

Luis Holanda Gomes Sobrinho

**Protótipo de Sistema de Informação para
Gerenciamento de Transações de Estoques em
Companhias de Saneamento Rural**

Picos - PI
5 de junho de 2017

Luis Holanda Gomes Sobrinho

Protótipo de Sistema de Informação para Gerenciamento de Transações de Estoques em Companhias de Saneamento Rural

Trabalho de Conclusão de Curso apresentado como exigência para obtenção de Bacharelado em Sistemas de Informação na Universidade Federal do Piauí, sob orientação do professor Leonardo Pereira de Sousa e coorientação do professor Márcio Alves de Macêdo.

Universidade Federal do Piauí
Campus Senador Helvídio Nunes de Barros
Bacharelado em Sistemas de Informação

Picos - PI
5 de junho de 2017

FICHA CATALOGRÁFICA

Serviço de Processamento Técnico da Universidade Federal do Piauí
Biblioteca José Albano de Macêdo

G633p Gomes Sobrinho, Luis Holanda

Protótipo de Sistemas de Informação para gerenciamento de transações de estoques em companhias de saneamento rural / Luis Holanda Gomes Sobrinho.– 2017.

CD-ROM : il. ; 4 ¾ pol. (60 f.)

Trabalho de Conclusão de Curso (Curso Bacharelado em Sistemas de Informação) – Universidade Federal do Piauí, Picos, 2017.

Orientador(A): Prof. Esp. Leonardo Pereira de Sousa

1. Engenharia de *Softwares*. 2. Controle de Estoques
3. Sistemas de Informação. I. Título.

CDD 005.1

SISTEMA DE INFORMAÇÃO PARA O GERENCIAMENTO DE TRANSAÇÕES DE
ESTOQUE EM COMPANHIA DE SANEAMENTO RURAL

LUIS HOLANDA GOMES SOBRINHO

Monografia aprovada como exigência parcial para obtenção do grau de
Bacharel em Sistemas de Informação.

Data de Aprovação

Picos – PI, 23 de junho de 2017


Prof. Esp. Leonardo Pereira de Sousa
Orientador


Prof. Esp. Ivenilton Alexandre de Souza Moura
Membro


Prof. Esp. Francisco das Chagas Imperes Filho
Membro

Agradecimentos

Agradeço a minha família, em especial aos meus pais, Josefa Holanda Gomes Silva e Valdir Gonçalves da Silva, pelo apoio e confiança depositados em mim.

Ao meu coorientador Márcio Alves de Macêdo e ao meu orientador Leonardo Pereira de Sousa, pela paciência, pela disposição em sanar as dúvidas e pelos conhecimentos compartilhados, durante o desenvolvimento deste trabalho.

A todos os meus professores do curso Sistemas de Informação, pelo conhecimento passado a mim, e que foram essenciais para a realização deste trabalho.

E a todos os colegas e amigos que me apoiaram e que de alguma forma compartilharam experiências, importantes para a realização deste trabalho ou para minha vida profissional.

Comece de onde você está. Use o que você tiver. Faça o que você puder.

Arthur Ashe

Resumo

Os sistemas de informação(SIs) estão cada vez mais presentes no mundo empresarial, as empresa os usam para controlar e/ou para obter informações sobre suas operações de forma rápida e dinâmica, que às auxiliam na implementação de estratégias competitivas. Tendo isso em mente, o objetivo deste trabalho é desenvolver um SI para controlar as transações de estoques, realizadas em uma empresa que atua na área de saneamento rural em vários municípios da região de Picos-PI. Além disso, O sistema deverá gerar informações que poderão auxiliar a empresa na tomada de decisões, tais como: quais itens do estoque precisam de reposição, quais os itens mais usados, quais municípios necessitam de maior atenção, etc. No decorrer do desenvolvimento do sistema foram feitas suscetivas reuniões com membros da empresa, conforme recomendação do modelo incremental da engenharia de software. A linguagem de programação adotada para o desenvolvimento do sistema foi o Java, devido a sua característica de ser multiplataforma e o servidor de banco de dados foi desenvolvido usando o SGBD MySQL. Após finalizado o desenvolvimento da primeira versão do sistema, o mesmo foi testado pelos usuários, a fim de comprovar sua eficácia e/ou corrigir erros encontrados durante o uso.

Palavras-chaves: Sistemas de Informação. Engenharia de software. Controle de estoques.

Abstract

Information systems (ISs) are increasingly present in the business world, companies use them to control and / or obtain information about their operations in a fast and dynamic way, which helps them to implement competitive strategies. With this in mind, the objective of this work is to develop an IS to control the inventory transactions carried out in a company that operates in the rural sanitation sector in several municipalities in the Picos-PI region. In addition, the system should generate information that can assist the company in making decisions, such as: what items in stock need replacement, which items are most used, which municipalities need more attention, etc. During the development of the system, meetings were held with members of the company, as recommended by the incremental model of software engineering. The programming language adopted for the development of the system was Java, due to its multiplatform characteristic and the database server was developed using the MySQL DBMS. After the development of the first version of the system was finished, it was tested by the Users to verify their effectiveness and / or correct errors encountered during use.

keywords: Information Systems. Software Engineering. Inventory control.

Lista de ilustrações

Figura 1 – Representação gráfica da logística da SISAR/PI	14
Figura 2 – Exemplo de diagrama de caso de uso UML.	21
Figura 3 – Representação de classe no padrão UML.	22
Figura 4 – Diagrama de caso de uso.	28
Figura 5 – Diagrama de classe do sistema.	30
Figura 6 – Tela de login no sistema	32
Figura 7 – Layout principal	33
Figura 8 – Tela de exibição de estoque.	35
Figura 9 – Lista de saídas de um estoque.	35
Figura 10 – Filtro das saídas de um estoque.	35
Figura 11 – Conteúdo lateral da tela dos estoques.	36
Figura 12 – Tela de transferência de itens dos estoques.	37
Figura 13 – Tela de aviso para o usuário.	37
Figura 14 – Importar estoque e entradas nas comunidades	38
Figura 15 – Modelo de arquivo CSV.	39
Figura 16 – Tela de barra de progresso.	39
Figura 17 – Formulário para cadastro ou edição de material ou equipamento.	40
Figura 18 – Relatório de demonstração de entradas dos municípios.	41
Figura 19 – Gráfico de avaliação de satisfação do usuário.	43
Figura 20 – Diagrama entidade relacionamento do banco de dados.	58

Lista de tabelas

Tabela 1 – Requisitos funcionais do sistema.	26
Tabela 2 – Requisitos não-funcionais do sistema.	27
Tabela 3 – Regras de negócio do Sistema	27
Tabela 4 – Lista de menus e submenus da barra de menus.	34
Tabela 5 – Documentação do caso de uso Cadastrar Materiais e Equipamentos.	51
Tabela 6 – Documentação do caso de uso Adicionar Itens ao Estoque.	51
Tabela 7 – Documentação do caso de uso Mostrar Itens do Estoque.	52
Tabela 8 – Documentação do caso de uso Cadastrar Usuário.	52
Tabela 9 – Documentação do caso de uso Mostrar Itens do Estoque.	53
Tabela 10 – Documentação do caso de uso Transferir Itens.	53
Tabela 11 – Documentação do caso de uso Atualizar Itens do Estoque.	54
Tabela 12 – Documentação do caso de uso Exportar Tabela.	54
Tabela 13 – Documentação do caso de uso Mostrar Relatórios.	55
Tabela 14 – Documentação do caso de uso Cadastrar Fornecedores.	56

Lista de abreviaturas e siglas

Casan	Companhia Catarinense de Água e Saneamento
Cagece	Companhia de Água e Esgoto do Ceará
Copasa	Companhia de Saneamento de Minas Gerais
CSV	Comma-Separated Values
CIT SISAR-PI	Controle Integrado de Transações SISAR-PI
DER	Diagrama Entidade Relacionamento
DSDM	Dynamic Systems Development Method
Funasa	Fundação Nacional de Saúde
IBGE	Instituto Brasileiro de Geografia e Estatística
PNAD	Pesquisa Nacional por Amostra de Domicílios
Sabesp	Companhia de Saneamento Básico do Estado de São Paulo
SGBD	Sistema Gerenciador de Banco de Dados
SI	Sistemas de Informação
SISAR	Sistema Integrado de Saneamento Rural
SISAR-PI	Sistema Integrado de Saneamento Rural do Piauí
SPT	Sistemas Processamento de Transações
TI	Tecnologia da Informação
XML	Extensible Markup Language
XP	Extreme Programming
UML	Unified Modeling Language

Sumário

1	Introdução	13
1.1	Objetivos	14
1.1.1	Objetivos Específicos	14
2	Referencial Teórico	16
2.1	Sistemas de Informação	16
2.1.1	Tipos de Sistemas de Informação	16
2.1.1.1	Sistemas de Automação	16
2.1.1.2	Sistemas Processamento de Transações (SPT)	16
2.1.1.3	Sistemas de Informações Gerenciais	17
2.1.2	Ciclo de Vida dos Sistemas de Informações	17
2.2	Engenharia de Software	17
2.2.1	Modelos de Processos de Software	18
2.2.1.1	Desenvolvimento em Cascata	18
2.2.1.2	Desenvolvimento Incremental	19
2.2.2	Desenvolvimento Ágil	19
2.2.2.1	Extreme Programming (XP)	19
2.2.2.2	Dynamic Systems Development Method (DSDM)	20
2.2.2.3	Scrum	20
2.3	Levantamento de Requisitos	20
2.4	UML	21
2.4.1	Diagrama de Casos de Uso	21
2.4.2	Diagrama de Classe	22
2.5	Linguagem Java	22
2.5.1	Integrated Development Environments (IDEs)	23
2.5.2	Modelo MVC (Model-View-Controller)	23
2.5.3	JavaFX	23
2.6	Banco de Dados MySQL	24
3	Desenvolvimento do Sistema	25
3.1	Levantamento dos Requisitos	25
3.1.1	Regras de Negócios	25
3.2	Diagramas UML	27
3.2.1	Diagrama de Casos de Uso	28
3.2.2	Diagrama de Classe	28
3.3	Projeto do Banco de Dados	29

3.3.1	Mini-mundo	29
3.4	Implementação do Sistema	31
4	Usabilidade do Sistema	32
4.1	Login no Sistema	32
4.2	Layout principal	33
4.3	Estoques	34
4.4	Transferências	36
4.5	Exportação e Importação de Tabelas	38
4.6	Formulários	40
4.7	Relatórios	40
4.8	Backup	41
5	Avaliação do Sistema	42
5.1	Adaptabilidade	42
5.2	Desempenho	42
5.3	Uso do Sistema	42
6	Conclusão	45
6.1	Trabalhos Futuros	45
	Referências	47
	Apêndices	50
	APÊNDICE A Documentação dos Casos de Uso	51
	APÊNDICE B Projeto Conceitual do Banco de Dados	57
	APÊNDICE C Diagrama Entidade Relacionamento	58
	APÊNDICE D Questionário de Avaliação do Sistema	59

1 Introdução

Segundo Censo Demográfico realizado pelo Instituto Brasileiro de Geografia e Estatística IBGE/2010, há cerca de 8,1 milhões de domicílios rurais no Brasil e de acordo com dados da Pesquisa Nacional por Amostra de Domicílios PNAD/2014, apenas 34,5% destes lares estão ligados a rede de abastecimento de água com ou sem canalização interna (FUNASA, 2015). Ainda de acordo com a PNAD quando se trata de esgoto sanitário, apenas 5,45% dos domicílios estão ligados a rede de coleta de esgotos.

No Brasil, há várias companhias de tratamento de águas e esgotos, como a Companhia de Saneamento Básico do Estado de São Paulo (SABESP), Companhia de Saneamento de Minas Gerais (COPASA), Companhia Catarinense de Água e Saneamento (CASAN), Companhia de Água e Esgoto do Ceará (CAGECE), etc (AESBE, 2011).

A maioria dos sistemas de águas rurais executados pela CAGECE são administrados pelo Sistema Integrado de Saneamento Rural (SISAR). O Sisar é uma divisão da CAGECE que atua especificamente em comunidades da zona rural e visa garantir, a longo prazo, o desenvolvimento e manutenção dos sistemas implantados pela Companhia de forma autossustentável (CAGECE, 2011).

O SISAR foi criado pela Cagece, em 1996, em parceria com o banco alemão KFW (CAGECE, 2011). A unidade do Piauí foi constituída em 17 de abril de 2004 (RIACHAONET, 2012) e atualmente, segundo a Sisar/PI, possui 42 comunidades rurais e sedes municipais filiadas, na microrregião de Picos-PI.

Cada comunidade dessas possui um estoque onde é armazenado materiais e equipamentos usados para manutenção dos sistemas de saneamento. Toda nova aquisição de equipamento entra em um estoque central, só então será distribuído para as comunidades. O sistema implementado neste trabalho foi feito tendo como base logística do Sistema Integrado de Saneamento Rural da região de Picos-PI (SISAR/PI). A Figura 1 mostra esta logística da SISAR/PI representada através de um grafo.

