



UNIVERSIDADE FEDERAL DE MATO GROSSO
INSTITUTO DE COMPUTAÇÃO
COORDENAÇÃO DE ENSINO DE GRADUAÇÃO
EM CIÊNCIA DA COMPUTAÇÃO

**UTILIZANDO A FERRAMENTA GENEXUS PARA A
IMPLEMENTAÇÃO DE UM MODELO DE PROECSSOS NA
ÁREA DE GESTÃO DE SAÚDE HOSPITALAR**

NICOLLAS AUGUSTO FARIAS SANTOS

Cuiabá – MT

2014

UNIVERSIDADE FEDERAL DE MATO GROSSO
INSTITUTO DE COMPUTAÇÃO
COORDENAÇÃO DE ENSINO DE GRADUAÇÃO
EM CIÊNCIA DA COMPUTAÇÃO

**UTILIZANDO A FERRAMENTA GENEXUS PARA A
IMPLEMENTAÇÃO DE UM MODELO DE PROECSSOS NA
ÁREA DE GESTÃO DE SAÚDE HOSPITALAR**

NICOLLAS AUGUSTO FARIAS SANTOS

Relatório apresentado à Coordenação do
Curso de Ciência da Computação da Universidade
Federal de Mato Grosso, para obtenção do título
de Bacharel em Ciência da Computação.

Cuiabá – MT

2014

UNIVERSIDADE FEDERAL DE MATO GROSSO
INSTITUTO DE COMPUTAÇÃO
COORDENAÇÃO DE ENSINO DE GRADUAÇÃO
EM CIÊNCIA DA COMPUTAÇÃO

NICOLLAS AUGUSTO FARIAS SANTOS

Relatório de Estágio Supervisionado apresentado à Coordenação de Estágio como uma das exigências para obtenção do título de Bacharel em Ciência da Computação da Universidade Federal de Mato Grosso

Aprovado por:

Prof. Nilton Hideki Takagi
Instituto de Computação
(Coordenador do Estágio Supervisionado)

Prof. Nelcilenio Virgílio de Souza Araújo
Instituto de Computação
(Orientador)

Leonardo Clemente Porto
SUPERVISOR

DEDICATÓRIA

À Deus graças pelo dom da vida.

*À minha família pelo apoio ao longo do curso, principalmente nas horas de
Incertezas e dúvidas, à todos meus colegas que sempre se mostraram prestativos,
fora e dentro de sala de aula, aos colegas de trabalho que me ajudaram com sua
experiência, e pôr fim aos professores com os quais tive
Oportunidade de formar uma amizade.*

AGRADECIMENTOS

Aos professores em geral do Departamento de Ciência da computação da UFMT, que foram responsáveis por grande parcela da minha formação profissional.

Ao professor Nelcileo Virgilio de Souza Araujo, do Departamento de Ciência da computação da UFMT pela atenção, ensinamentos e orientação que formaram importantes contribuições para o desenvolvimento deste trabalho.

A família que me apoio em todos os momentos me motivando e inspirando para a realização profissional, além dos momentos de incertezas.

Aos demais colegas de trabalho que sempre discutiam idéias sobre os problemas encontrados durante no trabalho.

SUMÁRIO

LISTA DE FIGURAS.....	7
LISTA DE SIGLAS E ABREVIATURAS	8
RESUMO.....	9
1. INTRODUÇÃO.....	11
2. REVISÃO DE LITERATURA	12
2.1 FERRAMENTA DE DESENVOLVIMENTO <i>GENEXUS</i>	12
2.2 JASPER REPORTS.....	15
2.2.1 <i>IREPORT</i>	16
2.3 GESTÃO DE PROCESSOS DE NEGÓCIO - BPM.....	17
2.4 MODELO DE PROCESSO DE SOFTWARE ITERATIVO E INCREMENTAL.....	19
3. MATERIAIS, TÉCNICAS E MÉTODOS	22
4. RESULTADOS	32
5. DIFICULDADES ENCONTRADAS	37
6. CONCLUSÕES	38
7. REFERÊNCIAS BIBLIOGRÁFICAS	39
APÊNDICE 1: Bandas de relatório do <i>iReport</i>	40
APÊNDICE 2: Elementos de Relatório do <i>Ireport</i>	42
APÊNDICE 3: <i>GeneXus Commit</i> e <i>Update</i>	44
ANEXO I – Organograma da Organização Ábaco.....	45
ANEXO II - Documento para Levantamento de Requisitos	46
ANEXO III - Documento para Realização dos Testes	48
ANEXO IV - Sistema de Gestão de Saúde AMETISTA.	49

LISTA DE FIGURAS

<i>Figura 1: GeneXus Janela Inicial.</i>	14
<i>Figura 2: Bandas do iReport.</i>	16
<i>Figura 3: Elementos do BPM</i>	18
<i>Figura 4: Modelo de Processo de Software Iterativo e Incremental.</i>	20
<i>Figura 5: Sistema Eventum utilizado pela Empresa, no setor do projeto Ametista.</i>	22
<i>Figura 6: Tela para a abertura de um Novo Chamado na ferramenta Eventum.</i>	23
<i>Figura 7: Fluxo do Setor de Operações Continuadas</i>	26
<i>Figura 8: Janela de Nota Interna do Sistema Eventum</i>	27
<i>Figura 9: Total de Esforços da equipe dosistema Ametista - MT-Saúde</i>	27
<i>Figura 10: Página do Share Point, exibindo as Pasta do Ametista MT -Saúde</i>	28
<i>Figura 11: Tela Inicial do Sistema Ametista Mt-Saúde</i>	29
<i>Figura 12: Exemplo de uma consulta sendo realizada no PL/SQL</i>	30
<i>Figura 13: Opção de Restaurar o Objeto no Histórico do GeneXus</i>	31
<i>Figura 14: Exemplo de pedidos realizados no mês de julho.</i>	35
<i>Figura 15: GeneXus Commit de Objetos do sistema AMETISTA.</i>	44
<i>Figura 16: GeneXus Update de Objetos do sistema AMETISTA.</i>	44
<i>Figura 17: Organograma da empresa Ábaco Tecnologia e Informação LTDA</i>	45

LISTA DE SIGLAS E ABREVIATURAS

LTDA	Limitada
MT	Mato Grosso
EV	<i>Evolution</i>
SRL	Sociedade com Responsabilidade Limitada
BD	Banco de Dados
XML	<i>Extensible Markup Language</i> - Linguagem Extensível de Marcação Genérica
JRXML	<i>JasperReports Extensible Markup Language</i> - Linguagem Extensível de Marcação Genérica do <i>JasperReports</i>
BPM	<i>Business Process Management</i> - Gestão de Processo de Negócios
TI	Tecnologia da Informação
XP	<i>eXtreme Programming</i> - Programação Extrema
SGQ	Sistema de Gestão da Qualidade
ISO	<i>International Organization for Standardization</i> – Organização Internacional para Padronização
CASE	<i>Computer Aided Software Engineering</i> – Engenharia de Software Auxiliada por Computador
PL/SQL	<i>Procedural Language/Structured Query Language</i> - Linguagem Procedural/Linguagem de Consulta Estruturada
SGBD	Sistema de Gerenciamento de Banco de Dados
IDE	<i>Integrated Development Environment</i> - Ambiente Integrado de Desenvolvimento
UFMT	Universidade Federal de Mato Grosso
WWW	<i>World Wide Web</i>

Implementações e Desenvolvimentos do *Software* AMETISTA.

Autor: Nicollas Augusto Farias Santos

Orientador: Nelcileo Virgílio de Souza Araújo

Supervisor: Leonardo Clemente Porto

RESUMO

Este Relatório de Estágio Supervisionado foi desenvolvido com o intuito de explicar todo o conteúdo pertinente ao período de estágio cumprido na empresa Ábaco Tecnologia de Informação LTDA. As atividades executadas na empresa durante o período de Estágio Supervisionado se referem as atividades de Implementação de *Software*. Este *Software* se trata do Modelo AMETISTA, um Modelo de Gestão de Saúde (Hospitalar), implantado no Estado de Mato Grosso, utilizado pelo plano de Saúde do MT - Saúde.

As fases de Implementação neste modelo de processo é o que merecem destaque no relatório, pois é a etapa onde o sistema sofre alterações de código e novas funcionalidades ou melhorias são acrescentadas em forma de incrementos de *software*. Para a codificação do sistema, a equipe de desenvolvimento responsável pelo Modelo AMETISTA se utiliza do *Genexus Ev1*, uma ferramenta de desenvolvimento para a plataforma *Web*.

Para que seja realizada alguma modificação necessária no sistema deve ser feito um "Chamado" no sistema *Eventum* pelo Analista de Negócio, que tem o contato com o cliente, aonde serão informado com detalhes os requisitos do pedido solicitado.

Outras fases do processo são a de realização de Testes, aonde serão verificadas se as tarefas pedidas foram integralmente realizadas, a de Versão "Liberada", que agrupa todas as modificações realizadas até a data em questão e que foram testadas para então disponibilizar para Homologação e então a própria fase de Homologação em que se realiza a atualização do sistema na máquina do cliente.

Também é de importância citar o conhecimento prático que foi adquirido na fase de Implementação do sistema AMETISTA, assim como o aprendizado da ferramenta *GeneXus*, além da necessidade do trabalho em equipe necessário para realizar as tarefas de uma organização que esta inserida no mercado há mais de 22 anos, ao qual possui suas metodologias visando a qualidade de seus produtos.

O Estágio Supervisionado trouxe uma gama de experiências importantes para a formação profissional, ampliando o conhecimento até então teórico, trazendo novas noções de programação, visão prática da Engenharia de *Software* no desenvolvimento de sistema, principalmente em se tratando da Gerência de Projetos e Ciclo de vida de *Softwares*, e a necessidade do comprometimento e responsabilidade pessoal como profissional na organização.

1. INTRODUÇÃO

Este trabalho, sendo um dos requisitos necessários para se obter o título de Bacharel em Ciência da Computação pela Universidade Federal do Mato Grosso, se trata do Relatório de Estágio Supervisionado que traz consigo as informações adquiridas durante a realização deste, como os fundamentos teóricos e práticos que foram aprendidos durante a realização das atividades, os métodos que foram aplicados e procedimentos realizados durante todo o período do Estágio Supervisionado cumprido na empresa Ábaco Tecnologia de Informação LTDA.

Tem este Relatório por objetivo expor ao corpo docente integrante desta Coordenação, ao Supervisor do Estágio Supervisionado e aos demais interessados, os conhecimentos adquiridos ao longo do curso de graduação submetidos a um ambiente no mercado de trabalho. Esta prática envolve a execução de atividades de Implementação que foram desenvolvidas no sistema de Gestão de Saúde AMETISTA.

A forma que este documento foi organizado se baseou na divisão por capítulos, contendo a Revisão de Literatura, trazendo a parte da fundamentação teórica e práticas do estágio, os Materiais, Técnicas e Métodos utilizados na área de trabalho, os Resultados obtidos após a finalização do período de estágio, as Dificuldades que foram encontradas para a realização deste e por fim as Conclusões que se trata das considerações finais.

Durante a realização do estágio foi possível praticar o que a Universidade havia sido capaz somente de ensinar, como a prática na programação de sistemas com grande porte, análise de banco de dados com milhares de registros, a convivência e iteração com os membros da equipe de trabalho e a realização de metodologias que visam o objetivo de qualidade do produto.

2. REVISÃO DE LITERATURA

De modo a fundamentar a parte prática da execução das atividades do Estágio Supervisionado, faz-se necessário algum embasamento teórico referente à: Ferramenta de desenvolvimento *GeneXus*, Ferramenta de geração de relatórios *Jasper* e *IReport*, Gestão de Processos de Negócio e o Modelo de Processo de *Software* Iterativo e Incremental.

2.1 FERRAMENTA DE DESENVOLVIMENTO *GENEXUS*

GeneXus é uma ferramenta desenvolvida pela empresa Uruguiaia Artech Consultores SRL, no qual é especializada em soluções para desenvolvimento, gerenciamento e manutenção automática de aplicativos. O objetivo era criar uma ferramenta que facilitaria o desenvolvimento de *softwares* orientados para aplicações corporativas.

