



Secretaria da Fazenda do Estado do Tocantins
Centro Interamericano de Administrações Tributárias

Produto: Serviços Transversais – Desenho Informático

Fevereiro/2016

Histórico de Revisões

Data	Descrição	Autor
09/10/2015	Versão 1. Criação do documento. Autorização e Registro de Navegação.	Juan León
06/11/2015	Inclusão: Big Data em Registro de Navegação	Juan León
20/11/2015	Inclusão: Desenvolver solução para Registro de Operações no Banco de Dados	Juan León
23/11/2015	Inclusão: Monitoramento do Serviços.	Juan León
27/01/2016	Ajustes nos casos de uso	Juan León
29/02/2016	Ajustes nos casos de uso, várias correções	Juan León
16/03/2016	Revisão e Ajustes	Juan León
04/04/2016	Revisão e Ajustes	Decio Carretta
05/04/2016	Revisão e Ajustes	Juan León



Tabela de Conteúdo

1. Introdução.....	4
2. Glossário	4
3. Gestão de Segurança	4
3.1. Identificação e Autenticação	4
3.1.1. <i>Desenho de Solução para Identificação e Autenticação</i>	4
3.2. Autorização.....	5
3.2.1. <i>Desenho para Autorização</i>	5
3.2.2. <i>Diagrama Entidade Relacionamento para Autenticação/Autorização</i>	7
3.3. Registro de Navegação	8
3.3.1. <i>Esquema de Captura de Dados</i>	9
3.3.2. <i>Configuração de um Java Web Servlet Filter em Spring</i>	9
3.3.3. <i>Recomendações para implementação de Data Analytics.</i>	10
3.3.4. <i>Implementação com ferramentas de Big Data</i>	11
3.4. Solução para Registro de Operações no BD	11
3.4.1. <i>Registro Log de Oracle</i>	11
3.4.2. <i>Desenho de Solução para Registro de Operações por tabela</i>	12
4. Monitoramento do Serviços	16
4.1. Introdução	16
4.2. Solução para Controle de Eventos Anormais	16
4.2.1. <i>Desenho da Solução</i>	17
4.2.2. <i>Gestão de Exceções</i>	18
4.2.3. <i>Registro de Exceções</i>	18
4.2.4. <i>Casos de uso para Controle de Eventos Anormais</i>	18
4.3. Implementar o uso do Zabbix para dar Suporte ao Controle de Qualidade.....	18
5. Gestão de Comunicação Eletrônica	18
5.1. Desenvolver solução para comunicação eletrônica via e-mail	18
5.1.1. <i>Solução Proposta para enviar e-mails desde Spring</i>	19
5.2. Desenvolver solução para comunicação eletrônica via SMS	21
5.2.1. <i>Solução Proposta</i>	22

1. Introdução

Este documento descreve a desenho informático dos Serviços Transversais: 1) Gestão de Segurança, 2) Monitoramento de Serviços e 3) Gestão de Comunicação Eletrônica. Esses serviços transversais são para o Novo Sistema Tributário da SEFAZ-TO baseado no documento de definição dos Serviços Transversais “Produto 1.1 - Definição dos serviços tecnológicos transversais - 201450312 v2.0.docx” (para detalhes da definição dos serviços transversais por favor consultar este documento).

2. Glossário

Certificado Digital: é um arquivo de computador que contém um conjunto de informações referentes à entidade para a qual o certificado foi emitido (seja uma empresa, pessoa física ou computador) mais a chave pública referente à chave privada que se acredita ser de posse unicamente da entidade especificada no certificado. É uma assinatura com validade jurídica que garante proteção às transações eletrônicas e outros serviços via internet, permitindo que pessoas e empresas se identifiquem e assinem digitalmente de qualquer lugar do mundo com mais segurança e agilidade.

ICP-Brasil: É uma estrutura composta de um ou mais certificadores denominados de Autoridades Certificadoras - AC que, através de um conjunto de técnicas e procedimentos de suporte a um sistema criptográfico baseando-se em certificados digitais, consegue assegurar a identidade de um usuário de mídia eletrônica ou assegurar a autenticidade de um documento suportado ou conservado em mídia eletrônica.

3. Gestão de Segurança

É vital importância ter um sistema robusto e seguro na Web, é por isso que se propõe a seguinte solução.

3.1. Identificação e Autenticação

Todos os usuários devem ser identificados no Novo Sistema Tributário para ter acesso aos serviços fornecidos pelo Sistema, exceto para aquelas opções (serviços) do sistema que sejam do escopo público como, por exemplo, o Portal da SEFAZ.

3.1.1. Desenho de Solução para Identificação e Autenticação

Implementar o uso do Active Directory (AD- Serviço de diretórios da Microsoft, gerencia configuração da rede, incluindo usuários e dispositivos) como base para Identificação e Autenticação dos Usuários. A SEFAZ irá instalar e configurar 2 servidores (principal e secundário)



para Active Directory do domínio **sefaz.to.gov.br**. Teremos o acesso ao Active Directory desde Java usando o protocolo LDAP (LightWeight Directory Access Protocol) assim **ldap://dc1.sefaz.to.gov.br:389**.