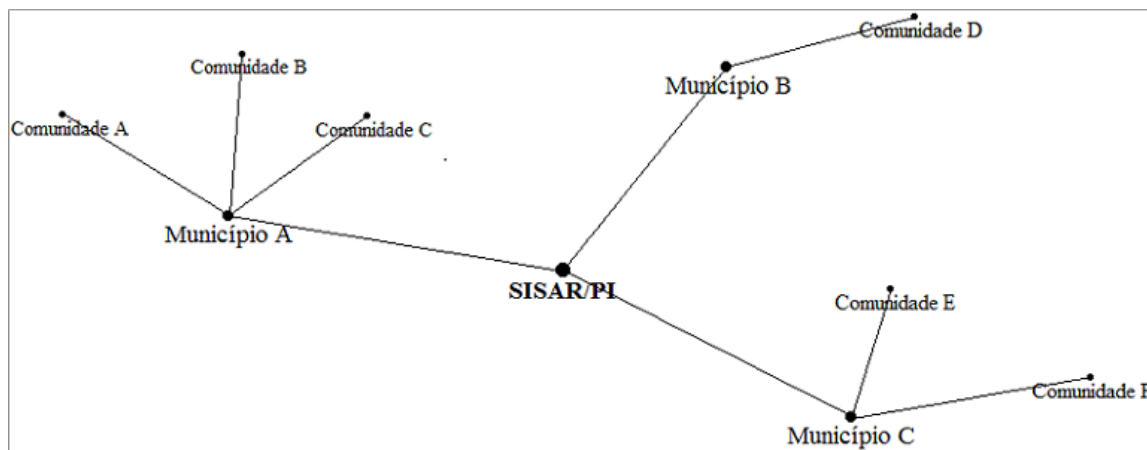
Mensalmente a SISAR/PI deve gerar relatórios contendo todos os lançamentos contábeis¹ gerados pelas sedes municipais filiadas. Atualmente, todo o controle dessas informações é feito através planilhas eletrônicas. Conforme novas comunidades se integram à Sisar/PI essas tabelas ficam cada vez mais complexas, pois, há muitas informações para serem tratadas.

Para resolver esse problema, criou-se um *software*² que fará o gerenciamento desses estoques de forma clara e de fácil usabilidade para seus usuários, fazendo controle entradas, saídas e gerar relatórios mensais automaticamente, que poderão ser exportados para o

¹ Registro de movimentação que ocorre no caixa de uma empresa.

² Programa de computador.

Figura 1 – Representação gráfica da logística da SISAR/PI



Fonte: O autor (2017)

formato CSV (Comma-Separated Values)³.

Os capítulos seguintes mostram o processo de construção desse *software*. No capítulo 2 será mostrado os conceitos e tecnologias adotadas e o embasamento teórico que levou à escolha de tais tecnologias. O capítulo 3 apresenta as fases do desenvolvimento do sistema, desde o levantamento dos requisitos até a implementação, seguindo as técnicas e modelos definidos pela engenharia de software. No capítulo 4 será visto o sistema já desenvolvido, mostrando suas principais funções e como ele resolve o problema proposto. O capítulo 5 mostra os métodos utilizados para avaliar o sistema e resultados obtidos. Por fim, o capítulo 6 apresentará as considerações finais e indicações de trabalhos futuros sobre o trabalho desenvolvido.

1.1 Objetivos

O objetivo deste trabalho é desenvolver um sistema de informação para controle e gerenciamento de estoques para a empresa Sisar/PI (Sistema Integrado de Saneamento Rural do Piauí). Através desse sistema, a Sisar/PI poderá gerenciar estoques de seus materiais e equipamentos em diferentes comunidades.

1.1.1 Objetivos Específicos

- Gerenciar estoques de materiais usados por companhias de saneamento rural.
- Implementar mecanismo de segurança para restringir o acesso de usuários sem autorização.

³ Arquivo de texto que contém conteúdo de células de uma tabela, o conteúdo de cada coluna é delimitado por um ponto e vírgula.

- Gerenciar entradas e saídas dos estoques.
- Gerar relatórios das transações realizadas e exportar para arquivos no formato CSV.

2 Referencial Teórico

Neste capítulo será apresentado o embasamento teórico que levou à escolhas das tecnologias e ferramentas usadas no desenvolvimento do *software* proposto neste trabalho.

2.1 Sistemas de Informação

Um sistema de informação pode ser definido como um conjunto de informações inter-relacionadas que coletam, manipulam e disseminam dados e informações. Essas informações podem auxiliar dentre outras coisas, no processo decisório de seu usuário (STAIR, 2006; REZENDE, 2011). Para Laudon e Laudon (2004) esses sistemas podem ter grande influencia no comportamento das empresas, eles podem auxiliar as empresas a se expandir a locais distantes, criar novas opções de produtos e serviços, tornar as tarefas e trabalhos mais organizados e podem até transformar radicalmente a forma como as empresas conduzem os negócios.

O maior objetivo do sistema de informação, independente de seu nível ou classificação, é auxiliar os processos de tomadas de decisões nas organizações. Se os sistemas de informação não se propuserem a atender esse objetivo, sua existência não será significativa para as organizações (REZENDE, 2011).

2.1.1 Tipos de Sistemas de Informação

Para atender as diferentes especializações das organizações, os sistemas de informação são divididos em diferentes tipos (LAUDON; LAUDON, 2004). De acordo com Gordon e Gordon (2006) eles são divididos de acordo com seu propósito em sistemas de automação, sistemas processamento de transações e sistemas de suporte à gestão e quanto a seu escopo em sistemas individuais, departamentais, empresariais e interorganizacionais. Neste trabalho será abordado apenas os sistemas quanto a seu propósito.

2.1.1.1 Sistemas de Automação

São usados para tornar as atividades mais fáceis e menos intensivas em mão de obra (GORDON; GORDON, 2006). Eles auxiliam a organização a implantar novas tecnologias e facilitam o gerenciamento de documentos (LAUDON; LAUDON, 2004).

2.1.1.2 Sistemas Processamento de Transações (SPT)

De acordo com Laudon e Laudon (2004), estes são sistemas integrados básicos que atendem ao nível operacional da empresa e realizam e processam as transações rotineiras,

como registros de pedidos de vendas, reservas de hotel, folhas de pagamento e etc.

Processar uma transação inclui atividades tais como gravar, arquivar e recuperar registros ou preencher formulários de pedidos e cheques. [...] dão suporte a empregados de nível operacional no desempenho das funções rotineiras do negócio através do fornecimento de dados para responder perguntas [...] (GORDON; GORDON, 2006)

2.1.1.3 Sistemas de Informações Gerenciais

Eles fornecem relatórios com informações que são usadas para auxiliar na tomada de decisões (GORDON; GORDON, 2006).

Os sistemas de informações gerenciais resumem e relatam as operações básicas da empresa. Os dados básicos de transações obtidos dos SPTs são comprimidos e comumente apresentados em longos relatórios produzidos segundo uma programação periódica (LAUDON; LAUDON, 2004).

2.1.2 Ciclo de Vida dos Sistemas de Informações

De acordo com Rezende (2005), um sistema de informação, quando não passa por atualizações tem um prazo de vida curto, de no máximo 5 anos. Vários fatores podem contribuir para a morte de um sistema de informação, Rezende (2011) destaca o desuso ou substituição e utilização de tecnologias precárias ou desatualizadas.

Rezende (2011) destaca 8 fases do ciclo de vida natural de um sistema de informação, são elas: **Concepção**, nesta fase são elaborados os requisitos que o sistema deve atender, **Construção**, é onde o sistema é, de fato, construído, **Implantação**, o sistema é apresentado ao cliente e/ou usuário, **implementação**, após *feedback*¹ gerado na fase de implementação, pode ser necessário adicionar, remover ou melhorar funções do sistema, **Maturidade**, é a fase estável do sistema, onde ele está atendendo todos os seus requisitos, **Declínio**, o sistema começa a se tornar depreciado e cair em desuso, **Manutenção**, são feitas, a fim de prolongar a vida do sistema e inevitavelmente ele chega a **Morte**.

2.2 Engenharia de Software

Para Rezende (2005) a Engenharia de Software é uma disciplina que reúne metodologias, métodos e ferramentas usadas desde o desenvolvimento até o fim do ciclo de vida de vida de um *software*. Sua meta é o desenvolvimento de sistemas com base na boa relação de custo-benefício (SOMMERVILLE, 2003) e que sigam uma abordagem sistemática, disciplinada e quantificável no desenvolvimento (PRESSMAN; MAXIM, 2016).

A Engenharia de Software caminha em paralelo com os Sistemas de Informação, visto que, ambos os temas são destinados às organizações, para auxiliar na tomada de decisão

¹ Processo de análise de determinada situação que retorna uma orientação de como se deve agir.

sob o foco do negócio (REZENDE, 2005). Para (AUDY; ANDRADE; CIDRAL, 2005) a implementação de sistemas de informação inclui um processo de *software* onde são cumpridas fases de desenvolvimento definidas pela engenharia de software, como levantamento de requisitos, análise, projeto, construção, teste, implantação e manutenção.

2.2.1 Modelos de Processos de Software

Um processo de *software* é um conjunto de atividades relacionadas que são necessárias para o desenvolvimento de um programa de computador (SOMMERVILLE, 2003). Para (PRESSMAN; MAXIM, 2016) a finalidade dos modelos de processos é tentar reduzir os inúmeros problemas que podem ocorrer durante o desenvolvimento de um *software*.

A seguir, será abordado os modelos de processos de *software* do tipo: **Cascata**, **Incremental** e *Pessoal e de equipe*.

2.2.1.1 Desenvolvimento em Cascata

No modelo cascata o desenvolvimento do *software* é dividido em estágios. Antes de iniciar o projeto deve ser feito um planejamento das atividades que serão desenvolvidas em cada estágio. (SOMMERVILLE, 2003) defende a existência de cinco principais estágios, são eles:

- **Análise e definição de requisitos:** São levantados os requisitos e estabelecidas metas;
- **Projeto de sistema e *software*:** Envolve identificação e descrição das abstrações fundamentais do sistema de *software*;
- **Implementação e teste unitário:** O projeto é desenvolvido como um conjunto de unidades de programa e são feitos testes em cada unidade destas;
- **Integração e teste de sistema:** As unidades individuais são integradas e testadas como um sistema completo;
- **Operação e manutenção:** O sistema é colocado em uso. Os erros encontrados são corrigidos e é feita melhorias no sistema.

O modelo cascata é frequentemente inadequado para os projetos de *software* atuais, pois, em princípio, cada estágio deve ser concluído para que seja iniciado o próximo (Pressman e Maxim (2016), Sommerville (2003)). Isso pode levar a um "estado de bloqueio" nos quais membros da equipe de projeto precisam esperar que outros membros da equipe finalizem suas tarefas (Pressman; Maxim, 2016).

2.2.1.2 Desenvolvimento Incremental

No desenvolvimento incremental é feita uma versão inicial do sistema, com as funcionalidades básicas, esta versão é apresentada ao cliente, que dá um *feedback* para a equipe de desenvolvimento, os desenvolvedores identificam os pontos fortes e fracos daquela versão e desenvolvem uma nova versão com correção de erros, melhorias e novas funcionalidades em relação a versão anterior. Esse processo se repete até que um sistema adequado seja entregue (SOMMERVILLE, 2003).

Ainda de acordo com Sommerville (2003), o desenvolvimento incremental é a abordagem mais comum para o desenvolvimento dos sistemas atuais.

2.2.2 Desenvolvimento Ágil

Os métodos de desenvolvimento ágeis foram criados com o objetivo de sanar fraquezas reais e perceptíveis da engenharia de software convencional, principalmente os longos prazos de tempo exigidos para se desenvolver um software com métodos convencionais, (PRESSMAN; MAXIM, 2016).

Segundo Soares (2004), a grande preocupação das metodologias ágeis é gastar menos tempo com a documentação e mais tempo com a implementação. Ainda de acordo com Soares (2004), os métodos ágeis não procuram prever possíveis novos fatores que possam surgir durante o desenvolvimento, os métodos ágeis se adaptam a eles.

2.2.2.1 Extreme Programming (XP)

Extreme programming (XP) é um modelo de desenvolvimento de *software* orientado a objetos focado em projetos cujo requisitos não são claros e estão sujeitos a frequentes mudanças (TELES, 2004). Segundo Pressman e Maxim (2016), os primeiros trabalhos associados ao *Extreme Programming* ocorreram durante o final do século de 1980 e a primeira publicação sobre o assunto foi feita em 1999, por Kent Beck.

A ideia de Kent Beck ao propor o XP não era criar uma nova metodologia ou outro processo, ele queria tornar os processos irrelevantes, fazendo com que os projetos se tornassem mais leves e rápidos (BECK, 2000).

Para Pressman e Maxim (2016) o XP envolve um conjunto de regras e práticas constantes no contexto de quatro atividades metodológicas:

- Planejamento: A atividade de planejamento começa com um conjunto de histórias que descrevem o funcionamento que servem como base para o levantamento dos requisitos;
- Projeto: O projeto XP procura sempre ser o mais simples possível. O projeto oferece um guia de implementação para uma história a medida que são acrescentados novos elementos;

- Codificação: Nesta fase os programadores começam a implementar o projeto. O XP recomenda que dois programadores trabalhem no código de uma mesma história, para fornecer um mecanismo de solução de problemas em tempo real. Outro ponto importante que o XP recomenda, é que antes de seguir com o desenvolvimento do código, devem ser feitos testes unitários².
- Teste: A medida que os testes unitários são organizados em uma sequência universal de testes, são feitos testes de integração³ e validação⁴ diariamente.

2.2.2.2 Dynamic Systems Development Method (DSDM)

DSDM (Metodologia de Desenvolvimento de Sistemas Dinâmicos, em português) é uma abordagem de desenvolvimento de *software* ágil que oferece uma metodologia voltada para projetos que têm restrição de prazo de entrega (PRESSMAN; MAXIM, 2016). A filosofia do modelo DSDM é que 80% da aplicação pode ser concluída em 20% do tempo que ela levaria para ser 100% concluída (BECK et al., 2001).

De acordo com Pressman e Maxim (2016), O DSDM pode ser combinado com a XP para fornecer uma abordagem combinada do modelo de processos confiável do DSDM com as práticas de construção incremental do XP.

2.2.2.3 Scrum

Segundo Pressman e Maxim (2016), o *Scrum* foi concebido por Jeff Sutherland e sua equipe de desenvolvimento em 1990. Sutherland criou uma metodologia que enfatizasse em projetos com prazo de entrega apertado, requisitos mutáveis e urgência de negócio.

A Scrum divide o desenvolvimento em iterações de trinta dias. Equipes pequenas, de até dez pessoas, são formadas por projetistas, programadores, engenheiros e gerentes de qualidade. Estas equipes trabalham em cima de funcionalidades definidas no início de cada sprint. A equipe é responsável pelo desenvolvimento desta funcionalidade. (SOARES, 2004).

