

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE
SISTEMAS**

MUNIR MACHADO MOZER

**IMPLEMENTAÇÃO DE UMA ROTINA PARA GERAÇÃO DE ORDENS
DE PRODUÇÃO**

TRABALHO DE CONCLUSÃO DE CURSO

**PATO BRANCO
2014**

MUNIR MACHADO MOZER

**IMPLEMENTAÇÃO DE UMA ROTINA DE GERAÇÃO DE ORDENS
DE PRODUÇÃO**

Trabalho de Conclusão de Curso de graduação, apresentado à disciplina de Trabalho de Diplomação, do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, da Universidade Tecnológica Federal do Paraná, Câmpus Pato Branco, como requisito parcial para obtenção do título de Tecnólogo.

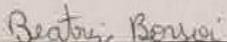
Orientador: Profa. Beatriz T. Borsoi

**PATO BRANCO
2014**

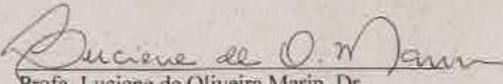
ATA Nº: 248

DEFESA PÚBLICA DO TRABALHO DE DIPLOMAÇÃO DO ALUNO MUNIR MACHADO MOZER.

Às 18:10 hrs do dia 10 de dezembro de 2014, Bloco V da UTFPR, Câmpus Pato Branco, reuniu-se a banca avaliadora composta pelos professores Beatriz Terezinha Borsoi (Orientadora), Luciene de Oliveira Marin (Convidada) e Lucília Yoshie Araki (Convidada), para avaliar o Trabalho de Diplomação do aluno Munir Machado Mozer, matrícula 950203, sob o título **Implementação de uma Rotina para Geração de Ordens de Produção**, como requisito final para a conclusão da disciplina Trabalho de Diplomação do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, COADS. Após a apresentação o candidato foi entrevistado pela banca examinadora, e a palavra foi aberta ao público. Em seguida, a banca reuniu-se para deliberar considerando o trabalho **APROVADO**. Às 18:35 hrs foi encerrada a sessão.



Profª. Beatriz Terezinha Borsoi, Dr.
Orientadora



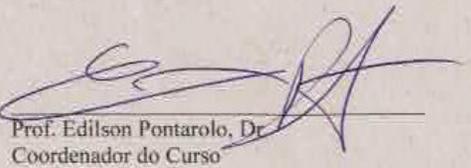
Profª. Luciene de Oliveira Marin, Dr.
Convidada



Profª. Lucília Yoshie Araki, M.Sc.
Convidada



Profª. Eliane Maria de Bortoli Fávero, M.Sc.
Coordenador do Trabalho de Diplomação



Prof. Edilson Pontarolo, Dr.
Coordenador do Curso

RESUMO

MOZER, Munir Machado. Implementação de uma rotina de geração de ordens de produção. 2014. 42 f. Monografia (Trabalho de Conclusão de Curso) - Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, Universidade Tecnológica Federal do Paraná, Câmpus Pato Branco. Pato Branco, 2014.

A automação de processos pelo uso de máquinas que possam ser programadas, como os robôs, trouxe um grande avanço para o segmento industrial. Contudo, além do aspecto de automação, a indústria pode beneficiar-se da informática por meio de sistemas de informação gerenciais sejam eles operacionais ou de apoio à tomada de decisão. Esses sistemas permitem gerenciar o fluxo do processo de negócio em termos de clientes, vendas, compras, fluxo de caixa. Esses sistemas também podem ser aplicados ao controle do processo produtivo com a alocação de matérias-primas e o acompanhamento do processo. Um sistema que permite o acompanhamento das atividades vinculadas à produção permite a geração de ordens de produção com a alocação do que é necessário para produzir, sejam matérias-primas, matérias manufaturadas, equipamentos e pessoas. O desenvolvimento de uma rotina de geração de ordens de produção é o objetivo deste trabalho. Essa rotina faz parte de um sistema maior que atende outras funcionalidades do processo produtivo. Para o desenvolvimento foi utilizada a linguagem Delphi com o banco de dados Oracle.

Palavras-chave: Ordens de produção. Banco de Dados Oracle. Linguagem Delphi. Aplicações desktop.

LISTA DE FIGURAS

| | |
|---|----|
| Figura 1 – Tela inicial do Sistema gerenciador de banco de dados Oracle SQL Developer ... | 15 |
| Figura 2 – IDE Delphi 7..... | 16 |
| Figura 3 – Diagrama de entidades e relacionamentos | 24 |
| Figura 4 – Menu acesso Tipo de Programação | 28 |
| Figura 5 – Formulário de tipo de programação..... | 28 |
| Figura 6 - Novo Item no lote de produção | 30 |
| Figura 7 – Ordens de produção geradas..... | 31 |
| Figura 8 – Filtro do cabeçalho do lote | 33 |
| Figura 9 – Filtro das programações | 33 |
| Figura 10 – Empenho das matérias-primas por ordem de produção | 36 |
| Figura 10 – Fluxo de produção..... | 37 |
| Figura 11 – Liberação por lote | 38 |
| Figura 12 – Liberação por ordem de produção | 38 |

LISTA DE QUADROS

| | |
|--|----|
| Quadro 1 – Iterações definidas | 18 |
| Quadro 2 – Requisitos funcionais estabelecidos para o sistema | 22 |
| Quadro 3 – Requisitos não funcionais estabelecidos para o sistema..... | 23 |
| Quadro 4 – Campos da tabela PC_PROGTIPO | 24 |
| Quadro 5 – Campos da tabela PC_PROG..... | 25 |
| Quadro 6 – Campos da tabela PC_PROG..... | 25 |
| Quadro 7 – Campos da tabela PC_PROGOP..... | 26 |
| Quadro 8 – Campos da tabela PC_PROGOPROT | 26 |
| Quadro 9 – Campos da tabela PC_PROGOPMAT | 26 |
| Quadro 10 – Campos da tabela PC_PROGOPCFG..... | 27 |
| Quadro 11 – Campos da tabela PC_PROGOPMPCFG..... | 27 |

LISTAGEM DE CÓDIGO

| | |
|---|----|
| Listagem 1 – Função para geração de queries no servidor | 32 |
| Listagem 2 – Função create da engine | 34 |
| Listagem 3 – Carga da programação de ordem de produção | 35 |

LISTA DE ABREVIATURAS E SIGLAS

| | |
|------|--|
| BPMS | <i>Business Process Management Systems</i> |
| CRM | <i>Customer Relationship Manager</i> |
| DB | <i>Data Base</i> |
| EAI | <i>Enterprise Application Integration</i> |
| ERP | <i>Enterprise Resource Planning</i> |
| IDE | <i>Integrated Development Environment</i> |
| IP | <i>Internet Protocol</i> |
| OPs | Ordens de Produção |
| RAD | <i>Rapid Application Development</i> |
| RF | Requisito Funcional |
| RNF | Requisito Não Funcional |
| SCM | <i>Supply Chain Manager</i> |
| SOAP | Simple Object Access Protocol |
| SQL | <i>Structured Query Language</i> |
| UDDI | Universal Description, Discovery and Integration |

SUMÁRIO

| | |
|---|-----------|
| 1 INTRODUÇÃO | 9 |
| 1.1 CONSIDERAÇÕES INICIAIS | 9 |
| 1.2 OBJETIVOS | 10 |
| 1.2.1 Objetivo Geral | 10 |
| 1.2.2 Objetivos Específicos | 10 |
| 1.3 JUSTIFICATIVA | 11 |
| 1.4 ESTRUTURA DO TRABALHO | 11 |
| 2 REFERENCIAL TEÓRICO | 12 |
| 2.1 INFORMATIZAÇÃO DE PROCESSOS INDUSTRIAIS | 12 |
| 2.2 ENGENHARIA DE PRODUTO | 13 |
| 3 MATERIAIS E MÉTODO | 14 |
| 3.1 MATERIAIS | 14 |
| 3.1.1 Oracle SQL Developer | 14 |
| 3.1.2 Delphi 7 | 16 |
| 3.1.3 Linguagem Delphi | 16 |
| 3.1.4 Banco de Dados Oracle | 17 |
| 3.1.5 DevExpress | 17 |
| 3.2 MÉTODO | 17 |
| 4 RESULTADOS | 20 |
| 4.1 ESCOPO DO SISTEMA | 20 |
| 4.2 MODELAGEM DO SISTEMA | 20 |
| 4.3 APRESENTAÇÃO DO SISTEMA | 27 |
| 4.4 IMPLEMENTAÇÃO DO SISTEMA | 32 |
| 5 CONCLUSÃO | 40 |
| REFERÊNCIAS | 41 |

1 INTRODUÇÃO

Este capítulo apresenta as considerações iniciais do trabalho, inserindo a rotina implementada no contexto de informatização da indústria, os objetivos e a justificativa e por fim a organização do texto por meio da apresentação dos capítulos.

1.1 CONSIDERAÇÕES INICIAIS

As indústrias são caracterizadas pela existência de diversos processos como os de produção, compras e vendas. O processo principal está relacionado ao produto que é obtido como realização do negócio que define finalidade da referida indústria. A linha de montagem de uma montadora de automóveis, a linha de produção de eletrodomésticos e a linha de produção para industrialização de alimentos são exemplos desses processos. Contudo, outros processos podem estar agregados ao principal, mesmo porque um produto pode ser composto por subprodutos obtidos a partir de outros processos dentro da mesma empresa.

O fluxo de trabalho de uma indústria, independentemente do produto produzido, possui basicamente o seguinte escopo: cadastro de itens utilizados na produção, definição da engenharia de produto, geração das ordens de produção, apontamento de produto acabado para que sejam gerados os saldos e gerenciamento das matérias-primas utilizadas no processo.

Engenharia de produto é a lista técnica que pode ser associada a uma receita que define o processo de construção de um produto. Essa lista é composta pelos itens (matérias-primas) necessários para a construção e pela quantidade de cada um dos itens que deve ser utilizada para compor o produto. A lista é composta pela sequência de uso desses itens pelo processo de produção. Essa sequência define um fluxo de produção. Em um sistema computacional, a engenharia de produto é representada pelo cadastro dessa espécie de receita, que são os itens necessários e as respectivas quantidades, e a sequência de uso dos itens para a composição do produto.

Utilizando a base cadastrada da engenharia de produto, a mesma é utilizada para a geração de ordens de produção. Uma ordem de produção é um registro dentro do sistema e terá um número único, que contém todas as informações relacionadas a esse número para que na realização do processo possa ser produzido o produto desejado. A ordem de produção ao ser gerada, analisa os estoques das matérias-primas, realizando a reserva das mesmas para a produção daquele produto. Esse processo é chamado de explosão da ordem de produção.

Quando o produto desejado possui subprodutos em sua estrutura, a rotina de geração de ordens de produção gera automaticamente uma ordem de produção para cada produto produzido (verificando a sua real necessidade através de configurações de estoque disponível).

Quando uma ordem de produção é liberada para a produção são realizadas as transferências das matérias-primas dos locais de estoque de armazenamento para o local de estoque de produção. No estoque de produção as matérias-primas serão consumidas por meio de uma rotina de apontamento de ordem de produção que fará a baixa das matérias-primas e realizará o suprimento do produto finalizado no estoque de venda.

A rotina de engenharia de produto foi implementada como trabalho de estágio supervisionado pelo autor deste trabalho de conclusão de curso. A definição e a implementação, por meio do desenvolvimento de um sistema computacional, para o gerenciamento da rotina de produção como uma sequência do estágio é o objetivo deste trabalho. Assim, neste texto são apresentadas a modelagem das tabelas, as consultas de banco de dados utilizadas, os códigos escritos e a implementação da rotina em si.

1.2 OBJETIVOS

A seguir são apresentados o objetivo geral e os objetivos específicos do trabalho.

1.2.1 Objetivo Geral

Implementar uma rotina de geração de ordens de produção para um sistema industrial.