Todos usuários do Sistema serão cadastrados no Windows Server Active Directory da unidade organizacional correspondente. Os usuários do Sistema poderão usar o CPF ou certificado digital para autenticar-se. Não existirá uma tabela de usuários para autenticação no banco de dados, só para informação adicionais sobre estes.

Para a conexão para o banco de dados deve-se criar um usuário do banco de dados para cada usuário criado no sistema.

3.2. Autorização

Uma vez o usuário é identificado e autenticado no Active Directory ou usando o certificado digital (seja usuário interno ou externo), a autorização para acesso às funcionalidades dos sistemas é concedida no mesmo sistema. Esse controle de acesso deve ser feito usando perfil do usuário, onde um usuário pode ter mais de um perfil.

Um perfil pode ter um ou vários papéis (roles). E um papel pode ter várias funcionalidades do sistema. Toda a informação para autorização deve ser salva no banco de dados Oracle.

3.2.1. Desenho para Autorização

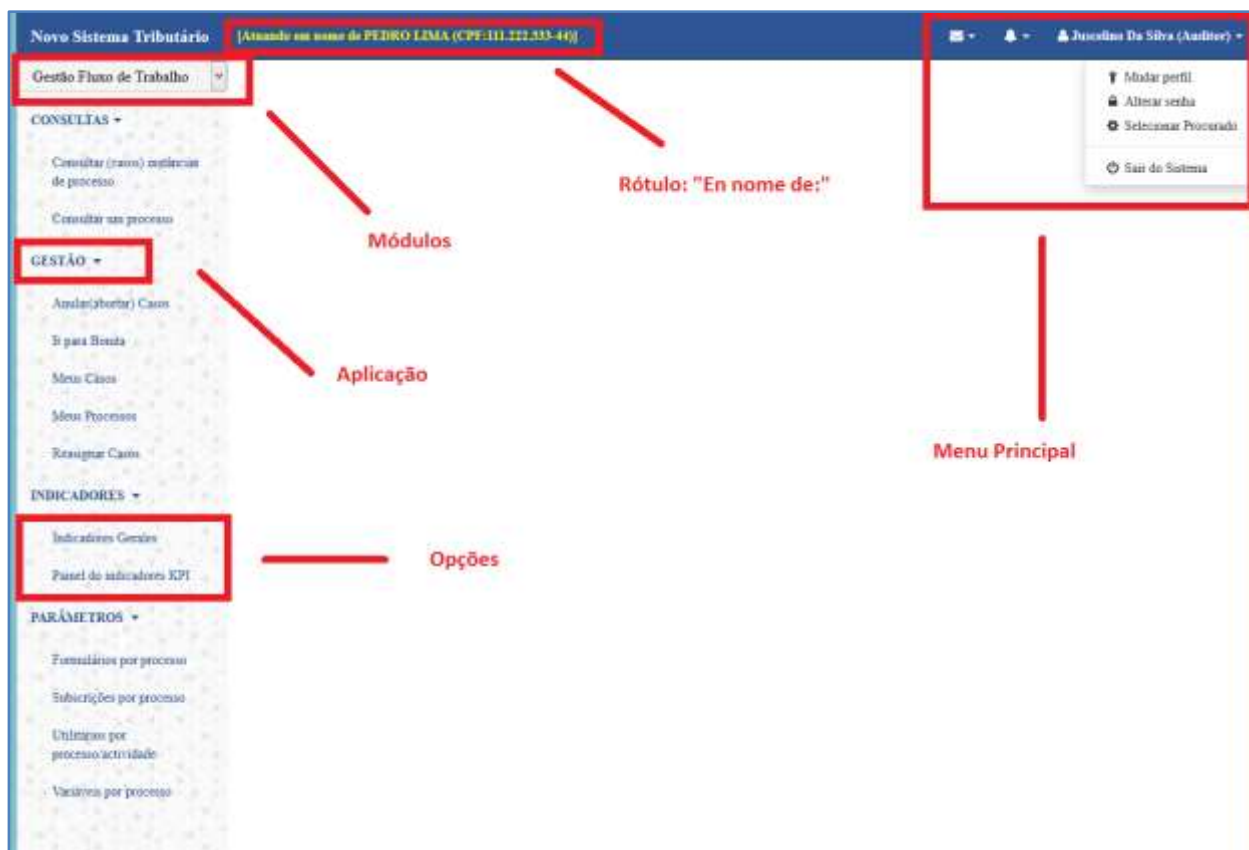
Então, só para esclarecer, um usuário pode ter assignados um ou vários perfis. Um perfil pode ter um ou vários papéis. Um papel (role) pode ter acesso a diferentes opções (permissões) do sistema, assim:

Exemplos¹: a) O usuário Juscelino Da Silva tem o perfil Auditor. O perfil Auditor tem os papéis: AuditorGestor (esse papel tem uma lista de funcionalidades para gestão de auditorias), BPMExecutor (papel para interagir com o BPM) e GEDConsultor (papel para consultar documentos no módulo Gestão Eletrônica do Documentos). b) A usuária Rejane Souza tem o perfil Auxiliar Recepção. Este perfil tem os papéis: GEDConsultor (acesso para ler documentos) e GEDGestor (acesso para usar, criar, alterar documentos).

