

DESCRIÇÃO TÉCNICA DE CASOS DE USO

Rede Monitor

UCs de LOGIN

Login

Efetuar login do usuário, gerando access_token e refresh_token armazenados em cookies httponly.

Fluxo principal:

1. O sistema valida os dados de login
2. O sistema busca o usuário e os roles pelo username e senha
3. O sistema gera o access_token com o username, roles, perfil e id de empresa (Se usuário admin, ID=-1) e tempo de expiração de 5 minutos
4. O sistema gera o refresh_token com tempo de expiração de 1 hora
5. O sistema seta os cookies como access_token e refresh_token na resposta
6. O frontend recebe a resposta com o access_token e perfil e redireciona para a página de dashboard

Fluxos alternativos:

A1: Usuário não encontrado

A2: Username ou senha em branco

Logout

Responsável por efetuar logout, eliminando os cookies de access_token e refresh_token

Fluxo principal:

1. O sistema seta os cookies de access_token e refresh_token com tempo de expiração igual a zero e os retorna para o frontend
2. O frontend redireciona para página de login

Refresh Access Token

Responsável por gerar novo access_token e atualizar o cookie para que o cookie com o novo token seja enviado nas próximas requisições.

Fluxo principal:

1. O sistema recebe o cookie de refresh_token
2. O sistema verifica a validade do refresh_token
3. O sistema busca os roles e outros dados do usuário
4. O sistema gera o novo access_token com username, roles, perfil e id da empresa (Se usuário admin, ID=-1) com expiração de 5 minutos
5. O sistema seta o cookie de access_token com o novo access_token
6. O sistema retorna o novo access_token, perfil e outros dados do usuário.

UCs de USUÁRIO

Criar Usuário

Responsável por registrar um novo usuário no banco de dados

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “usuario-all”
2. O sistema verifica se o usuário já existe registrado no sistema
 - a) Se sim, retorna erro 400 informando que o usuário já existe registrado pelo username
3. O sistema verifica se o ID da empresa recebido na requisição é igual a: -1
 - a) Se sim, nenhuma empresa é registrada para o usuário
 - b) Se não, é buscada a empresa pelo ID e setada no objeto de usuário a ser registrado
4. O sistema carrega o objeto de usuário com os dados enviados na requisição
5. O sistema registra o usuário

Atualizar Usuário

Responsável por atualizar dados do usuário já registrado no banco de dados

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “usuario-all”
2. O sistema busca o usuário pelo ID recebido na requisição
3. O sistema verifica se o novo username (recebido na requisição) é igual ao username do usuário buscado
 - a) Se não, o sistema verifica se existe algum usuário registrado com o novo username
 1. Se sim, o sistema retorna uma mensagem de erro 400 informando que já existe outro usuário com o username informado
4. O sistema verifica se o ID da empresa recebido na requisição é igual a: -1
 - a) Se sim, nenhuma empresa é registrada para o usuário
 - b) Se não, é buscada a empresa pelo ID e setada no objeto de usuário a ser registrado
5. O sistema carrega o objeto de usuário com os dados enviados na requisição
6. O sistema altera os dados do usuário

Buscar Usuário

Responsável por retornar dados do usuário pelo ID

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “usuario-all”
2. O sistema busca os dados do usuário, bem como, os grupos e empresa ao qual ele está vinculado e os retorna como resposta

Filtrar Usuário

Responsável por retornar a lista de usuários filtrada por parte do nome (não o username)

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “usuario-all”
2. O sistema busca os dados de todos os usuários que correspondem aos critérios de filtro e os retorna como resposta

Remover Usuário

Responsável pela remoção de um usuário do sistema

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “usuario-all”
2. O sistema verifica se o usuário existe pelo ID
 - a) Se não, o sistema retorna uma mensagem de erro 400 informando que o usuário não foi encontrado
3. O sistema deleta o usuário

Listar Grupos de Usuário por Usuário ID

Responsável por retornar a lista de grupos do usuário pelo ID dele

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “usuario-all”
2. O sistema busca todos os grupos ao qual o usuário pertence pelo ID dele e retorna esses grupos como resposta

