

ShopHero: uma plataforma online de shopping para parceiros

Aline Bini

Ana Livia Franco

Ana Pris

João Squinelato

Marcelo Pena

Thais Carvalho



Conteúdo

01

Proposta

02

Requisitos da plataforma

03

Wireframe

04

Modelo E-R

05

Modelo de agregação

06

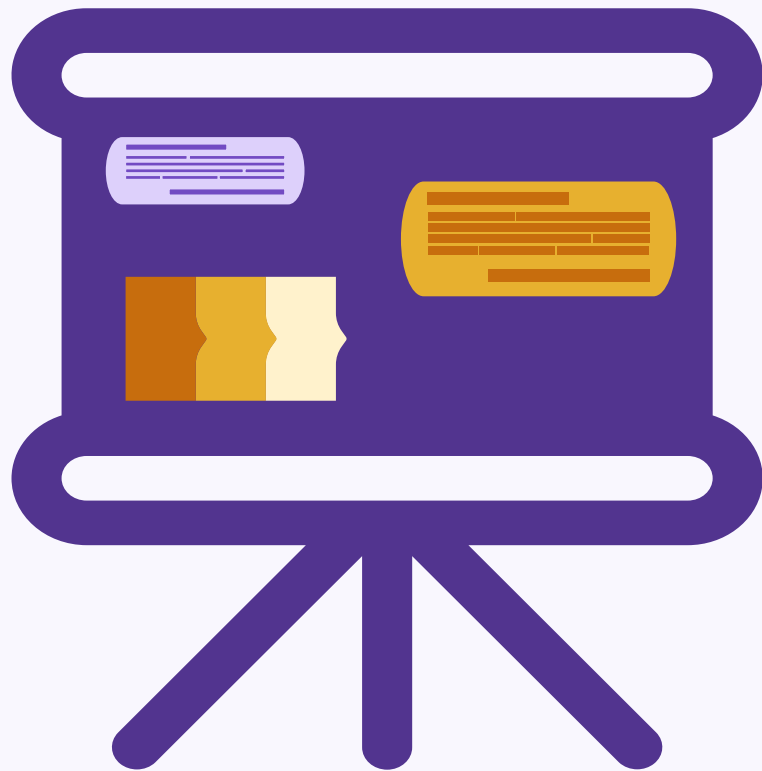
Modelo lógico

07

Modelo físico

08

Consultas



01

Proposta



Plataforma online de shopping

- Venda de eletrodomésticos e eletroeletrônicos;
- Empresas parceiras promovem ofertas na plataforma;
- Clientes encontrarão ofertas de diversas lojas em um único ambiente;
- Economia de tempo na busca por produtos.

Armazenamento e gerenciamento dos dados

- NoSQL;
- Amazon DynamoDB.

Ambiente de alta performance

- Conexão com plataforma dos parceiros em *near-real-time*;
- Altamente escalável;
- Alta disponibilidade;
- Baixa latência;
- Garantia de integração eficiente com as plataformas dos parceiros;
- Excelência na experiência do cliente.

Integração ShopHero com parceiros

- A plataforma ShopHero é como um hub;
- Recebe os dados dos clientes e repassa aos parceiros;
- Recebe os dados dos parceiros e seus produtos e exibe aos clientes.



02

Requisitos da plataforma

Requisitos funcionais



1 - Endereços do cliente;



2 - Cadastro de empresas parceiras;



3 - Agrupamento de produtos;



4 - Gestão de estoque;



5 - Logística de envio;



6 - Comentários e avaliações;



7 - Listas de desejo;



8 - Página de recomendações.



