

Instalando o e-cidade a partir dos fontes no Linux

O e-cidade destina-se a informatizar a gestão dos Municípios Brasileiros de forma integrada. Esta informatização contempla a integração entre os entes municipais: Prefeitura Municipal, Câmara Municipal, Autarquias, Fundações e outros. A economia de recursos é somente uma das vantagens na adoção do e-cidade, além da liberdade de escolha dos fornecedores e garantia de continuidade do sistema, uma vez apoiado pelo Ministério do Planejamento.

1. Instalação do Ubuntu Server 16.04.x LTS

O Ubuntu Server é a versão do sistema operacional da Canonical voltada para servidores. Como o uso em servidores exige um tempo maior de suporte, a versão indicada para o uso com esta documentação é a Ubuntu Server 16.04.x LTS.

1.1 Obtendo a mídia de instalação

A mídia de instalação pode ser obtida diretamente do site da Canonical e gravada em CD, DVD ou pendrive. No site <http://www.ubuntu.com/download/server> esta disponível a versão mais recente, mas utilizaremos a versão 16.04.x LTS disponível em <http://releases.ubuntu.com/xenial/> (ubuntu-16.04.2-server-amd64.iso) por questões de compatibilidade com o produto.

1.2 Instalando o Ubuntu Server 16.04.x LTS

O processo de instalação do sistema operacional não é coberto por esta documentação, mas recomendamos a instalação mínima padrão, em “pt_BR” Português do Brasil, somente com os softwares “standard system utilities” e “OpenSSH server”.

Dica: Maiores informações sobre a instalação e personalização do sistema operacional podem ser obtidas em: <https://help.ubuntu.com/16.04/serverguide/>

1.3 Editando os arquivos de configuração

Durante o processo de configuração descrito nesta documentação, é necessário editar e modificar diversos arquivos de texto. Para realizar estas modificações, sugerimos a utilização do editor “nano”, mas você pode usar qualquer outro editor de texto de sua preferência.

Dica: Mais informações sobre o editor de textos “nano” podem ser obtidas em: <http://www.nano-editor.org/>.

2. Instalação do PostgreSQL 9.5.x

PostgreSQL é um sistema gerenciador de banco de dados objeto relacional (SGBDOR), desenvolvido como projeto de código aberto sem custo de licença. Este é o programa que vai gerenciar e armazenar o banco de dados relacional utilizado pelo e-cidade.

Dica: Mais informações sobre o PostgreSQL podem ser obtidas em: <http://www.postgresql.org/>.

2.1 Configurando o repositório

Para instalar o PostgreSQL 9.5.x e suas dependências, utilizaremos o gerenciador de pacotes **apt**, que nos prove as últimas atualizações dos pacotes requeridos através de um canal seguro e confiável.

```
[linux]$ sudo touch /etc/apt/sources.list.d/pgdg.list
[linux]$ echo "deb http://apt.postgresql.org/pub/repos/apt/ xenial-pgdg main" | sudo tee
/etc/apt/sources.list.d/pgdg.list
```

Dica: Mais informações sobre a configuração do repositório podem ser obtidas em: <https://www.postgresql.org/download/linux>

```
[linux]$ sudo apt-key adv --keyserver keyserver.ubuntu.com --recv ACC4CF8
```

Dica: Mais informações sobre o gerenciador de pacotes “apt” podem ser obtidas em: <https://help.ubuntu.com/16.04/servergui/management.html>

2.2 Ajustando o idioma

Atenção: Este procedimento só é necessário se você não instalou o sistema operacional em “pt_BR” - Português do Brasil.

```
[linux]$ sudo apt -y install language-pack-gnome-pt language-pack-pt-base myspell-pt myspell-pt-br
wbrazilian wportuguese
```

2.3 Ajustando o encoding

2.3.1 Configurando e regerando o ‘locale’ Para o servidor do banco de dados, é necessário alterar a ordenação do locale, para realizar esta tarefa, edite o arquivo de configuração com o comando abaixo:

```
[linux]$ sudo nano /usr/share/i18n/locales/pt_BR
```