2.3 Levantamento de Requisitos

Na engenharia de *software*, requisito são as especificações que um sistema deve possuir para satisfazer as vontades dos clientes/usuários. Cysneiros e Leite (2001), em sua tese de doutorado, define requisito como sendo a "condição necessária para a obtenção de certo objetivo, ou para o preenchimento de certo objetivo".

Ainda segundo Cysneiros e Leite (2001), os requisitos podem ser divididos em dois tipos: requisitos funcionais e não-funcionais, onde, os requisitos funcionais expressam as

² São testes realizados em partes individuais de código fonte

³ Os testes são combinados e testados em grupos.

⁴ Verifica se o código estar funcionando como deveria, após a realização de testes.

funções que o sistema deve implementar, já os não-funcionais expressam condições de comportamento e restrições.

2.4 UML

UML (*Unified Modeling Language*, ou Linguagem de Modelagem Unificada em português) é uma linguagem de modelagem usada para descrever projetos de *software* através de representações gráficas. A UML é usada particularmente para representar projetos que utilizam o estilo orientado a objetos (FOWLER, 2007).

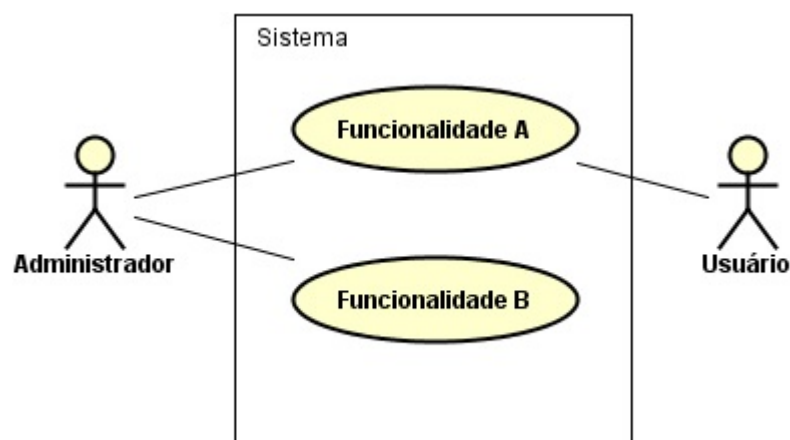
A UML é independente tanto de linguagens de programação quanto de processos de desenvolvimento. Isso quer dizer que ela pode ser utilizada para a modelagem de sistemas, não importando que linguagem de programação será utilizada na implementação do sistema, nem a forma (processo) de desenvolvimento adotada. (BEZERRA, 2015).

2.4.1 Diagrama de Casos de Uso

Casos de uso são abstrações de pequenas histórias narrando funcionalidades do sistema, que mostram como um ou mais usuários (chamados de atores) interagem com o sistema. (GUDWIN, 2015).

Na Figura 2 estar representado um diagrama de caso de uso, onde, há dois atores, o Administrador que interage com as funcionalidades A e B e Usuário que interage com a funcionalidade B. Os atores são representados por bonecos que lembram o corpo humano, o sistema por uma figura retangular e as funcionalidades são representadas por figuras ovais com uma descrição dentro.

Figura 2 – Exemplo de diagrama de caso de uso UML.



Fonte: O autor (2017)

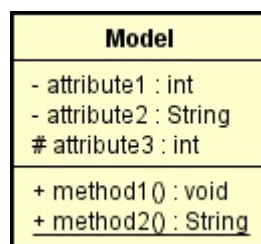
2.4.2 Diagrama de Classe

Os diagramas de classes são usados para modelar a estrutura de um sistema, eles ilustram um conjunto de classes do projeto, mostrando como cada classe se relaciona com as demais (SILVA; VIDEIRA, 2001).

Em princípio, um diagrama de classes representa uma visão do modelo estrutural estático, que pode ser entendido como a união de todos os diagramas de classe e de objetos, da mesma maneira que podemos projetar uma figura tridimensional em diversos planos bidimensionais. (GUDWIN, 2015)

Na Figura 3 pode ser vista uma classe representada usando a linguagem UML. Os sinais de +, - e # no início dos atributos e métodos indicam qual o seu modificador de acesso⁵ e o sublinhado indica que o método é estático.

Figura 3 – Representação de classe no padrão UML.



Fonte: O autor (2014)

2.5 Linguagem Java

A linguagem Java surgiu em 1991, a partir de um projeto de pesquisa financiado pela Sun Microsystems (ORACLE, 2017). Esse projeto resultou em uma linguagem similar a linguagem C++, o projeto teve dificuldade em emplacar no mercado, mas, com a explosão da Web, em 1993, a Sun resolveu utilizar o Java para adicionar conteúdo dinâmico em páginas Web (DEITEL; DEITEL, 2015).

Atualmente Java é uma das linguagens de programação mais usadas no mundo, aplicações desenvolvidas em Java podem ser encontradas em ambientes Web (MARUYAMA, 2002), Desktop (DEITEL; DEITEL, 2015) e Mobile (LECHETA, 2015).

Java usa o paradigma de orientação a objetos⁶ e é multiplataforma, ou seja, um *software* feito em Java pode ser executado em qualquer sistema operacional que possua a máquina virtual java instalada.

⁵ Palavras chave que indicam o nível de visibilidade de classes, atributos e métodos, dentro do contexto de um projeto de *software*.

⁶ Linguagens de programação que tentam aproximar o mundo real do virtual através simulações de objetos, que representam coisas do mundo real.

2.5.1 Integrated Development Environments (IDEs)

As IDEs fornecem uma série de ferramentas para agilizar o desenvolvimento de aplicações. Há várias IDEs que podem ser usadas para desenvolver aplicações em Java (DEITEL; DEITEL, 2015); as mais usadas são o Eclipse (ECLIPSE, 2017), NetBeans (NETBEANS, 2017). Tanto o Eclipse quanto o Netbeans podem ser usadas para desenvolver em outras linguagens além do Java.

Ambas as IDEs cumprem bem seu papel, porém, O Netbeans, por ser a IDE oficial do Java já vem com os principais *plugins* necessários para usar os recursos do Java, como por exemplo a biblioteca JavaFx. Já no Eclipse, vem apenas o essencial instalado, os demais *plugins* devem ser instalados e configurados pelo usuário.

2.5.2 Modelo MVC (Model-View-Controller)

O modelo MVC é um padrão de arquitetura de *software* que consiste em dividir os objetos em três camadas: modelo, visão e controlador (GAMMA, 2009).

De acordo com Freeman e Freeman (2009), a camada visão fornece uma apresentação do modelo (visualização gráfica para o usuário), a camada controlador recebe os dados do usuário (na camada visão) e determina o que isso significa para o modelo e a camada modelo contém todos os dados, estados e lógicas do sistema.

2.5.3 JavaFX

JavaFX, atualmente na versão 8, é a nova biblioteca GUI (Graphical User Interface)⁷ do Java. Segundo (HECKLER et al., 2014), o objetivo do JavaFx é criar aplicações que podem ser usadas em vários tipos de dispositivos, como sistemas embutidos, *smartphones*, TVs, *tablets* e *Desktops*.

Para construir as interfaces gráficas, o JavaFx utiliza FXML, uma linguagem baseada em XML⁸, que fornece uma estrutura para construir interfaces de usuário separando a lógica do código da aplicação (ORACLE, 2014).

Para facilitar a manipulação dos componentes do JavaFx o programador pode usar o programa *JavaFX Scene Builder*. Esta ferramenta permite que se construa interfaces gráficas arrastando e soltando os componentes, modificando a interface sem escrever códigos (DEITEL; DEITEL, 2015).

⁷ Fornece uma interface gráfica amigável pela qual o usuário interage com o sistema

⁸ Acrônimo de Extensible Markup Language, XML é uma linguagem de marcação pode ser facilmente interpretada pela maioria das linguagens de programação.

2.6 Banco de Dados MySQL

O MySQL é um Sistema de Gerenciamento de Banco de Dados (SGBD) de código aberto, que usa um sistema de alocação de memória muito rápido e baseado em processo(thread) (SUEHRING, 2002).

O MySQL foi projetado inicialmente para trabalhar com aplicações de pequeno e médio porte, hoje atende aplicações de grande porte [...]. Possui todas as características que um banco de dados de grande porte precisa, sendo reconhecido por algumas entidades como o banco de dados *open source*⁹ com maior capacidade para concorrer com programas similares de código fechados [...] (MILANI, 2007).

Há diversas ferramentas que auxiliam na criação de banco de dados. O *MySQL Workbench* é a ferramenta oficial do *MySQL* para modelar banco de dados, configurar servidores, administrar usuários, fazer backup e muitas outras funções de forma visual (MYSQL, 2017).

⁹ Programa de computador livre e de código aberto, que pode ser modificado por qualquer um.

3 Desenvolvimento do Sistema

Neste capítulo será descrito detalhes do desenvolvido do sistema, que a partir de agora será identificado com CIT SISAR-PI (Controle Integrado de Transações SISAR-PI). Aqui são mostradas as fases de levantamento de requisitos, modelagem de dados usando diagramas e implementação.

O desenvolvimento desse projeto seguiu as práticas do modelo de desenvolvimento incremental e alguns elementos do modelo ágil *Extreme Programming*. Ao longo do desenvolvimento, ocorreram uma série de reuniões com os *stakeholders*¹ que, normalmente, aconteciam a cada três semanas.

As reuniões consistiam em discutir dois pontos principais:

- **Validar funcionalidades:** Era apresentada as funcionalidades já implementadas aos clientes, que davam um rápido *feedback*. De acordo com o *feedback* a funcionalidade poderia ser alterada, mantida ou removida do projeto. Caso houvesse a necessidade de fazer alterações, as mudanças eram feitas e discutidas novamente na próxima reunião.
- **Novas funcionalidades:** Novas funcionalidades eram apresentadas e se discutia possíveis alterações, viabilidade e prazo para implementação.

3.1 Levantamento dos Requisitos

Os requisitos funcionais e não-funcionais do sistema foram levantados com base em reuniões com os clientes e usuários do sistema. Na Tabela 1 são mostrados requisitos funcionais e na Tabela 2 os requisitos não-funcionais.

Em ambas as tabelas, as colunas "Identificador" e "Descrição" são apresentadas um código que identifica o requisito e uma sucinta descrição do requisito, respectivamente. Na tabela 1, a coluna "Depende de" contém códigos dos requisitos dos quais possui alguma dependência, para ser implementado. A coluna "Categoria", da Tabela 2 contém a categoria na qual o requisito é classificado.

3.1.1 Regras de Negócios

As regras foram definidas ao longo das reuniões com os clientes, tendo como base a logística e as políticas de negócio desenvolvidos na SISAR-PI. As regras de negócio do sistema são listadas a seguir, na Tabela 3.

¹ São as partes interessadas no sistema (desenvolvedores, clientes, usuários)

Tabela 1 – Requisitos funcionais do sistema.

Identificador	Descrição	Depende de
RF01	Cadastro de novo materiais ou equipamento, que além de suas características básica, deve possuir uma foto e descrição de sua utilização	
RF02	Listar relação de itens dos estoques e todos itens utilizados, não necessariamente a um estoque ainda	RF01
RF03	Filtros de busca para facilitar a procura dos itens no estoque	RF02
RF04	Adicionar novos itens ao estoque e transferir itens entre estoques	RF02, RF03
RF05	Criar lista de transferência para facilitar a transferência de vários itens ao mesmo tempo	RF03, RF04
RF06	Exibir lista de transferências dos estoques, separadas por transferências de entrada e transferências de saídas dos estoques	RF04
RF07	Exibir lista de itens das transferências realizadas	RF06
RF08	Exportar lista de itens nos estoques para arquivos nos formatos PDF e CSV	RF02
RF09	Exibir relatórios de entradas e saídas dos estoques, exibindo valores em reais e separados de por grupo de material (hidráulico, elétrico, sistema de tratamento)	
RF10	Exibir relatório com balanço geral dos estoques (saldo anterior, valor das entradas, valor das saída e saldo atual.	
RF11	Cadastro de novos fornecedores, que deve conter endereço e informações de contato	
RF12	Exibir lista de fornecedores, com um link para acesso rápido a lista de todos os seus itens	RF11
RF13	Cadastro de usuários com nome de usuário e senha criptografada	RF11
RF14	Cadastro de funcionários, que devem ter uma maleta de itens associada a eles	RF11
RF15	Exibir lista de funcionários, com um link para acesso rápido a lista dos itens em suas maletas	RF14

Tabela 2 – Requisitos não-funcionais do sistema.

