

# SOLUTION NET

BEATRIZ FERNANDES – RM 95246

GABRIEL ALJARILA – RM 94187

GABRIEL SILVA – RM 93352

GUILHERME MELO – RM 94510

JÉSSICA FELIX – RM 95988

MATHEUS CORREA – RM 88286

RAFAEL FERNANDES – RM 94085

# DOCUMENTAÇÃO API DO NOSSO SISTEMA

Solução proposta .....	3
TABELA DOS ENDPOINTS .....	6
DIAGRAMA DE BLOCO .....	8

## Solução proposta

O projeto em questão é um sistema abrangente de gerenciamento de feedbacks, sugestões e solicitações de usuários. Seu principal objetivo é oferecer uma plataforma eficiente para que os usuários possam enviar suas opiniões, ideias e solicitações relacionadas a um determinado produto ou serviço, a nossa primeira ideia no 1º SPRINT, é que fosse destinado apenas para prefeituras, mas queremos que nosso sistema seja universal.

Para alcançar esse objetivo, o sistema contará com uma funcionalidade de chat baseada em um modelo de linguagem avançado,

conhecido como GPT (Transformador Gerador Pré-treinado). Esse chat GPT será integrado ao sistema para fornecer suporte aos usuários e ajudá-los a obter respostas rápidas e precisas para suas dúvidas ou problemas.

E caso a ideia seja aprovada, a ideia é que uma equipe específica fique responsável pela interação (FEEDBACK, SOLICITAÇÃO, SUGESTÃO).

Todas as conversas estabelecidas através do chat GPT serão registradas e armazenadas em um banco de dados, garantindo assim um registro completo das interações entre os usuários e o sistema. Além disso, todos os feedbacks, sugestões e solicitações enviados pelos usuários também serão armazenados no banco de dados.

Com relação ao armazenamento dos dados, nós optamos por utilizar o banco de dados Oracle. Ao escolher essa solução, pretende-se garantir a integridade e a disponibilidade dos dados coletados, bem como possibilitar consultas e análises eficientes sobre as informações armazenadas.

Em resumo, o sistema proposto visa proporcionar uma plataforma completa para gerenciamento de feedbacks, sugestões e solicitações de usuários, com o auxílio de um chat GPT para suporte. Todas as conversas do chat, bem como os feedbacks, sugestões e solicitações dos usuários, serão registrados e armazenados no banco de dados Oracle, assegurando a rastreabilidade e a segurança das informações.

## TABELA DOS ENDPOINTS

Bom, já na tabela dos endpoints, nós decidimos por enquanto, armazenar todas as informações juntas ao usuário, como no exemplo a seguir, já no banco de dados:

ID	NOME	EMAIL	TELEFONE	LOGRADOURO	BARRIO	CEP	CIDADE	UF	NUMERO	COMPLEMENTO	DATA_INTERACAO	DESCRICAO	TIPO	PRIORIDADE	EQUIPE_RESPONSAVEL
1	Lenildo Doidareu	lenildo@doidareu.com.br	1156157643	Rua das Dores	Vila do Remédio	12233222	São Paulo	SP	333	Bloco AP 121	15/05/2023	O menu do sistema poderia ser bem mais prático!	FEEDBACK	Baixa	
TELEFONE_EQUIPE_RESPONSAVEL			EMAIL_EQUIPE_RESPONSAVEL			DATA_CHATGPT			MENSAGENS_RECEBIDAS			MENSAGENS_ENVIADAS			ativo
						13/02/2023			Olá, em que posso ajudar? Claro! Você pode fazer u...			Olá. Quero fazer um feedback. Muito obrigado, me a...			1

6GET http://localhost:8080/usuarios

Send

200 OK

267 ms

378 B

Body

Basic

Query

Headers

Docs

Preview

Headers

Cookies

Timeline

1- {

2- "content": [

3- {

4- "nome": "Lenildo Doidareu",

5- "email": "lenildo@doidareu.com.br"

6- },

7- ],

8- "pageable": {

9- "sort": {

10- "sorted": true,

11- "unsorted": false,

12- "empty": false

13- },

14- "pageNumber": 0,

15- "pageSize": 0,

16- "offset": 0,

17- "unpaged": false,

18- "paged": true

19- },

20- "totalElements": 1,

21- "totalPages": 1,

22- "last": true,

23- "numberOfElements": 1,

24- "size": 0,

25- "number": 0,

26- "sort": {

27- "sorted": true,

28- "unsorted": false,

29- "empty": false

30- },

31- "first": true,

32- "empty": false

33- }

Enter a URL and send to get a response

Select a body type from above to send data in the body of a request

Por enquanto, é só uma ideia, poderemos mudar no futuro, mas basicamente o endpoint funcionaria da seguinte forma:

## Usuário

GET /usuarios: Retorna uma lista de todos os usuários.

GET /usuarios/{id}: Retorna os dados do usuário com o ID especificado.

POST /usuarios: Cria um novo usuário com os dados fornecidos no corpo da solicitação.

PUT /usuarios/{id}: Atualiza os dados do usuário com o ID especificado com os dados fornecidos no corpo da solicitação.

DELETE /usuarios/{id}: Exclui o usuário com o ID especificado.

## DIAGRAMA DE BLOCO

