

MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
IF Sul-Rio-Grandense - Campus Passo Fundo
Técnico em Informática

RELATÓRIO DE PROJETO

Projeto de Pesquisa apresentado ao curso Técnico em Informática do Instituto Federal Sul-Rio-Grandense Campus Passo Fundo, como parte dos requisitos necessários à obtenção do certificado de Técnico em Informática.

José Henrique Paludo Giombelli

Passo Fundo, outubro 2019.

SUMÁRIO

1 INTRODUÇÃO.....	3
2. JUSTIFICATIVA.....	4
3 OBJETIVOS.....	5
3.1 OBJETIVO GERAL.....	5
3.2 OBJETIVOS ESPECÍFICOS.....	5
4 DESENVOLVIMENTO.....	6
4.1 MINI MUNDO.....	6
4.2 ANÁLISE DE REQUISITOS.....	6
.....	6
4.3 MODELO ENTIDADE RELACIONAMENTO.....	7
4.4 CLASSES E PACOTES.....	8
4.5 MÉTODOS.....	8

1 INTRODUÇÃO

Em dias atuais a tecnologia está cada vez mais presente em nossas vidas, onde tudo está cada vez mais informatizado e conectado, tendo cada vez mais tráfego e armazenamento de informações, fazendo com que as planilhas excel sejam substituídas por soluções mais simples, automatizando vários processos antes realizados por pessoas, trazendo agilidade e produtividade tanto no meio corporativo quanto no pessoal.

Como consequência esses avanços trouxeram uma crescente demanda ao mercado de software, fazendo surgir novas linguagens, tecnologias e profissionais. Dessa forma pensou-se em implementar um sistema para gerenciar o empréstimo de máquinas e equipamentos de uma empresa especializada na locação de equipamentos para construção civil, localizada na cidade de passo fundo, tendo em vista que este setor é muito presente e ativo, havendo pouquíssimas soluções para este tipo de serviço que é bastante requisitado.

O sistema tratará de cadastrar, alterar, excluir ou listar os clientes, funcionários e equipamentos, bem como registrar os empréstimos feitos pelos clientes, controlar o estoque e gerar um relatório das atividades da empresa, trazendo mais produtividade e lucratividade, visando um maior aumento na demanda destes serviços em um possível crescimento da empresa.

2. JUSTIFICATIVA

A implementação deste sistema, justifica-se pelo fato de não haver um controle informatizado e mais detalhado sobre as atividades da empresa, onde os dados são salvos em um caderno trazendo pouca praticidade, redundância nos dados armazenados e nenhuma segurança. Esse sistema visa justamente isso, mais agilidade e praticidade na consulta dos registros e atividades da empresa, tendo garantia de que não haverá perda ou redundância em seus registros.

Este tipo de solução é pouco presente no ramo da prestação de serviços, onde o proprietário deste estabelecimento não se preocupou anteriormente em informatizar sua empresa, por achar que esta solução não valeria a pena seu investimento e o período para retorno financeiro é muito grande, mas após uma breve explicação e demonstração de como o sistema funcionará, convenceu-se da implantação do sistema em seu estabelecimento, desmistificando os mitos e dúvidas que o cercavam, comprovando-se assim mais produtividade e lucratividade em sua empresa.

3 OBJETIVOS

3.1 OBJETIVO GERAL

O objetivo geral deste trabalho é informatizar o controle de empréstimos de máquinas e equipamentos para uma empresa do ramo da construção civil.

3.2 OBJETIVOS ESPECÍFICOS

- Realizar o levantamento de requisitos e entrevista com o proprietário e funcionários;
- Esquematizar o sistema nos diagramas de casos de uso;
- Implantar o modelo conceitual, lógico e físico do banco de dados;
- Implementar o projeto de fato, na linguagem definida.

4 Desenvolvimento

4.1 Mini Mundo

O trabalho consistirá na criação de uma aplicação desktop para cadastro de clientes, funcionários e equipamentos, onde o sistema controlará a quantidade de máquinas no estoque da empresa, bem como o empréstimo de máquinas de cada cliente e sua data de devolução. Além disso o sistema gerará um relatório das atividades da empresa e também a busca de clientes, funcionários ou equipamentos através de uma barra de pesquisa e listagem.

