

**FACULDADE DE INFORMÁTICA E ADMINISTRAÇÃO PAULISTA**

Caike Dametto - RM: 558614

Guilherme Janunzzi - RM: 558461

**ExtremeHelp**

São Paulo/SP

2025

## SUMÁRIO

Sobre o Projeto.....	2
Modelo lógico.....	3
Modelo relacional.....	4
Scripts SQL.....	5
Prints de Execução.....	6-13
Integração com o Java.....	14
Link vídeo demonstração.....	15

## Sobre o Projeto ExtremeHelp

### O Problema

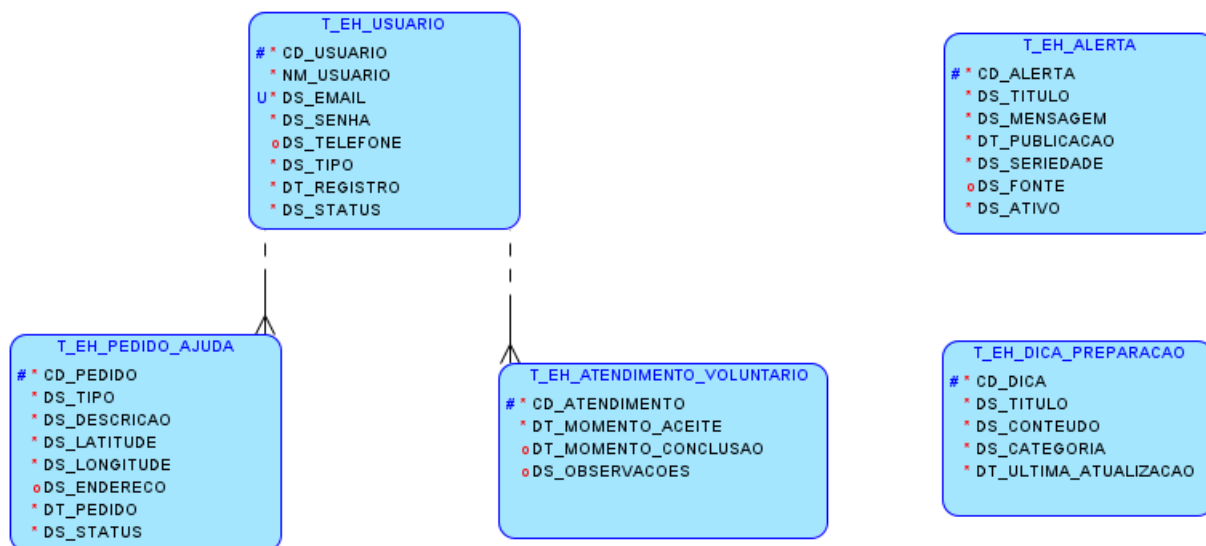
Em situações de emergência e desastres naturais, surge uma lacuna crítica entre a necessidade imediata da população afetada e a capacidade de resposta organizada. Pessoas em vulnerabilidade têm dificuldade em comunicar suas necessidades específicas, enquanto cidadãos e grupos dispostos a ajudar muitas vezes não sabem onde e como direcionar seus esforços de forma eficaz. Além disso, a disseminação de informações vitais, como alertas de risco e guias de prevenção, costuma ser fragmentada, diminuindo o poder de preparação da comunidade.

### A Solução

ExtremeHelp nasce como uma solução para este desafio, funcionando como um ecossistema digital centralizado para a gestão de crises comunitárias. A plataforma foi desenvolvida para:

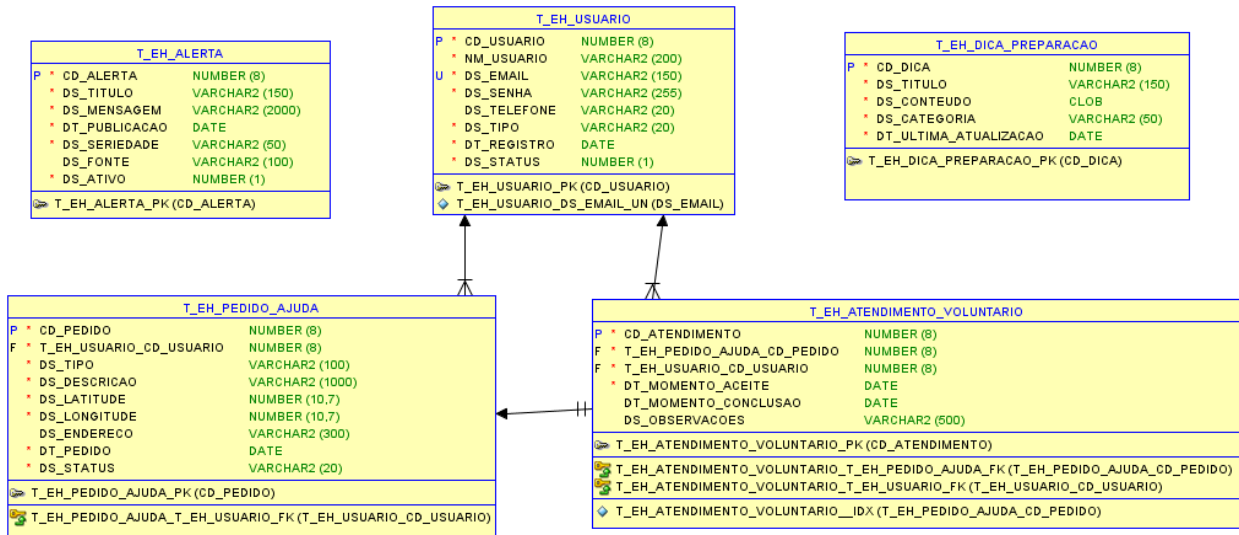
1. **Conectar Ajuda e Necessidade:** Criar um canal direto onde "Solicitantes" podem registrar pedidos de ajuda detalhados (ex: alimentos, abrigo, transporte, medicamentos) com sua localização exata. Do outro lado, "Voluntários" podem visualizar esses pedidos em um mapa ou lista e aceitar aqueles que podem atender, otimizando a distribuição de recursos.
2. **Centralizar a Comunicação Oficial:** Servir como um painel de controle para que "Administradores" (como a Defesa Civil ou ONGs) possam emitir alertas de emergência em tempo real para todos os usuários da plataforma, garantindo que informações críticas cheguem rapidamente a quem precisa.
3. **Promover a Preparação:** Oferecer uma base de conhecimento com "Dicas de Preparação", ensinando a população a se preparar para diferentes tipos de adversidades e a como agir durante uma emergência.