Identificador	Descrição	Categoria
RNF01	O sistema deve ter um controle de autenticação e autorização de usuários através de senhas criptografadas	Segurança
RNF02	O sistema deve fornecer ferramentas para fazer backup de todas as informações contidas no banco de dados	Confiabilidade
RNF03	O sistema deve exibir mensagens de aviso, de erro e informações para o usuário quando este tentar realizar alguma tarefa não permitida e/ou alguma alteração crítica no sistema	Usabilidade
RNF04	Deve ser exibido barras de progresso quando o usuário solicitar tarefas demoradas, que deixam o sistema ocioso por um tempo considerável	Usabilidade

Tabela 3 – Regras de negócio do Sistema

Identificador	Descrição	Depende de
RN01	Para acessar o sistema, o usuário deve informar um nome de usuário e senha cadastrados no sistema	
RN02	Os usuário podem possuir diferentes níveis hierárquicos, onde, usuários com nível maior têm acesso a todas as funções do sistema e usuários com nível menor apenas às funções básica	RN01
RN03	As nova aquisições de materiais devem sempre ser adicionadas no estoque central, para, só depois, serem transferidas para os estoques das comunidades	
RN04	Toda saída do estoque central para as comunidades deve ser tratado como um lançamento nos relatórios da contabilidade	RN03

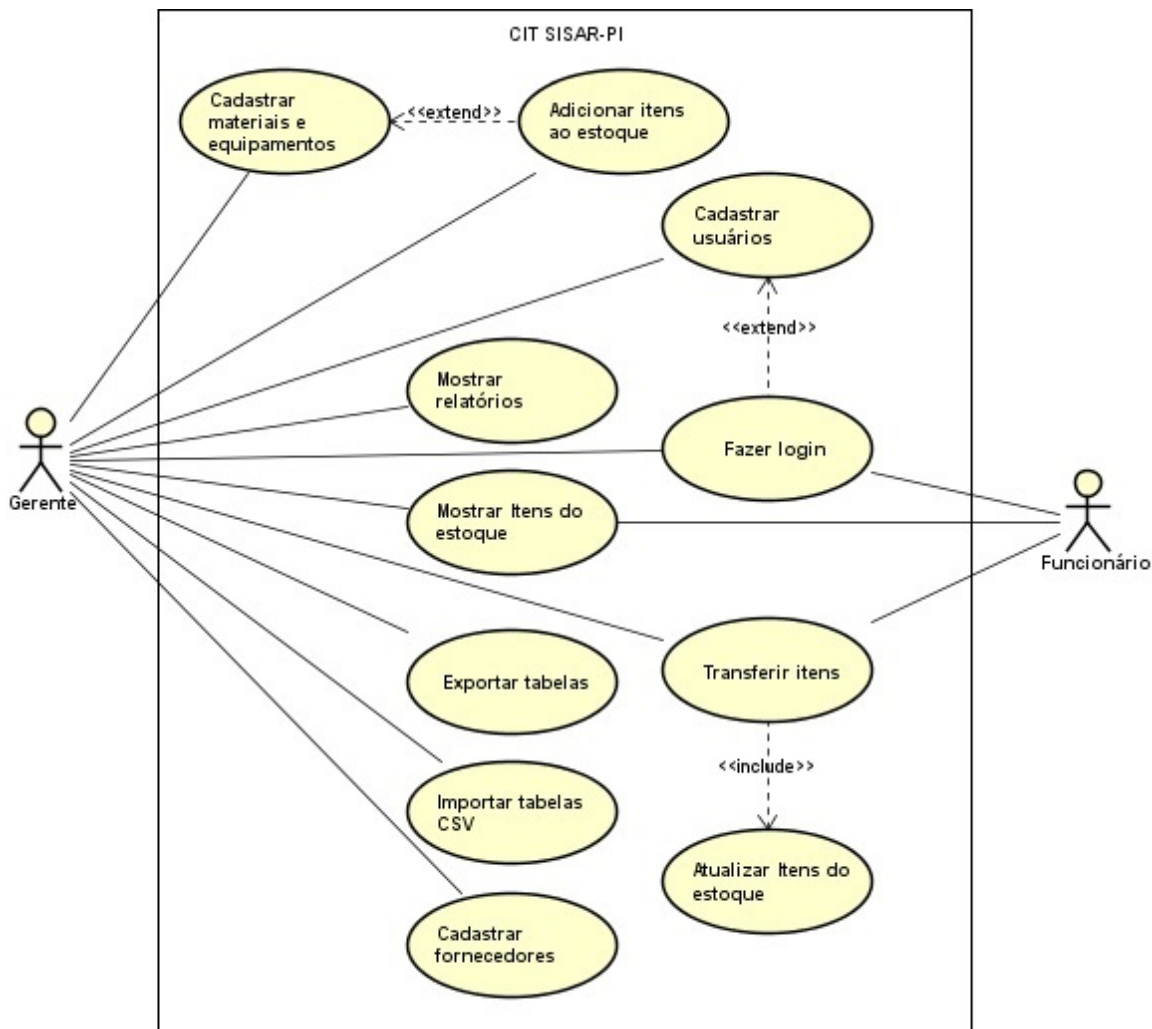
3.2 Diagramas UML

Esta seção possui alguns diagramas UML elaborados com base nos requisitos do sistema, que serão usados pelos desenvolvedores e/ou clientes, para facilitar a compreensão do funcionamento do sistema.

3.2.1 Diagrama de Casos de Uso

Na Figura 4, está representado o diagrama de caso de uso dos sistema. O sistema pode ter dois tipos de usuários: **Gerente** e **Funcionário**. O Gerente pode ter acesso à todas as funções do sistema, já o Funcionário, pode, apenas consultar os estoques e realizar transferências.

Figura 4 – Diagrama de caso de uso.



Fonte: O autor (2017)

No Apêndice A há a documentação sucinta de todos esses casos de uso. Na documentação estão detalhes como: os atores envolvidos, a sequência de operações necessárias para realizar a ação do caso de uso e restrições.

3.2.2 Diagrama de Classe

O diagrama de classe visto na Figura 5 mostra as classes que representam os objetos do sistema e a relação que cada um desses objetos se relaciona com os demais.

A classe **Item** representa os materiais e equipamentos que podem fazer parte dos estoques (classe **Stock**). O estoque herda os atributos de um município (classe **Municipality**) e um município pode ter vários estoques relacionados a ele. As classes **ItemStock** e **ItemSuitcase** representam um Item que faz parte de um determinado estoque ou na maleta de um funcionário, respectivamente. A classe **PhotoItem** representa a imagem de um item.

A classe **Provider** representa um fornecedor, um fornecedor pode ter vários itens associados a ele e pode ter informações de contato (representado pela classe **Contact**) e um endereço (representado pela classe **Address**). Contato e endereço pode manter relação com os estoques também.

Os usuário estão representados pela classe **User**. Um usuário pode possuir um funcionário associado a ele (classe **Employee**), os funcionários são usuários que estão em um nível hierárquico abaixo dos usuários normais e possuem restrições de acesso à determinadas funções do sistema.

Uma transferência (classe **Transfer**) representa uma movimentação ocorrida nos estoques. Uma transferência deve ter um ou mais itens, o usuário que está logado e/ou o estoque de origem e destino da transferência.

3.3 Projeto do Banco de Dados

O projeto do banco de dados foi feito em quatro fases: primeiro foi contextualizado o problema com base nos requisitos e foi elaborado um mini-mundo (Seção 3.3.1), o segundo passo foi identificar as entidades² no mini-mundo e criar o projeto conceitual do sistema (Apêndice B).

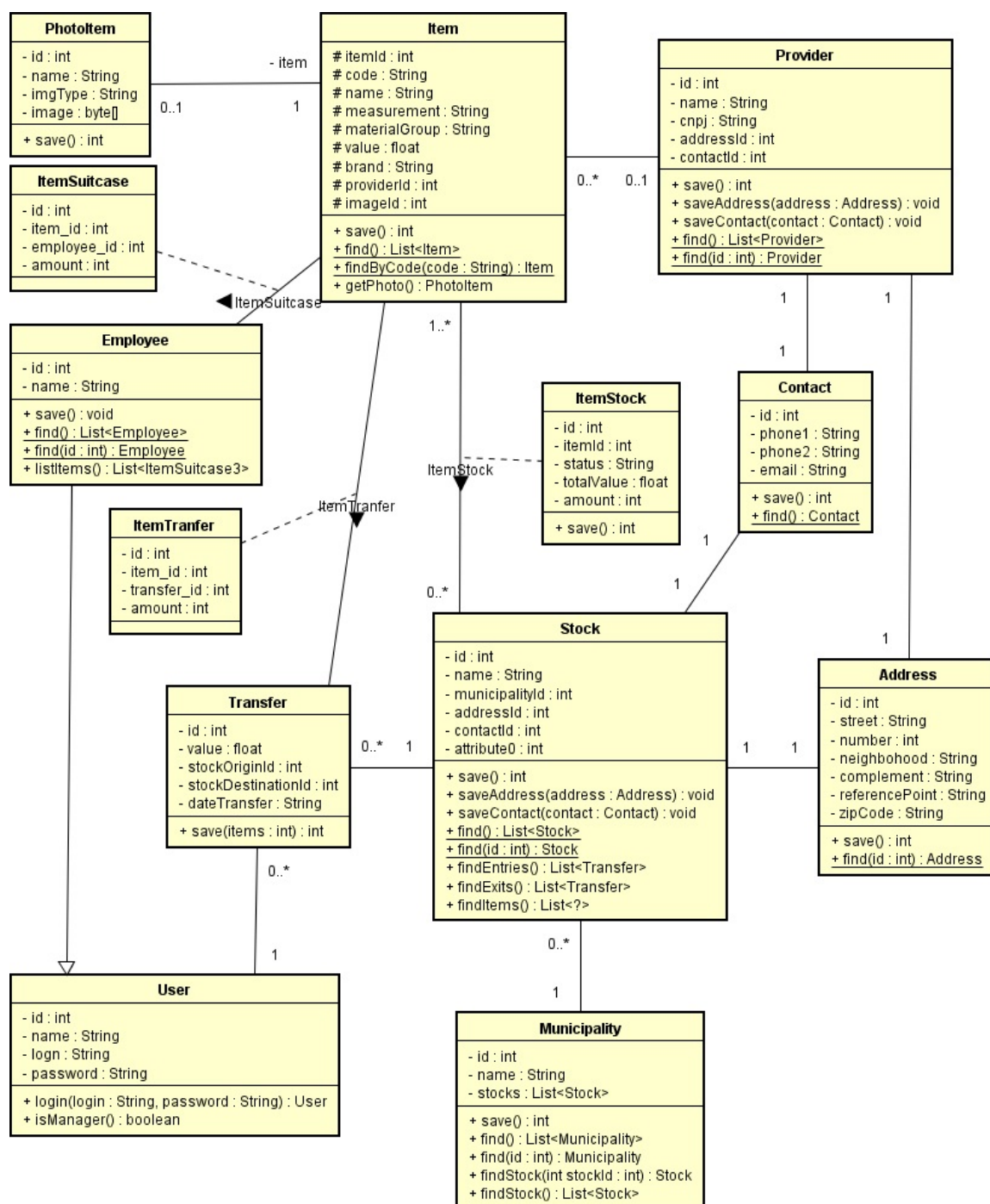
Com o projeto conceitual em mãos, o próximo passo foi modelar e definir as relações e restrições do banco de dados através de um diagrama entidade relacionamento (Apêndice C), por último, o banco de dados foi implementado utilizando o SGBD MySQL.

3.3.1 Mini-mundo

A SISAR/PI é uma empresa que atua no ramo de manutenção e instalação de equipamentos de saneamento rural cidade de Picos-PI e mais 23 municípios vizinhos, até a data de finalização deste trabalho. Cada município possui um ou mais estoques de materiais e equipamentos utilizados pela SISAR/PI. Há um estoque principal localizado na cidade de Picos por onde todos os novos equipamentos passam, para depois serem enviados para os estoques dos municípios.

² Em banco de dados, é um conjunto de informações que pode ser modeladas em forma de tabela.

Figura 5 – Diagrama de classe do sistema.



Fonte: O autor (2017)

Para facilitar o gerenciamento desses estoques, a SISAR/PI necessita de um sistema para controlar a entrada de novo itens nos estoques, transferências de itens entre estoques e gerar relatórios diversos.

O sistema deve permitir que se possa realizar cadastro de itens dos estoques (materiais e/ou equipamentos), que devem possuir informações tais como nome, código de identificação, unidade de medida, valor, marca, fornecedor, imagem e uma breve descrição.

Os estoques devem possuir um nome e endereço e contatos de seu lugar físico. Os estoques podem possuir nenhum ou vários itens e os itens podem está em vários estoques diferentes. Os itens em estoque devem conter a quantidade e a situação, que diz se o item está com pouca quantidade ou não.

Poderá ser feito transferências de itens de um estoque para outro, onde, deve ser registrado a data da transferência, a origem (estoque de onde foi transferido), o destino (estoque para onde está sendo enviado). É importante mencionar que poderá ser enviado vários itens ao mesmo tempo.

Deve ser feito ainda, o cadastro dos fornecedores. Os fornecedores contém nome, CNPJ, endereço e informações de contato como e-mail e telefone.

Deve haver dois tipos de usuários do sistema, os gerentes que têm acesso a todas funcionalidades e funcionários, que podem, apenas consultar itens no estoque e fazer transferências. Os usuários terão dados como nome, CPF, nome de *login*, senha, endereço e informações de contato. Os funcionários possui uma especie de maleta de equipamentos, que contém alguns materiais usados por eles durante os serviços prestados. Esta maleta pode possuir vários itens e os itens da maleta podem está em várias maletas e também em outros estoques.

3.4 Implementação do Sistema

O CIT SISAR-PI foi desenvolvido usando a linguagem de programação Java versão 8, SGBD MySQL, a interface gráfica foi implementada usando FXML junto a biblioteca JavaFx e a organização do projeto segue os padrões definido pelo modelo MVC.

Durante o desenvolvimento foram usadas ferramentas para auxiliar e agilizar a implementação, tais como: a IDE NetBeans 8.2, a ferramenta para manipulação de arquivos XML Scene Builder 2.0 e o MySQL Workbench 6.3 para criação do banco de dados.

4 Usabilidade do Sistema

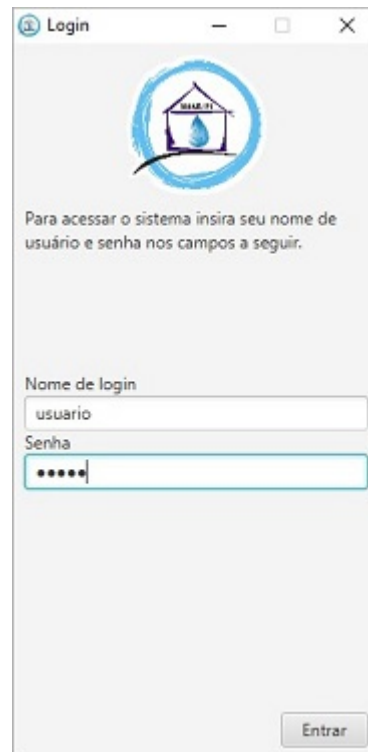
Neste capítulo será visto as funções do sistema disponíveis para o usuário, que são acessadas, na maioria dos casos, através da barra de menus (Figura 7b). Nas próximas seções para se referir aos menus será usada a seguinte notação: "Menu>Submenu>Sub submenu".