Requisitos não funcionais



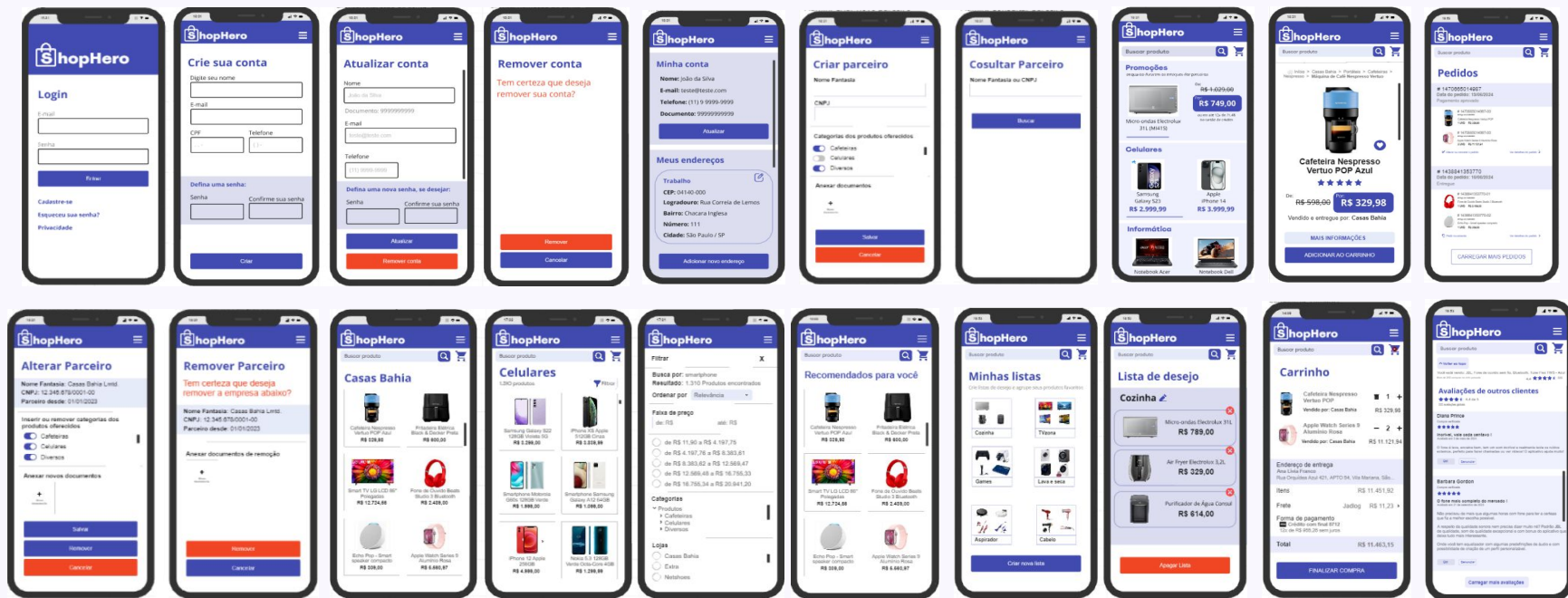
1 - Escalabilidade, baixa latência e consistência.



03

Wireframes

Protótipos mapeados para a plataforma

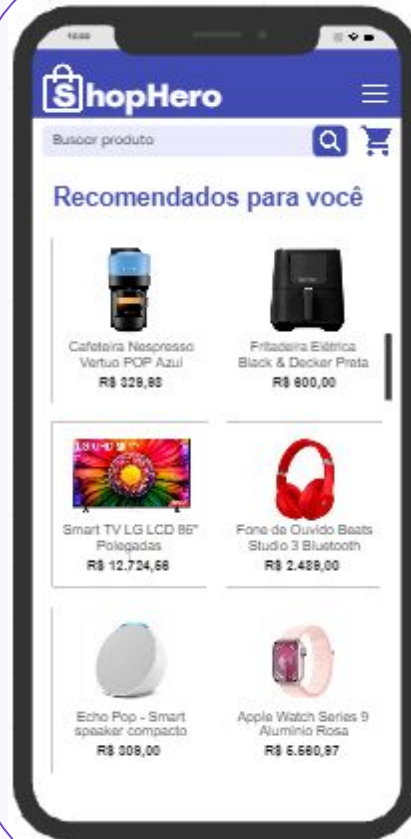


A funcionalidade e experiência garantida por cada protótipo pode ser encontrada em: <https://github.com/Squinelato/eEDB-016-2024-2>



Visão Cliente

Garante que o cliente possa consultar suas informações, bem como seus endereços cadastrados.



Visão Recomendação

Garante a consulta sobre quais produtos devem ser recomendados a cada cliente, enquanto o mesmo está ativo na plataforma.