1.2.2 Objetivos Específicos

- Apresentar a lógica implementada para a geração da ordem de produção a partir de uma rotina de engenharia de produto utilizando a linguagem Delphi e o Banco de Dados Oracle.
- Fornecer uma forma de facilitar o gerenciamento e o controle das ordens de produção para a indústria.

1.3 JUSTIFICATIVA

A escolha para o desenvolvimento de uma aplicação para o segmento industrial é decorrente de a indústria ainda ser um nicho de mercado altamente explorável em termos de automatização e informatização. O sistema desenvolvido, cuja rotina apresentada neste trabalho faz parte, pode ser categorizado como especialista por atender um determinado segmento de mercado e pela possibilidade de o mesmo adaptar-se às mudanças de cada cliente específico. As funcionalidades e a forma de composição do sistema visam atender diversos segmentos industriais, desde que essas indústrias possuem a sua produção baseada em engenharia de produto. E isso ocorre com a maioria das especialidades de um segmento cujo processo principal é uma linha de produção.

Para um melhor controle e automatização dos processos industriais, ordens de produção são geradas para que a fábrica saiba o que deve produzir naquele dia e em qual ordem para que todos os produtos estejam de acordo com as especificações técnicas.

O desenvolvimento do sistema foi baseado em uma indústria de plásticos, ou seja, que possui polímero como sua matéria-prima principal. Contudo, o sistema está estruturado de forma que possa ser portado para outros nichos específicos na industrialização.

A escolha do banco de dados Oracle deve-se pelo alto desempenho das aplicações que o mesmo pode proporcionar e aos recursos oferecidos como, por exemplo, a possibilidade de implementar procedimentos armazenados e gatilhos no próprio banco, embora eles não sejam exclusivos desse banco de dados. A escolha da linguagem Delphi decorre de a empresa para a qual o aplicativo foi desenvolvido utilizar essa linguagem de programação.

1.4 ESTRUTURA DO TRABALHO

Este texto está organizado em capítulos. Este é o primeiro e apresenta a introdução do trabalho, com os objetivos e a justificativa. O Capítulo 2 contém o referencial teórico que se refere a informatização de processos industriais. No Capítulo 3 estão os materiais utilizados para a modelagem e a implementação do aplicativo e o método, com as principais atividades realizadas para a modelagem e a implementação. O Capítulo 4 apresenta as funcionalidades básicas definidas para o sistema e a sua implementação. Por fim, no Capítulo 5, está a conclusão com as considerações finais.

2 REFERENCIAL TEÓRICO

Este capítulo apresenta a fundamentação conceitual para este trabalho. O referencial teórico abrange informatização de processos industriais e engenharia de produto. A engenharia de produto é parte da informatização de processos industriais. Esses processos compõem o *workflow* que define o fluxo de realização das atividades.

2.1 INFORMATIZAÇÃO DE PROCESSOS INDUSTRIAIS

No âmbito das organizações, informatização é um termo empregado de forma ampla para representar a utilização dos recursos de tecnologia de informação (ZWICKER et al., 2007). A informatização está relacionada aos processos porque visa auxiliar pessoas na realização das atividades dos processos ou mesmo realizar essas atividades ou parte delas de forma automática, sem a necessidade de intervenção humana.

A informatização de processos de negócio é crescente, no sentido de atividades serem realizadas automaticamente por aplicativos computacionais e pelas pessoas realizarem essas atividades com o auxílio desses aplicativos. Dentre esses aplicativos estão os *Enterprise Resource Planning* (ERP), *Supply Chain Manager* (SCM) e *Customer Relationship Manager* (CRM) que permitiram várias vantagens para as empresas, que abrangem desde o controle dos processos até transparência das suas contas e da adoção de melhores práticas (SCHEER, et al., 2004).

A automatização de processos de negócio evoluiu para os *workflows* que suportam, adaptam e normalizam os processos (CORREIA; O'NEILL, 2011). Nessa automatização, os aplicativos como ERP, SCM e CRM passam a ser as ferramentas que armazenam os dados de *workflow*. Esta integração de sistemas designa-se por *Enterprise Application Integration* (EAI) (SCHEER et al., 2004).

A definição, criação e gestão da execução dos *workflows* é feita por meio dos *Business Process Management Systems* (BPMS) que possuem a capacidade de interpretar os processo e as atividades utilizando as regras de negócio definidas, interagir com os atores e quando necessário invocar ferramentas e aplicações (CORREIA; O'NEILL, 2011).

Os *workflows* estão, geralmente, integrados com outros sistemas, como sistemas de gestão documental, bases de dados, sistemas de *e-mail*, sistemas de informação geográfica, aplicações de produção e sistemas independentes (PAPAZOGLU, 2008). Essas aplicações

fornece dados para o *workflow* e são alimentadas por dados provenientes das atividades que compõem os processos de negócio.

2.2 ENGENHARIA DE PRODUTO

A engenharia de produto se refere ao processo e os recursos necessários para o desenvolvimento de produtos. Chiavenato (2005) denomina desenvolvimento de produtos a área responsável pelos estudos e pesquisa sobre criação, adaptação, melhorias e aprimoramento dos produtos produzidos pela empresa.

A engenharia de produto envolve o planejamento e o projeto do produto, o processo de desenvolvimento e a gestão para o desenvolvimento de produtos.

A gestão do desenvolvimento de produtos está relacionada ao conjunto de processos, tarefas e atividades de planejamento, organização, decisão e ação envolvidos para que o sistema considerado alcance os resultados de sucesso esperados (CHENG; MELO FILHO, 2007).

O processo de desenvolvimento de produtos inclui basicamente a tradução das necessidades, interesses e conhecimento do consumidor e em uma especificação de algo que possa ser produzido (CARLOS; CARLOS, 2011). É responsabilidade desse processo desenvolver produtos que atendam às expectativas do mercado em termos de qualidade; que sejam inseridos no mercado no tempo adequado e mais rapidamente que os concorrentes; e que sejam produzidos a um custo compatível com o orçamento (TOLEDO, 2000).

Um processo pode ser definido como uma rede de atividades, com relacionamentos entre as mesmas, com critérios para indicar o início e o fim do processo e informações sobre as atividades realizadas, os envolvidos e as tecnologias de informação utilizadas e os dados vinculados às mesmas (WFMC-TC 1011, 1996).

Krishnan e Ulrich (2001) ressaltam que a atividade de desenvolvimento, planejamento e projeto de um novo produto não é uma tarefa simples, requer pesquisa, planejamento e controle consistentes e o uso de métodos sistemáticos.

3 MATERIAIS E MÉTODO

Este capítulo apresenta as ferramentas, as tecnologias e o método utilizados para a modelagem e a implementação de algumas funcionalidades do sistema. Os materiais se referem às ferramentas e às tecnologias utilizadas, como linguagem de programação, banco de dados, interface de desenvolvimento e editores para a modelagem. O método contém as principais atividades realizadas para o desenvolvimento do trabalho.

3.1 MATERIAIS

Para a modelagem e a implementação do sistema foram utilizadas as seguintes ferramentas e tecnologias:

- a) Oracle SQL Developer versão 3.2.20.09 para administração do banco de dados.
 - b) Delphi versão 7.0 como *Integrated Development Environment* (IDE) de desenvolvimento.
 - c) Delphi como linguagem de programação.
 - d) Oracle 11g para o banco de dados.
 - e) Componentes de terceiros usados em conjunto com o Delphi (Dev Express).
- Nas subseções a seguir essas ferramentas e tecnologias são apresentadas.

3.1.1 Oracle SQL Developer

Oracle SQL Developer é uma ferramenta para administração de bancos de dados Oracle que permite criar e gerenciar usuários e tabelas. A interface do aplicativo consiste, basicamente, de uma barra de ferramentas e do *Data Base* (DB) Explorer. A Figura 1 apresenta a tela inicial da ferramenta Oracle SQL Developer.

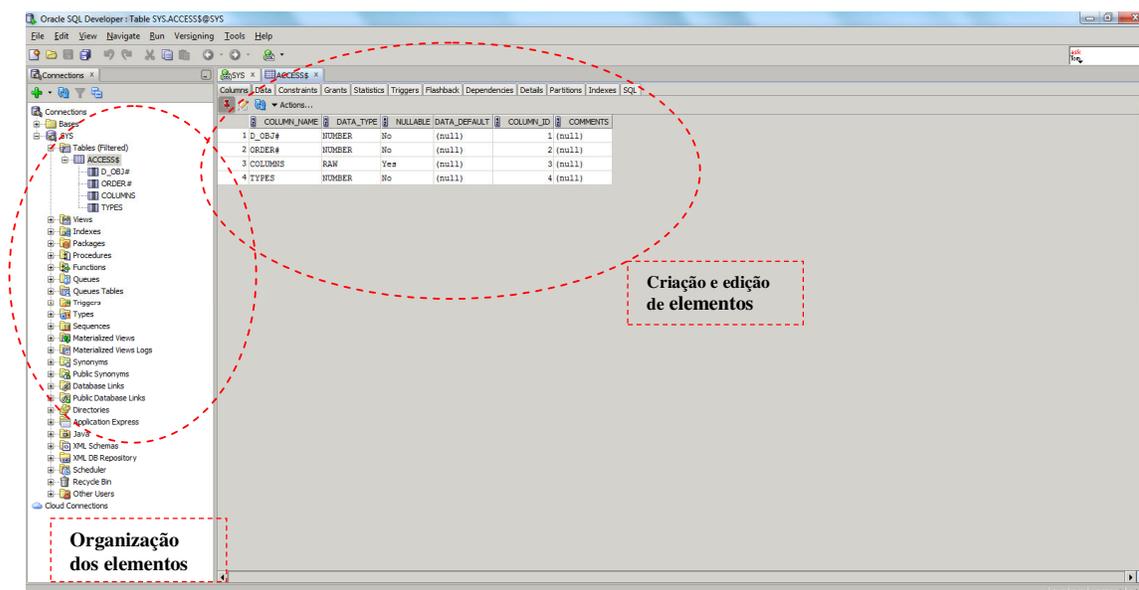


Figura 1 – Tela inicial do Sistema gerenciador de banco de dados Oracle SQL Developer

O sistema gerenciador de banco de dados Oracle SQL Developer está organizado em duas partes principais, além das barras de menus e de ferramentas que estão na parte superior da tela. Essas partes estão destacadas na Figura 3 e são:

a) Organização dos elementos – fornece uma visão geral dos bancos de dados criados e permite a seleção para edição de um banco de dados. O código de composição do banco de dados selecionado é apresentado na área de edição. Nessa área também é possível fazer alterações em banco de dados existentes, bem como criar tabelas de forma visual.

b) Criação e edição de elementos – essa área permite criar tabelas por meio de código *Structured Query Language (SQL)*, bem como inserir e editar dados, selecionar a visualização das tabelas e dos dados que estão armazenados nas tabelas.

Para adicionar uma nova conexão, basta clicar no botão '+' posicionado à esquerda e acima da organização dos elementos. Para isso é necessário informar os seguintes parâmetros:

- a) *Connection Name* – nome que será apresentado na lista de databases cadastrados.
- b) *UserName* – o nome do usuário (*schema*).
- c) *Password* – senha de acesso.
- d) *Hostname* – local em que o banco de dados encontra-se (*Internet Protocol - IP*).
- e) *Port* – porta em que o serviço Oracle está executando.
- f) *SID* – identificação única do banco de dados instalado, cada banco de dados tem o seu SID.

Os demais campos são configurações opcionais.

3.1.2 Delphi 7

A IDE Delphi 7 é uma ferramenta *Rapid Application Development* (RAD), desenvolvimento rápido de aplicações, e uma IDE podendo ser utilizado para o desenvolvimento de aplicações cliente/servidor e aplicações de uso genérico (SANTANA, 2014). Na Figura 2 está a tela principal da IDE Delphi 7.