Localize o bloco de configuração LC_COLLATE e END LC_COLLATE, e adicione dentro do bloco as seguintes linhas:

```
reorder-after <U00A0>
<U0020><CAP>;<CAP>;<CAP>;<U0020>
reorder-end
```

2.3.2 Regerando o ‘locale’ Para criação do cluster do e-cidade precisamos ter o locale pt_BR ISO-8859-1 disponível no sistema, para regerar utilize os comandos abaixo.

```
[linux]$ sudo localedef -i pt_BR -c -f ISO-8859-1 -A /usr/share/locale/locale.alias pt_BR
[linux]$ sudo locale-gen pt_BR
[linux]$ sudo dpkg-reconfigure locales
```

2.4 Instalando o PostgreSQL 9.5

Para realizar a instalação do PostgreSQL 9.5 através do gerenciador de pacotes execute os comandos:

```
[linux]$ sudo apt update
[linux]$ sudo apt -y install postgresql-9.5 postgresql-client-9.5 postgresql-contrib-9.5
```

2.5 Criando e configurando o cluster

O e-cidade utiliza encoding LATIN1 (ISO-8859-1) no cluster do PostgreSQL 9.5.x. Por isso vamos parar, remover e recriar o cluster que será utilizado pela aplicação.

Dica: Mais informações sobre a configuração do cluster: <http://www.postgresql.org/docs/9.5/static/sql-cluster.html>

```
[linux]$ sudo pg_dropcluster --stop 9.5 main
```

```
[linux]$ sudo pg_createcluster -u postgres -g postgres -e LATIN1 --locale="pt_BR.ISO-8859-1"
--lc-collate="pt_BR.ISO-8859-1" 9.5 ecidade
```

```
[linux]$ sudo nano /etc/postgresql/9.5/ecidade/pg_hba.conf
```

Localize e altere as linhas do arquivo de configuração que não estão comentadas (# na frente da linha), substituindo os valores correspondentes por **trust**, conforme a tabela abaixo.

| | | | | |
|-------|-----|----------|--------------|--------------|
| local | all | postgres | | trust |
| local | all | all | | trust |
| host | all | all | 127.0.0.1/32 | trust |
| host | all | all | :::1/128 | trust |

```
[linux]$ sudo nano /etc/postgresql/9.5/ecidade/postgresql.conf
```

Atenção: Os demais parâmetros, não descritos aqui, não precisam ser alterados.

Localize e altere as linhas do arquivo de configuração, descomentando-as se necessário (remover o caractere # do início da linha), substituindo os valores correspondentes conforme a indicação abaixo.

```
listen_addresses = '*'
max_connections = 20
bytea_output = 'escape'
max_locks_per_transaction = 256
default_with_oids = on
```

```
escape_string_warning = off
standard_conforming_strings = off
```

```
[linux]$ sudo systemctl start postgresql.service
```

```
[linux]$ psql -U postgres -h localhost -l
```

A saída do comando deve ser parecida com os valores abaixo.

| Nome | Dono | Codificação | Collate | Ctype | Privilégios de acesso |
|-----------|----------|-------------|------------------|------------------|-----------------------|
| postgres | postgres | LATIN1 | pt_BR.ISO-8859-1 | pt_BR.ISO-8859-1 | |
| template0 | postgres | LATIN1 | pt_BR.ISO-8859-1 | pt_BR.ISO-8859-1 | =c/postgres + |
| | | | | | postgres=CTc/postgres |
| template1 | postgres | LATIN1 | pt_BR.ISO-8859-1 | pt_BR.ISO-8859-1 | =c/postgres + |
| | | | | | postgres=CTc/postgres |

2.6 Criando a estrutura de banco de dados para o e-cidade

Vamos criar os usuários necessários para o funcionamento do e-cidade através dos comandos:

Atenção: Em ambientes de produção, nunca utilize senhas que tenham o mesmo nome da conta de usuário. Mais informações sobre segurança de senhas podem ser obtidas em: <http://cartilha.cert.br/senhas/>.

```
psql -U postgres -c "CREATE ROLE ecidade WITH SUPERUSER LOGIN PASSWORD 'ecidade';"
psql -U postgres -c "CREATE ROLE plugin WITH LOGIN PASSWORD 'plugin';"
psql -U postgres -c "CREATE ROLE dbseller WITH LOGIN PASSWORD 'dbseller';"
```

Em seguida, utilize o comando abaixo para criar o banco de dados:

```
psql -U postgres -c "CREATE DATABASE ecidade OWNER ecidade;"
```

3. Instalação do Apache

Apache é o servidor web mais usado em sistemas Linux. Servidores web são usados para servir páginas Web que, normalmente, são solicitadas pelos computadores clientes através de navegador, como o Firefox, Chromium, etc.

Dica: Mais informações sobre o Apache podem ser obtidas em: <http://www.apache.org/>.

3.1 Instalando o Apache

Para realizar a instalação do servidor web Apache através do gerenciador de pacotes, execute o comando:

```
[linux]$ sudo apt -y install apache2
```

3.2 Configurando a diretiva VirtualHost

```
[linux]$ sudo cp -a /etc/apache2/sites-available/000-default.conf  
/etc/apache2/sites-available/000-default.conf.dist
```

```
[linux]$ sudo nano /etc/apache2/sites-available/000-default.conf
```

Atenção: Os demais parâmetros, não descritos aqui, não precisam ser alterados.

Adicione as linhas que não existirem e edite as existentes dentro do bloco **VirtualHost**, conforme é exibido abaixo.

```
<VirtualHost *:80>  
...  
LimitRequestLine 16382  
LimitRequestFieldSize 16382  
Timeout 12000  
AddDefaultCharset ISO-8859-1  
SetEnv no-gzip 1  
<Directory /var/www/html>  
    Options -Indexes +FollowSymLinks +MultiViews  
    AllowOverride All  
    Require all granted  
</Directory>  
...  
</VirtualHost>
```

Dica: Mais informações sobre o VirtualHost podem ser obtidas em: <http://httpd.apache.org/docs/current/vhosts/examples.h>

4. Instalação do PHP 5.6 e PHP-FPM

PHP é uma linguagem interpretada livre, usada originalmente apenas para o desenvolvimento de aplicações presentes e atuantes no lado do servidor, capazes de gerar conteúdo dinâmico na World Wide Web.

O PHP-FPM é uma alternativa ao PHP FastCGI com muitas funcionalidades voltadas a sites com grande número de acessos e carga.

Dica: Mais informações sobre o PHP e PHP-FPM podem ser obtidas em: <http://www.php.net/> e <http://php-fpm.org/>

4.1 Configurando o repositório

Para instalar o PHP 5.6 e suas dependências, utilizaremos o gerenciador de pacotes **apt**, que nos prove as últimas atualizações dos pacotes requeridos através de um canal seguro e confiável.

```
[linux]$ sudo apt-add-repository -y ppa:ondrej/php  
[linux]$ sudo apt update
```

4.2 Instalando o PHP 5.6 e o PHP-FPM

Para realizar a instalação do PHP 5.6 através do gerenciador de pacotes, execute o comando:

```
[linux]$ sudo apt -y install php5.6 php5.6-bcmath php5.6-bz2 php5.6-cli php5.6-common php5.6-curl php5.6-gd  
php5.6-interbase php5.6-json php5.6-mbstring php5.6-mcrypt php5.6-pgsql php5.6-soap php5.6-sqlite3  
php5.6-xml php5.6-xmlrpc php5.6-zip php5.6-fpm
```

4.3 Configurando o PHP 5.6 e o PHP-FPM

```
[linux]$ sudo a2enmod proxy_fcgi setenvif rewrite
```

```
[linux]$ sudo a2enconf php5.6-fpm
```

```
[linux]$ sudo touch /var/log/php_errors.log  
[linux]$ sudo chown www-data:www-data /var/log/php_errors.log
```

```
[linux]$ sudo nano /etc/php/5.6/fpm/php.ini
```

Atenção: Os demais parâmetros, não descritos aqui, não precisam ser alterados.

