



Objetivo do Projeto: Realizar e manter registros de Clientes, Outorgas, Licenças Ambientais, Declaração Ambiental, Cadastro Ambiental Rural e Cadastro de Crédito Financeiro.

Página Login

ngOnInit(): Método responsável por carregar a página de login para o usuário.
get(): Com o método get(), são capturados os input's dos campos do formulário.
logar(): Método responsável por validar a condição de logado.
login(): Método responsável por validar o usuário no sistema, sendo o mesmo direcionado para o menu principal de acordo com o perfil do usuário.
onSubmit(): Método responsável por efetivar o login do usuário no sistema.
openModal(): Modal responsável por abrir a janela de recuperação de senha.

Página Home

linkCliente(): Método responsável pela navegação da página de Cliente do Menu lateral para o usuário com perfil de Administrador.
linkDadosPessoais(): Método responsável pela navegação da página de dados pessoais no Menu lateral para o usuário com perfil de Cliente.
linkAmbiental(): Método responsável pela navegação da página Ambiental no Menu lateral para a Home Ambiental.
linkUsuarios(): Método responsável pela navegação da página Administrador no Menu lateral para o usuário com perfil de administrador.
linkAlterarUsuario(): Método responsável pela alteração da senha do usuário.
linkOutorgas(): Método responsável pela navegação da página de Outorga.
linkAmbientais(): Método responsável pela navegação da página de Licença Ambiental.
linkDeclaracoes(): Método responsável pela navegação da página de Declaração Ambiental.
linkCredito(): Método responsável pela navegação da página de Cadastro Crédito Financeiro.

ngOnInit(): Método responsável por carregar as informações do dashboard.
retornaDiasVencimento(): Método responsável por retornar os aniversariantes do dia e mês.
vencimento(): Método responsável por retornar informações de vencimento das Licenças.
openModal(): Método responsável por abrir a janela para alteração de senha.

Página Cliente – Estruturado com as páginas cria-cliente e lista-cliente e o componente lista-cliente-item.

ngOnInit(): Responsável por carregar a página de Cliente, invoca os métodos <code>criaFormulario()</code> , <code>usuarioService.init()</code> , <code>clienteService.init()</code> , <code>usuarioService.getAll()</code> e <code>clienteService.getAll()</code> para listar o cliente de acordo com o usuário logado. Caso, o usuário possua o perfil de administrador, serão listados todos os Clientes e usuários com perfil de Administrador.
criaFormulario(): Neste método é instanciado o formulário através do <code>formBuild</code> , onde o <code>Validators.required</code> é utilizado para validação dos campos.
get(): Com o método <code>get()</code> , são capturados os input's dos campos do formulário.
acao(): Neste método é verificado se a ação é a de Criação ou Atualização dos dados do Cliente, o controle da ação é realizado através dos botões <code>pageTitle</code> e <code>botaoTitle</code> .
cadastraListaGlobal(): Método responsável por listar o cliente de acordo com o usuário logado, sendo invocado o método <code>clienteService.createGlobal()</code> . No caso, de usuário com perfil de administrador, serão listados todos os clientes cadastrados no sistema.
atualizaListaGlobal(): Utiliza-se o <code>usuarioService.update()</code> para listar os Clientes cadastrados no sistema, sendo possível através do menu individual, editar informações dos campos nome, CPF, RG, Telefone, Data de Nascimento, Patrimônio, Produtividade Agropecuária, Informações Adicionais, email, senha e Arquivos Seleccionados. Este recurso é acessível apenas pelo usuário com perfil de administrador.
onSubmit(): Responsável por enviar os dados do formulário para o banco de dados.
uploadFile():() Método responsável por setar a foto do Cliente na tela.
openGalery(): Método responsável por abrir a janela de acesso a galeria. Este recurso é acessível apenas pelo usuário com perfil de administrador.
openArquivos(): Método responsável por realizar o download dos arquivos das licenças.
uploadFileTo(): Método responsável por atualizar/substituir arquivo de foto do cadastro. Este recurso é acessível apenas pelo usuário com perfil de administrador.
salvarIr(): Método responsável por salvar o arquivo de Imposto de Renda.