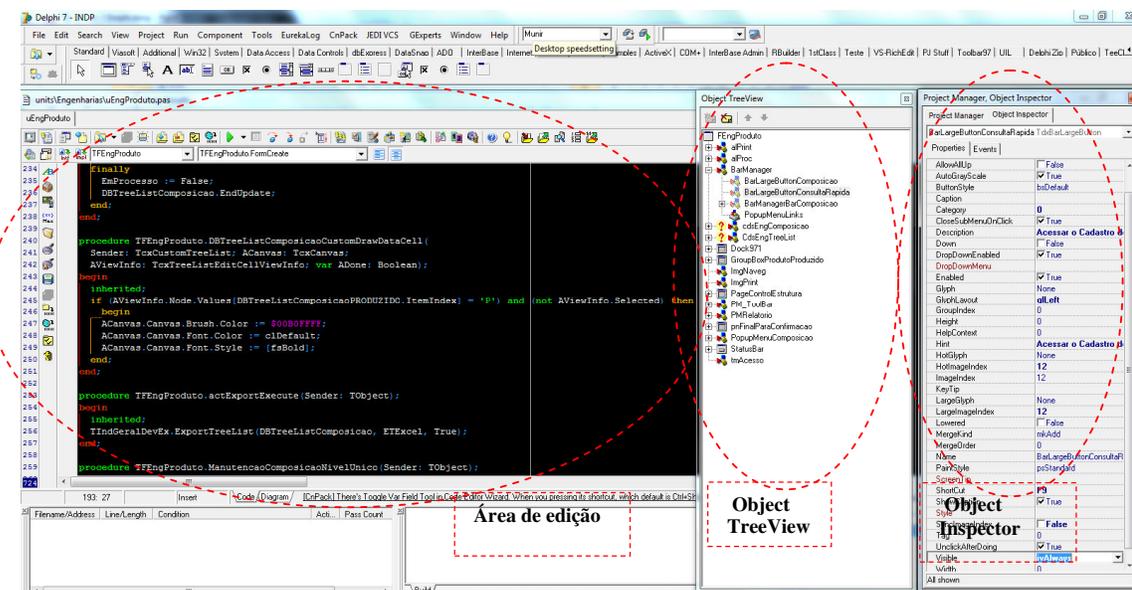


Figura 2 – IDE Delphi 7

As partes destacadas na Figura 3 indicam:

- Object TreeView* – a área na qual são organizados os elementos (arquivos) definidos durante a implementação do projeto. Esses elementos se referem ao código desenvolvido.
- Área de edição – nesta área o código é escrito e editado. Nesta área também pode ser composta a interface do sistema por meio de componentes visuais vinculados à ferramenta.
- Object Inspector* – apresenta as propriedades de cada componente visual do sistema.

3.1.3 Linguagem Delphi

Na versão 7 do Delphi algumas das tecnologias incluídas nas versões anteriores foram aperfeiçoadas e erros foram corrigidos (como por exemplo, no *Simple Object Access Protocol* (SOAP) e no DataSnap). A versão 7 também fornece suporte às tecnologias mais recentes

(como os temas do Windows XP e *Universal Description, Discovery and Integration* (UDDI)). Nessa versão, também é fornecido um compilador que não vise o processador Intel, mas sim a plataforma CIL do .NET. Delphi possui muitos recursos e trata-se de um ambiente complexo de programação que envolve muitos elementos.

3.1.4 Banco de Dados Oracle

O Banco de Dados Oracle Database 11g detém recordes mundiais em desempenho e escalabilidade em servidores com Windows, Linux e Unix. Permite migrar de um único servidor para *grid computing* sem necessidade de qualquer mudança em código (ORACLE, 2014).

O Banco de Dados Oracle 11g oferece melhores resultados por automatizar as tarefas administrativas e possui recursos de segurança. Sua funcionalidade *real application clusters* possibilita a mais alta disponibilidade dos dados possível (ORACLE, 2014).

3.1.5 DevExpress

A DevExpress VCL é um componente de terceiros que agrega mais de 180 componentes e bibliotecas otimizadas. Essas bibliotecas e componentes oferecem às aplicações funcionalidades do Microsoft Office e de interação com o usuário com base do Visual Studio ou Adobe Photoshop (DEVEXPRESS, 2014).

O componente *cxDBTreeList*, por exemplo, facilita a visualização de uma composição de um produto, sendo melhor que o *dbgrid* nativo do Delphi, no qual uma árvore de produção não seria possível representar.

3.2 MÉTODO

O processo de modelagem da rotina de engenharia de produto tem como base o modelo sequencial linear como descrito em Pressman (2008), complementado pelo processo unificado (KRUCHTEN, 2004). A modelagem e a implementação foram realizadas em etapas

(iterações). Inicialmente foram definidos os requisitos principais, obtendo-se uma visão geral da rotina. Complementos aos requisitos foram realizados à medida que as iterações ocorriam.

O processo unificado auxiliou a definir ciclos iterativos de modelagem e de implementação, que esteve centrada em funcionalidades que visaram facilitar e agilizar o desenvolvimento da rotina.

O Quadro 1 apresenta os processos (fluxos de trabalho) e as iterações desenvolvidas.

| Processos | 1ª iteração | 2ª iteração | 3ª iteração | 4ª iteração |
|-------------------|---|--|-------------------------------------|--|
| Requisitos | Visão geral da rotina. Definição dos requisitos básicos e essenciais | Refinamento dos requisitos | Avaliação dos requisitos | |
| Análise e projeto | Modelagem das principais tabelas do banco de dados | Ajustes de campos e definição de tabelas complementares | Revisão da modelagem | |
| Implementação | Base para a implementação do sistema: funções padrão | Interface do sistema e implementação da rotina de ordens de produção | Revisão do código visando melhorias | |
| Testes | Unitário | De modelo de interface | Unitário | De usuário, realizados por responsáveis por testes na empresa para a qual a rotina foi implementada. |

Quadro 1 – Iterações definidas

A seguir estão descritos os processos definidos para as atividades realizadas.

a) Requisitos

O levantamento dos requisitos iniciou com a necessidade do desenvolvimento de uma rotina para geração e acompanhamento de ordens de produção. O autor deste trabalho trabalhou em indústria, possuindo bom conhecimento das atividades envolvidas no processo de produção. E embora a rotina para geração e acompanhamento das ordens de produção tenha sido implementada para ser inicialmente utilizado em segmento industrial diferente do que o autor atuou, as rotinas de produção são semelhantes nos diversos segmentos industriais. Isso auxiliou no levantamento dos requisitos e entendimento das funcionalidades de negócio.

Além do conhecimento do autor, o levantamento dos requisitos foi realizado pela observação do processo em indústria. Essa rotina faz parte de um sistema que implementa diversas funcionalidades relacionadas à informatização do chão de fábrica e dos aspectos gerenciais de uma indústria. Assim, embora um segmento industrial tenha sido utilizado para

o levantamento de requisitos e implantação da primeira versão do sistema, o mesmo pode ser utilizado aplica a praticamente todo tipo de produção fabril.

Da necessidade de geração e acompanhamento das ordens de produção foram definidos os requisitos a serem implementados. Os requisitos foram listados como funcionais e não funcionais. Os requisitos foram complementados à medida que as iterações ocorriam. Varias alterações ocorreram como forma de definir mais adequadamente as funcionalidades do sistema visando atender os objetivos definidos para a rotina de gerenciamento de ordens de produção.

b) Análise e projeto do sistema

Com base nos requisitos foram definidas as tabelas do banco de dados com os seus atributos e relacionamentos.

c) Implementação

A implementação da rotina foi realizada utilizando a IDE Delphi 7. A implementação esteve centrada no desenvolvimento de uma rotina que representa o processo de geração de ordens de produção. Em termos de interface, o objetivo foi experimentar e testar a melhor forma de compor a tela e disponibilizar as informações na tela.

d) Testes

Os testes realizados foram estruturais (unitários e realizados pelo programador), estruturais para a verificação das funcionalidades, testes de interface e usabilidade para verificar a interação com o sistema. Os testes de interface, usabilidade e estruturais foram realizados pelos responsáveis pelos testes na empresa para a qual a rotina foi implementada.

4 RESULTADOS

Este capítulo apresenta o resultado obtido com o desenvolvimento deste trabalho. Inicialmente é apresentada uma descrição do sistema, que é uma rotina para geração de Ordens de Produção (OPs), em seguida é apresentada a modelagem dos requisitos definidos para essa rotina, o funcionamento da mesma (por meio de telas e explicação da interação) e a sua implementação. A apresentação da implementação ocorre por meio de listagens de códigos e respectivas explicações.

4.1 ESCOPO DO SISTEMA

A rotina de geração de ordens de produção desenvolvida tem a funcionalidade principal de fazer a interação com diversas funções e informações de dentro do sistema que automatiza uma indústria. Quando uma ordem é gerada, por meio da rotina implementada, é verificado o estoque das matérias-primas, realizada a reserva e a transferência das mesmas para o setor de produção. Além disso, é realizado o controle das etapas de produção por meio de um número identificador utilizado pela rotina. Esse identificador é denominado apontamento de produção.

Para um produto a ser produzido, na tela de programação de produção é informada a quantidade a ser produzida e o sistema gera as ordens de produção necessárias para que o produto final seja fabricado conforme as especificações do mesmo.

4.2 MODELAGEM DO SISTEMA

A listagem do Quadro 2 apresenta os requisitos funcionais identificados para a rotina implementada para gerenciamento de ordens de produção. Alguns dos requisitos são listados apenas para informação da funcionalidade. Eles não serão todos detalhados neste texto, pois eles não fazem parte do objetivo deste trabalho que é apresentar o funcionamento da visualização da engenharia de produto. No Quadro 2 a notação RF significa Requisito Funcional.

| Identificação | Nome | Descrição |
|----------------------|--|---|
| RF001 | Manter tipo da programação | Cadastrar um tipo de programação no sistema. Um tipo possui: ID, descrição, programação total. Excluir um tipo de programação do sistema. Editar um tipo de programação cadastrado, exceto o ID que é único. Consultar tipos de programação cadastrados. |
| RF002 | Manter programação de produção | Cadastrar programação de produção no sistema. A programação de produção é composta por tipo de programações. O cadastro da programação possui: ID, estabelecimento, tipo da programação, ID série da programação, data de criação, semana, mês, ano, usuário. Editar os campos da programação de produção exceto o código ID e o código ID série que são chave. Excluir uma programação de produção: ao excluir uma programação da produção devem ser excluídas também as ordens de produção que foram geradas, caso não estejam com o <i>status</i> de liberadas como “sim”. |
| RF003 | Manter itens da programação. | Cadastrar os itens da programação de produção no sistema. O item da programação possui: ID, Estabelecimento, ID da programação da produção, ID do tipo da programação, item, estrutura, quantidade, data de criação, observações, quantidade mínima. Editar: a quantidade a ser produzida e a observação somente se não houverem ordens de produção geradas. Excluir: Eliminar o item somente quando não houverem ordens de produção geradas. |
| RF004 | Manter ordens de produção. | Cadastrar as ordens de produção no sistema. A ordem de produção possui: ID, estabelecimento, ID da programação da produção, item, estrutura, nível da estrutura, quantidade à produzir, quantidade produzida, situação, data de criação, número de cópias. Editar: pode ser editada a quantidade a ser produzida e a estrutura, somente se a ordem não tenha sido liberada. |
| RF005 | Manter materiais das ordens de produção | Cadastrar os materiais das ordens de produção no sistema. O material da ordem de produção possui: ID, estabelecimento, ID da programação de produção, ID da composição do produto, item, estrutura, sequência da matéria prima, matéria prima, unidade de medida, qtde utilizada, tipo da matéria prima, grupo alternativo, sequência do grupo, porcentagem de quebra, data inclusão, data validade. Editar: podem ser editadas as matérias-primas e as quantidades, caso a ordem de produção não tenha sido liberada. |
| RF006 | Manter lotes dos materiais das ordens de produção. | Cadastrar os lotes dos materiais das ordens de produção. O lote do material da ordem de produção possui: ID, estabelecimento, ID dos materiais das ordens de produção, lote, Quantidade do lote. Editar: podem ser editados o lote e a quantidade, caso a ordem de produção não tenha sido liberada. |
| RF007 | Manter configurações da ordem de produção. | Configurações adicionais da ordem de produção gerada. A configuração da ordem de produção possui: ID, estabelecimento, produzido, inativo, tipo do número de lote, validade de dias, qtde programação mínima, tipo da explosão, |

| | | |
|-------|--|---|
| | | observações. As configurações são definidas pelo usuário ou por meio de cadastros pré-definidos. Esse cadastro está relacionado com a tabela de ordens de produção. |
| RF008 | Manter configurações dos materiais da ordem de produção. | Configurações adicionais dos materiais da ordem de produção gerada. A configuração dos materiais da ordem de produção possui: ID, estabelecimento, ID da ordem de produção, ID da composição, produzido, inativo, tipo do número de lote, validade de dias, quantidade programação mínima, tipo da explosão. As configurações são definidas pelo usuário ou por meio de cadastros pré-definidos. Está relacionada com a tabela de materiais da ordem de produção. |

