

1 - Roteiro de testes do UsuariosController

Localização

- Tela do Swagger da API Desafio de Arquitetura de Software

Objeto de teste

- Consultar Usuários a partir do UsuariosController método GetAllUsuarios

Caso de teste

- **Descrição:** Testar se o método GetAllUsuarios está funcionando
- **Pré-Condição:** Api Desafio finalizada
- **Procedimento:** Consultar todos os usuários da base de dados
- **Resultado esperado:** Objeto OkObjectResult

Localização

- Tela do Swagger da API Desafio de Arquitetura de Software

Objeto de teste

- Consultar Usuários a partir do UsuariosController método GetAllUsuarios

Caso de teste

- **Descrição:** Testar se o método GetAllUsuarios está funcionando
- **Pré-Condição:** Api Desafio finalizada
- **Procedimento:** Consultar todos os usuários da base de dados
- **Resultado esperado:** Status Code: 200

Localização

- Tela do Swagger da API Desafio de Arquitetura de Software

Objeto de teste

- Consultar Usuários Médicos a partir do UsuariosController método GetAllUsuariosMedicos

Caso de teste

- **Descrição:** Testar se o método GetAllUsuariosMedicos está funcionando
- **Pré-Condição:** Api Desafio finalizada
- **Procedimento:** Consultar todos os usuários que são Médicos da base de dados
- **Resultado esperado:** Status Code: 200

Localização

- Tela do Swagger da API Desafio de Arquitetura de Software

Objeto de teste

- Consultar Usuários Pacientes a partir do UsuariosController método GetAllUsuariosPacientes

Caso de teste

- **Descrição:** Testar se o método GetAllUsuariosPacientes está funcionando
- **Pré-Condição:** Api Desafio finalizada
- **Procedimento:** Consultar todos os Usuários que são Pacientes da base de dados
- **Resultado esperado:** Status Code: 200

Localização

- Tela do Swagger da API Desafio de Arquitetura de Software

Objeto de teste

- Consultar Usuários a partir do UsuariosController método GetAllUsuarios

Caso de teste

- **Descrição:** Testar se o método GetAllUsuarios está funcionando ou retorna um objeto null
- **Pré-Condição:** Api Desafio finalizada
- **Procedimento:** Consultar todos os usuários da base de dados
- **Resultado esperado:** NotNull

2 - Roteiro de testes do PacientesController

Localização

- Tela do Swagger da API Desafio de Arquitetura de Software

Objeto de teste

- Consultar Pacientes a partir do PacientesController método GetAllPacientes

Caso de teste

- **Descrição:** Testar se o método GetAllPacientes está funcionando
- **Pré-Condição:** Api Desafio finalizada
- **Procedimento:** Consultar todos os pacientes da base de dados
- **Resultado esperado:** Objeto OkObjectResult

Localização

- Tela do Swagger da API Desafio de Arquitetura de Software

Objeto de teste

- Consultar Pacientes a partir do PacientesController método GetAllPacientes

Caso de teste

- **Descrição:** Testar se o método GetAllPacientes está funcionando
- **Pré-Condição:** Api Desafio finalizada
- **Procedimento:** Consultar todos os pacientes da base de dados
- **Resultado esperado:** Status Code: 200

Localização

- Tela do Swagger da API Desafio de Arquitetura de Software

Objeto de teste

- Inserir um paciente na base de dados a partir do PacientesController método InsertPaciente

Caso de teste

- **Descrição:** Testar se o método InsertPaciente está funcionando
- **Pré-Condição:** Api Desafio finalizada
- **Procedimento:** Inserir um paciente na base de dados
- **Resultado esperado:** OkObjectResult indicando que funcionou

Localização

- Tela do Swagger da API Desafio de Arquitetura de Software

Objeto de teste

- Consultar Pacientes a partir do PacientesController método GetAllPacientes

Caso de teste

- **Descrição:** Testar se o método GetAllPacientes está funcionando ou retorna um objeto null
- **Pré-Condição:** Api Desafio finalizada
- **Procedimento:** Consultar todos os pacientes da base de dados
- **Resultado esperado:** NotNull

3 - Roteiro de testes do MedicosController

Localização

- Tela do Swagger da API Desafio de Arquitetura de Software

Objeto de teste

- Consultar Médicos a partir do MedicosController método GetAllMedicos

Caso de teste

- **Descrição:** Testar se o método GetAllMedicos está funcionando
- **Pré-Condição:** Api Desafio finalizada
- **Procedimento:** Consultar todos os médicos da base de dados
- **Resultado esperado:** Objeto OkObjectResult

Localização

- Tela do Swagger da API Desafio de Arquitetura de Software

Objeto de teste

- Consultar Médicos a partir do MedicosController método GetAllMedicos

Caso de teste

- **Descrição:** Testar se o método GetAllMedicos está funcionando
- **Pré-Condição:** Api Desafio finalizada
- **Procedimento:** Consultar todos os médicos da base de dados
- **Resultado esperado:** Status Code: 200

Localização

- Tela do Swagger da API Desafio de Arquitetura de Software

Objeto de teste

- Inserir um médico na base de dados a partir do MedicosController método InsertMedico

Caso de teste

- **Descrição:** Testar se o método InsertMedico está funcionando
- **Pré-Condição:** Api Desafio finalizada
- **Procedimento:** Inserir um médico na base de dados
- **Resultado esperado:** OkObjectResult indicando que funcionou

Localização

- Tela do Swagger da API Desafio de Arquitetura de Software

Objeto de teste

- Consultar Médicos a partir do MedicosController método GetAllMedicos

Caso de teste

- **Descrição:** Testar se o método GetAllMedicos está funcionando ou retorna um objeto null
- **Pré-Condição:** Api Desafio finalizada
- **Procedimento:** Consultar todos os médicos da base de dados
- **Resultado esperado:** NotNull