Visão Lista de Desejos

Permite consultar as listas de desejo e os produtos atrelados a elas para cada cliente.



Visão Carrinho

Garante que o cliente confira seu carrinho de compras atual, bem como os itens nele contidos.



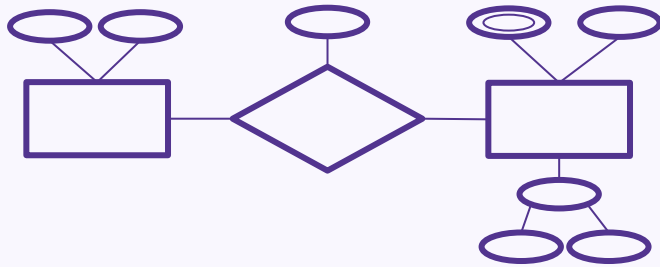
Visão Pedidos

Permite que os pedidos sejam consultados, assim como os itens adquiridos e os respectivos endereços de entrega.



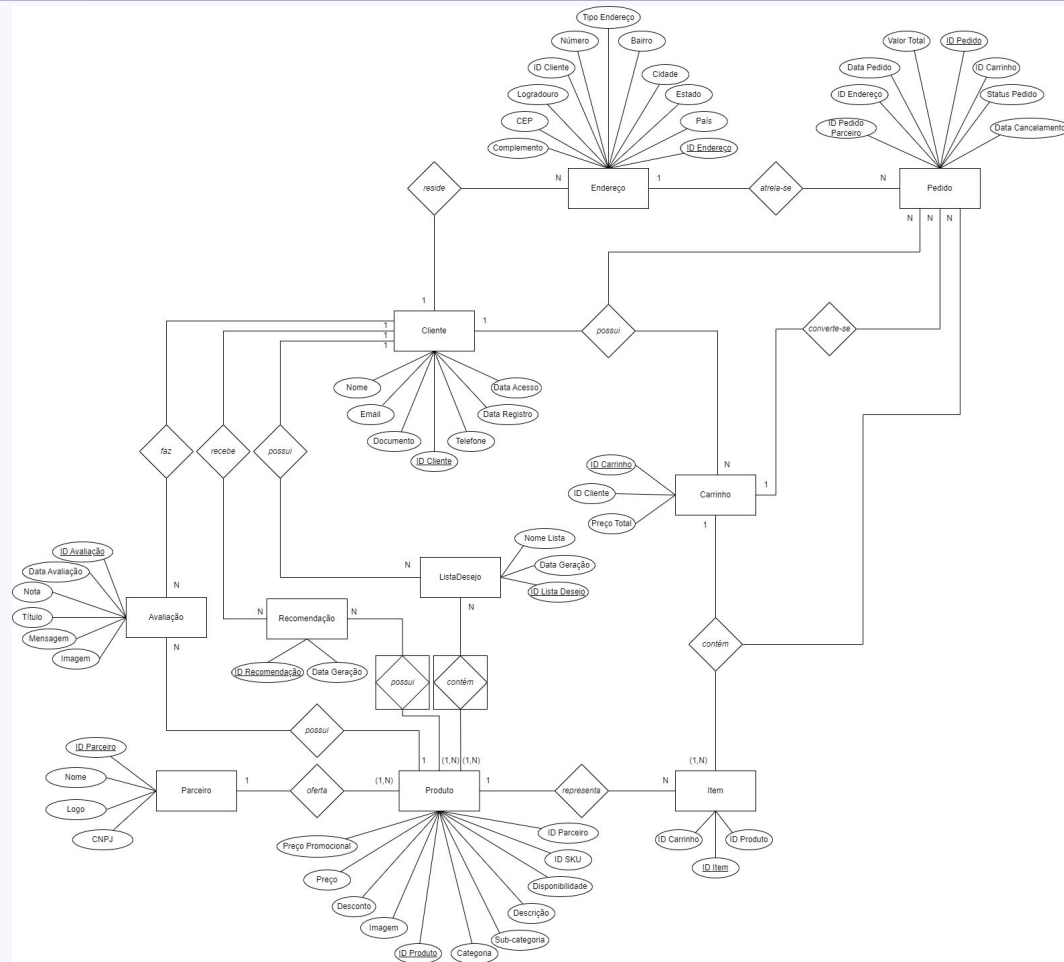
Visão Produto

Garante a consulta de detalhes sobre os produtos ofertados, bem como seus respectivos parceiros e a avaliação dos clientes sobre eles.



