

Utilização de dispositivos móveis para o monitoramento e controle de transdutores inteligentes aplicados na automação residencial

Rogéria Oliani¹, Alexandre César R. da Silva¹, Tércio Alberto dos Santos Filho²

¹Departamento de Engenharia Elétrica – Universidade Estadual Paulista (UNESP)
Ilha Solteira – SP – Brasil

²Departamento de Ciência da Computação - Universidade Federal de Goiás (UFG)
Catalão – GO - Brasil

rooliani@gmail.com, acrsilva@dee.feis.unesp.br, terciotas@gmail.com

Abstract. *Devices such as smartphones, tablets and PCs are already part of the reality of millions of people. The use of many devices with wireless network access enables them to communicate between themselves, thus increasing the comfort and mobility of those who use them. In this paper presents a system capable of controlling and monitoring transducers, such as: light, movement sensor, electric gate motor and others devices connected in wall socket, using tablets/smartphones from wherever the user is having access to Internet.*

Resumo. *Aparelhos como smartphones, tablets e microcomputadores já fazem parte da realidade de milhões de pessoas. O uso de diversos aparelhos com acesso à rede sem fio possibilita que estes se comuniquem entre si, aumentando assim, o conforto e a mobilidade de quem os utilizam. Neste trabalho apresenta-se um sistema capaz de controlar e monitorar transdutores, como: lâmpada, sensor de movimento, motor de portão eletrônico, bem como aqueles conectados às tomadas, utilizando tablets/smartphones, de qualquer lugar em que o usuário esteja com acesso à Internet.*

1. Introdução

A tecnologia faz parte cada vez mais do dia a dia das pessoas, proporcionando-lhes conforto, segurança e mobilidade. Atualmente os dispositivos móveis (celular, PDA's, tablets, iPhones) não são mais utilizados apenas para efetuar ligações, mas também para realizar diversas tarefas, tais como: controlar a agenda de compromissos, acessar à Internet, ler e-mail, efetuar filmagens.

Semelhantes ao modo de funcionamento dos microcomputadores, os *smartphones* e *tablets* possuem sistemas operacionais, no qual realizam o interfaceamento homem máquina, facilitando ao usuário a manipulação de seus dados. Um exemplo de sistema operacional é o Android [Mass 2012].

O Android é uma plataforma *open-source* criada pela Google para o desenvolvimento de aplicações para dispositivos móveis. Inclui um sistema operacional baseado em Linux e diversas aplicações, interface gráfica, navegador web, suporte a

multimídia, GPS (*Global Positioning System*), banco de dados integrado, jogos em 3D e muito mais [Lecheta 2010].

Baseado nas características dos dispositivos móveis, este trabalho foi desenvolvido com o propósito de ampliar o conforto, a segurança e a mobilidade dentro e fora das residências, proporcionando aos seus usuários controlar a iluminação, as tomadas, os sensores de movimento e os portões automáticos de garagem de suas residências de qualquer lugar em que estejam com acesso à *Internet*, fazendo uso de *smartphones* ou *tablets* com SO (Sistema Operacional) Android. Para tanto, foi desenvolvido o software denominado SAR (Software de Automação Residencial), o qual possui dois módulos SARC (Software de Automação Residencial Cliente), desenvolvido para *tables/smarthphones*, e SARS (Software de Automação Residencial Servidor) que foi instalado em uma placa Arduino Mega 2560. Ao término do sistema, foram realizados testes de comunicação entre o módulo do cliente e o servidor.

Na Seção 2 apresenta-se uma abordagem sobre Casas Inteligentes demonstrando algumas de suas características.

2. Casas Inteligentes

Na área de automação residencial vários trabalhos vêm sendo desenvolvidos envolvendo a rede elétrica e hidráulica de casas, equipamentos eletrônicos, bem como o mobiliário destas.

Segundo [Alves & Mota 2003], casas inteligentes são aquelas que possuem características capazes de tornar a vida, de quem nelas habita, mais simples; envolvendo aspectos, como: segurança, economia, conforto, ecologia e integração. A importância desses aspectos tem um valor distinto para cada indivíduo, em cada momento.

A gestão dos espaços internos e externos com o controle integrado de iluminação, irrigação, som ambiente, cortinas e de diversos outros equipamentos, seja local ou remoto, com o uso da *Internet*, tem conquistado cada vez mais clientes no mercado imobiliário e atendem pessoas da classe A, B e C [Loyola 2011].

