

MONITORAMENTO DE PLATAFORMAS DE POÇOS DE PETRÓLEO ATRAVÉS DE DISPOSITIVOS MÓVEIS

C. M. Costa¹, C. R. M Leite², I. C. Sá², P. F. Ribeiro Neto², B. G. Araújo³, R. A. M. Valentim³ e A. M. G. Guerreiro³

¹Núcleo de Estudos em Agroecologia - NEA, Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte, Ipangaçu-RN.

²Grupo de Engenharia de Software - GES, Universidade do Estado do Rio Grande do Norte, Mossoró-RN.

³Laboratório de Sistemas Inteligentes – LABSIS, Universidade Federal de Rio Grande do Norte, Natal-RN.
clayton.maciel@ifrn.edu.br¹

Artigo submetido em maio/2011 e aceito em fevereiro/2012

RESUMO

A expansão da Web, o advento das novas tecnologias de informação e comunicação e a massificação dos dispositivos móveis tornaram possíveis novas possibilidades de gerenciamento, interações e compartilhamento de informações. Na área petrolífera, sistemas de monitoramento utilizando dispositivos móveis vêm sendo utilizados para otimização de processos, monitoramento de dados, geração de

relatórios, geração de alertas e identificação de falhas em tempo-real nas plataformas petrolíferas. Contudo, ainda são poucos os sistemas que identificam problemas em tempo-real, já que a maioria não emprega técnicas de sistemas de tempo-real. Este artigo propõe o sistema SiMoS, um sistema de tempo-real para o monitoramento remoto de plataformas de poços de petróleo através de dispositivos móveis.

PALAVRAS-CHAVE: Sistema de Monitoramento, Sistema de Tempo-Real, Plataforma de Poços de Petróleo, Dispositivos Móveis.

MONITORING FOR OIL WELLS PLATFORMS THROUGH MOBILE DEVICES

ABSTRACT

The expansion of the Web, the advent of new information and communication technologies and popularization of mobile devices made possible new aspects for management, interaction and information sharing. In petroliferous area, monitoring systems on mobile devices have been used for process optimization, data monitoring, reporting, alarm generation and

identification of failures in real-time on oil wells platforms. However, there are few systems that identify efficiently problems due the fact they do not use techniques of real-time systems. This paper proposes the SiMoS system, a real-time system for remote monitoring oil wells platforms through mobile devices.

KEY-WORDS: Monitoring System, Real-Time System, Oil Wells Platform, Mobile Devices.

MONITORAMENTO DE PLATAFORMAS DE POÇOS DE PETRÓLEO ATRAVÉS DE DISPOSITIVOS MÓVEIS

INTRODUÇÃO

O desenvolvimento de aplicações para dispositivos móveis têm crescido bastante devido aos inúmeros benefícios dessas tecnologias, entre eles, alta disponibilidade e escalabilidade de informações. A utilização de dispositivos móveis tem sido empregada em diversas áreas, tais como: sistemas de controle de tráfego aéreo, sistemas de monitoramento, sistemas de manufatura e sistemas de sensoriamento eletrônico, sistema de monitoramento médico proporcionando as suas aplicações a executarem processos automatizados com novas concepções de gerenciamento e controle dos dados. Desta forma, esses dados precisam ter consistência lógica e temporal possuindo restrições de temporais em suas tarefas, sendo estas, características de Sistemas de Tempo-Real (STR). Os Sistemas de Tempo-Real são sistemas que promove o tratamento de dados baseado em prazos temporais para a execução de suas tarefas [1, 2, 3].

Na área petrolífera, sistemas de monitoramento utilizando dispositivos móveis vêm sendo utilizados para otimização de processos, monitoramento de dados, geração de relatórios, geração de alertas e identificação de falhas em tempo-real nas plataformas petrolíferas que geralmente se encontram distantes de suas bases gerenciais, como em [4, 5, 6, 7, 8]. Dessa forma, dificulta a detecção antecipada dos possíveis problemas, pois na maioria das vezes os mesmos ocorrem sem a tomada decisão eficaz.

Contudo, são poucos os sistemas que identificam problemas de forma eficiente, já que a maioria não emprega técnicas de sistemas de tempo-real. Além disso, no geral, sistemas para plataforma petrolífera apresentam alto custo de implantação e não utilizam tecnologias Web. Dessa forma, neste artigo, apresenta-se o sistema **SiMoS**, um sistema de tempo-real para o monitoramento remoto de plataformas de poços de petróleo através de dispositivos móveis.

Este artigo está organizado como se segue. A Seção II apresenta os Trabalhos relacionados. A Seção III discute a fundamentação teórica sobre sistemas de tempo-real e desenvolvimento de aplicações para dispositivos móveis. A Seção IV descreve a arquitetura, especificação e implementação do sistema **SiMoS**. A Seção V discute avaliações e experimentos do **SiMoS**. E finalmente, a Seção VI apresenta as conclusões e trabalhos futuros.

TRABALHOS RELACIONADOS

A avançada tecnologia de dispositivos móveis a custos acessíveis e a necessidade de compartilhamento de dados com alta disponibilidade e escalabilidade, em diversos formatos padronizados, motivaram o desenvolvimento de sistemas instalados sobre dispositivos móveis em diversas áreas de aplicações.

Na área petrolífera, ainda são poucas as propostas que surgem na literatura, em [4, 5], propõem-se um sistema de monitoramento que envia mensagens de alertas para o telefone celular informando alguma anormalidade ocorrida em alguma plataforma de petróleo distante de sua base gerencial. Em [8] é proposto um sistema para o monitoramento distribuído de plataformas automatizadas de poços de petróleo. Nesses sistemas, a eficiência para a detecção de problemas, intermitentes ou não, não é garantida, já que técnicas de sistemas inteligentes e técnicas de sistemas de tempo-real não são empregadas.