Página Usuário – Estruturado com as páginas cria-usuario e lista-usuario e o componente lista-usuario-item.

ngOnInit(): Responsável por carregar a página de Usuário, invoca os métodos criaFormulario(), usuarioService.init(), clienteService.init(), usuarioService.getAll() e clienteService.getAll() para listar os Usuários e Clientes para o usuário com perfil de Administrador.
criaFormulario(): Neste método é instanciado o formulário através do formBuild, onde o Validators.required é utilizado para validação dos campos.
get(): Com o método get(), são capturados os input's dos campos do formulário.
acao(): Neste método é verificado se a ação é a de Criação ou Atualização dos dados do Cliente, o controle da ação é realizado através dos botões pageTitle e botaoTitle.
atualizaListaGlobal(): Utiliza-se o usuarioService.update() para listar os usuários cadastrados no sistema, sendo possível através do menu individual, editar informações dos campos nome, email, senha e foto do perfil. Este recurso é acessível apenas pelo usuário com perfil de administrador.
onSubmit(): Responsável por enviar os dados do formulário para o banco de dados.
uploadFile():() Método responsável por setar a foto do usuário na tela.
openGalery(): Método responsável por abrir a janela de acesso a galeria. Este recurso é acessível apenas pelo usuário com perfil de administrador.
uploadFileTo(): Método responsável por salvar a foto do usuário no servidor.
deletePicture(): Método responsável pela exclusão do arquivo de foto substituído no perfil do usuário.
deletePicturePasta(): Método responsável pela exclusão do arquivo temporário de foto do sistema.
clearField(): Método utilizado para limpar os caracteres obscuros do campo de senha ao ser clicado.
verificarSenha(): Método utilizado para reescrever os caracteres obscuros do campo de senha ao retirar o cursor.
retornaDataNascimento(): Método responsável por retornar a data no formato dia/mês/ano.
retornaDataMMDDYYYY(): Método responsável por retornar a data no formato mês/dia/ano.

Página Home Ambiental

linkDA(): Método responsável por direcionar a navegação para a página de Declaração Ambiental.
linkLA(): Método responsável por direcionar a navegação para a página de Licença Ambiental.
linkCAR(): Método responsável por direcionar a navegação para a página de Cadastro Ambiental Rural.
linkOutorga(): Método responsável por direcionar a navegação para a página de Outorga.
linKCredito(): Método responsável por direcionar a navegação para a página de Cadastro de Crédito.
ngOnInit(): Método responsável por carregar a página Home Ambiental.

Página Outorga – Estruturado com as páginas cria-outorga e lista-outorga e o componente lista-outorga-item.

ngOnInit(): Responsável por carregar a página de Outorgas, invoca os métodos <code>criaFormulario()</code> , <code>clienteService.init()</code> e <code>clienteService.getAll()</code> para listar as Outorgas do cliente de acordo com o usuário logado serão listadas todas as Licenças Ambientais cadastradas e seus respectivos Clientes.
criaFormulario(): Neste método é instanciado o formulário através do <code>formBuild</code> , onde o <code>Validators.required</code> é utilizado para validação dos campos.
get(): Com o método <code>get()</code> , são capturados os <code>input's</code> dos campos do formulário.
acao(): Neste método é verificado se a ação é a de Criação ou Atualização de Outorga, onde o controle da ação é realizado através dos botões <code>pageTitle</code> e <code>botaoTitle</code> .
cadastraListaGlobal(): Utiliza-se o <code>outorgaService.init()</code> para carregar a página e o <code>outorgaService.createGlobal()</code> para listar os clientes cadastrados no sistema para que seja possível selecionar o Cliente para realizar o novo cadastro da Outorga.
atualizaListaGlobal(): Utiliza-se o <code>outorgaService.update()</code> para atualizar as Outorgas dos clientes cadastrados no sistema, sendo possível através do menu individual, editar informações dos campos cliente, descrição, vencimento da Outorga e arquivo selecionado. Este recurso é acessível apenas pelo usuário com perfil de administrador.
onSubmit(): Responsável por enviar os dados do formulário para o banco de dados.
openGalery(): Método responsável por acessar o arquivo desejado da galeria. Este recurso é acessível apenas pelo usuário com perfil de administrador.
uploadFileTo(): Método responsável por atualizar/substituir arquivo do cadastro. Este recurso é acessível apenas pelo usuário com perfil de administrador.

Página Licença Ambiental – Estruturado com as páginas cria-la e lista-la e o componente lista-la-item.