Na seguinte tela, mostram-se as opções (permissões) do sistema para Juscelino Da Silva com perfil Auditor. Ele está usando o **módulo** WRK (Gestão de Fluxo de Trabalho), onde tem várias **aplicações** (Gestão, Consultas, Parâmetros e Indicadores), nestas aplicações existem várias **opções** para cada uma. Por exemplo: Em **Consultas** tem as **opções** (telas): “Consultar um processo” e “Consultar instâncias de processo”.

¹ Os nomes dos exemplos são fictícios

NOTA: É importante mostrar somente aquelas opções no menu principal que estão disponíveis para o usuário final. Aquelas opções onde o usuário não tem acesso não devem ser mostradas.



Nesta imagem mostra-se o menu completo para o módulo **WRK** com seções para autorização

NOTA: todos os dados de módulos, aplicações, opções, etc. deverão estar gravados no banco de dados Oracle, o arquivo xml é só um exemplo de dados.



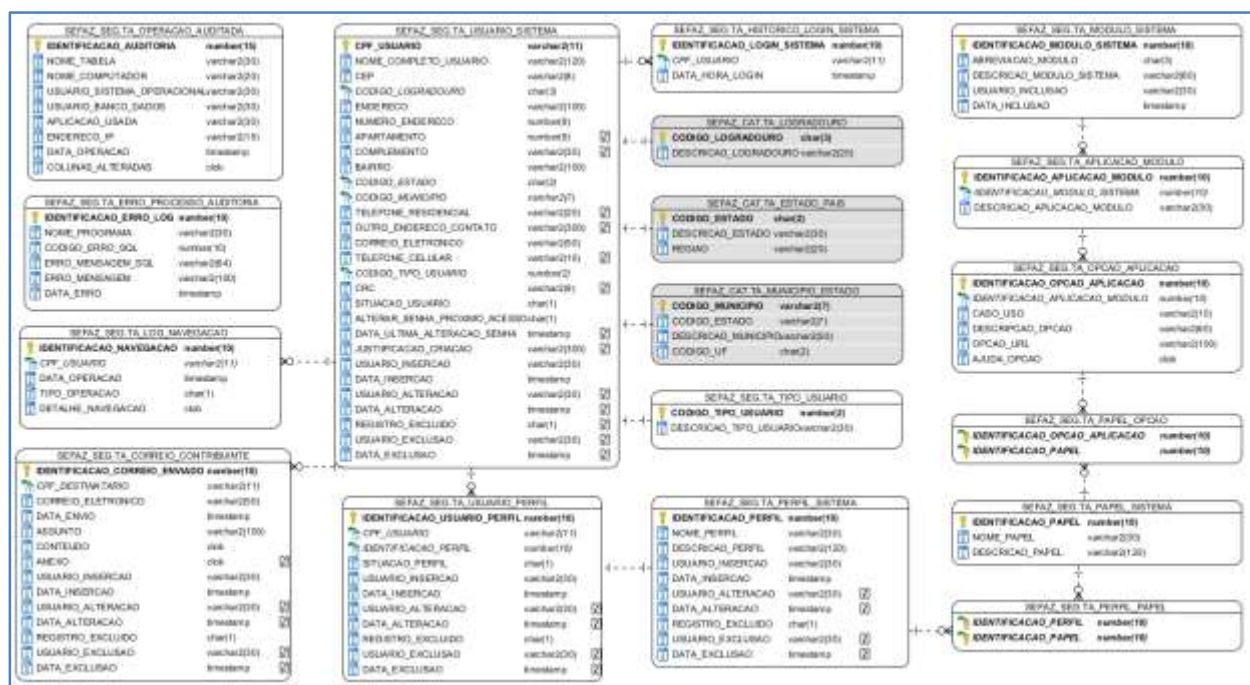
Aqui há uma lista dos módulos do sistema.

Esquema	Descrição
ARR	Arrecadação.
CCI	Cadastro de Contribuinte
CCC	Conta Corrente
COB	Cobrança
COM	Consulta
ECF	Equipamento Emissor de Cupom Fiscal
FIE	Fiscalização Estabelecimento
FTR	Fiscalização de Trânsito de Mercadorias
GED	Gerenciamento Eletrônico de Documentos
ISE	Isenções
NFE	NFE
OTR	Outras Receitas
REE	Regimes Especiais
RES	Restituições
SEG	Segurança do Sistema
WRK	Gestão de Fluxo de Trabalho

O controle de acesso é o processo de determinação dos direitos de acesso aos recursos da aplicação, em outras palavras, “quem acessa o que”.

3.2.2. Diagrama Entidade Relacionamento para Autenticação/Autorização

O seguinte diagrama mostra as entidades usadas para modelar esta solução para Autenticação/Autorização:



3.3. Registro de Navegação

Para todas as funcionalidades acessadas, o usuário logado e o momento em que ocorreram deverão ser registradas no banco de dados, independentemente se houve alteração de dados ou não. Por exemplo: Ativação de consultas, relatórios, opções do sistema, login do sistema, logon de sistema, etc.

Uma aplicação sem um log é uma aplicação muda, os administradores do sistema precisam saber que está acontecendo no sistema.

