

CENTRO UNIVERSITÁRIO FEEVALE

RAMON MARCELO RECH

MODELAGEM E IMPLEMENTAÇÃO DE UM SIMULADOR DE
MICROTERMINAL COLLETER PARA O SIGER

Novo Hamburgo, novembro de 2009.

RAMON MARCELO RECH

MODELAGEM E IMPLEMENTAÇÃO DE UM SIMULADOR DE
MICROTERMINAL COLLETER PARA O SIGER

Centro Universitário Feevale
Instituto de Ciências Exatas e Tecnológicas
Curso de Ciência da Computação
Trabalho de Conclusão de Curso

Professor Orientador: Roberto Affonso Schilling

Novo Hamburgo, novembro de 2009.

AGRADECIMENTOS

Gostaria de agradecer a todos os que, de alguma maneira, contribuíram para a realização desse trabalho de conclusão, em especial:

Aos amigos e às pessoas que convivem comigo diariamente, minha gratidão, pelo apoio emocional nos períodos mais difíceis do trabalho. Pela empresa na qual trabalho há muitos anos, me dando todo o apoio do qual necessito.

Um grande abraço a todos os professores que me ajudaram nessa conquista.

RESUMO

O processo de coleta de dados em chão de fábrica é vital para a tomada de decisões e manter a empresa competitiva no mercado. Quando surgem falhas entre *software* e *hardware*, ou mesmo havendo necessidade de implementar novas funcionalidades, é necessário ter o equipamento de coleta de dados para efetuar o teste de *software*. Sem o equipamento, é praticamente impossível efetuar um teste completo no *software*, o que pode comprometer a rotina do usuário. Este processo de testar o *software* com o equipamento gera um custo maior para o desenvolvedor. Caso seja necessário utilizado o coletor de dados do usuário, esse fica com um ponto de coleta comprometido, atrapalhando seus indicadores de produção. Sendo assim, este trabalho tem como objetivo desenvolver um *software* capaz de efetuar todas as funcionalidades de um microterminal, em forma de simulador. Neste trabalho, o equipamento selecionado para desenvolver esse simulador é o TED-1000 da empresa Colleter, em funcionamento com o ERP desenvolvido pela Rech Informática Ltda., o SIGER – Sistema de gestão empresarial Rech.

Palavras-chave: Coleta de dados. ERP. Simulador. *Software*. Testes de *software*.

ABSTRACT

The process of collecting data at the factory floor level is vital to make decisions and keep the company competitive in the market. When failures occur between software and hardware, or even in the need to implement new features, it's necessary to have the equipment to collect data to test the software. Without the equipment, it is virtually impossible to make a thorough test in the software, which can compromise the user's routine. This process of testing the software with the data collectors generates a greater cost to the developer. If it is necessary to use an user's data collector, he may have troubles in his operation, disturbing the indicators of his production. Thus, this work aims to develop a software capable of performing all the features of a microterminal, as a simulator. In this work, the equipment selected for developing this simulator is the TED-1000 from the company COLLET, running with the ERP developed by Rech Informática Ltda., the SIGER, Rech Business Management System.

Keywords: Data collection. ERP. Simulator. Software. Testing software.

LISTA DE FIGURAS

Figura 1-1 - Princípios adotados pela Rech Informática Ltda.....	16
Figura 1-2 - Fatores críticos de sucesso I.....	17
Figura 1-3 - Fatores críticos de sucesso II.....	18
Figura 1-4 – SIGER e seus módulos atuais.....	24
Figura 2-1 – Ciclo de Coleta de dados chão-de-fábrica.....	27
Figura 3-1 – Microterminais TED 1000.....	33
Figura 3-2 – Vista traseira do TED 1000.....	34
Figura 3-3 – Topologia de rede utilizada pela Colleter.....	35
Figura 5-1 – TEDCOM1.....	46
Figura 5-2 – Simulador de microterminal.....	47
Figura 5-3 - Monitor de apontamento por microterminal.....	48
Figura 5-4 - Monitor de coletores.....	49
Figura 5-5 - Configuração dos Coletores.....	49
Figura 5-6 - Tela de leitura de barras (Microterminal).....	50
Figura 5-7 - Diagrama da classe: Identificadores de produção.....	50
Figura 5-8 - Diagrama da classe: Código das máquinas.....	51
Figura 5-9 - Diagrama da classe: Funções microterminal.....	51
Figura 5-10 - Diagrama da classe: Motivo da parada.....	52
Figura 5-11 - Diagrama da classe: Motivo do refugio.....	52
Figura 5-12 - Diagrama da classe: Talão.....	53
Figura 5-13 - Códigos das máquinas.....	54
Figura 5-14 - Ficha de funções do microterminal.....	55
Figura 5-15 - Ficha de motivo de refugio.....	55
Figura 5-16 - Ficha de motivo de parada.....	56
Figura 5-17 - Talão de produção – SSCC.....	57

Figura 5-18 - Monitor de microterminal – status	57
Figura 5-19 - Apontamento por microterminal - status SIGER	58
Figura 5-20 - Monitor de microterminal – Status.....	58
Figura 5-21 - Monitor de microterminal – Status.....	59
Figura 5-22 - Monitor de microterminal – Status.....	59
Figura 5-23 - Monitor de microterminal – Status.....	59
Figura 5-24 - Monitor de microterminal – Status.....	60
Figura 5-25 - Monitor de microterminal – Status.....	60
Figura 5-26 - Monitor de microterminal – Status.....	61
Figura 5-27 - Monitor de microterminal – Status.....	61
Figura 5-28 - Monitor de microterminal – Status.....	62
Figura 5-29 - Monitor de microterminal – Status.....	62

LISTA DE TABELAS

Tabela 1 – Esquema de conexão do microterminal ao <i>hub</i>	47
--	----

LISTA DE ABREVIATURAS E SIGLAS

ABNT	Associação Brasileira de Normas Técnicas
CEP	Controle Estatístico de Processo
CCD	Charged Coupled Device
CLP	Controlador Lógico Programável
DLL	Dynamic link library
EANCOM	Versão do EDIFACT simplificada e voltada para comércio de bens e serviços
ECF	Emissor de Cupom Fiscal
EDI	Electronic Data Interchange
EDIFACT	EDI for Administration, Commerce and Transport
EED	Encontro de Encaminhamento da Demanda
ERP	Enterprise Resource Planning
GPD	Gestão de Pesquisa e Desenvolvimento
GRM	Gestão de Relacionamento com Mercado
OP	Ordem de Produção
PCP	Planejamento e Controle de Produção
PDV	Ponto de Venda
MRP I	Material Requirements Planning
MRP II	Manufacturing Resources Planning
RNC	Regiastro de Não Conformidade
SICLA	Sistema Interno de Controle de Ligações e Atendimentos
SIGE	Sistemas Integrados de Gestão Empresarial
SIGER	Sistema Integrado de Gestão Empresarial Rech
SSCC	Serial Shipping Container Code

SUMÁRIO

INTRODUÇÃO	12
1 CONHECENDO MAIS SOBRE A RECH INFORMÁTICA	15
1.1 A Rech Informática	15
1.2 Princípios da Rech Informática	16
1.3 Metodologia de treinamento	18
1.3.1 Metodologia de treinamento – GPD	18
1.3.2 Metodologia de treinamento – Relacionamento com o Mercado	19
1.4 Metodologia de trabalho	20
1.5 Metodologia de testes de software	21
1.6 SIGER	23
2 SIGER - SISTEMA DE GESTÃO EMPRESARIAL RECH	25
2.1 Surgimento do SIGER	25
2.2 A Estrutura de módulos	25
2.3 Gestão Industrial	26
2.3.1 Funcionalidade da coleta de dados	27
2.3.2 A necessidade de um simulador de coleta de dados	29
3 COLETA DE DADOS EM CHÃO DE FÁBRICA	30
3.1 Microcomputadores	30
3.2 Características	30
3.3 Considerações sobre uso de microcomputadores	31
3.4 Microterminais TED 1000 e TED 5000	32
3.4.1 Características do hardware	32
3.4.2 Infra-estrutura do TED 1000	34
3.4.3 Considerações sobre o microterminal Colleter	36
3.4.4 Homologação do microterminal	36
4 SOFTWARE DE SIMULAÇÃO DE UM MICROTERMINAL	38
4.1 Análise de chão de fábrica	38
4.2 Simulador	41
4.2.1 Protocolo de comunicação	42
5 SOLUÇÃO DESENVOLVIDA	44
5.1 Proposta de desenvolvimento	44
5.1.1 Ferramentas de desenvolvimento	45
5.1.2 Banco de dados	45
5.1.3 Protocolo de comunicação	45

5.2 Modelagem _____	48
5.3 Desenvolvimento _____	50
5.4 Testes de software _____	53
5.4.1 Fichas com códigos de barras _____	54
5.4.2 Apontamento normal _____	56
5.4.3 Apontamento de parada _____	58
5.4.4 Apontamento com refugio _____	60
5.5 Considerações _____	62
6 RESULTADOS OBTIDOS _____	64
6.1 Facilidade nas implementações _____	64
6.2 Ações não atendidas _____	65
CONCLUSÃO _____	66
REFERÊNCIAS BIBLIOGRÁFICAS _____	68
ANEXO A – FLUXO DE PRODUÇÃO DE ELEMENTOS - SIGER _____	70
ANEXO B – FLUXO DE PRODUÇÃO DE COMPONENTES / CONJUNTOS - SIGER _____	72

INTRODUÇÃO

As empresas desenvolvedoras de *software* ERP (*Enterprise Resource Planning*) ou SIGE (Sistemas Integrados de Gestão Empresarial, no Brasil) integram todos os dados e processos de uma organização em um único sistema (LAUDON, 2004). Ocorre que nem todo o *software* é livre de erros. A importância da incorporação de atividades de teste a cada etapa do processo de desenvolvimento de *software* é fortalecida também pelo fato de que quanto mais próximo de sua origem os erros forem detectados, menor será o custo e a dificuldade da correção (WEBER; ROCHA; NASCIMENTO, 2001). Mesmo assim, muitas organizações entendem que o teste é um processo feito após a conclusão, e não como parte do processo de desenvolvimento.

Num mundo onde a competitividade cresce a cada dia, para empresas sobreviverem precisam adaptar seus processos a novas realidades, dentre elas destaca-se a automação. Conforme Moraes (2001, p. 15) “Hoje se entende por automação qualquer sistema, apoiado em computadores, que substitua o trabalho humano e que vise a soluções rápidas e econômicas para atingir os complexos objetivos das indústrias e dos serviços.”

O controle do processo produtivo é responsável pelos apontamentos/leituras dos dados obtidos no chão de fábrica, permitindo que o fluxo de dados seja transmitido do chão de fábrica aos sistemas gerenciais (SOUZA, 2005, p.III) de forma ágil, fácil e consistente. Esta forma de integração possibilita aos níveis gerenciais tomar decisões em tempo hábil.

O fato é que esse processo de coleta de dados não pode sofrer interrupções, do contrário, a informação que chegaria à gerência da indústria não serviria mais como base de apoio para a tomada de decisão. Sendo assim, surge uma preocupação no desenvolvimento de novas funções exercidas entre o software ERP e o equipamento de coleta de dados. Para garantir o funcionamento do software sem falhas é necessário efetuar testes com o uso do

equipamento, mas muitas vezes a empresa que está utilizando o equipamento de coleta não possui um backup (outro equipamento no caso de um ficar indisponível) ou até mesmo um contrato de manutenção com a empresa que lhe vendeu o equipamento, contrato esse que mantém um plano de manutenção preventiva e substituição/reposição de outro periférico no caso de falha. O mesmo ocorre quando o software ERP apresenta alguma falha, que pode ser de configuração, customização ou mesmo falha de programação. Segundo Pressman, 2005, um software tem êxito quanto atende às necessidades e expectativas de seus usuários. A obtenção de software de qualidade requer disciplina, justificando assim a necessidade de um enfoque de engenharia.

Exemplo do acima citado ocorre na Rech Informática Ltda, empresa que desenvolve o sistema de PDV (Ponto de Venda), contando com o suporte técnico via internet e telefone de empresas que fabricam o emissor de cupom fiscal (ECF). Essas contam com uma área especializada no desenvolvimento de softwares e drivers para facilitar o uso e implementação de seus equipamentos por parte dos desenvolvedores. É o caso da Bematech, uma empresa fundada em 1990, em Curitiba, Paraná. Trata-se de um case de sucesso, pois se consegue simular todas as transações da impressora através do software, uma forma ágil e prática para solucionar problemas e implementar novas funcionalidades.

Usando essa idéia de *software* de simulação de periféricos, surge a oportunidade de criar outra aplicação: desenvolver um *software* simulador de um coletor de dados, no caso em questão, o modelo TED-1000 Microterminal tipo mesa com 16 caracteres no *display*, da empresa Colleter. Fundada em 1989, instalada na região de Santo Amaro - São Paulo - capital, a Colleter tem desde o início como meta oferecer produtos e serviços para o segmento de automação, especificamente para sistemas de captura automática de dados, através das tecnologias de código de barras e rede de coletores instalados *on-line* com aplicativos instalados em microcomputadores. Contudo, a empresa Colleter não oferece qualquer *software* de simulação das funcionalidades de seu equipamento.

Sendo assim, surge a oportunidade de estudar as vantagens de desenvolver um software para simular as funcionalidades do micro terminal Colleter TED-1000, o qual foi homologado pela Rech Informática Ltda e está em uso pelos seus clientes. A idéia é usar esse modelo de microterminal como *case* e, obtendo bons resultados com o uso do software, expandir o mesmo para outros modelos de coletores de dado.

Como objetivo específico pretende-se analisar as funcionalidades atuais do SIGER na parte de coleta de dados de chão-de-fábrica simultaneamente com o microterminal Colleter TED 1000. Para demonstrar os estudos, o trabalho divide-se em cinco capítulos.

O capítulo um corresponde à apresentação da empresa Rech Informática, o uso de suas tecnologias, metodologia de treinamento de novos colaboradores, sua forma de trabalho e relacionamento com o mercado.

O capítulo dois se refere ao produto de software desenvolvido pela equipe da Rech, o SIGER, onde é dado enfoque ao módulo de Gestão Industrial, mais diretamente nas funcionalidades de coleta de dados com o uso do microterminal Colleter TED 1000.

O capítulo três se refere aos processos de coleta de dados em chão-de-fábrica, apresentando características, funcionamento, vantagens e desvantagens.

No capítulo quatro, o microterminal colleter TED 1000 será apresentado, juntamente com a o procedimento de homologação de equipamentos a serem utilizados junto ao SIGER. Ainda neste capítulo, alguns problemas de comunicação entre SIGER x Colleter serão apresentados, juntamente com a idéia do desenvolvimento de um software que simule algumas das funcionalidades do periférico, baseado na idéia já utilizada pela Rech, que é o software de simulação Bematech (ECF).