## Modelo lógico



(Para uma melhor visualização o modelo está disponibilizado em arquivos .png e .pdf dentro da pasta /DataModeler/Imagens)

## Modelo relacional



(Para uma melhor visualização o modelo está disponibilizado em arquivos .png e .pdf dentro da pasta /DataModeler/Imagens)

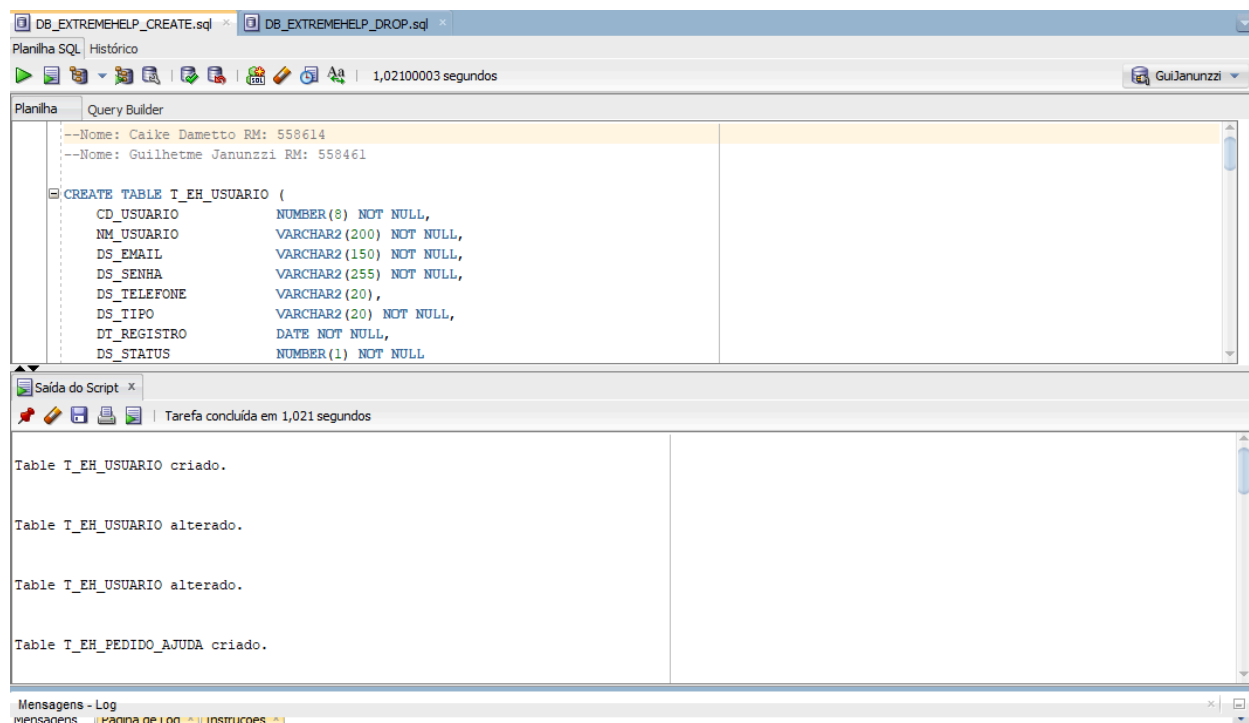
## Scripts SQL

Toda a estrutura do banco de dados Oracle, incluindo a criação de tabelas (DDL), procedures, functions e a inserção de dados, está versionada e disponível para execução.

Os scripts estão organizados em arquivos .sql e podem ser encontrados no diretório /SQL na raiz do projeto.