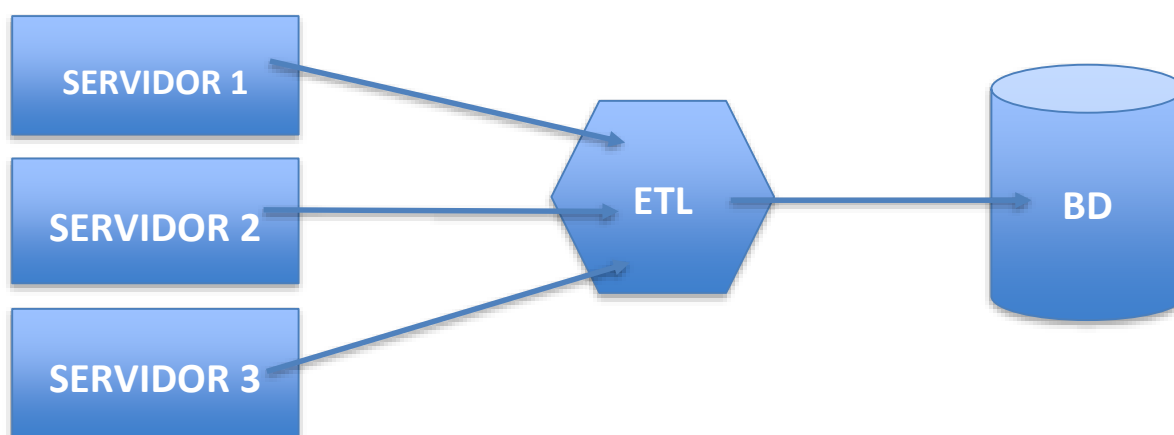
Para isso, é importante saber coisas por exemplo:

- Que opção do sistema foi acessada pelo que usuário num momento específico.
- Quais são as páginas mais usadas por papel (role).
- Quais são as páginas mais utilizadas em todo o sistema.
- As páginas que levam mais tempo para responder.
- A hora no dia de maior demanda de requisitos.
- A hora no dia de menor demanda de requisitos.

Hoje, existem vários formatos para uso arquivos simples (Plain Text), como: txt, csv, xml, json. O formato para Web Services tipo Restful é JSON no caso de uso de aplicações moveis ou streaming. Recomenda-se para gerar o log do sistema neste formato. Recomenda-se usar a especificação JSR 353: Java API for JSON Processing.

3.3.1. Esquema de Captura de Dados

Como o aplicativo será executado em vários servidores, deve-se criar um diretório **/log** na raiz do sistema, será gerado um arquivo no formato *LogOperações-<ServerName>-YYYYMMDD.log*. Onde são coletadas as operações por dia por servidor de aplicações.



No ETL (Extract, Transform, Load) pode consolidar a informação e fazer tarefas de transformação como por exemplo: adicionar o país, estado, cidade, latitude, longitude usando de acordo o endereço ip. Uma implementação em Java usando para “Geographic Location” está aqui: <http://memorynotfound.com/convert-ip-address-geo-location-java/>

Para a captura dos erros do sistema a opção dos arquivos planos é melhor.

3.3.2. Configuração de um Java Web Servlet Filter em Spring

Para criar um java filter é muito fácil só precisa fazer a seguinte configuração:

No arquivo web.xml

```
<filter>
  <filter-name>registroNavegacaoFilter</filter-name>
  <filter-class>org.springframework.web.filter.DelegatingFilterProxy</filter-class>
  <init-param>
    <param-name>targetBeanName</param-name>
    <param-value>registroNavegacaoFilterBean</param-value>
  </init-param>
</filter>

<filter-mapping>
  <filter-name>registroNavegacaoFilter</filter-name>
  <url-pattern>*.jsp</url-pattern>
</filter-mapping>
```

No arquivo **applicationContext.xml**

```
<bean name="registroNavegacaoFilterBean"
class="br.gov.to.sefaz.webapp.seguranca.filter.registroNavegacaoFilter"/>
```

Na classe registroNavegacaoFilter, pode-se usar a informação do “Request” para obter o usuário e a página que acessam.

3.3.3. Recomendações para implementação de Data Analytics.

As informações de log podem crescer muito (histórico) e dependendo do tamanho podem ser usadas de várias estratégias. Além disso, o log é trabalhado para fazer Data Analytics.

Na tabela a seguir, mostra-se várias soluções de acordo o tamanho de seus logs e também considerando o tempo de resposta:

Tamanho Total	Solução
<100GB	O ETL pode ser feito com o software Informática (SEFAZ-TO tem disponível esse software). O mesmo banco de dados transacional Oracle 12c usando In-Memory Acceleration. Pode integrar com o novo sistema tributário.
<500GB	É melhor usar ferramentas criadas para fazer Data Analytics de Log Management em tempo real como por exemplo: Splunk (splunk.com) ou LogRhythm (logrhythm.com)

Tamanho Total	Solução
>500GB	Um banco de dados assim precisa usar ferramentas de Big Data. Primeiro o banco de dados deve estar em formato MapReduce (Apache Hadoop) ou um banco de dados como Apache HBase para grandes tabelas. Aqui o formato JSON é muito importante. Para as consultas pode ser feita uma integração com Apache Pig, Hive com Java no novo sistema tributário.