O capítulo cinco refere-se ao estudo específico da coleta de dados em chão-de-fábrica, relatando situações comuns em apontamentos, falhas e a proposta de solução de desenvolvimento do software de simulação do microterminal.

Por último, conclusões sobre o desenvolvimento do projeto realizado, no qual se pretende desenvolver uma solução para não atrapalhar os processos do cliente da Rech, melhorando o tempo de resposta na solução de problemas e novas implementações, reduzindo custos e aumentando o ganho em satisfação.

1 CONHECENDO MAIS SOBRE A RECH INFORMÁTICA

A Rech Informática Ltda, abriu a oportunidade de modelar e implementar um software de simulação de um microterminal coletor de dados, mais especificamente para o modelo TED 1000 da Colleter, empresa fabricante do equipamento. A idéia é construir o software com as tecnologias utilizadas pela empresa, bem como sua metodologia de trabalho. A intenção é desenvolver uma ferramenta que sirva como apoio a solução de problemas e possibilitar a implementação de novos recursos para o *ERP*.

1.1 A Rech Informática

A Rech Informática Ltda, empresa na qual foi elaborado este trabalho de conclusão, com sede na cidade de Novo Hamburgo/RS, vem atuando desde 1990 no mercado de *software* para gestão empresarial. A Rech atua predominantemente no estado do Rio Grande do Sul, mais especificamente no Vale do Sinos e região metropolitana de Porto Alegre, mas possui alguns clientes nos estados de Santa Catarina, Paraná, Paraíba e São Paulo. A empresa foi constituída a partir de um alicerce sólido de valores humanos, conjugando experiência, seriedade e competência nas áreas de informática para gestão empresarial.

A Rech Informática foi fundada em 15 de maio de 1990 pelos irmãos Rovani Marcelo Rech e Carlos Vanderlei Rech, os quais desde o início estiveram comprometidos em ofertar ao mercado produtos e serviços de alta qualidade. Foi idealizado o desenvolvimento de um software completo, construído de acordo com as necessidades de cada cliente, atendendo a legislação em vigor , visando à eliminação de retrabalhos e desperdícios nas empresas, através do conhecimento de gestão empresarial conjugado com o uso de tecnologias viáveis e produtivas. O objetivo principal da Rech Informática é a conquista diária da satisfação de seus clientes, fazendo com que se tornem parceiros numa relação cliente x fornecedor. Para isto, a empresa investe constantemente em novas tecnologias e no treinamento de pessoal,

aprimorando o SIGER e a qualidade dos serviços prestados. Com esse comprometimento em atingir os objetivos propostos, a Rech Informática constantemente vem investindo na qualificação da equipe e na própria estrutura da empresa, tornando-a um ambiente agradável de trabalho, e em harmonia com a natureza. Estes investimentos são tanto em estrutura física, quanto em recursos de pessoal, em busca de melhorar cada vez mais o relacionamento da empresa com seus clientes, fornecedores e colaboradores.

1.2 Princípios da Rech Informática

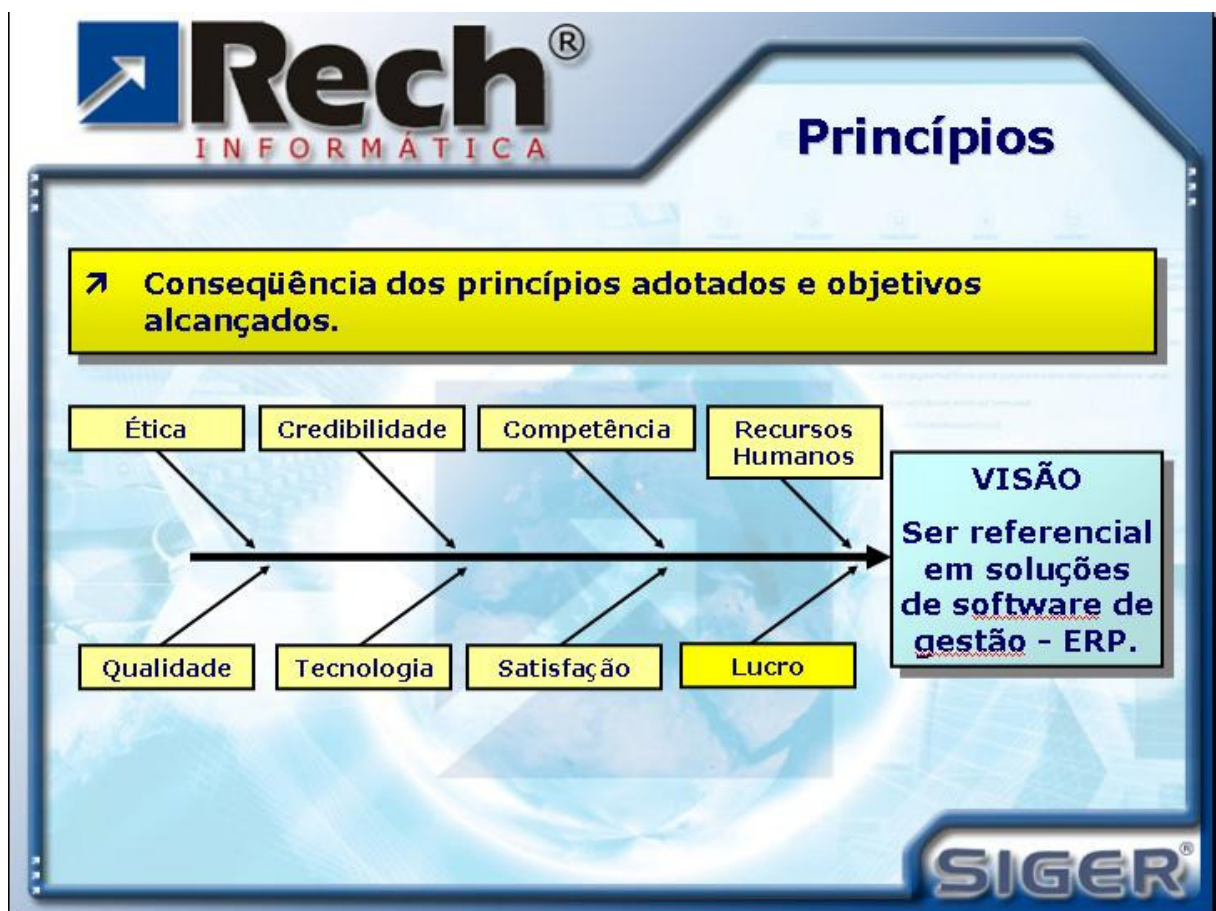


Figura 1-1 - Princípios adotados pela Rech Informática Ltda.

Fonte: Apresentação RECH – IENH - 2008

- Ética e justiça nos relacionamentos internos e externos;
- Credibilidade conquistada a partir da humildade, respeito, seriedade, transparência, comprometimento e crescimento sustentado;
- Valorização do pessoal, espírito de equipe e aprimoramento contínuo;
- Foco na qualidade total como receita para o sucesso;

- Superar as expectativas dos clientes com soluções de qualidade;
- Lucro como consequência dos princípios adotados e objetivos alcançados.



Figura 1-2 - Fatores críticos de sucesso I.
Fonte: Apresentação RECH – IENH – 2008



Figura 1-3 - Fatores críticos de sucesso II.
Fonte: Apresentação RECH – IENH – 2008

1.3 Metodologia de treinamento

A empresa optou por adotar duas metodologias de treinamento de novos colaboradores, com o intuito de torná-los capazes e profissionais num prazo curto de tempo. Uma metodologia destinada para área de desenvolvimento de sistemas e outra para a área de relacionamento com o mercado.

1.3.1 Metodologia de treinamento – GPD

O setor de GPD - Gestão de Pesquisa e Desenvolvimento abrange o levantamento de requisitos (que geralmente é feito juntamente com o pessoal da área de relacionamento com o mercado), especificação de requisitos, análise de requisitos, modelagem de dados, interface, programação, teste de software, pesquisa e assistência técnica para o ambiente interno.

Inicialmente não havia uma metodologia definida, mas se iniciou esse trabalho com entidades, inclusive a própria Feevale, de montar uma equipe de estagiários em conclusão de

2º grau. Mesmo sem nenhuma experiência em programação de computador, o grupo é incorporado num programa de treinamento, o qual ocorre durante um período de seis meses. Com o sucesso dos primeiros grupos, essa técnica foi aprimorada, sendo que atualmente é possível formar um grupo em três meses de treinamento interno, diretamente na ferramenta de trabalho, tudo sendo realizado e monitorado através de um colaborador com mais experiência na área do GPD.

Concluído esse primeiro passo, as equipes recebem um treinamento para conhecimento geral do SIGER. Esse é executado por um colaborador do GRM – Gestão de Relacionamento com o Mercado. Nesse treinamento, com duração de 5 dias, são transmitidos conhecimentos dos módulos do SIGER, ficando um espaço aberto para os colaboradores em treinamento darem opiniões e sugestões no sistema, as quais serão repassadas aos analistas da empresa. O papel do instrutor, além do treinamento em si, é de identificar com qual área cada um dos novatos mais se enquadra e mostra interesse, pois a tendência de incluí-lo em uma célula ou outra começa pela identificação do perfil.

O próximo passo é um contato com o mercado. Mais uma vez ocorre durante cinco dias, onde são marcadas visitas em clientes. Um colaborador do GRM juntamente com o estagiário efetua a visita de implantação, permitindo ao novo colaborador um contato direto com a metodologia aplicada pela Rech, dando a oportunidade de conhecer os processos do cliente.

Por último ocorre, durante 30 dias, um estágio numa empresa parceira da Rech, a qual fabrica o equipamento e o *software* de coleta de registros em cartão ponto, integrando seu sistema com o de folha de pagamento da Rech.

1.3.2 Metodologia de treinamento – Relacionamento com o Mercado

O setor de GRM - Gestão de Relacionamento com o Mercado, abrange todo e qualquer movimento em que ocorre algum tipo de contato diretamente com o cliente e/ou contatos a potenciais clientes.

Diferentemente do GPD, a seleção dos técnicos para composição da equipe do GRM é feita através de uma seleção de currículos, onde são avaliadas experiências anteriores, desde gestão industrial até áreas contábeis e fiscais, de preferência que demonstrem algum contato anterior relacionado ao uso do SIGER.

Uma vez feita uma seleção de turma, em média 5 ou 6 colaboradores, monta-se o cronograma de treinamento funcional no SIGER, o qual é realizado detalhadamente em todos os módulos, transmitindo a idéia com a qual o sistema foi montado, suas formas de navegação e padrões de linguagem. O período dessa incubação é de 3 meses, sendo feito na sala de treinamento da Rech, onde há disponibilidade para projeção de dados e um computador para cada aprendiz. Durante o treinamento interno, o instrutor tem o papel de identificar o perfil de cada aluno, destinando cada um para uma área específica de atuação no SIGER.

A última etapa trata de levar os novos colaboradores a conhecer a rotina externa, ou seja, farão visitas a campo para aprender os processos dos clientes e problemas comuns do cotidiano. Essa última etapa tem duração média de 15 dias.

1.4 Metodologia de trabalho

A RECH, para melhor atender seus clientes e cuidar de seus processos internos, desenvolveu uma ferramenta para gerenciar todas as suas atividades internas: o SICLA – Sistema Interno de Controle de Ligações e Atendimentos. Essa ferramenta está sendo aperfeiçoada diariamente como adaptação à evolução dos processos da empresa e às necessidades demandadas pelo mercado. Essa ferramenta também serve para administrar a fila de produção da Rech, onde todos os pedidos feitos pelos clientes são devidamente registrados. Uma equipe se reúne semanalmente, onde são discutidas as solicitações recebidas durante a semana. Nesse momento é feita uma análise superficial e avaliado se haverá algum custo de desenvolvimento para o cliente. Datas de entrega são definidas, onde posteriormente os clientes serão notificados do aceite de sua alteração/implementação no SIGER e a data prevista para a entrega.

Nesse mesmo encontro EED – Encontro de Encaminhamento da Demanda, após pré-análise, a solicitação é direcionada para um analista do GPD, o qual assume a responsabilidade do cumprimento dos prazos. Esse mesmo analista tem a função de investigar a solicitação, analisar os requisitos ou mesmo deslocar-se até o cliente para ver diretamente o solicitado, podendo dar sugestões alternativas, desde que as mesmas atendam devidamente a necessidade apontada.

Uma vez esclarecidos os requisitos, o analista processa o pedido do cliente transformando-o em uma OP – Ordem de Produção, documento utilizado pela Rech para

detalhar a solicitação recebida. Nessa OP, é feito todo o detalhamento necessário para que um programador receba a mesma e possa realizar as implementações no sistema. A elaboração da OP, dependendo do tamanho da solicitação, pode levar semanas até sua total conclusão. Esse documento é formado por matrizes de campo, *layout* de tela, *layout* de listagem, tabelas e todos os relacionamentos necessários para o total funcionamento da implementação.

Uma vez concluída a implementação, dá-se início aos testes do *software*, lembrando que a Rech sempre reforça para sua equipe de programadores que o melhor teste é aquele feito durante a implementação. Por estar focada no código fonte, a atenção redobrada pode evitar erros que passam despercebidos nos testes de validação, fato que vem trazendo bons resultados. Os testes são feitos de forma individual pela mesma pessoa que executou a implementação. Um outro grupo, ainda do GPD, realiza um teste estrutural e funcional, teste este que tem duração de 60 minutos diários, tempo que pode variar de acordo com o acúmulo de OP's em processo com necessidade de teste.

Após os testes do GPD, é a vez da equipe do GRM efetuar o seu teste, sempre com a presença do técnico que levantou o requisito junto ao cliente. Trata-se de uma validação funcional: se atender a necessidade solicitada, a OP passa para o *status* de validada, retornando ao GPD para oficializar a implementação na versão instalada no cliente.

Por último, um técnico do GRM desloca-se até o cliente entregando o seu pedido e realizando um treinamento para o uso da funcionalidade, momento em que é feita a validação por parte do solicitante. Uma vez validada, ao retornar para a sede da Rech, a OP será encerrada.

1.5 Metodologia de testes de software

Uma das definições de qualidade de *software* disponibilizadas é a de Pressman (1995, p. 724), pela qual qualidade de software é:

Conformidade a requisitos funcionais e de desempenho explicitamente declarados, a padrões de desenvolvimento claramente documentados e a características implícitas que são esperadas de todo software profissionalmente desenvolvido.

Uma das questões mais fundamentais na execução de um teste é caracterizar quais (e quantos) testes são necessários para garantir a qualidade do produto.