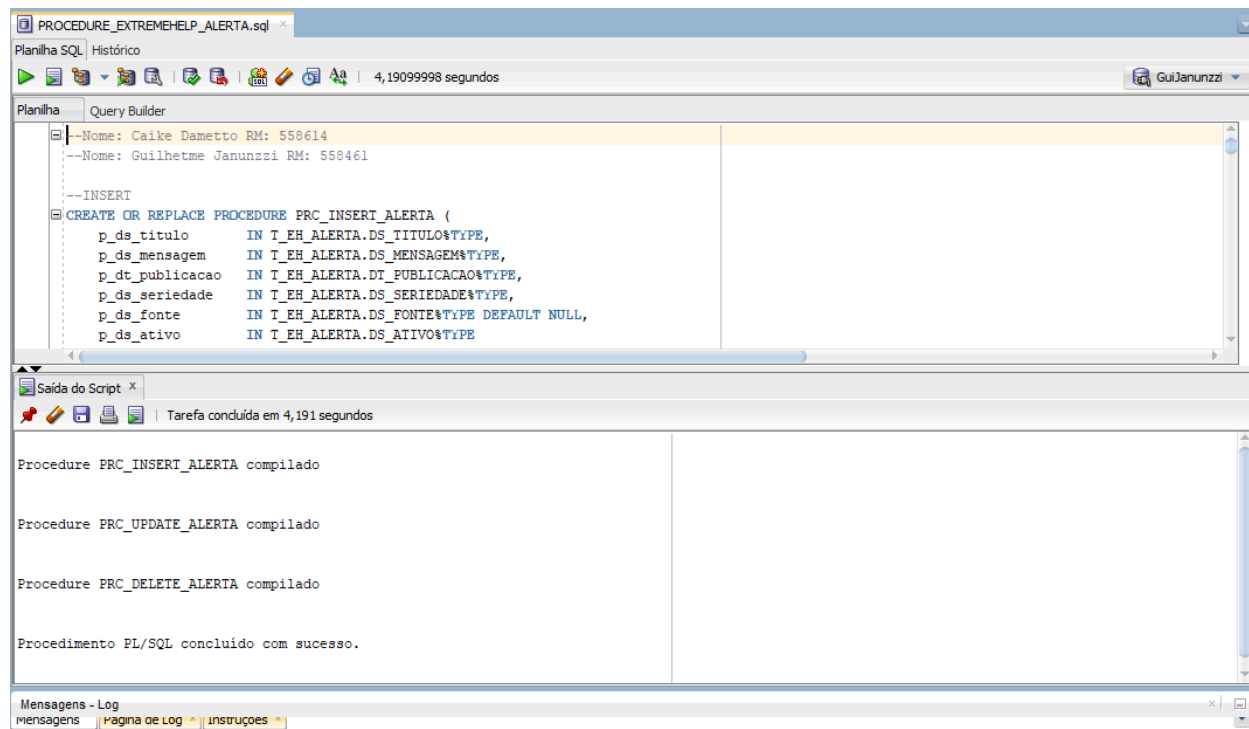
## Prints de Execução

### Create:

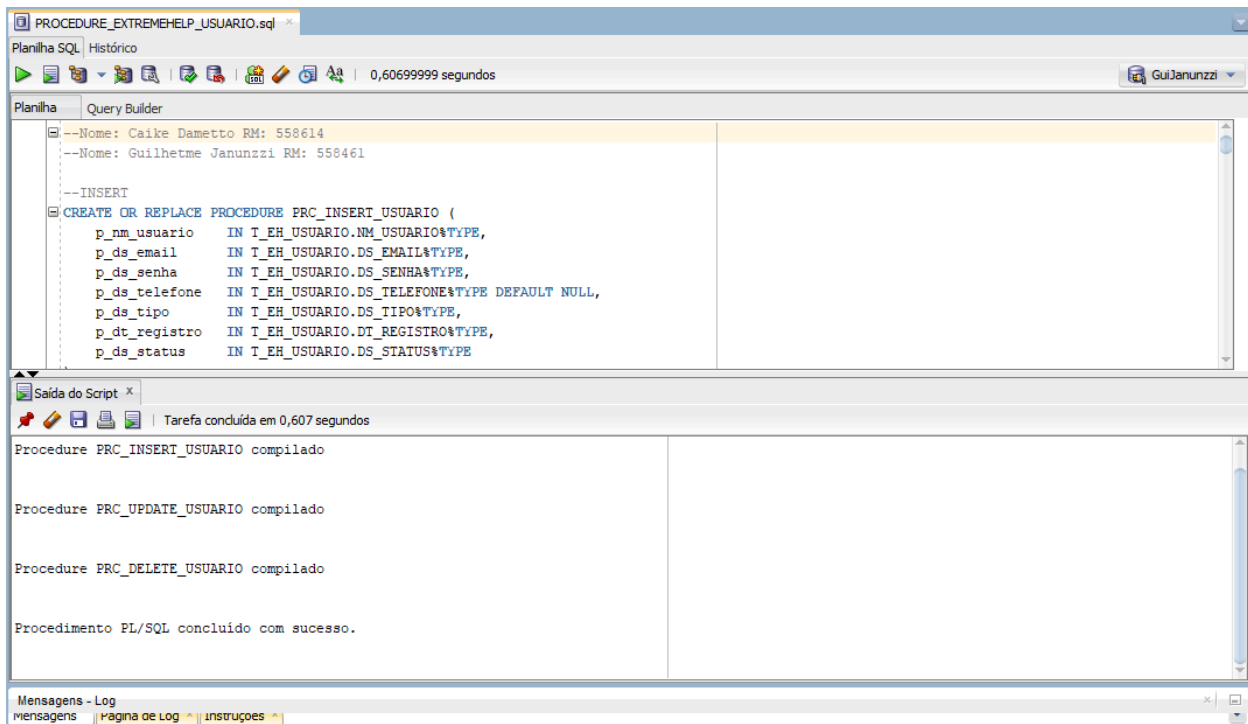


## Procedures

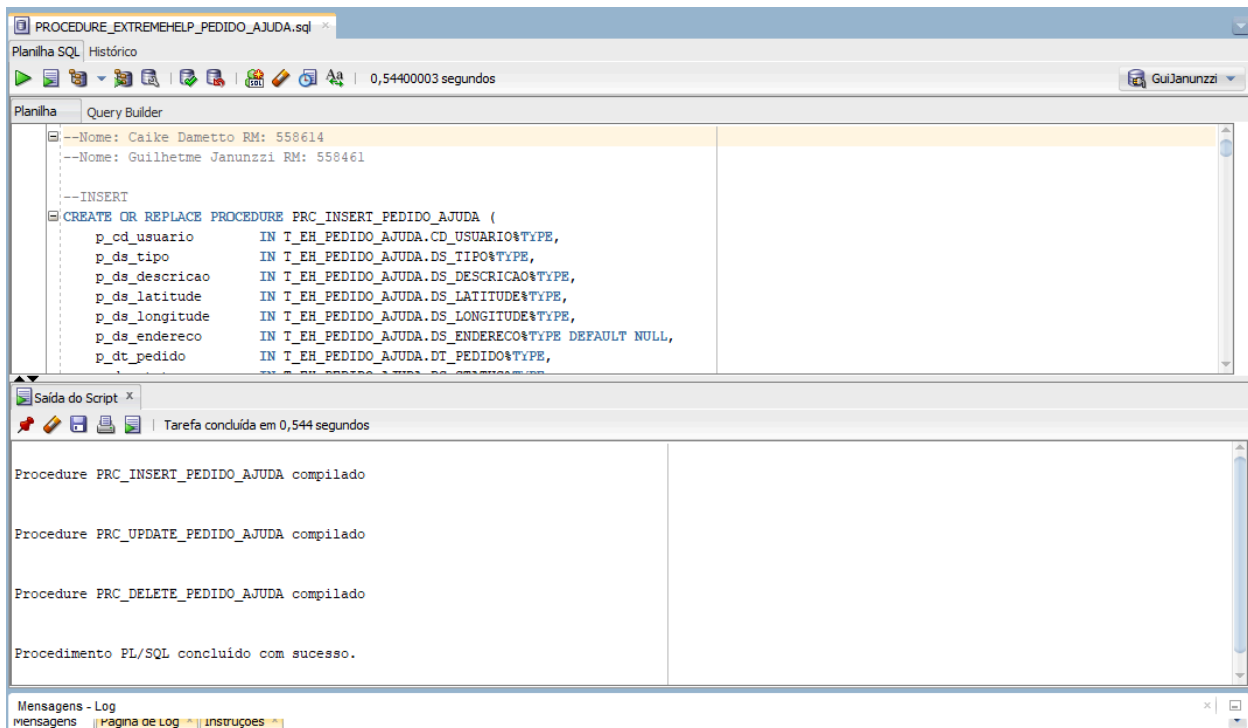
### Alerta:



## Usuario:



## Pedido de Ajuda:





## Atendimento voluntário:

PROCEDURE\_EXTREMEHELP\_ATENDIMENTO.sql

Planilha SQL Histórico

0,70200002 segundos

GuiJanunzzi

Planilha Query Builder

```
--Nome: Caike Dametto RM: 558614
--Nome: Guilhetme Janunzzi RM: 558461

--INSERT
CREATE OR REPLACE PROCEDURE PRC_INSERT_ATEND_VOL (
    p_cd_pedido          IN T_EH_ATENDIMENTO_VOLUNTARIO.CD_PEDIDO%TYPE,
    p_cd_usuario         IN T_EH_ATENDIMENTO_VOLUNTARIO.CD_USUARIO%TYPE,
    p_dt_momento_aceite  IN T_EH_ATENDIMENTO_VOLUNTARIO.DT_MOMENTO_ACEITE%TYPE,
    p_dt_momento_conclusao IN T_EH_ATENDIMENTO_VOLUNTARIO.DT_MOMENTO_CONCLUSAO%TYPE DEFAULT NULL,
    p_ds_observacoes     IN T_EH_ATENDIMENTO_VOLUNTARIO.DS_OBSERVACOES%TYPE DEFAULT NULL
)
IS
BEGIN
```

Saída do Script

Tarefa concluída em 0,702 segundos

Procedure PRC\_INSERT\_ATEND\_VOL compilado

Procedure PRC\_UPDATE\_ATEND\_VOL compilado

Procedure PRC\_DELETE\_ATEND\_VOL compilado

Procedimento PL/SQL concluído com sucesso.