4.1 Login no Sistema

Ao executar o CIT SISAR-PI a primeira tela que aparecerá para o usuário é uma tela de *login*(Figura 6) onde deve ser informado um nome de usuário e senha, para iniciar uma nova sessão no sistema. Caso o usuário seja um gerente, ele será redirecionado para uma versão completa do sistema, onde é possível acessar todas as funcionalidades disponíveis (Figura 7).

Caso o usuário seja um funcionário, ele terá acesso apenas às funções relacionadas ao controle de estoque, como visualizar o estoque e realizar transferências. A tela do funcionário é idêntica à tela do gerente, com exceção da barra de menu, que terá apenas o menu Estoque.

Figura 6 – Tela de login no sistema



Nome de login
usuario

Senha

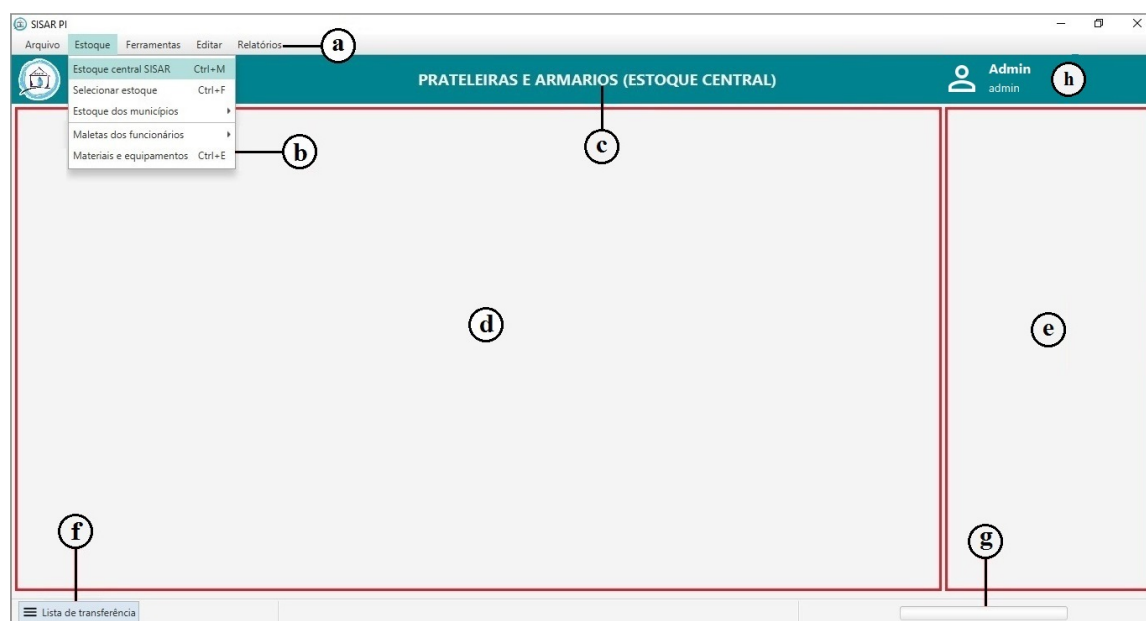
Entrar

Fonte: O autor (2017)

4.2 Layout principal

Na Figura 7 pode ser visto o *layout*¹ principal, comum a todas as telas do sistema. Para facilitar a identificação dos elementos internos das telas há balões com letras, identificado algumas partes da figura.

Figura 7 – Layout principal



Fonte: O autor (2017)

A seguir são detalhados cada um dos elementos mostrados na Figura 7:

- Barra de menus (Figura 7a) - Contém os menus que dão acesso às funcionalidades do sistema. Quando um desses menus é clicado, aparece uma lista de submenus, como visto na Figura 7b. Na Tabela 4 são listados todos os menus da barra de menus. Há três colunas na tabela, organizadas hierarquicamente, onde, a primeira coluna mostra os menus e a segunda e a terceira coluna mostram os submenus.
- Barra de título (Figura 7c) - mostra um título para o conteúdo mostrado.
- Conteúdo principal (Figura 7d) - Nesta área é mostrado as informações que o usuário solicita, como por exemplo os itens dos estoques, relatórios, etc.
- Conteúdo secundário (Figura 7e) - Pode conter filtros de conteúdo, botões ou outras informações relacionadas ao conteúdo principal exibido.
- Botão lista de transferência (Figura 7f) - Este botão fica habilitado apenas nas telas de estoque. Quando clicado pelo usuário, é aberto uma lista de transferência abaixo da tabela onde está listado os itens do estoque.

¹ Interface gráfica

- Barra de progresso (Figura 7g) - Quando o sistema está realizando uma tarefa demorada, esta barra mostra para o usuário que o sistema está realizando algo.
- Área do usuário (Figura 7h) - Exibe o nome e login do usuário. Ao clicar nesta área, será exibido para o usuário uma tela de configuração de sua conta.

Tabela 4 – Lista de menus e submenus da barra de menus.

Menu	Submenu	Sub submenu
Arquivo	Exportar para	CSV
		PDF
	Importar	Entradas e saídas de estoque
	Fazer backup	
	Sair	
Estoque	Estoque central SISAR	
	Selecionar estoque	
	Estoque dos municípios	Lista de municípios
	Materiais e equipamentos	
Ferramentas	Configurações	
Editar	Estoques	
	Comunidades	Nova comunidade
		Editar comunidade
	Municípios	Novo município
		Editar município
	Materiais e equipamentos	Novo material
		Editar material
	Fornecedores	Novo fornecedor
		Editar fornecedor
	Relatórios	Estoque central SISAR
Entradas nos municípios		
Entradas e saídas dos municípios		
Demonstração mensal - Entradas x Saídas		

4.3 Estoques

Os estoques podem ser acessados em "Estoques>Estoques dos municípios>Nome do município>Estoque". Na figura 8 temos a tela de um estoque.

No canto superior esquerdo (Figura 8a) há três abas, quando a aba **Estoque** é selecionada será mostrado a lista de itens no estoque (Figura 8b), na aba **Entradas** é mostrado a lista de entradas do estoque e a aba **Saídas** mostra a lista de saídas do estoque.

A Figura 9 mostra a lista de saídas do estoque (A tela de entradas e saídas são idênticas). Por padrão são mostradas as entradas e saídas do mês atual, porém, há dois

Figura 8 – Tela de exibição de estoque.

The screenshot shows a software interface for displaying inventory. At the top, there are three tabs: 'Estoque', 'Entradas', and 'Saídas', with 'Estoque' selected. Below the tabs is a table with columns: 'Código', 'Nome', 'Unidade', 'Grupo', 'Quantidade', 'Valor (R\$)', 'Valor Total (R\$)', and 'Situação'. A context menu is open over the row for 'Alicate Amperímetro' (code 0028), showing options: 'Editar item', 'Exportar para', 'Adicionar a lista de transferência', and 'Transferir para'. To the right of the table is a 'Filtros' panel with fields for 'Nome', 'Unidade de Medida', 'Grupo', 'Marca', 'Fornecedor', and 'Situação', along with a 'Limpar Filtros' button. At the bottom, there is a search field for 'Código' and 'Cancelar'/'Avançar' buttons.

Código	Nome	Unidade	Grupo	Quantidade	Valor (R\$)	Valor Total (R\$)	Situação
0028	Alicate Amperímetro	UND	ELÉTRICO	2			BAIXO
0093	Bombeador 54S-5/52	UND	HIDRÁULICO	1			BAIXO
0160	Cabo 16 mm	MT	ELÉTRICO	1			BAIXO
0162	Cabo 4 mm	MT	ELÉTRICO	3			BAIXO
0163	Cabo 6 mm	MT	ELÉTRICO	1			BAIXO
0171	Cadeado PADO 35 mm	UND	HIDRÁULICO	3	2,00	12,00	BAIXO
0172	Caibos	MT	HIDRÁULICO	6	0,18	0,18	OK
0199	CAPS Sold.(20)	UND	HIDRÁULICO	1	15,00	15,00	BAIXO
0271	Chave 24 boca e anel	UND	HIDRÁULICO	1	4,08	4,08	BAIXO
0271	Conector sindal 25mm 600 V	UND	ELÉTRICO	1	4,08	4,08	BAIXO
0271	Contato auxiliar p/ contactor Siemens Pequeno NC NO	UND	ELÉTRICO	3	11,77	35,31	BAIXO
0271	Filtro para bomba de água	UND	HIDRÁULICO	1	749,00	749,00	BAIXO

Fonte: O autor (2017)

Figura 9 – Lista de saídas de um estoque.

The screenshot shows a software interface for displaying inventory outputs. At the top, there are three tabs: 'Estoque', 'Entradas', and 'Saídas', with 'Saídas' selected. Below the tabs is a table with columns: 'Transferido para', 'Valor Total (R\$)', and 'Data'. The table contains one row with the following data:

Transferido para	Valor Total (R\$)	Data
MOURÕES	1.718,26	11/05/2017

Fonte: O autor (2017)

campos na parte direita da tela onde o usuário poderá escolher a data inicial e final em que ocorreram as transferências (Figura 10).

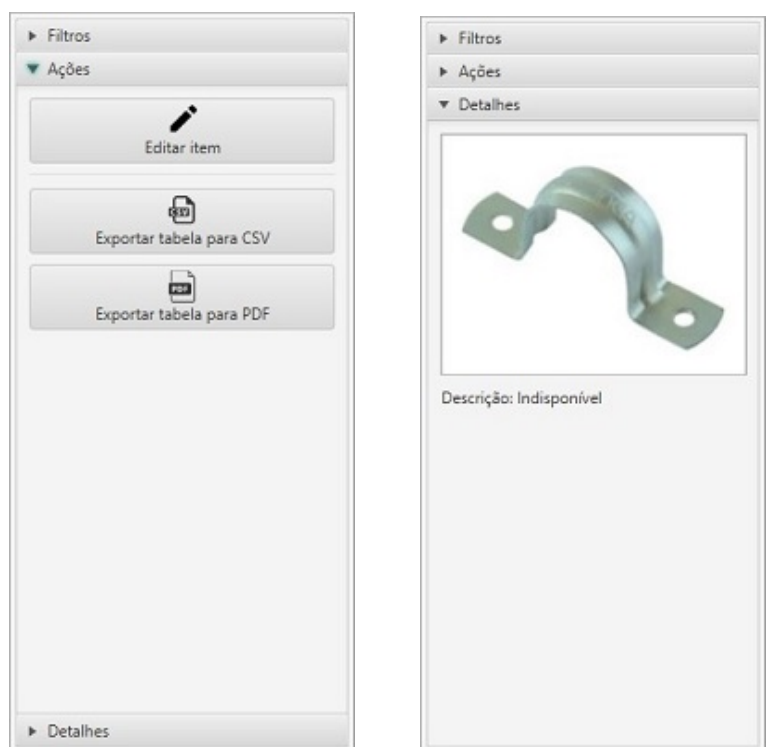
Figura 10 – Filtro das saídas de um estoque.

The screenshot shows a software interface for filtering inventory outputs. It contains two date input fields: 'Data Inicial' and 'Data Final'. Each field has a calendar icon to its right for date selection.

Fonte: O autor (2017)

Na lateral direita da tela de estoque (Figura 8d) há um menu com três abas. A aba **Filtros** contem um conjunto de campos para filtrar o conteúdo da tabela, a aba **Ações** (Figura 11a) contém alguns botões que executam tarefas relativas ao conteúdo principal da tela de estoque e a aba **Detalhes** (Figura 11b) mostra uma imagem e uma descrição de um item quando o usuário clica sobre o mesmo, na lista de itens do estoque.

Figura 11 – Conteúdo lateral da tela dos estoques.



(a) Ações relacionadas ao estoque.

(b) Detalhes do item selecionado.