4.2 Análise de Requisitos

Requisitos Funcionais:

RF1-O sistema deverá cadastrar clientes com os seguintes dados: CPF, nome, RG, data de nascimento, sexo, número de telefone, E-mail, endereço, CEP, número, complemento, bairro, cidade e estado. ✓

RF2-O sistema deverá permitir o cadastro, listagem, remoção e atualização dos dados dos clientes, além de gerar um código para cada cliente, a partir do cadastro. ✓

RF3-O sistema deverá também cadastrar funcionários com os seguintes dados: código do funcionário, nome, CPF, cargo do funcionário, número de telefone e senha de acesso. ✓

RF4-O sistema deverá permitir o cadastro, listagem, remoção e atualização dos dados dos funcionários, também deve ser gerado pelo sistema um código para cada funcionário. ✓

RF5-O sistema também deverá cadastrar novas máquinas e equipamentos, com nome, tipo de maquinário e marca. ✓

RF6-O sistema deverá gerar um código também para cada equipamento cadastrado, além de poder ser feita sua listagem, remoção ou atualização. ✓

RF7-A cada mês deverá ser gerado um relatório das atividades da empresa.

RF8-O sistema deverá fazer também o controle de estoque da empresa, na devida parte de empréstimo de maquinários e equipamentos.

Requisitos Não Funcionais:

RNF1-Todas as funções de cadastrais deve ser feitas em uma única janela do programa. ✓

RNF2-Deverá ser mostrado uma notificação quando o estoque for inferior à 30%.

RNF3-Não poderá ser um cliente com status de equipamento pendente.

RNF4-Não deverá ser aceito o cadastro de um cliente, funcionário ou equipamento com códigos iguais.



4.3 Modelo Entidade Relacionamento

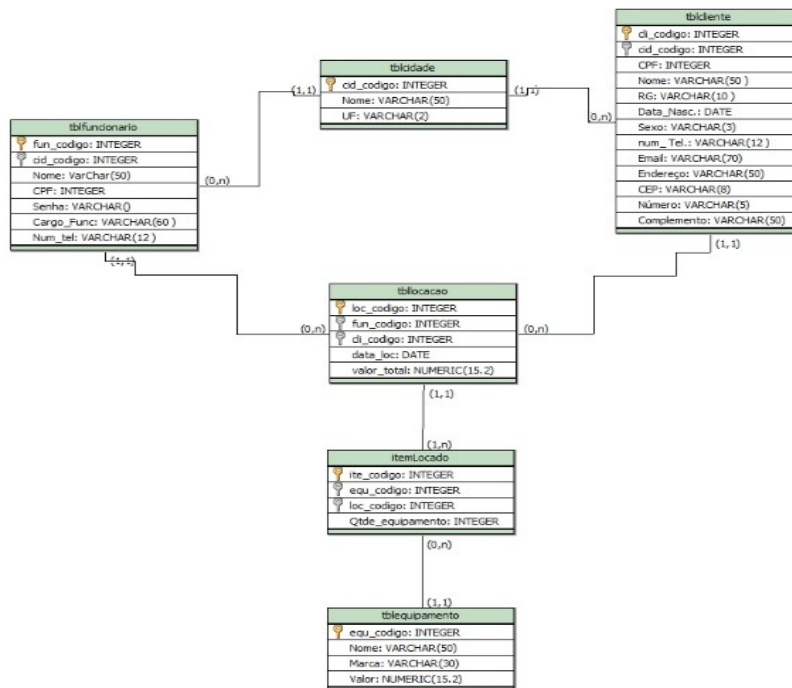


Figura 1. Modelo lógico do sistema

Após esta etapa inicial, foi realizado a modelagem do sistema através do modelo entidade-relacionamento ou modelo ER, essa parte é muito importante para definir como o sistema funcionará, bem como os atributos de cada entidade sua relação com outras tabelas através das chaves primárias e estrangeiras. A modelagem conceitual e lógica foi feita com o software brmodelo e está representada conforme a figura 1.