Além de comodidade, a tecnologia empregada nas Casas Inteligentes pode auxiliar para a segurança de crianças e idosos, bem como ajudar pessoas com deficiência a ter maior independência [Daquino 2012].

Outras funcionalidades que podem ser aplicadas às casas inteligentes é o controle do consumo energético, possibilitando uma energia mais verde e casas mais eficientes, com o controle exato de como e quando é gasta a energia. No caso de um veículo elétrico, por exemplo, poderia calcular a parcela exata de consumo alocada ao seu abastecimento [Inovcity 2010].

3. Ferramentas utilizadas no desenvolvimento do sistema

Nesta seção apresentam-se os procedimentos utilizados para o desenvolvimento do sistema e as ferramentas utilizadas. É importante destacar que o sistema foi dividido em duas fases de desenvolvimento, sendo: programação dos dispositivos embarcados *tablets/smartphones* e a programação da placa Arduino Mega 2560 e do *Shield¹ Ethernet W5100* para o controle efetivo dos transdutores utilizando à *Internet*.

¹ *Shield* – placas de circuito contendo outros dispositivos.

Na Seção 3.1, apresentam-se as características dos *hardwares* utilizados para o desenvolvimento do módulo servidor e na Seção 3.2 apresentam-se as linguagens de programação utilizadas no desenvolvimento do software para os dispositivos embarcados, bem como para o desenvolvido para a placa Arduino.

3.1. Hardwares

Para o desenvolvimento do módulo de servidor foram utilizadas três placas: Arduino Mega 2560, *Shield Ethernet* W5100 e um *Shield* de Automação e Alarme Residencial V2.

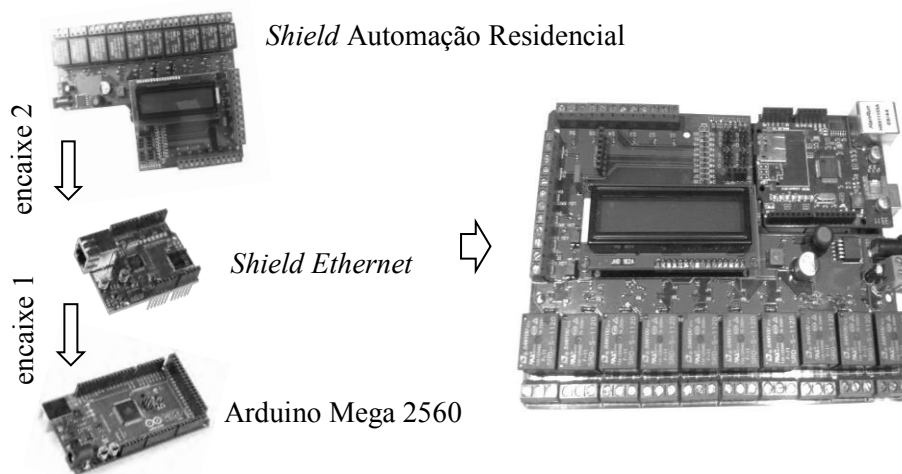
A placa Arduino Mega 2560, a qual possui um microcontrolador baseado no ATmega2560, possui 54 pinos de entradas/saídas digitais, 16 entradas analógicas, 4 UARTs (portas seriais de hardware), um oscilador de cristal de 16 MHz, uma conexão USB, uma entrada de alimentação, uma conexão ICSP (*In-Circuit Serial Programming*), um botão de reset e 256 KB de memória flash.

O *Shield Ethernet* W5100 possibilita à placa Arduino se comunicar com outros *hosts* através da *Internet*, ou até ter um servidor web embarcado, o qual viabiliza aos usuários acessarem páginas dentro do Arduino. Este *shield* fornece acesso à rede IP (*Internet Protocol*) através dos protocolos TCP (*Transmission Control Protocol*) ou UDP (*User Datagram Protocol*), bem como possui um *slot* para cartão SD que pode ser usado para armazenar arquivos [Quines 2008].

A placa de automação residencial é um circuito de condicionamento de sinal que recebe sinais elétricos de comando da placa Arduino e realiza o controle ou o monitoramento do ambiente em que esteja implementado.

Na Figura 1 apresenta-se o encaixe das placas para automação residencial; sendo, primeiramente, conectado o *Shield Ethernet* à placa Arduino Mega 2560 e, posteriormente, estas ao *Shield* de Automação e Alarme Residencial V2.

Figura 1 - Encaixe das placas para automação residencial.



Fonte: Elaborada pelo autor.