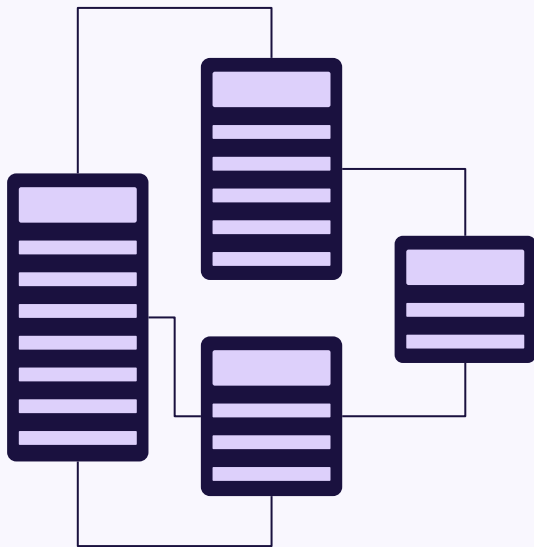
04

Modelo Entidade Relacionamento



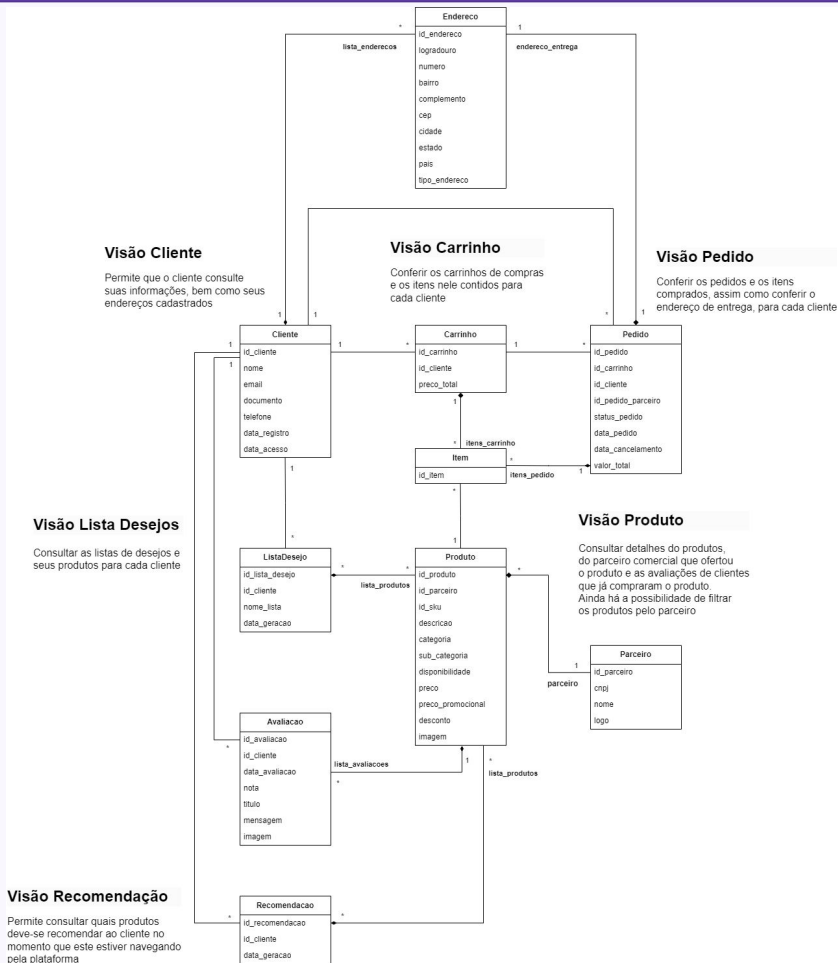
Entidades:

Cliente
 Endereço
 Carrinho
 Pedido
 Item
 Produto
 Avaliação
 ListaDesejo
 Recomendação
 Parceiro