A empresa é presidida por Breogon Gonda, Engenheiro de *Software* pela Faculdade de Engenharia - UDELAR, Cofundador e gênio na área da tecnologia, também considerado o engenheiro do ano em 1986 no Uruguai. O Vice Presidente da Artech é Nicolás Jodal, Engenheiro de Sistemas pela Faculdade de Engenharia - UDELAR e co-autor do projeto *GeneXus*.

A Artech (2014) define *GeneXus* como: " ... a primeira ferramenta inteligente para criar, desenvolver e manter de forma automática, aplicações multiplataforma de missão crítica que facilmente se adaptam às mudanças do negócio e às novas possibilidades oferecidas pela evolução tecnológica."

GeneXus então se trata de uma ferramenta que captura o conhecimento que esta contida nas visões dos usuários e o sistematiza em uma base de conhecimento puro, permitindo gerar aplicações em múltiplas plataformas e arquiteturas (Cross Plataforma). Sua idéia básica é automatizar tudo que for automatizável. Ou seja, se baseando nos requerimentos dos usuários, ela realiza o projeto e ainda faz a geração e manutenção automáticas da base de dados e dos programas da aplicação.

Durante sua criação, foram feitas pesquisas na qual se encontrou um paradigma que inovou as idéias da empresa: "desenhar como sabemos que é" para "desenhar como vemos", criado por Filippo Brunelleschi artista e arquiteto, no ano de 1417 e usado até hoje. A inspiração de Filippo fez todos repensarem a respeito, de modo que tudo passou de uma abordagem complexa e confusa para simples e objetiva.

Logo a Artech consultou seus funcionários, no qual cada um possuía sua visão da área em que atuava e de como deveria ficar o sistema, e depois passaram esta informação em forma de dados. Desta maneira, foram estabelecidas várias regras com os atributos, nomenclaturas e estruturas. Passou então a existir múltiplas ferramentas de auxílio, precisando apenas passar a visão dos dados.

Cada desenvolvedor desenvolve a aplicação em um alto nível, se utilizando principalmente de uma linguagem declarativa, usando o conhecimento contido na visão do usuário e o inserindo em uma base de conhecimento, sendo possível assim gerar a aplicação para diversas plataformas e arquiteturas.

Para que isto ocorra, o *GeneXus* possui um módulo de normalização, que cria e mantém uma estrutura de banco de dados ideal baseado no conhecimento dos usuários. Desta forma o Analista pode ter todo seu foco voltado apenas para o conhecimento do negócio em questão, não sendo necessário se preocupar com detalhes de implementação.

Também não é necessário que o programador se tenha um conhecimento profundo da programação em si, como era tipicamente comum, pois o *GeneXus* cria da estrutura pronta o código fonte necessário para a linguagem e base de dados que foram escolhidos no desenvolvimento. Com isto o *GeneXus* traz uma maior agilidade na produção do *software* e reduz seus custos de manutenção.

"A partir da modelagem do sistema desejado, *GeneXus* cria automaticamente o banco de dados, o código dos aplicativos, a interface do usuário para o cliente e os serviços necessários para o servidor." - (Artech, 2014)

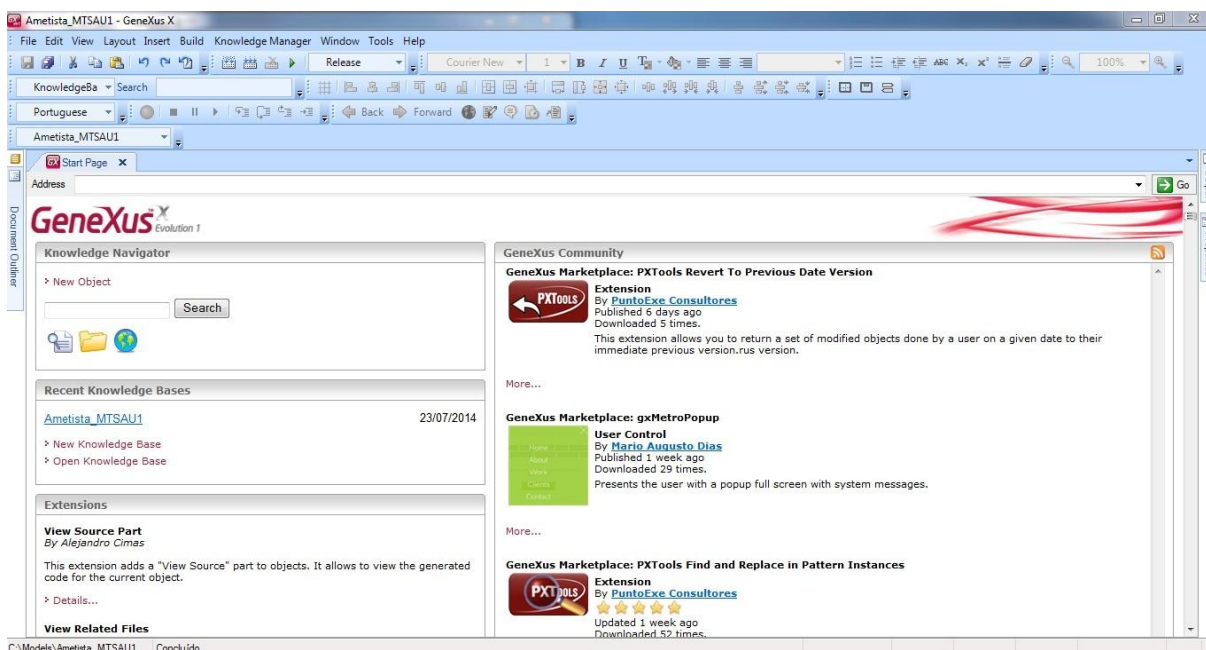


Figura 1: *GeneXus* Janela Inicial.

Foi percebido também, durante a criação do *GeneXus*, que apenas uma corporação que estivesse paralisada teria uma base de dados estável, ao qual não seria necessário alterações futuras. O problema seria como modificar um sistema já estável para que fosse capaz de utilizar uma nova base de dados que conteria enormes alterações?

Para a resolução deste problema, ao modificar algo e automaticamente ao executar, a ferramenta gerava o sistema novamente, compilando e alterando todas as tabelas, painéis, consultas, e tudo que possuía relação com a nova alteração realizada.

Caso algo fosse alterado na estruturada do BD, era feito um "*REORG*", que é uma reorganização no BD, mostrando para o usuário da ferramenta o que será afetado no processo e caso esteja de acordo, a opção de prosseguir. Desta maneira é sempre possível analisar as modificações que podem ser críticas.

Para resolver o problema com as tabelas do BD, foi utilizado a idéia de "Tabelas Ampliadas", que é uma tabela temporária, criada automaticamente e sendo usada para concatenar os dados da original. Após finalizar a concatenação, a Tabela Original é excluída e a Tabela Temporária substitui a original, voltando para o nome que possuía antes. Esta descoberta causou a revisão de pensamentos, pois não se perderiam dados e as visões dos usuários iriam continuar válidas - (Artech, 2014)

2.2 JASPER REPORTS

“JasperReports é uma solução open source poderosa e flexível para geração de relatórios. O visual designer iReport permite tirar total vantagem do poder do JasperReports sem necessidade de conhecimento profundo do formato XML nativo do JasperReports.” - (SMART, 2008)

O *JasperReports* foi criado em 2001 por Teodor Danciu, sendo uma biblioteca escrita em Java com o código fonte open source e projetada para ajudar desenvolvedores a criar relatórios tanto para aplicações *Desktop* como para *Web*. Tem como objetivo aperfeiçoar o processo de criação desses relatórios gerando-os com base em arquivos *XML* bem estruturados.

Ele possui uma flexibilidade que permite de maneira fácil a sua integração a aplicações Java de caráter empresarial, mas não possui um editor de relatórios visual integrado. Desta maneira, para se utilizá-lo diretamente se faz necessário a manipulação de sua estrutura de relatórios *XML*, o que é uma atividade relativamente técnica, que pode demandar muito tempo, ainda mais se for começar um relatório do começo, sendo necessário o uso de cálculos para o posicionamento dos componentes no relatório.

Em 9 de outubro de 2002, o italiano Giulio Toffoli lançou uma ferramenta para gerar relatórios visuais, chamada *iReport*. “[...] Sua característica era desenvolver relatórios gerando o formato *XML* padrão do *JasperReports*. Isso tornou mais acessível e intuitivo o uso dos relatórios escritos em *JasperReports*” (GONÇALVES, 2009).

Com a popularidade dessa ferramenta, em 2005 a JasperSoft (empresa mantedora do *JasperReports*) a tornou oficial na construção de relatórios para o *JasperReports*, além de continuar seu desenvolvimento.

2.2.1 IREPORT

iReport permite criar rapidamente através de uma interface gráfica rica e intuitiva qualquer tipo de relatório com facilidade, o que faz com que desenvolvedores que estão aprendendo essa tecnologia tenha acesso a todas as funções do *JasperReports*, além de agilizar o desenvolvimento de relatórios complexos e dinâmicos mesmo para usuários mais capacitados. Gonçalves (2009,p.3) enfatiza:

"[...] o desenvolvedor é capaz de criar qualquer tipo de relatório de forma simples e rápida. Mesmo sabendo que o *iReport* desenvolve um formato XML usado pelo *JasperReports*, não é difícil de manipular, há uma vantagem em usar esta ferramenta. Se o desenvolvedor é um usuário iniciante no formato XML do *JasperReports*, o *iReport* supre suas necessidades evitando que seja necessário fazer modificações no código fonte. Caso seja experiente neste formato, o *iReport* minimiza o tempo na criação dos mais complexos relatórios."

Com o *iReport*, o relatório fica dividido em diversas camadas, que na ferramenta são separadas por linhas horizontais, chamadas de *Bands* ou Bandas em português (figura 2). Cada banda possui um comportamento específico e quando estas se juntam a alguma fonte de dados, como uma consulta no BD, estas serão impressas de maneiras e tempos diferentes. Para mais detalhes das bandas, veja o APÊNDICE 1.

Title
Page Header
Column Header
Detail 1
Column Footer
Page Footer
Last Page Footer
Summary
No Data

Figura 2: Bandas do *iReport*

O *iReport* permite ao usuário "desenhar" um relatório, se utilizando de uma paleta que contém os componentes a serem utilizados, de maneira que possamos arrastá-los e solta-los no relatório (Drag and Drop), posicionando-os e lhe dando forma. Para maiores detalhes sobre os componentes utilizados pelo *iReport*, veja o APÊNDICE 2.

Desta maneira, muita das dificuldades que eram trazidas ao montar um relatório, principalmente seu layout, aonde se utilizava apenas o *XML*, começou a ficar mais simples, sem a necessidade de se aprofundar em todas as *tags* e atributos utilizados, sem mencionar que não era mais necessário se utilizar de diversos cálculos para posicioná-los corretamente.

Utilizando o *iReport*, ao salvar o relatório será gerado um arquivo com a extensão 'jrxml' que significa *JasperReportXML* e um outro arquivo compilado em Java com a extensão 'jasper', utilizado pelo JasperReport, tudo de forma automática. Para se utilizar desta ferramenta é necessário ter o Sun Java 2 JDK 1.5 ou superior instalado na máquina, pois para compilar os arquivos .jasper é necessário a instalação da distribuição do Java SE.

2.3 GESTÃO DE PROCESSOS DE NEGÓCIO - BPM

A Gestão de Processos de Negócio ou BPM (*Business Process Management*), segundo a BPMN (BALDAM et al., 2008, p.19), envolve a descoberta, projeto e entrega de processos de negócios juntamente com o controle executivo, administrativo e supervisorio desses processos.

A gestão por processos de negócios é a disciplina de modelar, automatizar, gerenciar e otimizar processos de negócios através do seu ciclo de vida com o propósito de lhes agregar valor (KHAN *apud* OLIVEIRA, 2008, p. 19)

Este conceito traz a união entre a gestão de negócios de uma empresa com a TI (Tecnologia da Informação), de modo a focar na otimização dos procedimentos que são realizados na organização, melhorando os resultados obtidos se utilizando de melhorias nos processos de negócio.



Figura 3: Elementos do BPM

Para realizar este objetivo, ele se utiliza de diversos métodos, técnicas e ferramentas para analisar, modelar, publicar, otimizar e controlar os processos que envolvem os recursos humanos, aplicações, documentos e outras informações.

