

Sprint 2 - Java - Evotech

Versão: 2.0

Link do repositório:

https://github.com/jvs4nt/SPRINT1_JAVA_EVOTECH

Link do vídeo de apresentação:

<https://www.youtube.com/watch?v=2ZsDtepJTEI>

Integrantes da equipe:

RM554328 - João Vitor de Santana dos Santos

- Desenvolvimento do backend
- Versionamento do projeto

RM552671 - Willian Daniel Oliveira Dantas

- Desenvolvimento do backend
- Estruturação dos endpoints

RM554021 - Sara Ingrid da Silva

- Ajustes na modelagem das entidades
- Liderança da entrega
- Desenvolvimento do backend

Cronograma de desenvolvimento:

Fase	Duração	Atividades
Planejamento	2 horas	Definição de novos requisitos, arquitetura e design do sistema

Configuração do Ambiente (se necessário)	20 minutos	Instalação e configuração do Java, Spring Framework, e ferramentas de desenvolvimento
Desenvolvimento das novas entidades e implementação do HATEOAS	6 horas	Implementação de modelos, repositórios, serviços e controladores REST
Testes	1 hora	Testes unitários, de integração e de sistema
Revisão e Refinamento	1 hora	Correção de bugs, otimização de desempenho e melhorias de código
Documentação	30 minutos	Atualização da documentação técnica e do usuário

Objetivo da solução:

O objetivo desta solução é desenvolver um sistema robusto e eficiente para a Odontoprev, uma empresa líder no setor de planos odontológicos. Este sistema visa otimizar e modernizar os processos de gestão de clientes, atendimentos, tratamentos e rede credenciada, proporcionando uma experiência aprimorada tanto para os usuários internos quanto para os clientes da Odontoprev.

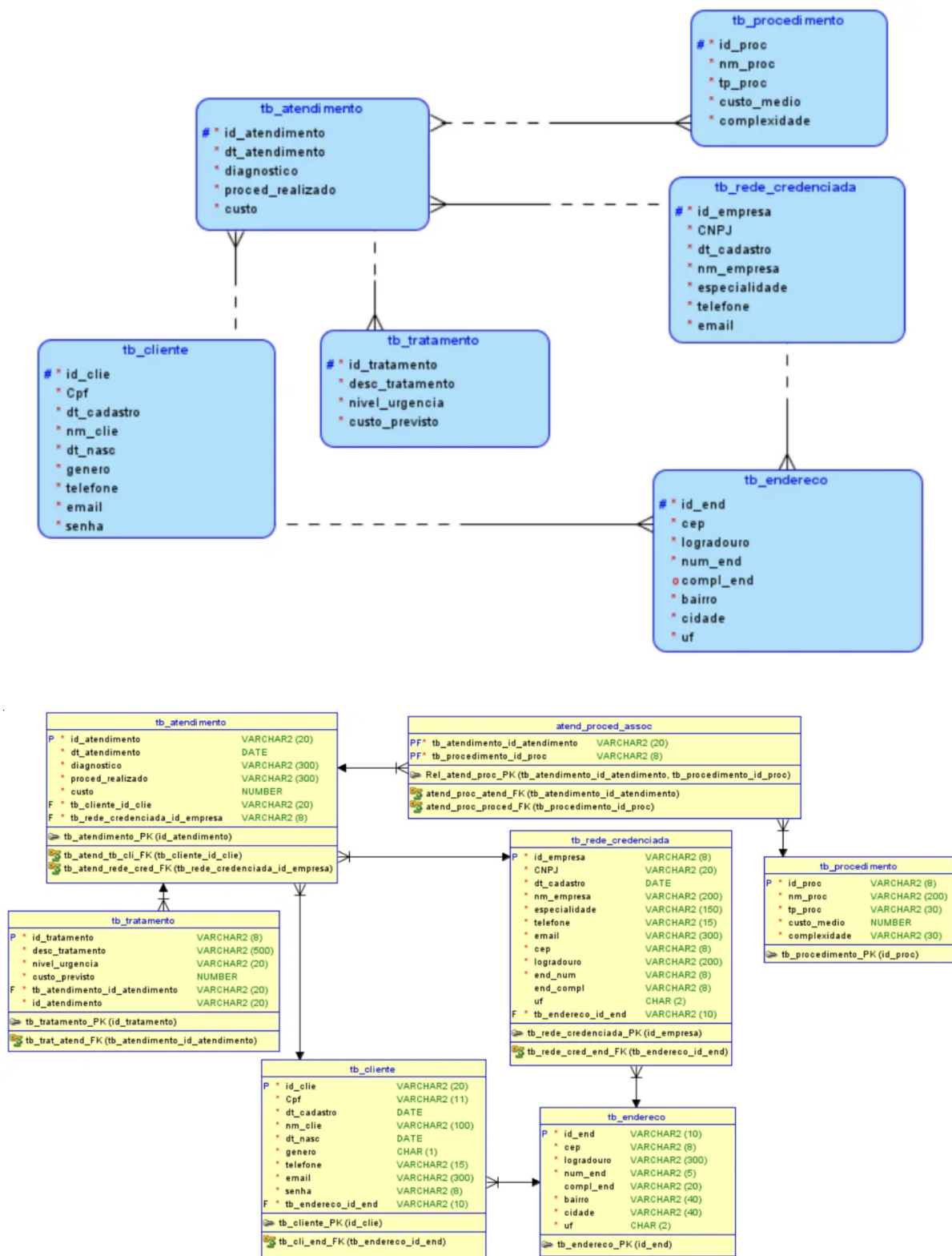
Principais aspectos da solução:

- Gameificação no processo de consultas dos clientes, incentivando-os à utilizar os serviços cada vez mais.
- Controle eficiente de atendimentos, facilitando o agendamento, acompanhamento e histórico de consultas e procedimentos realizados.
- Cadastro e gestão de tratamentos odontológicos, possibilitando uma visão clara dos serviços oferecidos e sua execução.
- Administração da rede credenciada, otimizando a relação entre a Odontoprev e os profissionais e clínicas parceiras.
- Implementação de uma API RESTful com HATEOAS, garantindo uma interface de programação moderna, flexível e de fácil integração com outros sistemas.

Com esta solução, a Odontoprev poderá melhorar significativamente a qualidade de seus serviços, aumentar a eficiência operacional e proporcionar uma experiência mais satisfatória para seus clientes e parceiros, consolidando ainda mais sua posição de liderança no mercado de planos odontológicos e

fazendo com que utilizem os serviços com mais frequência, o que casa perfeitamente com o desafio.

Diagrama de Classes:



Evidências dos testes realizados:

OBS. Sinta-se livre para testar os endpoints, todos seguem o mesmo padrão então basta mudar a entidade passada na URL e se atentar aos campos necessários

Endpoints

Cliente

- **GET** `/clientes` : Retorna todos os clientes.
- **GET** `/clientes/{id}` : Retorna um cliente pelo ID.
- **POST** `/clientes` : Cria um novo cliente.
- **PUT** `/clientes/{id}` : Atualiza um cliente existente.
- **DELETE** `/clientes/{id}` : Deleta um cliente pelo ID.

Atendimento

- **GET** `/atendimentos` : Retorna todos os atendimentos.
- **GET** `/atendimentos/{id}` : Retorna um atendimento pelo ID.
- **POST** `/atendimentos` : Cria um novo atendimento.
- **PUT** `/atendimentos/{id}` : Atualiza um atendimento existente.
- **DELETE** `/atendimentos/{id}` : Deleta um atendimento pelo ID.

Tratamento

- **GET** `/tratamentos` : Retorna todos os tratamentos.
- **GET** `/tratamentos/{id}` : Retorna um tratamento pelo ID.
- **POST** `/tratamentos` : Cria um novo tratamento.
- **PUT** `/tratamentos/{id}` : Atualiza um tratamento existente.
- **DELETE** `/tratamentos/{id}` : Deleta um tratamento pelo ID.

Rede Credenciada

- **GET** `/rede-credenciada` : Retorna todos os tratamentos.
- **GET** `/rede-credenciada/{id}` : Retorna um tratamento pelo ID.
- **POST** `/rede-credenciada` : Cria um novo tratamento.
- **PUT** `/rede-credenciada/{id}` : Atualiza um tratamento existente.
- **DELETE** `/rede-credenciada/{id}` : Deleta um tratamento pelo ID.

Demonstração da API com links do HATEOAS:

Cientes:

The screenshot shows a REST client interface with the following details:

- Method:** POST
- URL:** http://localhost:8080/clientes
- Status:** 201 Created
- Time:** 20.3 s
- Timestamp:** Just Now
- Body (JSON):**

```
1 {
2   "endereco": {
3     "idEndereco": null,
4     "logradouro": "Rua teste",
5     "numero": "123",
6     "complemento": "complemento teste",
7     "bairro": "bairro teste",
8     "cidade": "cidade teste",
9     "estado": "SP",
10    "cep": "09876-321"
11  },
12  "nomeCompleto": "João teste",
13  "email": "joao@teste",
14  "telefone": "11912345678"
15 }
```
- Preview (JSON):**

```
8   "bairro": "bairro teste",
9   "cidade": "cidade teste",
10  "estado": "SP",
11  "cep": "09876-321"
12 },
13 "nomeCompleto": "João teste",
14 "email": "joao@teste",
15 "telefone": "11912345678",
16 "atendimentos": null,
17 "_links": {
18   "self": {
19     "href":
20       "http://localhost:8080/clientes/75fdb4f1-2243-43c9-8576-b9d9afd3f666"
21   },
22   "allClientes": {
23     "href":
24       "http://localhost:8080/clientes"
25   }
26 }
```
- Footer:** Beautify JSON, \$.store.books[*].author

GET ▼ http://localhost:8080/clientes/75fdb4f1-2243-43c9-8576-b9d9a Send 200 OK 51 ms 517 Just Now ▼

Params Body ● Auth Headers 4 Scripts Docs

URL PREVIEW
http://localhost:8080/clientes/75fdb4f1-2243-43c9-8576-b9d9afd3f666 📋

QUERY PARAMETERS Import from URL Bulk Edit

+ Add 🗑️ Delete all 👁 Description

name	value
------	-------

PATH PARAMETERS

❗ Path parameters are url path segments that start with a colon ':' e.g. ':id' ×

Preview ▼

```
1 {
2   "idCliente": "75fdb4f1-2243-43c9-8576-b9d9afd3f666",
3   "endereco": {
4     "idEndereco": "d3044ae1-0ae2-4d26-aff9-404a457a260e",
5     "logradouro": "Rua teste",
6     "numero": "123",
7     "complemento": "complemento teste",
8     "bairro": "bairro teste",
9     "cidade": "cidade teste",
10    "estado": "SP",
11    "cep": "09876-321"
12  },
13  "nomeCompleto": "João teste",
14  "email": "joao@teste",
15  "telefone": "11912345678",
16  "atendimentos": [],
17  "_links": {
18    "self": {
```

\$.store.books[*].author ?

GET ▼ http://localhost:8080/clientes Send 200 OK 224 ms 609 B Just Now ▼

Params Body ● Auth Headers 4 Scripts Preview Headers 3 Cookies Tests 0/0 → Mock Console

URL PREVIEW
http://localhost:8080/clientes 📋

QUERY PARAMETERS Import from URL Bulk Edit

+ Add 🗑️ Delete all 👁 Description

name	value
------	-------

PATH PARAMETERS

❗ Path parameters are url path segments that start with a colon ':' e.g. ':id' ×

Preview ▼

```
1 {
2   "_embedded": {
3     "clienteList": [
4       {
5         "idCliente": "75fdb4f1-2243-43c9-8576-b9d9afd3f666",
6         "endereco": {
7           "idEndereco": "d3044ae1-0ae2-4d26-aff9-404a457a260e",
8           "logradouro": "Rua teste",
9           "numero": "123",
10          "complemento": "complemento teste",
11          "bairro": "bairro teste",
12          "cidade": "cidade teste",
13          "estado": "SP",
14          "cep": "09876-321"
15        },
16        "nomeCompleto": "João teste",
17        "email": "joao@teste",
18        "telefone": "11912345678",
19        "atendimentos": [],
```

\$.store.books[*].author ?

Tratamentos:

POST http://localhost:8080/tratamentos Send 201 Created 343 ms 248 B Just Now

Params Body Auth Headers 4 Scripts Preview Headers 3 Cookies Tests 0/0 → Mock Console

JSON

```

1 {
2   "descricao": "Canal estágio I"
3 }

```

Preview

```

1 {
2   "idTratamento": "324758fd-dd2a-443f-a924-abd7b08e2a5e",
3   "descricao": "Canal estágio I",
4   "_links": {
5     "self": {
6       "href": "http://localhost:8080/tratamentos/324758fd-dd2a-443f-a924-abd7b08e2a5e"
7     },
8     "allTratamentos": {
9       "href": "http://localhost:8080/tratamentos"
10    }
11  }
12 }

```

Beautify JSON \$.store.books[*].author

GET http://localhost:8080/tratamentos/324758fd-dd2a-443f-a924-abd7b08e2a5e Send 200 OK 112 ms 248 B Just Now

Params Body Auth Headers 4 Scripts Preview Headers 3 Cookies Tests 0/0 → Mock Console

URL PREVIEW
http://localhost:8080/tratamentos/324758fd-dd2a-443f-a924-abd7b08e2a5e