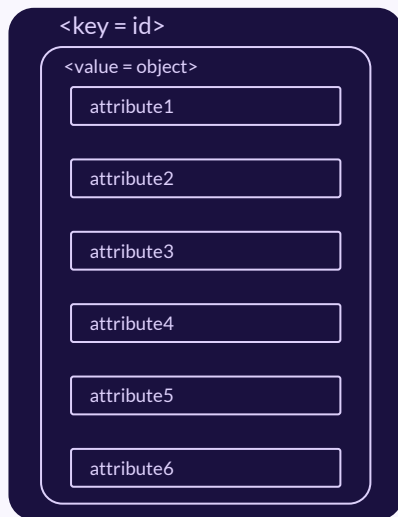
05

Modelo de Agregação



Visões:

Cliente
Carrinho
Pedido
Produto
ListaDesejo
Recomendação



06

Modelo lógico

Tabela Carrinho

<Key = id_carrinho>

<Value = Object>

id_cliente (GSI)

preco_total

itens_carrinho (agregado)

Tabela ListaDesejo

<Key = id_lista_desejo>

<Value = Object>

id_cliente (GSI)

nome_lista

data_geracao

lista_produtos (agregado)

Tabela Recomendacao

<Key = id_recomendacao>

<Value = Object>

id_cliente (GSI)

data_geracao

lista_produtos (agregado)

Tabela Pedido

<Key = id_pedido>

<Value = Object>

id_carrinho (GSI)

id_cliente (GSI)

id_pedido_parceiro

status_pedido

data_pedido

data_cancelamento

valor_total

itens_pedido (agregado)

endereco_entrega (agregado)

Tabela Cliente

<Key = id_cliente>

<Value = Object>

nome

email

documento

telefone

data_registro

data_acesso

lista_enderecos (agregado)

Tabela Produto

<Key = id_produto>

<Value = Object>

id_parceiro (GSI)

id_sku

descricao

categoria

sub_categoria

disponibilidade

preco

preco_promocional

desconto

imagem

parceiro (agregado)

lista_avaliacoes (agregado)

Tabelas:

Carrinho

ListaDesejo

Recomendacao

Pedido

Cliente

Produto

Tabela ListaDesejo

<Key = id_lista_desejo>

<Value = Object>

id_cliente (GSI)

nome_lista

data_geracao

lista_produtos (agregado)

```
1 {  
2   "id_lista_desejo": "",  
3   "id_cliente": "",  
4   "nome_lista": "",  
5   "data_geracao": "",  
6   "produtos": [  
7     {  
8       "id_produto": "",  
9       "descricao": ""  
10    }  
11  ]  
12 }
```



07

Modelo físico



github.com/Squinelato/eEDB-016-2024-2

```
35 TabelaListaDesejo:
36   Type: AWS::DynamoDB::Table
37   Properties:
38     TableName: ListaDesejo
39     AttributeDefinitions:
40       - AttributeName: "id_lista_desejo"
41         AttributeType: "S"
42       - AttributeName: "id_cliente"
43         AttributeType: "S"
44       - AttributeName: "data_geracao"
45         AttributeType: "S"
46     KeySchema:
47       - AttributeName: "id_lista_desejo"
48         KeyType: "HASH"
49     ProvisionedThroughput:
50       ReadCapacityUnits: 5
51       WriteCapacityUnits: 5
52     GlobalSecondaryIndexes:
53       - IndexName: "gsi_cliente"
54         KeySchema:
55           - AttributeName: "id_cliente"
56             KeyType: "HASH"
57           - AttributeName: "data_geracao"
58             KeyType: "RANGE"
59         Projection:
60           ProjectionType: "ALL"
61         ProvisionedThroughput:
62           ReadCapacityUnits: 5
63           WriteCapacityUnits: 5
64     Tags:
65       - Key: "Projeto"
66         Value: "Repositorio de Dados e NoSQL"
67
```

AWS CloudFormation

Infraestrutura como código (IaC)