Fonte: O autor (2017)

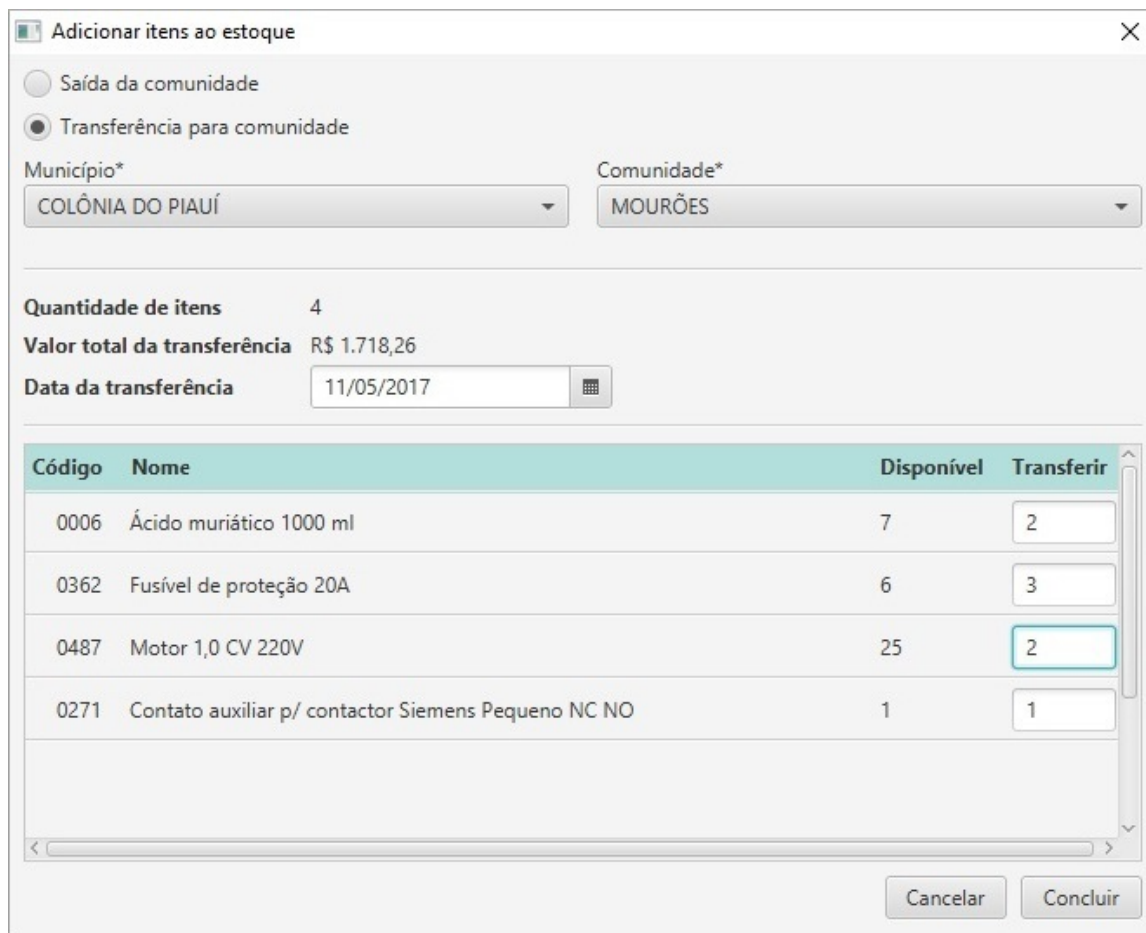
4.4 Transferências

Para o fazer uma transferência de itens de um estoque para outro, o usuário pode clicar no botão do canto inferior esquerdo da tela (Figura 7f), será exibida uma lista de transferência abaixo da lista de estoques (Figura 8e) onde o usuário pode adicionar os itens que desejar transferir.

Para adicionar itens na lista de transferência, o usuário pode informar o código do item (Figura 8f) ou selecionar os itens na tabela (para selecionar mais de um item deve pressionado o botão Ctrl e clicar nos itens desejados) e clicar com o botão direito do mouse sobre um dos itens selecionados, irá aparecer um menu (Figura 8c) então deve ser selecionado o menu **Adicionar a lista de transferência**. Caso não se queira adicionar itens na lista de transferência e ir direto para a tela de transferência (Figura 12 deve ser selecionado o menu **Transferir para**.)

Ao clicar no botão Avançar (Figura 8g) será aberta a tela de transferência (Figura 12). Na parte superior da tela há duas opções para o usuário selecionar. Caso seja selecionado a opção "Saída da comunidade", os itens selecionados serão retirados de circulação, caso a opção selecionada seja "Transferência para comunidade", o usuário precisa selecionar uma comunidade para onde os itens serão transferidos.

Figura 12 – Tela de transferência de itens dos estoques.



Adicionar itens ao estoque

Saída da comunidade
 Transferência para comunidade

Município* COLÔNIA DO PIAUÍ
Comunidade* MOURÕES

Quantidade de itens 4
Valor total da transferência R\$ 1.718,26
Data da transferência 11/05/2017

Código	Nome	Disponível	Transferir
0006	Ácido muriático 1000 ml	7	2
0362	Fusível de proteção 20A	6	3
0487	Motor 1,0 CV 220V	25	2
0271	Contato auxiliar p/ contactor Siemens Pequeno NC NO	1	1

Cancelar Concluir

Fonte: O autor (2017)

Abaixo há uma tabela contendo os itens que foram selecionados. A coluna **Disponível** indica a quantidade de itens disponíveis no estoque e na coluna **Transferir** há campos para o usuário informar a quantidade de itens que deseja enviar. Caso seja informado um valor superior ao disponível no estoque aparecerá um aviso informando que aquela ação não é permitida (Figura 13).

Figura 13 – Tala de aviso para o usuário.



Fonte: O autor (2017)

4.5 Exportação e Importação de Tabelas

A função de exportação de tabelas pode ser acessada através da barra de menus "Arquivo>Exportar>Para PDF/CSV", clicando com o botão direito do mouse sobre uma tabela e acessando o menu "Exportar> Para PDF/CSV" ou ainda através dos menus laterais como na Figura 11a. Para converter as tabelas para PDF foi usado a biblioteca iText (ITEXT, 2017).

Também é possível realizar importação de tabelas no formato CSV para o CIT SISAR-PI, através do menu "Arquivo>Importar>Entradas e saídas de estoque". Será mostrado uma tela para o usuário (Figura 14) onde é possível selecionar um arquivo CSV que contém as movimentações realizadas em uma comunidade, em determinado período de tempo e deve ser escolhido para qual estoque será enviado os itens.

Figura 14 – Importar estoque e entradas nas comunidades

Código	Nome	Quantidade	Transferir	
0038	Anel o-ring viton marrom cabeçote 053	5	0	
0043	Anel o-ring viton marrom valvula 053/133/0610-(2016) (borra...	1	0	
0046	Arame farpado 500mt	1	0	
0047	Arame Galvanizado Nº 12	6	5	
0049	Arco de Serra	1	0	
0050	Arco de Serra - reforçado	4	0	
0051	Areia filtrante	2	0	

Fonte: O autor(2017)

É importado para o sistema apenas as linhas do arquivo CSV que satisfaçam os seguinte critérios:

- A primeira coluna deve ser um número inteiro e no sistema deve haver m item registrado com o mesmo código;

- A terceira e a quarta coluna não podem ficar em branco e deve possuir um valor inteiro maior que zero.
- Caso apenas a terceira coluna não um valor e a quarta coluna sim, será importado apenas a saída e vice-versa;

Figura 15 – Modelo de arquivo CSV.

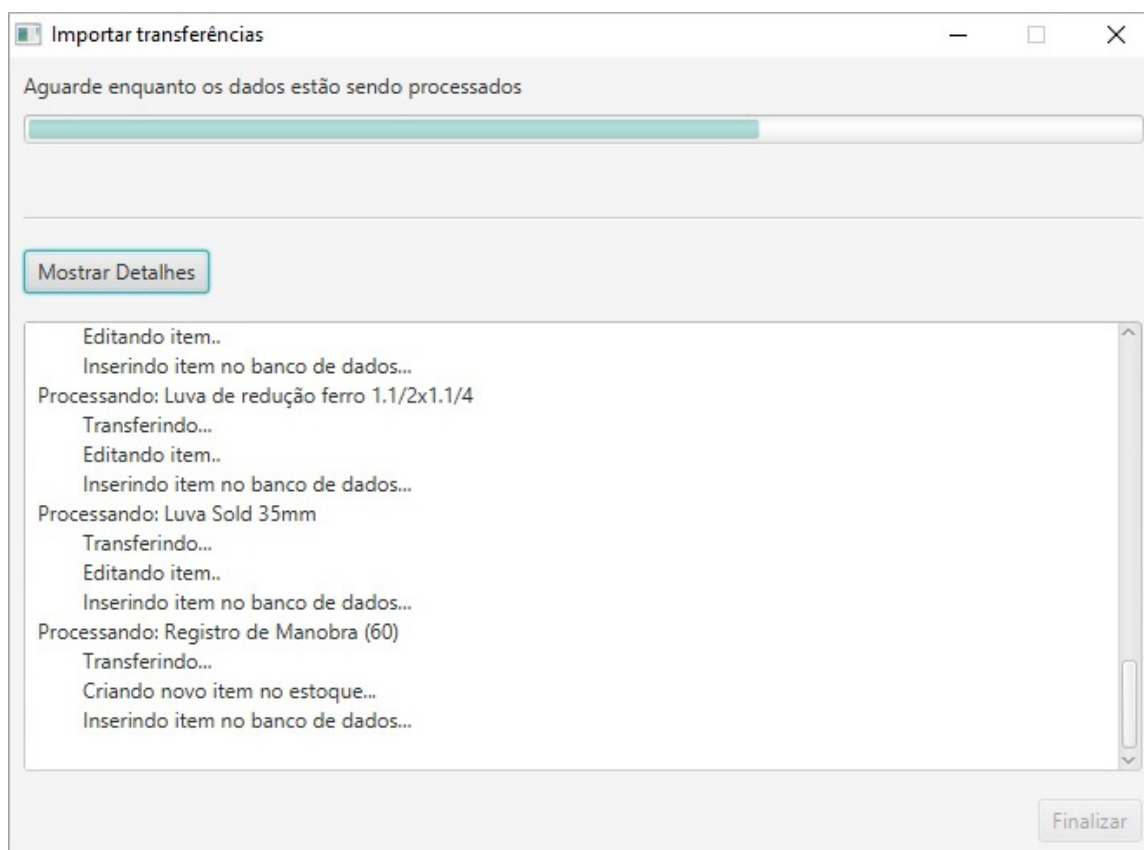
```
1 N°;IDENTIFICAÇÃO DO ITEM;ENTRADA;SALDO ATUAL
2 171;Cadeado PADO 25mm;15;10
3 415;Lima triangular de ferro;17;12;
4 429;Luva de correr DEFOFO;20;12
```

Fonte: O autor (2017)

A Figura 15 mostra um exemplo de arquivo CSV que pode ser importado para o CIT SISAR-PI.

Por se tratar de uma operação que pode demorar um pouco, enquanto os dados estão sendo salvos no banco de dados o progresso é mostrado para o usuário, conforme é mostrado na Figura 16.

Figura 16 – Tela de barra de progresso.



Fonte: O autor (2017)

4.6 Formulários

Na Figura 17 temos uma tela para cadastro ou edição de um material ou equipamento. O CIT SARPI-PI possui outras telas de formulário semelhantes, para os fornecedores, funcionários comunidades. Os campos marcados com (*) são obrigatórios, casos o usuário informe valores pra esses campos e tente salva as informações, aparecerá uma janela de diálogo semelhante a da Figura 13 informando que há informações inválidas no formulário.

Figura 17 – Formulário para cadastro ou edição de material ou equipamento.

A imagem mostra uma janela de software intitulada "Inserir Novo Material ou Equipamento". O formulário contém os seguintes campos e elementos:

- Nome*:** Campo de texto com o valor "Abraçadeira (18x24)".
- Grupo de Materiais:** Menu suspenso com o valor selecionado "HIDRÁULICO".
- Marca:** Campo de texto com o valor "Marca do Item".
- Valor (R\$)*:** Campo de texto com o valor "4,60".
- Unidade de Medida:** Menu suspenso vazio.
- Fornecedor:** Menu suspenso vazio.
- Imagem:** Área de visualização de uma imagem de uma abraçadeira metálica com um mecanismo de trava dourado. Abaixo da imagem há um botão "Selecionar Imagem".
- Descrição do material ou equipamento:** Área de texto grande e vazia.
- Legenda:** "Capos marcados com * são obrigatorios."
- Botões:** "Cancelar" e "Salvar" no canto inferior direito.

Fonte: O autor (2017)

Para acessar os formulários de cadastro/edição o usuário pode clicar com o botão direito do mouse sobre uma tabela e acessar o menu "Novo/Editar", através dos menus laterais (Figura 11a) ou através da barra de menu, no menu Editar.

4.7 Relatórios

O CIT SISAR-PI pode gerar alguns relatórios, tais como, relatórios de estradas e saídas dos municípios e comunidades, demonstração mensal (comparação saldo anterior

com o saldo atual) e relatórios de lançamentos contábeis. Esses relatórios podem ser exportados para CSV ou PDF.

Na figura 18 é mostrado um relatório contábil de entradas dos municípios. De acordo com a opção selecionada na lateral direita da figura, esse relatório pode mostrar informações de mensais, anuais ou por um período personalizado definido pelo usuário.

Figura 18 – Relatório de demonstração de entradas dos municípios.



Comunidade	Elétrico (R\$)	Hidráulico (R\$)	Sistema de Tratamento (R\$)	Valor Total (R\$)
BARRA D'ALCANTARA	542,95	141,81	0,00	684,76
MALHADA	271,12	12.705,01	1.774,83	14.750,96
MOURÕES	14,40	837,50	11,80	863,70
SANTO ANTONIO	0,00	55,63	0,00	55,63
SÃO MIGUEL	0,00	1,93	0,00	1,93
SERRA VELHA I	0,00	2,29	0,00	2,29
SISAR/PI	11,77	5,68	11,80	29,25
VERA MENDES	89,94	114,64	1,02	205,60

Filtros

Relatório Mensal

Ano: 2017

Mês: MAIO

Fonte: O autor (2017)

4.8 Backup

Pode ser acessado através do menu "Arquivo>Fazer Backup" ou através do atalho do teclado Ctrl+Shift+B. Esta função faz uma cópia de segurança de todas as informações salvas no banco de dados. Ao selecionar essa função, aparecerá uma janela para o usuário escolher onde deseja salvar o *backup*. Dependendo da quantidade de informações salvas no banco de dados, essa tarefa pode demorar bastante tempo, enquanto o *backup* não é finalizado, aparecerá uma barra de progresso (similar ao da Figura 16).

5 Avaliação do Sistema

Para comprovar sua eficiência, o CIT SISAR-PI foi avaliado sob três aspectos: adaptabilidade, desempenho e usabilidade. Sob o aspecto da adaptabilidade, foi analisado a capacidade de adaptação do sistema a diferentes plataformas, com relação ao desempenho, foi testado a eficiência na execução de suas principais funções, por fim, foi avaliado a facilidade de uso do sistema pelo cliente.

5.1 Adaptabilidade

O CIT SISAR-PI pode ser executado em qualquer sistema operacional que tenha a máquina virtual Java instalada, ele foi testado nas plataformas *Windows 7*, *Windows 10* e *Linux Ubuntu 15.10*, em todas as plataformas apresentou desempenho similar. O único ponto negativo quanto a este quesito foi a adaptação a diferentes tamanhos de telas. O sistema roda perfeitamente na resolução de tela 1366x768, em outras resoluções como por exemplo 1024x768 pode haver perda de conteúdo.