Cria vínculo Usuário/Grupo

Responsável por vincular um grupo de usuário ao usuário

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “usuario-all”
2. O sistema busca o usuário e o grupo pelos IDs presentes na requisição
3. O sistema registra um carrega um UsuarioGrupoMap com o usuário e o grupo buscados e registra os dados no banco de dados

Remover Vínculo Usuário/Grupo

Responsável por remover um vínculo entre usuário e grupo pelo ID do vínculo

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “usuario-all”
2. O sistema verifica se o vínculo existe
 1. Se não, retorna uma mensagem de erro 400 informando que o vínculo não foi encontrado.
3. O sistema deleta o vínculo

UCs de GRUPO DE USUÁRIO

Criar Grupo de Usuário

Responsável por registrar um novo grupo de usuário no banco de dados

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “usuario-all”
2. O sistema verifica se o grupo de usuário já existe registrado no sistema pelo nome
 - a) Se sim, retorna erro 400 informando que o grupo de usuário já existe registrado pelo nome
3. O sistema carrega o objeto de grupo de usuário com os dados enviados na requisição
4. O sistema registra o grupo de usuário

Atualizar Grupo de Usuário

Responsável por atualizar dados do grupo de usuário já registrado no banco de dados

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “usuario-all”
2. O sistema busca o grupo de usuário pelo ID recebido na requisição
3. O sistema verifica se o novo nome (recebido na requisição) é igual ao nome do grupo de usuário buscado
 - a) Se não, o sistema verifica se existe algum grupo de usuário registrado com o novo nome
 1. Se sim, o sistema retorna uma mensagem de erro 400 informando que já existe outro usuário com o nome informado
4. O sistema carrega o objeto de grupo de usuário com os dados enviados na requisição
5. O sistema altera os dados do grupo de usuário

Buscar Grupo de Usuário

Responsável por retornar dados do grupo de usuário pelo ID

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “usuario-all”
2. O sistema busca os dados do grupo de usuário e os retorna como resposta

Filtrar Grupo de Usuário

Responsável por retornar a lista de grupos de usuários filtrada por parte do nome

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “usuario-all”
2. O sistema busca os dados de todos os grupos de usuários que correspondem aos critérios de filtro e os retorna como resposta

Remover Grupo de Usuário

Responsável pela remoção de um grupo de usuário do sistema

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “usuario-all”
2. O sistema verifica se o grupo de usuário existe pelo ID
 - a) Se não, o sistema retorna uma mensagem de erro 400 informando que o grupo de usuário não foi encontrado
3. O sistema deleta o grupo de usuário

Listar Roles por Grupo de Usuário ID

Responsável por retornar a lista de roles pelo ID do grupo

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “usuario-all”
2. O sistema busca todos os roles do grupo de usuário pelo ID do grupo e retorna esses roles como resposta

Cria vínculo UsuárioGrupo/Role

Responsável por vincular um role ao grupo de usuário

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “usuario-all”
2. O sistema busca o grupo de usuário e o role pelos IDs presentes na requisição
3. O sistema registra um carrega um RoleGrupoMap com o grupo de usuário e o role buscados e registra os dados no banco de dados

Remover Vínculo UsuarioGrupo/Role

Responsável por remover um vínculo entre grupo de usuário e role pelo ID do vínculo

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “usuario-all”
2. O sistema verifica se o vínculo existe
 1. Se não, retorna uma mensagem de erro 400 informando que o vínculo não foi encontrado.
3. O sistema deleta o vínculo

UCs de ROLE

Criar Role

Responsável por registrar um novo role no banco de dados

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “usuario-all”
2. O sistema verifica se o role já existe registrado no sistema
 - a) Se sim, retorna erro 400 informando que o role já existe registrado pelo nome
3. O sistema carrega o objeto de role com os dados enviados na requisição
4. O sistema registra o role

