MANUAL DE
DESENVOLVIMENTO ver 1.0

KernelVB
(athcsm4 / virtualboss)

COPYRIGHT © GRUPO PROEVENTO - ATHENAS SOFTWARE & SYSTEMS

SUMÁRIO

COPYRIGHT © GRUPO PROEVENTO - ATHENAS SOFTWARE & SYSTEMS

1 - APRESENTAÇÃO

Este é o Manual BETA de **DESENVOLVIMENTO** do FPSPROJET e busca ser o conjunto de regras e padrões de implementação para os desenvolvedores de sistemas utilizando o PS Kernel, de propriedade da Athenas Software and Systems.

2 - PADRÕES PL/SQL

2.1 - Prefixos

2.1.1 – Objetos/Schema

Domains	dom_
Stored Procedure (Func ou Proc)	sp_
Stored Procedure (Internas)	spi_
Triggers Function	tgf_
Triggers	tg_ (mesmo nome da trigger function)
Sequences	(nome da tabela)_(nome do campo)_seq
Primary Key	pk_
Foreign Key	fk_
Index	idx_(nome da tabela)_(nome do campo)
Flags	flag_

2.1.2 – Tabelas/Schema

Tabelas de Sistema	sys_
Tabelas	*_ (prefixo definido pelo subsistema que ela participa, justamente para agregar as tabelas de mesmo subsistema no BD)
View	view_

2.1.3 – Variáveis de PL/SQL

Variáveis de Código	var_
Parâmetros de Function/Procedure	pr(nome do campo com as iniciais em letras maiúsculas) Ex: prNomeTabela

3 – INSTRUÇÕES DE CÓDIGO PL/pgSQL

3.1 - Comandos SQL.

Tipos de variáveis devem estar em letra maiúscula.

Ex:

```
SELECT cod_tipo FROM foo;
EXECUTE PROCEDURE sp_teste('BOO');
EXECUTE PROCEDURE spi_dummy('FOO');
var_teste INTEGER;
```

3.2 – Estrutura do DB/Scheme

Nomes de tabelas, campos e funções (inclusive as criadas por nós) devem estar em letra minúscula.

Ex:

```
CREATE OR REPLACE FUNCTION sp_dummy(prCod INTEGER) RETURNS
VARCHAR AS
$BODY$
DECLARE

prCod ALIAS FOR $1;
var_teste INTEGER;
var_timestamp TIMESTAMP WITHOUT TIME ZONE;

BEGIN
      SELECT numero FROM foo INTO :var_teste;
      var_timestamp := current_timestamp;
      --current_timestamp is a func.
      var_teste := var_teste + 30;
      RETURN var_teste || ' ' || var_timestamp;
END;

$BODY$
LANGUAGE 'pgplsql' VOLATILE;
```

3.2 – Extras

- **Strings** são definidas por apóstrofes em vez de aspas.
- O limitador de código que o postgre pede será o **$BODY$** .

4 - POLÍTICA DE SCRIPTS GERADOS

Quando for gerado um script novo, ele deve ser armazenado na pasta '/_estudo/PGSQL/nome_criador_script/' com o nome 'SCRIPT_NOME_DATA_E_HORA.sql' se for pra ser armazenado. Caso ele deva ser executado colocar na raiz da pasta PGSQL. Após rodar o script deve ser colocado na pasta _runned. Os scripts de referência estarão na pasta _ref.

5 - ABREVEATURAS - CAMPOS

CIDADE	
CLIENTE	
COBRANÇA	
CÓDIGO	
CONSUMIDOR	
DATA	
DATA HORA	
DESCONTO	

Observação: Todos os itens criados que forem temporários no banco, tais como: tabelas, views, campos, índices, stored procedures, functions e etc. devem estar com o prefixo **tmp_** na frente do nome já com os prefixos estabelecidos acima. Ex: **tmp_sp_teste(), tmp_view_cad_cli.**

TEXTO SOBRE MODULO ... PARA USAR DEPOIS

São pastas que recebem o nome da tabela do banco de dados e são localizadas dentro do diretório do site onde contém muitos outros arquivos e pastas de funcionalidades diferentes. A sua instalação inicia-se fora do sistema. Deve-se criar um pasta com a seguinte nomenclatura “modulo*_nomeTabela*”, onde o texto sublinhado é o local que receberá o nome da tabela na qual o módulo refere-se. Após criado a pasta e renomeado conforme a explicação anterior, deve-se ir até a pasta *_tools/Atualizador.* Dentro desta pasta, têm um arquivo nomeado *AthCopyFiles*, dê dois clicks sobre o mesmo. O sistema iniciará totalmente totalmente sem a sua funcionalidade. Por isso, vá até o menu *Arquivos/Carregar Configuração.* Será aberto uma janela para procurar o respectivo arquivo de configuração. Este arquivo estará sempre em *_tools/Atualizador,* junto com o aplicativo utilizado. Ao ser carregado o arquivo de configuração chamado *v1.0.1.cmf,* aparecerá no aplicativo os arquivos de origem e todos as pastas de destino. A função deste aplicativo será de copiar todos os arquivos que encontram-se dentro da pasta *_font* para dentro de todos os módulos, mesmo aqueles já criados(como os arquivos são iguais em todos os módulos do sistema, não há problema algum ao sobre escrever estes). Verifique se realmente foi realizado com sucesso esta operação indo até a pasta do módulo criado. Após a verificação crie uma nova pasta dentro do módulo que está sendo configurado e renomeie para *lang.* Copie o arquivo que está em *_fontes/lang* e cole dentro desta nova pasta criada anteriormente também chamada *lang.* Com esta instalação concluída, inicia-se a configuração do módulo no sistema. Para fazer esta configuração, primeiramente é necessário que acesse a aba de “Sistema”, dentro do Container “Módulo” e clique no ícone “Módulo”. Ao clicar neste ícone, será visualizada uma Dialog que listará algumas informações referentes aos registros dos módulos, podendo ser Deletado ou Visualizado. Por padrão, a Dialog que lista os módulos do sistema, terá um elemento do tipo combo que, ao clicar neste, listará todos os Direitos para este módulo.

Abaixo, uma imagem que explicará melhor a posição desta combo, veja.