



SISTEMA PARA GESTÃO DE LOCAÇÃO DE EQUIPAMENTOS

Luan H. AVELAR¹; Perses de VILHENA²; Vinícius L. NASCIMENTO³; Paulo C. dos Santos⁴

RESUMO

O presente trabalho descreve o processo de desenvolvimento de um software para gerenciar empresas de locação de ferramentas e equipamentos para construção civil. Foram utilizadas tecnologias para o desenvolvimento de aplicações web com acesso a banco de dados. Adotou-se um processo de software iterativo incremental, com a participação do cliente ao longo das etapas de implementação. Ao final obteve-se um produto de acordo com as necessidades do mesmo, com interfaces amigáveis e processo básico de segurança no acesso às funcionalidades e dados do sistema.

Palavras-chave: Processo de Software; Gerenciamento de Locação; Tecnologia Web.

1. INTRODUÇÃO

Para tornar possível o gerenciamento de clientes e transações de locações de equipamentos para construção civil, empresas de prestação de serviços de locação necessitam de aplicações que possam controlar e agilizar os processos internos. Algumas empresas às vezes adotam o uso planilhas eletrônicas, que apesar de interessantes, na maior parte dos casos, não supre totalmente a necessidade das empresas de forma prática e otimizada. Este tipo de software consome tempo e recursos humanos que poderiam ser mais bem empregados.

Softwares para gerenciamento mostram-se uma alternativa ideal e otimizada para tais aplicações, podendo desenvolver e implementar funções específicas ao uso a que se destina, como cadastro e gerenciamento de clientes, estoques, prazos e valores de locação, etc. Tal praticidade previne erros e utiliza tempo e recursos humanos de forma eficiente, além de tratar os dados de forma mais técnica, organizada e segura.

Trata-se de uma área ampla, o desenvolvimento de um software para o gerenciamento de uso geral, para vários segmentos de empresas, torna-se uma tarefa árdua devido à vasta gama de casos de uso. Tal dificuldade torna mesmo os sistemas mais específicos produtos de softwares caros e às vezes de difícil acesso, principalmente para pequenos empresários.

Pensando em empresas de pequeno porte, especificamente empresas que trabalham com locação, este projeto busca implementar um software prático e de baixo custo para gerenciamento

¹ Instituto Federal de Educação, Ciência e Tecnologia do Sul de Minas Gerais – IFSULDEMINAS – Campus Muzambinho. Muzambinho/MG - E-mail: avelarluan@gmail.com

² Instituto Federal de Educação, Ciência e Tecnologia do Sul de Minas Gerais – IFSULDEMINAS – Campus Muzambinho. Muzambinho/MG - E-mail: persesvilhena01@gmail.com

³ Instituto Federal de Educação, Ciência e Tecnologia do Sul de Minas Gerais – IFSULDEMINAS – Campus Muzambinho. Muzambinho/MG - E-mail: vinilam.nascimento@gmail.com

⁴ Instituto Federal de Educação, Ciência e Tecnologia do Sul de Minas Gerais – IFSULDEMINAS – Campus Muzambinho. Muzambinho/MG - E-mail: paulo.santos@muz.ifsuldeminas.edu.br



de clientes, estoque e locações, que possa ser utilizado para controlar prazos e valores das locações, além de manter dados acessíveis para consultas.

2. FUNDAMENTAÇÃO TEÓRICA

Ao desenvolver um projeto de software busca-se utilizar uma linguagem de fácil compreensão e com recursos adequados para suprir a necessidade do desenvolvedor, para que seja eficiente e de baixa complexidade durante o desenvolvimento.

Segundo Tiobe (2017), a linguagem de programação conhecida como Java é, atualmente, a linguagem de programação muito utilizada em escalas globais no desenvolvimento de softwares e aplicações, estando em uso por, aproximadamente, 14,93% dos desenvolvedores. De acordo com Oracle (2010), “com *Java Enterprise Edition* o desenvolvimento de aplicações pode ser rápido e prático. O foco da plataforma é prover os desenvolvedores com um conjunto expressivo de *APIs*, reduzindo tempo de desenvolvimento, complexidade e melhorando a performance”.

3. MATERIAIS E MÉTODOS

Para o desenvolvimento do software utilizou-se o modelo espiral, buscando a participação do cliente no processo de desenvolvimento e gerando assim resultados mais próximos do esperado. O cliente participou do processo de levantamento dos requisitos necessários para atender as suas necessidades, no processo foi considerada também a viabilidade de implementação. O resultado final foi a obtenção de um conjunto de funcionalidades desejadas pelo cliente e viável de ser implementadas pela equipe de desenvolvedores.