Atualizar Role

Responsável por atualizar dados do role já registrado no banco de dados

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “usuario-all”
2. O sistema busca o role pelo ID recebido na requisição
3. O sistema verifica se o novo nome (recebido na requisição) é igual ao nome do role buscado
 - a) Se não, o sistema verifica se existe algum role registrado com o novo nome
 1. Se sim, o sistema retorna uma mensagem de erro 400 informando que já existe outro role com o nome informado
4. O sistema carrega o objeto de role com os dados enviados na requisição
5. O sistema altera os dados do role

Buscar Role

Responsável por retornar dados do role pelo ID

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “usuario-all”
2. O sistema busca os dados do role e os retorna como resposta

Filtrar Role

Responsável por retornar a lista de roles filtrada por parte do nome

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “usuario-all”
2. O sistema busca os dados de todos os roles que correspondem aos critérios de filtro e os retorna como resposta

Remover Role

Responsável pela remoção de um role do sistema

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “usuario-all”
2. O sistema verifica se o role existe pelo ID
 - a) Se não, o sistema retorna uma mensagem de erro 400 informando que o role não foi encontrado
3. O sistema deleta o role

UCs de EMPRESA

Criar Empresa

Responsável por registrar uma nova empresa no banco de dados

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “empresa-all”
2. O sistema verifica se a empresa já existe registrado no sistema
 - a) Se sim, retorna erro 400 informando que a empresa já existe registrado pelo nome
3. O sistema carrega o objeto de empresa com os dados enviados na requisição
4. O sistema registra a empresa

Atualizar Empresa

Responsável por atualizar dados da empresa já registrada no banco de dados

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “empresa-all”
2. O sistema busca a empresa pelo ID recebido na requisição
3. O sistema verifica se o novo nome (recebido na requisição) é igual ao nome da empresa buscada
 - a) Se não, o sistema verifica se existe alguma empresa registrada com o novo nome
 1. Se sim, o sistema retorna uma mensagem de erro 400 informando que já existe outra empresa registrada com o nome informado
4. Se está tentando mudar de temporário para regular
 - a) Então, seta a data de início regular para data atuais
5. O sistema carrega o objeto de empresa com os dados enviados na requisição
6. O sistema altera os dados de empresa

Atualizar Empresa (Não Admin)

Responsável por atualizar dados da empresa alteráveis por usuário não administrador

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “no-admin-update-empresa”
2. O sistema busca a empresa pelo ID recebido na requisição
3. O sistema verifica se o novo nome (recebido na requisição) é igual ao nome da empresa buscada
 1. Se não, o sistema verifica se existe alguma empresa registrada com o novo nome
 1. Se sim, o sistema retorna uma mensagem de erro 400 informando que já existe outra empresa registrada com o nome informados
4. O sistema carrega o objeto de empresa com os dados limitados da empresa enviados na requisição
5. O sistema altera os dados da empresa

Buscar Empresa

Responsável por retornar dados da empresa pelo ID

Fluxo principal:

1. O sistema verifica se o usuário logado possui um dos seguintes roles: “empresa-all”, “empresa-get”
2. O sistema busca os dados da empresa e os retorna como resposta

Filtrar Empresa

Responsável por retornar a lista de empresas filtrada por parte do nome

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “empresa-all”
2. O sistema busca os dados de todas as empresas que correspondem aos critérios de filtro e os retorna como resposta

Remover Empresa

Responsável pela remoção de uma empresa do sistema

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “usuario-all”
2. O sistema verifica se a empresa existe pelo ID
 - a) Se não, o sistema retorna uma mensagem de erro 400 informando que a empresa não foi encontrado
3. O sistema para todos os monitoramentos de todos os dispositivos vinculados a empresa
4. O sistema deleta a empresa

UCs de CONFIGURAÇÕES

Atualizar Configurações

Responsável por atualizar dados de configurações do sistema.