Com esta integração e aperfeiçoamento dos processos, de forma que metas e estratégias sejam alinhadas com a corporação, a BPM pode trazer uma maior eficiência dos processos e permitir trazer as atividades de diferentes funções aos fatores competitivos da organização.

Sua utilização se tornou popular devido a eficiência e agilidade que trás aos processos de uma empresa que o tenha implementado, principalmente em se tratando de um mercado de trabalho competitivo, se tornando fundamental possuir as ferramentas e métodos que trazem melhor qualidade para os seus produtos.

As ferramentas que são utilizadas para estes objetivos citados podem ser chamadas de Sistemas de Gestão de Processos de Negócio que monitoram o andamento dos processos de maneira rápida e barata por sua automatização. É por meio destas ferramentas que os gestores podem analisar e gerenciar os processos baseados em dados reais.

Desta maneira a empresa pode visualizar vários aspectos dos seus processos, como por exemplo, onde estão os gargalos (processos demorados), os atrasos que certas tarefas sofrem, o quanto estão atrasados em relação as suas metas e a frequência que isto ocorre. A organização então obtém informações cruciais para seu desempenho, identificando e analisando problemas de forma fácil e ágil.

Podemos então dizer que a Gestão de Processos de Negócios ou BPM é um agrupamento de diversas atividades relacionadas e com o mesmo objetivo, que é gerar um "produto" ao cliente, visando sua qualidade e eficiência. Segundo a Fundação Nacional da Qualidade (FNQ,2008), é fundamental ainda que as organizações tenham o conhecimento de seus clientes, seus requisitos e que cada atividade adiciona o valor na busca do atendimento as suas necessidades, para assim alcançar este objetivo.

2.4 MODELO DE PROCESSO DE SOFTWARE ITERATIVO E INCREMENTAL

"... um processo de *software* é um conjunto de atividades e resultados associados que produzem um produto de *software*. Assim, um processo de *software* se dá pela construção de uma estrutura de um grupo de atividades que resultam em um produto de *software*. Este processo deve contribuir na redução de custos, aumento na qualidade e de produção." (SOMMERVILLE, 2007, p. 6).

O Modelo de processo de *Software* Iterativo e Incremental é um modelo de desenvolvimento de *Software* que acabou sendo criado em resposta ao Modelo Cascata, que era o mais tradicionalmente utilizado, devido aos seus pontos fracos. Ele pode ser visto como uma generalização do Modelo Cascata, já que se utiliza deste mas com etapas que são divididas e entregues de maneira incremental.

Os dois padrões mais conhecidos de sistemas iterativos são o *RUP* (Processo Unificado da Rational) e o Desenvolvimento Ágil de *Software*. Além disto este desenvolvimento iterativo e incremental é parte fundamental de metodologias usadas para o desenvolvimento de *Software*, como a Programação extrema (*XP*).

Segundo Bezerra (2006), um processo de desenvolvimento de software seguindo o Modelo de Processo Iterativo e Incremental divide o desenvolvimento do produto em ciclos. Nestes ciclos podem ser visto as fases de Análise, Projeto, Implementação e Testes.

Cada ciclo é associado a uma parte dos requisitos levantados, e são desenvolvidos por suas fases. Então para cada ciclo completo são feitas entregas parciais do *Software*, o que facilita a identificação e correção de erros entre os componentes, e até uma maior facilidade na mudança dos requisitos do produto como todo.

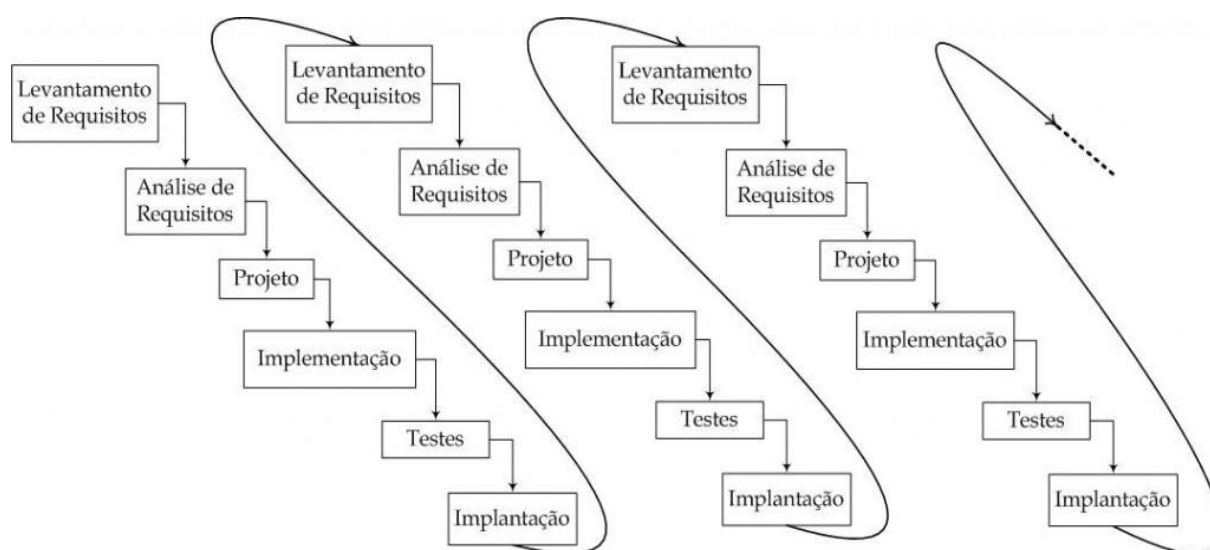


Figura 4: Modelo de Processo de Software Iterativo e Incremental

Esta entrega parcial, ou versões do sistema trazem diversos benefícios entre os quais:

- Necessidades não especificadas nas fases iniciais podem ser desenvolvidas nos incrementos.
- Cada iteração produz um conjunto de itens utilizáveis.
- Os feedbacks de iterações anteriores podem ser usados nos próximos incrementos.
- Os incrementos podem ser desenvolvidos por menos profissionais.
- Entrega dos incrementos permite o cumprimento do prazo especificado.
- Facilita a manutenção dos “módulos”.
- Redução de riscos, já que custos são aplicados a um único incremento e não ao *Software* todo.
- Maior agilidade no tempo de desenvolvimento, pois os desenvolvedores focam em resultados menores e mais claros.

Também traz as desvantagens:

- Número de iterações não pode ser definido no início do processo.
- O fim do processo não pode ser previamente definido.
- Gerenciamento e manutenção do sistema completo podem se tornar complexos.
- Gerenciamento do custo é mais complexo devido ao número de iterações (verba pode acabar).
- O usuário pode julgar a primeira versão do *Software* pensando que ele já corresponde ao todo.

3. MATERIAIS, TÉCNICAS E MÉTODOS

Atualmente, a empresa Ábaco Tecnologia de Informação LTDA se utiliza de um sistema web chamado *Eventum* para o controle dos processos exercidos por seus funcionários. É por esta ferramenta que a empresa gerencia as atividades que estão ou serão desenvolvidas, sendo possível a sua migração de funcionário e setor até a sua realização.

As atividades a serem executadas começam com a abertura de um Novo Chamado no sistema que é feito pelo "Reclamante", um funcionário da empresa com a necessidade que um serviço seja executado, como a correção de um *bug* (erro) ou o desenvolvimento de uma melhoria. O Reclamante pode abrir o chamado de acordo com o pedido de algum cliente, ou por verificar a necessidade de tal tarefa.

PDMS Ábaco (Logout) | Abrir Chamado | Listar Chamados | Busca Avançada | Associar Emails | Meus Chamados | Estatísticas | Relatórios | FAQ Interna | Ajuda

MT - CEPROMAT / MT-SAÚDE - AMETISTA-OP.CONT | Alternar

Usuário Padrão: Nicollas Augusto Farias Santos (Preferências)

palavras-chave | Buscar | id # | Exibir

Palavra(s)-chave: | Buscar

Designado: Qualquer | Categoria: qualquer | Limpar Filtros

Prioridade: qualquer | Status: qualquer

Busca Avançada | Buscas Salvas:

Filtros atuais: Esconder Chamados Fechados: Yes

Resultados da Busca (48 chamado(s) encontrado(s), 1 - 5 exibido(s)) ¹²

ocultar/exibir: [busca rápida] [busca avançada] [filtros atuais]

Prioridade	Número do Chamado	Reclamante	Data de Criação	Designado	Tempo Gasto	Categoria	Grau de Impacto	Versão	Status	Data de Mudança de Status	Data da Última Ação	Erro no ambiente	Módulo Ativo - Ametista	Objeto Caminho	Resumo	Exportar Dados: <input type="checkbox"/>	Data Esperada de Resolução
Altíssima	61900	Nicollas Augusto Farias Santos	seg, 30 jun 2014, 10:10:29 AMT	Nicollas Augusto Farias Santos	1d 6h 0m (30h 0m)	Melhorias	Alto		Concluído teste	ter, 08 jul 2014, 06:31:48 AMT	Update: 15d 4h ago				Verificar o erro ao gerar uma baixa de debito no sistema.		
Altíssima	60985	Eduardo Rahel Muniz Ferreira	ter, 10 jun 2014, 05:44:14 AMT	Leonardo Clemente Porto, Valdenice Braga de Souza	0h 0m	Melhorias	Altíssimo		Concluído Analise Negocio	ter, 10 jun 2014, 05:46:12 AMT	Update: 43d 5h ago	Produção	Mód Financeiro-Control de co-participação - Conta Corrente		PROCESSAR GUIAS DE CONSULTA SEM NUMERO		
Altíssima	60520	Eduardo Rahel Muniz Ferreira	seg, 02 jun 2014, 13:09:47 AMT	Eduardo Rahel Muniz Ferreira	0h 0m	Melhorias	Altíssimo		Concluído teste	ter, 08 jul 2014, 06:27:40 AMT	Update: 15d 4h ago	Produção	Contas Médicas		BLOQUEAR CAMPO DE VALOR E QTDE INFORMADO		
Altíssima	60234	Eduardo Rahel Muniz Ferreira	qua, 28 mai 2014, 12:10:38 AMT	Eduardo Rahel Muniz Ferreira	13h 0m	Melhorias	Altíssimo		Disponível para início de teste	ter, 15 jul 2014, 09:21:33 AMT	Note: 0d 2h ago	Produção	Prestadores		PROCESSO PARA APLICAR AS TABELAS SIMPRO E BRASINDICE		
Altíssima	60177	Eduardo Rahel Muniz Ferreira	ter, 27 mai 2014, 12:41:35 AMT	Leonardo Clemente Porto	7h 30m	Melhorias	Altíssimo		Concluído teste	seg, 14 jul 2014, 04:36:31 AMT	Update: 9d 6h ago	Produção	Beneficiário		BLOQUEAR USUÁRIOS NÃO RECADASTRADOS		

< << 1 Exibir >> >

Linhas Por Página: 5 Definir

☒ Esconder Chamados Fechados

Eventum 2.2
Copyright © 2003 - 2008 MySQL AB, 2008 - 2009 Sun Microsystems Inc.

Figura 5: Sistema *Eventum* utilizado pela Empresa, no setor do projeto Ametista.

Com o chamado criado, o reclamante tem que dizer qual a Categoria do Chamado, como exemplo: de implementação, correção de *bug* ou melhoria do sistema. Depois têm que definir qual a prioridade que este chamado deve receber e o seu grau de impacto no sistema, ambos possuindo os valores de Altíssimo, Alto, Médio, Baixo e leve. Deverá também definir qual ser a pessoa Designada para a realização da tarefa do chamado, sendo um funcionário (como ele mesmo) ou até um setor e então escrever um Resumo sobre o que se trata o chamado e sua descrição.

É possível anexar documentos ao chamado para que se haja uma maior compreensão sobre o pedido que esta sendo feito, se utilizando, por exemplo, dos documentos da empresa usados para o levantamento de requisitos ou documentos padronizados (Templates) para a descrição de problemas a serem tratados. Assim a pessoa designada pode ter todas as informações necessárias consigo através do chamado.