Em [9, 10], são propostos sistemas de monitoramento em tempo-real de plataformas de poços de petróleo. Tais sistemas foram implementados por um sistema *SCADA (Supervisory Control And Data Acquisition)*, o qual é proprietário e dedicado. Os sistemas propostos apresentam uma abordagem intrusiva, por utilizarem softwares e hardwares complementares proprietários para a realização de suas tarefas. Outro fato importante é que em [9], seus dados não são armazenados em banco de dados, apresentando um sistema de busca ineficiente e com pouca flexibilidade.

Em [6, 7], é apresentado um sistema que utiliza técnicas de tempo-real. O sistema possui uma interface Web e os dados são armazenados no *SGBD PostgreSQL*, facilitando a busca e o processamento das informações. Contudo, não foi implementado para dispositivos móveis e sim para uso em *PC Desktop* através de *Applet Java*, o que torna a interface lenta com o usuário dificultando o gerenciamento preciso, rápido e eficaz.

FUNDAMENTAÇÃO TEÓRICA

As características dos Sistemas de Tempo-Real (STR) e as particularidades do desenvolvimento de aplicações sobre Dispositivos Móveis constituem a parte essencial para a compreensão da especificação e implementação do sistema **SiMoS**. Nesta Seção apresentam-se os principais conceitos de STR e de sistemas sobre Dispositivos Móveis.

- **STR**

Os Sistemas em Tempo-Real (STR) são sistemas que reagem a estímulos oriundos de um ambiente dentro de prazos específicos. Tipicamente, são inerentes a aplicações críticas, no qual pode envolver o risco de vidas humanas, perdas econômicas ou até mesmo danos no meio ambiente [11].

Um STR consiste de um sistema controlador e um sistema controlado, como ilustrado na Figura 1. O sistema controlado é o ambiente que disponibiliza dados para o sistema controlador. O sistema controlador atua conforme os dados que vêm do sistema controlado.

Os dados podem ser obtidos de um sensor ou de uma rede de sensores. Em sistemas críticos é indispensável que o sistema controlador obtenha informações sobre o estado real do ambiente de forma que os resultados obtidos sejam os esperados, de outra forma, estes podem ser catastróficos.

Por isso, os dados com restrição de tempo não dependem apenas dos resultados lógicos, mas também do tempo em que os resultados são produzidos. A Figura 2 apresenta um sistema de freios computadorizado para automóveis, esse sistema é composto de um atuador, um sensor de velocidade. O freio no pneu do automóvel tem que ser acionado numa restrição de tempo quando um usuário pisa no pedal.

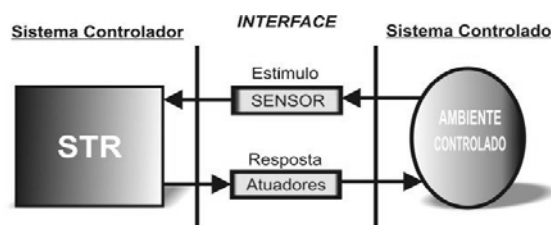


Figura 1 – Sistema controlador e sistema controlado.

Geralmente, o desenvolvimento de um sistema crítico demanda alto custo, devido às suas características, que são: grande complexidade e dimensão, concorrência ou simultaneidade nas ações, uso de dispositivos de entrada e saída especiais, segurança, confiabilidade e determinismo temporal. Contudo, utilizando extensões de aplicações já existentes e ferramentas livres e de código-aberto é possível desenvolver sistemas com baixo custo e ainda garantir desempenho, como é o caso do **SiMoS**.

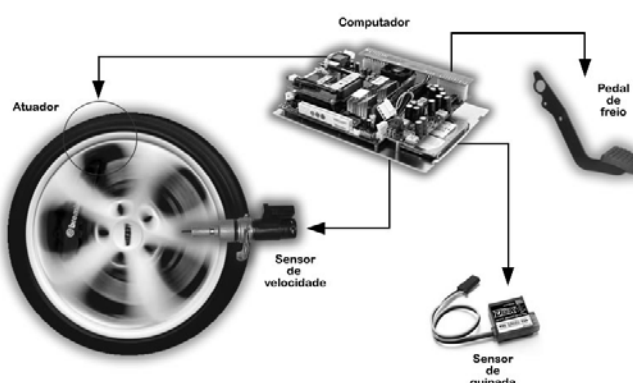


Figura 2 – Freio computadorizado.

• DISPOSITIVOS MÓVEIS

Segundo [12], um dispositivo móvel é qualquer equipamento ou Periférico que possa ser transportado com conteúdo e esteja acessível em qualquer lugar, são exemplos de dispositivos móveis: *papers*, telefones celulares, *smartphones*, *PDA*s (Assistentes Digitais Pessoais), aparelhos domésticos que suportam acesso a uma rede, como: microondas, geladeiras, e assim por diante.

Os dispositivos móveis são formados basicamente por um processador, uma memória volátil, geralmente RAM ou SRAM, um dispositivo de armazenamento, sendo a memória FLASH bastante utilizada, e interfaces, que se apresentam em diversas dimensões: 101x65, 96x65, 154x90,

320x240, etc. Contudo, ao contrário de um computador tradicional, os dispositivos móveis, em sua grande maioria, possuem recursos computacionais limitados, o que torna a especificação e implementação de aplicações desafiadora para este tipo de plataforma.