Para o desenvolvimento deste trabalho, bem como para efetuar os devidos testes, além da placa Arduino Mega 2560, *Shield Ethernet* W5100 e *Shield* de Automação e Alarme Residencial V2 foram utilizados os seguintes equipamentos: um roteador; um modem; duas lâmpadas; uma tomada; um sensor de movimento; um controle de portão

eletrônico; um *tablet* com Sistema Operacional Android; um *smartphone* com Sistema Operacional Android.

3.2. Linguagens de programação

Foram utilizadas no desenvolvimento deste trabalho as linguagens de programação Arduino e Java. A linguagem Arduino foi utilizada na programação da placa Arduino e a linguagem Java, em conjunto com o Android SDK (*Software Development Kit*), no desenvolvimento do sistema manipulado pelo usuário no *tablet/smartphone*.

A linguagem Arduino é baseada em linguagem C, e utiliza um software livre, Arduino v 1.0.1, para a escrita dos códigos do sistema criado para a placa Arduino, bem como para efetuar o *upload* na placa deste. A linguagem C é uma linguagem genérica que, apesar de ser de alto nível, compartilha recursos tanto de alto nível, quanto de baixo, pois permite acesso e programação direta de microprocessadores [Dornelles 1997].

O Android SDK possui uma biblioteca API e uma ferramenta para desenvolver, testar e debugar aplicativos para Androids utilizando a linguagem Java. Uma das grandes vantagens é a diversidade de aparelhos encontrados no mercado utilizando o sistema operacional Android. Atualmente são várias as empresas que desenvolvem referidos aparelhos, como, por exemplo: Motorola, LG, Samsung, HTC e outros. Outra característica relevante é que o mesmo possui o código aberto, e permite uma programação personalizada, sendo possível customizar os aparelhos baseados em sua característica.

4. Desenvolvimento do Sistema de Automação Residencial (SAR)

O Sistema de Automação Residencial desenvolvido foi denominado de SAR foi dividido em dois módulos:

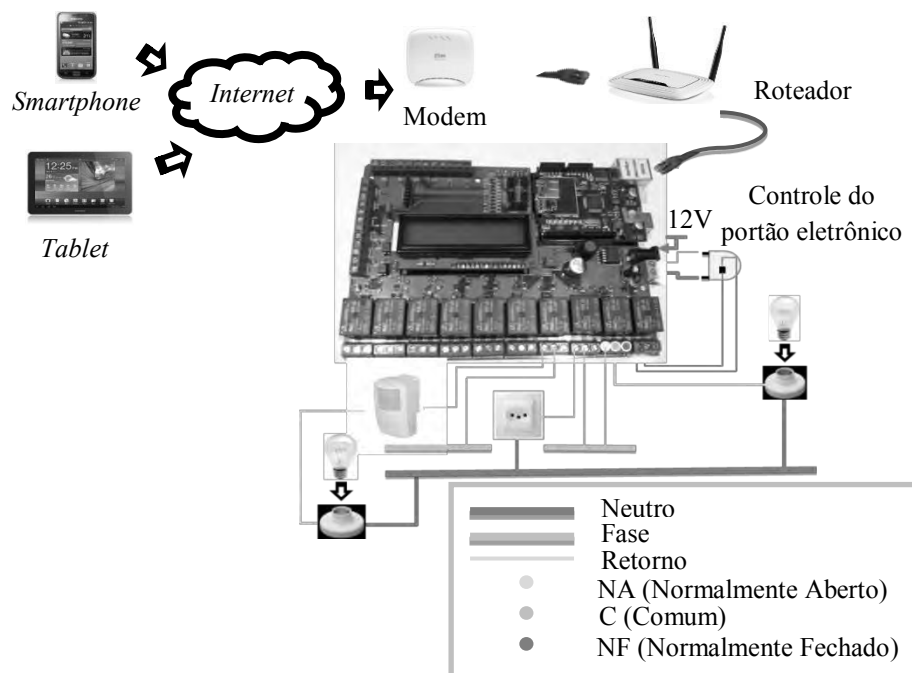
1. SARS (Software de Automação Residencial Servidor) - desenvolvido para a placa Arduino;
2. SARC (Software de Automação Residencial Cliente) - desenvolvido para *tablet/smartphone* com SO Android.

Conforme apresentado na Figura 2, o SARC enviará a requisição solicitada pelo usuário através da *Internet*. Por intermédio do modem, do roteador e do *Shield Ethernet*, esta requisição chegará à placa Arduino na qual o SARS a interpretará e ativará/desativará o relé localizado no *Shield* de Automação Residencial, conforme solicitado.