3.3.4. Implementação com ferramentas de Big Data

Log Mining é um problema típico de Big Data porque cumpre (em inglês):

- Too many bytes (Volume). Muitos bytes.
- Too high rate (Velocity).
- Too many sources (Variety)

3.4. Solução para Registro de Operações no BD

3.4.1. Registro Log de Oracle

Oracle Database (RDBMS) salva todas as operações feitas no banco de dados em seus arquivos redolog. Os arquivos redologs são o registro histórico das alterações feitas no banco de dados como resultado das transações e ações internas do Oracle.

Os redologs protegem contra a perda de integridade em caso de falhas de energia ou erros em discos rígidos. Os redologs trabalham de forma cíclica em grupos de redologs online.

Oracle permite salvar um grupo cheio de redologs em um ou mais destinos offline, conhecido como “archived redo logs”. O processo de virar redolog em archived redologs é chamado archiving. Esse processo é possível somente quando o banco de dados está configurado em mode ARCHIVELOG.

Com os archived redologs é possível ler as transações e operações feitas usando a ferramenta Oracle LogMiner. Para isso, é necessário ter um banco de dados com a mesma estrutura do banco original, onde estão os archived log para importar neste banco de dados.

O processo de leitura dos arquivos (LogMiner) pode ser complicado quando o número de tabelas em um sistema o número de transações é muito elevado, vai gerar uma grande quantidade de

archived logs por dia. Se por exemplo: é necessário conhecer uma transação feita no mês passado Pode ser uma tarefa demorada obter a informação.

3.4.2. Desenho de Solução para Registro de Operações por tabela

Quando precisamos só auditar algumas tabelas específicas pode usar um registro de log usando trigger do banco de dados e salvando a informação em pista de auditoria numa tabela.

Portanto, em uma tabela de operações para cada DML (Data Manipulation Language) fica assim:

Operação	Valor Anterior	Valor Novo
Insert	N/A	Sim (:NEW)
Update	Sim (:OLD)	Sim (:NEW)
Delete	Sim (:OLD)	N/A

A operação DELETE não é permitida no sistema. Toda exclusão de registro no sistema deve ser feita usando exclusão lógica. Para cada tabela para auditar é necessário ter as seguintes colunas adicionais:

Operação	Valor Anterior	Valor Novo	Descrição
Insert	usuario_insercao	varchar2(30) not null	Login do usuário responsável pela inserção do registro
Insert	data_insercao	timestamp not null	Data e hora de inserção do registro
Update	usuario_alteracao	varchar2(30)	Login do usuário responsável pela alteração do registro
Update	data_alteracao	timestamp	Data e hora de alteração do registro
Delete	registro_excluido	char(1) default 'N'	Identificador de registro excluído Sim ou Não
Delete	usuario_exclusao	varchar2(30)	Login do usuário responsável pela exclusão do registro

Operação	Valor Anterior	Valor Novo	Descrição
Delete	data_exclusao	timestamp	Data e hora de exclusão do registro

De acordo com esse modelo o trigger automaticamente deve preencher para: inclusão (usuario_insercao, data_insercao), alteração (usuario_alteracao, data_alteracao) e exclusão (registro_excluido, usuario_exclusao, data_exclusao).

Para o cadastro das pistas de auditoria, pode usar a seguinte tabela:

```
-- tabela dos resultados da auditoria
drop table SEFAZ_SEG.TA_OPERACOES_AUDITADAS cascade constraints;
create table SEFAZ_SEG.TA_OPERACOES_AUDITADAS (
  id_auditoria      number(15) not null,    /* Identificador sequencial de auditoria */
  nome_tabela       varchar2(30) not null,   /* Nome de tabela que foi alterada */
  nome_computador   varchar2(20) not null,   /* Nome do computador onde foi alterada a informacao */
  usuario_sistema_operacional varchar2(20) not null, /* Login do usuário do sistema operativo responsável pela alteração */
  usuario_banco_dados varchar2(30) not null, /* Login do usuário do banco de dados responsável pela alteração */
  aplicacao_usada   varchar2(50) not null,   /* Programa usado para fazer a alteração */
  endereco_ip       varchar2(15) not null,   /* Endereco ip o computador usado */
  data_operacao     timestamp not null,      /* Data e hora da alteração do registro */
  colunas_alteradas xmltype not null        /* Os valores que foram alterados */
);
COMMENT ON TABLE SEFAZ_SEG.TA_OPERACOES_AUDITADAS IS 'Armazena a informação de auditoria de alteracoes';
COMMENT ON COLUMN SEFAZ_SEG.TA_OPERACOES_AUDITADAS.id_auditoria IS 'Identificador sequencial de auditoria';
COMMENT ON COLUMN SEFAZ_SEG.TA_OPERACOES_AUDITADAS.nome_tabela IS 'Nome de tabela que foi alterada';
COMMENT ON COLUMN SEFAZ_SEG.TA_OPERACOES_AUDITADAS.nome_computador IS 'Nome do computador onde foi alterada a informacao';
COMMENT ON COLUMN SEFAZ_SEG.TA_OPERACOES_AUDITADAS.usuario_sistema_operacional IS 'Login do usuário do sistema operativo responsável pela alteração';
COMMENT ON COLUMN SEFAZ_SEG.TA_OPERACOES_AUDITADAS.usuario_banco_dados IS 'Login do usuário do banco de dados responsável pela alteração';
COMMENT ON COLUMN SEFAZ_SEG.TA_OPERACOES_AUDITADAS.aplicacao_usada IS 'Programa usado para fazer a alteração';
COMMENT ON COLUMN SEFAZ_SEG.TA_OPERACOES_AUDITADAS.endereco_ip IS 'Endereco ip o computador usado';
COMMENT ON COLUMN SEFAZ_SEG.TA_OPERACOES_AUDITADAS.data_operacao IS 'Data e hora da alteração do registro';
COMMENT ON COLUMN SEFAZ_SEG.TA_OPERACOES_AUDITADAS.colunas_alteradas IS 'Os valores que foram alterados';

CREATE SEQUENCE SEFAZ_SEG.SEQ_OPERACOES_AUDITADAS INCREMENT BY 1 MINVALUE 1 START WITH 1 NOMAXVALUE
NOCYCLE NOCACHE NOORDER;
```

