**Passo a Passo – APP CONSISTE**

**CONSISTE + XTRBrasil + Instalações Consiste**

Procedimento Item 10

Trello – Organizador de tarefas

https://trello.com/b/i2zGkvcl/eng-de-dados

Interface gráfica do usuário, Texto, Site

Descrição gerada automaticamente

Revisão - coloco depois de falar com Daniel Assis – quando der algum problema que não conseguiu resolver.

To do – pega o projeto leva para seu nome (arrastar) definindo que este projeto será trabalhado por vc

Interface gráfica do usuário, Texto, Aplicativo, chat ou mensagem de texto

Descrição gerada automaticamente

Arrastar quando terminar – Concluído

Ao entregar três informações

1. Qual é código operação a ser usado ED-001 de acordo com o instrução iT-ED-001
2. Necessário Reoperar – vai para operação ou vai fazer check list? OU Não necessário Reoperar
3. O que foi feito.

Ex:

* Instrução de trabalho usada: OP-001
* Necessário reoperar a carga: SIM
* O que foi feito no app: “informar o que foi feito”

Interface gráfica do usuário, Texto, Aplicativo, chat ou mensagem de texto

Descrição gerada automaticamente

Exemplo: e734 Agentes Econômicos Fiscalizados pela ANP - DPC 10/02/2023

<https://gitlab.consiste.com.br/consiste>

Fluxo – Pegar GitLab o Programa pai

XTR Content Modelos – onde pego os dos Migradores

XTR Content – Pasta que coloco os códigos

**Seguir a Instrução de trabalho +**

Formulário de CARGA: contém dados temporais errados - Não edita

* **Carregar os documentos que vem da CURADORIA:**

[\\thor.consiste.com.br\sys\migra\XTRBrasilCuradoria](file:///\\thor.consiste.com.br\sys\migra\XTRBrasilCuradoria)

Copiar o documentos CSV e Excel para pasta local onde o projeto será trabalhado

Criar uma pasta eXXXXArquivos e colar os arquivos

* **Abrir pasta para iniciar o projeto + clonar modelo do GITLAB:**

**OBS: sempre do XTR content**

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

1. **Em - Consiste Consultoria e Sistemas Ltda – Grupo do GITLAB entre na pasta XTR Content Modelos – selecione o modelo de acordo com o trabalho a ser realizado – Nome da Entidade**

Detectando que se trata do ED 001 – CSV

|  |  |
| --- | --- |
|  | Conjunto de pastas que identificam ser uma entidade de CSV – ED 001 |

|  |  |
| --- | --- |
| Interface gráfica do usuário, Texto, Aplicativo, Email  Descrição gerada automaticamente | 1. Clonar migrador-CSV no gitLab da Consiste -> Playground (no gitlab escolha o migrador – entre – clone - clone with SSH) 2. Abrir o Git Bash na pasta do arquivo e clonar o modelo para máquina local. 3. COMANDO - git clone “nome do arquivo modelo CLONADO” |

* **Pra Mudar Branch para develop – git bash na pasta do arquivo**

1. Cd “nome da pasta do arquivo”
2. Git checkout develop

* **Excluir a pasta .git – oculta**

1. No Explore – VIEW (barra principal) – Show – Hidden Items
2. Irá aparecer o arquivo .GIT
3. Deletar o arquivo .git – com isso vc quebra a Branch com o GITLAB

* **Iniciar ajustes do APP**
  + Pasta Cycle-controller – editar os aquivos que vieram do modelo: e yml - pad++

1. -play.sh
   * + indice eXXXX – colocar o nome do app
2. yml
   * + Editar o nome do arquivo – eXXX – colocar o nome do APP

**Ajustes pasta files**

1. Backup – Preparar um histórico do arquivo original
   1. Cole o arquivo original dos dados
   2. Para não ocupar muito espaço selecione todas as linhas EXCETO as 10 primeiras e apague - fazer o um backup das 10 primeiras linhas (apagar o restante) – para ter uma memória de como estava no início o arquivo
2. dataIngestorAssets – sem ação ...\files\dataIngestorAssets\log
3. dataNormalizadorAssets – sem ação
4. Input
5. Arquivos JSON
   1. Metadata.json – retirar dados do formulário de carga:

Exemplo:

"linkorigem": "https://www.gov.br/anp/pt-br/...-abertos",

"origem": "ANP",

"meta\_qtd\_e734": 1

* 1. Controller.json – "elasticSearchServer" – deve ser chaveado com "REVIEW" ou “PRODUCTION”

"bulksize": 100 - mostra a quantidade de linhas que via ser carregada por vez – testar de 100 a 1000, de acordo com o tamanho do arquivo.

* 1. fieldsByType.json - Deve ser copiado todas os nomes das colunas e classificar os tipos que podem ser:

CEP, CNPJ, CONTROLE,CPF,DATA, MUNICIPIO, NUMERO, TEXTO, UF

***OBS: Números que não serão utilizados para algum tipo de cálculo devem ser considerados “TEXTO”***

tratarCamposDoTipoTEXTO(docOut);

tratarCamposDoTipoCEP(docOut);

tratarCamposDoTipoLongitude(docOut);

tratarCamposDoTipoLatitude(docOut);

tratarCamposDoTipoCPF(docOut);

tratarCamposDoTipoCNPJ(docOut);

criarCampoGeoHash(docOut);

tratarCamposDoTipoDATA(docOut);

tratarCamposDoTipoEMAIL(docOut);

tratarCamposDoTipoNUMERO(docOut);

tratarCamposDoTipoString(docOut);

consultaMuni(docOut);

* 1. Input.json

Carga – ver no origem – entidade big – número de carga (ver o que não está em produção)