Localize e altere as linhas do arquivo de configuração, descomentando-as se necessário (remover o caractere ; do início da linha), substituindo os valores correspondentes conforme a indicação abaixo.

```
short_open_tag = On
```

```
[linux]$ sudo systemctl restart php5.6-fpm.service  
[linux]$ sudo systemctl restart apache2.service
```

5. Instalação do LibreOffice

O LibreOffice é uma suite de escritório livre compatível com as principais suítes de escritório do mercado. Oferece todas as funções esperadas de uma suite profissional: editor de textos, planilha, apresentação, editor de desenhos e banco de dados. No e-cidade, o LibreOffice é utilizado para exportar documentos baseados em modelos de formulários.

Dica: Mais informações sobre o LibreOffice podem ser obtidas em: <https://pt-br.libreoffice.org/>.

5.1 Instalando o LibreOffice

Para realizar a instalação do LibreOffice através do gerenciador de pacotes, execute o comando:

```
[linux]$ sudo apt -y install libreoffice-writer ttf-mscorefonts-installer
```

6. Instalação do e-cidade

6.1 Obtendo o pacote de instalação

O pacote de instalação pode ser obtido diretamente do site do Portal do Software Público. No endereço da comunidade do e-cidade (<https://softwarepublico.gov.br/social/e-cidade>) estão disponíveis as últimas versões e suas atualizações. Como estamos realizando uma nova instalação, você deve baixar o pacote **completo**, um exemplo de nome do pacote seria: `e-cidade-x.x.xx-linux.completo.tar.bz2`.

6.2 Configurando o servidor

```
[linux]$ sudo useradd -d /home/dbseller -g www-data -G sudo,adm,cdrom,dip,plugdev -k /etc/skel -m -s  
/bin/bash dbseller
```

```
[linux]$ sudo passwd dbseller
```

Dica: Neste exemplo, defina a senha como o mesmo nome de usuário, como “dbseller”.

Atenção: Em ambientes de produção, nunca utilize senhas que tenham o mesmo nome da conta de usuário. Mais informações sobre segurança de senhas podem ser obtidas em: <http://cartilha.cert.br/senhas/>.

```
[linux]$ sudo nano /etc/login.defs
```

Atenção: Os demais parâmetros, não descritos aqui, não precisam ser alterados.

Localize e altere a linha do arquivo de configuração, descomentando-a se necessário (remover o caractere # do início da linha), substituindo o valor correspondente conforme a indicação abaixo.

```
UMASK          002
```

Em seguida, edite o arquivo de configuração de variáveis do apache com o comando abaixo:

```
[linux]$ sudo nano /etc/apache2/envvars
```

Adicione a linha abaixo no final do arquivo.

```
umask 002
```

6.3 Instalando o e-cidade

```
[linux]$ tar xjf e-cidade-x.x.xx-linux.completo.tar.bz2
```

```
[linux]$ sudo cp -r e-cidade-x.x.xx-linux.completo/e-cidade /var/www/html
```

6.4 Configurando o e-cidade

```
[linux]$ sudo cp /var/www/html/e-cidade/libs/db_conn.php.dist /var/www/html/e-cidade/libs/db_conn.php
```

Utilize o seguinte comando para editar o arquivo de configuração:

```
[linux]$ sudo nano /var/www/html/e-cidade/libs/db_conn.php
```