Na placa de Automação Residencial estão sendo utilizados quatro relés. Na Figura 2, visualizando da direita para a esquerda, no primeiro relé está conectado o controle de portão eletrônico, no segundo a lâmpada, no terceiro a tomada e no quarto o sensor de movimento.

Os relés funcionam como interruptores que permitem ou não que a tensão da rede elétrica da residência chegue até os transdutores, permitindo que se acenda/apague a lâmpada, ligue/desligue o sensor de movimento, a tomada e o controle do portão eletrônico.

Figura 2 - Esquemático do sistema implementado.



Fonte: Elaborada pelo autor.

Para a requisição realizada pelo usuário através do *tablet/smartphone* trafegue pela *Internet* e chegue ao seu destino, é necessário que este seja identificado através de seu IP. O IP identifica um dispositivo tanto na rede local, quanto na rede mundial de computadores. O *Shield Ethernet* utilizado neste trabalho é identificado na rede local por um IP, o qual foi determinado no software e programado na placa Arduino. Assim, quando receber um dado da rede externa o roteador consegue localizá-lo na rede local, realizando um NAT (*Name Address Translate*), pois, a *Internet* e a rede local são redes distintas e possuem IPs dedicado a sua rede.

Neste trabalho os testes foram realizados através de uma conexão utilizando IP dinâmico. Para tanto, foi necessário que o destino das requisições realizadas através do *tablet/smartphone*, por IP, fosse realizada através do uso de DNS (*Domain Name System*).

Os Serviços Dinâmicos DNS - como o do No-IP, o qual foi utilizado neste trabalho, possibilitam que um dispositivo seja localizado na *Internet*, mesmo com a mudança de seu IP (IP Dinâmico), pois estes sistemas, através de seus aplicativos, informam de tempos em tempos o IP atual, efetuando automaticamente a atualização do redirecionamento.

4.1. SARS - Software Desenvolvido para a placa Arduino (servidor)

Como sua própria denominação específica, o SARS atua como um Servidor Web recebendo as requisições que são enviadas pelo cliente (SARC), executando processos, conforme as requisições recebidas, e enviando respostas aos clientes, com ou sem dados.

O trabalho foi desenvolvido para efetuar o controle de quatro relés, os quais serão ativados ou desativados conforme as requisições enviadas pelo software cliente,

bem como informar o status (ativado/desativado) de três destes relés (relés que controlam o sensor, lâmpada e tomada). O relé que ativa o controle remoto do portão eletrônico não fornece seu status, em virtude do mesmo ser desativado 200 ms (milissegundos) após a sua ativação.

Por questão de segurança, as requisições feitas pelos clientes são autenticadas (usuário e senha) e enviadas utilizando-se o método POST através do uso do protocolo de transferência HTTP (*HyperText Transfer Protocol*).

4.2. SARC - Software desenvolvido para *Tablet/Smartphone* (cliente)

O SARC foi desenvolvido para o Sistema Operacional Android (*Tablet/Smartphone*), o qual é utilizado pelo usuário para efetuar o controle dos transdutores conectados a placa de Automação Residencial, e atua como um cliente, em uma relação cliente/servidor, com o software desenvolvido para a placa Arduino (SARS).

Na Figura 3 apresenta-se a *Interface* do SARC, na qual é possível identificar sete botões: Tomada – ativa ou desativa a tomada; Lâmpada – acende ou apaga a lâmpada; Sensor – liga ou desliga o sensor; Garagem – aciona o controle do portão eletrônico da garagem por 20 ms; *Internet* – ativa ou desativa o uso da *Internet* pelo sistema; Atualizar – habilita ou desabilita a atualização do status do SARS a cada 1 segundo; Sair – encerra a aplicação.

Figura 3 - Interface do SARC.



Fonte: Elaborada pelo autor.

Ao lado dos botões “*Internet*” e “*Atualizar*” são exibidas as seguintes informações:

- Tempo de Resposta – tempo decorrido entre o envio da requisição e o retorno de sua resposta, em segundos;
- Tempo Médio – tempo médio de resposta;
- Atualizações – número de atualizações realizadas.