Entidade: eXXXX

Encoding:

1. Source – colocar o arquivo .CSV original para arquivo
2. Pasta Ingestor – Importar para o eclipse (caso JAVA)
   1. Editados os comandos para execução do APP

BusinessRules

Exemplo:

Pegar as colunas e associar conforme o que está no XTR Entidades BIG – Chamar o programa

e parear conforme o que se encontra na entidade – Lapis (editar) - Guia

    colunasNovas = {'UF': 'uf',

                    'Município': 'municipio',

                    'Bairro': 'bairro',

                    'Endereço': 'endereco',

                    'CNPJ/CPF': 'cnpjoucpf',

                    'Agente Econômico': 'agente\_economico',

                    'Segmento Fiscalizado': 'segmento\_fiscalizado',

                    'Data DF': 'data\_fiscalizacao',

                    'Procedimento de Fiscalização': 'procedimento\_fiscalizacao',

                    'Resultado': 'resultado',

                    'Número do Documento': 'numero\_documento',

                    }

1. Source – colocar o arquivo .CSV original para arquivo

**Elastic**

<http://xtrdb01.consiste.com.br:5601/s/dev-e-manager/app/dev_tools#/console>

Tela de celular com aplicativo aberto

Descrição gerada automaticamente

Para fazer os ajustes ou base nova verificar o que existe no Elastic sobre a entidade, com o seguinte comando: (eXXXX\_cX = numero e versão da entidade).

***GET eXXXX\_cX /\_search***

Caso a entidade já tenha dados e seja necessário ajustes no Banco de Dados de teste (Elastic) delete os dados para nova atualização.

***DELETE eXXXX\_Cx***

Outros Comandos:

Contar dados:

***GET eXXXX\_cX/\_count***

***{***

***"query": {***

***"match": {***

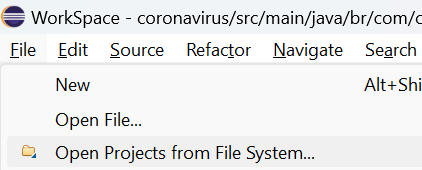
***"periodo": "2020"***

***}***

***}***

***}***

Uma vez apagado os dados no BD no Elastic entre no ECLIPCE IDE para fazer os ajustes necessários no código:



Editar app no JAVA

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

O que deve ser verificado está no Atualizadorpadrão & DataTranspoter.java

Deve ser enviado o Runnable JAR

Fele/export:

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

Criar pasta de entrega

Rodar Pasta como no eclipse pelo bash: java -Dfile.encoding=UTF-8 -jar eXXXX.jar

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamenteREMOVER Tudo que for git e .log - tirar

Controller.json - Mudar - "elasticSearchServer": "PRODUCTION",

checar se está subindo para o XTROrigem do DB01

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

Testar se está tudo ok: no XTRBrasil

Interface gráfica do usuário, Aplicativo, Word

Descrição gerada automaticamentesempre o que tem asterisco (é o que trabalhamos)

Para testar o XTR deve pegar do DB01 que tem Porta final “9200”.

<http://xtrdb01.consiste.com.br:9200/>

salvar

*Devem ser chacados:*

Municípios / estado / ano – procurar inconsistências

Contar item do documentos DB1: GET eXXXX\_cX/\_count

Para desfazer a mudança feita acima.

apagar

<http://xtrdb01.consiste.com.br:9200/>

salvar

ENTREGA

No trello tem o check list

DB1 quantidade de dados = planihas

NA PASTA DE ENTREGA  
Controller.json - Mudar - "elasticSearchServer": "PRODUCTION",

|  |  |  |  |
| --- | --- | --- | --- |
|  | Dentro vazia |  | REMOVER Tudo que for git e .log – tirar  Backup-vazia |

Copiar o .JAR, pasta INPUT e OUTPUT e LOG para a pasta XTROperacoes no Thor

[\\thor.consiste.com.br\sys\migra\XTRBrasilOperacoes\e734\files\input](file:///\\thor.consiste.com.br\sys\migra\XTRBrasilOperacoes\e734\files\input)

Trello

SELCIONAR a entidade adicionar o que foi feito:

* Instrução de trabalho usada: OP-001
* Necessário reoperar a carga: SIM
* O que foi feito no app: “descrever o que foi feito”

clon

*VERSIONAMENTO*

1. Git lab selecionar o local onde será colocada a entidade pronta.

<https://gitlab.consiste.com.br/consiste>

1. em Search by Name – coloque no nome da entidade para encontrar o local para fazer o versionamento

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

1. Click na Entidade e faça - Clone with SSH
2. No CPU local iniciar o processo de vercionamento
   1. git clone [git@gitlab.consiste.com.br:consiste/xtr-content/agentes-economicos-fiscalizados-pela-anp.git](mailto:git@gitlab.consiste.com.br:consiste/xtr-content/agentes-economicos-fiscalizados-pela-anp.git)

Git clone “CODIGO CLONADO”

* 1. Entre na pasta criada

Cd “nome da pasta clonada”

* 1. $ git checkout develop

Mudar a Branch para develop

* 1. Jogar os documentos da entidade para dentro do novo arquivo pelo explorer e - $ git status

Checar os documentos a serem commitados

* 1. $ git add .

Adiciona todos os documentos ao Git

* 1. $ git commit -m "agente-economicos-fiscalizados-pela-anp"

Commita todos os documentos adicionados

$ git git push --set-upstream origin develop

Empurra os documentos para o GITLAB

1. No GITLAB verificar se os documentos foram adicionados.
2. Interface gráfica do usuário, Texto, Aplicativo, Email

   Descrição gerada automaticamente

Short Cut

Methods -> Ctrl + space

Identar Ctrl shift F