Em seguida, atualize as configurações conforme o exemplo abaixo.

```
$DB_USUARIO = "ecidade";  
$DB_SENHA = "ecidade";  
$DB_SERVIDOR = "localhost";  
$DB_PORTA = "5432";  
$DB_PORTA_ALT = "5432";  
$DB_BASE = "ecidade";
```



```
[linux]$ sudo cp /var/www/html/e-cidade/config/plugins.json.dist /var/www/html/e-cidade/config/plugins.json
```

Utilize o seguinte comando para editar o arquivo de configuração:

```
[linux]$ sudo nano /var/www/html/e-cidade/config/plugins.json
```

Em seguida, atualize as configurações conforme o exemplo abaixo.

```
{
  "AcessoBase" : {
    "usuario" : "plugin",
    "senha" : "plugin"
  }
}
```

```
[linux]$ sudo nano /var/www/html/e-cidade/libs/config.mail.php
```

Em seguida, atualize as configurações, com os dados do seu servidor de e-mail, conforme o exemplo abaixo.

```
$sClass = 3;
$sSsl = ''; // qual o tipo de ssl q sera utilizados ( ex: tls), por padrao é vazio
$bAuth = true; // autenticacao habilitada
$sUser = 'exemplo@exemplo.com.br';
$sPass = 'senhaexemplo';
$sHost = 'smtp.exemplo.com.br';
$sPort = '25';
```

```
[linux]$ psql -U ecidade -d ecidade -f e-cidade-x.x.xx-linux.completo/sql/e-cidade-x.x.xx.sql 2>
/tmp/erros.txt
```

Após a carga, execute o comando abaixo para otimizar as consultas:

```
[linux]$ psql -U ecidade -d ecidade -c "VACUUM ANALYZE VERBOSE;"
```

```
[linux]$ sudo chown -R dbseller.www-data /var/www/html/e-cidade
[linux]$ sudo chmod -R 775 /var/www/html/e-cidade
[linux]$ sudo chmod -R 777 /var/www/html/e-cidade/tmp
```

6.5 Agendando a migração da auditoria

```
[linux]$ crontab -e
```

Em seguida, adicione o seguinte bloco ao final do arquivo:

```
# DBSeller - Agendamento do script para migracao do account para nova estrutura "db_auditoria"
*/30 * * * * cd /var/www/html/e-cidade/integracao_externa/auditoria; ./auditoria_migracao.sh 0 10 1 >
log/auditoria_migracao.log 2>&1
*/10 * * * * cd /var/www/html/e-cidade/integracao_externa/auditoria; php -q auditoria_adiciona_fila.php >
log/auditoria_adiciona_fila.log 2>&1
```

Atenção: Lembre-se de ajustar o caminho para o diretório do e-cidade conforme o ambiente em que foi instalado.

6.6 Agendando a verificação do gerenciador de tarefas

```
[linux]$ sudo mkdir -p /opt/dbseller
[linux]$ sudo nano /opt/dbseller/scripts/ativadorGerenciadorTarefasEcidade.sh
```

Em seguida, adicione as linhas indicadas abaixo.

```
#!/bin/bash
sDiretorioEcidade="/var/www/html/e-cidade"
if [ ! -d $sDiretorioEcidade ]; then
    echo "Diretorio $sDiretorioEcidade nao existe"
    exit 1;
fi

sContador=$(ps aux | grep '<defunct>' | grep -v 'grep' | wc -l)
if [ "$sContador" -gt "5" ]; then
    sPid=$(ps aux | grep "con4_gerenciadortarefas002.php" | grep -v "grep" | awk '{print $2}');
    kill ${sPid}
fi

sTarefa=$(ps aux | grep "con4_gerenciadortarefas002.php" | grep -v "grep");
if [ "$sTarefa" == "" ]; then
    cd $sDiretorioEcidade;
    /usr/bin/php -q FrontIntegracaoExterna.php --executable con4_gerenciadortarefas002.php >
    tmp/log_gerenciadortarefas.log 2> tmp/erros_gerenciadortarefas.log &
fi
```