ngOnInit(): Responsável por carregar a página de Licença Ambiental, invoca os métodos <code>criaFormulario()</code> , <code>clienteService.init()</code> e <code>clienteService.getAll()</code> para listar as Licenças Ambientais de acordo com o usuário logado. Caso, o usuário possua o perfil de administrador, serão listadas todas as Licenças Ambientais cadastradas e seus respectivos Clientes.
criaFormulario(): Neste método é instanciado o formulário através do <code>formBuild</code> , onde o <code>Validators.required</code> é utilizado para validação dos campos.
get(): Com o método <code>get()</code> , são capturados os input's dos campos do formulário.
acao(): Neste método é verificado se a ação é a de Criação ou Atualização das Licenças Ambientais, onde o controle da ação é realizado através dos botões <code>pageTitle</code> e <code>botaoTitle</code> .
cadastraListaGlobal(): Utiliza-se o <code>outorgaService.init()</code> para carregar a página e o <code>outorgaService.createGlobal()</code> para listar os clientes cadastrados no sistema para que seja possível selecionar o Cliente para realizar o novo cadastro da Outorga.
atualizaListaGlobal(): Utiliza-se o <code>licencaAmbientalService.update()</code> para atualizar as Licenças Ambientais dos clientes cadastrados no sistema, sendo possível através do menu individual, editar informações dos campos cliente, descrição, vencimento da Licença Ambiental e arquivo selecionado. Este recurso é acessível apenas pelo usuário com perfil de administrador.
onSubmit(): Responsável por enviar os dados do formulário para o banco de dados.
openGalery(): Método responsável por acessar o arquivo desejado da galeria. Este recurso é acessível apenas pelo usuário com perfil de administrador.
uploadFileTo(): Método responsável por atualizar/substituir arquivo do cadastro. Este recurso é acessível apenas pelo usuário com perfil de administrador.
deleteFile(): Método responsável pela exclusão de arquivos inseridos nos cadastros das licenças.

Página Declaração Ambiental – Estruturado com as páginas cria-da e lista-da e o componente lista-da-item.

ngOnInit(): Responsável por carregar a página de Declaração Ambiental, invoca os métodos `criaFormulario()`, `clienteService.init()` e `clienteService.getAll()` para listar as Declarações Ambientais do cliente de acordo com o usuário logado. Caso, o usuário possua o perfil de administrador, serão listadas todas as Declarações Ambientais cadastradas e seus respectivos Clientes.

criaFormulario(): Neste método é instanciado o formulário através do `formBuild`, onde o `Validators.required` é utilizado para validação dos campos.

get(): Com o método `get()`, são capturados os input's dos campos do formulário.

acao(): Neste método é verificado se a ação é a de Criação ou Atualização das Declarações Ambientais, onde o controle da ação é realizado através dos botões `pageTitle` e `botaoTitle`.

cadastraListaGlobal(): Utiliza-se o `declaracaoAmbientalService.init()` para carregar a página e o `declaracaoAmbientalService.createGlobal()` para listar os clientes cadastrados no sistema para que seja possível selecionar o Cliente para realizar o novo cadastro de Declaração Ambiental.

AtualizaListaGlobal(): Utiliza-se o `declaracaoAmbientalService.update()` para atualizar as Declarações Ambientais dos clientes cadastrados no sistema, sendo possível através do menu individual, editar informações dos campos cliente, descrição, vencimento da Declaração Ambiental e arquivo selecionado. Este recurso é acessível apenas pelo usuário com perfil de administrador.

onSubmit(): Responsável por enviar os dados do formulário para o banco de dados.

openGalery(): Método responsável por acessar o arquivo desejado da galeria. Este recurso é acessível apenas pelo usuário com perfil de administrador.

uploadFileTo(): Método responsável por atualizar/substituir arquivo do cadastro. Este recurso é acessível apenas pelo usuário com perfil de administrador.

deleteFile(): Método responsável pela exclusão de arquivos inseridos nos cadastros das licenças.

Página Cadastro Ambiental Rural– Estruturado com as páginas cria-car e lista-car e o componente lista-car-item.

ngOnInit(): Responsável por carregar a página de Cadastro Ambiental Rural, invoca os métodos <code>criaFormulario()</code> , <code>clienteService.init()</code> e <code>clienteService.getAll()</code> para listar os Cadastros Ambiental Rural do cliente de acordo com o usuário logado. Caso, o usuário possua o perfil de administrador, serão listados todos os Cadastros Ambiental Rural e seus respectivos Clientes.
criaFormulario(): Neste método é instanciado o formulário através do <code>formBuild</code> , onde o <code>Validators.required</code> é utilizado para validação dos campos.
get(): Com o método <code>get()</code> , são capturados os <code>input's</code> dos campos do formulário.
acao(): Neste método é verificado se a ação é a de Criação ou Atualização do Cadastro Ambiental rural, onde o controle da ação é realizado através dos botões <code>pageTitle</code> e <code>botaoTitle</code> .
cadastraListaGlobal(): Utiliza-se o <code>cadastroRuralAmbientalService.init()</code> para carregar a página e o <code>cadastroRuralAmbientalService.createGlobal()</code> para listar os clientes cadastrados no sistema para que seja possível selecionar o Cliente para realizar o novo cadastro Ambiental Rural.
AtualizaListaGlobal(): Utiliza-se o <code>cadastroRuralAmbientalService.update()</code> para atualizar os Cadastros Ambiental Rural dos clientes cadastrados no sistema, sendo possível através do menu individual, editar informações dos campos cliente, descrição, vencimento do Cadastro Ambiental Rural e arquivo selecionado. Este recurso é acessível apenas pelo usuário com perfil de administrador.
onSubmit(): Responsável por enviar os dados do formulário para o banco de dados.
openGalery(): Método responsável por acessar o arquivo desejado da galeria. Este recurso é acessível apenas pelo usuário com perfil de administrador.
uploadFileTo(): Método responsável por atualizar/substituir arquivo do cadastro. Este recurso é acessível apenas pelo usuário com perfil de administrador.
deleteFile(): Método responsável pela exclusão de arquivos inseridos nos cadastros das licenças.