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “config-all”
2. O sistema busca as configurações no banco de dados
3. O sistema carrega o objeto de configurações com os dados de configurações enviados na requisição
4. O sistema altera os dados das configurações
5. O sistema envia os dados da nova configuração para todas as instâncias do microsserviço “disp-monitor” via mensagem.
6. As instâncias do microsserviço “disp-monitor” recebem a mensagem e atualizam todas as suas threads de monitoramento de dispositivo com os dados da nova configuração

Buscar Configurações

Responsável por retornar dados das configurações

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role: “config-all”
2. O sistema busca os dados das configurações e os retorna como resposta
 1. O sistema busca na tabela config
 2. O sistema busca informações sobre cada monitor_server do sistema
 1. O sistema carrega dados sobre: uso de memória e CPU de cada monitor_server
3. O sistema retorna também informações sobre uso de memória e CPU do microsserviço principal

UCs de MONITOR SERVER

Criar MonitorServer

Responsável por registrar um novo monitor_server no banco de dados

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “config-all”
2. O sistema verifica se o monitor_server já existe registrado no sistema pelo host
 - a) Se sim, retorna erro 400 informando que o monitor_server já existe registrado pelo host
3. O sistema carrega o objeto de monitor_server com os dados enviados na requisição
4. O sistema registra o monitor_server

Atualizar MonitorServer

Responsável por atualizar dados do monitor_server já registrado no banco de dados

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “config-all”
2. O sistema busca o monitor_server pelo ID recebido na requisição
3. O sistema verifica se o novo host (recebido na requisição) é igual ao host do monitor_server buscado
 - a) Se não, o sistema verifica se existe algum monitor_server registrado com o novo host
 1. Se sim, o sistema retorna uma mensagem de erro 400 informando que já existe outro monitor_server com o host informado
4. O sistema carrega o objeto de monitor_server com os dados enviados na requisição
5. O sistema altera os dados do monitor_server

Buscar MonitorServer

Responsável por retornar dados do monitor_server pelo ID

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “config-all”
2. O sistema busca os dados do monitor_server e os retorna como resposta

Filtrar MonitorServers

Responsável por retornar a lista de monitor_servers filtrada por parte do host

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “config-all”
2. O sistema busca os dados de todos os monitor_servers que correspondem aos critérios de filtro e os retorna como resposta

Remover MonitorServer

Responsável pela remoção de um monitor_server do sistema

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “usuario-all”
2. O sistema verifica se o monitor_server existe pelo ID
 - a) Se não, o sistema retorna uma mensagem de erro 400 informando que o monitor_server não foi encontrado
3. O sistema deleta o monitor_server

UCs de DISPOSITIVO

Criar Dispositivo

Responsável por registrar um novo dispositivo no banco de dados

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “dispositivo-all”
2. O sistema verifica se o ID da empresa do dispositivo corresponde ao ID de empresa vinculado ao usuário logado
3. O sistema verifica se o dispositivo já existe registrado no sistema pelo nome e pela empresa
 - a) Se sim, retorna erro 400 informando que o role já existe registrado pelo nome e empresa
4. O sistema carrega o objeto de dispositivo com os dados enviados na requisição
5. O sistema registra o dispositivo
6. O sistema starta o monitoramento do dispositivo

Atualizar Dispositivo

Responsável por atualizar dados do dispositivo já registrado no banco de dados

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “dispositivo-all”
2. O sistema verifica se o ID da empresa do dispositivo corresponde ao ID de empresa vinculado ao usuário logado
3. O sistema busca o dispositivo pelo ID recebido na requisição
4. O sistema verifica se o novo nome (recebido na requisição) é igual ao nome do dispositivo buscado
 - a) Se não, o sistema verifica se existe algum dispositivo registrado com o novo nome e o ID da empresa
 1. Se sim, o sistema retorna uma mensagem de erro 400 informando que já existe outro dispositivo na empresa com o nome informado
5. O sistema carrega o objeto de dispositivo com os dados enviados na requisição
6. O sistema altera os dados do dispositivo
7. O sistema envia os dados do dispositivo para as instâncias do microsserviço “disp-monitor” via mensagem
8. As instâncias do microsserviço “disp-monitor” recebem a mensagem e verificam se há alguma thread de monitoramento em execução vinculada ao dispositivo recebido via mensagem
 - a) Se sim, seta o novo dispositivo no objeto da thread encontrada