Em posse dos dados fornecidos pelo cliente e conhecendo as suas necessidades, foi utilizada uma linguagem de modelagem que segundo UML (2005), em sua documentação, contribui para especificar, visualizar e documentar modelos de sistemas, incluindo sua estrutura e *design*, foi utilizada na criação de diagramas que funcionam como um projeto em alto nível de software, tais como Diagrama de Caso de Uso conforme figura 1, Diagrama de Classe, entre outros.

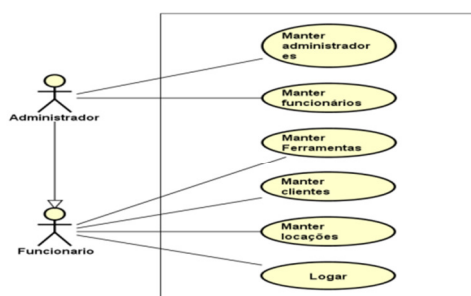


Figura 1: Diagrama de caso de uso



Na fase de implementação do software, utilizou-se a IDE (*Integrated Development Environment*) IntelliJ Idea, o *framework* JSF (*Java Server Faces*) que possibilitaram a execução do código Java em um servidor web *Apache Tomcat*, tornando o alcance mais abrangente para garantir o acesso fácil ao sistema, por meio de requisições web. O tratamento e organização de dados foi facilitado pela *JPA* (*Java Persistence Application Programming Interface*) e *Hibernate*, ferramentas que tratam da criação e manipulação do banco, tornando-o mais próximo do paradigma de programação orientado à objetos.

Ao desenvolver uma aplicação web, é importante que a interface com o usuário final seja agradável e de fácil uso, o que foi possível alcançar com a utilização dos *frameworks* *Bootstrap* e *Primefaces*, recursos utilizados para a produção de interfaces de usuários, que também facilitaram desenvolvimento, pois possuem bons recursos e os mesmos se mostram fáceis de serem utilizados pela equipe de desenvolvimento.

Na medida em que ocorreu o desenvolvimento do sistema, resultados parciais foram apresentados ao cliente, o qual sugeriu adaptações e mudanças que tornaram mais adequadas às suas necessidades até que, em consenso com a equipe de desenvolvimento, chegou-se a versão final, que foi testada na prática. Foram aplicados os conceitos de testes unitários de integração.

4. RESULTADOS E DISCUSSÕES

Figura 2: Página de Locação

Figura 3: Estado da Ferramenta

Concluído o processo de desenvolvimento, obteve-se um software que possibilita manter registros de ferramentas, clientes, histórico de locações realizadas e pendentes, entre outras. Usuários credenciados no sistema podem gerenciar as locações da empresa, conforme figura 2. Visou-se a flexibilidade e praticidade do sistema, outros administradores podem ser cadastrados, ou categorias podem ser criadas para permitir acesso apenas para certas funcionalidades do sistema, processo esse que pode ser efetuado pelos próprios usuários, por meio de uma interface amigável,



desde que tenha privilégios no sistema. O sistema final também permitiu gerenciar a disponibilidade de cada ferramenta, conforme figura 3, possibilitou aos usuários definir *status* para as mesmas, no intuito de suprir as necessidades e para atualizar o registro das ferramentas indicando quais estão disponíveis para locação, as indisponíveis, aquelas em manutenção, etc.

A segurança também foi levada em conta por meio da limitação do acesso, permitindo que apenas administradores, ou aqueles por eles autorizados, acessem funcionalidades restritas ou alterem dados sensíveis, acessíveis apenas por processo de autenticação com senha criptografada.

5. CONCLUSÕES

O sistema web desenvolvido ofereceu a praticidade necessária de acordo com as decisões tomadas juntamente com o cliente. Enquanto a linguagem Java ofereceu praticidade no desenvolvimento. Observou-se que o desempenho da aplicação poderá ainda ser melhorado nas próximas versões. Ainda assim, como resultado final, foi disponibilizado um sistema que se mostrou de uso prático e fácil, que poderá agilizar o processo de gerenciamento na área a que se destinada.

REFERÊNCIAS

ORACLE. **The Java EE 5 Tutorial**. 2010. Disponível em:

<<https://docs.oracle.com/javaee/5/tutorial/doc/bnaaw.html>>. Acesso em: 20 jun. 2017.

TIOBE. **Programming language Kotlin jumps into the top 50**. 2017. Disponível em:

<<https://www.tiobe.com/tiobe-index/>>. Acesso em: 20 jun. 2017.

UML. **What is UML**. 2005. Disponível em: <<http://www.uml.org/what-is-uml.htm>>. Acesso em: 23 jun. 2017.