Muitos acreditam que o objetivo de um teste de software é verificar se o mesmo está livre de erros. Na verdade é impossível efetuar esta verificação, pois não é possível garantir que um sistema está 100% livre de erros.

O objetivo do teste de software deve ser sempre o de encontrar erros no sistema, refutando a afirmação de que o produto está correto. Ou seja, o teste que revela problemas é considerado um sucesso. É um processo destrutivo, sob o ponto de vista psicológico. Obviamente, o propósito em se revelar problemas é consertá-los.

As causas mais prováveis para ocorrência de erros são:

- Falha de comunicação;
- Complexidade do sistema;
- Erros de programação;
- Mudança de requisitos no decorrer de um projeto;
- Pressão para terminar o produto o mais rápido possível.

Quanto antes o defeito for revelado, menor o custo agregado e maior a probabilidade de efetuar a correção do erro (WEBER; ROCHA; NASCIMENTO, 2001).

Encontrar e consertar erros quando o próprio programador os detecta tem custo baixo, pois não existe custo de comunicação, o erro não tem de ser explicado para ninguém, não precisa entrar para um sistema de acompanhamento de erros (RNC), além de não influenciar o trabalho de mais ninguém, principalmente do cliente que utiliza o *software*.

O teste de *software* deve utilizar técnicas devidamente estabelecidas, onde as tarefas são distribuídas de forma sistemática. Sendo necessária a criação de um plano de testes, estabelecendo diretivas a serem seguidas. As técnicas não devem ser aplicadas isoladamente, e sim de forma complementar.

O teste do *software* engloba a utilização das etapas de verificação, validação e teste.

Verificação: garante que uma funcionalidade foi implementada seguindo os padrões do processo de desenvolvimento, de forma a facilitar o entendimento e manutenções das rotinas. Deve responder a seguinte pergunta: “Estamos construindo corretamente o *software*, de acordo com os padrões?”.

Validação: garante que uma funcionalidade foi implementada conforme os pré-requisitos e suas especificações, atingindo as necessidades do usuário final. Deve responder a seguinte pergunta: “Estamos construindo o *software* correto, de acordo com as especificações?”.

Teste: atividade de avaliar uma funcionalidade por meio de sua execução, a fim de encontrar erros. Deve considerar cenários com dados reais, bem como todas as exceções previstas no domínio da implementação e suas conseqüências.

Baseando-se nesses fatos, a Rech, preocupada com a qualidade do *software* para o bom funcionamento em seus clientes, ajudando-os a atingirem seus objetivos e torná-los mais competitivos, construiu e vem aperfeiçoando sua metodologia de teste de *software*. Foi quando surgiu, em maio de 1995, o Procedimento de Testes Rech, visando a garantia da qualidade, através do conjunto de atividades técnicas aplicadas durante todo o processo de desenvolvimento. Define-se os conceitos e atividades para garantir que os produtos e processos de *software* estão em conformidade com seus requisitos especificados, aderindo aos seus planos e padrões estabelecidos. A Rech entende que a qualidade de um *software* é obtida a partir da utilização de uma metodologia durante todo o processo de desenvolvimento.

1.6 SIGER

O SIGER – Sistema de Gestão Empresarial RECH, é composto por módulos, os quais podem ser adquiridos separadamente ou em conjunto, conforme a necessidade de cada empresa. É totalmente construído pela equipe da Rech, na sede da empresa, usando tecnologias avançadas, aprimorando as mesmas para a realidade atual. O SIGER é um ERP que não é vendido, não tem custo de implantação para o cliente, todo esse investimento é feito pela RECH, a qual aposta numa parceira de longo prazo, pois não considera junto o desembolso de alto custo de implantação e customização da parte do cliente, pois caso o sistema apresentado não atenda as suas necessidades, um provável cancelamento de contrato pode surgir, o que não é bom para nenhum dos lados. O Cliente pagou e não teve o retorno esperado e a Rech não conseguiu atingir seus objetivos.

Em entrevista para a revista Ação Lojista, o diretor Rovani argumentou: “Não vendemos o *software*, trabalhamos com prestação de serviços. Somos comprometidos com o funcionamento do sistema do cliente”.

A Rech Informática desenvolve soluções em sistema de informações para gestão empresarial. Com foco no entendimento do negócio do cliente e seus relacionamentos com o mercado busca, através de metodologias aplicadas, traduzir as necessidades em ferramentas simples e eficazes que venham a agilizar a tomada de decisões.

Atendendo a diversos ramos de atividade, a Rech vem se aprimorando na implantação e suporte de uma solução completa que atenda a todas as áreas de uma empresa de forma integrada.



Figura 1-4 – SIGER e seus módulos atuais.
Fonte: Apresentação RECH – IENH – 2008

2 SIGER - SISTEMA DE GESTÃO EMPRESARIAL RECH

O SIGER é estruturado em módulos, onde há equipes destinadas a manter o funcionamento e estudar novas funcionalidades e tecnologias a serem agregadas. A criação de novos padrões ou mesmo a melhoria dos processos atuais são testados durante um determinado período de tempo. Obtendo sucesso, são publicados mensalmente à todos os demais programadores e analistas, sempre com o aval da direção da empresa.

2.1 Surgimento do SIGER

Surgiu em 1990, quando foi idealizado o desenvolvimento de um *software* ERP completo, que eliminasse por inteiro os retrabalhos e desperdícios nas empresas, através do conhecimento de gestão empresarial com o uso de tecnologias produtivas e viáveis. Em 1994 o SIGER atingia 50 clientes ativos, atualmente esse número ultrapassa a marca de 550.

2.2 A Estrutura de módulos

Foi optado pelo desenvolvimento do sistema na estrutura de módulos, permitindo que cada cliente monte a solução juntamente com a equipe de consultoria da Rech, de forma que atenda completamente sua gestão sem ter que adquirir partes do *software* que não lhe interessam. Assim, pode facilmente adaptar o SIGER aos seus processos e interesses, de acordo com suas condições financeiras. Essa estrutura de módulos facilita a aquisição de outros, pois todo o sistema possui o mesmo padrão de navegação e validação de campos, bem como toda a interface com o usuário. Portanto, ao adquirir um novo módulo, o usuário já estará familiarizado com o ambiente.

2.3 Gestão Industrial

O enfoque principal deste trabalho será dado no módulo de Gestão Industrial, o qual está em constante implementação. Vem sendo construído com critérios voltados aos processos industriais, com freqüentes visitas a clientes juntamente com consultores. Outra preocupação da Rech é o aperfeiçoamento de seus técnicos na área industrial, por isso vem investindo em cursos de aperfeiçoamento, tais como: CEP, MRP I, MRP II, 5's.

Esse módulo tem por objetivo permitir o registro das estruturas de produtos e suas variações de forma, automatizando os processos de produção (puxada ou empurrada), relacionando insumos, horas e recursos necessários ao atendimento do plano de produção. Também procura facilitar o processo de tomada de decisões, bem como padronizar de forma flexível os processos de fabricação e seus registros (apontamentos), além de obter um melhor aproveitamento da mão-de-obra empregada. Procura, ainda, manter sob controle os prazos de entrega através da extração de relatórios com os estágios do processo fabril. Com a utilização do conceito de ordem de produção, se busca manter o histórico de cada evento, garantindo consultas e extração de listagens de custo de produção, horas aplicadas e recursos utilizados.

Conforme explica Favaretto (apud BOARETTO, 2004, p.1) os investimentos feitos em sistemas ERP geralmente não proporcionam os benefícios esperados para as empresas contratantes, pois as mesmas ignoram os dados operacionais críticos do chão-de-fábrica.

Para produzir com eficiência é necessário informar aos centros produtivos quais operações devem ser executadas e a quantidade atingível em cada dia produtivo para assim alcançar os resultados almejados. A necessidade de um setor para coordenar os centros produtivos e gerenciar os estoques de matéria prima torna-se fundamental para que os objetivos sejam alcançados. “A programação e controle da produção consiste essencialmente em um conjunto de funções inter-relacionadas que objetivam comandar o processo produtivo e coordená-lo com os demais setores administrativos da empresa” (ZACARELLI, 1976, p.1).

Este setor necessita com um conjunto mínimo de informações, compreendendo: saldo em estoque, previsão de vendas, componentes necessários para os produtos elaborados, engenharia de cada componente, capacidade produtiva da planta, tempos de processo, etc. São informações dispersas entre os vários setores da empresa, sendo necessária a classificação das mesmas para programar e controlar a produção. O produto resultante do PCP é a ordem de fabricação, na qual vai a informação de quando, quem, onde, quanto, com o que e como será

produzido. Estas informações necessitam estar disponíveis a todos os setores da empresa, mesmo não estando diretamente ligado à produção, para que se tenha a visão macro do processo e suas necessidades.

O PCP consiste num papel fundamental, além de programar, deve controlar se a execução das ordens está conforme o planejado e ajustá-las caso ocorram imprevistos, o que não é muito difícil de ocorrer. De uma forma geral, pode-se dizer que as atividades de controle e comando da produção seguem os seguintes passos:

- Estabelecer o que deve ser produzido e o quanto de recursos (tempo, material e equipamentos) deve ser empregado;
- Efetuar a coleta de informações sobre o que foi realizado e o quanto de recursos foram utilizados;
- Comparar o previsto com o que foi realizado;
- Tomar providências considerando as diferenças verificadas.

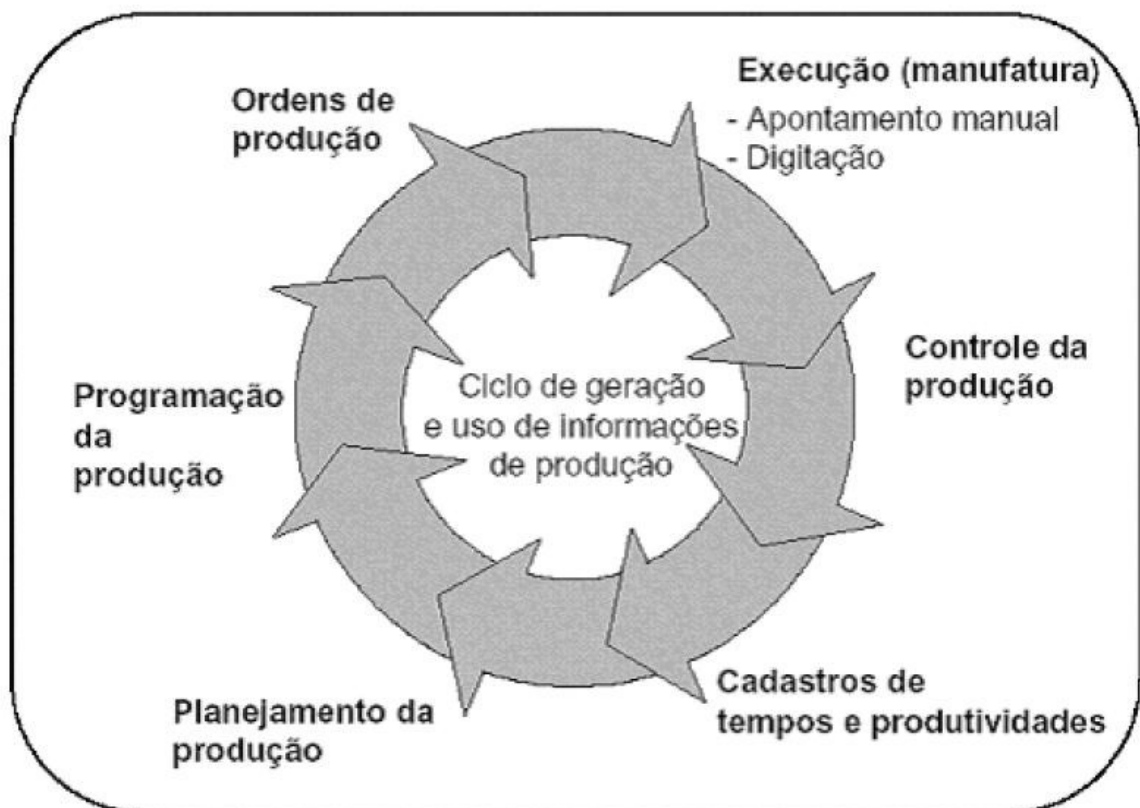


Figura 2-1 – Ciclo de Coleta de dados chão-de-fábrica.
Fonte: (BOARETTO, 2004)

2.3.1 Funcionalidade da coleta de dados

A coleta de dados no SIGER consiste na reflexão de todos os movimentos ocorridos durante o processo produtivo, desde os apontamentos de quantidade produzida, apontamento de tempo de processo, destinação de quantidade refugada, a qual pode ser reciclada, sucateada ou re-classificada, além de apontamentos de recursos do tipo máquinas, uso de ferramentas e material de consumo. Manutenção produtiva e preventiva são outros destaques de apontamentos suportados pelo SIGER.

Atualmente existem diversas formas de coleta de dados no SIGER, dentre elas algumas em destaque:

- Integração com base externa: O SIGER ainda não está preparado para interagir com o CLP da máquina, sistemas alternativos podem fazer tal função, integrando diretamente com a base de dados, refletindo os apontamentos no ERP;
- Leitura óptica: Consiste na leitura do código de barras extraído da OP diretamente num microcomputador, código de barras que pode conter o setor produtivo ou não. Essa leitura é feita através de um leitor de código de barras, onde sempre é aconselhado o uso de modelos homologados pela RECH. Esses leitores podem ser com ou sem fio;
- Digitação manual: Quando o ciclo produtivo é de baixa escala, a coleta de dados pode ser feita por digitação manual, processo bastante ágil, mas não recomendado, pois essa modalidade permite que ocorram erros de digitação, o qual com leitura de código de barras é mínimo;
- Lote: Um lote é formado por um conjunto de ordens de produção, montado por critérios que se adaptam a realidade da empresa, podendo ser diário, turno, hora, ou grupo de produtos. Em geral, utilizado quando a necessidade da informação do que foi realizado não é instantânea;
- Apontamento por microterminal: Usado quando existem vários pontos de coleta de dados e a quantidade de ordens de produção em processo é de larga escala. Para essa configuração é necessário a aquisição de um microcomputador, um *hub*, um microterminal e um leitor óptico. Esse conjunto atende um ponto de leitura. Para pontos adicionais basta adquirir outro microterminal e outro leitor óptico.

2.3.2 A necessidade de um simulador de coleta de dados

Este trabalho tem como objetivo apresentar uma proposta de modelagem e implementação de um simulador de microterminal Colleter para o SIGER especificamente na opção de apontamento de produção por microterminal. Como o SIGER está em constante construção nessa área da gestão, surgem alguns motivos para implementar tal *software*:

- Velocidade na solução de problemas no caso de surgimento de falhas;
- Baixo custo na solução das falhas;
- Equipamento sempre disponível;
- Facilidade na implementação de novas funcionalidades;
- Maior tempo para executar os testes;
- Facilidade de simular o ambiente do cliente;
- A Rech teria que adquirir todos os tipos de equipamentos de coleta de dados;
- Caso optar pelo uso do equipamento do cliente, esse ficaria sem poder efetuar os apontamentos pelo microterminal durante a ausência dos mesmos.