Abrir Novo Chamado (Projeto Atual: MT - CEPROMAT / MT-SAÚDE - AMETISTA-OP.CONT)

Categoria: * Nova implementação

Prioridade: * Altissima

Grau de Impacto: * Altissimo

Designado: * Leonardo Clemente Porto
Maria José Shimakawa
Nicollas Augusto Farias Santos
Rafaela Piedade Guimarães

Resumo: * MUDAR CALCULO DE VENCIMENTO DE CARTÃO INATIVO

Descrição Inicial: * Alterar a forma que é calculada a validade do cartão de dependentes de titulares inativo optante que são maiores de idade.

Tempo Estimado de Desenvolvimento: (em horas)

Privado: ☐ Sim ☒ Não

Módulo Ativo - Ametista: * Por favor escolha uma opção (Módulo)

Adicionar Arquivos

Arquivos: Escolher arquivo PRC-OPC-01-...-55456.doc
Escolher arquivo Nenhum arquivo selecionado

Nota: O tamanho máximo permitido no momento para arquivos enviados é 10.0 Mb

☐ Manter tela aberta para abrir novo chamado **Enviar** **Redefinir**

* Campos obrigatórios

Figura 6: Tela para a abertura de um Novo Chamado na ferramenta Eventum

O Designado do chamado, além de receber o chamado também recebe uma notificação no e-mail, empresarial, do funcionário/setor. Ao constatar este novo chamado, caso seja possível, deve inicializar a realização da tarefa, mas caso esteja ocupado deve verificar a prioridade dos itens em questão para a decisão de qual chamado deverá atender.

Ao terminar a solicitação que foi designada, deverá retornar o chamado para o reclamante que caso a tarefa tenha sido totalmente finalizada, poderá finalizar o chamado. Estes encaminhamentos, finalização e fechamento do chamado são feitos pelas mudanças de seu "*status*" pelo *Eventum*.

Estes status têm o objetivo de mostrar em que etapa o chamado esta ou estava, assim permitindo que todos os colaboradores que tenham acesso ao chamado possam saber em que situação ele se encontra. No Setor de Desenvolvimento de Operações Continuadas, ao qual se encontra atualmente o sistema AMETISTA, também são feitas as realizações destes procedimentos.

Grande parte destes chamados é aberta pelo Analista de Negócio, criando o chamado com seu status inicial, "Novo Pedido", e então iniciando a etapa de "Análise de Negócio". Ao ser concluído esta etapa, o analista anexa ao chamado o documento com os requisitos que foram levantados, com o problema a ser tratado. É enviado o chamado para a equipe de Desenvolvimento que irá continuar o processo, possuindo o conhecimento do assunto através do documento anexado, podendo decidir junto com o analista qual a melhor forma de realizar a tarefa a ser tratada, de modo a atender a necessidade do cliente.

Após estas etapas, se inicia a de Codificação, aonde inicialmente o chamado fica com o líder do projeto e com o status de "Distribuído pra Codificação". Desta maneira, o líder do projeto tem a decisão de para qual desenvolvedor será encaminhado o pedido, podendo gerir melhor sua equipe e sabendo da capacidade pessoal de cada desenvolvedor.

O Desenvolvedor escolhido irá então receber o chamado e quando começar a realizá-lo, muda o status deste para "Iniciado a Codificação". O desenvolvedor deve analisar todos os documentos em anexos para poder compreender e realizar o pedido. Como é uma etapa aonde é possível surgir várias dúvidas, principalmente com funcionários novos, o Líder ou o Analista estão sempre presentes para respondê-las e chegarem a uma solução adequada.

Caso durante o desenvolvimento de um chamado apareça outro que tenha uma prioridade maior, ou o desenvolvedor não consegue continuar a realização do chamado por algum empecilho, como a falta de informações por parte do cliente (que pode estar viajando por exemplo) e ser necessário a espera da resposta, ele deve então mudar o status do chamado para paralisado até poder dar continuidade no desenvolvimento.

Ao finalizar a codificação do chamado, o desenvolvedor muda o status dele para "Concluído Codificação" e devolve o chamado. Após isto, vem a etapa de testes, aonde o Analista irá abrir os chamados que possuem o status de "Disponível para início de teste", muda para "Iniciado Testes" e então realiza todos os testes necessários para verificar que o chamado foi atendido de forma integral. Para realizar estes testes, o desenvolvedor que havia sido designado anteriormente deve ter salvado todas as modificações realizadas e então enviado elas para o ambiente de "Desenvolvimento", aonde todos os colaboradores do projeto podem realizar a baixa destas modificações apenas atualizando sua versão local do sistema através do GeneXus. Posteriormente o Líder do projeto pode passar estas informações para o ambiente de "Homologação".

É possível também o desenvolvedor perceber que não é possível completar o chamado, por algum motivo, então ele deve devolver o chamado para a etapa anterior, de análise, justificando o motivo de tal ato. O Analista irá rever o pedido e dará um retorno, ou finalizará o chamado.

Porém, na etapa de testes, para cada cliente existe um cronograma de versão que quando alcançada, o líder da equipe gera todos os executáveis correspondentes a esta versão e executa o procedimento chamado de "Liberação de Versão". Neste momento os chamados que estavam concluídos a codificação passam para a fase de testes, mudando o status dos chamados para o "Disponível para testes", para então o analista iniciar os testes.

Caso ocorra de algum chamado que está vinculado com esta versão não poder ser atendido até a data estipulada, o líder da equipe entra em contato com o Analista que, por sua vez, entrará em contato com o cliente para através da conversa entrar em um consenso a respeito de uma nova data para a liberação deste, geralmente para a próxima data de liberação.

Figura 7: Fluxo do Setor de Operações Continuadas

Todas estas mudanças de status são feitas através de postagens de notas internas (Figura 8), onde cada responsável por dar sequência em um chamado diz que irá iniciar o processo ou o que foi feito. Através destas Notas Internas o funcionário aponta a quantidade de tempo que foi gasta para realizar a atividade.

Postar Nova Nota Interna

De: Nicollas Augusto Farias Santos <nicollas.santos@abaco.com.br>

Destinatários: Issue #63533 Lista de Notificação (Members: Eduardo Rahel Muniz Ferreira, Nicollas Augusto Farias Santos)

Título: * Re: VOLTAR STATUS DA GUIA PARA NÃO CONFERIDA

Destinatários Extras da Nota: Alessandro Henrique Pelissari, Alex Scharf, Ana Caroline Ferreira, André Luiz Costa Cruz. [Limpar Seleções](#)

Enviar nota para reclamante?: ☐ Sim ☒ Não

Adicionar Destinatários Extras à Lista de Notificação?: ☐ Sim ☒ Não

Novo Status do Chamado #63533: Paralisada codificacao

Tempo Gasto: Note Discussion (in minutes)

Tipo de Tempo: SELECIONE

Designado: Larissa Camilla Santos Ibiapina, Leonardo Clemente Porto, Maria José Shimakawa, Nicollas Augusto Farias Santos. [Limpar Seleções](#)

[Postar Nota Interna](#) [Cancelar](#)

Figura 8: Janela de Nota Interna do Sistema Eventum

Também existe o lançamento dos esforços, aonde o colaborador coloca as atividades exercidas, que não possuem chamados, como treinamentos ou reuniões, postando nestes o tempo que foi gasto para a medição do esforço total gasto pelo colaborador (Figura 9).

Total de Esforços					
Colaborador	Tempo	Hora	Extra	Extra	Filtrar
Leonardo Clemente Porto			0	0	
Israel Felipe Brunni de Oliveira			0	0	
Eduardo Rahel Muniz Ferreira			0	0	
Nicollas Augusto Farias Santos	8.390	139:50	0	0	
				00:00	

Figura 9: Total de Esforços da equipe dosistema Ametista - MT-Saúde

Além do sistema *Eventum*, existe outra ferramenta web conhecida como *Share Point* (Figura 10), aonde são compartilhados os arquivos de uso comum entre os integrantes de determinado Projeto, como os Documentos de Homologação e de Liberação de Versão, scripts e arquivos executáveis referentes a uma Versão, e arquivos de interesse em geral, como o Manual de Gestão da Qualidade, Normas diversas e *Templates* de documentos padronizados para serem utilizados por qualquer colaborador que necessite elaborar um.

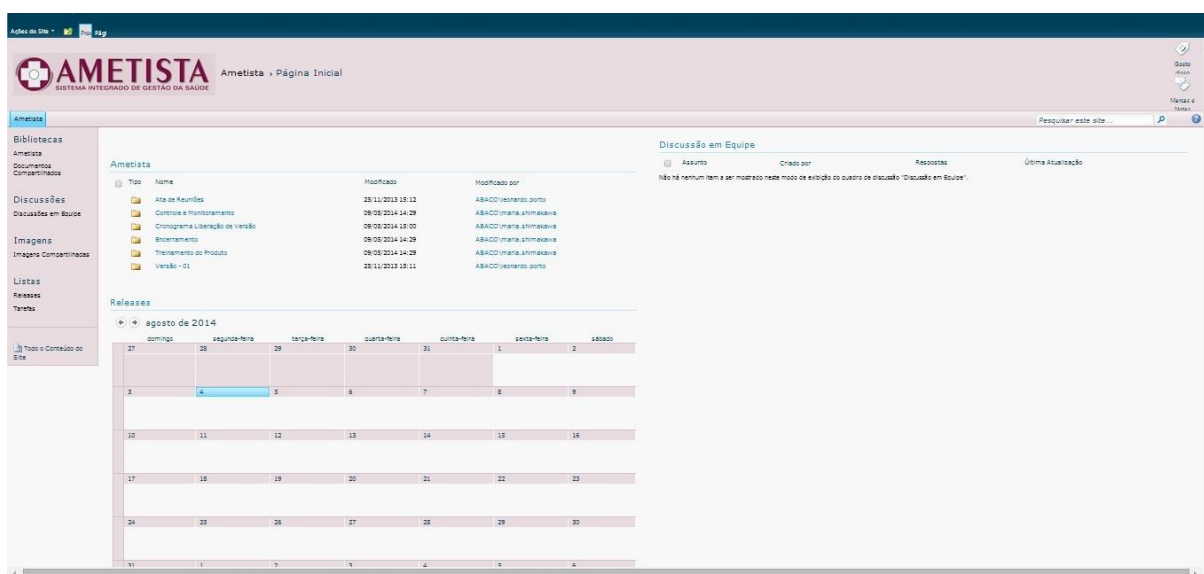


Figura 10: Página do Share Point, exibindo as Pasta do Ametista MT -Saúde

É o Sistema de Gestão da Qualidade (SGQ), que busca a certificação da qualidade de seus serviços e da satisfação do cliente, que gerencia todos os fluxos de processos das áreas de escopo da empresa (Fábrica de *Software*, Operações continuadas, Outsourcing e Treinamentos), visando garantir a total conformidade de suas ações com as normas previstas na NBR ISO 9001:2008.

A ferramenta para Desenvolvimento utilizado no sistema AMETISTA (figura 11), é o *GeneXus* em sua versão EV1. Já as principais ferramentas CASE que são usadas para dar suporte aos processos são o Microsoft Office Word, aonde são descritos os levantamentos de requisitos que foram levantados, e os documentos pertinentes ao sistema, o Microsoft Office Excel para vários fins como o cronograma de versão do sistema, o *PL/SQL Developer* para a consulta ao BD, sempre que necessário, o *JasperReport Server* para a geração de relatórios e o *IRport* para a construção destes relatórios.



The image shows the initial screen of the Sistema Ametista Mt-Saúde. At the top, there are logos for 'mt saúde', 'Governo de Mato Grosso', and 'SIAPAS'. Below the logos, there is a 'LOGIN' section with fields for 'Usuário:' and 'Senha:', and an 'Entrar' button. A link 'Primeiro acesso? leia.' is provided below the login fields. To the right of the login section, there is a large area with the 'mt saúde' and 'Governo de Mato Grosso' logos, the text 'TRABALHANDO JUNTOS !', and the contact information for the SIAPAS support system: 'E-mail para Suporte ao Sistema SIAPAS: suporte@mtsaude.mt.gov.br'. At the bottom left, there is a 'Calendário' (Calendar) showing the month of August 2014, with the 4th of August highlighted. At the bottom right, there is a 'Notícias' (News) section with a table of recent news items.