Atualmente, muitos dispositivos de comunicação portátil, como celulares, apresentam suporte à plataforma Java para o desenvolvimento de aplicações. Tais dispositivos implementam em seus sistemas, uma máquina virtual que roda aplicações JME (*Java Micro Edition*). O JME inclui interface com o usuário, segurança robusta e protocolos de comunicação em rede [13].

O JME oferece um conjunto poderoso de recursos para a implementação de aplicações críticas [13]. Por exemplo, técnicas de tolerância a falhas, garantindo alta confiabilidade, segurança, disponibilidade e consistência dos dados das aplicações. Neste trabalho, o sistema **SiMoS**, que se trata de um sistema crítico, busca se beneficiar ao máximo desses recursos, como discutido na próxima seção.

SISTEMA PROPOSTO

O **SiMoS** é um sistema em tempo-real para o monitoramento remoto de plataformas de poços de petróleo através de dispositivos móveis. Conforme apresentado na Figura 3, esse sistema consiste de uma rede de sensores (ambiente de negócio e obtenção de dados), de uma aplicação Web de gerenciamento orientada a serviços (**SiMoS – Servidor** - ambiente de supervisão e controle) e de uma aplicação Web cliente (**SiMoS – Cliente** - ambiente de visualização e controle utilizando dispositivos móveis). Como está ilustrado na arquitetura do **SiMoS**, um fluxo contínuo e intenso de dados é gerado por uma rede de sensores instalados e configurados em uma ou mais plataforma de poços de petróleo. Os dados obtidos são tratados e armazenados em um SGBD convencional obedecendo a prazos específicos de tempo (1 segundo). Por fim, os dados tratados podem ser visualizados e analisados em forma de gráficos através da aplicação servidora (**SiMoS - Servidor**) ou através de dispositivos móveis (**SiMoS - Cliente**).

O **SiMoS** apresenta uma junção de características de sistemas de tempo-real, com estímulos periódicos e respostas, de sistemas distribuídos (cliente-servidor), onde o servidor possui a prerrogativa de garantir o serviço enquanto o cliente de consumi-lo, e de sistemas orientados a serviços (SOA), onde plataformas e tecnologias de diferentes implementações interoperam.[15].

- **REDE DE SENSORES**

Na área petrolífera, redes de sensores são muito utilizadas para as mais diversas finalidades. Um caso de uso bastante interessante é sua utilização em plataformas de produção de poços de petróleo, as quais fazem a manutenção da produção de petróleo em poços após sua perfuração. Como explicaremos a seguir, uma rede de sensores comumente utilizada em plataformas de produção se compõe por um sensor de pressão, um sensor no peso suspenso, um sensor na chave hidráulica e um sensor na chave flutuante. Como apresentado em [14], esses sensores abrange

todas as operações de uma plataforma de produção. Tal rede foi utilizada nos experimentos do **SiMoS**. Contudo, este sistema pode ser utilizado para quaisquer redes de sensores, inclusive, em outros contextos.

O sensor de pressão monitora a pressão de uma bomba hidráulica responsável por jogar fluido dentro de um poço de petróleo. Este sensor, como ilustrado na Figura 4, é formado por um diafragma que é pressionado quando uma pressão passa no local. A força sobre o diafragma produz uma deformação sobre os fios da armadura provocando variações de resistência ôhmica. Tais variações são convertidas em valores de corrente elétrica na ordem de 4 a 20 mA (*miliampéres*).

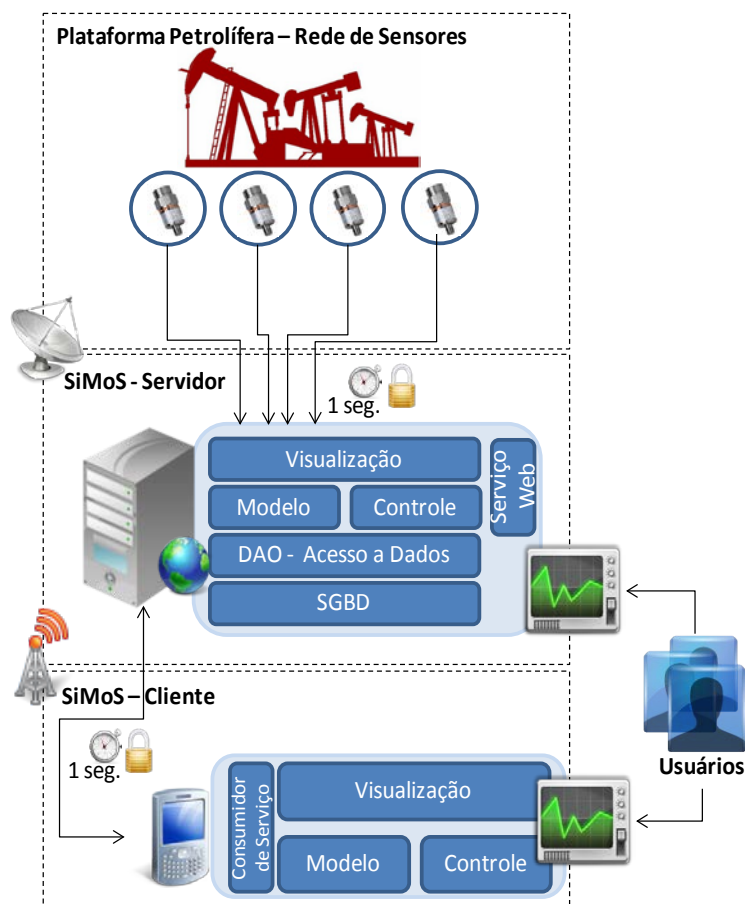


Figura 3 – Arquitetura do sistema SiMoS.