Buscar Dispositivo

Responsável por retornar dados do dispositivo pelo ID

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “dispositivo-all”
2. O sistema verifica se o ID da empresa do dispositivo corresponde ao ID de empresa vinculado ao usuário logado
3. O sistema busca os dados do dispositivo e os retorna como resposta

Filtrar Dispositivo

Responsável por retornar a lista de dispositivos filtrada por parte do nome e empresa

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “dispositivo-all”
2. O sistema verifica se o ID da empresa do dispositivo corresponde ao ID de empresa vinculado ao usuário logado
3. O sistema busca os dados de todos os dispositivos que correspondem aos critérios de filtro e

os retorna como resposta

Remover Dispositivo

Responsável pela remoção de um dispositivo do sistema

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “dispositivo-all”
2. O sistema verifica se o ID da empresa do dispositivo corresponde ao ID de empresa vinculado ao usuário logado
3. O sistema verifica se o dispositivo existe pelo ID
 - a) Se não, o sistema retorna uma mensagem de erro 400 informando que o dispositivo não foi encontrado
4. O sistema deleta o dispositivo
5. O sistema para o monitoramento do dispositivo, caso o dispositivo esteja sendo monitorado

UCs de EVENTO

Listar Eventos por Dia

Responsável por retornar a lista de eventos filtrados pelo dia de ocorrência

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “dispositivo-all”
2. O sistema verifica se o ID da empresa do dispositivo do evento corresponde ao ID de empresa vinculado ao usuário logado
3. O sistema busca todos os eventos, filtrando pelo dia de ocorrência, e retorna os dados como resposta

Listar Eventos por Intervalo

Responsável por retornar a lista de eventos filtrados pelo intervalo de datas de ocorrência

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “dispositivo-all”
2. O sistema verifica se o ID da empresa do dispositivo do evento corresponde ao ID de empresa vinculado ao usuário logado
3. O sistema busca todos os eventos, filtrando por dia entre o início e fim do intervalo recebido na requisição, e retorna os dados como resposta

Busca Evento

Responsável por buscar um evento pelo ID

Fluxo principal:

1. O sistema verifica se o usuário logado possui o role “dispositivo-all”
2. O sistema verifica se o ID da empresa do dispositivo do evento corresponde ao ID de empresa vinculado ao usuário logado
3. O sistema busca o evento pelo ID recebido na requisição e retorna os dados como resposta

UCs de PAGAMENTOS

Listar Pagamentos

Responsável por listar os pagamentos para a empresa de desde o dia que a empresa se tornou regular até o pagamento do mês atual ou do anterior caso o dia atual seja menor que o dia de pagamento.

Fluxo principal:

1. Busca na tabela config o valor de pagamento
2. Busca na tabela empresa a data que a empresa se tornou regular
3. Busca na tabela empresa o dia de pagamento e a data de que esteja pago até ela
4. Carrega todos os pagamentos com suas datas de pagamento e se estão pagos
5. Calcula o débito total em cima dos pagamentos não pagos
6. Retorna os pagamentos e o débito total

Regularizar Dívida

Responsável por efetuar o pagamento de toda a dívida. Isto é, todos os pagamentos não pagos somados.

Fluxo principal:

1. Carrega dados da empresa
2. Calcula a data da última parcela
3. Seta o valor da última parcela no campo “ pagoAte”
4. Salva no banco de dados
5. Retorna mensagem sobre até onde agora está pago

Gerar Pagamento Pix QRCode

Responsável por gerar o QRCode da dívida total dos pagamentos e retornar ao frontend para ser pago via PIX

Fluxo principal:

1. Busca empresa e config no banco de dados
2. Calcula o valor total da dívida
3. Gera a imagem QRCode em formato base64
4. Retorna a imagem do QRCode em formato base64 e o texto do pix copia e cola

UCs de MONITORAMENTO DE DISPOSITIVOS

UpdateDispositivo

Função responsável por atualizar o flag de “sendoMonitorado” de um dispositivo, bem como de enviar os dados do dispositivo para “disp-monitor” e para o frontend via websocket.