Através desses motivos, a construção de um protótipo baseado em um modelo de microterminal ganhou força e foi aceita pela direção da Rech. Se o modelo a ser implementado obtiver sucesso, poderá ser ampliado a outros modelos e marcas de microterminais, sempre com a homologação da Rech. Algumas das funcionalidades de apontamentos de produção existentes no SIGER podem ser implementadas no protótipo, sendo que, dependendo do êxito, abre-se opção para implementar todas as demais futuramente.

3 COLETA DE DADOS EM CHÃO DE FÁBRICA

Os sistemas ERP atualmente estão presentes na maioria das empresas. Existem aquelas que utilizam outros métodos por ainda não terem investido num sistema integrado. A coleta de dados semi-automatizada em pontos chave do chão-de-fábrica, permite aos operários efetuarem entradas de dados diretamente no sistema, fornecendo *feedback* do processo produtivo em tempo real, oferecendo aos gerentes setoriais a informação necessária para a tomada de decisão em tempo hábil. Apontamentos de exceções fazem parte do processo produtivo, sendo necessário serem devidamente apontadas no momento em que ocorrem.

3.1 Microcomputadores

Uma das alternativas de disponibilizar pontos de coleta de dados no chão-de-fábrica é o uso de microcomputadores, acessando diretamente o sistema utilizado pela empresa. Esta opção é utilizada desde o final da década de 1980, pois naquela época havia poucos dispositivos específicos para tal tarefa. Atualmente, o uso de computadores está novamente freqüente, em função do baixo custo de aquisição e reaproveitamento de microcomputadores obsoletos para outros setores da empresa. Indicado também quando o volume de apontamentos não é muito significativo, situação que não justifica o uso de outros equipamentos.

3.2 Características

Como características gerais, a maior parte dos equipamentos será composta por um microcomputador, um monitor de vídeo para visualização de mensagens e a coleta de dados em si. Um leitor de código de barras das ordens de produção e um teclado para eventual

entrada manual. Necessário, ainda, uma interface de rede possibilitando a comunicação com o servidor de aplicação do sistema da empresa.

Esses microcomputadores instalados para coleta de dados normalmente executam uma única aplicação, assim não necessitam de grande poder de processamento, sendo viável e indicado nas situações em que a empresa não possa investir em outras tecnologias, reaproveitando microcomputadores obsoletos de outros setores.

Para disponibilizar um microcomputador no chão-de-fábrica, são requeridos alguns cuidados especiais quanto ao ponto físico, rede elétrica e dados. Os pontos de coleta normalmente estão posicionados em locais com considerável fluxo de pessoal. Para evitar a ocorrência de quedas ou danos, o móvel empregado deve ser afixado, comportando o microcomputador dentro de sua área, evitando batidas e quedas.

A utilização de uma rede elétrica isolada para alimentar os circuitos computacionais é desejável, uma vez que a rede do ambiente industrial apresenta baixa qualidade (ruído). O uso de estabilizadores de tensão, juntamente com um bom aterramento são fundamentais para o bom funcionamento do computador, pois do contrário, pode ser danificado fluxo de dados e partes físicas do microcomputador

Quanto à rede de dados, deve-se ter o cuidado para que os cabos não passem próximos a máquinas geradoras de ruído (ex.: motores elétricos) evitando a interferência eletromagnética. O correto é seguir as normas da ABNT para cabeamento estruturado de redes, descritos na norma “NBR 14565 – Procedimento Básico para Elaboração de Projetos de Cabeamento de Telecomunicações para Rede Interna Estruturada” (JUNIOR, 2007).

3.3 Considerações sobre uso de microcomputadores

Infelizmente algumas pessoas ainda apresentam aversão à tecnologia e, caso não tenham recebido uma orientação adequada, podem gerar falhas propositais, inviabilizando a solução adotada para os apontamentos de produção.

Quando o microcomputador apresentar problemas e precisar ser retirado para manutenção, pode-se facilmente re-alocar outro para a mesma função. Em contrapartida, o custo de manutenção pode ser elevado considerando a quantidade de paradas que o mesmo pode sofrer por estar em um ambiente de produção, os quais geralmente não apresentam

características apropriadas para um microcomputador, tais como: ambientes com excesso de poeira, temperatura elevada, características que prejudicam diretamente o uso do microcomputador.

Um microcomputador pode ocupar um espaço físico que muitas vezes a empresa não tem disponível. O *layout* das máquinas e posições de trabalho é arranjado de tal forma que otimiza os tempos de produção, mas não prevê um espaço que acomode com segurança um microcomputador.

O consumo de energia de um computador com monitor, conforme Torres (2001, p.1189), está na faixa de 300 VA (200 watts), isso pode variar, pois computadores atuais tendem a ser mais econômicos que modelos mais antigos (MAIOLLA, 2007). Como é comum o re-aproveitamento de microcomputadores oriundos de outros setores, a tendência é que estes sejam os responsáveis por grande parte do consumo elétrico. Contabilizando tudo, pode-se ter um custo elevado de energia elétrica.

3.4 Microterminais TED 1000 e TED 5000

A empresa Colleter Informática Ltda, fundada em 1989, com sede em São Paulo, têm como meta oferecer produtos e serviços para o segmento de automação, especificamente para sistemas automáticos de captura de dados, através da leitura de código de barras e coletores *on-line* interligados e gerenciados por microcomputadores.

A empresa possui área de desenvolvimento de produtos, o que a torna apta a desenvolver projetos conforme as necessidades de seus clientes. Conforme a página da empresa na internet (COLLETER, 2009), a mesma possui mais de 20.000 pontos de automação.

A linha de produtos TED 1000 / TED 5000 surgiu da “... necessidade de leitura de código de barras em pontos espalhados em uma planta industrial ou comercial, com os dados convergindo para um banco de dados central.” (COLLETER, 2009).

3.4.1 Características do hardware

A peça central dos microterminais TED 1000 / TED 5000 é um processador Motorola GP que trabalha com frequência de 19,66 MHz e é complementado por uma

memória Flash de 18 Kb utilizada para programas internos além de 128 Kb (expansíveis a 380 Kb) para o armazenamento de dados coletados quando em modo *off-line*. A memória é alimentada por uma bateria interna recarregável de NiMH (níquel metal hidreto) que mantêm os dados mesmo que o microterminal esteja sem alimentação elétrica.

A diferença entre os modelos de microterminais é que o TED 1000 apresenta uma tela de cristal líquido para a visualização de mensagens que possibilita a leitura em 2 linhas com 16 caracteres em cada linha, além de ter um teclado com 16 teclas (caracteres numéricos e 5 teclas de função além do ponto), enquanto que o TED 5000 apresenta uma tela de cristal líquido de 2 linhas com 40 caracteres por linha e um teclado de 42 teclas (teclas alfanuméricas mais teclas de funções).



Figura 3-1 – Microterminais TED 1000.

Fonte: (COLLETER, 2009)

Para que os dados coletados sejam enviados para o sistema gerencial da empresa, é necessário que os microterminais estejam conectados a um microcomputador que terá a função de gerenciá-los. A comunicação entre o microcomputador gerenciador e os microterminais pode ocorrer através das seguintes opções:

- Uma interface serial padrão RS232C para COM1 ou COM2, comunicação direta com computadores;
- Uma interface serial padrão RS48512 bidirecional;

- Opcionalmente oferece interface TCP/IP para rede ethernet 13 (TED 1010) ou interface para comunicação via Rádio Frequência Wi-Fi 14 (TED 1020).



Figura 3-2 – Vista traseira do TED 1000.
Fonte: (COLLETER, 2009)

Para a conexão de periféricos, os microterminais apresentam uma porta com interface *wedge* (teclado) que possibilita a conexão de leitores externos para código de barras como canetas, laser e CCD. Possui ainda uma interface para *slot reader* incorporada ao gabinete que permite a leitura de cartões magnéticos e cartões com código de barras. Em ambas, as *interfaces* podem realizar a decodificação dos principais padrões de códigos de barras: code 39, code 39 *full* ASCII, UPC, EAN, *Codebar*, 2 de 5 entrelaçado, 2 de 5 industrial e *code* 128. Possui ainda uma porta serial para a comunicação com impressoras ou balanças eletrônicas, caso seja necessário.

Além da sinalização pela tela de cristal líquido, o microterminal apresenta um indicador luminoso (LED) e um indicador audível tipo “*beep*”. Opcionalmente, é possível instalar uma bateria de NiMH que garante uma autonomia de até 12 horas de funcionamento para o microterminal, quando este não apresenta alimentação através de energia elétrica.

3.4.2 Infra-estrutura do TED 1000

Para a conexão dos microterminais em rede é utilizado um controlador *multiserial* com programação própria e capacidade de gerenciar até 16 microterminais. Este controlador é denominado *hub* pelo fabricante. O *hub* deve estar conectado através de uma porta serial a um microcomputador que fará o gerenciamento, podendo ser empregados até quatro *hub*'s por microcomputador. Dessa forma, é possível expandir o gerenciamento para até 64 microterminais, efetuando a troca de informações destes com o sistema gerencial da empresa.

É importante lembrar que os microcomputadores atuais possuem apenas uma porta serial, sendo necessária a aquisição de placas *multiserials* para expandir esta capacidade.

A instalação dos microterminais é facilitada, pois não há necessidade de configuração específica. O *hub* se encarregará de gerenciá-los. A conexão entre os microterminais e o *hub* ocorre através de uma interface serial padrão RS485 de comunicação bidirecional. Já a comunicação entre o *hub* e o microcomputador ocorre através de uma interface serial RS232.

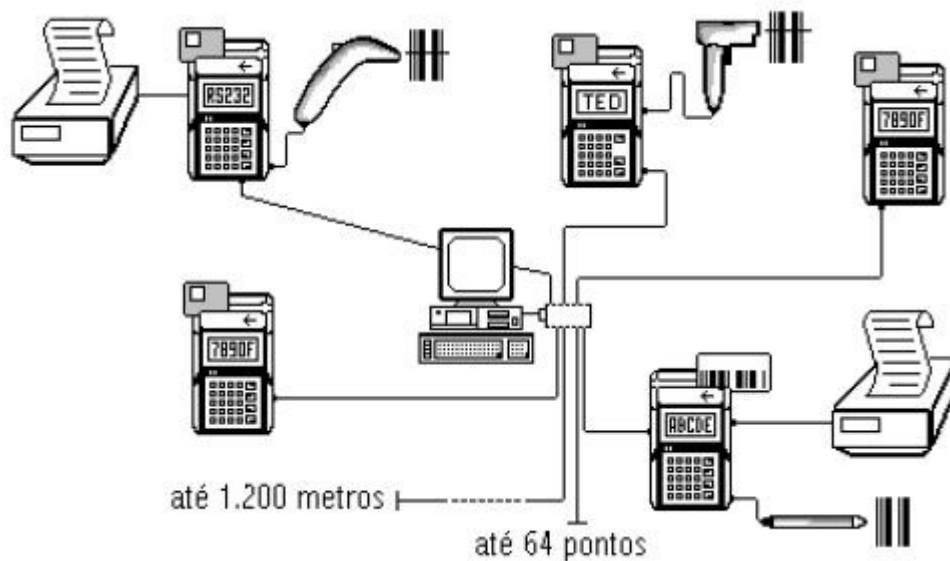


Figura 3-3 – Topologia de rede utilizada pela Colleter.

Fonte: (COLLETER, 2009)

Cada microterminal pode ser instalado a até 1200 metros de distância do *hub*. Deve-se considerar que o comprimento máximo do cabo de comunicação não exceda 1500 metros. A interligação pode ser feita através de cabo telefônico ou cabo de rede *ethernet* (par trançado). Estes cabos podem ser instalados em conduítes junto a cabos de energia e comunicações, mas afastados de fontes geradoras de ruído (como motores elétricos).

Para aplicações *on-line* (onde o microterminal precisa estar conectado a um microcomputador gerenciador) são fornecidas bibliotecas com funções de alto nível para várias linguagens de programação, podendo ser facilmente integradas a sistemas desenvolvidos em *Clipper*, *C*, *Visual Basic* e *Delphi*. Estas bibliotecas contam com inúmeras funções comportamentais de operações prontas, para facilitar a adaptação ao sistema. Exemplos de funções oferecidas pelas bibliotecas são:

- Parâmetros de comunicação;
- Parâmetros de comportamento;
- Recepção de dados coletados (teclado e leitor);
- Envio de mensagens ao *display*;
- Envio de texto à impressora do terminal;
- Envio de sinalização sonora (*beep*) e *led's*;
- Captura de dados de balança digital;
- Envio / recepção de sinais digitais.

3.4.3 Considerações sobre o microterminal Colleter

Apresenta características interessantes como solução para coleta de dados em chão-de-fábrica:

- Baixo consumo de energia;
- Comunicação de rede simplificada, apesar de sua topologia em estrela, obriga o uso de um par de comunicação específico para cada terminal;
- Facilidade de programação;
- Reduzido número de paradas para manutenção;
- Grandes distâncias entre microterminal e *hub*.

3.4.4 Homologação do microterminal

O aparelho testado é simples, possui um *display* com iluminação própria e teclado. Exibe mensagens e retorno de entrada no teclado. Possui processador Z80 8 bits. Comunica através da porta serial. Permite conectar leitor de código de barra de pinagem 270 (diferente do teclado que, é 180).

Existem dois modelos: o TED1000 e o TED5000. O TED1000 tem dimensão pequena, *display* 16x2 caracteres e teclado numérico com algumas teclas de funções,

semelhante ao Passo TR100, porém menor. Também existe o TED1000 de 4 linhas. O TED5000 possui dimensão maior com *display* de 40x2 caracteres, teclado alfanumérico e numérico. Ambos são alimentados por uma fonte *bivolt*.

Permite comunicação protocolo RS232 (quando diretamente ligado no PC) ou RSR435 (longa distância) para ser utilizado em rede com outros terminais. Não necessita de *jumper* para alternar entre os protocolos. Para ser utilizado em rede precisa de um *hub* específico que pode conectar 8 terminais. Pode ser ligado outro *hub* simultaneamente a outra porta serial. O *hub* é alimentado por fonte *bivolt*.

A Rech obteve contato com um *hub* de oito portas e dois modelos de microterminal: TED1000 de *display* 16x2 caracteres e TED5000 de 40x2 caracteres.