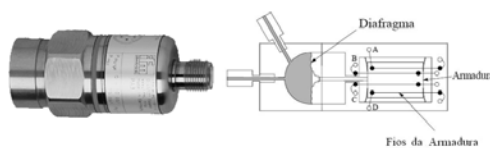


Figura 4 - Sensor de pressão utilizando o princípio do diafragma.

O sensor da chave hidráulica e o sensor da chave flutuante são utilizados somente quando chaves mecanizadas são usadas para apertar ou folgar tubos da coluna que forma o poço. Estes sensores apresentam a mesma característica do sensor de pressão que foi explicado anteriormente.

O sensor no peso suspenso chamado de célula de carga, como mostra a Figura 5, marca o peso da coluna ou de um tubo que forma o poço. A célula de carga faz marcações de corrente elétrica baseando-se na variação da resistência ôhmica de um sensor instalado na própria. Por sua vez, o sensor, denominado extensômetro, detecta a deformação da célula de carga e produz uma variação a resistência ôhmica. Assim, quando o cabo de sustentação da coluna é submetido a uma força sofre uma dilatação que é detectada pelo extensômetro e a célula de carga e transforma a variação de resistência em variação de corrente elétrica na ordem de 4 a 20 mA.

A rede de sensores envia sinais de corrente elétrica regularmente a cada segundo, os quais são capturados por uma placa conversora elétrica que implementa um software supervisor, o qual converte pulsos elétricos em dados computacionais. Esta placa, por sua vez, transmite os dados capturados através de uma antena de rádio ou satélite a uma frequência estática, dedicada e segura.

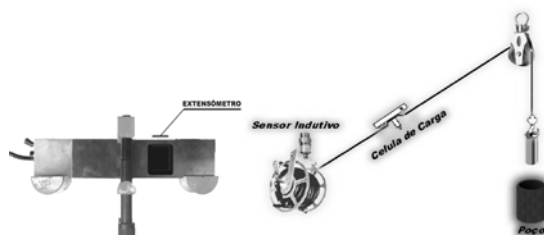


Figura 5 – Célula de carga utilizando o princípio da resistência elétrica.

- **SIMOS - SERVIDOR**

O **SiMoS - Servidor** é responsável pelo gerenciamento, tratamento e persistência dos dados no Sistema Gerenciador de Banco de Dados (SGBD). No sistema, os dados são disponibilizados através de Serviços Web, os quais podem ser acessados diretamente por os usuários ou acessados por dispositivos móveis que contenham a versão cliente do **SiMoS (SiMoS - Cliente)** e que possuam tecnologia necessária para acesso à internet, como 3G, Wi-Fi, GPRS, entre outros.

Ao disponibilizar as informações a partir de serviços na Web, o **SiMoS** também assume características da arquitetura de sistemas distribuídos cliente-servidor, onde o servidor possui a prerrogativa de garantir o serviço enquanto o cliente de consumi-lo. Segundo [15], em uma arquitetura cliente-servidor, uma aplicação é modelada como um conjunto de serviços fornecidos pelos servidores e um conjunto de clientes que usam esses serviços. Conseqüentemente, ao permitir o consumo desses serviços a partir de outros programas, o **SiMoS - Servidor** reúne características inerentes às arquiteturas Orientadas a Serviços (SOA), pois permite que plataformas e tecnologias de implementação diferentes possam ser usadas em diferentes partes das empresas para se interoperarem.

- **SIMOS - CLIENTE**

O *SiMoS - Cliente* é um aplicação livre desenvolvida para dispositivos de comunicação móveis, celulares, utilizando primitivas de gerenciamento temporais. O *SiMoS - Cliente* garante características de sistemas de tempo-real admitindo a responsabilidade de requisitar novas informações da aplicação servidora a cada segundo, a partir de uma conexão segura, e disponibilizar os mesmos em um programa específico que executará em uma plataforma distinta em um dispositivo móvel.

- **ESPECIFICAÇÃO E INTERFACES**

O *SiMoS* foi projetado visando qualidade, escalabilidade e um baixo acoplamento, garantido assim uma maior flexibilidade tanto para a manutenção quanto para o aprimoramento do mesmo. Para isso foi adotado como base os padrões de projeto MVC (*Model, View, Control*) e DAO (*Data Access Object*). O MVC é um padrão que tem como objetivo dividir a aplicação em três camadas, sendo a camada de Modelo responsável por representar o domínio da aplicação e as regras de negócio, a camada de Visualização para representar a interface com os usuários e seus possíveis eventos e a camada de Controle para gerenciar o fluxo da aplicação. O padrão DAO, por sua vez, foi incorporado para garantir o melhor acesso, persistência e consulta aos dados do SGBD.

O sistema *SiMoS* disponibiliza diversas interfaces gráficas, as quais estão divididas em: Interfaces *SiMoS-Servidor* e Interfaces *SiMoS-Cliente*. As Interfaces *SiMoS-Cliente*, após a tela de autenticação do usuário, estão divididas em: menu, controle e listagem de poços, análise gráfica em tempo-real e análise histórica. A Figura 6(a) apresenta a interface de menu, responsável por dar acesso ao usuário para cadastro do poço, análise gráfica em tempo-real e análise histórica. A Figura 6(b) apresenta a interface de listagem de poços de petróleo.