A informação de auditoria mostra-se na seguinte imagem, o formato para salvar a diferença de valores nas colunas é XMLTYPE.

ID_AUDITORIA	NOME_TABELA	NOME_COMPUTADOR	USUARIO_SIST_OPERATIVO	USUARIO_BANCO_DADOS	PROGRAMA	ENDERECO_IP	DATA_OPERACAO	COLUNAS_ALTERADAS
1	12 EXEMPLO	unknown	Ing. Juan León Solís	SEGURIDAD	SQL Developer 127.0.0.1	20/11/2015 02:40:24,232000000 PM		
2	13 EXEMPLO	unknown	Ing. Juan León Solís	SEGURIDAD	SQL Developer 127.0.0.1	20/11/2015 02:40:34,168000000 PM (DDLTYPE)		
3	14 EXEMPLO	unknown	Ing. Juan León Solís	SEGURIDAD	SQL Developer 127.0.0.1	20/11/2015 02:41:01,227000000 PM (DDLTYPE)		

Ver Valor

```

<?xml version="1.0" encoding="WINDOWS-1252"?>
<columns rowid="AAAAA8AAGAAAEJAND">
  <col name="VALOR_TEXTO">
    <old>HOLA PRUEBA4</old>
    <new>TESTING 4 ...</new>
  </col>
  <col name="VALOR_NUMERICO">
    <old>121.25</old>
    <new>70,15</new>
  </col>
</columns>

```

Carregar Descartar Definir NULL Editar

Ayuda Cancelar

O programa PL/SQL para gerar o trigger automaticamente está disponível para usar em SEFAZ.

Devem existir os seguintes objetos do Banco de Dados

```
-- tabla de errores al generar la auditoria

drop table SEFAZ_SEG.TA_ERRO_PROCESSO_AUDITORIA cascade constraints;

create table SEFAZ_SEG.TA_ERRO_PROCESSO_AUDITORIA (
  id_erro_log      number not null,      /* Identificador sequencial de log */
  nome_programa    varchar2(30) not null, /* Nome do programa que lanca erro */
  codigo_erro_sql  number not null,      /*Codigo de erro SQLCODE */
  erro_mensagem_sql varchar2(64) not null, /* Mensagem de erro SQLERRM */
  erro_mensagem    varchar2(100) not null, /* Mensagem de erro SQLERRM */
  data_erro        timestamp not null    /* Data e hora do erro */
);

COMMENT ON TABLE SEFAZ_SEG.TA_ERRO_PROCESSO_AUDITORIA IS 'Armazena a informação dos erros do processo de auditoria';
COMMENT ON COLUMN SEFAZ_SEG.TA_ERRO_PROCESSO_AUDITORIA.id_erro_log IS 'Identificador sequencial de log';
COMMENT ON COLUMN SEFAZ_SEG.TA_ERRO_PROCESSO_AUDITORIA.nome_programa IS 'Nome do programa que lanca erro';
COMMENT ON COLUMN SEFAZ_SEG.TA_ERRO_PROCESSO_AUDITORIA.codigo_erro_sql IS 'Código de erro SQLCODE';
COMMENT ON COLUMN SEFAZ_SEG.TA_ERRO_PROCESSO_AUDITORIA.erro_mensagem_sql IS 'Mensagem de erro SQLERRM';
COMMENT ON COLUMN SEFAZ_SEG.TA_ERRO_PROCESSO_AUDITORIA.data_erro IS 'Data e hora do erro';
CREATE SEQUENCE SEFAZ_SEG.seq_erro_processo_auditoria INCREMENT BY 1 MINVALUE 1 START WITH 1 NOMAXVALUE NOCYCLE NOCACHE NOORDER;
create or replace procedure SEFAZ_SEG.salvar_log_erro (p_nome_programa varchar2,p_codigo_erro_sql number,
p_erro_mensagem_sql varchar2,
p_erro_mensagem varchar2) as
PRAGMA AUTONOMOUS_TRANSACTION;
begin
insert into SEFAZ_SEG.erro_processo_auditoria (id_erro_log, nome_programa, codigo_erro_sql, erro_mensagem_sql,
erro_mensagem, data_erro)
values(SEFAZ_SEG.seq_erro_processo_auditoria.nextval, p_nome_programa, p_codigo_erro_sql, p_erro_mensagem_sql,
p_erro_mensagem, current_timestamp);
commit;
end;
/
```




4. Monitoramento do Serviços

4.1. Introdução

Este documento descreve a implementação de uma solução para Controle de Eventos Anormais e Suporte ao Controle de Qualidade para o Novo Sistema Tributário da SEFAZ-TO baseado no documento de definição de Monitoramento dos Serviços do documento de Serviços Transversais. Para detalhes da definição consultar o documento referido (versão 20150312).