Data	Título
11/07/2014	SESSÕES DE FISIOTERAPIA
02/01/2014	Exagerou na ceia? Conheça alimentos para desinchar
02/01/2014	Doze doenças comuns no verão
02/01/2014	Estudo afirma que cirurgia comum no joelho pode ser desnecessária em muitos casos
07/10/2013	Tudo sobre Câncer de mama
10/09/2013	Videojogo melhora habilidades mentais de idosos
10/09/2013	Mais de um terço da população mundial pode ter deficiência de vitamina D
10/09/2013	Pesquisadores da Universidade Harvard anunciaram nesta sexta-feira o início dos
03/09/2013	Pela primeira vez, mais da metade dos brasileiros está acima do peso
03/09/2013	Estudo identifica 13 alterações genéticas envolvidas na esquizofrenia

Hoje: 4 de Agosto de 2014.

Mais Notícias..

Figura 11: Tela Inicial do Sistema Ametista Mt-Saúde

O *PL/SQL* é uma linguagem L4G (Linguagem de quarta geração), utilizada como uma extensão da linguagem padrão *SQL*, que fornece uma interface para o SGBD da Oracle Corporation.

A linguagem *SQL* é uma linguagem declarativa que permite fazer solicitações de uma forma relativamente simples, porém não possui uma estrutura de controle que permite, como exemplo, executar um ciclo iterativo. Assim o *PL/SQL* veio para permitir a manipulação dos dados do BD de maneira complexa, transmitindo um bloco de programação ao SGBD ao invés de enviar uma solicitação *SQL*.

Para o fim de se realizar consultas e iterações com o banco de dados utilizado pelo sistema AMETISTA, era utilizado a ferramenta *PL/SQL Developer* (Figura 12), que é um *IDE* (*Integrated Development Environment*) para desenvolvimento de programas armazenados em Banco de Dados da Oracle. Através deste, eram feitas os procedimentos que envolviam o BD, como montar consultas para exibir relatórios.

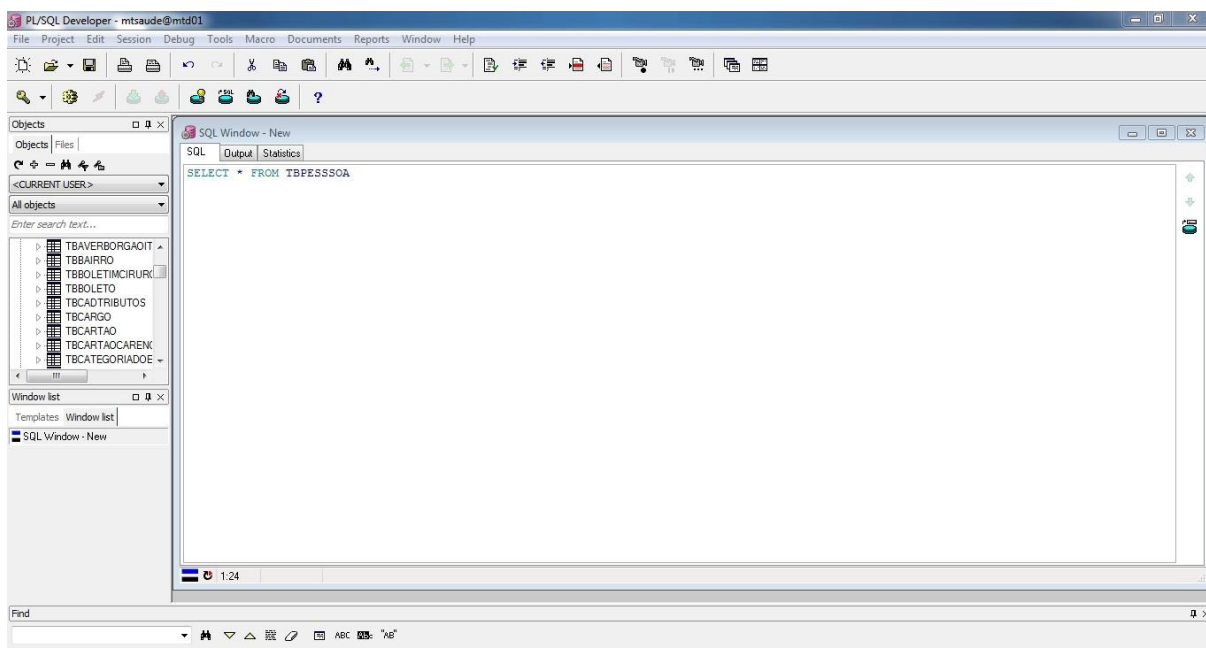


Figura 12: Exemplo de uma consulta sendo realizada no PL/SQL

O Modelo AMETISTA é dividido em um grande número de Objetos, que são codificados de acordo com os pedidos, podendo ser Web Panels, Transações, Procedimentos e Relatórios. O Desenvolvedor interage com estes Objetos de acordo com a necessidade de alterações nestes, ou ainda criam novos Objetos para a realização de um chamado, e daí iniciam seu desenvolvimento.

Para evitar que se perca as informações do Objeto original, antes das modificações realizadas, o próprio *GeneXus* possui um "*History*", ou seja, um histórico das alterações que foram feitas e salvas. Deste modo, podemos voltar para o Objeto original apenas clicando para escolhê-lo como o "Objeto Atual" (Figura 13).

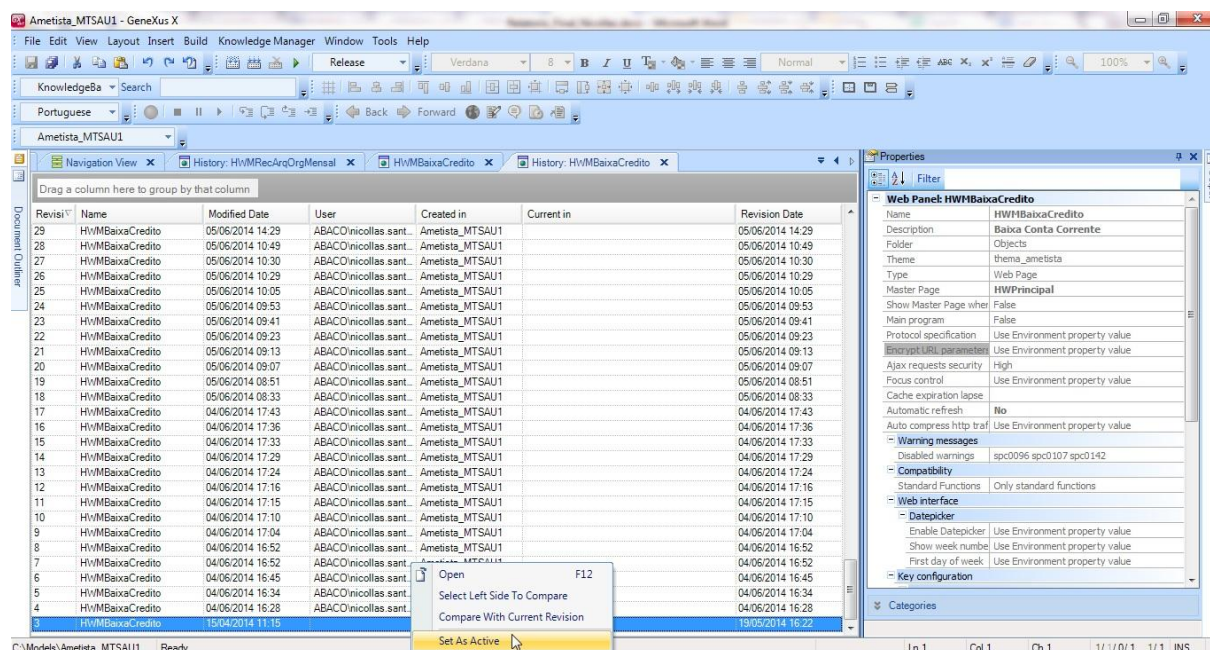


Figura 13: Opção de Restaurar o Objeto no Histórico do GeneXus

Ao ter finalizado as modificações necessárias do chamado em questão e verificado que esta tudo de acordo, o desenvolvedor realiza um "*Commit*", que é selecionar estes objetos que sofreram modificações e enviá-los para a base de homologação. No sistema AMETISTA este processo é feito pelo próprio GeneXus, aonde a base de homologação esta em um "servidor nas nuvens" chamado GX Server, oferecido pela própria empresa da ferramenta. Desta forma basta apenas realizar este *Commit* pela ferramenta, enviando os objetos selecionados por meio da internet.

Após ter realizado o *Commit*, os membros da equipe do AMETISTA podem estar realizando a transferência destes objetos que foram enviados para sua versão local do sistema, realizando um "*download*" destes. Para isto é possível utilizado o próprio GeneXus, que oferece a opção de "*Update*", fazendo a busca por objetos novos que foram enviados e então transferindo e substituindo os da máquina local. Para visualizar as janelas de *Commit* e *Update* do GeneXus, veja o APÊNDICE 3.

4. RESULTADOS

O tempo despendido no estágio supervisionado, dedicado à realização das atividades propostas, foram significativas, obtendo o conhecimento do funcionamento de uma organização que já esta inserida no mercado a tantos anos, a importância da interação com os membros da equipe, com seus conhecimentos e experiências, assim como o amadurecimento profissional e a visão prática do que antes era então apenas teórico na faculdade.

Em seu começo foi o aprendizado em relação ao sistema que já havia sido implantado no cliente, suas ferramentas de apoio e desenvolvimento, além de como o Setor de Operações Continuadas operava, principalmente a sua fase de Implementação. A faculdade foi de importância para ter a base que seria usada para exercer as atividades desenvolvidas, trazendo uma maior facilidade na adaptação as dificuldades encontradas.

O modelo Ametista se trata de um sistema que faz a Gestão de Saúde, possibilitando uma gestão mais pró-ativa e eficiente, integrando as principais unidades de um Instituto de Assistência à Saúde, trazendo uma automação nos processos, o redesenho destes e a centralização das atividades operacionais.

Devido ao Sistema Ametista ser desenvolvido inteiramente utilizando a ferramenta *GeneXus*, toda a sua manutenção e alterações necessárias se utilizam dela. Foi preciso esforço para ter o conhecimento da ferramenta que possui sua linguagem própria, assim como diversas funcionalidades. Para isto foi notado a importância da união entre a equipe para que se chegasse a resultados satisfatórios.

O objetivo do Sistema Ametista é a completa administração e integração da área Assistencial do Instituto, abrangendo as Unidades de Assistência, colaborando na organização e funcionamento, garantindo que serviços prestados sejam eficientes e eficazes, com ganho significativo nos controles das ações. Veja mais no anexo IV as informações a respeito do sistema e seu escopo.

Durante o tempo no estágio foram realizados diversos chamados que se tratavam desde a implementação de novas funcionalidades, a melhorias e testes do sistema. Estes chamados eram criados a partir de reuniões e conversas com o cliente, que já utilizava o sistema e percebia a necessidade das mudanças.

Nisto existe uma diferença entre se trabalhar com um sistema que já está implantado no cliente com um em que não está. Quando o cliente já possui o sistema pode perceber facilmente do que o sistema é capaz e as necessidades que ainda não são atendidas, sendo que algumas possuem urgência em serem atendidas.

Ao finalizar as mudanças pedidas e os testes, já é possível estar demonstrando ao cliente como esta ficando o trabalho e obter a sua aprovação. Tudo é realizado nas necessidades do cliente, que esta acompanhando o progresso. Existe também atividades que acabam sendo "críticas" por envolver dados que já foram gerados pelo cliente, sendo necessário uma maior cautela.

Como exemplo, o escopo do sistema que lida com finanças, na geração de débitos e créditos dos beneficiários do Instituto de Saúde, aonde pequenos erros podem acabar gerando diferenças no valor final, sendo que mesmo centavos podem afetar o cliente. Pode ser necessário ter que vir a trabalhar diretamente com a base de produção, que é a utilizada diretamente pelo cliente, onde não se pode cometer erros nem alterações indevidas, o que causaria enormes conseqüências. É preciso muita cautela e visão clara do sistema para realizar estas tarefas.

É visto então a importância de se entender as regras de negócio que estão associadas a cada chamado, além dos conceitos e metodologias que estão empregadas. Isto é facilitado pela utilização da ferramenta GeneXus, que trabalha com uma linguagem de alto nível, fazendo com que o desenvolvedor não fique preso demais a linguagem utilizada ou a estrutura do sistema, podendo ter seu foco na aplicação das regras de negócio e a forma de se realizar as mudanças exigidas.