5.2 Desempenho

Quanto ao desempenho, foi testado o tempo que o sistema gasta na busca dos itens pertencentes a um determinado estoque e as exibe para o usuário. Foi utilizado um notebook com processador Intel core i3, com 2.4 GHz velocidade de processamento e 6gb de memória RAM. Foram feitas buscas em estoques com 30, 50, 100 e de 500 tipos diferentes de itens.

Em estoques com 30 itens, o tempo busca foi de, em média, 0,42 segundos, com 50 itens foi 0,58 segundo, com 100 itens foi 1,02 segundo e para busca de 500 itens, o tempo foi de 4,02 segundos.

Levando em consideração que atualmente o maior estoque da SISAR-PI possui 47 itens, pode-se considerar que os resultados foram satisfatórios, porém se porventura forem ocorrer situações em que o volume dos estoques for muito grande, esse tempo de espera pode ser incômodo para os usuários.

5.3 Uso do Sistema

Para avaliar a usabilidade, o sistema ficou a disposição do principal usuário do sistema, por cerca de duas semanas, após esse período de testes, foi aplicado um questionário para avaliação do sistema (Apêndice D). O questionário é composto por 14 perguntas, que

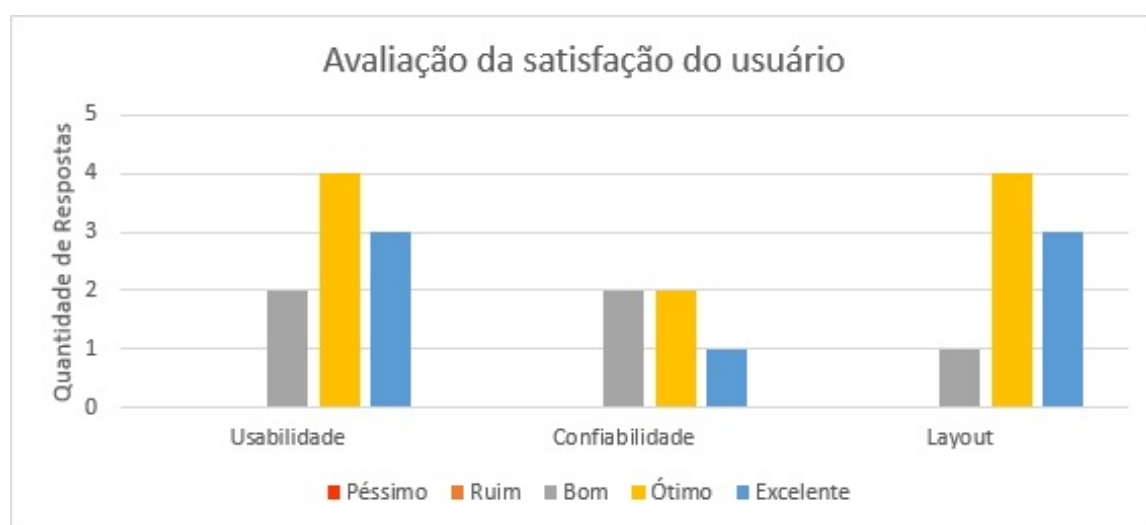
abordam os aspectos como: facilidade de uso, clareza das informações, segurança, visual, etc.

Para facilitar a compreensão das informações obtidas através do questionário, as perguntas foram divididas em três grupos:

- Usabilidade: Este critério avalia a forma como o usuário interage com o sistema.
- Confiabilidade: Avalia se as informações geradas no sistema realmente refletem as informações geradas pela empresa e se o sistema tem mecanismos para manter estas informações seguras.
- *Layout* Neste critério o sistema é avaliado levando em conta a forma como as informações são mostradas para o usuário.

Os resultados obtidos através do questionário são demonstrados, na Figura 19. É importante ressaltar que uma mesma pergunta pode se enquadrar em mais de um dos grupos citados acima, como por exemplo: a forma como os botões estão dispostos, pode ser avaliado tanto em relação a usabilidade quanto em relação ao *layout* do sistema.

Figura 19 – Gráfico de avaliação de satisfação do usuário.



Fonte: O autor (2017)

Como pode-se observar na Figura 19, aceitação o sistema foi boa, porém, ainda há melhorias a serem feitas e problemas a serem resolvidos. Segundo relato do usuário, a usabilidade é boa e o visual está agradável, porém, algumas informações dos relatórios não estão apresentadas de acordo com o esperado, alguns não estão refletindo informações verdadeiras e em certos momentos, demoram alguns segundos para serem exibidas, após a solicitação das mesmas.

Para solucionar os problemas discutidos nesta seção, deve ser revisado o código usado para pesquisar as informações no banco de dados. Quanto ao problema de demora na

exibição de informações, deve ser avaliado a possibilidade de otimizar os algoritmos de busca utilizados e/ou implementar *Threads*¹ para carregar dados em segundo plano.

¹ Parte menor que um processo pode se dividir para executar duas ou mais tarefas simultaneamente.

6 Conclusão

O presente trabalho mostrou o processo de desenvolvimento de um sistema de informação gerencial com foco no gerenciamento de estoques. A primeira versão do sistema foi concluída e está em fase de testes. Ela permite a seus usuários criar estoques de materiais e equipamentos de saneamento rural, permite que se transfira um ou mais itens dos estoques, consulta de movimentações de acordo com períodos selecionados, controlar as entradas e saídas dos estoques, gerar relatórios, exportar tabelas para CSV e PDF e criar contas de usuários protegidas por senha.

Durante os testes foram encontrados erros, principalmente relacionados a importação das planilhas de texto para o sistema. Na teoria, para ser realizado uma movimentação no estoque, é necessário que haja a quantidade que se deseja transferir disponível no estoque de destino, porém, nas planilhas usadas atualmente pela SISAR-PI podem ser encontrados registros de entradas em estoque que não foi dado baixa no estoque de destino, o que acaba gerando incoerências quando essas são lidas pelo sistema.

Para solucionar esse problema, uma alternativa seria editar manualmente os dados das transferências no banco de dados, para estes ficarem em conformidade com as informações contidas nas tabelas das planilhas. Isso resolve o problema parcialmente, os relatórios de entradas e saídas por exemplo são exibidos corretamente, pois precisam apenas do valor total de cada transferência. No entanto, caso se queira saber o total de itens do tipo Hidráulico que entraram em um estoque, em determinado período tempo, não há como informações precisas pois os itens que fazem parte de algumas transferências são desconhecidos.

Até o momento, o sistema, apesar de ainda conter falhas, satisfaz os objetos propostos. No entanto, processo de desenvolvimento ainda não foi concluído, algumas falhas que precisam ser corrigidas e funcionalidades aprimoradas. Semanalmente novos aprimoramentos são apresentados para o cliente, novos requisitos são discutidos e falhas são corrigidas.

6.1 Trabalhos Futuros

A SISAR-PI, além de gerenciar seus estoques, necessita também controlar o custo das viagens realizadas durante o atendimento das comunidades. Basicamente esse custo pode incluir: gasto de combustível, tempo de serviço dos funcionários, alimentação, etc.

Para trabalhos futuros, pretende-se adicionar um novo módulo para gerenciar esses custos de viagens, além disso, deseja-se aprimorar a adaptabilidade do sistema a telas com diferentes resoluções, solucionar *bugs*¹ existentes e modificar o sistema para permitir

¹ Erro no código fonte de um programa de computador

que várias máquinas possam acessar o servidor de banco de dados ao mesmo tempo.

Referências

- AESBE. *14 companhias de saneamento estão entre as maiores empresas do Brasil, segundo levantamento do Valor Econômico*. 2011. Disponível em: <<http://www.aesbe.org.br/conteudo/4468>>. Acesso em: 22 de Março de 2017. Citado na página 13.
- AUDY, J. L. N.; ANDRADE, G. K. de; CIDRAL, A. *Fundamentos de sistemas de informação*. 1. ed. São Paulo, SP: Bookman editora, 2005. Citado na página 18.
- BECK, K. *Extreme programming explained: embrace change*. [S.l.]: addison-wesley professional, 2000. Citado na página 19.
- BECK, K. et al. Manifesto for agile software development. 2001. Citado na página 20.
- BEZERRA, E. *Princípios de Análise e Projeto de Sistema com UML*. [S.l.]: Elsevier Brasil, 2015. Citado na página 21.
- CAGECE. *Sistema Integrado de Saneamento Rural*. 2011. Disponível em: <<https://www.cagece.com.br/2013-01-28-19-25-06/sisar>>. Acesso em: 21 de Março de 2017. Citado na página 13.
- CYSNEIROS, L. M.; LEITE, J. Requisitos não funcionais: da elicitação ao modelo conceitual. *PhDTese, PUC-RJ*, 2001. Citado na página 20.
- DEITEL, P.; DEITEL, H. *Java how to program: Early object*. 10. ed. New Jersey: Pearson, 2015. Citado 2 vezes nas páginas 22 e 23.
- ECLIPSE. *IDE Platforms*. 2017. Disponível em: <<https://eclipse.org/ide/>>. Acesso em: 24 de Maio de 2017. Citado na página 23.
- FOWLER, M. *UML Essencial: Um breve guia para linguagem-padrão para modelagem de objetos*. 3. ed. São Paulo, SP: Bookman Editora, 2007. Citado na página 21.
- FREEMAN, E.; FREEMAN, E. Use a cabeça: Padrões de projeto. Alta Books, Rio de Janeiro, RJ, 2009. Citado na página 23.
- FUNASA. *Panorama do Saneamento Rural no Brasil*. 2015. Disponível em: <<http://www.funasa.gov.br/site/engenharia-de-saude-publica-2/saneamento-rural/panorama-do-saneamento-rural-no-brasil/>>. Acesso em: 21 de Março de 2017. Citado na página 13.
- GAMMA, E. *Padrões de Projetos: Soluções Reutilizáveis*. [S.l.]: Bookman editora, 2009. Citado na página 23.
- GORDON, S. R.; GORDON, J. R. *Sistemas de Informação: Uma Abordagem Gerencial*. 3. ed. Rio de Janeiro, RJ: Grupo Gen-LTC, 2006. Citado 2 vezes nas páginas 16 e 17.
- GUDWIN, R. R. *Engenharia de Software: Uma abordagem prática*. 2. ed. [S.l.: s.n.], 2015. Citado 2 vezes nas páginas 21 e 22.

- HECKLER, M. et al. *JavaFX 8: Introduction by Example*. [S.l.]: Apress, 2014. Citado na página 23.
- ITEXT. *Your developer journey with iText begins here*. 2017. Disponível em: <<http://developers.itextpdf.com/>>. Acesso em: 27 de Março de 2017. Citado na página 38.
- LAUDON, K. C.; LAUDON, J. P. *Sistemas de informação gerenciais: Administrando a empresa digital*. São Paulo, SP: Pearson, 2004. Citado 2 vezes nas páginas 16 e 17.
- LECHETA, R. R. *Google Android: Aprenda a criar aplicações para dispositivos móveis com o Android SDK*. 5. ed. São Paulo, SP: Novatec Editora, 2015. Citado na página 22.
- MARUYAMA, H. *XML and Java: developing Web applications*. [S.l.]: Addison-Wesley Professional, 2002. Citado na página 22.
- MILANI, A. *MySQL: Guia do programador*. [S.l.]: Novatec Editora, 2007. Citado na página 24.
- MYSQL. *MySQL Workbench 6.3*. 2017. Disponível em: <<https://www.mysql.com/products/workbench/>>. Acesso em: 27 de Maio de 2017. Citado na página 24.
- NETBEANS. *NetBeans IDE - The Smarter and Faster Way to Code*. 2017. Disponível em: <<https://netbeans.org/features/index.html>>. Acesso em: 24 de Maio de 2017. Citado na página 23.
- ORACLE. *Why Use FXML*. 2014. Disponível em: <http://docs.oracle.com/javase/8/javafx/fxml-tutorial/why_use_fxml.htm>. Acesso em: 08 de Abril de 2017. Citado na página 23.
- _____. *The History of Java Technology*. 2017. Disponível em: <<http://www.oracle.com/technetwork/java/javase/overview/javahistory-index-198355.html>>. Acesso em: 16 de Maio de 2017. Citado na página 22.
- PRESSMAN, R.; MAXIM, B. *Engenharia de Software: Uma Abordagem Profissional*. 8. ed. São Paulo, SP: McGraw Hill Brasil, 2016. Citado 4 vezes nas páginas 17, 18, 19 e 20.
- REZENDE, A. F. d. A. D. A. *Tecnologia da informação: Aplicada a sistemas de informação empresariais*. 8. ed. São Paulo, SP: Editora Atlas S.A, 2011. Citado 2 vezes nas páginas 16 e 17.
- REZENDE, D. A. *Engenharia de software e sistemas de informação*. 3. ed. Rio de Janeiro, RJ: Brasport, 2005. Citado 2 vezes nas páginas 17 e 18.
- RIACHAONET. *Picos sedia o III Seminário sobre Saneamento Rural*. 2012. Disponível em: <<http://www.riachaonet.com.br/picos-sedia-o-iii-seminario-sobre-saneamento-rural/>>. Acesso em: 21 de Março de 2017. Citado na página 13.
- SILVA, A. M. R. da; VIDEIRA, C. A. E. *UML, Metodologias e Ferramentas CASE*. 1. ed. Lisboa: Centro Atlântico, 2001. Citado na página 22.
- SOARES, M. dos S. Metodologias ágeis extreme programming e scrum para o desenvolvimento de software. *Revista Eletrônica de Sistemas de Informação ISSN 1677-3071 doi: 10.21529/RESI*, v. 3, n. 1, 2004. Citado 2 vezes nas páginas 19 e 20.

SOMMERVILLE, I. *Engenharia de software*. 6. ed. São Paulo, SP: Addison Wesley, 2003. Citado 3 vezes nas páginas 17, 18 e 19.