Mensagens - Log

mensagens | Página de Log | Instruções

## Dica de preparação:

PROCEDURE\_EXTREMEHELP\_DICA.sql

Planilha SQL Histórico

0,56099999 segundos

GuiJanunzzi

Planilha Query Builder

```
--Nome: Caike Dametto RM: 558614
--Nome: Guilhetme Janunzzi RM: 558461

--INSERT
CREATE OR REPLACE PROCEDURE PRC_INSERT_DICA (
    p_ds_titulo          IN T_EH_DICA_PREPARACAO.DS_TITULO%TYPE,
    p_ds_conteudo        IN T_EH_DICA_PREPARACAO.DS_CONTEUDO%TYPE,
    p_ds_categoria       IN T_EH_DICA_PREPARACAO.DS_CATEGORIA%TYPE,
    p_dt_ultima_atualizacao IN T_EH_DICA_PREPARACAO.DT_ULTIMA_ATUALIZACAO%TYPE
)
IS
BEGIN
```

Saída do Script

Tarefa concluída em 0,561 segundos

Procedure PRC\_INSERT\_DICA compilado

Procedure PRC\_UPDATE\_DICA compilado

Procedure PRC\_DELETE\_DICA compilado

Procedimento PL/SQL concluído com sucesso.

Mensagens - Log

mensagens | Página de Log | Instruções

## Blocos:

1 -

Planilha SQL | Histórico | 0,199 segundos | GuiJanunzzi

Planilha | Query Builder

```
--Nome: Caike Dametto RM: 558614
--Nome: Guilhetme Janunzzi RM: 558461

--Relatório de Voluntários Ativos com Detalhes da Última Atividade

SET SERVEROUTPUT ON;
```

Saída do Script x | Tarefa concluída em 0,199 segundos

```
--- Relatório de Voluntários Produtivos (Com >= 1 Atendimentos Concluídos) ---
-----
Voluntário ID: 2
Nome: Bruno Costa
Email: bruno.costa@extremehelp.com
Total de Atendimentos Concluídos: 1
Data do Último Atendimento Concluído: 02/06/2025 15:30:00
-----
Voluntário ID: 5
Nome: Elisa Mendes
Email: elisa.mendes@extremehelp.com
Total de Atendimentos Concluídos: 1
Data do Último Atendimento Concluído: 08/06/2025 02:45:29
--- Fim do Relatório ---

Procedimento PL/SQL concluído com sucesso.
```

Mensagens - Log  
mensagens | Página de Log | Instruções

2 -

Planilha SQL | Histórico | 0,19400001 segundos | GuiJanunzzi

Planilha | Query Builder

```
--Nome: Caike Dametto RM: 558614
--Nome: Guilhetme Janunzzi RM: 558461

--Análise de Pedidos de Ajuda Antigos e Pendentes por Tipo

SET SERVEROUTPUT ON;

DECLARE
  v_dias_antigo NUMBER := 7; -- Pedidos pendentes há mais de 7 dias são considerados antigos
  v_min_pedidos_antigos_por_tipo NUMBER := 1; -- Tipos de ajuda com pelo menos X pedidos antigos

```

Saída do Script x | Tarefa concluída em 0,194 segundos

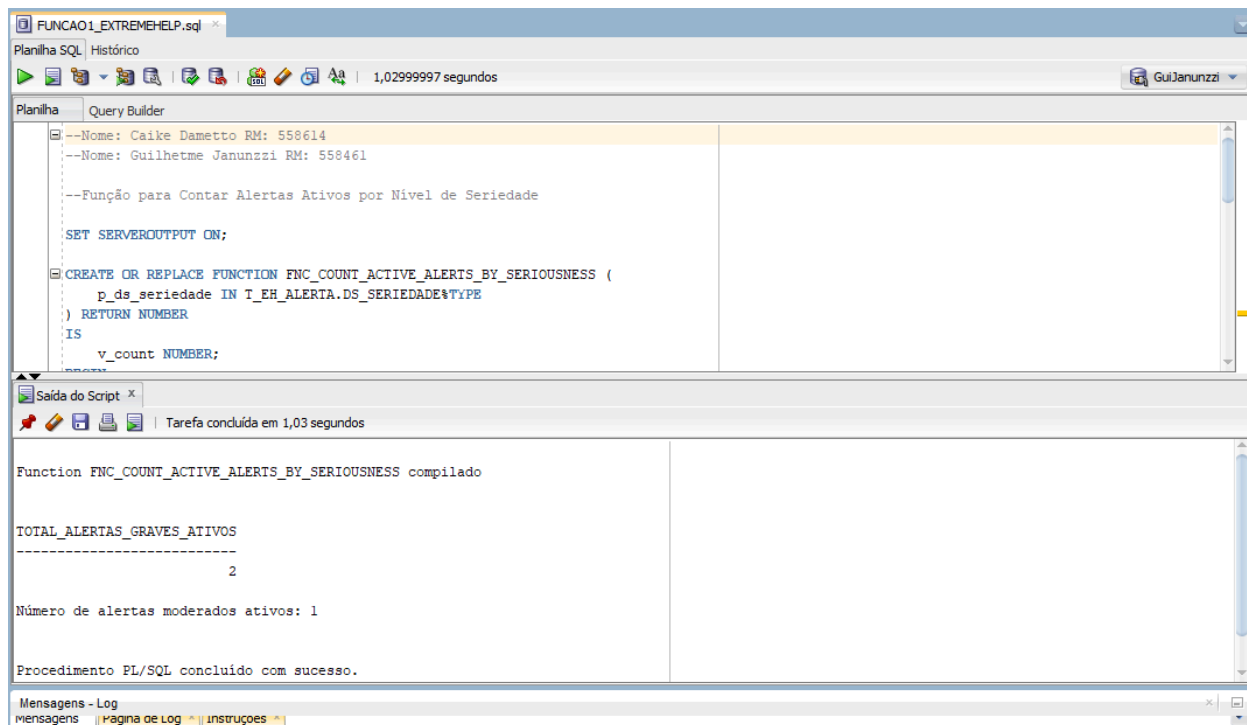
```
--- Análise de Pedidos de Ajuda Antigos e Pendentes (Mais de 7 dias) ---
-----
Tipo de Ajuda: Alimentação
Total de Pedidos Antigos Pendentes: 1
Usuários com Pedidos Antigos Pendentes para este Tipo:
  - Solicitante: Ana Silva (Email: ana.silva@extremehelp.com)
    Pedido ID: 1 | Data: 01/06/2025
    Descrição: Solicitação de cesta básica para família de 4 pessoas. Urgente...
--- Fim da Análise ---

Procedimento PL/SQL concluído com sucesso.
```