Quadro 2 – Requisitos funcionais estabelecidos para o sistema

A listagem do Quadro 3 apresenta os requisitos não funcionais identificados para a rotina. Os requisitos não funcionais explicitam regras de negócio, restrições à rotina, por exemplo, requisitos de qualidade, desempenho, segurança e outros. No Quadro 3 a notação RNF significa Requisito Não Funcional.

| Identificação | Nome | Descrição |
|---------------|--|--|
| RNF001 | A programação de produção só pode ser criada se houver o cadastro do tipo da programação. | O tipo de programação deve ser criado com um ID único, uma descrição e a opção de programa total, que serve para que no momento da geração das ordens de produção possa ou não ser visualizado o estoque atual ou não. |
| RNF002 | Programação de produção deverá ter uma sequência chamada de série única para cada tipo de programação. | Ao gerar uma programação do tipo 1, a série deve iniciar em 1, se for criada o tipo 2, a série deve iniciar em 1 também. |
| RNF003 | Os campos de data de criação e usuário devem ser fixos com a data da real criação e usuário que criou a programação. Os campos de semana, mês, e ano podem ser alterados pelo usuário. | O campo de data de criação deve ser fixo para que seja vista a data de criação da programação, bem como o usuário e os demais campos para que o usuário possa programar sua produção para uma determinada semana ou mês. |
| RNF004 | Um item da programação só pode ser informado se o mesmo for do tipo produzido = sim e que contenha uma estrutura válida. | Ao digitar na tela um produto que não se ajuste à esses pré-requisitos, o sistema deve mostrar uma mensagem informando o usuário. |
| RNF005 | Ao informar um item para a programação, o sistema deve realizar validações como quantidade múltipla de produção e lote econômico. | O sistema deve informar uma mensagem perguntando se o usuário quer alterar a quantidade para o lote econômico de produção e apenas avisar que o múltiplo de produção do item deve de ser de X em Y, alterando automaticamente para o próximo múltiplo. |
| RNF006 | Ao gerar uma programação, cujos itens foram importados de um pedido de compra, deve-se colocar na observação do item da | Também será salvo em uma tabela auxiliar esse vínculo. A observação serve para uma melhor visualização da programação, bem como filtro para alguma pesquisa e relatório para a produção. |

| | | |
|--------|---|---|
| | programação essa informação. | |
| RNF007 | Ao gerar as ordens de produção, se um subproduto não tiver uma estrutura válida, deve-se continuar a geração sem a geração da ordem de produção do item em questão. | O sistema deve mostrar uma mensagem para o usuário que para o produto X não foram gerados as ordens de produção, pois o mesmo não apresenta uma estrutura válida. |
| RNF008 | Ao gerar a ordem de produção, a mesma deve assumir um número único de controle chamado de ordem de produção. | Esse número será usado para diversas outras funções dentro do sistema como apontamentos de produção, impressão de ordens de produção para o chão de fábrica e demais controles internos. |
| RNF009 | Os campos de quantidade a produzir e quantidade produzida devem ser alterados conforme sequência na produção. | O campo de quantidade a produzir deve ser alterado caso o usuário deseje alterar a quantidade a ser produzida durante o processo (diminuição da quantidade). A quantidade produzida deve ser atualizada sempre que o usuário fizer um apontamento de ordem de produção. |
| RNF010 | Ao reduzir a quantidade de uma ordem de produção, os materiais que deveriam ser utilizados para a produção devem ser alterados. | Nunca deve ser alterado para uma quantidade inferior do que já foi produzido. Se houver materiais empenhados para essa ordem de produção, devem ser retornados para o local de estoque de origem. |
| RNF011 | Geração de ordem de produção não verifica o saldo das matérias-primas no primeiro momento. | Primeiro o sistema informa que para produzir o item X serão necessárias as quantidades Y do item A001, Z do item A002, sem verificar estoque inicialmente. |
| RNF012 | Alteração da quantidade da matéria-prima utilizada após a geração da ordem de produção. | O usuário pode alterar a quantidade de uma matéria-prima para mais ou para menos após a geração da ordem de produção, contanto que a mesma não tenha sido liberada para a produção. |
| RNF013 | Verificar a necessidade de estoque das matérias-primas utilizadas para a produção. | Verifica cada ordem de produção e suas matérias-primas uma a uma, analisando se existe a matéria-prima necessária para a produção. Caso exista, o sistema deve realizar a reserva da mesma para evitar conflito com outra ordem de produção. |
| RNF014 | Verificar a necessidade de estoque dos subprodutos utilizados para a produção. Programação total = não. | Caso um subproduto tenha em estoque, a ordem de produção desse produto não será gerada, bem como os subprodutos desse subproduto. |
| RNF015 | Opção de programação total = sim. | Com essa opção “sim”, o sistema não verifica se existe em estoque os subprodutos nem o produto pai. Ele gera as ordens de produção para todos os produtos produzidos = sim. |
| RNF016 | Opção de programação total. | O produto pai sempre será produzido, independente de como a opção de programa total estiver. |
| RNF017 | Se alguma matéria-prima não possua estoque suficiente, não liberar a ordem de produção. | Avisar ao usuário qual matéria-prima não possui a quantidade necessária para a produção, para que o usuário tome alguma providência. |

Quadro 3 – Requisitos não funcionais estabelecidos para o sistema

A Figura 3 apresenta o diagrama de entidades e relacionamentos do banco de dados. Essa Figura contém as tabelas (descritas a seguir) e os relacionamentos entre as mesmas.

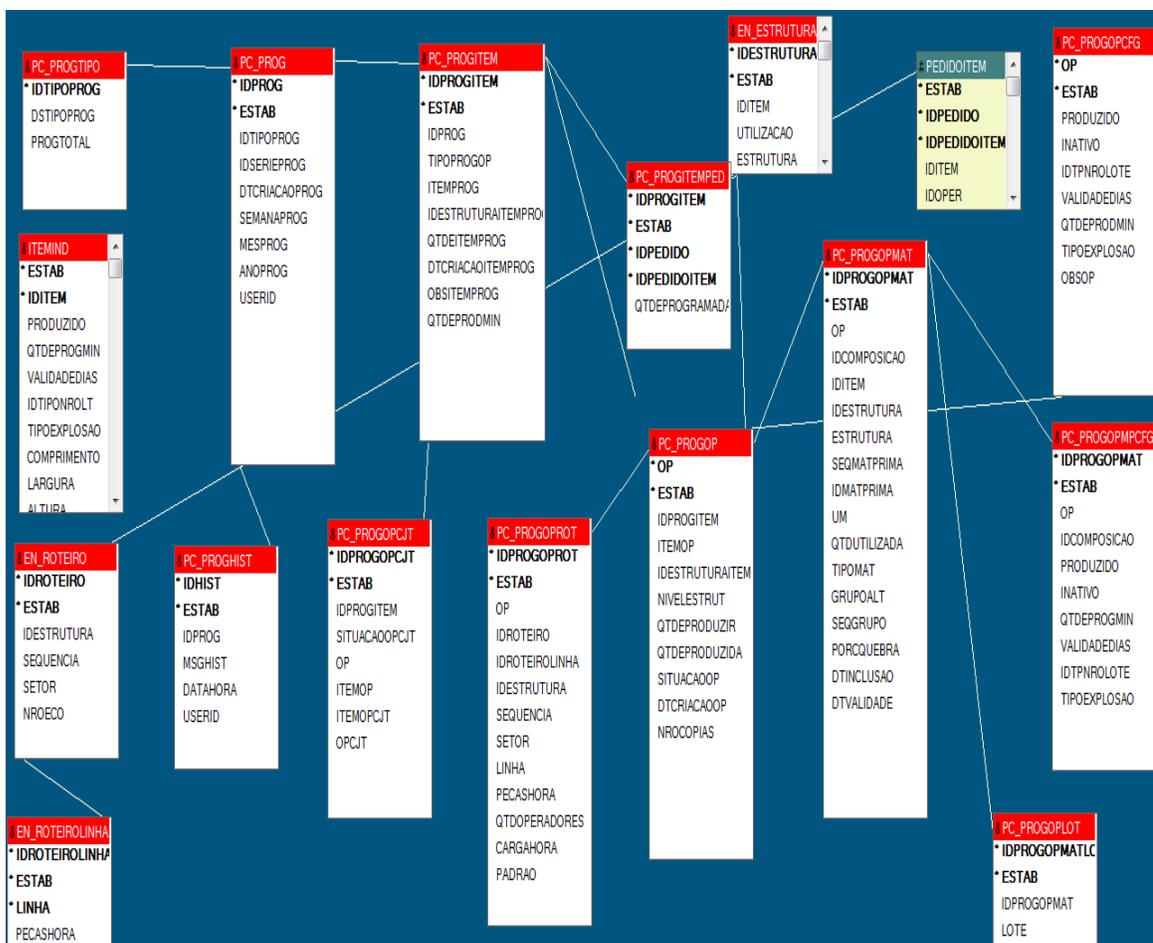


Figura 3 – Diagrama de entidades e relacionamentos

A tabela PC_PROGTIPO (Quadro 4) armazena os tipos da programação, que definirá algumas especificações da programação.

| Campo | Tipo | Nulo | PK | FK | Obs. |
|--------------|-------------|-------------|-----------|-----------|----------------------|
| IDTIPOPROG | Inteiro | Não | Sim | Sim | FK da tabela PC_PROG |
| DSTIPOPROG | String | Sim | Não | Não | |
| PROGTOTAL | String | Sim | Não | Não | |

Quadro 4 – Campos da tabela PC_PROGTIPO

No Quadro 5 estão os campos da tabela PC_PROG que armazena os dados do cabeçalho da programação.

| Campo | Tipo | Nulo | PK | FK | Obs. |
|---------------|-------------|-------------|-----------|-----------|----------------------------------|
| IDPROG | Inteiro | Não | Sim | Sim | FK com PC_PROGITEM e PC_PROGHIST |
| ESTAB | Inteiro | Não | Sim | Sim | |
| IDTIPOPROG | Inteiro | Não | Não | Sim | FK com PC_PROGTIPO |
| IDSERIEPROG | Inteiro | Não | Não | Não | |
| DTCRIACAOPROG | DateTime | Não | Não | Não | |
| SEMANAPROG | Inteiro | Sim | Não | Não | |
| MESPROG | Inteiro | Sim | Não | Não | |
| ANOPROG | Inteiro | Sim | Não | Não | |
| USERID | String | Não | Não | Sim | FK com PUusuario |

Quadro 5 – Campos da tabela PC_PROG

A tabela PC_PROGITEM cujos campos estão apresentados no Quadro 6 contém os itens da programação, os quais serão produzidos.