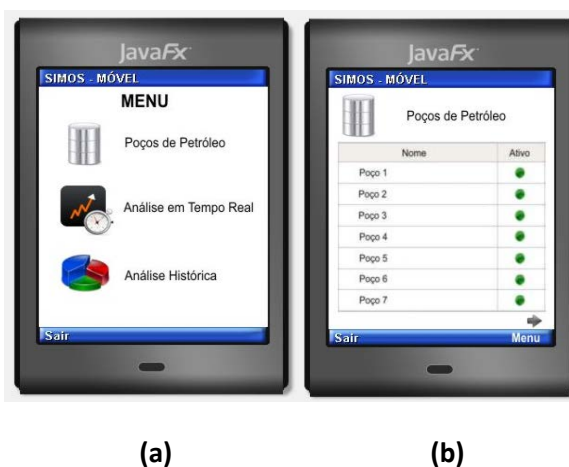


Figura 6 - SiMoS: (a) Menu Principal; (b) interface de Listagem de poços.



(a)

(b)

Figura 7 – SiMoS: (a) interface de Análise Gráfica em tempo-real; (b) interface de Análise Histórica.

A interface de análise gráfica em tempo-real do **SiMoS** consiste em mostrar em tempo-real as informações dos sensores. Deste modo, a cada período de 1 segundo é feita uma consulta no **SiMoS-Servidor** capturando a última informação gravada, e plotando-a no gráfico. A Figura 7(a) apresenta os dados do sensor Pressão da Bomba em tempo-real.

A interface de análise histórica permite aos usuários fazer consultas sobre os dados dos poços armazenados no servidor. A Figura 7(b) apresenta o gráfico resultante dos dados do poço 2 entre 03/06/2010 – 21:40 e 03/06/2010 – 21:50.

• IMPLEMENTAÇÃO

Além dos padrões de projetos *MVC* e *DAO* adotados, o **SiMoS** foi implementado utilizando diferentes tecnologias de apoio priorizando os softwares livres e de código aberto. Tal metodologia torna este trabalho relevante, já que diversos sistemas para plataforma petrolífera apresentam alto custo de aquisição e implantação [4, 5, 8], além disso, não se beneficiam das vantagens das tecnologias disponibilizadas no ambiente Web, como transmissão de dados pela Internet e disponibilização dos dados em páginas dinâmicas.

As principais tecnologias utilizadas pelo **SiMoS** foram: (i) *PostgreSQL*: Sistema Gerenciador de Banco de Dados, o qual permite o armazenamento e a manipulação dos dados de forma descritiva; (ii) *Java Enterprise Edition* (JEE), plataforma Java para aplicações corporativas e internet, permitindo a criação de aplicações mais robustas, multicamadas e capazes de rodarem em servidores de aplicação; (iii) *Java Micro Edition* (JME), plataforma Java para aplicações executadas em dispositivos móveis, possuindo interfaces flexíveis, com suporte a segurança e aos protocolos de rede utilizados na internet; (iv) Glassfish, servidor de aplicações responsável pela execução e gerenciamento do sistema, bem como disponibilização dos serviços ofertados; (v) *Java Server Faces* (JSF), framework para desenvolvimento rápido (RAD) que incorpora o padrão MVC; (vi) *Richfaces*, uma biblioteca de componentes de código aberto que integra recursos

Asynchronous Javascript and XML (AJAX) às aplicações JSF, para a criação de aplicações Web ricas de forma mais ágil; (vii) *Facelets*, framework para construção de *templates* Web; (viii) *JUnit*, para auxílio no testes unitários, visando examinar o menor dos componentes de forma isolada; (ix) *SSL/TLS*, protocolo de comunicação seguro utilizado para criptografar os dados transmitidos entre o servidor e o cliente móvel. Por fim, (x) *JFreeChart*, para manipulação e visualização de dados em forma de gráficos. Esse pacote oferece ferramentas prontas de edição de gráficos, o que agiliza bastante no desenvolvimento do sistema.

Para garantir as restrições temporais impostas com detecção de problemas em tempo-real, o **SiMoS** utiliza a linguagem Java, que disponibiliza a classe *Calendar* do pacote *java.util*, a qual oferece primitivas temporais para garantir restrições de tempo. Através desta classe é possível marcar o instante de tempo para o cálculo do tempo computacional de uma transação. Do mesmo pacote, também é utilizada a classe *Date* para apresentar possíveis status de uma plataforma de poço de petróleo.

AVALIAÇÃO E EXPERIMENTOS

Para análise do sistema **SiMoS**, foram realizados dois testes, um para verificar a viabilidade de sua utilização, através do envio de várias mensagens, e outro para testar a capacidade da aplicação localizada no dispositivo móvel.

• TESTE DE UTILIZAÇÃO DO SISTEMA

Para os testes foi simulado um ambiente com uma única plataforma de poço de petróleo em plena operação, um computador Core 2 Duo, 4gb de memória, Windows 7 32 bits com o **SiMoS-Servidor** instalado, um celular de modelo N95 com o sistema **SiMoS-Cliente** e um roteador de capacidade 54 Mpbs.

Os testes e experimentos que buscam verificar a capacidade do **SiMoS** consiste no envio de várias mensagens para o **SiMoS-Servidor** com o propósito de observar os **tempos médios que a mesma leva para receber uma nova mensagem e processá-la** (tempo de processamento da arquitetura), assim como também o **tempo médio para que uma mensagem seja enviada até o celular** e apresentada ao cliente (tempo de envio).