De fato há uma simplicidade ao se utilizar o GeneXus para o desenvolvimento de um *software*, trabalhando com os objetos que são criados na ferramenta e as suas relações. O GeneXus agiliza bastante o processo, criando de maneira automatizada a estrutura a partir da base de dados criada, se utilizando das regras de negócio implementadas.

O desenvolvedor consegue verificar as relações entre os objetos e como a sua alteração irá impactar no sistema. No caso de ser necessário visualizar ou voltar um objeto a seu "estado" antigo, o GeneXus fornece um "*history*" do objeto, o que é muito útil para não haver sobreposição entre objetos gerados pela equipe ou a perda de dados, podendo fazer comparações entre o atual estado do objeto com seu antigo.

Esta agilidade e simplicidade que o GeneXus oferece ajuda em alcançar os prazos estabelecidos com o cliente. Mas claro que este possui suas dificuldades, devido a possuir suas próprias peculiaridades e, sendo um sistema proprietário, há uma dificuldade em ter informações de seu funcionamento, assim como a busca por informações de diferentes fontes.

Como o GeneXus envolve não só a programação do sistema, como a construção da sua base de dados e regras de negócio, a faculdade foi importante para ter o conhecimento de como gerenciar esta estrutura, da visão do banco de dados, de como utilizar metodologias de trabalho e o entendimento de como realizar as tarefas em geral.

Para a construção de relatórios que eram pedidos pelo cliente, foi utilizado o IReport com o JasperServer para comunicação e geração dos relatórios no sistema. O IReport possui uma facilidade para sua utilização, no qual após ter indicado uma fonte de dados, que pode ser uma consulta no Banco de Dados no qual irá retornar os dados necessários, deve-se somente "desenhar" o relatório, arrastando e posicionando os componentes.

De fato a maior dificuldade era criar a consulta ao BD, no qual era necessário se conhecer as tabelas envolvidas, suas relações e informações, para que não retornasse dados incorretos. A "montagem" em si do relatório era mais simples, tendo que conhecer somente como as Bandas funcionavam e os componentes a serem utilizados.

No estágio foi desenvolvido diversos relatórios novos, além da revisão de todos para uma questão de padronização e verificação de erros dos dados gerados. Após ter criado o relatório e verificado que estava gerando dados corretos, vindo de forma bem estruturada, era enviado este para o servidor, o JasperServer, que armazenava o relatório e o gerava sempre que necessário, podendo se utilizar de filtros, como uma data, para isto. Este relatório pode ser visualizado em html, ou gerado para pdf, txt, entre outros...

Abaixo uma imagem que mostra um exemplo de chamados que foram atendidos no mês de julho, assim como o tempo gasto para a realização de cada um deles, o que pode ser visto pelos líderes de equipes para fins de gerência e a visão dos esforços de cada desenvolvedor.

Data	Nro Pedido	Pedido	Tempo
26/06/2014	61375	CRIAR RELATÓRIO DE EXTRATO DE UTILIZAÇÃO BENEFICIÁRIO	2100
30/06/2014	60902	Lançamento de Esforços JUNHO / 2014	800
30/06/2014	61375	CRIAR RELATÓRIO DE EXTRATO DE UTILIZAÇÃO BENEFICIÁRIO	360
30/06/2014	61900	Verificar o erro ao gerar uma baixa de debito no sistema.	1800
01/07/2014	60667	validação_baixa_arquivo_retorno_Arrecadação.	210
01/07/2014	61442	CRIAR RELATÓRIO DE VALOR GERAL COPARTICIPADO	260
02/07/2014	61391	Gerar mensalidade verificar se existe debito manual	330
11/07/2014	61154	ADICIONAR CAMPO NO EXTRATO DE UTILIZAÇÃO DO BENEFICIÁRIO	625
14/07/2014	57803	ALTERAR O FILTRO DE PESQUISA DO LANÇAMENTO DE OPM	400
15/07/2014	51462	INTEGRIDADE CADASTRO PRODUTOS	60
15/07/2014	58728	Regra no cadastro de beneficiário	335
15/07/2014	60234	PROCESSO PARA APLICAR AS TABELAS SIMPRO E BRASINDICE	780
18/07/2014	62855	CORRIGIR PROCESSO DE APLICAÇÃO DE TABELA DE PRODUTOS PARA PRESTADORES	370
18/07/2014	62863	ACRESCENTAR CAMPO ESPECIALIDADE NO RELATÓRIO	60
18/07/2014	62892	ACRESCENTAR CAMPO NO EXTRATO DE UTILIZAÇÃO DE COPARTICIPAÇÃO	60

Figura 14: Exemplo de pedidos realizados no mês de julho

No final foram finalizadas diversos chamados que variavam dependendo das necessidades do cliente, além de ser gasto bastante esforço para o estudo das ferramentas utilizadas, na revisão de chamados para a garantia que este estava de acordo com o pedido, testes do sistema para verificar seu funcionamento e no caso de problemas a sua solução, a análise dos processos da organização, entre outros. Muitos destas atividades foram realizadas de maneira proativa para obter rapidamente o conhecimento necessário e adequação a organização.

Entre estas tarefas realizadas, estão:

- Desenvolvimento e Melhoria do sistema utilizando a ferramenta *GeneXus* em sua versão *EVI*;
- Construção de Relatórios utilizando a ferramenta *iReport 5.5.0*.
- Interação com o BD utilizado pelo sistema, usando a ferramenta *PQ/SQL Developer Oracle 8.0.3.1510*, para a comunicação.
- Análise e Levantamento de requisitos.
- Participação de reuniões e treinamentos a respeito do sistema e das ferramentas.
- Realização de testes em tarefas realizadas, verificando a existência de problemas, e caso exista, desenvolvendo sua solução.

Apesar de no final do estágio supervisionado ainda possuir diversas necessidades ainda para o sistema do cliente, se via claramente o crescimento do sistema ao longo do tempo, sendo gratificando ter contribuído para que este alcançasse um maior nível de excelência, tornando-o, a cada nova implementação ou melhoria, mais completo. Tudo acabou superando as expectativas pessoais e profissionais, trazendo novas visões e conhecimentos que agregaram para a formação profissional.

5. DIFICULDADES ENCONTRADAS

Para ingressar na empresa, foi realizada antes a participação de um curso administrado pela própria empresa e após seu termino foi feito o recrutamento das pessoas participantes para fazerem parte da empresa. Antes deste curso que era focado no treinamento de uma ferramenta de desenvolvimento chamada *GeneXus*, não havia tido informações a respeito dela.

Apesar de ser inteiramente novidade a ferramenta *GeneXus*, as suas ideias e conceitos já me eram familiares pelos anos vivenciados na universidade, o que tornou o seu entendimento mais suavizado. Mesmo assim, era possível perceber que esta ferramenta possuía suas próprias particularidades em comparação com o que já tinha conhecimento.

O que dificultou muito ainda é o fato da ferramenta *GeneXus* ser de propriedade intelectual privada, ou seja, é uma ferramenta privativa aonde se é necessário um pagamento para a utilização da mesma. Por este fato, se encontra pouco a respeito da ferramenta e problemas específicos se tornam ainda mais complicados de serem resolvidos, mesmo com a ajuda dos colegas de trabalho que eram mais experientes.

Outra dificuldade também foi, logicamente, a compreensão das Regras de Negócio utilizada pelo sistema, que são muitas devido ao próprio sistema possuir muitas funções. E o fato de ser necessário seguir todo um fluxo de trabalho, ao qual nunca havia participado, para a realização das tarefas esperadas.

Foi se necessária dedicação para a compreensão do que se era pedido, além da realização das atividades através das ferramentas disponíveis. Graças ao auxílio do líder de equipe e aos colegas de trabalho que ajudavam em qualquer necessidade, assim como o bom convívio com estes, e as horas de estudos sobre as ferramentas utilizadas, principalmente o *GeneXus*, foi possível absorver de forma rápida o conhecimento necessário para realização das atividades atribuídas.

6. CONCLUSÕES

Todo o trabalho realizado na empresa, durante o tempo do Estágio Supervisionado, apresentou-se como uma nova experiência, além de conhecimentos preciosos para a formação profissional. Foi necessário esforço e dedicação para obter noções maiores de programação de sistemas, se utilizando dos métodos da empresa.

Foi necessário o estudo e aprendizado sobre a principal ferramenta de desenvolvimento utilizada, o *GeneXus*, no qual até então não dispunha de qualquer informação. Apesar disso, a base para adquirir o manuseio desta ferramenta já tinha sido adquirido durante o curso de Ciência da Computação na UFMT.

Havia também o desafio de se compreender o sistema, que já havia sido implantado e estava em sua fase de Operações Continuadas, que também era desconhecido. Até então não havia tido envolvimento com um sistema de tamanho porte, com todas suas funcionalidades e regras de negócio. Toda esta interação foi facilitada devido a disposição dos membros da equipe que prontamente me auxiliaram em minhas dúvidas e problemas.

Outro desafio foi se adequar a metodologia da empresa, seguindo toda a sua estrutura para a qualidade e eficiência das tarefas realizadas. Foi proporcionada uma nova visão de como funciona a Gerência de Processos, sendo vista na prática como se da a gerência e administração de cada processo que ocorre, desde a etapa de Elaboração até sua Conclusão.

Devido a tudo isto, a realização deste trabalho foi uma experiência gratificante e importante, no qual se é visto a seriedade com que uma organização administra seus serviços, de maneira a estimular e exigir dos funcionários a sua dedicação e compromisso na realização de suas atividades com ela, visando certificações de qualidade dos seus produtos.

No final do período de realização do Estágio Supervisionado, posso dizer que as expectativas pessoais foram atendidas, além do grande conhecimento adquirido e experiências gratificantes, devido principalmente a colaboração dos integrantes da empresa. Tudo isto se tornará um marco para a formação profissional.

7. REFERÊNCIAS BIBLIOGRÁFICAS

SANTOS, Fabricio de Los. **GeneXus no Brasil**. Disponível em: <<http://www.fabriciodelossantos.com/Genexus.html/>>. Acesso em: 14 jul. 2014.

ARTECH. **Genexus Hoje**. Disponível em: <www.genexus.com/files/genexus-hoje-abril?pt>. Acesso em: 28 jul. 2014.

SOMMERVILLE, Ian. **Engenharia de Software**. 8ª Edição. São Paulo: Pearson Education. 2007.

GONÇALVES, Edson. **Desenvolvendo Relatórios Profissionais com iReport para Netbeans IDE**. Rio de Janeiro: Editora Ciência Moderna Ltda. 2009.

JASPERSOFT COMMUNITY. **iReport Ultimate Guide**. Disponível em: <<http://community.jaspersoft.com/documentation/iReport-ultimate-guide>>. Acesso em: 20 jul. 2014.

MACEDO, Alexandre. **Relatórios em Java – JasperReports e iReport**. Disponível em: <<http://www.k19.com.br/artigos/relatorios-em-java-JasperReports-e-irepor/>>. Acesso em: 20 jul. 2014.

SMART, John Ferguson. **Java Power Tools**. 1ª Edição. Editora O'Reilly & ASSOC. May 2, 2008.

BEZERRA, Eduardo. **Princípios de Análise e Projeto de Sistemas com UML**. 2ª Edição. Rio de Janeiro: Campus. 2006.

FNQ, Fundação Nacional da Qualidade. **Critérios de Excelência**. São Paulo: Fundação Nacional de Qualidade, 2008.

OLIVEIRA, F. M. **Aplicação do Business Process Management (BPM) nas Organizações**. 2008. 100f. Trabalho de conclusão de curso (tecnólogo) Faculdade de tecnologia da zona leste, São Paulo, 2008.

BALDAM, R. **Gerenciamento de processos de negócios. BPM - Business Process Management**. 2ª Edição. São Paulo: Érica, 2008. 240 p.

KHAN, R. N. **Business Process Management: a practical guide**. 1ª Edição. Tampa, FL: Meghan-Kiffer Press, 2004.

APÊNDICE 1: Bandas de relatório do *iReport*.