Fluxo principal:

- 1) O flag “sendoMonitorado” na tabela dispositivo é alterado para “true”
- 2) Dados do dispositivo são retornados para o frontend de todos os usuários admins ou que pertencem a empresa do dispositivo, via websocket
- 3) Os dados do dispositivo são enviados para as instâncias do microsserviço “disp-monitor” via mensagem para que cada instância busque o monitoramento vinculado ao dispositivo. Quando encontra o monitoramento, seta o objeto dispositivo na thread de monitoramento correspondente.

Start Monitoramento

Responsável por alocar uma nova thread, em um dos monitor_servers disponíveis, para realizar os testes de conexão com o dispositivo.

Fluxo principal:

- 4) Verifica se o usuário logado possui o role “dispositivo-monitoramento-all”
- 5) Verifica se o dispositivo existe no banco de dados
 - a) Se não, retorna uma mensagem de erro 400 informando que o usuário não foi encontrado
- 6) Verifica se o dispositivo já está sendo monitorado
 - a) Faz uma busca em todos os monitors_servers para saber se o dispositivo já está sendo monitorado
 - b) Se o dispositivo já está sendo monitorado, executa “**UpdateDispositivo**”
- 7) Tenta startar o monitoramento em um dos monitor_servers registrados. Isto é, alocar a thread de monitoramento
 - a) Se houver um problema de acesso ao monitor_server, uma mensagem de log é enviada para saída padrão informando isto e é passado para o próximo monitor_server
 - b) Se o limite de monitoramentos for excedido no monitor_server corrente, tenta startar no próximo até startar em um ou tentar em todos
 - c) Se o limite de monitoramento estiver esgotado em todos os monitor_servers, retorna uma mensagem de erro 400 informando sobre o limite excedido
 - d) Se o monitoramento foi startado com sucesso
 - 1) Executa “**UpdateDispositivo**”
 - 2) Salva no banco de dados como configuração o índice do próximo monitor_server. Exemplo, se foi startado no monitor_server de índice 0, o índice do próximo monitor_server é 1, caso existam mais que 1 monitor_servers.
 - 3) Uma mensagem via websocket é enviada para os admins e os usuários da empresa do dispositivo. A mensagem enviada tem como dados a quantidade total de dispositivos da empresa e a quantidade de dispositivos sendo monitorados (são informações gerais sobre os dispositivos)

Stop Monitoramento

Responsável por parar a execução da thread de monitoramento do dispositivo e removê-lo da lista de dispositivos monitorados no monitor_server correspondente.

Fluxo principal:

- 1) Verifica se o usuário logado possui o role “dispositivo-monitoramento-all”
- 2) Verifica se o dispositivo existe no banco de dados
 - a) Se não, retorna uma mensagem de erro 400 informando que o usuário não foi encontrado
- 3) Tenta parar o monitoramento nos monitor_servers até encontrar o monitoramento em um deles e pará-lo.
 - a) O endpoint de stop uma instância de “disp-monitor” é chamado ela se encarrega de buscar o monitoramento e parar a thread dele caso encontre.
- 4) Se nenhuma instância de monitor_server encontrar na sua lista de monitoramentos o monitoramento do dispositivo buscado
 - a) Executa “**UpdateDispositivo**”
 - b) Retorna uma mensagem de erro 400 informando que o dispositivo já não está sendo monitorado

Start Todos os Dispositivos da Empresa

Responsável por startar todos os dispositivos de uma empresa pelo ID dela

Fluxo principal:

- 1) Verifica se o usuário logado possui o role “dispositivo-monitoramento-all”
- 2) Tenta startar todos os dispositivos da empresa pelo ID deles
 - a) Chama “**Start Monitoramento**” para cada dispositivo da empresa
 - 1) Caso um dispositivo seja startado com sucesso, o contador de starts bem sucedidos é incrementado
 - 2) Caso o limite de monitoramentos tenha sido excedido, uma mensagem de log informando isto é enviada para saída padrão via Logger
 - 3) Caso o dispositivo já tenha sido startado antes, apenas passa para o próximo start
- 3) Uma mensagem é retornada para o frontend informando quantos dispositivos foram startados do total de dispositivos da empresa

Stop Todos os Dispositivos da Empresa

Responsável por parar todos os dispositivos de uma empresa, sendo monitorados

Fluxo principal:

- 1) Verifica se o usuário logado possui o role “dispositivo-monitoramento-all”
- 2) Tenta parar todos os dispositivos da empresa pelo ID deles
 - a) Chama “**Stop Monitoramento**” para cada dispositivo da empresa
 - 1) Caso o dispositivo já não esteja sendo monitorado, passa para o próximo stop

Start Todos os Dispositivos

Responsável por startar todos os dispositivos registrados no sistema

Fluxo principal:

- 1) Verifica se o usuário logado possui o role “dispositivo-monitoramento-all”
- 2) Tenta startar todos os dispositivos do sistema
 - a) Chama “Start Monitoramento” para cada dispositivo
 - 1) Caso um dispositivo seja startado com sucesso, o contador de starts bem sucedidos é incrementado
 - 2) Caso o limite de monitoramentos tenha sido excedido, uma mensagem de log informando isto é enviada para saída padrão via Logger
 - 3) Caso o dispositivo já tenha sido startado antes, apenas passa para o próximo start
- 3) Uma mensagem é retornada para o frontend informando quantos dispositivos foram

startados do total de dispositivos da empresa

Stop Todos os Dispositivos

Responsável por parar o monitoramento de todos os dispositivos registrados no sistema

Fluxo principal:

- 1) Verifica se o usuário logado possui o role “dispositivo-monitoramento-all”
- 2) Tenta parar todos os dispositivos do sistema
 - a) Utiliza o service de “stopAllMonitoramentos” do microsserviço “disp-monitor” para parar todos os monitoramentos
 - b) Seta todos os dispositivos como não monitorados
 - c) Envia mensagem via websocket com atualizações do número de dispositivos sendo monitorados

A Thread de Monitoramento

A thread de monitoramento é responsável por iniciar o lote de envio de pacotes ICMP ao dispositivo para teste da conexão. Após o fim do lote de testes, a thread executando na instância de “disp-monitor” executa algumas tarefas referentes ao resultado dos testes, descritas abaixo:

Fluxo principal:

- 1) Realiza os testes e armazena o resultado em variáveis
- 2) Após o teste
 - a) Se o número de falhas for maior que a porcentagem de falhas que indica falha na conexão
 - 1) Se o status do dispositivo for INATIVO:
 - A) Muda para ATIVO, calcula o tempo de inatividade e incrementa o tempo de inatividade total
 - b) Se o número de falhas for menor que a porcentagem
 - 1) Se o status do dispositivo for ATIVO:
 - A) Muda para INATIVO, seta a variável de tempo de inatividade para a data/hora atual e incrementa a quantidade de quedas na conexão
 - 3) Calcula a latência média e seta ela no objeto do dispositivo
 - 4) Envia o dispositivo via mensagem para o microsserviço “main-api” que recebe a mensagem, salva a latência média, status, data de atualização do dispositivo no banco de dados e envia os dados, via websocket, para todos os usuários admin ou vinculados a empresa do dispositivo, conectados
 - 5) Se o tempo entre o último evento registrado (ou o início da execução da thread) e o tempo atual for maior ou igual ao período de registro de eventos
 - a) O evento é criado e enviado via mensagem para o microsserviço “main-api” que recebe a mensagem e registra o evento no banco de dados
 - 6) Se uma falha de IO ocorrer na execução do comando PING, um log com a mensagem correspondente é enviado para a saída padrão