O modelo lógico conforme representado na figura 1, detalha como se dará o funcionamento do sistema, sendo este composto por seis entidades: cidade, cliente, funcionário, equipamento, item locado e locação, onde a relação das tabelas será descrita a seguir:

- Relação cidade x cliente: se dará por meio do código da cidade, onde um cliente poderá ter uma e apenas uma cidade cadastrada em seus registros, já a tabela cidade poderá ter zero ou muitos clientes cadastrados.
- Relação cidade x funcionário: de forma semelhante a relação anterior, um funcionário poderá ter apenas uma cidade cadastrada em seus registros sendo referenciada pelo código da tabela cidade, já a tabela cidade poderá receber zero ou muitos funcionários referenciados.

- Relação funcionário x locação: uma locação pode receber um e apenas um funcionário referenciado pelo seu código, já um funcionário pode ter zero ou muitas locações, sendo referenciado o código do funcionário para a presente tabela.
- Relação cliente x locação: um cliente pode ter zero ou muitas locações, mas na tabela locação pode ter zero ou muitos clientes, estes referenciados pelo seu código.
- Relação locação x item locado: nessa relação um ou muitos item locado podem ser associados com uma locação, já uma locação pode ter um e apenas um item locado.
- Relação item locado x equipamento: um item locado poderá receber um e apenas um equipamento, referenciado através do código do equipamento, já um equipamento poderá receber zero ou muitos itens locados.

Essa última associação entre as tabelas equipamento, locação e item locado é uma associativa, onde a tabela de item locado fará a “intermediação” por assim dizer, entre as tabelas locação e equipamento. Dessa forma será possível fazer a geração de relatórios de atividades da empresa e a quantidade de equipamentos emprestados para cada cliente em uma locação.

4.4 Classes e Pacotes

Para este projeto foi utilizado a geração de classes via java persistence API (JPA), onde as classes foram importadas diretamente do banco para o netbeans utilizando-se está API, juntamente com o driver do PostgreSQL. A estrutura do projeto foi separada por pacotes, onde:

- br.edu.ifsul.classes: este pacote conterá as classes do projeto;
- br.edu.ifsul.forms: este pacote conterá os formulários gerados através do form de amostra detalhe/mestre do java swing, onde fara toda a parte de inserção, atualização, remoção e listagem dos dados registrados.
- br.edu.ifsul.testes:este último pacote é destinado para teste de persistência de dados no banco, auxiliando na manutenção do crud e interface gráfica.

4.5 Métodos

Os métodos para manutenção e testes de persistência, foram criados a partir do mapeamento feito pelo JPA, trazendo mais agilidade e menos escrita de linhas de código. Abaixo é apresentado um trecho de código para persistir uma cidade.

```
EntityManagerFactory emf =
Persistence.createEntityManagerFactory("ProjetoLocatech_TI.2019PU");
EntityManager em = emf.createEntityManager();
Tblcidade cid = new Tblcidade();
```



```
cid.setNome("Veranopolis");
cid.setUf("RS");
em.getTransaction().begin();
em.persist(cid);
em.getTransaction().commit();
em.close();
emf.close();
```

Conforme o código apresentado acima é criado um novo objeto do tipo cidade, onde nas duas linhas abaixo é setado o nome e unidade federativa da cidade, logo abaixo, as informações serem apresentadas inicia-se uma nova transação com o banco para persistir os dados dos atributos cidade, após fecha a transação com o banco de dados e os dois últimos comandos fecha a conexão.

4.6 Formulários

Os formulários gerados pelo form de amostra mestre/detalhe farão parte da inserção, listagem, edição e remoção dos dados da aplicação. A seguir é mostrado uma das telas para manutenção crud da classe cidade.

Cid Codigo	Nome	Uf
1	Passo Fundo	RS
3	veranopolis da serra	RS
4	São Jorge	RS

Cid Codigo: null
Nome: null
Uf: null

Novo Remover Atualizar Salvar

Figura 2. Tela de Listagem da Classe Cidade

Nessa tela, além da listagem das cidades, será possível adicionar uma nova cidade através do botão novo. Clicando em uma cima de uma linha na tabela de listagem é possível editar um campo cadastrado, caso deseje remover basta selecionar o campo e remover o campo desejado, por fim o botão salvar permite salvar todas as alterações feitas na interface enviando-as ao banco de dados imediatamente.