Página Cadastro Crédito Financeiro – Estruturado com as páginas credito e lista-credito e o componente lista-credito-item.

ngOnInit(): Responsável por carregar a página de Cadastro Crédito Financeiro, invoca os métodos <code>criaFormulario()</code> , <code>clienteService.init()</code> e <code>clienteService.getAll()</code> para listar as linhas de Créditos Financeiros do cliente de acordo com o usuário logado. Caso, o usuário possua o perfil de administrador, serão listadas todas as linhas de créditos e seus respectivos Clientes.
criaFormulario(): Neste método é instanciado o formulário através do <code>formBuild</code> , onde o <code>Validators.required</code> é utilizado para validação dos campos.
get(): Com o método <code>get()</code> , são capturados os input's dos campos do formulário.
acao(): Neste método é verificado se a ação é a de Criação ou Atualização dos Créditos Financeiros, onde o controle da ação é realizado através dos botões <code>pageTitle</code> e <code>botaoTitle</code> .
cadastraListaGlobal(): Utiliza-se o <code>creditoService.init()</code> para carregar a página e o <code>creditoService.createGlobal()</code> para listar os clientes cadastrados no sistema para que seja possível selecionar o Cliente para realizar o novo cadastro de Crédito Financeiro.
AtualizaListaGlobal(): Utiliza-se o <code>creditoService.update()</code> para atualizar as linhas de crédito dos clientes cadastrados no sistema, sendo possível através do menu individual, editar informações dos campos cliente, descrição, data de aprovação do crédito, data de expiração do crédito, valor do crédito e arquivo selecionado. Este recurso é acessível apenas pelo usuário com perfil de administrador.
obter_dt_aprovacao_credito(): Método invocado para captura da data de aprovação do crédito financeiro. A data capturada, será utilizada para tratamento de exceções no momento do envio dos dados para o banco de dados.
obter_dt_expiracao_credito(): Método invocado para captura da data de expiração do crédito financeiro. A data capturada, será utilizada para tratamento de exceções no momento do envio dos dados ao banco de dados.
onSubmit(): Responsável por enviar os dados do formulário para o banco de dados.
openGalery(): Método responsável por acessar o arquivo desejado da galeria. Este recurso é acessível apenas pelo usuário com perfil de administrador.
uploadFileTo(): Método responsável por atualizar/substituir arquivo do cadastro. Este recurso é acessível apenas pelo usuário com perfil de administrador.
deleteFile(): Método responsável pela exclusão de arquivos inseridos nos cadastros das licenças.

deletePicture(): Método responsável pela exclusão do arquivo de foto que fora substituído no perfil do usuário.
deletePicturePasta(): Método responsável pela exclusão do arquivo de foto temporário do sistema.
deleteArquivosPasta(): Método responsável pela exclusão dos arquivos do Cliente.
deletaArquivosBotao(): Método responsável por deletar arquivos de IR e CNH.
clearField(): Método utilizado para limpar os caracteres obscuros do campo de senha ao ser clicado.
verificarSenha(): Método utilizado para reescrever os caracteres obscuros do campo de senha ao retirar o cursor.

Páginas de Relatórios:

ngOnInit(): Método responsável por carregar a página de login para o usuário.
closeModal(): Método responsável por abrir a janela do filtro do relatório.
clearFilters(): Método responsável por limpar os campos do filtro do relatório.
obter_id_cliente(): Método responsável por obter o id do cliente selecionado.
formatarData(): Método responsável por formatar a data no padrão dia/mês/ano.
registroDataHora(): Método responsável por gerar data e hora no rodapé da página do relatório.
obter_dt_nascimento_cliente_ini(): Método responsável por obter a data inicial do filtro do relatório.
obter_dt_nascimento_cliente_fim(): Método responsável por obter a data final do filtro do relatório.
relatorio_cli(): Método responsável por gerar o relatório do filtro selecionado pelo usuário.
buildTableBody(): Método responsável por instanciar a tabela do relatório.
table(): Método responsável por definir o escopo da tabela.
exportPdf(): Método responsável pela configuração da página do relatório.