4 SOFTWARE DE SIMULAÇÃO DE UM MICROTERMINAL

Com base nas informações relatadas até o momento, pode-se agora iniciar o estudo da ferramenta a ser desenvolvida: um *software* simulador de microterminal. Um fato que auxiliou muito a pesquisa foi uma visita em um dos clientes da Rech, onde o objetivo principal foi observar a rotina de apontamentos de produção em tempo real, juntamente com o gerente industrial, o qual estava à disposição para esclarecer eventuais dúvidas. Foi um momento de aprendizado de processo de gestão, pois analisando os procedimentos pode-se notar que algumas rotinas executadas pelos operadores no sistema poderiam ser implementadas de outra forma, a qual facilitaria o cotidiano dos operadores de produção, economizando tempo e evitando erros de digitação.

4.1 Análise de chão de fábrica

A visita ocorreu na data de 24 de março de 2009, na cidade de Novo Hamburgo/RS. É uma empresa que fabrica componentes (saltos, solados e tacos) para cadeia calçadista, com produção média de mais de 100.000 pares/dia, utilizando em torno de 100 injetoras. Essa empresa recebe seus pedidos por EDI - *Electronic Data Interchange* (Intercâmbio Eletrônico de Dados), o que significa troca de dados entre empresas através de meio eletrônico, utilizando um padrão conhecido pelas partes. Trata-se da implementação de rotinas de intercâmbio de mensagens eletrônicas no padrão EDIFACT/EANCOM. A especificação destes documentos é um padrão internacional definido pela ONU. É simplificada, documentada e difundida no Brasil pela GS1. Muitas vezes o processo de recebimento do pedido até a entrega do mesmo ocorre em um ou dois dias e, em alguns casos na mesma data do recebimento. Os pedidos são importados para o SIGER, sendo que na seqüência serão importados para o módulo de Gestão Industrial do SIGER, no formato de ordens de produção com a explosão da engenharia do produto. A partir das ordens geradas, o sistema gera

etiquetas adesivas no formato GOL (Grupo de Otimização Logística do Setor Calçadista) para os pedidos de clientes que já aderiram ao novo padrão de intercâmbio de dados. Essas etiquetas são geradas com a quantidade de saltos que pode ser alocada na devida embalagem do produto, na qual será anexada a etiqueta com o número do SSCC: *Serial Shipping Container Code* (Número Serial de Unidades Logística), que é um número de identificação padrão de 18 dígitos, usado na identificação de unidades logísticas destinadas principalmente ao controle e rastreabilidade de mercadorias no embarque, transporte, recebimento e armazenagem. Ex: Caixas, Fardos, Pacotes, *Pallets*, etc.

A partir da carteira de pedidos, o gerente industrial faz a programação da fábrica, alocando as ordens de produção para as injetoras disponíveis, orientando-se pela data de entrega dos pedidos. Caso a carga ultrapassar a capacidade do dia, uma possível renegociação do prazo de entrega pode ocorrer juntamente ao cliente.

A partir desse ponto se inicia um processo de monitoração dos pedidos, baseando-se diretamente no apontamento de produção, procedimento que consta na leitura do código de barras de todas as etiquetas geradas para cada pedido a partir da ordem de produção. O apontamento de produção consiste na leitura do código de barras contido na etiqueta, na qual consta ainda a descrição do cliente, número da ordem de produção, número do pedido, referência do produto, unidade de medida, quantidade, cor e data de entrega. No momento em que todas as quantidades de saltos, solas e tacos são injetados, é feita a leitura do código de barras da etiqueta. Em seguida, o produto é acondicionado na unidade logística, na qual a etiqueta de produção é anexada. A monitoração ocorre tanto por parte do gerente industrial como também pelo setor de faturamento da empresa pois, dependendo da operação realizada no sistema, a posição do pedido vai alterando:

- Não importado – Não virou ordem de produção;
- Não programada – Ordem de produção não programada;
- Parcialmente programado – Apenas uma parte da Ordem de Produção gerou etiqueta de produção;
- Totalmente programado – Está pronto para ser produzido;
- Em produção/parado – iniciou a produção mas houve uma parada no processo;
- Produzido – Está totalmente produzido;

- Encerrado – Está pronto para ser faturado.

Um fato curioso nessa empresa é o de haver apenas um ponto de apontamento de produção para mais de uma injetora, ou seja, vários operadores utilizam o mesmo ponto de coleta de dados para efetuar a leitura da etiqueta. A rotina de apontamento de produção atual consiste em:

- Deslocamento da injetora até o ponto de coleta de dados;
- Informação do código do operador de produção;
- Informação do código da injetora;
- Leitura da etiqueta;
- Alocação da unidade logística na expedição;
- Retorno ao posto de trabalho.

Se tivesse um ponto de leitura para cada injetora, a informação código do operador de produção e o código da injetora poderiam ser informados uma única vez, na entrada do sistema. O gerente industrial alega que isso seria um investimento alto, pois possuem em torno de 100 injetoras ativas, além de não haver espaço físico para comportar com segurança todos esses pontos de leitura.

Além de servirem como parâmetro de atualização do andamento da produção dos pedidos, os apontamentos são analisados diariamente na composição do mapa de eficiência e produtividade, o qual pode ser feito por referência produzida, máquina (injetora), dia, turno, semana, quinzena ou mês. A composição do mapa pode ser em forma de listagem com percentuais e quantidades, bem como em forma de gráficos.

O gerente industrial relata que algumas falhas de apontamento ocorrem diariamente, falhas essas que acabam comprometendo a análise dos resultados de eficiência e produtividade:

- Falhas funcionais no software;
- Apontamento em injetora incorreta;
- Apontamento em operador incorreto;
- Falha na leitura do código de barras;

- Leitura de etiqueta incorreta.

O equipamento atual em uso pela empresa é composto por 8 microcomputadores com monitor, tecla, mouse e leitor ótico de código de barras, equipamento esse que está perto de ser substituído por microterminais Colleter. São oito equipamentos para atender apontamentos de mais ou menos 100 injetoras diariamente. A substituição por microterminais deve-se ao fato de uma futura mudança de *layout* de chão de fábrica, onde o espaço será ainda mais reduzido, de modo a deixar as injetoras muito próximas do setor de expedição da empresa.

4.2 Simulador

O propósito da visita nesse cliente da Rech, foi o de demonstrar que a rotina de apontamento de produção é de certa forma, complicada para o operador, pois o mesmo necessita informar seu código, injetora, etiqueta. Quando ocorrem paradas, precisa ainda indicar um código que sinaliza o início de uma parada e o próprio motivo da parada. O mesmo ocorre quando retorna a apontar, pois precisa antes indicar o fim da parada de produção. O simulador deverá contemplar todas as operações acima indicadas.

Esses aspectos reforçam ainda mais o motivo desse projeto, que é a criação de um *software* de simulação de microterminal Colleter. Como já foi reportado pelo gerente industrial, o surgimento de falhas na rotina de apontamentos muitas vezes é atribuído ao sistema. Para simular uma possível falha, é imprescindível reproduzir o ambiente onde o erro ocorreu, mas seria necessário ter o mesmo equipamento do cliente, neste caso o *hub* e o microterminal Colleter. Para não adquirir um equipamento ou até mais de um, pois podem ocorrer situações de haver falhas em apontamentos simultâneos vindo de coletores diferentes, a Rech optou por criar um simulador de algumas das principais funcionalidades do microterminal Colleter. Se obtiver sucesso, poderá expandir suas funcionalidades e estudar a possibilidade de usar o simulador em outros modelos de microterminal.

Com o simulador, a rotina de efetuar testes no *software* SIGER, bem como implementar novas funcionalidades nas rotinas de apontamentos de produção com uso de microterminais, será facilitada e agilizada, obtendo assim maior quantidade de recursos no sistema e podendo oferecer um *software* com mais qualidade.

4.2.1 Protocolo de comunicação

Outro fato motivador para não implementar as demais funções no protótipo é o de manter a maior parte possível do controle de fluxo de dados do simulador SIGER, de forma centralizada, pois se for possível manter uma comunicação síncrona com os demais microterminais, não haverá motivos para implementar demais funcionalidades, as quais podem ser bem distintas entre fabricantes e modelos. Como a pretensão é explorar outras marcas e modelos de microterminais, mantendo uma comunicação síncrona utilizando arquivo físico temporário, tudo indica que para desenvolvimentos em outros coletores, o esforço seja menor.

As funções disponíveis pelo microterminal Colleter vão bem além das que serão implementadas. Como se trata de um protótipo, optou-se por implementar funções básicas, com possibilidade de abranger as mesmas dependendo dos resultados obtidos. Vejamos o quadro completo de funções do microterminal:

- INITCOM - abre comunicação para 2 HUB's;
- ENVTMSG - envia mensagem ao display;
- GETCNT - lê mensagem do display;
- ENVTIMP - envia mensagem para impressora do TED;
- SETNCAN - seta o n. de canais ativos;
- ENVBEEP - tocar n. "beeps".
- ENVCMBEEP - tocar "beep" de inicialização;
- ENVLMTEC - limite de digitação;
- ENVASENHA - modo senha;
- ENVDSENHA - sai do modo senha;
- LEENTRADA - lê status da entrada digital;
- GETSERIAL - lê entrada serial secundária;
- ENVDDIG - desliga saída digital;
- ENVDELAY – pausa;
- ENVCLDISP - apaga display;

- CLOSECOM - finaliza comunicação.

As funções acima são da *DLL* de comunicação fornecida pela Colleter, sendo TEDCOM1.DLL e TEDCOM2.DLL, utilizando porta COM 1 e COM 2, respectivamente.

5 SOLUÇÃO DESENVOLVIDA

O entendimento dos processos de produção é requisito básico para o andamento desse projeto. Num mercado cada vez mais competitivo, ter a informação precisa de como anda a produção, bem como estatísticas de eficiência e produtividade, são fundamentais na tomada de decisão.

A idéia da construção de um simulador de microterminal pode ser encarada como um passo importante para futuras implementações, uma vez que a RECH cada vez mais pretende se especializar nesse ramo de *software*: gestão industrial, ou seja, aprimorar-se cada vez mais em manufatura, oferecendo produtos eficientes aos seus clientes.

O simulador, para obter sucesso, deverá proporcionar aos desenvolvedores a simulação do ambiente do cliente, podendo executar os mesmos passos feitos no chão-de-fábrica, sem a necessidade de instalação de equipamentos (microterminal, *hub*, cabeamento e leitor ótico de código de barras) em sua estação de trabalho. No caso de surgimento de uma falha, a mesma pode ser corrigida instantaneamente, causando o mínimo impacto no ambiente do cliente. Além de efetuar correções com mais agilidade, terá disponibilidade de maior tempo para efetivação dos testes no SIGER, uma vez que não precisa efetuar a devolução do equipamento, como ocorre na maioria dos casos. Facilita principalmente a implementação de novas situações ocorridas em apontamentos, atendendo a grande maioria das solicitações efetuadas pelos usuários.

5.1 Proposta de desenvolvimento

O SIGER, com surgimento em 1990, vem evoluindo para novas tecnologias, onde suas metodologias de construção estão em constante aprimoramento, com desenvolvimentos de bibliotecas nativas e padrões de programação. Diante disso, a pretensão de construir um

simulador de microterminal surgiu da idéia de um *software* disponibilizado para uma empresa que fornece equipamentos de PDV, o qual auxilia o setor de pesquisa e desenvolvimento, auxiliando na construção do ERP.

5.1.1 Ferramentas de desenvolvimento

O SIGER atualmente é capaz rodar em qualquer plataforma Windows a partir do W98, sendo que, além de ser preferencialmente instalada a base de dados em servidores Windows, pode também ser instalado em servidores Novell ou Linux. Seu desenvolvimento é feito principalmente pelo Net Express versão 3.0.14 SP1, desenvolvido por Micro Focus Corporation. Alguns recursos extras no sistema são desenvolvidos com o uso do Delphi 7, da Borland. Para o desenvolvimento desse protótipo, será utilizado o Delphi 7 em virtude de ser uma implementação que não faz parte do ERP SIGER, o propósito é de criar uma ferramenta de apoio ao desenvolvimento de novas implementações, auxiliando também no procedimento de testes da empresa, pois a intenção é que o *software* esteja livre de erros quando chegar nas mãos do cliente.

5.1.2 Banco de dados

O banco de dados a ser utilizado por parte do SIGER é nativo do Net Express. Já para o simulador não haverá necessidade de armazenamento: o mesmo será feito de forma temporária, onde um protocolo de comunicação formado por troca de mensagens definirá a forma que SIGER irá se comunicar com o simulador.

5.1.3 Protocolo de comunicação

Esse protocolo de comunicação veio através do estudo das DLL's TEDCOM1.DLL e TEDCOM2.DLL, fornecidas pela Colleter. É chamado de protocolo XON / XOFF, onde o microterminal faz com que a comunicação dos dados seja controlada pelo microcomputador através do envio para o TED do XOFF (13 hexadecimal). Após o envio do XOFF o TED suspende a transmissão até receber um XON (11 hexadecimal). O envio do XOFF pelo microcomputador deve ser antes ou logo após a chegada dos dados enviados pelo TED e nunca durante, pois tal procedimento acarretará um erro no dado interrompido. Caso seja necessário interromper a comunicação enviando XOFF durante um dado enviado, este dado

deverá ser desconsiderado. Desta forma, evita-se que o microcomputador e o TED enviem dados ao mesmo tempo com o risco de perda.

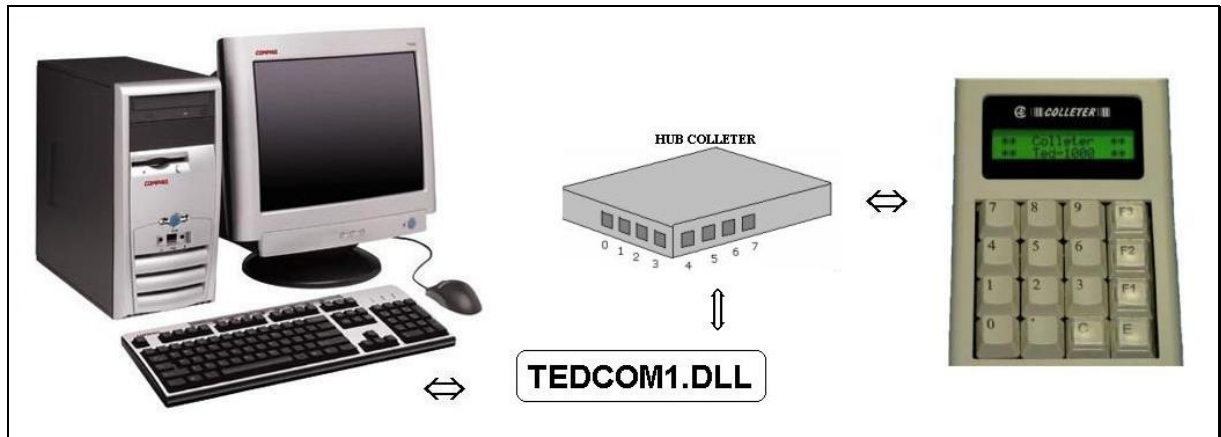


Figura 5-1 – TEDCOM1

Fonte: Do autor

A figura acima indica a forma como funciona a comunicação entre o microterminal e o microcomputador. A TEDCOM1.DLL fica instalada no computador, através dela acontece o protocolo de comunicação (XON / XOFF). Essa DLL é responsável para enviar os dados recebidos do computador para a porta COM 1. Esta se encontra conectada ao *hub*, esse, por conseqüência, está conectada ao microterminal Colleter. Uma vez dado início a comunicação, ou seja, o microcomputador envia um sinal para a DLL que se encarrega de verificar se o microterminal está adequadamente instalado, em seguida responde ao computador que a conexão foi estabelecida. A partir desse ponto, dispondo das funcionalidades disponíveis na DLL, o computador envia mensagens ao display do TED aguardando solicitações ou leituras.



