm.jdbc.bind      : binding parameter (1:VARCHAR) <- [maria@gmail.com]
```