YAML ou JSON

Recursos e propriedades

```

1  [
2  {
3      "id_lista_desejo": "ld001",
4      "id_cliente": "c001",
5      "nome_lista": "Gadgets para o home office",
6      "data_geracao": "2024-06-27",
7      "produtos": [
8          {
9              "id_produto": "pr002",
10             "descricao": "Notebook Dell XPS"
11         },
12         {
13             "id_produto": "pr003",
14             "descricao": "Mouse Gamer Razer DeathAdder"
15         },
16         {
17             "id_produto": "pr004",
18             "descricao": "Teclado mecânico Corsair K95 RGB Platinum"
19         }
20     ]
21 },
22 {
23     "id_lista_desejo": "ld002",
24     "id_cliente": "c002",
25     "nome_lista": "Upgrade para o setup de trabalho",
26     "data_geracao": "2024-06-27",
27     "produtos": [
28         {
29             "id_produto": "pr001",
30             "descricao": "Monitor LG UltraWide 29\""
31         },
32         {
33             "id_produto": "pr005",
34             "descricao": "Headset Gamer HyperX Cloud Alpha"
35         }
36     ]
37 },
38 {
39     "id_lista_desejo": "ld003",

```

Dados

ChatGPT

Ajustes manuais


```

9  def insert(table_name: str, file_path: str) -> NoReturn:
10     """
11     Insert a list of JSON items into a DynamoDB table using the batch
12     write operator in chunks of 25 items each.
13
14     Keyword arguments:
15     table_name -- The name of the DynamoDB table
16     file_path -- the path to the file containing a JSON list of items
17     """
18     print(f"Creating DynamoDB client")
19     dynamodb = boto3.resource("dynamodb", region_name="us-east-1")
20     table = dynamodb.Table(table_name)
21
22     print(f"Opening file '{file_path}'")
23     with open(file_path, 'r', encoding='utf-8') as file:
24
25         json_list = json.load(file, parse_float=Decimal)
26
27         batches = batched(json_list, 25)
28         print("Splitting items into batches of 25 items")
29
30         for index, batch in enumerate(batches):
31
32             print(f"Inserting {index+1} batch with {len(batch)} items")
33
34             with table.batch_writer() as batch_writer:
35
36                 print(f"Inserting items into '{table_name}' table: ", end="")
37                 for item in batch:
38                     print(".", end="")
39                     batch_writer.put_item(Item=item)
40
41     print(f"\nItems inserted successfully!")

```

Insert

Cloud9

Código Python

Tables (6) [Info](#)

Actions ▾

Delete

Create table

Any tag key ▾

Any tag value ▾

< 1 >



<input type="checkbox"/>	Name ▲	Status	Partition key	Sort key	Indexes	Deletion protection	Read capacity mode	Write capacity mode
<input type="checkbox"/>	Carrinho	✓ Active	id_carrinho (S)	-	1	⊖ Off	Provisioned (5)	Provisioned (5)
<input type="checkbox"/>	Cliente	✓ Active	id_cliente (S)	-	0	⊖ Off	Provisioned (5)	Provisioned (5)
<input type="checkbox"/>	ListaDesejo	✓ Active	id_lista_desejo (S)	-	1	⊖ Off	Provisioned (5)	Provisioned (5)
<input type="checkbox"/>	Pedido	✓ Active	id_pedido (S)	-	2	⊖ Off	Provisioned (5)	Provisioned (5)
<input type="checkbox"/>	Produto	✓ Active	id_produto (S)	-	1	⊖ Off	Provisioned (5)	Provisioned (5)
<input type="checkbox"/>	Recomendacao	✓ Active	id_recomendacao (S)	-	1	⊖ Off	Provisioned (5)	Provisioned (5)

> ListaDesejo



Actions ▾

Explore table items

Overview

Indexes

Monitor

Global tables

Backups

Exports and streams

Permissions - new

Additional settings

General information [Info](#)

Partition key
id_lista_desejo (String)

Sort key
-

Capacity mode
Provisioned

Table status
✔ Active

Alarms
✔ No active alarms

Point-in-time recovery (PITR) [Info](#)
⊖ Off

Resource-based policy [Info](#)
⊖ Not active

▼ Additional info

Table class
[DynamoDB Standard](#)