Para medir o total de transmissão ocorrido durante o envio de uma mensagem para o **SiMoS-Servidor**, foi utilizado o RTT (*Round Trip Time*), que corresponde ao tempo gasto de ida e volta de uma mensagem trafegada pela rede. Sendo assim, foi aplicada a equação (1) para realizar as medições pretendidas.

$$T_G = \frac{RTT}{2} \quad (1)$$

O RTT é calculado da seguinte maneira: O remetente envia uma mensagem para o destinatário e inicia temporizador para o cálculo do tempo. Quando o destinatário recebe a mensagem, automaticamente envia uma mensagem de confirmação, conhecida como *ACK (Acknowledge)* ao remetente. Quando o remetente recebe o *ACK*, desliga o temporizador, obtendo o tempo total de ida e vinda de uma mensagem. Ao dividir o tempo pela metade, obtém o tempo médio só de ida da mensagem, correspondente à variável T_G na equação (1).

Para realizar o cálculo médio dos tempos de processamento da arquitetura, o *timestamp* é armazenada em milissegundos (ms) no banco de dados nas seguintes situações: Recebimento da mensagem; Mensagem preparada e enviada ao celular; Recebimento do *ACK*.

Após isso, é possível realizar o cálculo médio entre cada operação realizada, subtraindo o tempo entre elas. Para o teste, foram enviadas 1000 mensagens para que se possa ter uma média mais precisa dos tempos. O envio de cada mensagem é feito após a confirmação da mensagem de recebimento da anterior. A Figura 8 mostra o gráfico dos tempos de transmissão de cada mensagem enviada ao *SiMoS-Servidor* somado ao tempo de processamento da mesma no servidor. A Figura 9 mostra o gráfico do tempo de envio de mensagens para o celular, em que está instalado o *SiMoS-Cliente*.

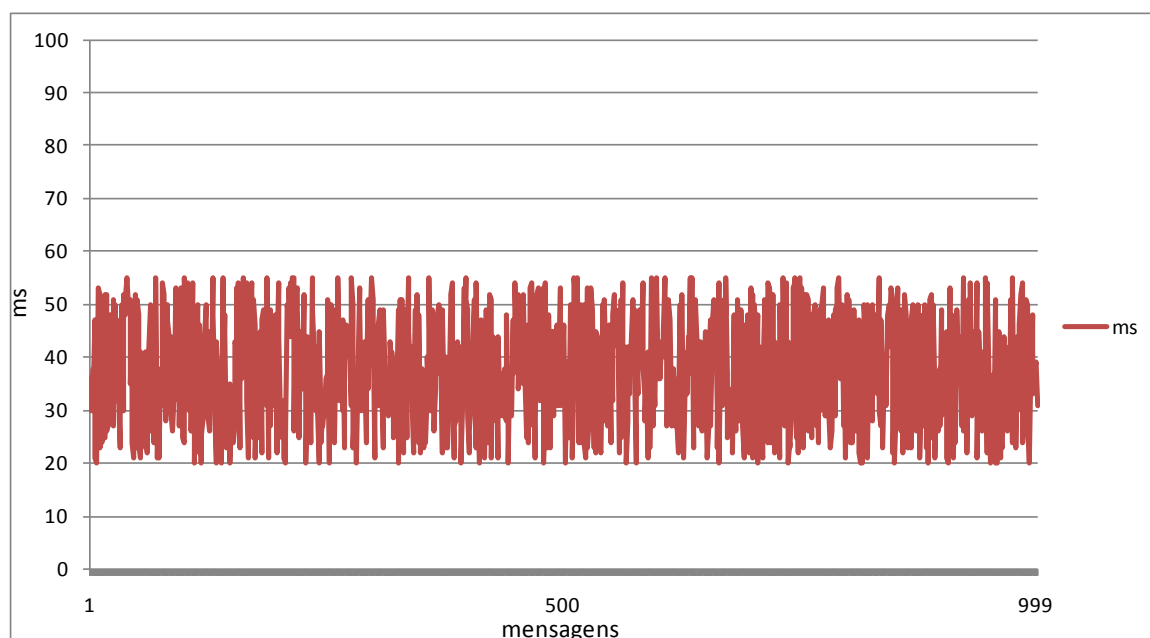


Figura 8 – Gráfico de envio e processamento de mensagens no SiMoS-Servidor.