STAIR, G. W. R. R. M. *Princípios de sistemas de informação: Uma abordagem gerencial*. 6. ed. São Paulo, SP: Pioneira, 2006. Citado na página 16.

SUEHRING, S. Mysql: a bíblia. In: *MySQL: a bíblia*. [S.l.]: Campus, 2002. Citado na página 24.

TELES, V. M. *Extreme programming*. São Paulo: Novatec, 2004. Citado na página 19.

Apêndices

APÊNDICE A – Documentação dos Casos de Uso

Tabela 5 – Documentação do caso de uso Cadastrar Materiais e Equipamentos.

Nome do Caso de Uso	Cadastrar Materiais e Equipamentos
Atores	Gerente
Resumo	Este caso de uso representa o processo de cadastro de um novo material ou equipamento
Fluxo Principal	
Ações do Ator	Ações do Sistema
1. Informar dados do item	
	2. Registrar item no banco de dados
Restrições/Validações	1. Os campos Nome e Valor devem ser validados
	2. Os campos Nome e Valor são obrigatórios

Tabela 6 – Documentação do caso de uso Adicionar Itens ao Estoque.

Nome do Caso de Uso	Adicionar Itens ao Estoque
Atores	Gerente e Funcionário
Resumo	Este caso de uso representa o processo de adição de um item no estoque central.
Fluxo Principal	
Ações do Ator	Ações do Sistema
1. Selecionar itens	
2. Informar quantidade que a ser transferida	
	3. Selecionar estoque central
	4. Adicionar itens ao estoque selecionado
Restrições/Validações	1. Deve ser selecionado pelo menos 1 item

Tabela 7 – Documentação do caso de uso Mostrar Itens do Estoque.

Nome do Caso de Uso	Mostrar Itens do Estoque
Atores	Gerente e Funcionário
Resumo	Este caso de uso representa o processo de listagem de itens de um determinado estoque
Fluxo Principal	
Ações do Ator	Ações do Sistema
1. Selecionar estoque	
	2. Exibir itens
Restrições/Validações	

Tabela 8 – Documentação do caso de uso Cadastrar Usuário.

Nome do Caso de Uso	Cadastrar Usuários
Atores	Gerente
Resumo	Este caso de uso representa o processo de cadastro de um novo usuário do sistema
Fluxo Principal	
Ações do Ator	Ações do Sistema
1. Informar informar dados do usuário	
	2. Validar dados informados
	3. Registrar usuário
Restrições/Validações	1. O campo <i>Login</i> e Senha são obrigatórios
	2. O <i>Login</i> deve ser único

Tabela 9 – Documentação do caso de uso Mostrar Itens do Estoque.

Nome do Caso de Uso	Fazer <i>Login</i>
Atores	Gerente e Funcionário
Resumo	Este caso de uso representa o processo de <i>login</i> no sistema
Fluxo Principal	
Ações do Ator	Ações do Sistema
1. Informar nome de usuário e senha	
	2. Autenticar usuário
Restrições/Validações	1. Deve ser selecionado um estoque
	2. Deve ser selecionado pelo menos 1 item

Tabela 10 – Documentação do caso de uso Transferir Itens.

Nome do Caso de Uso	Transferir Itens
Atores	Gerente e Funcionário
Resumo	Este caso de uso representa o processo de transferência dos itens entre estoques
Fluxo Principal	
Ações do Ator	Ações do Sistema
1. Executar caso de uso Mostrar Itens do Estoque	
2. Selecionar itens	
3. Informar quantidade que a ser transferida	
4. Selecionar estoque de destino	
	5. Associar transferência ao usuário logado
	6. Adicionar itens ao estoque selecionado
	7. Registrar transferência
Restrições/Validações	1. Deve ser selecionado um estoque
	2. Deve ser selecionado pelo menos 1 item

Tabela 11 – Documentação do caso de uso Atualizar Itens do Estoque.

Nome do Caso de Uso	Atualizar Itens do Estoque
Atores	
Resumo	Este caso de uso representa o processo de atualização de informações de um item do estoque
Fluxo Principal	
	1. Receber novos dados de um item
	2. Registrar novos dados do item
Restrições/Validações	

Tabela 12 – Documentação do caso de uso Exportar Tabela.

Nome do Caso de Uso	Exportar Tabela
Atores	Gerente
Resumo	Este caso de uso representa os passos necessários para exportar uma tabela do sistema
Fluxo Principal	
Ações do Ator	Ações do Sistema
1. Selecionar tabela	
	2. Exportar tabela
Restrições/Validações	Deve haver uma tabela selecionada
Fluxo Alternativo I - Exportar para CSV	
Ações do Ator	Ações do Sistema
	1. Converter tabela em arquivo CSV
Restrições/Validações	
Fluxo Alternativo II - Exportar para PDF	
Ações do Ator	Ações do Sistema
	1. Converter tabela em arquivo PDF
Restrições/Validações	

Tabela 13 – Documentação do caso de uso Mostrar Relatórios.

Nome do Caso de Uso	Mostrar Relatórios
Atores	Gerente
Resumo	Este caso de uso representa o processo de emissão de relatórios
Fluxo Principal	
Ações do Ator	Ações do Sistema
1. Selecionar período de tempo	
	2. Exibir relatório
Restrições/Validações	
Fluxo Alternativo I - Entradas dos Municípios	
Ações do Ator	Ações do Sistema
	1. Processar valor total das entradas de cada município
Restrições/Validações	1. Não deve ser exibido municípios que não tenham entradas
Fluxo Alternativo II - Entradas e Saídas dos Municípios	
Ações do Ator	Ações do Sistema
	1. Processar valor das entradas e saídas dos municípios separadas por grupo (Elétrico, Hidráulico, Sistema de Tratamento)
Restrições/Validações	1. Não deve ser exibido municípios que não tenham entradas
Fluxo Alternativo III - Demonstração Mensal - Entradas x Saídas	
Ações do Ator	Ações do Sistema
	1. Processar saldo anterior, entradas e saídas e saldo atual de cada município
Restrições/Validações	

Tabela 14 – Documentação do caso de uso Cadastrar Fornecedores.

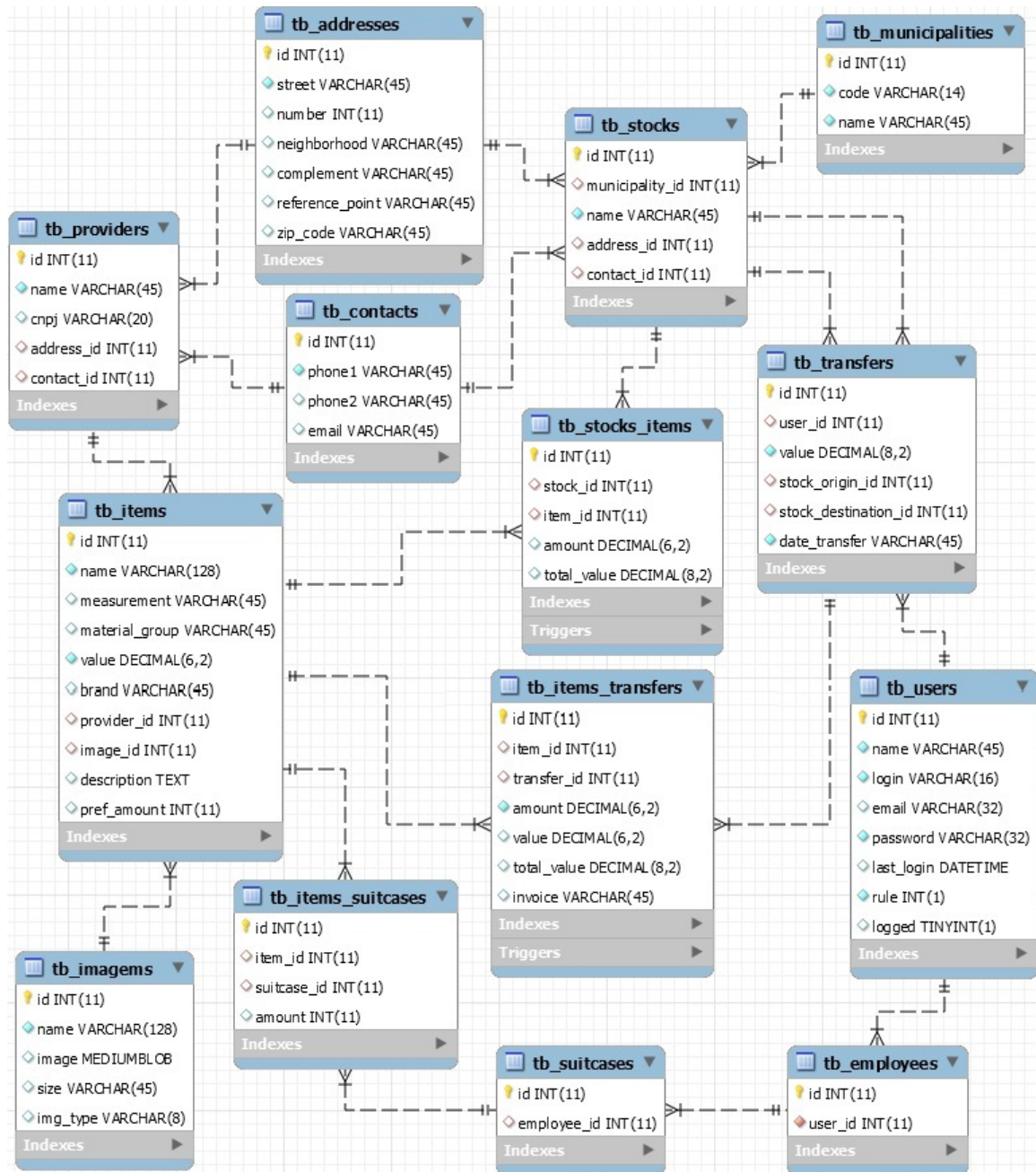
Nome do Caso de Uso	Cadastrar Fornecedores
Atores	Gerente
Resumo	Este caso de uso representa o processo de cadastro de novos fornecedores no sistema
Fluxo Principal	
Ações do Ator	Ações do Sistema
1. Informar dados do fornecedor	
	2. Registrar fornecedor no banco de dados
Restrições/Validações	1. O campo CNPJ deve ser validado
	2. O campo Nome é obrigatório

APÊNDICE B – Projeto Conceitual do Banco de Dados

- Municipalities(id, name)
- Addresses(id, street, number, neighborhood, complement, reference_point, zip_code)
- Contacts(id, phone1, phone2, email)
- Stocks(id, municipality_id, name, address_id, contact_id)
- Providers(id, name, cnpj, address_id, contact_id)
- Imagens(id, name, image, size, img_type)
- Items(id, code, name, measurement, material_group, value, brand, provider_id, image_id, description)
- Transfers(id, value, stock_origin_id, stock_destination_id, date_transfer)
- Items_Transfers(id, item_id, transfer_id, amount, total_value)
- Users(id, name, login, password, last_login, rule)
- Employees(id, user_id, salary, contact_id)
- Suitcases(id, employee_id)
- Items_Suitcases(id, item_id, suitcase_id, amount)

APÊNDICE C – Diagrama Entidade Relacionamento

Figura 20 – Diagrama entidade relacionamento do banco de dados.



Fonte: O autor (2017)

APÊNDICE D – Questionário de Avaliação do Sistema

Para que a nossa avaliação seja eficaz, por favor, atente para os detalhes:

- Caso você não saiba opinar sobre alguma das questões, deixe-a em branco.
 - Dê a nota que achar justa, não se preocupe se a nota for baixa.
 - Utilize a última questão para fazer qualquer crítica ao sistema.
1. As informações são facilmente encontradas? Por exemplo, informações dos estoques e relatórios são facilmente e rapidamente encontradas?
 Péssimo Ruim Bom Ótimo Excelente
 2. O sistema fornece informações completas e suficientes? Por exemplo, quando você solicita um relatório, o resultado é completo e suficiente?
 Péssimo Ruim Bom Ótimo Excelente
 3. O sistema fornece informações confiáveis? Por exemplo, as informações dos relatórios são confiáveis e precisas?
 Péssimo Ruim Bom Ótimo Excelente
 4. O sistema é seguro? Por exemplo, o usuário pode utilizar senhas de acesso ao sistema?
 Péssimo Ruim Bom Ótimo Excelente
 5. As informações são apresentadas de forma adequada? Por exemplo, os estoques e relatórios são mostrados de forma a destacar as informações mais importantes?
 Péssimo Ruim Bom Ótimo Excelente
 6. Os menus e botões estão dispostos de forma adequada? Por exemplo, eles são facilmente encontrados?
 Péssimo Ruim Bom Ótimo Excelente
 7. Os nomes utilizados no sistema são facilmente entendidos?
 Péssimo Ruim Bom Ótimo Excelente



**TERMO DE AUTORIZAÇÃO PARA PUBLICAÇÃO DIGITAL NA BIBLIOTECA
"JOSÉ ALBANO DE MACEDO"**

Identificação do Tipo de Documento

- Tese
 Dissertação
 Monografia
 Artigo

Eu, LUIS HOLANDA GOMES SOBRINHO,
autorizo com base na Lei Federal nº 9.610 de 19 de Fevereiro de 1998 e na Lei nº 10.973 de
02 de dezembro de 2004, a biblioteca da Universidade Federal do Piauí a divulgar,
gratuitamente, sem ressarcimento de direitos autorais, o texto integral da publicação
PROTÓTIPO DE SISTEMA DE INFORMAÇÃO PARA
GERENCIAMENTO DE TRANSAÇÕES DE ESTOQUES EM
COMPANHIAS DE SANEAMENTO RURAL

de minha autoria, em formato PDF, para fins de leitura e/ou impressão, pela internet a título
de divulgação da produção científica gerada pela Universidade.

Picos-PI 04 de Julho de 2017.

Luis Holanda Gomes Sobrinho
Assinatura

Luis Holanda Gomes Sobrinho
Assinatura