| Campo | Tipo | Nulo | PK | FK | Obs. |
|---------------------|-------------|-------------|-----------|-----------|----------------------------------|
| IDPROGITEM | Inteiro | Não | Sim | Sim | FK com PC_PROGOP, PC_PROGITEMPED |
| ESTAB | Inteiro | Não | Sim | Sim | FK com PC_PROGOP, PC_PROGITEMPED |
| IDPROG | Inteiro | Não | Não | Não | FK com PC_PROG |
| TIPOPROGOP | Inteiro | Não | Não | Não | |
| ITEMPROG | String | Não | Não | Não | |
| IDESTRUTURAITEMPROG | Inteiro | Sim | Não | Não | |
| QTDEITEMPROG | Double | Não | Não | Não | |
| DTCRIACAOITEMPROG | DateTime | Não | Não | Não | |
| OBSITEMPROG | String | Sim | Não | Não | |
| QTDEPRODMIN | Double | Sim | Não | Não | |

Quadro 6 – Campos da tabela PC_PROG

A tabela PC_PROGOP, que está representada no Quadro 7, contém as ordens de produção que foram geradas a partir dos itens da programação.

| Campo | Tipo | Nulo | PK | FK | Obs. |
|-------------------|-------------|-------------|-----------|-----------|--------------------|
| OP | Inteiro | Não | Sim | Sim | FK com PC_PROGITEM |
| ESTAB | Inteiro | Não | Sim | Sim | FK com PC_PROGITEM |
| IDPROGITEM | Inteiro | Não | Não | Não | |
| ITEMOP | String | Não | Não | Não | |
| IDESTRUTURAITEMOP | Inteiro | Não | Não | Não | |
| NIVELISTRUT | Inteiro | Sim | Não | Não | |
| QTDEPRODUIR | Double | Não | Não | Não | |
| QTDEPRODUIZA | Double | Sim | Não | Não | |

| | | | | | |
|-------------|-----------|-----|-----|-----|--|
| SITUACAOOP | Inteiro | Não | Não | Não | |
| DTCRIACAOOP | Date Time | Sim | Não | Não | |
| NROCOPIAS | Inteiro | Sim | Não | Não | |

Quadro 7 – Campos da tabela PC_PROGOP

No Quadro 8 estão os campos da tabela PC_PROGOPROT utilizada para armazenar o roteiro de produção que cada item deve passar para ser produzido.

| Campo | Tipo | Nulo | PK | FK | Obs. |
|----------------|-------------|-------------|-----------|-----------|------------------|
| IDPROGOPROT | Inteiro | Não | Sim | Não | |
| ESTAB | Inteiro | Não | Sim | Sim | FK com PC_PROGOP |
| OP | Inteiro | Sim | Não | Não | FK com PC_PROGOP |
| IDROTEIRO | Inteiro | Sim | Não | Não | |
| IDROTEIROLINHA | Inteiro | Sim | Não | Não | |
| IDESTRUTURA | Inteiro | Sim | Não | Não | |
| SEQUENCIA | Inteiro | Sim | Não | Não | |
| SETOR | Inteiro | Sim | Não | Não | |
| LINHA | String | Sim | Não | Não | |
| PECASHORA | Double | Sim | Não | Não | |
| QTDOPERADORES | Inteiro | Sim | Não | Não | |
| CARGAHORA | Double | Sim | Não | Não | |
| PADRAO | String | Sim | Não | Não | |

Quadro 8 – Campos da tabela PC_PROGOPROT

A tabela PC_PROGOPMAT armazena as matérias-primas utilizadas para produzir as ordens de produção geradas na tabela PC_PROGOP. Os campos da tabela PC_PROGOPMAT estão apresentados no Quadro 9.

| Campo | Tipo | Nulo | PK | FK | Obs. |
|--------------|-------------|-------------|-----------|-----------|---------------------|
| IDPROGOPMAT | Inteiro | Não | Sim | Sim | FK com PC_PROGOPLOT |
| ESTAB | Inteiro | Não | Sim | Sim | FK com PC_PROGOPLOT |
| OP | Inteiro | Sim | Não | Sim | FK com PC_PROGOP |
| IDCOMPOSICAO | Inteiro | Sim | Não | Não | |
| IDITEM | String | Sim | Não | Não | |
| IDESTRUTURA | Inteiro | Sim | Não | Não | |
| ESTRUTURA | Inteiro | Sim | Não | Não | |
| SEQMATPRIMA | Inteiro | Sim | Não | Não | |
| IDMATPRIMA | String | Sim | Não | Não | |
| UM | String | Sim | Não | Não | |
| QTDUTILIZADA | Double | Sim | Não | Não | |
| TIPOMAT | String | Sim | Não | Não | |
| GRUPOALT | String | Sim | Não | Não | |
| SEQGRUPO | Inteiro | Sim | Não | Não | |
| PORCQUEBRA | Double | Sim | Não | Não | |
| DTINCLUSAO | Date Time | Sim | Não | Não | |
| DTVALIDADE | Date Time | Sim | Não | Não | |

Quadro 9 – Campos da tabela PC_PROGOPMAT

O Quadro 10 contém os campos da PC_PROGOPCFG que armazena as configurações de cada ordem de produção.

| Campo | Tipo | Nulo | PK | FK | Obs |
|--------------|-------------|-------------|-----------|-----------|------------------|
| OP | Inteiro | Não | Sim | Sim | FK com PC_PROGOP |
| ESTAB | Inteiro | Não | Sim | Sim | FK com PC_PROGOP |
| PRODUZIDO | String | Sim | Não | Não | |
| INATIVO | String | Sim | Não | Não | |
| IDTPNROLOTE | Inteiro | Sim | Não | Não | |
| VALIDADEDIAS | Inteiro | Sim | Não | Não | |
| QTDEPRODMIN | Double | Sim | Não | Não | |
| TIPOEXPLOSAO | Inteiro | Sim | Não | Não | |
| OBSOP | String | Sim | Não | Não | |

Quadro 10 – Campos da tabela PC_PROGOPCFG

A tabela PC_PROGOPMPCFG contém as configurações das matérias-primas que fazem parte das ordens de produção. Os campos dessa tabela são apresentados no Quadro 11.

| Campo | Tipo | Nulo | PK | FK | Obs |
|--------------|-------------|-------------|-----------|-----------|-------------------|
| IDPROGOPMAT | Inteiro | Não | Sim | Sim | FK com PC_PROGMAT |
| ESTAB | Inteiro | Não | Sim | Sim | FK com PC_PROGMAT |
| OP | Inteiro | Sim | Não | Não | |
| IDCOMPOSICAO | Inteiro | Sim | Não | Não | |
| PRODUZIDO | String | Sim | Não | Não | |
| INATIVO | String | Sim | Não | Não | |
| QTDEPROGMIN | Double | Sim | Não | Não | |
| VALIDADEDIAS | Inteiro | Sim | Não | Não | |
| IDTPNROLOTE | Inteiro | Sim | Não | Não | |
| TIPOEXPLOSAO | Inteiro | Sim | Não | Não | |

Quadro 11 – Campos da tabela PC_PROGOPMPCFG

4.3 APRESENTAÇÃO DO SISTEMA

A parte principal da rotina de geração de ordens de produção é feita na tela de programação da produção. Após realizar o cadastro do tipo da programação, ou por meio do menu “P.P.C.P / Tipo de Programação” ou dentro da própria tela de programação, no campo de tipo da programação pressionando a tecla F4, são preenchidos os campos necessários como apresentados na Figura 4.

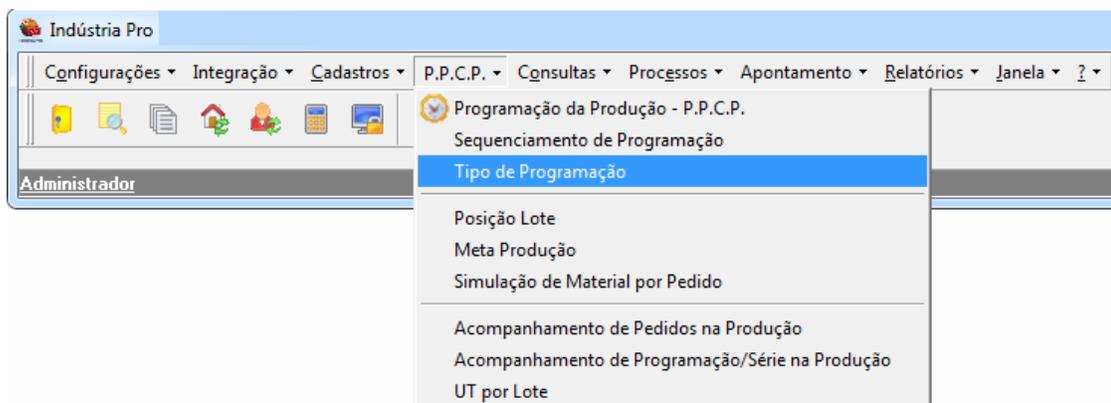


Figura 4 – Menu acesso Tipo de Programação

A Figura 5 apresenta o formulário de tipo de programação.

The screenshot displays the 'Tipo de Programação' form within the 'Indústria Pro' application. The form is titled 'Tipo de Programação' and includes a toolbar with various icons. The form is divided into two tabs: 'Informações Normais' and 'Informações Personalizadas'. The 'Informações Normais' tab is active, showing the following fields:

- Tipo de Programação: 1
- Descrição: PRODUÇÃO DE EIXOS
- Controle da Programação: P - Padrão
- Programação Total: Sim
- Gerar empenho das OPs?: Sim
- Gerar Cód. Lote de Estoque: Não
- Opções da Programação: 10 - Agrupar OPs de Componentes Iguais - Todos os Itens
- Tipo de Nro. Cavaletes: 20 - Numeração Individual por Cavalete
- Sigla no Cód. Lote: Não
- Cadastro de Lote Apont.: Sim

Figura 5 – Formulário de tipo de programação

No formulário apresentado na Figura 5. Campos:

a) Empresa: vem preenchido automaticamente pelo sistema, ou seja, empresa selecionada para o trabalho. Esse dado é apresentado na barra de título do formulário. Na Figura 5 o nome da empresa não é visível porque a tela foi recortada de maneira a ficar mais claramente visível.

b) Tipo da Programação: previamente cadastrado para a utilização da mesma.

c) Série: campo sequencial por tipo de programação.

d) Prioridade: campo que o usuário pode definir uma prioridade para produção (campo informativo).

e) Cor: o cliente pode definir uma cor para imprimir as ordens de produção para controle visual durante o processo de fabricação.

f) Mês, Ano e Data prevista produção: são obtidos automaticamente do sistema, mas o usuário pode alterar para o mês que deseja programar.

Após isso, é possível incluir o primeiro produto a se produzido, clicando no botão Novo Item (Figura 6 a e b). A Figura 6 (partes a e b) refere-se à tela para programação da produção. A apresentação da tela foi segmenta para facilitar a visualização.

Indústria Pro

Configurações | Integração | Cadastros | P.P.C.P. | Consultas | Processos | Apontamento | Relatórios | Janela

Administrador

Programação da Produção

Empresa: 1 | Tipo de Programação: PRODUÇÃO DE EIXOS | Série: 1 | Prioridade: 650 | Cor (D.F.): BRANCO | Mês: 9 - Setembro | Ano: 2.014 | Data de Criação: 24/09/2014 | Data Pre: 24/09/2014

Novo Item | Excluir Item | Liberar Programação | Pedidos em Carteira

| Seq. | Item no Lote | Qtde. a Prod. | Item | Descrição | Referência | Estrutura |
|------|--------------|---------------|--|-----------|--------------|-----------|
| 1 | 1.00000 | 5001.000685 | EIXO EXPANSIVO CORPO EM AÇO 3". GARRAS INTERCALADAS EM AÇO | | GF1110102076 | 1 |

Detalhes

Gerar OPs | Nova OP | Excluir OP | Re-Gerar Empenho Material | Alterar Empenho | Alterar Fluxo de Produção

Ordens de Produção | Pedidos de Venda | Agrupamentos das Ordens de Produção

| Nível | Liberada | Nro. OP | Qtde. a Produzir | Componente | Descrição Componente | Referência | Estrutura | Nro Roteiro |
|------------------------------------|----------|---------|------------------|------------|----------------------|------------|-----------|-------------|
| Clique aqui para definir um filtro | | | | | | | | |

< Nenhuma OP Gerada/Encontrada >

(a)

(b)

Figura 6 - Novo Item no lote de produção

O usuário deverá informar um produto que esteja configurado como Produzido = Sim, caso contrário o sistema mostrará uma mensagem de que o produto é inválido. Caso o produto seja válido, o sistema preencherá automaticamente alguns campos como descrição e estrutura padrão.