Atenção: Lembre-se de ajustar o caminho para o diretório do e-cidade conforme o ambiente em que foi instalado.

```
[linux]$ sudo chmod +x /opt/dbseller/scripts/ativadorGerenciadorTarefasEcidade.sh
```

Em seguida, edite o crontab do usuário root através do comando:

```
[linux]$ sudo crontab -e
```

E adicione o seguinte bloco ao final do arquivo:

```
# DBSeller - Agendamento do script de verificacao do gerenciador de tarefas
*/15 * * * * cd /opt/dbseller/scripts && su -s /bin/bash www-data -c "bash
ativadorGerenciadorTarefasEcidade.sh"
```

6.7 Acessando o e-cidade

Após finalizar os procedimentos de instalação e configuração, abra um navegador e acesse o e-cidade através do endereço IP ou nome do servidor. EX: `http://ip_do_servidor/e-cidade`, utilize como nome de usuário **dbseller**, e deixe o campo senha em branco.

Atenção: Certifique-se que o bloqueador de pop-up esteja permitindo o acesso ao endereço IP ou nome do servidor em que o e-cidade foi instalado.

7. Instalação do e-cidadeonline

7.1 Instalando o e-cidadeOnline

```
[linux]$ sudo cp -r e-cidade-x.x.xx-linux.completo/e-cidadeonline /var/www/html
```

7.2 Configurando o e-cidadeOnline

```
[linux]$ sudo cp /var/www/html/e-cidadeonline/libs/db_conn.php.dist  
/var/www/html/e-cidadeonline/libs/db_conn.php
```

Utilize o seguinte comando para editar o arquivo de configuração:

```
[linux]$ sudo nano /var/www/html/e-cidadeonline/libs/db_conn.php
```

Em seguida, atualize as configurações conforme o exemplo abaixo.

```
$DB_INSTITUICAO = 1;  
$DB_SERVIDOR = "localhost";  
$DB_BASEDADOS = "ecidade";  
$DB_USUARIO = "ecidade";  
$DB_SENHA = "ecidade";  
$DB_PORTA = "5432";
```

```
[linux]$ sudo nano /var/www/html/e-cidadeonline/libs/config.mail.php
```

Em seguida, atualize as configurações, com os dados do seu servidor de e-mail, conforme o exemplo abaixo.

```
$sClass = 3;  
$sSsl = ''; // qual o tipo de ssl q sera utilizados ( ex: tls), por padrao é vazio  
$bAuth = true; // autenticacao habilitada  
$sUser = 'exemplo@exemplo.com.br';  
$sPass = 'senhaexemplo';  
$sHost = 'smtp.exemplo.com.br';  
$sPort = '25';
```

```
[linux]$ sudo chown -R dbseller.www-data /var/www/html/e-cidadeonline
[linux]$ sudo chmod -R 775 /var/www/html/e-cidadeonline
[linux]$ sudo chmod -R 777 /var/www/html/e-cidadeonline/tmp
```

7.3 Acessando o e-cidadeonline

Após finalizar os procedimentos de instalação e configuração, abra um navegador e acesse o e-cidadeonline através do endereço IP ou nome do servidor. EX: `http://ip_do_servidor/e-cidadeonline`, utilize como nome de usuário **dbseller**, e deixe o campo senha em branco.

8. Configurando a carga de integrações

Dica: Preste atenção aos horários de agendamento do crontab de múltiplas integrações. Procure verificar o tempo de duração da execução de cada integração, ajustando o horário de execução conforme a necessidade.

Dica: Sempre que possível, execute manualmente a linha agendada para certificar o correto funcionamento do agendamento.