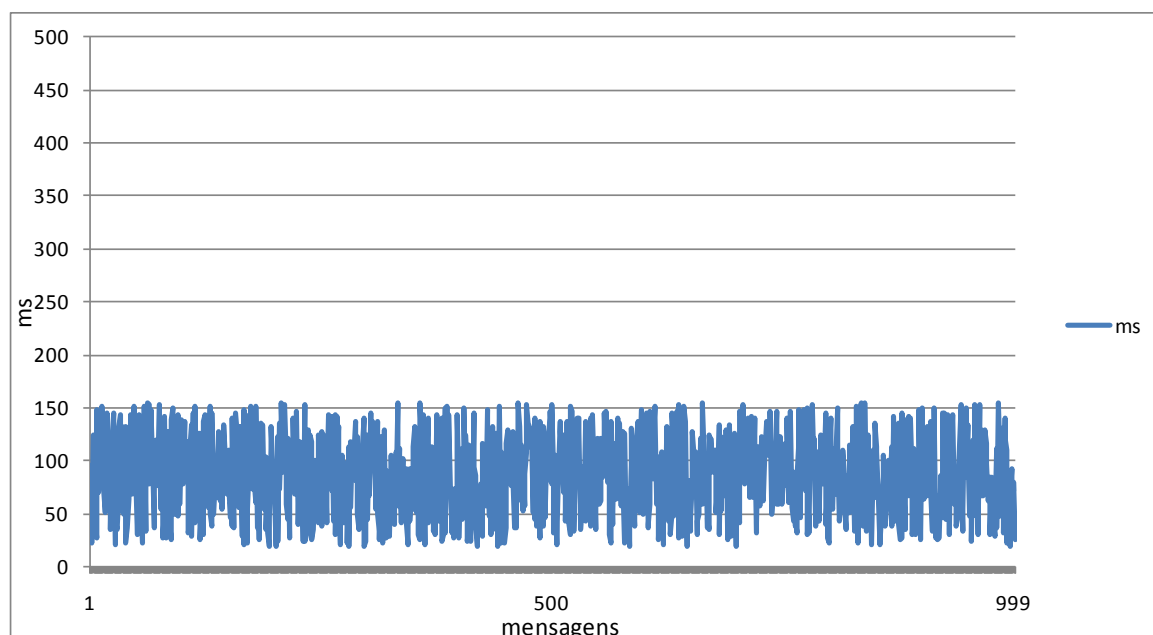


Figura 9 – Gráfico de envio e processamento de mensagens no SiMoS-Cliente.

De acordo com os resultados, o tempo de transmissão das mensagens e de processamento da arquitetura se deu na ordem de milissegundos. Devido a isso, a aplicação atende às necessidades exigidas pelo processo de monitoramento e envio de mensagens, já que apresenta um tempo curto, que possibilita uma rápida tomada de decisão em caso de urgências/problemas nas plataformas de poços de petróleo.

• TESTE DE STRESS

O teste de *stress* consiste na realização de testes de performance que ultrapassam os limites de operação normal de um software. Serve para assegurar que o sistema conseguirá responder a um certo número de usuários acessando seus serviços. Serve principalmente para testar a robustez, disponibilidade e tratamento de erros sob uma carga pesada de requisições ao sistema [21].

Foi utilizado a ferramenta *JMeter*, desenvolvido em Java que permite a realização de testes funcionais de performance em diferentes tipos de sistemas e servidores, como, por exemplo, Web, SOAP, Banco de Dados via JDBC (Java *Database Connectivity*) entre outros [21].

O teste foi realizado para testar o tempo total de comunicação entre o *SiMoS-Servidor* e o *SiMoS-Cliente*, trocando entre eles exatamente as mesmas mensagens utilizadas pela arquitetura. A mensagem enviada foi:

- “100 || 2000 || 392 || 2123”

Ao receber a mensagem, o celular automaticamente enviava uma mensagem ACK de volta ao computador. Foram enviadas 1000 mensagens em um intervalo de tempo de 500 segundos,

resultando no envio de 2 mensagens a cada segundo, independente da confirmação de resposta da anterior. O gráfico que mostra o tempo médio de cada comunicação dos testes está representado na Figura 10.

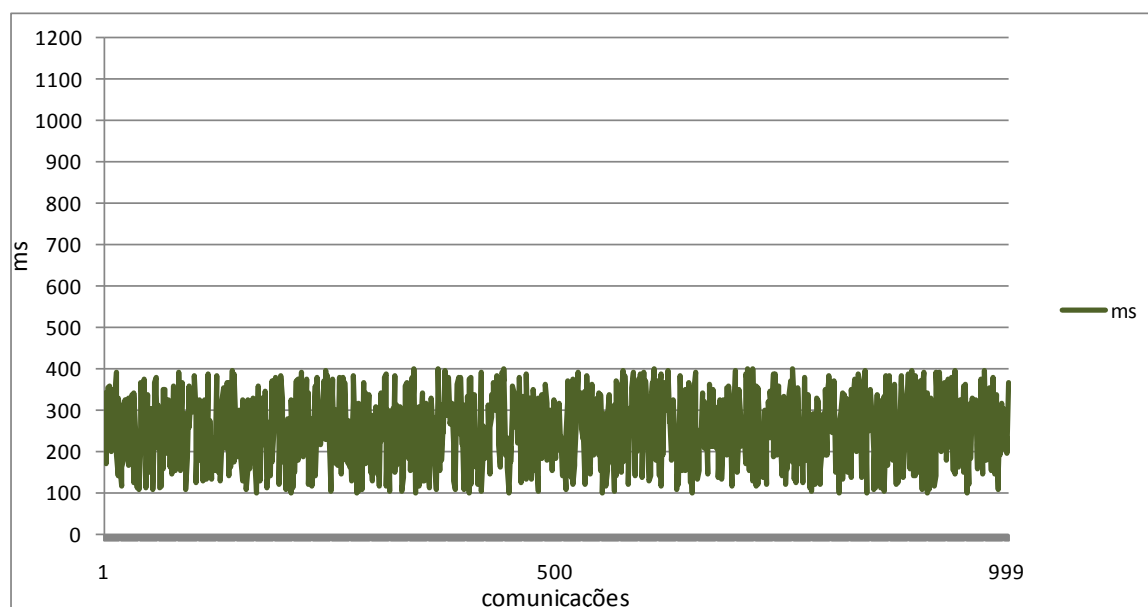


Figura 10 – Gráfico com o resultado do teste de comunicação.

O teste apresentou bons resultados de desempenho, já que a aplicação necessita de operações de comunicação em um curto espaço de tempo, e os resultados apresentaram valores em ordem de milissegundos, uma média de 248 milissegundos de comunicação, e um desvio médio de 39 milissegundos entre as comunicações.