Após informar o produto, deve ser informada a quantidade a ser produzida. Se o produto possuir quantidade múltipla cadastrada, o sistema mostrará uma mensagem e alterará a quantidade para o próximo múltiplo seguinte. Caso o produto tenha lote econômico cadastrado, será mostrada uma tela para o usuário tomar uma decisão: se quer que a quantidade a ser produzida fique do modo como ele informou ou que seja alterado para a

quantidade do lote econômico. A configuração de múltiplo sobrepõe a configuração de lote econômico.

Depois de ser informado o produto e a quantidade a ser produzida, é possível gerar as ordens de produções clicando no botão Gerar OPs. É nesse momento que o sistema realizará todos os cálculos de estoque e necessidades para gerar as ordens de produção, que será descrito na implementação do sistema.

O sistema mostrará uma mensagem caso algum subproduto cadastrado dentro do produto pai não tenha estrutura ou roteiro de produção para produzir o mesmo, para que o usuário tome as devidas providências posteriormente e gere as ordens de produção corretamente.

Feito a geração das ordens de produção, o usuário deve liberar as mesmas para a produção, clicando no botão Liberar Programação. É nesse momento que o sistema gerará o empenho e as movimentações de locais de estoque para as matérias-primas que serão necessárias para montar o produto e seus subprodutos que o usuário informou. Não é possível liberar um produto para a produção caso este não tenha o saldo disponível necessário para a produção, o usuário pode realizar alterações no produto e tornar a gerar a ordem de produção novamente. A Figura 7 apresenta as ordens de produção geradas.

The screenshot displays the 'Programação da Produção' interface. At the top, there are filters for 'Empresa', 'Tipo de Programação', 'Série', 'Prioridade', 'Cor (O.F.)', 'Mês', 'Ano', 'Data de Criação', 'Data Prevista Produção', and 'User ID'. The main table lists production orders with the following data:

| Seq. Item no Lote | Item | Descrição | Referência | Estrutura | Disc. Estrutura | Status | Opções | Tipo Nº Cavalete | Observações |
|-------------------|---------------------|--|--------------|-----------|-----------------|---------------------|--|------------------|-------------|
| 1 | 1.00000.5001.000665 | EKID EXPANSIVO CORPO EM AÇO 2". GARRAS INTERCALADAS EM AÇO | GF1110102076 | 1 | PADRÃO | 20 - Fichas Geradas | 10 - Agrupar OPs de Componentes Iguais - Todos os Itens 20 - Numeração Individual por Cavalete | | |

The 'Detalhes' section shows a table with columns for 'Nível', 'Librada', 'Nro. OP', 'Qtde. a Produzir', 'Componente', 'Descrição', 'Componente', 'Referência', 'Estrutura', 'Nro. Roteiro', 'Local Destino', and 'Compens'. The 'Consumo Real' section includes a table with columns for 'Empenho', 'Fluxo de Produção', 'Configurações da OP', and 'Informações Personalizadas da OP'.

Figura 7 – Ordens de produção geradas

Após a liberação da ordem de produção, o usuário poderá realizar a produção e o apontamento da ordem para o consumo das matérias-primas e a posterior geração do saldo do produto produzido, que não será demonstrado nesse trabalho.

4.4 IMPLEMENTAÇÃO DO SISTEMA

A implementação do sistema inicia-se pela criação de um formulário padrão utilizado pela empresa, que é usado em juntamente com uma *engine* que faz a manutenção de uma tabela cabeçalho por meio do dicionário de dados gravado no banco de dados. Com isso, é possível realizar a manipulação de diversas tabelas utilizadas no processo por meio do termo `NestedDataSet(Mestre/Detalhe)`.

Utilizando uma função criada pela empresa para realizar as conexões entre o cliente e o servidor/banco de dados é chamada a função que criará as *queries* no servidor de modo que os `ClientsDataSet` no cliente fiquem relacionadas como mestre/detalhe. A Listagem 1 contém o código dessa função.

```
{ Configura o NestedDataSet para o cdsProgDet }
  if dmConexao.CallDynMethod(_rpcDB_GetNestedDataSet, [VarArrayOf([
    'QrPC_ProgItem' { Query Detalhe },
    'PC_PROGITEM' { Tabela Detalhe },
    '*' { Campos Detalhes },
    'IDPROG;ESTAB' { Relacionamento },
    'DsPC_Prog' { DataSource Mestre},
    'QrPC_Prog' { Query Mestre },
    '' { Chave Mestre },
```

Listagem 1 – Função para geração de queries no servidor

Essa função vincula oito tabelas que são utilizadas para a programação da produção. Porém, esse trecho do código mostrará apenas a primeira parte para entendimento de como é o funcionamento.

Após a criação do formulário, que está vinculado à *engine*, ao ser chamado o mesmo abrirá uma tela inicial na qual o usuário poderá informar alguns filtros para melhor visualização, ou abrir todos os registros contidos naquela tabela. Isso é realizado por meio de uma consulta no banco de dados. O parâmetro necessário para essa consulta é o estabelecimento selecionado no momento. A Figura 8 apresenta o formulário para filtragem de cabeçalho de lote.

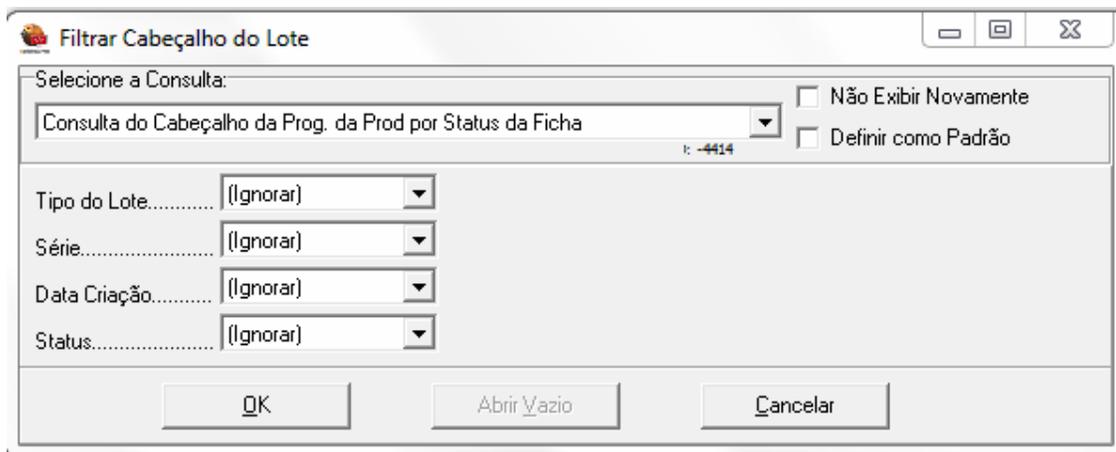


Figura 8 – Filtro do cabeçalho do lote

A tela é aberta com o filtro desejado, mostrando as informações dentro de um `cxGridDBTableView`. Nessa tela o usuário poderá acessar o registro que ele desejar clicando duas vezes sobre o registro desejado. A Figura 9 apresenta a listagem de registros.

| Estab. | Tipo lote | DESCRICAO | Série | Cor lote | Mês | Ano | Data Criação | Data Produção | Prioridade |
|--------|-----------|-----------|-------|----------|-----|-------|--------------|---------------|------------|
| 999 | 1 | Padrão | 1 | BRANCO | 5 | 2.014 | 23/05/2014 | 23/05/2014 | 1 |
| 999 | 1 | Padrão | 2 | BRANCO | 5 | 2.014 | 26/05/2014 | 26/05/2014 | 2 |
| 999 | 1 | Padrão | 3 | BRANCO | 5 | 2.014 | 28/05/2014 | 28/05/2014 | 3 |
| 999 | 1 | Padrão | 4 | BRANCO | 6 | 2.014 | 26/06/2014 | 26/06/2014 | 4 |
| 999 | 1 | Padrão | 5 | BRANCO | 6 | 2.014 | 26/06/2014 | 26/06/2014 | 5 |

Figura 9 – Filtro das programações

A tela que abrirá após o usuário escolher o registro desejado ou criar uma nova programação contém os campos para serem informados todos os parâmetros do produto que deseja ser produzido.

O formulário que abrirá na sequência terá diversos campos que devem ser preenchidos e cada campo contém a sua validação não permitindo que o usuário digite alguma informação inválida. Essas validações são feitas no evento dos componentes chamados `OnChange` e `OnValidate`. Cada campo tem a sua validação específica, com a mensagem de aviso para cada situação.

No `create` da `engine` é realizada a criação do `DataModule` utilizado na aplicação, bem como a ativação de alguns eventos padrões utilizados. A Listagem 2 apresenta o código desse evento.

```

inherited Create(AOwner);
  ClasseDoForm := TFManProgramacaoProducao;
  TablePrincipal := 'PC_PROG';
  FdMProgramacaoProducao := TDmProgramacaoProducaoC.Create(NIL);
  FdMProgramacaoProducao.OnChangePadrao := DoCustomChangePadrao;
  { Metodos para PCP_ProgCab }
  CdsEdPrinc.BeforeOpen :=
FdMProgramacaoProducao.CdsPC_ProgBeforeOpen;
  CdsEdPrinc.AfterOpen :=
FdMProgramacaoProducao.CdsPC_ProgAfterOpen;
  OnAfterCreateForm := DoAfterCreateForm;
  OnAfterCreateBrowse := DoAfterCreateBrowse;

```

Listagem 2 – Função create da engine

Além do *create* da Engine há o evento *DoAfterCreateForm*. Nesse evento os *ClientDataSets* que estão no *DataModulo* são vinculados aos *DataSources* que estão no formulário. Eles controlarão as informações juntamente com os *cxGrids*.

Nesse momento também é realizada a verificação se o *DataModule* foi criado, evitando novos problemas.

```

with TFManProgramacaoProducao(Form) do
  begin
    if (not Assigned(Self.FdMProgramacaoProducao)) then
      raise EAvviso.CreateCausa('Não Foi Possível Carregar a
Programação!',
        'O Data Module (DmProgramacaoProducao) Não Foi
Instanciado.',
        'Por Favor, Feche a Rotina e Tente Novamente...');
    Self.FdMProgramacaoProducao.FHandle := Handle;
    FdMProgramacaoProducao := Self.FdMProgramacaoProducao;

    { Configura nos DataSource o Respectivo DataSet }
    DsProgItem.DataSet := FdMProgramacaoProducao.CdsProgItem;
    DsOrdemProd.DataSet := FdMProgramacaoProducao.CdsOrdemProd;
    DsOrdemProdCfg.DataSet :=
FdMProgramacaoProducao.CdsOrdemProdCfg;
    DsOrdemProdMat.DataSet :=
FdMProgramacaoProducao.CdsOrdemProdMat;

```

```

DsOrdemProdRot.DataSet                               :=
FDmProgramacaoProducao.CdsOrdemProdRot;
end;

```

Listagem 3 – Carga da programação de ordem de produção

Após a criação e atribuição dos métodos pode ser iniciada a inclusão dos itens a ser produzidos no lote de produção, por meio do botão Novo Item.