Mensagens - Log  
mensagens | Página de Log | Instruções

## Funções:

1-



Funcao01\_EXTREMEHELP.sql

Planilha SQL | Histórico

1,02999997 segundos

GuiJanunzzi

Planilha Query Builder

```
--Nome: Caike Dametto RM: 558614
--Nome: Guilhetme Janunzzi RM: 558461

--Função para Contar Alertas Ativos por Nivel de Seriedade

SET SERVEROUTPUT ON;

CREATE OR REPLACE FUNCTION FNC_COUNT_ACTIVE_ALERTS_BY_SERIOUSNESS (
  p_ds_seriedade IN T_EH_ALERTA.DS_SERIEDADE%TYPE
) RETURN NUMBER
IS
  v_count NUMBER;
```

Salida do Script x

Tarefa concluída em 1,03 segundos

Function FNC\_COUNT\_ACTIVE\_ALERTS\_BY\_SERIOUSNESS compilado

TOTAL\_ALERTAS\_GRAVES\_ATIVOS

---

2

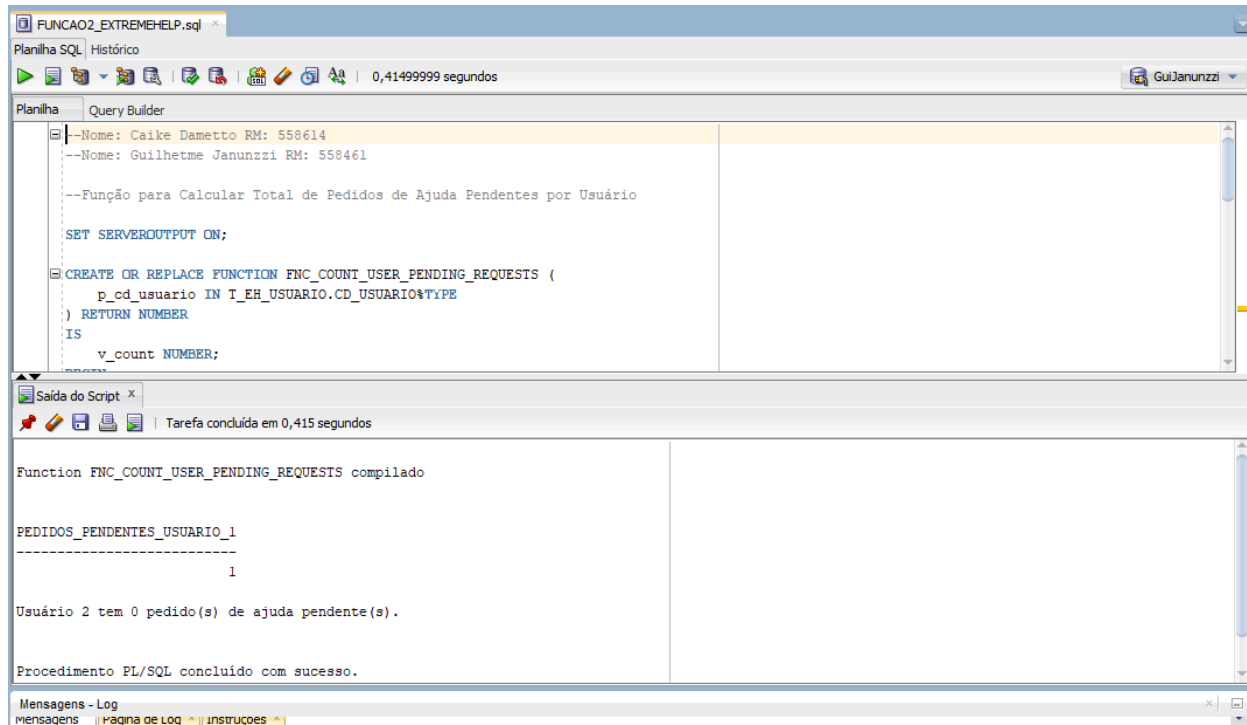
Número de alertas moderados ativos: 1

Procedimento PL/SQL concluído com sucesso.

