

# Documentação

## Controllers

### AdminMasterController:

Método	Rota	Descrição
GET	/AdminsMaster/{nome}/{senha}	Verifica as credenciais de login do administrador. Retorna as informações do administrador se as credenciais estiverem corretas ou retorna um status HTTP 401 (Unauthorized) se as credenciais estiverem incorretas.

### AdminController:

Método HTTP	Rota	Descrição
GET	/Admins/{nome}/{senha}	Retorna um único administrador com nome de usuário <b>nome</b> e senha <b>senha</b> . Se o administrador não for encontrado, a resposta é <b>401 Unauthorized</b> .
GET	/Admins/getAll/{nomeAdminMaster}/{senhaAdminMaster}	Retorna uma lista de todos os administradores. A solicitação deve ser feita com credenciais de administrador mestre válidas (com nome de usuário <b>nomeAdminMaster</b> e senha <b>senhaAdminMaster</b> ). Se as credenciais não forem válidas, a resposta é <b>401 Unauthorized</b> .
GET	/Admins/getOne/{nomeAdminMaster}/{senhaAdminMaster}/{id}	Retorna um único administrador com ID <b>id</b> . A solicitação deve ser feita com credenciais de administrador mestre válidas. Se as credenciais não forem válidas, a resposta é <b>401 Unauthorized</b> . Se o administrador não for encontrado, a resposta é <b>404 Not Found</b> .

Método HTTP	Rota	Descrição
GET	<code>/Admins/getAllByInstituto/{nomeAdminMaster}/{senhaAdminMaster}/{idInstituicao}</code>	Retorna uma lista de todos os administradores que pertencem à instituição com ID <b>idInstituicao</b> . A solicitação deve ser feita com credenciais de administrador mestre válidas. Se as credenciais não forem válidas, a resposta é <b>401 Unauthorized</b> . Se a instituição não for encontrada, a resposta é <b>404 Not Found</b> .
POST	<code>/Admins/{nomeAdminMaster}/{senhaAdminMaster}/{idInstituicao}</code>	Cria um novo administrador e o associa à instituição com ID <b>idInstituicao</b> . A solicitação deve ser feita com credenciais de administrador mestre válidas. Se as credenciais não forem válidas, a resposta é <b>401 Unauthorized</b> . Se a instituição não for encontrada, a resposta é <b>404 Not Found</b> . Se houver um erro ao criar o administrador, a resposta é <b>500 Internal Server Error</b> .
PUT	<code>/Admins/{nomeAdminMaster}/{senhaAdminMaster}/{id}</code>	Atualiza um administrador existente com ID <b>id</b> . A solicitação deve ser feita com credenciais de administrador mestre válidas. Se as credenciais não forem válidas, a resposta é <b>401 Unauthorized</b> . Se o administrador não for encontrado, a resposta é <b>404 Not Found</b> . Se houver um erro ao atualizar o administrador, a resposta é <b>500 Internal Server Error</b> .
DELETE	<code>/Admins/{nomeAdminMaster}/{senhaAdminMaster}/{id}</code>	Exclui um administrador existente com ID <b>id</b> . A solicitação deve ser feita com credenciais de administrador mestre válidas. Se as credenciais não forem válidas, a resposta é <b>401 Unauthorized</b> . Se o administrador não for encontrado, a resposta é <b>404 Not Found</b> . Se houver um erro ao excluir o administrador, a resposta é <b>500 Internal Server Error</b> .

## InstituicoesController:

HTTP Method	Endpoint	Parâmetros	Produce	Descrição
GET	/Instituicoes/admin/{nomeAdmin}/{senhaAdmin}/{id}	<b>nomeAdmin</b> (String), <b>senhaAdmin</b> (String), <b>id</b> (UUID)	JSON	Retorna uma instituição específica a partir do ID informado.
GET	/Instituicoes/adminMaster/{nomeAdmin}/{senhaAdmin}/{id}	<b>nomeAdmin</b> (String), <b>senhaAdmin</b> (String), <b>id</b> (UUID)	JSON	Retorna uma instituição específica a partir do ID informado para administradores mestres.
GET	/Instituicoes/getAll/{nomeAdmin}/{senhaAdmin}	<b>nomeAdmin</b> (String), <b>senhaAdmin</b> (String)	JSON	Retorna uma lista com todas as instituições.
POST	/Instituicoes/{nomeAdmin}/{senhaAdmin}	<b>nomeAdmin</b> (String), <b>senhaAdmin</b> (String), corpo (JSON)	JSON	Adiciona uma nova instituição.
PUT	/Instituicoes/{nomeAdmin}/{senhaAdmin}/{id}	<b>nomeAdmin</b> (String), <b>senhaAdmin</b> (String), <b>id</b> (UUID), corpo (JSON)	JSON	Altera uma instituição específica a partir do ID informado.
DELETE	/Instituicoes/{adminNome}/{senhaAdmin}/{id}	<b>nomeAdmin</b> (String), <b>senhaAdmin</b> (String), <b>id</b> (UUID)	JSON	Remove uma instituição específica a partir do ID informado.