- **Title:** Essa é a primeira banda visível na impressão do seu relatório, e aparecerá apenas na primeira página.
- **Page Header:** Essa banda aparece em todas as páginas, sempre no início do seu relatório, exceto na primeira página, que ficará abaixo da banda Title. A altura especificada durante a fase de desenho do seu relatório não vai ser alterada (exceto que haja componentes redimensionáveis, como por exemplo, um sub-relatório ou um texto muito longo)
- **Column Header:** Essa banda só aparece no começo de cada interação com a banda 'detail'. Normalmente é utilizada para pôr nome nas colunas do seu relatório.
- **Detail:** Essa banda é onde ficará a exibição de seus dados. Ela vai se repetir até preencher toda a página, levando em consideração os espaços das próximas bandas. Se a página chegar no final e não foram impressos todos os dados, uma nova página será criada e o relatório começará a imprimir tudo novamente começando todo o fluxo novamente.
- **ColumnFooter:** Essa banda só aparece no final de cada interação como a banda 'detail', ela tem comportamento semelhante com a banda column header.
- **Page Footer:** Essa banda representa o rodapé da página do seu relatório, ela será impressa no final de cada página, exceto na última página, onde é impresso a banda Last Page Footer em seu lugar. É muito utilizada para informações como o número de páginas, uma data ou até mesmo o logo de sua empresa.
- **Last Page Footer:** Essa banda substitui o rodapé das páginas regulares na última página de seu relatório. Se você precisar de um rodapé diferenciado no

final de todas as páginas do seu relatório, você vai precisar utilizar essa banda. Se você não definir uma altura para ela, não será interpretada e todos os rodapés de seu relatório serão iguais aos definidos na banda Page Footer.

- **Summary:** Essa banda é conhecida como o rodapé do relatório. Ela será impressa apenas na última página logo abaixo da banda ColumnFooter. É muito comum utilizar essa banda para pôr algumas informações como um total geral do relatório ou até mesmo um gráfico.
- **Background:** Essa banda dá a possibilidade de você por uma marca d'água em seu relatório, ou qualquer outra coisa de efeito similar.
- **noData:** Essa banda é utilizada quando a fonte de dados utilizada no seu relatório está vazia. Você pode por uma mensagem personalizada, por exemplo, em caso de não houver dados. Você pode por imagens e o que quiser nessa banda.

APÊNDICE 2: Elementos de Relatório do *Ireport*.

- **Break:** Quebra uma página a partir dele.
- **Chart:** Gera um gráfico a partir de valores determinados.
- **CrossTab:** Desenvolve o que chamamos em programas de planilhas eletrônicas de tabelas dinâmicas (referencia cruzada). É muito comum utilizar em relatórios gerenciais.
- **Ellipse:** Desenha elipses no relatório.
- **Frame:** Cria quadros que podem conter outros elementos em seu interior. É muito utilizado para criar blocos dinâmicos em seu relatório.
- **HTML:** Permite utilizar marcações HTML em seu relatório.
- **Image:** Utilizado para exibir imagens em seu relatório. Podem ser imagens dinâmicas (preenchidas com alguma fonte de dados, por exemplo) ou estáticas, definindo diretamente o caminho do diretório da imagem.
- **BarCode:** Permite utilizarmos códigos de barras nos relatórios
- **GenericElement:** Permitem manipuladores personalizados para serem conectados em determinado momento da exportação do relatório.
- **List:** Possibilita criar outra query no mesmo relatório, ou seja, uma nova consulta.
- **Spiderchart:** Gráfico processado usando a biblioteca JFreeChart
- **Table:** Usado para criar tabelas.

- **Line:** Usado para inserir linhas no relatório.
- **Map:** Usado para mostrar mapas do Google Maps. Esse elemento possui dois parâmetros de configuração bem específicos que são latitude e longitude.
- **Rectangle:** Usado para criar retângulos. Contem uma cor no background e uma cor na borda. Normalmente é utilizado para separar elementos ou dar destaque em alguma informação.
- **RoundRectangle:** Possui as mesmas características do Rectangle, porem suas bordas são arredondadas.
- **Sort:** Usado para ordenar uma coluna dinamicamente.
- **StaticText:** Representa um texto estático no relatório, pode ser texto de uma linha ou de múltiplas linhas, mas sempre estático.
- **SubReport:** Cria um relatório dentro do outro. Isso é mais conhecido como master-detail, onde possuímos um relatório pai e ‘n’ relatórios filhos.
- **TextField:** Utilizado para criar os campos dinâmicos dos relatórios. É neste elemento que você se conecta a um determinado campo do banco de dados para exibir suas informações, por exemplo.

APÊNDICE 3: *GeneXus Commit e Update.*

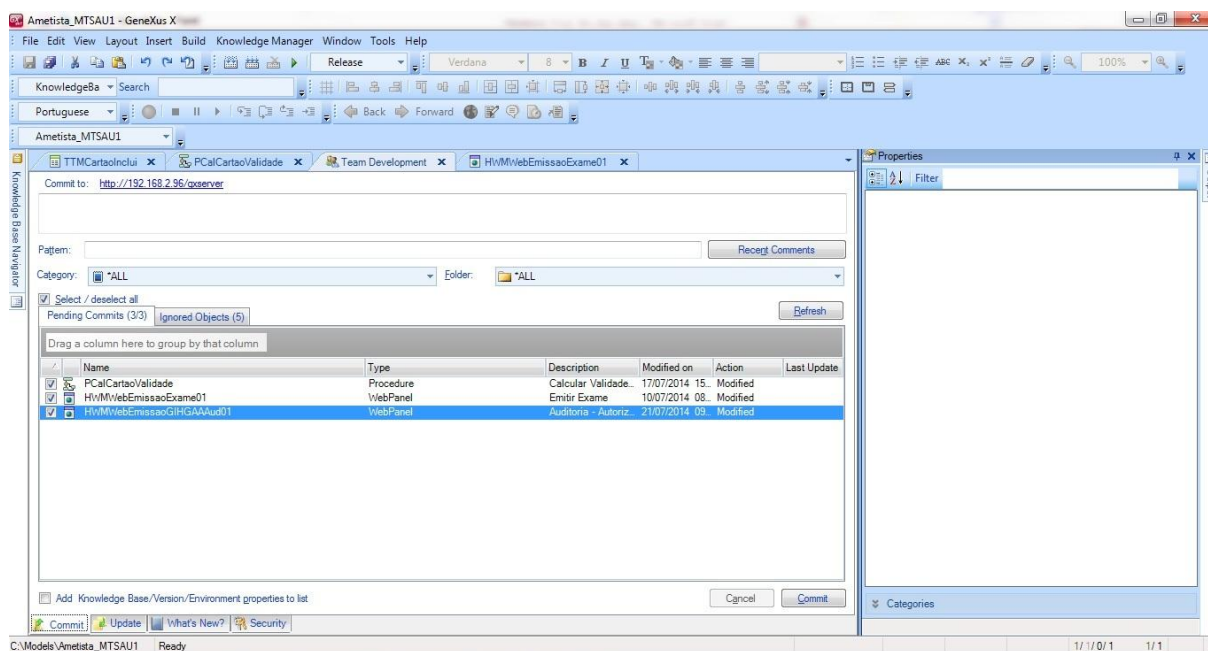


Figura 15: *GeneXus Commit* de Objetos do sistema AMETISTA.

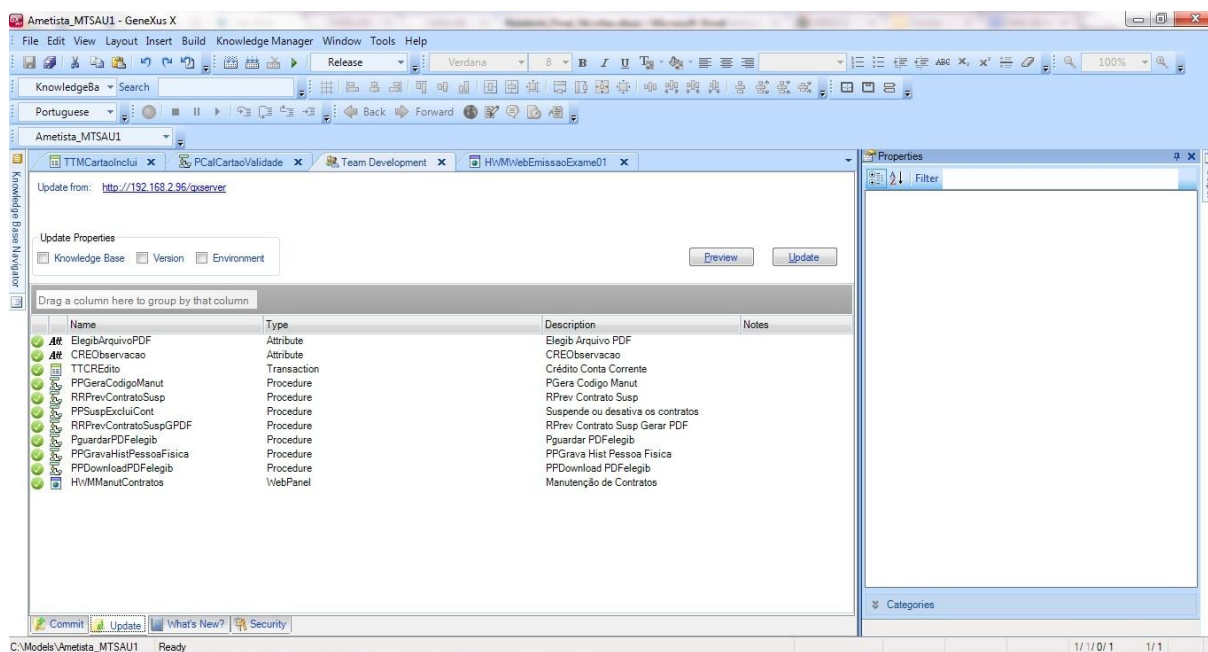


Figura 16: *GeneXus Update* de Objetos do sistema AMETISTA.

ANEXO I – Organograma da Organização Ábaco

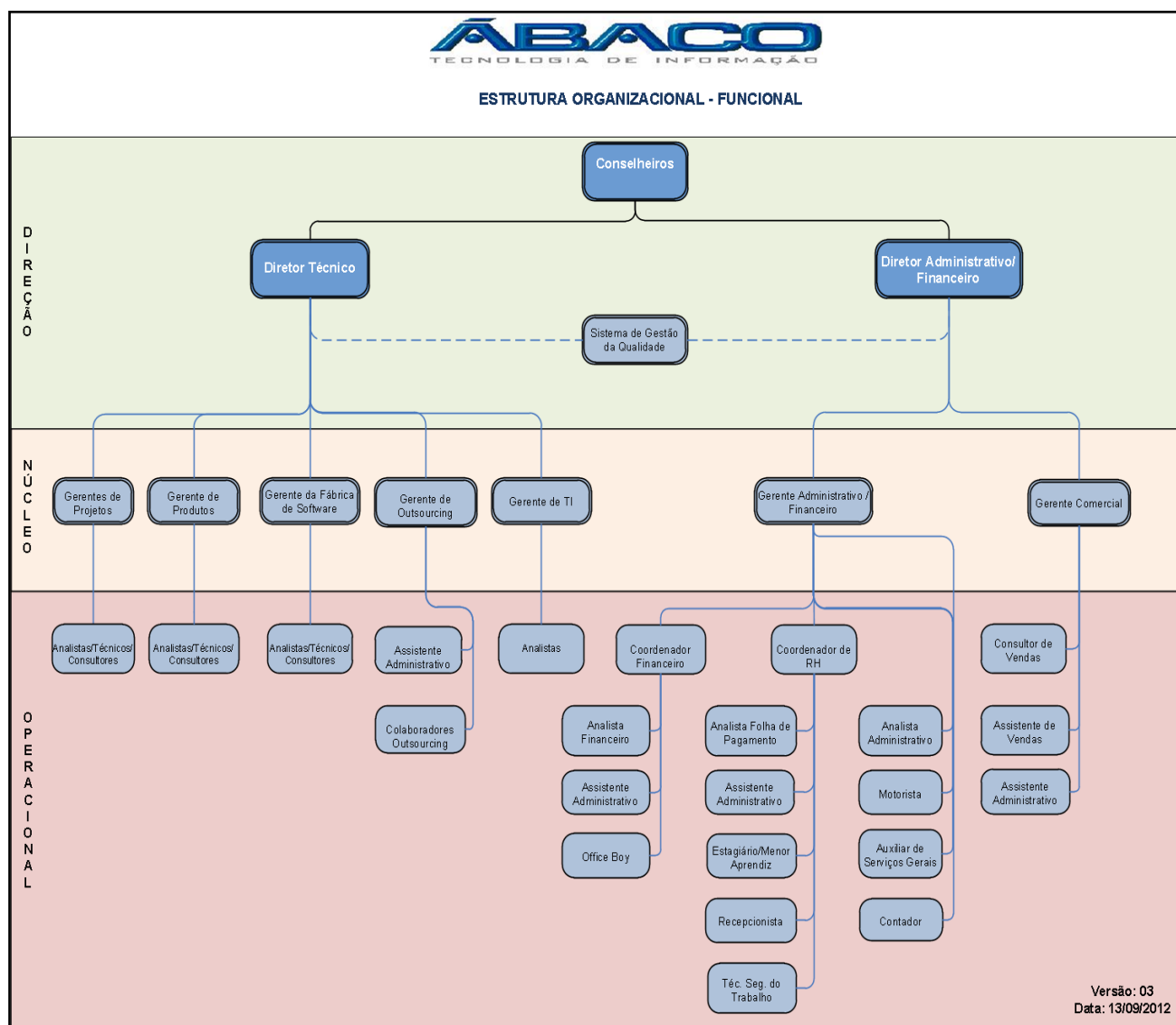


Figura 17: Organograma da empresa Ábaco Tecnologia e Informação LTDA

ANEXO II - Documento para Levantamento de Requisitos

	SGQ – Sistema de Gestão da Qualidade		
	Levantamento Simplificado de Requisitos		
	Versão	Data Criação	Data Aprovação
	15	14/06/2011	30/09/2013
	ID	Elaborado por	Aprovado por
	PRC-OPC-01	Cinthyá Gazal; Fausto Campos	Maria José Ferreira de Lima Shimakawa