Mensagens - Log

mensagens | Página de Log | Instruções

2-



Funcao02\_EXTREMEHELP.sql

Planilha SQL | Histórico

0,41499999 segundos

GuiJanunzzi

Planilha Query Builder

```
--Nome: Caike Dametto RM: 558614
--Nome: Guilhetme Janunzzi RM: 558461

--Função para Calcular Total de Pedidos de Ajuda Pendentes por Usuário

SET SERVEROUTPUT ON;

CREATE OR REPLACE FUNCTION FNC_COUNT_USER_PENDING_REQUESTS (
  p_cd_usuario IN T_EH_USUARIO.CD_USUARIO%TYPE
) RETURN NUMBER
IS
  v_count NUMBER;
```

Salida do Script x

Tarefa concluída em 0,415 segundos

Function FNC\_COUNT\_USER\_PENDING\_REQUESTS compilado

PEDIDOS\_PENDENTES\_USUARIO\_1

---

1

Usuário 2 tem 0 pedido(s) de ajuda pendente(s).

Procedimento PL/SQL concluído com sucesso.

Mensagens - Log

mensagens | Página de Log | Instruções

## Relatorios:

1-

The screenshot shows a SQL query editor window titled 'Relatorio1.sql'. The query is as follows:

```

T_ER_ATENDIMENTO_VOLUNTARIO av ON u.CD_USUARIO = av.CD_USUARIO
WHERE
    u.DS_TIPO = 'VOLUNTARIO'
    AND av.DT_MOMENTO_CONCLUSAO IS NOT NULL
GROUP BY
    u.CD_USUARIO, u.NM_USUARIO, u.DS_EMAIL
HAVING
    COUNT(av.CD_ATENDIMENTO) > 0
)
WHERE rn <= 5
ORDER BY
    TOTAL_ATENDIMENTOS_CONCLUIDOS DESC, NM_USUARIO ASC;
  
```

The results pane shows the following data:

DS_EMAIL	TOTAL_ATENDIMENTOS_CONCLUIDOS
bruno.costa@extremehelp.com	1
elisa.mendes@extremehelp.com	1

2-

The screenshot shows a SQL query editor window titled 'Relatorio2.sql'. The query is as follows:

```

--Nome: Caike Dametto RM: 558614
--Nome: Guilherme Janunzzi RM: 558461
--Tempo Médio de Conclusão de Pedidos de Ajuda por Tipo

SELECT
    pa.DS_TIPO AS TIPO_PEDIDO,
    COUNT(pa.CD_PEDIDO) AS TOTAL_PEDIDOS_CONCLUIDOS,
    ROUND(AVG(av.DT_MOMENTO_CONCLUSAO - pa.DT_PEDIDO), 2) AS TEMPO_MEDIO_CONCLUSAO_DIAS
FROM
    T_ER_PEDIDO_AJUDA pa
JOIN
    T_ER_ATENDIMENTO_VOLUNTARIO av ON pa.CD_PEDIDO = av.CD_PEDIDO
  
```

The results pane shows the following data:

TIPO_PEDIDO	TOTAL_PEDIDOS_CONCLUIDOS	TEMPO_MEDIO_CONCLUSAO_DIAS
Abrigo Temporário	1	5
Alimentação	1	1,23

3-

The screenshot shows the SQL Developer interface with a query titled 'Relatorio3.sql'. The query is designed to list users with multiple pending requests. The results are displayed in a table with two columns: 'EMAIL\_SOLICITANTE' and 'TOTAL\_PEDIDOS\_NAO\_CONCLUIDOS'.

```
--Nome: Caíke Dametto RM: 558614
--Nome: Guilherme Janunzzi RM: 558461

--Solicitantes com Múltiplos Pedidos Não Concluídos

SELECT
  u.NM_USUARIO AS NOME_SOLICITANTE,
  u.DS_EMAIL AS EMAIL_SOLICITANTE,
  COUNT(pa.CD_PEDIDO) AS TOTAL_PEDIDOS_NAO_CONCLUIDOS
FROM
  T_EH_USUARIO u
JOIN
  T_PEDIDO pa ON u.USUARIO = pa.USUARIO
```

EMAIL_SOLICITANTE	TOTAL_PEDIDOS_NAO_CONCLUIDOS
ana.silva@extremehelp.com	2
daniel.farias@extremehelp.com	2

4-

The screenshot shows the SQL Developer interface with a query titled 'Relatorio4.sql'. The query is designed to show the frequency of active alerts by severity level. The results are displayed in a table with two columns: 'DS\_SERIEDADE' and 'QUANTIDADE\_ALERTAS\_ATIVOS'.

```
--Nome: Caíke Dametto RM: 558614
--Nome: Guilherme Janunzzi RM: 558461

--Frequência de Alertas Ativos por Nível de Seriedade

SELECT
  DS_SERIEDADE,
  COUNT(CD_ALERTA) AS QUANTIDADE_ALERTAS_ATIVOS
FROM
  T_EH_ALERTA
WHERE
  DS_ATIVO = 1 -- Considera apenas alertas ativos (1 = ativo)
```