8.1 Configurando a integração de carga do cubo do BI

```
[linux]$ sudo cp -a /var/www/html/e-cidade/libs/db_cubo_bi_config.php.dist
/var/www/html/e-cidade/libs/db_cubo_bi_config.php
```

Utilize o seguinte comando para editar o arquivo de configuração:

```
[linux]$ sudo nano /var/www/html/e-cidade/libs/db_cubo_bi_config.php
```

Em seguida, atualize as configurações conforme o exemplo abaixo.

```
$configCuboBi['ftp']['server']      = "endereco.ip.do.servidor";
$configCuboBi['ftp']['usuario']     = "nome-usuario-ftp";
$configCuboBi['ftp']['senha']       = "senha-do-usuario-ftp";
$configCuboBi['ftp']['diretorio']   = "diretorio-de-destino";
$configCuboBi['ftp']['passive_mode'] = true;
```

8.2 Configurando a integração de carga do Webservice HORUS BASICO

8.2.1 Crie o arquivo de configuração de através do comando:

Dica: Mais informações sobre o Webservice HORUS BASICO podem ser obtidas em: <http://portalsaude.saude.gov.br/index.ministerio/principal/leia-mais-o-ministerio/218-sctie-raiz/daf-raiz/ceaf-sctie/qualifarsus-raiz/eixo-informacao/11-eixo-informacao/8508-envio-de-dados-webservice>

```
[linux]$ sudo cp -a /var/www/html/e-cidade/libs/db_config_horus.php.dist  
/var/www/html/e-cidade/libs/db_config_horus.php
```

Utilize o seguinte comando para editar o arquivo de configuração:

```
[linux]$ sudo nano /var/www/html/e-cidade/libs/db_config_horus.php
```

Em seguida, atualize as configurações conforme o exemplo abaixo.

Atenção: Consulte a documentação das URLs no portalsaude.saude.gov.br antes de utilizar a integração.

```
// Homologação  
$sUrlWebService = "http://189.28.128.37/horus-ws-basico/RecebeDadosWS?wsdl";  
$sLogin        = "nome-usuario-horus";  
$sSenha        = "senha-do-usuario-horus";
```

8.3 Configurando as integrações de limpeza de cadastros duplos CGM e CGS

```
[linux]$ crontab -e
```

Em seguida, adicione o seguinte bloco ao final do arquivo:

```
# DBSeller - Agendamento do script para limpeza de cadastro duplos CGM  
0 22 * * * cd /var/www/html/e-cidade/ ; /usr/bin/php FrontIntegracaoExterna.php --executable duplos.php >  
duplos.log 2>&1
```

Atenção: Lembre-se de ajustar o caminho para o diretório do e-cidade conforme o ambiente em que foi instalado.

```
[linux]$ crontab -e
```

Em seguida, adicione o seguinte bloco ao final do arquivo:

```
# DBSeller - Agendamento do script para limpeza de cadastro duplos CGS  
0 22 * * * cd /var/www/html/e-cidade/ ; /usr/bin/php FrontIntegracaoExterna.php --executable duploscgs.php  
> duploscgs.log 2>&1
```

Atenção: Lembre-se de ajustar o caminho para o diretório do e-cidade conforme o ambiente em que foi instalado.

8.4 Configurando a integração GISS

O script de execução da integração GISS possui dois parametros de configuração:

- 1) Se o programa for agendado sem nenhum parâmetro, ou com o parametro 1, a integração processará somente os registros alterados.

```
.../iss4_gissonline002.php 1 > /dev/null 2>&1
```

- 2) Se o programa for agendado com o parâmetro 2, a integração processará todos os registros.

```
.../iss4_gissonline002.php 2 > /dev/null 2>&1
```

```
[linux]$ crontab -e
```

Em seguida, adicione o seguinte bloco ao final do arquivo:

```
# DBSeller - Agendamento do script para execução da integração GISS
0 22 * * * cd /var/www/html/e-cidade/ ; /usr/bin/php FrontIntegracaoExterna.php --dir
integracao_externa/gissonline --executable integracao_externa/gissonline/iss4_gissonline002.php >
/dev/null 2>&1
```

Atenção: Lembre-se de ajustar o caminho para o diretório do e-cidade conforme o ambiente em que foi instalado.