Figura 5-2 – Simulador de microterminal

Fonte: Do autor

A figura acima mostra a idéia projetada para o simulador de microterminal. Diferentemente da figura anterior, nota-se que não existe a DLL, *hub* nem o microterminal. O simulador tem a incumbência de simular as funcionalidades da DLL, *hub* e do microterminal. Assim, o simulador terá uma tela de digitação, simulando um microterminal, onde será possível efetuar a leitura dos códigos de barras bem como solicitar funções. Na mesma tela, visualizará os dados informados bem como os dados retornados pelo microcomputador após uma leitura/solicitação. Como a simulador conterà as funcionalidades constantes na DLL, será possível efetuar as operações com ele. Lembrando que o objetivo é permitir desenvolver novas funcionalidades no SIGER sem o uso dos periféricos, da forma em que a idéia se apresenta, o protótipo resolverá os problemas dos testes das funcionalidades. Como o simulador será construído baseado na DLL, qualquer erro que acontece no ambiente real que não possa ser detectado com o simulador, será um provável erro físico, ou seja, problema de comunicação com o *hub*, microterminal, porta queimada, cabeamento, etc.

A forma de fazer a comunicação entre SIGER e o simulador e o microterminal Colleter é um protocolo baseado no uso de um arquivo em formato texto, de modo síncrono. O SIGER fica esperando o simulador receber a leitura. Quando efetuada, será escrita em formato texto em uma determinada pasta e o simulador passa a esperar a resposta do SIGER. O sistema lê a informação, processa e responde com outro arquivo na mesma pasta. Apaga o arquivo deixado pelo simulador e passa a esperar um novo arquivo em formato texto. Caso essa comunicação não seja eficaz, outra forma será estudada para ser implementada no simulador.

Tabela 1 – Esquema de conexão do microterminal ao *hub*.

Número do microterminal	Porta <i>Hub</i>
1	0
2	1
3	2
4	3
5	4
6	5
7	6
8	7

Fonte: do autor

Inicialmente serão implementadas as seguintes funções:

- Abertura da porta de comunicação;
- Envio de mensagem/texto;
- Envio de “beep” de confirmação de leitura;
- Recebimento de digitação;
- Fechamento da porta de comunicação.

As funções mencionadas acima serão inicialmente implementadas no simulador, a fim de avaliar os resultados obtidos. Sendo satisfatórios, tende-se a futuramente prosseguir com a implementação de novas funcionalidades.

5.2 Modelagem

Como o simulador não necessitará de banco de dados, utilizará uma *string* para cada microterminal conectado ao *hub*, sendo uma para envio e outra de recebimento de dados, denominadas “RITALR00.TXT” e “RITALL00.TXT”, respectivamente. Como o *display* do TED1000 tem apenas 32 caracteres, a *string* não pode ultrapassar esse limite.

Figura 5-3 - Monitor de apontamento por microterminal

Fonte: SIGER

A figura acima exhibe a última leitura efetivada no microterminal ou solicitação de troca de função, sendo possível uma monitoração dos apontamentos realizados fora do ambiente de chão de fábrica.



Figura 5-4 - Monitor de coletores
Fonte: Do autor

A figura 5.4, possui função semelhante ao monitor de apontamento por microterminal do SIGER, porém, irá gravar um log das funções realizadas no simulador.

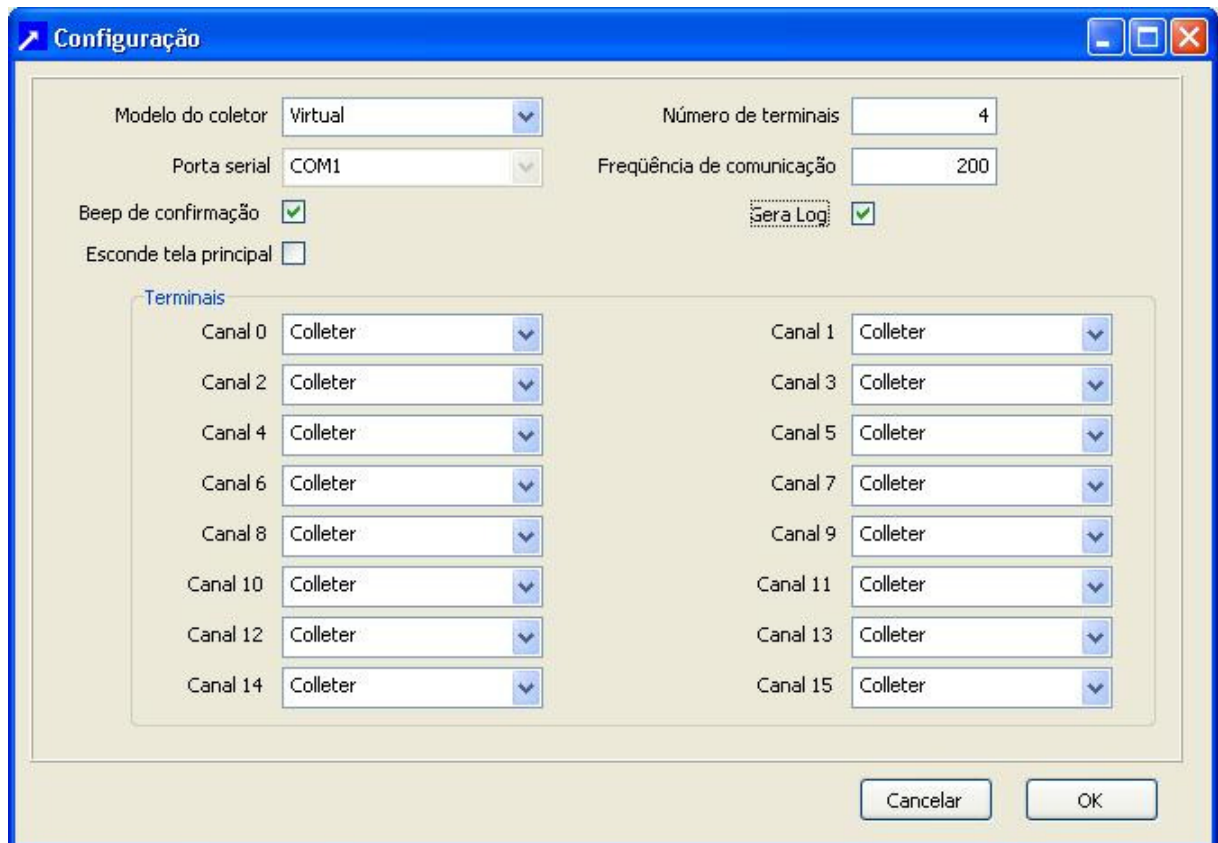


Figura 5-5 - Configuração dos Coletores
Fonte: Do autor

A figura acima representa a janela de configuração dos simuladores, indicando em qual porta os mesmos estão conectados, quantidade de simuladores ativos, frequência de comunicação em milissegundos, se envia um “beep” de leitura recebida, geração de log de remessa e retorno.

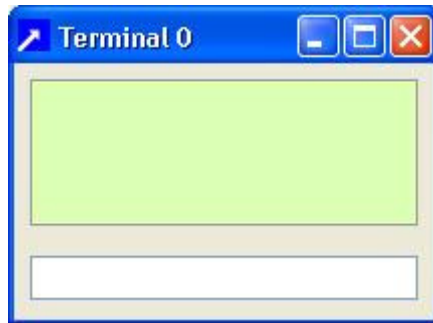


Figura 5-6 - Tela de leitura de barras (Microterminal)

Fonte: Do autor

Por último, a tela de entrada de dados, a qual representa o visor do microterminal TED1000, com duas linhas de 16 caracteres e a *string* de leitura. A *string* é enviada para a tela para ser capturada pelo Siger o qual, após processamento da informação efetuará o retorno da leitura no mesmo *display*.

5.3 Desenvolvimento

O simulador foi desenvolvido na ferramenta proposta no anteprojeto, em Delphi 7, a qual foi disponibilizada pela Rech Informática. Apesar do protótipo não ter banco de dados, funcionando através do protocolo síncrono juntamente com o Siger, algumas tabelas com as quais o software interage diretamente, destacam-se:

ID-IDENTIFICADORES DE PRODUÇÃO
- CODIGO_OPERADOR: Integer
- NOME_OPERADOR: String
- REGISTRO_OPERADOR: Integer
INCLUI
ALTERA
EXCLUI

Figura 5-7 - Diagrama da classe: Identificadores de produção

Fonte: Do autor

O diagrama da classe identificador de produção refere-se ao cadastro de operadores de produção, os quais estão autorizados a efetuar os apontamentos nos microterminais, mediante prévia identificação. Essa identificação pode ser impressa em seu próprio crachá de identificação pessoal.

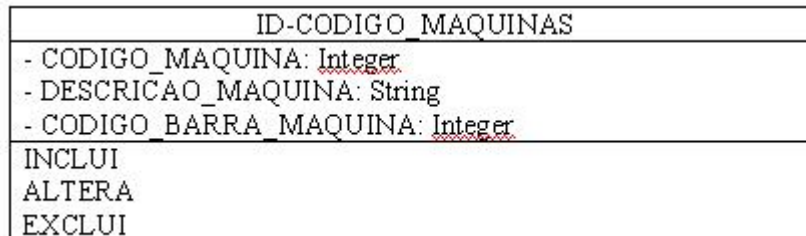


Figura 5-8 - Diagrama da classe: Código das máquinas
Fonte: Do autor

O diagrama acima é necessário para quando em um mesmo microterminal for realizado o apontamento de produção efetiva de recursos ativos diferentes. Assim, é possível avaliar a capacidade individual de cada máquina.

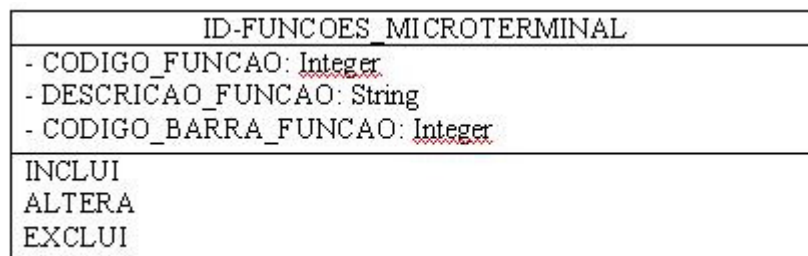


Figura 5-9 - Diagrama da classe: Funções microterminal
Fonte: Do autor

A figura acima mostra as funções que o terminal pode executar: elas são cadastradas e implementadas no SIGER, sendo chamadas pelo simulador a partir da leitura de um código de barras.

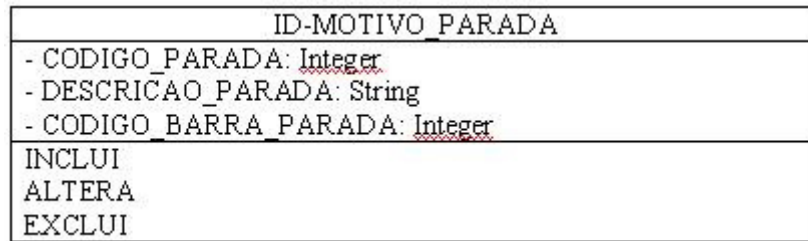


Figura 5-10 - Diagrama da classe: Motivo da parada
Fonte: Do autor

Os motivos das paradas precisam ser registrados, pois interferem diretamente na eficiência do operador / recursos ativos. Também podem ser informadas através da leitura de código de barras, facilitando e agilizando a operação para o identificador de produção. Caso o operador não faça o apontamento da parada, para o sistema de eficiência e produtividade, é como se ele estivesse operando sem interrupção, interferindo nos seus próprios resultados e dando a falsa impressão ao gerente industrial que a fábrica como um todo está produzindo com capacidade máxima.

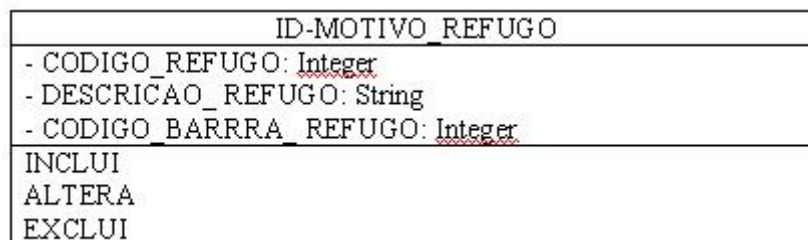


Figura 5-11 - Diagrama da classe: Motivo do refugo
Fonte: Do autor

Da mesma forma como para os motivos de paradas, o sistema procede para com os motivos dos refugos. É muito importante que a peça produzida com alguma anormalidade, seja devidamente analisada para identificar o motivo da não conformidade. Pode ser defeito da matéria prima, de ferramenta mal acoplada na máquina, entre outros. A identificação correta da não conformidade ajuda a manter a qualidade da produção nos demais recursos.



Figura 5-12 - Diagrama da classe: Talão

Fonte: Do autor

O talão é o documento em posse do operador de produção, o qual serve para identificar o que está sendo produzido, em qual quantidade e tamanho. A simples leitura do código de barras deste talão abastece todo o sistema de manufatura, refletindo nos estoques físicos e utilização de recursos, afetando diretamente o MRP.

Além do desenvolvimento do simulador, chamado de RICOLET.EXE, o qual foi feito em Delphi 7, também foram necessárias diversas adaptações no SIGER, o qual acabou ficando com o controle sobre as ações feitas no simulador. Assim, espera-se que seja possível adaptar o simulador para outros modelos de microterminal, sem grandes implementações.