QUERY PARAMETERS Import from URL Bulk Edit

+ Add Delete all Description

name	value

PATH PARAMETERS

Path parameters are url path segments that start with a colon ':' e.g. ':id'

Preview

```

1 {
2   "idTratamento": "324758fd-dd2a-443f-a924-abd7b08e2a5e",
3   "descricao": "Canal estágio I",
4   "_links": {
5     "self": {
6       "href": "http://localhost:8080/tratamentos/324758fd-dd2a-443f-a924-abd7b08e2a5e"
7     },
8     "allTratamentos": {
9       "href": "http://localhost:8080/tratamentos"
10    }
11  }
12 }

```

\$.store.books[*].author

GET ▼ http://localhost:8080/tratamentos Send 200 OK 84 ms 346 B Just Now ▼

Params Body Auth Headers 4 Preview Headers 3 Cookies Tests 0 / 0 → Mock Console

URL PREVIEW
http://localhost:8080/tratamentos

QUERY PARAMETERS Import from URL Bulk Edit
+ Add Delete all Description

name	value

PATH PARAMETERS
Path parameters are url path segments that start with a colon ':' e.g. ':id'

Preview ▼

```

1 {
2   "_embedded": {
3     "tratamentoList": [
4       {
5         "idTratamento": "324758fd-dd2a-443f-a924-abd7b08e2a5e",
6         "descricao": "Canal estágio I",
7         "_links": {
8           "self": {
9             "href": "http://localhost:8080/tratamentos/324758fd-dd2a-443f-a924-abd7b08e2a5e"
10          },
11         "allTratamentos": {
12           "href": "http://localhost:8080/tratamentos"
13         }
14       }
15     ]
16   },
17   "_links": {
18     "self": {
19       "href": "http://localhost:8080/tratamentos"
20     }
21   }
22 }

```

\$.store.books[*].author

Atendimento:

POST ▼ http://localhost:8080/atendimentos Send 201 Created 110 ms 513 Just Now ▼

Params Body Auth Headers 4 Scripts Docs Preview Headers 3 Cookies Tests 0 / 0

JSON ▼

```

1 {
2   "cliente": {
3     "idCliente": "f04b613b-7f7d-45d7-9679-e0f7ca066d8c"
4   },
5   "tratamento": {
6     "idTratamento": "fa4d52f8-2959-449c-a5c2-73ca43d0cba7"
7   },
8   "data": "2024-11-01",
9   "observacoes": "Agendar limpeza geral"
10 }

```

Preview ▼

```

10 },
11 "tratamento": {
12   "idTratamento": "fa4d52f8-2959-449c-a5c2-73ca43d0cba7",
13   "descricao": null
14 },
15 "data": "2024-11-01",
16 "observacoes": "Agendar limpeza geral",
17 "_links": {
18   "self": {
19     "href": "http://localhost:8080/atendimentos/963bdb55-8ccd-4afe-bf3d-1269f680e1e8"
20   },
21   "allAtendimentos": {
22     "href": "http://localhost:8080/atendimentos"
23   }
24 }
25 }

```

Beautify JSON

\$.store.books[*].author

GET http://localhost:8080/atendimento Send 200 OK 187 ms 139.8 KB Just Now

Params Body Auth Headers 4 Scripts Preview Headers 3 Cookies Tests 0/0 → Mock Console

URL PREVIEW
http://localhost:8080/atendimentos

QUERY PARAMETERS Import from URL Bulk Edit
+ Add Delete all Description

name	value

PATH PARAMETERS
Path parameters are url path segments that start with a colon ':' e.g. ':id'

Preview

```

1 {
2   "_embedded": {
3     "atendimentoList": [
4       {
5         "idAtendimento": "963bdb55-8ccd-4afe-bf3d-
1269f680e1e8",
6         "cliente": {
7           "idCliente": "f04b613b-7f7d-45d7-9679-
e0f7ca066d8c",
8           "endereco": {
9             "idEndereco": "c6a508e0-e4bd-48a5-956b-
6da086302912",
10            "logradouro": "rua teste",
11            "numero": "121",
12            "complemento": "complemento teste",
13            "bairro": "bairro teste",
14            "cidade": "cidade teste",
15            "estado": "SP",
16            "cep": "19102-999"
17          },
18          "nomeCompleto": "joao santos",

```

\$.store.books[*].author

Rede Credenciada:

POST http://localhost:8080/rede-credenciada Send 201 Created 237 ms 503 B Just Now

Params Body Auth Headers 4 Scripts Preview Headers 3 Cookies Tests 0/0 → Mock Console