**UsuariosController:**

HTTP Method	Endpoint	Parâmetros	Descrição
GET	/Usuarios/{email}/{senha}	email: String, senha: String	Retorna as informações do usuário que realizou login
GET	/Usuarios/getAllUsersIntituto/{idInstituto}/{emailUsuario}/{senhaUsuario}	idInstituto: UUID, emailUsuario: String, senhaUsuario: String	Retorna todos os usuários vinculados a uma instituição
GET	/Usuarios/getAllUsuariosInstitutoPaginado/{idInstituto}/{emailUsuario}/{senhaUsuario}	idInstituto: UUID, emailUsuario: String, senhaUsuario: String, pagina: Integer (opcional, default=0), tamanho: Integer (opcional, default=10)	Retorna todos os usuários vinculados a uma instituição, de forma paginada
GET	/Usuarios/{id}	id: UUID	Retorna as informações de um usuário específico, a partir do seu ID
POST	/Usuarios/{nomeInstituicao}/{senhaInstituicao}	nomeInstituicao: String, senhaInstituicao: String	Cria um novo usuário, vinculando-o a uma instituição

HTTP Method	Endpoint	Parâmetros	Descrição
PUT	/Usuarios/{emailUsuario}/{senhaUsuario}	emailUsuario : String, senhaUsuario: String	Atualiza as informações do usuário que realizou login
PUT	/Usuarios/admin/{nomeAdmin}/{senhaAdmin}/{id}	nomeAdmin: String, senhaAdmin: String, id: UUID, dto: UsuariosDTO (request body)	Atualiza um usuário existente na base de dados. Requer autenticação de um admin da mesma instituição do usuário.
PUT	/Usuarios/online/{emailUsuario}/{senhaUsuario}/{online}	emailUsuario : String, senhaUsuario: String, online: Boolean	Altera o status de "online" de um usuário na base de dados. Requer autenticação do próprio usuário.
DELETE	/Usuarios/{emailUsuario}/{senhaUsuario}	emailUsuario : String, senhaUsuario: String	Remove um usuário da base de dados. Requer autenticação do próprio usuário.
DELETE	/Usuarios/admin/{nomeAdmin}/{senhaAdmin}/{id}	nomeAdmin: String, senhaAdmin: String, id: UUID	Remove um usuário da base de dados. Requer autenticação de um admin da mesma instituição do usuário.

## QuestõesController:

HTTP Method	Endpoint	Parâmetros	Descrição
GET	/Questoes/{id}	UUID id	Busca uma questão pelo seu ID. Retorna um objeto ResponseEntity com o status HTTP e um possível corpo contendo a questão buscada, além de alguns links que podem ser acessados após a busca.
GET	/Questoes/LimitRange/{idInstituicao}/{limit}	UUID idInstituicao, int limit	Busca uma lista de questões aleatórias com um tamanho específico para a instituição correspondente ao ID informado. Retorna um objeto ResponseEntity com o status HTTP e um possível corpo contendo as questões buscadas, além de alguns links que podem ser acessados após a busca.
GET	/Questoes/getAll/{idInstituicao}	UUID idInstituicao	Busca uma lista com todas as questões da instituição correspondente ao ID informado. Retorna um objeto ResponseEntity com o status HTTP e um possível corpo contendo as questões buscadas, além de alguns links que podem ser acessados após a busca.
POST	/Questoes/{adminNome}/{senhaAdmin}	String adminNome, String senhaAdmin, QuestoesDTO questoesDto	Cadastra uma nova questão para a instituição do admin correspondente ao nome e senha informados. Retorna um objeto ResponseEntity com o status HTTP e um possível corpo contendo a questão cadastrada, além de alguns links que podem ser acessados após o cadastro.
PUT	/Questoes/{adminNome}/{adminSenha}/{id}	UUID id, String adminNome, String adminSenha, QuestoesDTO questoesDTO	Atualiza uma questão pelo seu ID. Retorna um objeto ResponseEntity com o status HTTP e um possível corpo, além de alguns links que podem ser acessados após a atualização.

HTTP Method	Endpoint	Parâmetros	Descrição
DELETE	/Questoes/{adminNome}/{senhaAdmin}/{id}	String adminNome, String senhaAdmin, UUID id	Deleta uma questão pelo seu ID. Retorna um objeto ResponseEntity com o status HTTP e um possível corpo contendo a mensagem de sucesso ou de erro.

### NoticiasController:

HTTP Method	Endpoint	Parâmetros	Descrição
GET	/Noticias/{idInstituicao}/{index}/{size}	int <b>page</b> , int <b>size</b> , UUID <b>idInstituicao</b>	Retorna uma lista paginada de notícias da instituição com o ID especificado
GET	/Noticias/{id}	UUID <b>id</b>	Retorna a notícia com o ID especificado
GET	/Noticias/getAll/{idInstituicao}	UUID <b>idInstituicao</b>	Retorna todas as notícias da instituição com o ID especificado
POST	/Noticias/{adminNome}/{senhaAdmin}	String <b>adminNome</b> , String <b>senhaAdmin</b> , NoticiasDTO <b>noticiasDto</b>	Cria uma nova notícia para a instituição do administrador com o nome e senha especificados
PUT	/Noticias/{adminNome}/{adminSenha}/{id}	UUID <b>id</b> , String <b>adminNome</b> , String <b>adminSenha</b> , NoticiasDTO <b>noticiasDTO</b>	Atualiza a notícia com o ID especificado para a instituição do administrador com o nome e senha especificados
DELETE	/Noticias/{adminNome}/{adminSenha}/{id}	UUID <b>id</b> , String <b>adminNome</b> , String <b>adminSenha</b>	Exclui a notícia com o ID especificado para a instituição do administrador com o nome e senha especificados