CONCLUSÕES E TRABALHOS FUTUROS

Neste artigo apresentamos o sistema **SiMoS**, um sistema de tempo-real para o monitoramento remoto de plataformas de poços de petróleo através de dispositivos móveis. Baseado nos princípios de sistemas de tempo-real, o sistema **SiMoS** busca garantir uma maior precisão para detecção de problemas devido à menor perda de dados. Além disso, apresenta um baixo custo de aquisição e implantação, visto que todo o sistema prioriza os softwares livres e de código aberto: *PostgreSQL*, *JME*, *JEE* e *Glassfish*. A plataforma Java foi escolhida por ser uma linguagem multiplataforma que disponibiliza mecanismos para tratar as restrições temporais das aplicações. Por fim, os experimentos realizados mostraram que o **SiMoS** é um sistema eficaz para detecção de problemas de plataformas de poços de petróleo.

Durante o desenvolvimento deste trabalho foram identificados alguns aspectos para trabalhos futuros, entre eles estão: (i) realização de experimentos reais de campo em diversas plataformas petrolíferas; (ii) implementação de técnicas de *data mining* sobre os dados para alcançar novas concepções e (iii) implementação da aplicação **SiMoS – Cliente** para a plataforma Java KVM (*Kilo*

Virtual Machine), a qual é uma implementação da máquina virtual Java otimizada para o uso em dispositivos de recursos limitados.

REFERÊNCIAS BIBLIOGRÁFICAS

1. FERNANDES, Y.Y.M.P. Técnica de Controle de Concorrência Semântico para Sistemas de Gerenciamento de Banco de Dados em Tempo-Real. Campina Grande, 2005. Dissertação de mestrado-Universidade Federal de Campina Grande, 2005.
2. LEITE, C.R.M. Linguagem de Consulta para Aplicações em Tempo-Real. Campina Grande, 2005. Dissertação de mestrado-Universidade Federal de Campina Grande, 2005.
3. RIBEIRO NETO, P.F. Mecanismos de Qualidade de Serviços para o Gerenciamento de Dados e Transações em Tempo-Real. Campina Grande, 2006. Tese de doutorado-Universidade Federal de Campina Grande, 2006.
4. DONG, J., HAI-PING, W. Disaster Recovery Project for Short Message of Personal Access Phone System. Journal of Liaoning University of Petroleum & Chemical Technology, 2006.
5. WEI-MIN, W., LIU. Design and Implementation of Short Message Service Platform System. Journal of Liaoning University of Petroleum & Chemical Technology, 2006.
6. COSTA, C.M., LEITE, C.R.M., RIBEIRO NETO, P.F. AMSO: Automated System for Monitoring in Real-time Oil Wells On-shores using Sensors Network. International Congress on Engineering and Computer Education - ICECE, Buenos Aires, Argentina, 2009.
7. COSTA, C.M., LEITE, C.R.M., RIBEIRO NETO, P.F. Desenvolvimento de um componente de software - SAMS: Sistema Automático de Monitoramento em Tempo-Real de Sondas de Produção de Petróleo. VIII Conferência Internacional de Aplicações Industriais - INDUSCON, Poços de Caldas, Minas Gerais, 2008.
8. SALIM, D., BATISTA, T.V., MEDEIROS, A.A.D. A CORBA-Based System for the Monitoring of Petroleum Fields. OTM Workshops, p.34-35, 2004.
9. SOUZA, R. B., MEDEIROS, A. A. D., NASCIMENTO, J. M. A., MAITELLI, A. L., GOMES, H. P. SISAL – um Sistema Supervisório para Elevação Artificial de Petróleo. Rio Oil & Gas Expo and Conference, 2006.
10. Geograf Eletrônico 3.1. Disponível em: <http://www.editelautomacao.com.br/>. Acesso em: 20 de novembro 2010.
11. RIBEIRO NETO, P.F ET AL. Uma aplicação de bancos de dados em tempo-real para redes de sensores. VI Workshop de Tempo Real (WTR) - SBRC, p. 45-52, 2004.
12. SILVA, M.G.M., CONSOLO, A.T. Uso de dispositivos móveis na educação - o SMS como auxiliar na mediação pedagógica de cursos a distância. Disponível em: www.5e.com.br/infodesign/146/Dispositivos_moveis.pdf. Acesso em: 23 de novembro de 2010.
13. JOHNSON, T.M. Java para dispositivos móveis - Desenvolvendo aplicações com j2me. 1ª Edição, Novatec, ISBN 978-85-7522-143-3, 2007, p. 336, 2007.
14. COSTA, C.M. Desenvolvimento de um Sistema de Monitoramento para Automação de Sondas de Produção de Petróleo (SAMS). Mossoró, 2007. Monografia-Universidade do Estado do Rio Grande do Norte, 2007.

-
15. SOMMERVILLE, I. Engenharia de Software. 8ª Edição. Pearson Addison-Wesley, ISBN 978-85-88639-28-7, 2007.
 16. BESTRAVOS, A., LIN, K.-J., SON, S.H. Real-Time Database Systems: Issues and Applications. Kluwer Academic Publishers, 1997.
 17. SHAW, A.C. Sistemas e software de tempo real. trad. Porto Alegre, Bookman, 2003.
 18. ZANIOLO, C., CERI, S., FALOUTSOS, C., SNODGRASS, R.T., SUBRAHMANIAN, V.S., ZICARI, R. Advanced Database Systems. Morgan Kaufmann Publishers Inc, 1997.
 19. AUGUSTO, J.A.S. Introdução Sensores Básicos e Princípios de Funcionamento. Faculdade de Lisboa - Portugal, 2006.
 20. MARTINS FILHO, L.C. CIC 180 – Controle de Processos por Computador. Universidade Federal de Ouro Preto - Ouro Preto, 2003.
 21. JMETER. The Apache Jakarta Project – Jmeter – Disponível em: <http://jakarta.apache.org/jmeter>. Acesso em: 20 de outubro de 2010.