DS_SERIEDADE	QUANTIDADE_ALERTAS_ATIVOS
GRAVE	2
INFORMATIVO	1
MODERADO	1

5 -

The screenshot shows the SQL Developer interface with a query titled 'Relatorio5.sql'. The query is a SELECT statement that joins 'vol.NM\_USUARIO' with 'vol.DS\_EMAIL' and 'vol.DT\_REGISTRO', and counts the number of completed requests ('COUNT(av.CD\_PEDIDO) AS TOTAL\_PEDIDOS\_CONCLUIDOS') for each user. The results are displayed in a table with columns: EMAIL\_VOLUNTARIO, DATA\_REGIS, and TOTAL\_PEDIDOS\_CONCI. The data shows two users: bruno.costa@extremehelp.com and elisa.mendes@extremehelp.com, with registration dates 02/06/2025 and 08/06/2025 respectively.

```
--Nome: Caike Dametto RM: 558614
--Nome: Guilherme Janunzzi RM: 558461

--Número de Pedidos Concluídos por Voluntários Registrados Recentemente

SELECT
vol.NM_USUARIO AS NOME_VOLUNTARIO,
vol.DS_EMAIL AS EMAIL_VOLUNTARIO,
TO_CHAR(vol.DT_REGISTRO, 'DD/MM/YYYY') AS DATA_REGISTRO_VOLUNTARIO,
COUNT(av.CD_PEDIDO) AS TOTAL_PEDIDOS_CONCLUIDOS
FROM
T_EH_USUARIO vol
```

EMAIL_VOLUNTARIO	DATA_REGIS	TOTAL_PEDIDOS_CONCI
bruno.costa@extremehelp.com	02/06/2025	
elisa.mendes@extremehelp.com	08/06/2025	

**DROP:**

The screenshot shows the SQL Developer interface with a script titled 'DB\_EXTREMEHELP\_DROP.sql'. The script contains five DROP TABLE statements, each with CASCADE CONSTRAINTS, for tables T\_EH\_USUARIO, T\_EH\_PEDIDO\_AJUDA, T\_EH\_ATENDIMENTO\_VOLUNTARIO, T\_EH\_ALERTA, and T\_EH\_DICA\_PREPARACAO. The execution results show that all five tables were successfully eliminated.

```
DROP TABLE T_EH_USUARIO CASCADE CONSTRAINTS;

DROP TABLE T_EH_PEDIDO_AJUDA CASCADE CONSTRAINTS;

DROP TABLE T_EH_ATENDIMENTO_VOLUNTARIO CASCADE CONSTRAINTS;

DROP TABLE T_EH_ALERTA CASCADE CONSTRAINTS;

DROP TABLE T_EH_DICA_PREPARACAO CASCADE CONSTRAINTS;
```

Table T\_EH\_USUARIO eliminado.

Table T\_EH\_PEDIDO\_AJUDA eliminado.

Table T\_EH\_ATENDIMENTO\_VOLUNTARIO eliminado.

Table T\_EH\_ALERTA eliminado.

## Integração com o Java

A integração entre a nossa aplicação Java e o banco de dados Oracle é realizada através do ecossistema Spring Boot, que automatiza e simplifica a comunicação. A conexão segue uma arquitetura em camadas:

1. **Spring Data JPA (Camada de Abstração):** No topo, utilizamos o Spring Data JPA. Ele nos permite interagir com o banco de dados através de interfaces de repositório (como `PedidoAjudaRepository`), sem a necessidade de escrever queries SQL manualmente. Nós definimos métodos como `findAll()` ou `findById()`, e o Spring Data JPA cuida de gerar as consultas por baixo dos panos.
2. **Hibernate (Camada ORM):** Como implementação do JPA, o Hibernate atua como a ponte principal. Ele é um framework de Mapeamento Objeto-Relacional (ORM) que "traduz" nossas classes Java (as Entidades, como `Usuario` e `PedidoAjuda`) em tabelas e colunas do banco de dados, e vice-versa. Quando salvamos um objeto `Usuario` em Java, o Hibernate é quem gera o comando `INSERT INTO T_EH_USUARIO ...` em SQL.
3. **JDBC e HikariCP (Camada de Conexão):** Na camada mais baixa, o Driver JDBC da Oracle é o "tradutor" final que permite ao Java se comunicar diretamente com o protocolo do banco de dados Oracle. O Spring Boot utiliza o HikariCP, um pool de conexões, para gerenciar e otimizar as conexões de banco de dados, garantindo performance e resiliência.

Em resumo, a integração é configurada no arquivo `application.properties`, onde definimos a URL, usuário e senha. A partir daí, o Spring Data JPA e o Hibernate abstraem toda a complexidade, permitindo que nosso código Java manipule objetos de forma natural, enquanto eles cuidam da tradução para operações SQL no banco de dados Oracle.

**Link vídeo demonstração**

<https://youtu.be/abaXL1tRnIs>