Indexes
1 global, 0 locals

DynamoDB stream
⊖ Off

Time to Live (TTL) [Info](#)
⊖ Off

Replication Regions
0 Regions

Encryption
Owned by Amazon

Date created
June 28, 2024, 19:01:23 (UTC-03:00)

Deletion protection
⊖ Off

Integrations [Info](#)
0

Amazon Resource Name (ARN)

arn:aws:dynamodb:us-east-1:389957581042:table/ListaDesejo



08

Consultas

ListaDesejo

Autopreview

View table details

▼ Scan or query items

☐ Scan

☒ Query

Select a table or index

Index - gsi_cliente

Select attribute projection

Projected attributes

id_cliente (Partition key)

c002

data_geracao (Sort key)

Between

2024-04-01

☒ Sort descending

and

2024-07-01

► Filters

Run

Reset

Items returned (2)

↺

Actions ▼

Create item

< 1 > ⚙

<input type="checkbox"/>	id_lista_desejo (String) ▼	data_geracao ▼	id_cliente ▼	nome_lista ▼	produtos ▼
<input type="checkbox"/>	Id006	2024-06-28	c002	Presente para meu sobrinho	[{ "M": { "id_produto": { "S": "pr003" }, "descricao": { "S": "Mouse Gamer Razer DeathAdder" } ...
<input type="checkbox"/>	Id002	2024-06-27	c002	Upgrade para o setup de trabalho	[{ "M": { "id_produto": { "S": "pr001" }, "descricao": { "S": "Monitor LG UltraWide 29\"

1. Recupera todas as listas de desejos dos últimos três meses de um cliente específico, ordenando-as por suas respectivas datas de geração

Recomendacao

Autopreview

View table details

▼ Scan or query items

☐ Scan

☒ Query

Select a table or index

Index - gsi_cliente

Select attribute projection

Projected attributes

id_cliente (Partition key)

c001

data_geracao (Sort key)

Equal to

Enter sort key value

☒ Sort descending

► Filters

Run

Reset

Items returned (2)



Actions ▼

Create item


< 1 > ⚙️

<input type="checkbox"/>	id_recomendacao (String) ▼	data_geracao ▼	id_cliente ▼	lista_produtos ▼
<input type="checkbox"/>	rec006	2024-06-28	c001	[{ "M": { "id_produto": { "S": "pr001" }, "descricao": { "S": "Monitor LG UltraWide 29\"
<input type="checkbox"/>	rec001	2024-06-27	c001	[{ "M": { "id_produto": { "S": "pr002" }, "descricao": { "S": "Notebook Dell XPS" } }]

2. Recupera todas as recomendações de produtos geradas para um cliente específico, ordenando-as por suas respectivas datas de geração

3. Permite que um cliente visualize o conteúdo do seu carrinho de compras atual

Carrinho

 [View table details](#)

▼ Scan or query items

☐ Scan

☒ Query

Select a table or index

Table - Carrinho

Select attribute projection

All attributes

id_carrinho (Partition key)

cr004

► Filters

Run

Reset

Completed. Read capacity units consumed: 0.5

Items returned (1)

↺

Actions ▼

Create item

< 1 >

⊗

⌕

<input type="checkbox"/>	id_carrinho (String)	id_cliente	itens_carrinho
<input type="checkbox"/>	cr004	c004	[{"M":{"id_item":{"S":"ci005"},"produto":{"L":[{"M":{"preco":{"N":"1100"},"id_sku":{"S":"sku001"},"disponibilida

Attributes

☒ View DynamoDB JSON

```
1 {
2   "id_carrinho": "cr004",
3   "id_cliente": "c004",
4   "itens_carrinho": [
5     {
6       "id_item": "ci005",
7       "produto": [
8         {
9           "categoria": "Tecnologia",
10          "desconto": 100,
11          "descricao": "Monitor LG Ultrawide 29\"",
12          "disponibilidade": "em estoque",
13          "id_parceiro": "pp001",
14          "id_produto": "pr001",
15          "id_sku": "sku001",
16          "imagem": "url_imagem_monitor",
17          "parceiro": {
18            "cnpj": "12.345.678/0001-09",
19            "id_parceiro": "pp001",
20            "logo": "url_logo_techmania",
21            "nome": "Techmania"
22          },
23          "preco": 1100,
24          "preco_promocional": 1000,
25          "sub_categoria": "Monitores"
26        }
27      ]
28    }
29  ],
30  "preco_total": 1000
31 }
```