4.2. Solução para Controle de Eventos Anormais

As situações anormais ocorridas nos sistemas deverão ser interceptadas com procedimento padrão, de forma que, a mensagem que surja ao usuário seja amigável, enquanto que, os detalhes do problema serão encaminhados aos responsáveis para apuração, tratamento e correção.

A falta de um tratamento estruturado de exceção é um inconveniente para a criação de um sistema confiável, portanto assim a inclusão de um sistema bem projetado de exceções permite um equilíbrio adequado entre eficiência, portabilidade e confiabilidade.

Um evento anormal é aquele que pode surgir na execução de uma operação e que está relacionada com tal ação. Para o seu tratamento requer uma mudança imediata no curso normal de execução (captura de erros). Os tipos de eventos anormais podem ser: erro de software e erro de aplicação.

No caso de um evento anormal (exceção) é necessário propagar os erros para os níveis superiores (comunicação entre as camadas: persistência, lógica de negócios e apresentação) para ser tratado adequadamente. Não deve perder a precisão da exceção quando mude para outro nível mais elevado (até o usuário final). Não pode haver exceções em pontos redundantes, ou seja, não pode ser chamada a mesma exceção para um evento específico do negócio em vários pontos diferentes dentro do sistema.

Erro de software: estado válido para o qual o sistema chega devido à presença de erros (bugs). Os erros podem-se manifestar de forma intermitente ou transitória. Existem também aqueles erros que emergem com certeza sob um conjunto de condições conhecidas. Esses erros geram resultados não desejados. Por exemplo: 1) erro de cálculo da conta corrente do contribuinte por não considerar que o contribuinte está isento de pagar um imposto por sua atividade econômica. 2) Erro no cálculo de multas e juros em uma cidade que teve um dia de férias em seu prazo de pagamento.

Erro de aplicação: Estado inválido para o qual o sistema chega devido a circunstâncias que acontecem de uma forma justificável e inevitável durante a execução. Esses erros podem ser



incidentes (quando eles são previsíveis e enumeráveis) ou emergências (quando são situações raras e pouco frequentes, embora possam ser previsíveis, eles não coincidem com o atual nível de abstração. Por exemplo: 1) Dividir uma dívida em 0 parcelas, 2) Perda de comunicação do servidor de aplicativo e o banco de dados, 3) O valor de um pagamento seja um número negativo.

Para os erros de aplicação de tipo persistência é necessário ter um mapeamento (tradução Inglês para Português) dos erros mais comuns de Oracle Database. A recuperação de erro a nível transacional deve-se usar rollback. Savepoint (check point) são aceitos sempre e quando eles garantir a integridade das transações.

Para os erros de interface com sistemas externos (BPM, GED, etc.) deve-se garantir que não existem transações inconsistentes ou incompletas no caso de um erro.

4.2.1. Desenho da Solução

Java tem sua própria implementação de captura e propagação de erros usando a declaração da cláusula **throws** em cada método usado. Para olhar o uso das exceções declarativas em Java pode ver o <https://docs.oracle.com/javase/tutorial/essential/exceptions/declaring.html>.

As vantagens da utilização de exceções declarativas são:

- Separação clara de código de controle de erro do código fonte normal.
- Não exigem verificação e propagação do valor de status ou de erro
- O programa não pode evitar
- Complexidade reduzida
- Mais fácil de ler
- Mais eficiente
- Menos erros

Na fase de projeto do sistema deve-se identificar cada uma das exceções para cada módulo ou componente do Novo Sistema Tributário. Estas exceções devem ser documentadas e indicar os motivos possíveis que gera a exceção.

É importante identificar (número) exclusivamente um erro e que isto (identificação) seja mostrado para o usuário final, quando o erro aparece em qualquer tela que ele está usando assim reduzir o tempo de pesquisa no repositório de erros pelo Equipe de Suporte. E recomendável (<http://docs.oracle.com/javase/8/docs/api/java/util/UUID.html>) usar a Classe Java UUID, por exemplo:

```
import java.util.UUID;  
UUID uuid = UUID.randomUUID();  
String randomUUIDString = uuid.toString();
```

4.2.2. Gestão de Exceções

No documento de Arquitetura Do Sistema (20150820) é descrito o uso de exceções no sistema, seção 4.8.5 Gestão de Exceções (página 46).

4.2.3. Registro de Exceções

No documento de Serviços Transversais, implementação de Gestão De Segurança (20151120) na seção Registro de Navegação é descrito como podem ser cadastrados eventos no sistema em arquivos planos em 4.1 Esquema de Captura de Dados (página 15).

Outro produto que pode ser usado é Apache Nifi (<https://nifi.apache.org/>).