JSON

```

1 {
2   "cnpj": "12345678000123",
3   "nmEmpresa": "Clínica Exemplo",
4   "dtCadastro": "2024-11-08",
5   "especialidade": "Odontologia",
6   "telefone": "11987654321",
7   "email": "contato@clinicaexemplo.com",
8   "cep": "12345678",
9   "logradouro": "Rua Exemplo",
10  "numero": "123",
11  "complemento": "Sala 1",
12  "uf": "SP"
13 }

```

Preview

```

1 {
2   "idEmpresa": "61e1bf78-d017-4fae-a225-6da491cd92fd",
3   "cnpj": "12345678000123",
4   "dtCadastro": "2024-11-08",
5   "nmEmpresa": "Clínica Exemplo",
6   "especialidade": "Odontologia",
7   "telefone": "11987654321",
8   "email": "contato@clinicaexemplo.com",
9   "cep": "12345678",
10  "logradouro": "Rua Exemplo",
11  "endNum": null,
12  "endCompl": null,
13  "uf": "SP",
14  "endereco": null,
15  "_links": {
16    "self": {
17      "href": "http://localhost:8080/rede-
credenciada/61e1bf78-d017-4fae-a225-6da491cd92fd"
18    },
19    "allRedesCredenciadas": {
20      "href": "http://localhost:8080/rede-credenciada"

```

Beautify JSON

\$.store.books[*].author

GET http://localhost:8080/rede-credenciada Send 200 OK 108 ms 1115 B Just Now

Params Body Auth Headers 4 Scripts Docs Preview Headers 3 Cookies Tests 0/0 → Mock Console

URL PREVIEW
http://localhost:8080/rede-credenciada

QUERY PARAMETERS Import from URL Bulk Edit
+ Add Delete all Description

name value

PATH PARAMETERS
Path parameters are url path segments that start with a colon ':' e.g. 'id'

Preview

```

1 {
2   "_embedded": {
3     "redeCredenciadaList": [
4       {
5         "idEmpresa": "07f19c18-e5af-4161-9156-
a76a0cbd3a1b",
6         "cnpj": "12345678000123",
7         "dtCadastro": "2024-11-08",
8         "nmEmpresa": "Clínica Exemplo",
9         "especialidade": "Odontologia",
10        "telefone": "11987654321",
11        "email": "contato@clinicaexemplo.com",
12        "cep": "12345678",
13        "logradouro": "Rua Exemplo",
14        "endNum": null,
15        "endCompl": null,
16        "uf": "SP",
17        "endereco": null,
18        "_links": {
19          "self": {
20            "href": "http://localhost:8080/rede-

```

\$.store.books[*].author

PUT http://localhost:8080/rede-credenciada Send 200 OK 142 ms 503 B Just Now

Params Body Auth Headers 4 Scripts Docs Preview Headers 3 Cookies Tests 0/0 → Mock Console

JSON

```

1 {
2   "cnpj": "12345678000123",
3   "nmEmpresa": "Clínica Editada",
4   "dtCadastro": "2024-11-08",
5   "especialidade": "Odontologia",
6   "telefone": "11987654321",
7   "email": "contato@clinicaexemplo.com",
8   "cep": "12345678",
9   "logradouro": "Rua Exemplo",
10  "numero": "123",
11  "complemento": "Sala 1",
12  "uf": "SP"
13 }

```

Preview

```

1 {
2   "idEmpresa": "07f19c18-e5af-4161-9156-a76a0cbd3a1b",
3   "cnpj": "12345678000123",
4   "dtCadastro": "2024-11-08",
5   "nmEmpresa": "Clínica Editada",
6   "especialidade": "Odontologia",
7   "telefone": "11987654321",
8   "email": "contato@clinicaexemplo.com",
9   "cep": "12345678",
10  "logradouro": "Rua Exemplo",
11  "endNum": null,
12  "endCompl": null,
13  "uf": "SP",
14  "endereco": null,
15  "_links": {
16    "self": {
17      "href": "http://localhost:8080/rede-
credenciada/07f19c18-e5af-4161-9156-a76a0cbd3a1b"
18    },
19    "allRedesCredenciadas": {
20      "href": "http://localhost:8080/rede-credenciada"

```

Beautify JSON

\$.store.books[*].author

Versionamento da Documentação

Versão	Data	Descrição	Autor
--------	------	-----------	-------

1.0 (Sprint 1)	07/10/2024	Versão inicial da documentação	Equipe de Desenvolvimento
2.0 (Sprint 2)	08/11/2024	Implementação da entidade Rede Credenciada, ajustes de relacionamentos através do Spring e implementação do HATEOAS	Equipe de Desenvolvimento