JSON


Ln 2, Col 25

⊗ Errors: 0

⚠ Warnings: 0

4. Permite que o cliente visualize suas informações pessoais, incluindo detalhes de contato e endereços associados

Cliente

 [View table details](#)

☐ Scan

☒ Query

Select a table or index

Table - Cliente

Select attribute projection

All attributes

id_cliente (Partition key)


c005

Filters

Run

Reset



Items returned (1)



Actions

Create item

< 1 >

<input type="checkbox"/>	id_cliente (Str...	data_acesso	data_registro	docume...	email	lista_enderecos	nome	telefone
<input type="checkbox"/>	c005	2024-06-27	2023-08-05	345.678.90...	ana.rodrigu...	[{"M": {"tipo_end...	Ana Rodrig...	(41) 98765-4321

Attributes  View DynamoDB JSON

```
1 {
2   "id_cliente": "c005",
3   "data_acesso": "2024-06-27",
4   "data_registro": "2023-08-05",
5   "documento": "345.678.901-00",
6   "email": "ana.rodrigues@email.com",
7   "lista_enderecos": [
8     {
9       "bairro": "Água Verde",
10      "cep": "80240-210",
11      "cidade": "Curitiba",
12      "complemento": "Bloco C, apto 401",
13      "estado": "PR",
14      "id_endereco": "e009",
15      "logradouro": "Av. República Argentina",
16      "numero": "2500",
17      "pais": "Brasil",
18      "tipo_endereco": "residencial"
19    },
20    {
21      "bairro": "Centro",
22      "cep": "65432-109",
23      "cidade": "Curitiba",
24      "complemento": "Loja 10",
25      "estado": "PR",
26      "id_endereco": "e010",
27      "logradouro": "Rua do Comércio",
28      "numero": "150",
29      "pais": "Brasil",
30      "tipo_endereco": "comercial"
31    }
32  ],
33   "nome": "Ana Rodrigues",
34   "telefone": "(41) 98765-4321"
35 }
```

JSON

Ln 16, Col 21

 Errors: 0

 Warnings: 0



Produto

Autopreview

View table details

▼ Scan or query items

☐ Scan

☒ Query

Select a table or index

Index - gsi_parceiro

Select attribute projection

Projected attributes

id_parceiro (Partition key)

pp003

preco (Sort key)

Equal to

Enter sort key value

☐ Sort descending

► Filters

Run

Reset

Items returned (2)



Actions ▼

Create item




1



<input type="checkbox"/>	id_produto (String)		lista_avaliacoes	parceiro	preco	preco_promocional	sub_categoria
<input type="checkbox"/>	pr003	gem...	[]	{ "logo": { "...	350	300	Periféricos
<input type="checkbox"/>	pr006	gem...	[]	{ "logo": { "...	375	350	Periféricos

Pedido

 Autopreview

View table details

▼ Scan or query items

☐ Scan

☒ Query

Select a table or index

Index - gsi_carrinho

Select attribute projection

Projected attributes

id_carrinho (Partition key)

cr001

► Filters

Run

Reset

✔ Completed. Read capacity units consumed: 0.5

Items returned (2)

↺

Actions ▼

Create item

< 1 >

🔍

🔗

<input type="checkbox"/>	id_pedido (String) ▼	endereco_entrega ▼	id_carrinho ▼	id_cliente ▼	id_pedido_parceiro ▼	itens_pedido ▼	status_pedido ▼	valor_total ▼
<input type="checkbox"/>	p001-1	{ "tipo_endereco": { ...	cr001	c001	pp001	[{ "M": { "id_sk...	pagamento con...	1009.09
<input type="checkbox"/>	p001-2	{ "tipo_endereco": { ...	cr001	c001	pp001	[{ "M": { "id_sk...	pagamento con...	314.29

6. Recupera todos os pedidos gerados a partir do fechamento de um carrinho específico, onde os produtos comprados podem ser de diferentes parceiros



Obrigado