A inclusão do novo item ocorre pela inserção de um registro no ClientDataSet da tabela PC_PROG. Nesse momento ocorrem validações relacionadas aos campos mais importantes (Exemplo: o item que precisa de validação se é produzido = Sim ou se existe estrutura valida para o mesmo, entre outras.).

Feita a inserção do item a ser produzido, com a quantidade a ser produzido preenchida, e as configurações de agrupamento devidamente escolhidas, podem ser geradas as ordens de produção referentes ao produto selecionado. No exemplo apresentado será produzido um eixo de uma máquina pneumática. Não serão apresentados detalhes da composição do produto para manter sigilo.

A geração de ordens de produção consiste em várias etapas de um processo que carrega na memória as informações desejadas por meio de consultas criadas no banco de dados e a manipulação das informações utilizando ClientsDataSets. Primeiramente é carregada a estrutura do produto desejado em memória, buscando uma consulta pré-definida no dicionário de consultas, passando os parâmetros esperados pela mesma (estabelecimento, item, estrutura, quantidade a ser produzida). Essa consulta vem por padrão filtrada para mostrar apenas os subprodutos que são do tipo Produzido = Sim. Dessa forma são geradas as ordens de produção apenas para os produtos é que necessário produzir e não para os comprados ou que estão em estoque (caso a configuração de produção total estiver = não).

Nesse momento de geração das ordens de produção é feita a verificação da configuração selecionada no cabeçalho para agrupamento de itens iguais, com isso, caso exista um mesmo subproduto que precise ser produzido dentro do item principal, a rotina agrupará os produtos iguais e suas quantidades, para o processo se tornar mais fácil e rápido de controlar. Se a configuração estiver para não agrupar os itens iguais, então é gerada uma ordem de produção para cada item.

Exemplificando: isso significa que a empresa gasta 10 minutos para fazer o “*setup*” de uma máquina para produzir um item, de quantidade 10. Se não forem agrupados os itens iguais, haveria duas ordens de produção para o mesmo item, fazendo com que a empresa

precise programar duas ordens de produção e realize o *setup* duas vezes para o mesmo item, perdendo assim 10 minutos de um tempo que poderia ser empregado em outra ordem de produção.

No momento da geração das ordens de produção, outra rotina chamada é a geração do empenho das matérias-primas utilizadas na produção desses itens em questão. Para isso, a rotina em questão verifica o estoque disponível de cada matéria-prima e a quantidade necessária para a produção. Se o estoque é suficiente, a rotina realiza uma transação em uma tabela de controle do estoque, que retira o saldo disponível da matéria-prima e o coloca no saldo empenhado. A movimentação física não ocorre, apenas o *status* do saldo necessário para a produção é atualizado.

Na Figura 10, é possível ver a quantidade prevista e a quantidade reservada, bem como o seu local de consumo. A quantidade prevista é a quantidade necessária para produzir a ordem de produção em questão (8855). A quantidade reserva é a quantidade que o sistema colocou com *status* de empenhado. O local de consumo é o padrão do item cadastrado nas configurações do item, isso serve para uma empresa que tenha vários locais de estoque e em cada local de estoque ele armazena um tipo de material.

| Empenho | | Fluxo de Produção | | Pacotes de Produção | | Número de Série | | Consumo Real | | Configurações da OP | | Informações Personalizadas da OP | |
|------------------------------------|------|-------------------|----------------|---------------------|-----------|-----------------|---------------|----------------|---------------------------------------|---------------------|----------------|----------------------------------|--|
| Nro. | Seq. | U.M. | Qtde. Prevista | Qtde. Reserva | U.M. Peça | Qtde Peças | Local Consumo | Material | Descrição Material | Data Prevista | Custo Unitário | Custo Total | |
| Clique aqui para definir um filtro | | | | | | | | | | | | | |
| 8855 | 1 | PC | 1,00000 | 1,00000 | PC | 1,00000 | | 1 9009.000001 | ENG - CORPO EM AÇO CÂMERA ÚNICA 75x50 | 30/09/2014 | 13,38350 | 13,38350 | |
| 8855 | 2 | PC | 1,00000 | 1,00000 | PC | 1,00000 | | 1 9011.000001 | ENG - PONTEIRA LADO DO BICO | 30/09/2014 | 0,00000 | 0,00000 | |
| 8855 | 3 | PC | 1,00000 | 1,00000 | PC | 1,00000 | | 1 9011.000002 | ENG - PONTEIRA LADO OPOSTO DO BICO | 30/09/2014 | 0,00000 | 0,00000 | |
| 8855 | 4 | UN | 1,00000 | 1,00000 | UN | 1,00000 | | 1 9009.000004 | ENG - KIT DE MONTAGEM C.Ú. 75x50 | 30/09/2014 | 56,63050 | 56,63050 | |
| 8855 | 5 | KG | 2,11600 | 2,11600 | M | 2,11600 | | 87 6019.000762 | TUBETE 76,2 X 5MM | 30/09/2014 | 2,53000 | 5,35348 | |

Figura 10 – Empenho das matérias-primas por ordem de produção

Caso a quantidade reserva seja menor que a quantidade prevista, isso quer dizer que não havia saldo suficiente da matéria-prima disponível para a produção e com isso, no momento de liberar a ordem para a produção, dependendo das configurações do sistema, a mesma não poderá ser liberada para a produção. Uma ordem de produção pode ser liberada mesmo não havendo quantidade suficiente em estoque dos materiais utilizados para produzi-la. Existem empresas que não controlam saldo de matéria-prima, então mesmo o sistema informando que o saldo está negativo, ela tem a matéria-prima em estoque físico.

Toda a matéria-prima pode ser alterada durante a geração da ordem de produção, ou seja, caso uma determinada matéria-prima não tenha em estoque e exista alguma matéria-prima Y alternativa em estoque, que contenha as mesmas características de fabricação, pode ser feita alteração do mesmo por meio do botão alterar empenho.

Outra rotina que é chamada durante a geração de ordens de produção é a que gera o fluxo de produção (Figura 10). O fluxo de produção é o roteiro que o produto deve seguir para ser produzido. Um exemplo para isso seria a montagem de uma caneta. Nesse processo pode-se montar a tampa antes de inserir o tubo com a tinta no bulbo. Para que a montagem ocorra da forma correta é necessário seguir um fluxo de produção corretamente. No momento da geração do fluxo de produção, também é possível alterar a ordem do roteiro e os processos que o mesmo passará, como foi descrito com as matérias-primas.

| Empenho | Fluxo de Produção | Pacotes de Produção | Número de Série | Consumo Real | Configurações da OP | | |
|------------------------------------|-------------------|---------------------|-----------------|-------------------------|---------------------|------------|-------------|
| Nro. OP | Seq. | Posto OP | Operação | Descrição da Operação | Peças/Hora | Tempo Peça | Observações |
| Clique aqui para definir um filtro | | | | | | | |
| 8855 | 1 | 200 | 200 | MONTAR CONFORME DESENHO | 2,94118 | 0,34000 | a |
| 8855 | 2 | 201 | 201 | VERIFICAR MEDIDAS | 11,76471 | 0,08500 | a |
| 8855 | 3 | 201 | 230 | TESTAR VAZAMENTO | 11,76471 | 0,08500 | a |
| 8855 | 4 | 202 | 8 | EMBALAR | 11,76471 | 0,08500 | a |

Figura 10 – Fluxo de produção

Até esse momento não salvamos nenhuma informação no banco de dados, a não ser o cabeçalho do produto que será produzido. Quando realizado o processo de salvar será incluso nas tabelas necessárias todas as informações que estão na memória, inclusive as alterações que por ventura o usuário realizou previamente. O usuário pode salvar no banco de dados a cada alteração ou geração de ordem de produção que realizar. Porém, se o usuário desejar apenas visualizar as informações para alguma tomada de decisão, o mesmo pode fazer tudo em memória e realizar as alterações que desejar em seu “teste”.

No momento em que as ordens de produção são salvas no banco de dados, também são fixadas as *flags* nas matérias-primas que serão empenhadas para a produção dos itens em questão, com isso o saldo fica reservado, mas não empenhado.

Para a liberação das ordens de produção geradas, existem algumas configurações gerais que o sistema considera nesse momento, mas vamos nos atentar para um dos mais importantes, que é o tipo da liberação, que são:

a) **Por item da programação:** são agrupadas todas as matérias-primas de todas as ordens de produção geradas do item da programação e verifica o estoque de todas elas como se fossem de uma ordem de produção somente. Esse controle serve para empresas que precisam produzir todas as ordens de produção de uma só vez, independente se existe estoque suficiente para as mesmas ou não. A Figura 11 apresenta a liberação por lote.

| Programação..... | | PRODUÇÃO DE EIXOS | | 1 | | | | | | | |
|--|--|-------------------|-----------|---------------|---------------------------------------|--------------|-------------|-------------|----------|--------------------------|----|
| Série..... | | 650 | | | | | | | | | |
| Local Est. Destino..... | | | | | | | | | | | |
| <input type="checkbox"/> Liberar Todos os Itens | | | | | | | | | | | |
| Arraste o título de uma coluna aqui para agrupar por ela | | | | | | | | | | | |
| Etab | Tipo Lote | Série | Seq. Item | Descrição | Referência | Qtde | Status | Negativos | Liberado | | |
| | 1 | 1 | 650 | 1 5001.000665 | EIXO EXPANSIVO CORPO EM AÇO 3"; GARRA | GF1110102076 | 1,20 | | 10 | <input type="checkbox"/> | |
| Material | Descrição | Qtde | U.M. | Custo | Custo Total | Físico | Reserva Tot | Saldo | Local | Origen | Re |
| 3024.000010 | PLACA DE IDENTIFICACAO POLICARBONATO | 1,00000 | PC | R\$ 1,00 | R\$ 1,00 | -147,69100 | 29,00000 | -148,69100 | | 28 | |
| 3017.000180 | MOLA VALVULA DE AR 1/8"BSP | 1,00000 | UN | R\$ 0,16 | R\$ 0,16 | 1984,00000 | 2,00000 | 1983,00000 | | 18 | |
| 3016.002070 | ANEL ORING 4,40X1,8MM | 0,01000 | CT | R\$ 38,00 | R\$ 0,38 | 48,36000 | 0,03000 | 48,35000 | | 18 | |
| 3016.003412 | ANEL ORING 8,00X2,00MM | 0,01000 | CT | R\$ 38,00 | R\$ 0,38 | 38,64000 | 0,02000 | 38,63000 | | 18 | |
| 1003.001802 | PINO INTERNO DA VALVULA 1/8" | 1,00000 | PC | R\$ 0,12 | R\$ 0,12 | 172,00000 | 2,00000 | 171,00000 | | 18 | |
| 1003.001800 | CORPO DA VALVULA DE AR 1/8" BSP | 1,00000 | PC | R\$ 0,32 | R\$ 0,32 | 272,00000 | 2,00000 | 271,00000 | | 18 | |
| 6019.000360 | TUBETE 36X4MM | 0,01600 | M | R\$ 0,65 | R\$ 0,01 | 111,85326 | 0,01600 | 111,83726 | | 24 | |
| 1003.001803 | CHAPEU DA VALVULA 1/8" | 1,00000 | PC | R\$ 0,25 | R\$ 0,25 | 43,00000 | 2,00000 | 42,00000 | | 18 | |
| 4001.001626 | CÂMARA DE AR EM BORRACHA 16X26MM | 1,76000 | M | R\$ 23,75 | R\$ 41,80 | 742,91786 | 1,76000 | 741,15786 | | 72 | |
| 2004.005397 | BR SAE 8620 LAMINADO 2"1/8 53,97MM | 1,14000 | M | R\$ 0,00 | R\$ 0,00 | 737,14930 | 1,14000 | 736,00930 | | 87 | |
| 6019.000762 | TUBETE 76,2X5MM | 2,11600 | KG | R\$ 2,53 | R\$ 5,35 | -560,58700 | 2,11600 | -562,70300 | | 87 | |
| 2005.002222 | BR SAE 1045 LAMINADO 7/8" 22,22MM | 0,08600 | M | R\$ 0,00 | R\$ 0,00 | 25,44000 | 0,08600 | 25,35400 | | 87 | |
| 2003.007550 | TM ST 52.3 LAMINADO 75X50MM | 1,88500 | M | R\$ 7,10 | R\$ 13,38 | 449,22700 | 5,18000 | 447,34200 | | 87 | |
| 2005.003175 | BR SAE 1045 LAMINADO 1"1/4 31,75MM | 0,04500 | M | R\$ 0,00 | R\$ 0,00 | 57,86000 | 0,04500 | 57,81500 | | 87 | |
| 5000.000022 | VALVULA DE AR 1/8" BSP | 1,00000 | PC | R\$ 1,60 | R\$ 1,60 | -3070,14800 | 40,00000 | -3071,14800 | | 1 GA | |
| 3017.000005 | MOLA CONICA EIXO CÂMARA ÚNICA | 3,00000 | UN | R\$ 22,30 | R\$ 6,69 | 2382,67928 | 1,26000 | 2382,37928 | | 5 | |
| 9011.000001 | ENG - PONTEIRA LADO DO BICO | 1,00000 | PC | R\$ 0,00 | R\$ 0,00 | 8,00000 | 11,00000 | 7,00000 | | 1 | |
| 9011.000002 | ENG - PONTEIRA LADO OPOSTO DO BICO | 1,00000 | PC | R\$ 0,00 | R\$ 0,00 | 13,00000 | 11,00000 | 12,00000 | | 1 | |
| 9009.000004 | ENG - KIT DE MONTAGEM C.Ú. 75X50 | 1,00000 | UN | R\$ 56,63 | R\$ 56,63 | -1,00000 | 11,00000 | -2,00000 | | 1 | |
| 9009.000001 | ENG - CORPO EM AÇO CÂMARA ÚNICA 75X50 | 1,00000 | PC | R\$ 13,38 | R\$ 13,38 | -1,00000 | 12,00000 | -2,00000 | | 1 | |
| 9013.000002 | ENG - GARRA DE AÇO ALTA 1036 COM MOLAS | 15,00000 | CJ | R\$ 0,45 | R\$ 6,69 | -465,00000 | 213,00000 | -480,00000 | | 1 | |
| 4006.001036 | GARRA MICROFUNDIDA ALTA ERA1036 | 15,00000 | PC | R\$ 0,00 | R\$ 0,00 | 4183,00000 | 63,00000 | 4168,00000 | | 4 | |