O funcionamento do SIGER e do RICOLET.EXE é de forma dependente, o simulador precisa estar em execução quando o SIGER for acionado, pois o último fará a leitura de quando simuladores estão ativos, enviando uma mensagem para cada um deles indicando que pode ser dado início aos apontamentos de produção.

5.4 Testes de software

Os testes efetuados seguiram as metodologias internas da Rech Informática, os quais se resumem a Verificação, Validação e Testes (VV&T). Pressman (1995, p. 794) justifica a aplicação desta estratégia argumentando que “erros lógicos e pressuposições incorretas são inversamente proporcionais à probabilidade de que um caminho de programa seja executado”.

Vantagens:

O programa pode ser testado por partes;

O programador sabe quais partes do programa são atingidas em determinado teste;

O programador sabe quais partes do programa lidam com que tipos de dados;

O programador conhece os limites internos do programa;

O programador pode criar testes específicos conforme o algoritmo utilizado.

Os testes consistem entradas, todas informadas com códigos de barras, exceto quando se referem a apontamentos de não conformidades, tais como apontamentos com quantidade refugada, onde é necessário que o operador de produção informe quantas peças são boas e quantas possuem defeito.



5.4.1 Fichas com códigos de barras

As fichas abaixo são emitidas pelo SIGER, sendo utilizadas pelo protótipo a fim de simular os apontamentos realizados, criando um ambiente semelhante ao chão de fábrica, permitindo assim aos desenvolvedores da Rech Informática simular as rotinas que acontecem no cotidiano dos clientes.

CODIGOS DAS MAQUINAS		
- 0000001 - MAQ. 01 VERTICAL 3R MARCA EXPRESS	MAQ. 01	Posto:01
 000000100197		
Ou digite 0000001001 e tecle Enter		
- 0000002 - MAQ. 02 VERTICAL 3R MARCA EXPRESS	MAQ. 02	Posto:01
 000000200197		
Ou digite 0000002001 e tecle Enter		
- 0000003 - MAQ. 03 VERTICAL 3R MARCA EXPRESS	MAQ. 03	Posto:01
 000000300197		
Ou digite 0000003001 e tecle Enter		
- 0000004 - MAQ. 04 VERTICAL 3R MARCA EXPRESS	MAQ. 04	Posto:01
 000000400197		
Ou digite 0000004001 e tecle Enter		
- 0000005 - MAQ. 05 VERTICAL 3R MARCA EXPRESS	MAQ. 05	Posto:01
 000000500197		
Ou digite 0000005001 e tecle Enter		






Gerado por SIGER - Rech Informática - www.rech.com.br

Figura 5-13 - Códigos das máquinas
Fonte: SIGER

FICHA DE FUNÇÕES DO MICROTERMINAL (COLETOR)	
- Função	INICIO DE PRODUÇÃO
	 0001009999
	Ou Digite .1 e tecla Enter
- Função	PRODUÇÃO INTEGRAL/FIM DE PRODUÇÃO
	 0002009999
	Ou Digite .2 e tecla Enter
- Função	PRODUÇÃO PARCIAL/EXCEDENTE
	 0003009999
	Ou Digite .3 e tecla Enter
- Função	IDENTIFICAÇÃO DE OPERADOR
	 0004009999
	Ou Digite .4 e tecla Enter
- Função	INICIO DE PARADA
	 0005009999
	Ou Digite .5 e tecla Enter
- Função	FIM DE PARADA
	 0006009999
	Ou Digite .6 e tecla Enter
- Função	PRODUÇÃO INTEGRAL/PARCIAL COM REFUGO
	 0007009999
	Ou Digite .7 e tecla Enter

Gerado por SIGER - Ruck Informática - www.ruck.com.br

Figura 5-14 - Ficha de funções do microterminal
Fonte: SIGER

FICHA DE MOTIVOS DE REFUGO	
0005	BOLHAS
	 000500000996
	Ou digite 0005 e tecla Enter
0001	DEFEITO GERAL
	 000100000996
	Ou digite 0001 e tecla Enter
0004	FORA DA DENSIDADE
	 000400000996
	Ou digite 0004 e tecla Enter
0002	FORA DE COR
	 000200000996
	Ou digite 0002 e tecla Enter
0003	QUEIMADO
	 000300000996
	Ou digite 0003 e tecla Enter

Gerado por SIGER - Ruck Informática - www.ruck.com.br

Figura 5-15 - Ficha de motivo de refugo
Fonte: SIGER






FICHA DE MOTIVOS DE PARADA	
0015	AGUARDAR TEMPERATURA AGU  001500000098
Ou digite 0015 e tecla Enter	
0007	CONSERTO MATRIZ CON  000700000098
Ou digite 0007 e tecla Enter	
0004	FALTA DE LUZ FAL  000400000098
Ou digite 0004 e tecla Enter	
0003	FALTA DE OPERADOR FAL  000300000098
Ou digite 0003 e tecla Enter	
0001	FALTA DE PEDIDO FAL  000100000098
Ou digite 0001 e tecla Enter	

Figura 5-16 - Ficha de motivo de parada
Fonte: SIGER

5.4.2 Apontamento normal

O teste a seguir mostra a seqüência gerada na tela de 32 caracteres do simulador de microterminal ao efetuar um apontamento de um talão de produção, mais precisamente de um SSSC, passo a passo, exibindo na tela as mesmas informações que apareceriam num microterminal Colleter. O resultado foi positivo, pois o apontamento de produção foi concretizado pelo SIGER. Este apontamento consiste apenas na leitura de um código de barras, sem informação de operador de produção, máquina, parada ou refugio. É a operação mais comum de ser realizada, utilizada nos casos em que a eficiência pessoal ou de uma determinada máquina não precise de análise individual, importando o rendimento global da fábrica. Neste caso, quando o SIGER for executado e entrar na opção de monitoração de microterminal, emitirá uma mensagem para os simuladores indicando que está disponível o início das leituras dos talões de produção, sem necessidade de informar operadores e máquinas.

TCC II - RAMON MARCELO RECH	
SSCC / Código de Série da Unidade Logística 078904800000000026	
GTIN / Nro Global Item 07899344965792	Batch / Lote
COUNT / Qtd 50	Descrição do Produto SOLA TCC 3.0 PRETO
CLIENTE: NOME DO CLIENTE SSSC GOL	
OS:	LARG.: ENT.: 22/09/2009
FAB.:	1
PED.: 2	OC:



(01)07899344965792



(00)078904800000000026

Figura 5-17 - Talão de produção – SSCC
Fonte: SIGER

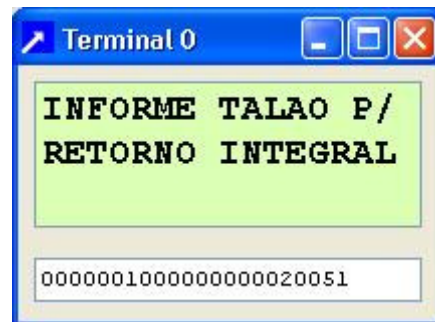


Figura 5-18 - Monitor de microterminal – status
Fonte: Do autor

The screenshot shows a software window titled "Apontamento por microterminal". It contains several data entry fields and summary statistics:

- Fábrica:** [Empty field]
- Tipo de coletor:** 2-Colleter
- Terminal:** 0
- Hora verificação:** [Empty field]
- Ordem de produção:** 1
- Quantidade total da OP:** 1.000,000
- Total aprovado:** 50,000
- Total produzido:** 50,000
- Total refugado:** [Empty field]
- Número do talão:** 2
- Centro de custo:** 51 FABRICA INTERNA - TR
- Produto produzido:** 9025565 SOLA TCC
- Ref.:** TCC
- Data e hora inicial:** 20/10/2009 Ter 19:07
- Linha do arquivo:** 0000001000000000020051 00000
- Status do coletor:** RETORNO OK!

At the bottom, there are two buttons: "Op's em processo" and "Cancelar".

Figura 5-19 - Apontamento por microterminal - status SIGER
Fonte: SIGER



Figura 5-20 - Monitor de microterminal – Status
Fonte: Do autor

5.4.3 Apontamento de parada

O apontamento de parada de produção já é um pouco mais criterioso, exigindo certa atenção pelo operador durante o procedimento de apontamento. Seguindo os passos conforme forem solicitados no *display* do simulador, não devem ocorrer problemas quanto ao processamento dos dados pelo SIGER. Enquanto o operador não enviar uma informação válida conforme a solicitação, uma mensagem de código inválido será enviada ao *display*, até que seja recebida uma informação válida. A única forma de interromper esse procedimento, é com a leitura de um novo código de barras da ficha de funções do microterminal. Vejamos passo a passo a seguir:

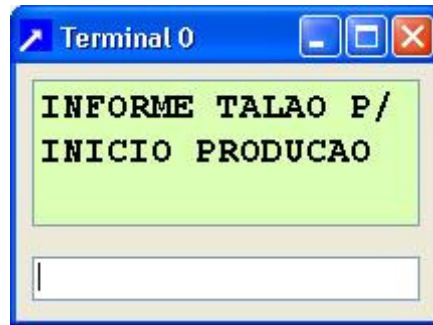


Figura 5-21 - Monitor de microterminal – Status
Fonte: Do Autor

Primeiro passo: SIGER solicitou o apontamento de início de produção.

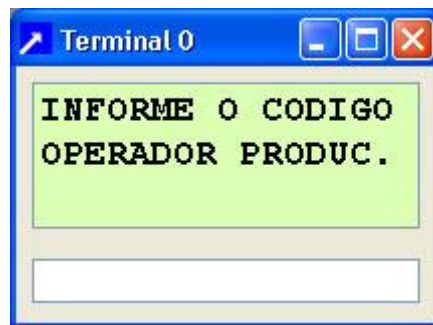


Figura 5-22 - Monitor de microterminal – Status
Fonte: Do Autor

Segundo passo: Após ter lido o código de barras indicando que deseja informar o início de uma parada, o SIGER envia ao simulador uma mensagem indicando para informar o código do operador de produção.

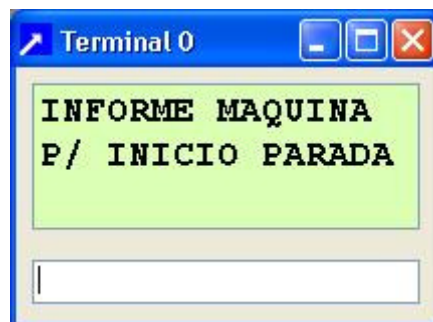


Figura 5-23 - Monitor de microterminal – Status
Fonte: Do Autor

Terceiro passo: Após a informação do código do operador, a próxima informação a ser solicitada é o código da máquina para o registro de início de parada.

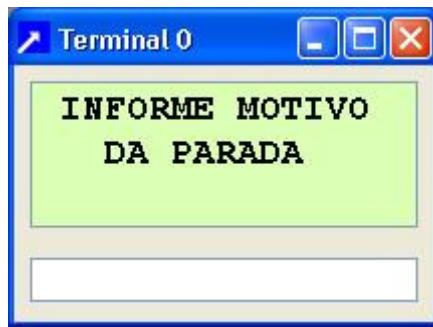


Figura 5-24 - Monitor de microterminal – Status
Fonte: Do Autor

Quarto passo: Já com a máquina devidamente informada e validada pelo SIGER, finalmente vem a solicitação do motivo da parada. Após essa ser validada, a próxima mensagem que aparecerá na tela do simulador é de que o retorno foi OK, mensagem adotada como padrão quando a seqüência de apontamento obteve sucesso.

5.4.4 Apontamento com refugio

Na rotina de apontamento de produção com quantidade refugada exige-se ainda mais cautela por parte do operador. A tela inicial é a mesma das demais, a qual solicita o talão para efetivar o apontamento de produção. Enquanto o operador não enviar uma informação válida conforme a solicitação, uma mensagem de código inválido será enviada ao *display*, até que seja recebida uma informação válida. A única forma de interromper esse procedimento é com a leitura de um novo código de barras da ficha de funções do microterminal. Porém, esse tipo de apontamento possui um número maior de passos a serem realizados, vejamos a seguir:

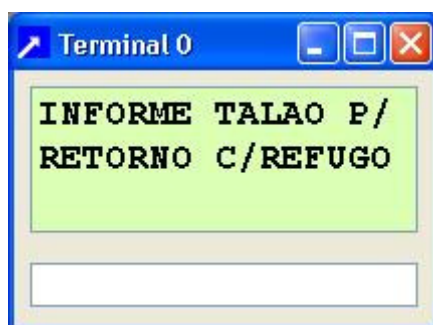


Figura 5-25 - Monitor de microterminal – Status
Fonte: Do Autor

Primeiro passo: SIGER solicitou o apontamento de início de produção. Nesse ponto, o operador indicou que quer fazer apontamento com quantidade refugada. SIGER enviou mensagem ao simulador indicando para informar qual o número do talão de produção, o qual é informado pela leitura do código de barras.

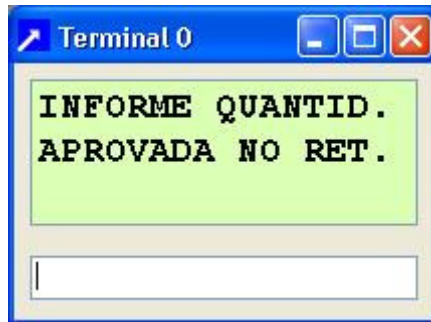


Figura 5-26 - Monitor de microterminal – Status
Fonte: Do Autor

Segundo passo: Após a validação do código de barras do talão informado, a mensagem enviada para o simulador é a de informar a quantidade aprovada para este talão, ou seja, a quantidade de peças boas obtidas durante o processo produtivo.

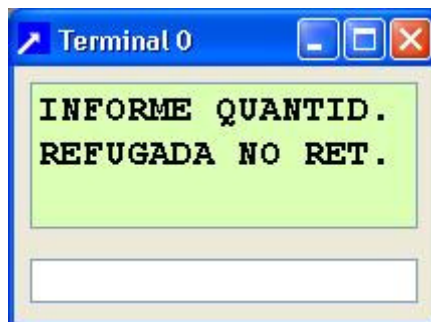


Figura 5-27 - Monitor de microterminal – Status
Fonte: Do Autor

Terceiro passo: Depois de ter recebido a quantidade de peças boas, que inclusive pode ser igual a zero, a informação enviada ao simulador é a de informar a quantidade de peças não conformes, lembrando que SIGER precisa guardar todas as informações em memória até que o procedimento de apontamento seja concluído. A simulação foi feita usando o simulador para apenas um microterminal, mas poderia haver 16 ativos, sendo necessário gerenciar a informação de todos ao mesmo tempo, com operações diferentes em cada um deles.

