|  |  |  |
| --- | --- | --- |
| Rota | Verbo HTML | Detalhe |
| /eventos | GET ; POST | .Listar todos os eventos caso utilizador esteja autenticado. Caso utilizador não esteja autenticado, retornar erro 401.  .Criar novo evento caso todos os campos estejam preenchidos e utilizador atual seja administrador. Caso utilizador não possua permissões, retornar erro 401. |
| /eventos/:id | GET ; PUT DELETE | .Ler detalhes do evento com id correspondente caso utilizador esteja autenticado. Caso utilizador não esteja autenticado, retornar erro 401. Retornar erro 404 se id não corresponder a nenhum evento.  .Editar evento com id correspondente apenas se todos os campos estiverem preenchidos e utilizador for administrador.  Eliminar evento com id correspondente, apenas se o utilizador atual seja administrador. Caso utilizador não possua permissões, retornar erro 401. |
| /bolsas | GET ; POST | .Listar todas as bolsas caso utilizador esteja autenticado. Caso utilizador não esteja autenticado, retornar erro 401.  .Criar nova bolsa caso todos os campos estejam preenchidos e utilizador seja administrador. Caso utilizador não possua permissões, retornar erro 401. |
| /bolsas/:id | GET ; PUT  DELETE | .Ler detalhes da bolsa com id correspondente caso utilizador possua sessão iniciada. Caso utilizador não esteja autenticado, retornar erro 401.Retornar erro 404 se id não corresponder a nenhuma bolsa.  .Atualizar bolsa com id correspondente, apenas se todos os campos estiverem preenchidos e utilizador for administrador. Eliminar bolsa com id correspondente, apenas se o utilizador atual seja administrador. Caso utilizador não possua permissões, retornar erro 401. |
| /alumni | GET; POST | .Ler todos os estudantes inscritos na plataforma caso utilizador esteja autenticado. Caso utilizador não esteja autenticado, retornar erro 401.  . Criar novo utilizador apenas se todos os campos estiverem preenchidos e utilizador não estiver autenticado. Caso utilizador esteja autenticado, retornar erro 401. |
| /alumni/:numero | GET ; PUT | .Ler detalhes do estudante com número correspondente, ler *skills* do estudante com número correspondente, ler links do estudante com número correspondente e ler cursos do estudante com número correspondente.  .Atualizar detalhes do estudante com número correspondente, apenas se todos os campos estiverem preenchidos (excluindo o CV que é facultativo) e utilizador estiver autenticado. Caso utilizador não esteja autenticado retornar erro 401. Retornar erro 404 se número não corresponder a nenhum estudante. |
| /alumni/:numero/skills | GET; | Retorna todas as skills de um alumni. |
| /alumni/:numero/skills/:skillId | PUT; DELETE; POST | Cria, faz update ou apaga uma skills de um alumni |
| /alumni/:numero/tools | GET; | Retorna todas as tools de um alumni. |
| /alumni/:numero/ tools/:id | PUT; DELETE; POST | Cria, faz update ou apaga uma tool de um alumni |
| /alumni/:numero/cursos | GET; | Retorna todos os cursos de um alumni. |
| /alumni/:numero/ cursos/:id | PUT; DELETE; POST | Cria, faz update ou apaga um curso de um alumni |
| /testemunhas/:id | GET ; PUT DELETE | .Ler detalhes da testemunha com id correspondente, apenas se utilizador estiver autenticado. Caso utilizador não esteja autenticado retornar erro 401. Retornar erro 404 se id não corresponde a nenhuma testemunha.  .Atualizar testemunha com id corresponde, apenas se todos os campos estiverem preenchidos e utilizador for administrador. Eliminar testemunha com id correspondente apenas se utilizador atual seja administrador. Caso utilizador não possua permissões, retornar erro 401. |
| /auth/signin/alumni | POST | Login Alumni. |
| /auth/signin/professor | POST | Login Professor. |
| /auth/signup/alumni | POST | Registar Alumni. |