Figura 11 – Liberação por lote

b) **Por ordem de produção:** A verificação do saldo das matérias-primas é realizada por ordens de produção e caso uma matéria-prima não tenha saldo suficiente para realizar a produção, a ordens de produção não é liberada (Figura 12).

| Programação..... | | PRODUÇÃO DE EIXOS | | 1 | | | | |
|--|-------|-------------------|---------------|----------------------------|---|-----------------|---------------|-------------------------------------|
| Série..... | | 650 | | | | | | |
| Local Est. Destino..... | | | | | | | | |
| <input type="checkbox"/> Liberar Todos os Itens com Estoque Disponível | | | | | | | | |
| Tipo Lote | Série | Seq. Item | Item | Referência | Descrição Item | Qtd Produzir | Negativos | Liberado? |
| | 1 | 650 | 1 5001.000665 | GF1110102076 | EIXO EXPANSIVO CORPO EM AÇO 3"; GARRAS INTERCALADAS | 1,00000 | 10 | <input type="checkbox"/> |
| NroFicha | Ni | Item da Ficha | Referência | Descrição | Qtde | Data de Criação | Qtd Negativos | Liberada? |
| Clique aqui para definir um filtro | | | | | | | | |
| + | 8860 | 3 1001.162603 | | COPO DA VEDACAO CÂMARA | 1,00000 | 07/10/2014 | 0 | <input checked="" type="checkbox"/> |
| + | 8861 | 3 1001.162601 | | ESPIGAD LB VEDACAO CÂMARA | 1,00000 | 07/10/2014 | 0 | <input checked="" type="checkbox"/> |
| + | 8862 | 3 1001.162602 | | ESPIGAD LOB VEDACAO CÂMARA | 1,00000 | 07/10/2014 | 0 | <input checked="" type="checkbox"/> |
| + | 8863 | 3 1001.000001 | | PARAFUSO FURADO CÂMARA | 1,00000 | 07/10/2014 | 1 | <input type="checkbox"/> |
| + | 8865 | 3 4006.001036 | | GARRA MICROFUNDIDA ALTA | 15,00000 | 07/10/2014 | 0 | <input checked="" type="checkbox"/> |
| + | 8867 | 3 1003.001800 | | CORPO DA VALVULA DE AR | 1,00000 | 07/10/2014 | 0 | <input checked="" type="checkbox"/> |
| + | 8868 | 3 1003.001802 | | PINO INTERNO DA VALVULA | 1,00000 | 07/10/2014 | 0 | <input checked="" type="checkbox"/> |
| + | 8869 | 3 1003.001803 | | CHAPEU DA VALVULA 1/8" | 1,00000 | 07/10/2014 | 0 | <input checked="" type="checkbox"/> |
| + | 8864 | 2 9013.000002 | | ENG - GARRA DE AÇO ALTA | 15,00000 | 07/10/2014 | 0 | <input checked="" type="checkbox"/> |
| + | 8866 | 2 5000.000022 | GA 10 0014 | VALVULA DE AR 1/8" BSP | 1,00000 | 07/10/2014 | 0 | <input checked="" type="checkbox"/> |
| + | 8856 | 1 9009.000001 | | ENG - CORPO EM AÇO CÂMARA | 1,00000 | 07/10/2014 | 0 | <input checked="" type="checkbox"/> |
| + | 8857 | 1 9011.000001 | | ENG - PONTEIRA LADO DO B | 1,00000 | 07/10/2014 | 0 | <input checked="" type="checkbox"/> |
| + | 8858 | 1 9011.000002 | | ENG - PONTEIRA LADO OPOS | 1,00000 | 07/10/2014 | 0 | <input checked="" type="checkbox"/> |
| + | 8859 | 1 9009.000004 | | ENG - KIT DE MONTAGEM C.Ú | 1,00000 | 07/10/2014 | 6 | <input type="checkbox"/> |
| + | 8855 | 0 5001.000665 | GF1110102076 | EIXO EXPANSIVO CORPO EM | 1,00000 | 07/10/2014 | 3 | <input type="checkbox"/> |

Figura 12 – Liberação por ordem de produção

Na Figura 11, alguns itens aparecem na cor vermelha, isso quer dizer que não há saldo suficiente das matérias-primas em questão para a produção do lote. E, assim, o *flag* de “liberado” está desmarcado e não poderá ser habilitado porque não há matéria-prima na quantidade necessária.

Na Figura 12, algumas linhas estão em cor vermelha, isso quer dizer que existem matérias-primas com saldo insuficiente para atender as ordens de produção. Assim, algumas delas o *flag* de “liberado” está desmarcado.

Ao salvar a rotina de liberação das ordens de produção, então é feita a transferência das matérias-primas do seu local de estoque de origem para o local de estoque de destino, que geralmente é a produção, onde os mesmos serão consumidos através das rotinas de apontamento de ordens de produção do sistema.

5 CONCLUSÃO

O objetivo deste trabalho foi implementar uma rotina de ordem de produção para um sistema de gerenciamento de produção industrial. O módulo do sistema, denominado rotina, foi implementado como planejando.

A realização desse projeto possibilitou a utilização do conhecimento adquirido trabalhando em uma indústria durante seis anos e isso fez com que o desenvolvimento das rotinas fosse realizado com maior cuidado e prevendo diversas possibilidades de melhorias que uma empresa poderia ter ao implantar o sistema. O desafio foi grande pela inexperiência, ao iniciar o projeto, decorrente da falta de conhecimento em programação. Contudo, foi uma oportunidade de aprendizado em termos de programação, de desenvolvimento de projetos e de entendimento de processos industriais.

O desenvolvimento deste projeto foi uma oportunidade de trabalhar com os componentes da DevExpress, que acelera muito o processo de criação de telas. Um aspecto relevante no desenvolvimento do projeto é em relação às manutenções. Isso porque os formulários, as funções e as *procedures* criadas pela área de tecnologia da empresa puderam ser facilmente reaproveitados. Com elas não houve a necessidade de preocupar-se em como é realizada uma consulta no banco de dados ou feita a conexão, validação de licenças de usuários, entre outras funcionalidades. Essas funcionalidades já estavam prontas.

A linguagem Delphi oferece muitos recursos e funcionalidades. Utilizando os seus componentes corretamente e as tecnologias adequadas é possível desenvolver com mais agilidade. Os recursos da linguagem permitem, por exemplo, dividir o processamento entre a máquina cliente e o servidor. Além disso, a linguagem permite compilar para Android e iOS. Uma das desvantagens dessa linguagem é o seu ambiente de desenvolvimento que possui valor de aquisição bastante alto.

REFERÊNCIAS

- CARLOS, Karen Laiza S. P; CARLOS, Tiago Paveli H. **Engenharia do produto**. 2011. Disponível em: <<http://engproducaoufersa.blogspot.com.br/p/engenharia-do-produto.html>>. Acesso em: 13 mai. 2014.
- CHENG, Lin C.; MELO FILHO, Leonel. D. R. **QFD: Desdobramento da função qualidade na gestão de desenvolvimento de produtos**. São Paulo: Editora Blucher, 2007.
- CHIAVENATO, Idalberto. **Gestão de pessoas**. Rio de Janeiro: Elsevier, 2005.
- CORREIA, Filipe Miguel A. C.; O'NEILL, Henrique. **Avaliação da automatização de processos de negócio em serviços partilhados**. In: 8º Congresso Nacional da Administração Pública. Desafios e Soluções. Instituto Nacional de Administração. P. 370-393, 2011.
- DEVEXPRESS. **VCL Controls**. Disponível em: <<https://www.devexpress.com/Products/VCL/>> . Acesso em: 04 abr. 2014.
- KRISHNAN, Vish; ULRICH, Karl T. **Product development decisions: a review of the literature**. Management Science, 2001.
- KRUCHTEN, Philippe. **Introdução ao RUP: rational unified process**. 2. ed. Rio de Janeiro, RJ: Ciência Moderna, 2004.
- ORACLE. **Banco de Dados Oracle 11g**. Disponível em: <<http://www.oracle.com/br/solutions/midsized/oracle-products/database/index.html>>. Acesso em: 03 abr. 2014.
- PAPAZOGLU, Michael P. **Web services: principles and technology**. England: Pearson Education, 2008.
- PRESSMAN, Roger. **Engenharia de software**, 5 ed. 2002. Rio de Janeiro: McGrawHill.
- SANTANA, Gladiston. **O que é o Delphi e qual a sua diferença quando comparado a outros RADs**. Disponível em: <<http://www.htmlstaff.org/ver.php?id=13>>. Acesso em: 03 abr. 2014.
- SCHEER, August-Wilhelm; ABOLHASSAN, Ferri; JOST, Wolfram; KIRCHMER, Mathias, **Business Process Automation, ARIS in Practice**; Springer; New York, 2004.
- TOLEDO, José Carlos. **Gestão do processo de desenvolvimento de produto**. 2000.
- WFMC TC-1011. **Workflow management coalition. Terminology and glossary**. Document Number WFMC-TC-1011, version 2.0, 1996.
- ZWICKER, Ronaldo; SOUZA, Cesar Alexandre de; VIDAL, Antonio Geraldo da Rocha; SIQUEIRA, José de Oliveira. **Grau de Informatização de Empresas: um Modelo Estrutural**

Aplicado ao Setor Industrial do Estado de São Paulo. **RAE – Eletrônica**, v. 6, n. 2, jul./dez., p. 1-27, 2007.