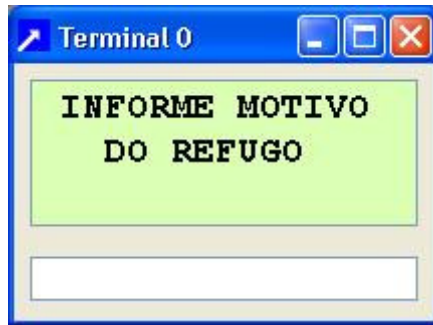


Figura 5-28 - Monitor de microterminal – Status
Fonte: Do Autor

Quarto passo: Recebidas as quantidades aprovada e refugada, peças boas e peças com não conformidades, respectivamente, finalmente o usuário pode informar o motivo do refugo, ou seja, indicar a causa que pela qual nem todas as peças produzidas estão aptas para serem comercializadas.

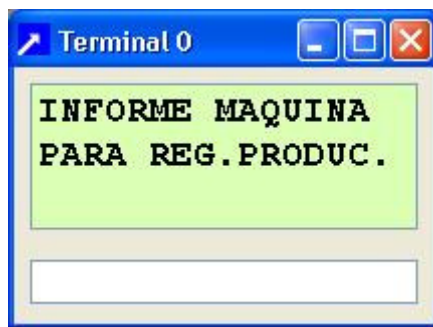


Figura 5-29 - Monitor de microterminal – Status
Fonte: Do Autor

Quinto passo: Caso esteja configurado no sistema que controle máquinas no apontamento de produção, a última informação solicitada é justamente o código da máquina, também informado através da leitura do código de barras. Se após todos esses passos o SIGER validar todas as informações e conseguir realizar todos os registros necessários, a mensagem exibida no simulador será a de que o apontamento foi realizado com sucesso, caso contrário, o usuário deverá repetir todos os passos novamente.

5.5 Considerações

Durante a fase de testes do simulador, foi possível detectar o quanto trabalhoso pode ser um simples *report* de produção. No último exemplo, o qual é o mais complexo

apresentado, o operador de produção precisa completar uma seqüência de movimentos até concretizar o apontamento de produção. Esse movimento poderia ser ainda maior, nos casos em que haja um subproduto no processo, o mesmo também devendo também ser identificado ao sistema de gestão. Não é o caso do simulador, já que esta operação não foi inclusa neste projeto.

6 RESULTADOS OBTIDOS

Os resultados obtidos com a implementação do simulador foram satisfatórios aos propósitos deste projeto. Foi possível reproduzir a maioria das situações que ocorrem diretamente com o uso dos equipamentos (*hub*, microterminal, leitor óptico). Da forma que foi estruturada a integração entre simulador e SIGER, caso haja um erro no apontamento de produção, pode ser facilmente detectado. Caso o problema não seja no software, a causa mais provável passa a ser no equipamento que está no chão de fábrica.

6.1 Facilidade nas implementações

Com o auxílio do simulador, passa a ser muito mais fácil para a equipe de desenvolvimento implementar novas funções no SIGER, bem como fazer testes nas atuais quando ocorre o relato de um erro por parte de algum cliente. Como o SIGER é totalmente integrado, um apontamento de produção gera reflexo em toda a gestão, abrangendo controle de estoque, compras, contabilidade, vendas e logística. Com a quantidade de implementações diárias no SIGER, torna-se necessário testar todas as funcionalidades do mesmo de tempos em tempos. A rotina de apontamento por microterminal era um problema, pois sem o equipamento tornava-se difícil uma execução dos casos de teste. Com o simulador, esse problema desapareceu pois, facilmente podem-se efetuar todos os testes com exatidão.

Além de auxiliar a equipe de desenvolvimento em novas implementações, o simulador servirá de ferramenta de apoio ao GRM, pois quando o atendimento apontava para o microterminal, sempre era necessário chamar uma pessoa do desenvolvimento para acompanhar o atendimento ao cliente, para abrir o código fonte e tentar identificar a causa do problema. Agora, com a ajuda do simulador, o atendente poderá fazer uma análise prévia do problema relatado sem a necessidade de chamar um técnico do desenvolvimento.

6.2 Ações não atendidas

São poucas as ações não atendidas, uma delas é o fato de o simulador conseguir efetuar apenas uma entrada por vez. Acabou não sendo previsto nessa situação, o simulador fica rodando residente em um microcomputador, podendo-se abrir 16 simuladores na tela, mas somente é possível ler um código de barras por microterminal a cada vez, ou seja, não se pode dar duas ou mais entradas ao mesmo tempo para ver a reação do SIGER e do simulador com quantidade significativa de dados em transição.

Será necessário um estudo para contornar esse problema, podendo ser possível realizar leituras no simulador em diversos microcomputadores, uma única central fazendo o tratamento das informações recebidas e efetuando a resposta aos pontos de origem.

CONCLUSÃO

As empresas procuram soluções em sistemas de informação para gestão empresarial. Necessitam de ferramentas simples e eficazes que venham a agilizar a tomada de decisões.

A coleta de dados em chão de fábrica necessita de agilidade e precisão, pois a linha de produção é o processo principal de uma indústria, e se tratando de processo principal, esse deve estar sempre bem alinhado, com o mínimo de falhas possíveis, pois qualquer problema afeta a gestão como um todo.

O objetivo desse trabalho foi atingido. A solução desenvolvida atingiu as metas do projeto, e foi feito de uma forma para, futuramente, agregar outros modelos de microterminais, ampliando sua área de apoio.

A maior parte do processamento era para ficar por conta do simulador de microterminal, mas acabou ficando no SIGER por uma questão de centralização de código. Essa idéia acabou prejudicando um pouco as perspectivas iniciais do projeto, que eram de fazer com que o simulador se encarregasse de manter o controle das ações de todos os terminais ativos, efetuando o gerenciamento e repassando os dados para o SIGER, aguardando a resposta de sucesso ou não da operação realizada.

Se o simulador tivesse ficado com a autonomia de gerenciamento dos dados, seria possível efetuar apontamento de múltiplos terminais simultaneamente. Essa solução demandaria um tempo bem maior de desenvolvimento do protótipo, mas seria a solução de uma das ações não atendidas pelo projeto.

Da forma em que foi feito, o simulador recebe a informação que pode iniciar a leitura, repassa os dados de entrada e aguarda o ERP processar a solicitação. Sendo assim, grande parte do desenvolvimento ficou encapsulada no SIGER o que, dependendo do ponto

de vista não é ruim, pois para agregar outro microterminal ficou mais fácil, não será necessário todo o controle dentro do simulador. No SIGER, as funções permanecerão inalteradas.

A idéia inicial era de poder até comercializar esse simulador, mas a empresa optou por mantê-lo como ferramenta de desenvolvimento e testes.

Um ponto que não havia sido percebido no projeto, referenciando aos testes práticos é que, antes do simulador, somente o pessoal da área de desenvolvimento conseguia fazer testes sem o equipamento (*hub*, microterminal, leitor óptico), pois tinham consigo a ferramenta de desenvolvimento para depurar o programa, conseguindo assim, efetuar alguns testes sem o equipamento e mãos. Lembrando que esses testes levavam horas para serem realizados e não podiam ter certeza que foram corretamente executados.

Agora a equipe de campo também consegue realizar os testes, inclusive diretamente no cliente. O simulador não precisa de instalação, é um executável. Basta copiá-lo na mesma pasta de execução do SIGER que pode simular os apontamentos diretamente no cliente, identificando na origem se é problema de *software* ou *hardware*.

REFERÊNCIAS BIBLIOGRÁFICAS

BEMATECH. **Acesso a informações sobre a empresa.** Disponível em <<http://www.bematech.com.br/sobre/default.asp>>. Acesso em 23/03/2009.

BOARETTO, Neury; KOVALESKI, João Luiz; SCANDELARI, Luciano. **Coleta de dados e monitoramento de chão-de-fábrica na manufatura discreta – integração com as ferramentas de gestão.** Bauru: 2004. XI SIMPEP

CDL NOVO HAMBURGO. **Revista Ação Logista.** 04. ed. Novo Hamburgo: Gráfica Palloti, 2009.

COLLETER. **Acesso a informações sobre a empresa.** Disponível em <<http://www.colleter.com.br>>. Acesso em 23/03/2009.

GS1. **Acesso a informações sobre a empresa.** Disponível em < <http://www.gs1brasil.org.br> >. Acesso em 03/11/2009.

JUNIOR, Eraldo Garcia; DIAS, Guilherme Alfredo Dentzien; TELLÓ Marcos. **Ensaio e Certificação de instalações Elétricas em Automação de Prédios Inteligentes: I COBRAPI.** Disponível em: <<http://www.ee.pucrs.br/~gcem/imagens/cobrap.pdf>> Acesso em: 12 nov. 2007.

LAUDON, Kenneth C. (2004). *Sistemas de Informações gerenciais : administrando a empresa digital.* São Paulo: Editora:Prentice hall, p61.

MORAES, Cícero Couto de; CASTRUCCI, Plínio de Lauro. **Engenharia de Automação Industrial.** Rio de Janeiro: LTC, 2001. p295.

PRESSMAN, R.S. (2005). **Software Engineering: A Practitioner's Approach,** volume 6.

PRESSMAN, R. S. **Engenharia de software.** São Paulo: Makron Books. 1995, 724 p.

PRESSMAN, R. S. **Engenharia de software.** São Paulo: Makron Books. 1995, 794 p.

PRODANOV, Cleber Cristiano. **Manual de metodologia científica.** 3º Ed. Novo Hamburgo: Feevale, 2003

RECH INFORMÁTICA LTDA. **MI-SIGER- Metodologia de Implantação do SIGER.** 01/2009 Novo Hamburgo, 2009.

RECH INFORMÁTICA LTDA. **MT - Metodologia de Treinamento Rech Informática Ltda.** 01/2009 Novo Hamburgo, 2009.

RECH INFORMÁTICA LTDA. **Teste de software - Procedimento de testes Rech.** 01/2009 Novo Hamburgo, 2009.

Revista Ação lojista, maio 2009 Ano III edição 04, Câmara de Dirigentes Lojistas de Novo Hamburgo.

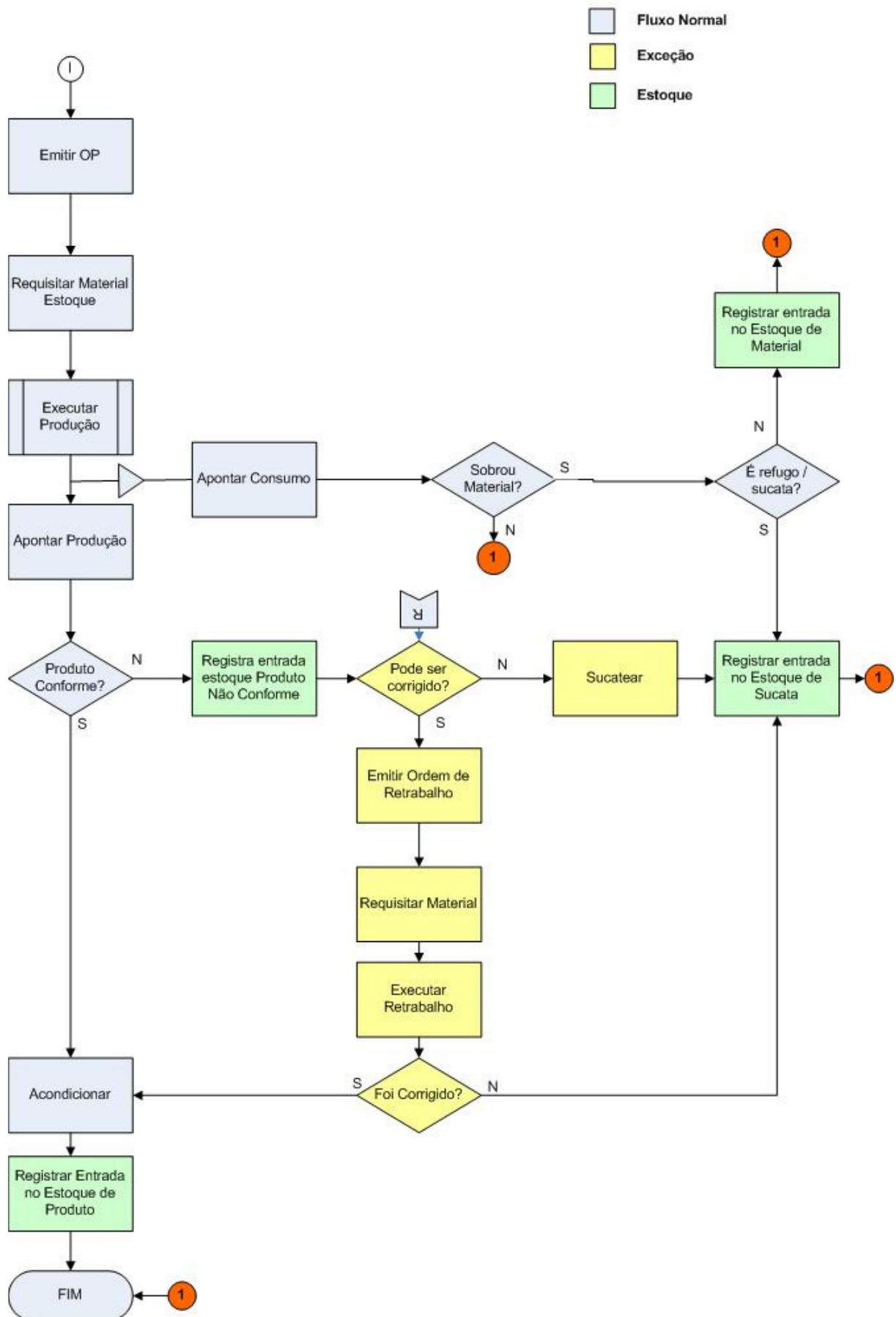
SOUZA, Alessandro José de. **Sistema de Gerência de Informação de Processos Industriais via WEB.** Natal: 2005. 68p. Dissertação (Pós Graduação em Engenharia Elétrica) – Centro de Tecnologia, Universidade Federal do Rio Grande do Norte, 2005.

WEBER, Kival Chaves; ROCHA, Ana Regina Cavalcanti da; NASCIMENTO, Célia Joseli do (Org.). **Qualidade e Produtividade em Software.** 4. ed. São Paulo: Makron, 2001. 188 p.

ZACCARELLI, Sérgio Batista. **Programação e Controle da Produção.** 4ª ed. São Paulo: Pioneira, 1976. 292p.

ANEXO A – FLUXO DE PRODUÇÃO DE ELEMENTOS - SIGER

FLUXO DE PRODUÇÃO DE ELEMENTOS



**ANEXO B – FLUXO DE PRODUÇÃO DE COMPONENTES /
CONJUNTOS - SIGER**

FLUXO DE PRODUÇÃO DE COMPONENTES / CONJUNTOS