4.2.4. Casos de uso para Controle de Eventos Anormais

Aqui está a lista dos casos de uso criadas na definição. Os casos 1, 2 e 3 são, certamente, os casos de uso que devem ser desenvolvidos pela fábrica de software. Para o caso de uso 4 são atividades de tipo Operations (DevOps) de suporte que podem ser implementadas com o uso de ferramentas como: System Center 2012 R2 (Service Manager), Mantis Bug Tracker, Bugzilla, WebIssues. Minha recomendação é o uso de MantisBT. Deve-se integrar uma solução para cadastrar os eventos anormais em MantisBT. Os casos de uso 2 e 3 serão adicionados no caso de uso 1, porque são funcionalidades (tarefas) extensíveis que não precisam um caso de uso individual.

4.3. Implementar o uso do Zabbix para dar Suporte ao Controle de Qualidade

Para que se tenha o controle da qualidade das aplicações desenvolvidas, tempos de resposta, transações concluídas com sucesso ou não, deverá ser criado procedimento padronizado de controle que registre os eventos e permita a sua gestão.

O Zabbix é um produto gestado pela SEFAZ-TI.

5. Gestão de Comunicação Eletrônica

5.1. Desenvolver solução para comunicação eletrônica via e-mail



Desenvolver solução para comunicação eletrônica via e-mail utilizando a biblioteca de e-mail do Spring Framework

5.1.1. Solução Proposta para enviar e-mails desde Spring

Uma consideração importante é as contas de correio eletrônico que vai usar na SEFAZ-TO. Essas contas devem estar criadas y acessíveis através do VMware Zimbra.

Spring Framework suporta o envio de e-mail. É muito fácil a configuração, siga estes passos:

Paso1: Configuração no arquivo applicationContext.xml

```
<bean id="mailSender" class="org.springframework.mail.javamail.JavaMailSenderImpl">
  <property name="host" value="mail.mycompany.com"/>
</bean>

<!-- this is a template message that we can pre-load with default state -->
<bean id="templateMessage" class="org.springframework.mail.SimpleMailMessage">
  <property name="from" value="customerservice@mycompany.com"/>
  <property name="subject" value="Your order"/>
</bean>
```

Paso2: No lugar do Paso1 pode usar uma configuração para Spring Boot usando:

```
@Configuration
public class MailConfig {

    @Value("${email.host}")
    private String host;

    @Value("${email.from}")
    private String from;

    @Value("${email.subject}")
    private String subject;

    @Bean
    public JavaMailSender javaMailService() {
        JavaMailSenderImpl javaMailSender = new JavaMailSenderImpl();
        javaMailSender.setHost(host);
        return javaMailSender;
    }

    @Bean
    public SimpleMailMessage simpleMailMessage() {
        SimpleMailMessage simpleMailMessage = new SimpleMailMessage();
        simpleMailMessage.setFrom(from);
        simpleMailMessage.setSubject(subject);
        return simpleMailMessage;
    }
}
```

Paso 3: Os métodos para enviar o correio (aqui um **EXEMPLO**, a fábrica tem que melhorar esse desenvolvimento, não só copiar)

```
public void sendMail(String from, String[] to, String subject, String msg, List attachments) throws
MessagingException {

    // Creating message
    sender.setHost("smtp.gmail.com");
    MimeMessage mimeMsg = sender.createMimeMessage();
    MimeMessageHelper helper = new MimeMessageHelper(mimeMsg, true);
    Properties props = new Properties();
    props.put("mail.smtp.starttls.enable", "true");
    props.put("mail.smtp.auth", "true");
    props.put("mail.smtp.port", "425");

    Session session = Session.getDefaultInstance(props, null);

    helper.setFrom(from);

    helper.setTo(StringUtils.join(to, ","));

    helper.setSubject(subject);
    helper.setText(msg + "<html><body><h1>hi welcome</h1><body></html>", true);

    Iterator it = attachments.iterator();

    while (it.hasNext()) {
        FileSystemResource file = new FileSystemResource(new File((String) it.next()));
        helper.addAttachment(file.getFilename(), file);
    }

    // Sending message
    sender.send(mimeMsg);
}
```

Aqui tem um exemplo para envios a várias pessoas com arquivos anexo e formato html. O importante gerar vários métodos com diferentes funcionalidades, por exemplo: enviar só um destinatário, enviar com anexos, enviar sem anexo, enviar e-mail simples, enviar e-mail html, etc.

Outra alternativa é Apache Commons E-mail (<http://commons.apache.org/proper/commons-email/userguide.html>). Ele suporta e-mail com anexos, html formatados, segurança, etc.

5.2. Desenvolver solução para comunicação eletrônica via SMS

Desenvolver solução para comunicação eletrônica via SMS utilizando chamadas Web Service ou URL do serviço de SMS Gateway via Internet com Shortcode contratado.

5.2.1. Solução Proposta

Se SEFAZ-TO precisa enviar comunicações aos contribuintes usando uma interface SMS, deve-se comprar um dispositivo SMS Gateway, onde o SDK do aparelho deve suportar Java. Esse SDK deve ser parte da solução que vai comprar. Ou também pode-se considerar a opção do serviço de envios das mensagens usando Web Services com alguma telefônica móvel de Tocantins. Isto é uma consideração administrativa.