4 - Roteiro de testes do TipoUsuariosController

Localização

- Tela do Swagger da API Desafio de Arquitetura de Software

Objeto de teste

- Consultar TipoUsuario a partir do TipoUsuariosController método GetAllTipoUsuario

Caso de teste

- **Descrição:** Testar se o método GetAllTipoUsuario está funcionando
- **Pré-Condição:** Api Desafio finalizada
- **Procedimento:** Consultar todos os TipoUsuario da base de dados
- **Resultado esperado:** Objeto OkObjectResult

Localização

- Tela do Swagger da API Desafio de Arquitetura de Software

Objeto de teste

- Consultar TipoUsuario a partir do TipoUsuariosController método GetAllTipoUsuario

Caso de teste

- **Descrição:** Testar se o método GetAllTipoUsuario está funcionando
- **Pré-Condição:** Api Desafio finalizada
- **Procedimento:** Consultar todos os TipoUsuario da base de dados
- **Resultado esperado:** Status Code: 200

Localização

- Tela do Swagger da API Desafio de Arquitetura de Software

Objeto de teste

- Consultar TipoUsuario a partir do TipoUsuariosController método GetAllTipoUsuario

Caso de teste

- **Descrição:** Testar se o método GetAllTipoUsuario está funcionando ou retorna um objeto null
- **Pré-Condição:** Api Desafio finalizada
- **Procedimento:** Consultar todos os TipoUsuario da base de dados
- **Resultado esperado:** NotNull

5 - Roteiro de testes do EspecialidadesController

- Localização

- Tela do Swagger da API Desafio de Arquitetura de Software

Objeto de teste

- Consultar Especialidades a partir do EspecialidadesController método GetAllEspecialidades

Caso de teste

- **Descrição:** Testar se o método GetAllEspecialidades está funcionando
- **Pré-Condição:** Api Desafio finalizada
- **Procedimento:** Consultar todas as Especialidades da base de dados
- **Resultado esperado:** Objeto OkObjectResult

- Localização

- Tela do Swagger da API Desafio de Arquitetura de Software

Objeto de teste

- Consultar Especialidades com Médicos a partir do EspecialidadesController método GetAllEspecialidadeComMedicos

Caso de teste

- **Descrição:** Testar o método GetAllEspecialidadeComMedicos
- **Pré-Condição:** Api Desafio finalizada
- **Procedimento:** Consultar todas as Especialidades com médicos da base de dados
- **Resultado esperado:** Objeto OkObjectResult

- Localização

- Tela do Swagger da API Desafio de Arquitetura de Software

Objeto de teste

- Consultar Especialidades a partir do EspecialidadesController método GetAllEspecialidades

Caso de teste

- **Descrição:** Testar se o método GetAllEspecialidades está funcionando
- **Pré-Condição:** Api Desafio finalizada
- **Procedimento:** Consultar todos as especialidades da base de dados
- **Resultado esperado:** Status Code: 200
-

- Localização

- Tela do Swagger da API Desafio de Arquitetura de Software

Objeto de teste

- Inserir uma especialidade na base de dados a partir do EspecialidadesController método InsertEspecialidade

Caso de teste

- **Descrição:** Testar se o método InsertEspecialidade está funcionando
- **Pré-Condição:** Api Desafio finalizada
- **Procedimento:** Inserir uma especialidade na base de dados
- **Resultado esperado:** OkObjectResult indicando que funcionou

- Localização

- Tela do Swagger da API Desafio de Arquitetura de Software

Objeto de teste

- Consultar Especialidade a partir do EspecialidadesController método GetAllEspecialidades

Caso de teste

- **Descrição:** Testar se o método GetAllEspecialidades está funcionando ou retorna um objeto null
- **Pré-Condição:** Api Desafio finalizada
- **Procedimento:** Consultar todas as especialidades da base de dados
- **Resultado esperado:** NotNull

6 - Roteiro de testes do ConsultasController

- Localização

- Tela do Swagger da API Desafio de Arquitetura de Software

Objeto de teste

- Consultar Consultas a partir do ConsultasController método GetAllConsultas

Caso de teste

- **Descrição:** Testar se o método GetAllConsultas está funcionando
- **Pré-Condição:** Api Desafio finalizada
- **Procedimento:** Consultar todas as Consultas da base de dados
- **Resultado esperado:** Objeto OkObjectResult

- Localização

- Tela do Swagger da API Desafio de Arquitetura de Software

Objeto de teste

- Consultar Consultas a partir do ConsultasController método GetAllConsultas

Caso de teste

- **Descrição:** Testar se o método GetAllConsultas está funcionando
- **Pré-Condição:** Api Desafio finalizada
- **Procedimento:** Consultar todas as consultas da base de dados
- **Resultado esperado:** Status Code: 200
-

- Localização

- Tela do Swagger da API Desafio de Arquitetura de Software

Objeto de teste

- Inserir uma consulta na base de dados a partir do ConsultasController método InsertConsulta

Caso de teste

- **Descrição:** Testar se o método InsertConsulta está funcionando
- **Pré-Condição:** Api Desafio finalizada
- **Procedimento:** Inserir uma consulta na base de dados
- **Resultado esperado:** OkObjectResult indicando que funcionou

- Localização

- Tela do Swagger da API Desafio de Arquitetura de Software

Objeto de teste

- Consultar Consultas a partir do ConsultasController método GetAllConsultas

Caso de teste

- **Descrição:** Testar se o método GetAllConsultas está funcionando ou retorna um objeto null
- **Pré-Condição:** Api Desafio finalizada
- **Procedimento:** Consultar todas as consultas da base de dados
- **Resultado esperado:** NotNull