LEVANTAMENTO SIMPLIFICADO DE REQUISITOS

Preparado por:	Data:
Cliente:	Sistema:

1. Objetivo

--

2. Cenário Atual

--

3. Cenário Proposto

--

4. Exceções

--

5. Impacto em outros clientes (somente quando for produto, em modelo único)

--

6. Funcionalidades que sofrerão impacto em função deste requisito

--

7. Instruções de uso da funcionalidade

--

8. Parametrizações necessárias para execução

--

9. Observações

--

Arquivo	Restrição	Página
---------	-----------	--------

PRC-OPC-01- Levantamento Simplificado de Requisitos.doc		Público	1 de 2
	SGQ – Sistema de Gestão da Qualidade Levantamento Simplificado de Requisitos		
	Versão	Data Criação	Data Aprovação
	15	14/06/2011	30/09/2013
	ID	Elaborado por	Aprovado por
	PRC-OPC-01	Cinthyá Gazal; Fausto Campos	Maria José Ferreira de Lima Shimakawa

REGISTRO DE ALTERAÇÃO		
Data	Revisado/Modificado por	Descrição da mudança
_____	_____	_____

Arquivo	Restrição	Página
PRC-OPC-01- Levantamento Simplificado de Requisitos.doc	Público	47 de 2


Observação: Os campos abaixo somente deverão ser preenchidos para os projetos que utilizam medição de horas de análise e desenvolvimento.

Total de Horas de Análise	
Total de Horas Previsto para Desenvolvimento	

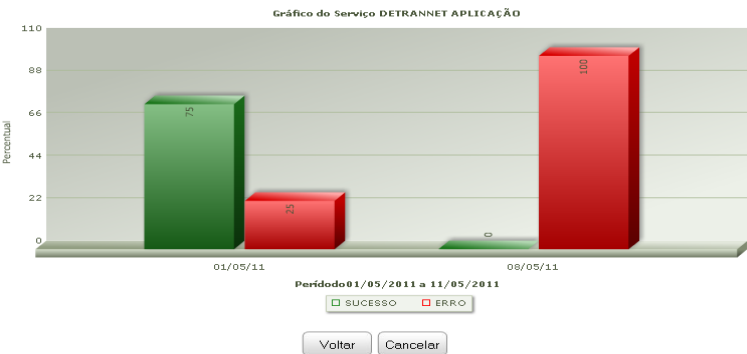
Data	Função	Nome	Assinatura
_____	_____	_____	_____

Arquivo	Restrição	Página
PRC-OPC-01- Levantamento Simplificado de Requisitos.doc	Público	2 de 2

ANEXO III - Documento para Realização dos Testes


	SGQ – Sistema de Gestão da Qualidade Documento de Teste – Operações Continuadas		
	Versão	Data Criação	Data Aprovação
	11	16/05/2011	04/06/2014
	ID	Elaborado por	Aprovado por
	PRC-OPC-03	Maria José Ferreira de Lima Shimakawa	Ana Caroline Ferreira

Data de Emissão: ____/____/____

Nome do Analista de Teste	Inserir o nome do analista responsável pela execução do teste.						
Sistema/Módulo/Funcionalidade	Nome do Projeto/sistema Nome do Módulo Nome da funcionalidade						
Número da Versão	Numero da versão do projeto						
Número do chamado	Inserir o número do chamado						
Ambiente	Inserir o ambiente (ex: Local, desenvolvimento, Homologação)						
Etapa	Inserir a informação se trata-se Testes ou Homologação.						
Correções a serem efetuadas	<p>1)Relatar os tipos de testes: (Performance, requisitos funcionais, requisitos não funcionais e outros)</p> <p>2)Relatar as inconformidades encontradas, enumerando os problemas e se possível inserindo um print da telas para demonstração da inconformidade</p> <p>3) Relatar o nome do(s) objeto(s)</p> <p>Conforme exemplo abaixo:</p> <p>1) Teste dos requisitos funcionais</p> <p>2)Na tela dos gráficos não está disponível o botão “ Imprimir” citado no caso de uso fluxo FP6.</p> 						
Registro de Alteração	<table border="1"> <thead> <tr> <th>Data</th><th>Revisado/Modificado por</th><th>Descrição da Mudança</th></tr> </thead> <tbody> <tr> <td>-----</td><td>-----</td><td>-----</td></tr> </tbody> </table>	Data	Revisado/Modificado por	Descrição da Mudança	-----	-----	-----
Data	Revisado/Modificado por	Descrição da Mudança					
-----	-----	-----					

Arquivo	Restrição	Página
PRC-OPC-03- Documento de Teste Operações continuadas.doc	Público	1 de 1

ANEXO IV - Sistema de Gestão de Saúde AMETISTA.

	<p style="text-align: center;">AMETISTA - Sistema de Gestão da Saúde</p> <p style="text-align: right;">Preparado por: Leonardo Porto</p>
---	---

Descrição

O Sistema de Gestão da Saúde (AMETISTA), surgiu para possibilitar uma Gestão mais pró-ativa e eficiente, integração entre os principais unidades de um Instituto de Assistência à Saúde, através da tecnologia em benefício da gestão, redesenho dos processos e centralização das atividades operacionais.

Justificativa

O projeto tem como objetivo a administração completa e integrada da área Assistencial do Instituto, abrangendo as Unidades de Assistência, colaborando na organização e funcionamento, garantindo assim que os serviços prestados sejam eficientes e eficazes, com ganho significativo nos controles das ações da mesma. Tudo isso, visando fornecer aos segurados uma assistência médica, odontológica e laboratorial de melhor qualidade, visando dotar ao Instituto de meios ágeis e modernos para administrar os serviços de assistência básica em saúde dos segurados.

- Falta de monitoramento e controle das atividades executadas nas unidades de assistência em tempo hábil.
- Falta de informações em tempo real.
- Desenvolvimento de vários sistemas e implantados de maneira descentralizada limitadas às suas áreas de abrangência.
- Redundância das informações que causavam várias divergências em resultados de pesquisa não dando a devida segurança aos gestores nas tomadas de decisões.
- Dificuldade para visualização de informações gerenciais de forma ágil.
- Falta de controle nos prontuários médicos.

Benefícios

AMETISTA, também possibilita:

- Apoio à área Assistencial.
- Oferecer informações sobre a gestão do negócio em tempo real.
- Prontuário Eletrônico Unificado.
- Redução dos desperdícios de recursos.
- Valorização dos profissionais da área de assistência.
- Melhoria no processo de controle, possibilitando um melhor intercâmbio e compartilhamento de informações entre as unidades de assistência.
- Melhoria no gerenciamento e controle do cadastro único do cartão do segurado.
- Facilidade de operação e segurança nos dados.
- Sistema totalmente baseado em internet (WEB).
- Extração de informações gerenciais e a visualização em gráficos.
- Gestão da escala dos médicos.
- Marcação e Atendimento das consultas totalmente integradas e automatizadas.
- Controle completo dos Exames Laboratoriais;
- Emissão de Laudos Rx , Ultra-som e Laboratoriais, via WEB.

Arquivo	Restrição	Página
SISTEMA DE GESTÃO DA SAÚDE.doc	Público	1 de 10



AMETISTA - Sistema de Gestão da Saúde

Preparado por: Leonardo Porto

Metodologia para implantação do produto

A implantação do produto é executada através do Processo de Implantação de Produto (PIP) dividido nas seguintes atividades, em consonância com as melhores práticas de gerenciamento de projetos do PMBOK e alinhadas com o processo unificado Rational (RUP):

- ✓ Institucionalização da sistemática;
- ✓ Formalização dos grupos gestores e executores;
- ✓ Divulgação das responsabilidades;
- ✓ Apresentação de todas as funcionalidades para cada gestor;
- ✓ Identificação das customizações a serem efetuadas;
- ✓ Diagnóstico da infra-estrutura existente;
- ✓ Formalização do Plano de Projeto;
- ✓ Preparação do ambiente de testes;
- ✓ Preparação do ambiente de treinamento;
- ✓ Preparação do ambiente de homologação;
- ✓ Homologação das funcionalidades customizadas;
- ✓ Formação das turmas para treinamento;
- ✓ Ministrar treinamento;
- ✓ Disponibilizar aplicativo em ambiente de produção;
- ✓ Emitir Termo de Encerramento da Implantação;
- ✓ Iniciar fase de manutenção.

Escopo do Sistema Integrado de Gestão da Saúde – AMETISTA

O Sistema de Gestão do Plano de Assistência Básica da Saúde visar dotar a administração de meios ágeis e modernos para administrar os serviços de assistência básica em saúde do Instituto. Nesta solução todas as informações pertinentes ao plano de saúde estarão centralizadas em uma única base de dados permitindo as equipes de gestão na área de saúde possam planejar ações que visem a melhorar a gestão dos serviços de assistência básica em saúde a todos os segurados do plano. O Ametista é dividido em 10 (dez) módulos, sendo:

1. **Cadastro** – Responsável pelo cadastro das Unidades de Assistência, do segurado e emissão do cartão único.
2. **Marcação e Atendimento** – Responsável pela organização do fluxo de atendimento dos segurados.
3. **Consulta Clínica** – Responsável pelo registro dos vários tipos de consultas médicas: (Saúde da Mulher, Re-hidratação Oral, Saúde do Trabalho, Acolhimento, Grupos de Tratamento).
4. **Odontologia** – Responsável pelo registro dos vários tipos de atendimentos odontológicos.

Arquivo	Restrição	Página
SISTEMA DE GESTÃO DA SAÚDE.doc	Público	50 de 10



AMETISTA - Sistema de Gestão da Saúde

Preparado por: Leonardo Porto

5. **Regulação** – Responsável por garantir e regular o uso efetivo dos Procedimentos de saúde.
6. **Laboratório** – Responsável por garantir a eficiência laboratorial, todos os procedimentos de acompanhamento dos exames são contemplados nesse módulo.
7. **Imagem e Transparência** – Responsável pelo controle de exames de diagnósticos por imagem.
8. **Ambulatório** – Responsável pela administração do funcionamento das unidades ambulatoriais.
9. **Posto de Urgência e Emergência** – Responsável pela administração do setor de Urgência e Emergências.
10. **Auditoria/Contas médicas** – Responsável pela Auditoria e Contas Médicas.
11. **Calculo Atuarial** – Geração de dados para realização de cálculo atuarial.
12. **Vacinação** – Controle de Vacinas.
13. **Financiamento** – Visibilidade e controle do crédito consignado.
14. **Financeiro** – Visibilidade das contribuições para o Plano do segurado.

Arquivo	Restrição	Página
SISTEMA DE GESTÃO DA SAÚDE.doc	Público	3 de 10