8.5 Configurando a integração SIGCORP

```
[linux]$ crontab -e
```

Em seguida, adicione o seguinte bloco ao final do arquivo:

```
# DBSeller - Agendamento do script para execução da integração SIGCORP
0 22 * * * cd /var/www/html/e-cidade/ ; /usr/bin/php FrontIntegracaoExterna.php --dir
integracao_externa/sigcorp --executable integracao_externa/sigcorp/iss4_sigcorp001.php > /dev/null 2>&1
```

Atenção: Lembre-se de ajustar o caminho para o diretório do e-cidade conforme o ambiente em que foi instalado.

8.6 Configurando a integração WEBISS

```
[linux]$ crontab -e
```

Em seguida, adicione o seguinte bloco ao final do arquivo:

```
# DBSeller - Agendamento do script para execução da integração WEBISS
0 22 * * * cd /var/www/html/e-cidade/ ; /usr/bin/php FrontIntegracaoExterna.php --executable
integracao_externa/webiss/iss4_webiss001.php > /dev/null 2>&1
```

Atenção: Lembre-se de ajustar o caminho para o diretório do e-cidade conforme o ambiente em que foi instalado.

8.7 Configurando a integração de geração das tabelas débitos

```
[linux]$ crontab -e
```

Em seguida, adicione o seguinte bloco ao final do arquivo:

```
# DBSeller - Agendamento do script para geração das tabelas débitos
0 20 * * * cd /var/www/html/e-cidade/integracao_externa/debitos ; ./debitos.sh &> /dev/null
```

Atenção: Lembre-se de ajustar o caminho para o diretório do e-cidade conforme o ambiente em que foi instalado.

8.7.2 Configurando a integração débitos

O script de execução da integração débitos necessita de um arquivo de configuração para ser executado. Para criar o script de configuração, execute os comandos abaixo:

```
[linux]$ cd /var/www/html/e-cidade/integracao_externa/debitos/libs
[linux]$ cp -a debitos.conf.dist debitos.conf
```

Em seguida edite o arquivo através do comando:

```
[linux]$ nano debitos.conf
```

Informe o valor das variáveis conforme as descrições abaixo:

```
# Permite definir a periodicidade com que a manutencao da tabela debitos
# sera executada. O valores permitidos sao: -1=desativa, 0=todos ou dia da semana (1=segunda, 2=terca ...
# 7=domingo).
#
# Exemplo:
# DEBITOS_PERIODO="7"
#
DEBITOS_PERIODO=""

# Permite definir o numero de dias no passado para manter os registros na debitos.
#
# Exemplo:
# DEBITOS_DIAS_MANTER="14"
```

```
#
DEBITOS_DIAS_MANTER=""

# Permite definir o código da(s) instituição(ões) que deve(em) ser gerada débitos. Se houver
# mais de uma instituição, os seus códigos devem ser separados por espaço.
#
# Exemplo:
# DEBITOS_INSTITUICOES="1"
# DEBITOS_INSTITUICOES="1 3"
#
DEBITOS_INSTITUICOES=""
```

Atenção: Lembre-se de ajustar o caminho para o diretório do e-cidade conforme o ambiente em que foi instalado.

Glossário

| Termo | Significado |
|-------------------|---|
| encoding | http://en.wikipedia.org/wiki/Character_encoding |
| cluster | http://www.postgresql.org/docs/8.2/interactive/creating-cluster.html |
| apt-get | http://www.debian.org/doc/manuals/apt-howto |
| apt-get | http://pt.wikipedia.org/wiki/Advanced_Packaging_Tool |
| nano | http://www.nano-editor.org |
| Firefox | http://www.mozilla.org/firefox |
| DBSeller Sistemas | http://www.dbseller.com.br |
| proxy | http://pt.wikipedia.org/wiki/Proxy |
| squid | http://pt.wikipedia.org/wiki/Squid |

pdf(); doc- > doc();doc->html(); ?>