5. Resultados

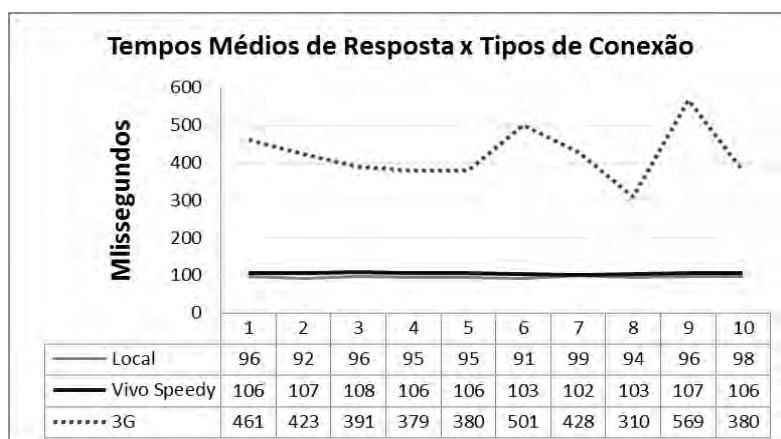
Com o objetivo de verificar a viabilidade do Sistema de Automação Residencial (SAR) no envio e recebimento de dados, foram efetuados testes utilizando rede local, *Internet* Banda Larga Vivo Speedy (plano de 1 Mb/s) e *Internet* Móvel 3G Vivo.

O Tempo Médio de Resposta (TMR) mínimo obtido foi 91 ms (rede local) e o máximo 569 ms (*Internet* Móvel 3G Vivo). Apesar da expressiva diferença entre o mínimo e o máximo alcançado nos diferentes tipos de conexões utilizados, o sistema apresentou um tempo de resposta viável. No tempo de resposta mais lento, o usuário aguardou 0,56 segundo para obter uma resposta, o que representa mais da metade do tempo que o sistema leva para efetuar a atualização automática, que foi determinada para ocorrer a cada segundo.

Para a realização das análises foram realizadas dez medições do Tempo Médio de Resposta, obtido através de cem atualizações cada, totalizando mil atualizações (repostas obtidas), para cada tipo de conexão (Local, *Internet* Vivo Speedy e 3G Vivo). O *Shield Ethernet* conectado a placa Arduino permaneceu em todos os testes comunicando-se com a *Internet* por meio da conexão Vivo Speedy, exceto quando da realização dos testes utilizando-se conexão local. Independente do tipo de conexão, as requisições foram realizadas e tiveram um retorno de cem por cento dos dados.

No Gráfico 1 apresenta-se os resultados dos testes realizados para obtenção do TMR do SAR, no qual se pode observar a oscilação tida com a conexão 3G Vivo, bem como os resultados obtidos com a conexão Local e Speedy, as quais apresentaram pouquíssimas variações, com um TMR mínimo de 91 e 102 ms, e um máximo de 99 e 108 ms, respectivamente.

Gráfico 1 - Resultado dos testes realizados para obtenção do TMR do SAR.



Fonte: Elaborada pelo autor.

6. Conclusão e Trabalhos Futuros

O trabalho desenvolvido resultou em um sistema denominado SAR, no qual foram efetuados testes que demonstraram a sua viabilidade no tocante ao envio e recebimento de dados.

O controle e monitoramento de sensor, lâmpada, tomada e controle de portão eletrônico apresentaram-se eficazes, ao ponto que todas as requisições recebidas pelo SARS foram executadas de maneira satisfatória.

Assim, com base nos testes realizados, o SAR demonstrou ser um sistema viável, e que pode ser expandido com a agregação de outros transdutores e funções, com foco nas pessoas e no meio ambiente, propiciando mais conforto a aquelas, e respeito a este, com a utilização responsável da energia e das águas.

Como possíveis trabalhos futuros pode-se apontar a implementação de um Banco de Dados utilizando uma estrutura de dados JSON (*JavaScript Object Notation*), a ser armazenado em um cartão micro SD (*Shield Ethernet*), contendo os dados dos usuários, tipos de requisições, data/hora dos acessos a fim de se manter um histórico destes, bem como os tipos de usuários, controlando, assim, o nível de acesso ao sistema.

Agradecimentos

Os autores agradecem ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (Processo: 307255/2009-3) e a Universidade de Jales – UNIJALES.

Referências

- Alves, José Augusto, e José Mota. *Casas Inteligentes*. 2003. <http://www.centroatl.pt/titulos/solucoes/casasinteligentes.php3> (acesso em 11 de 11 de 2012).
- Daquino, Fernando. *Casas inteligentes serão realidade em 10 anos, afirmam cientistas*. 31 de março de 2012. <http://www.tecmundo.com.br/casas/21501-casas-inteligentes-serao-realidade-em-10-anos-afirmam-cientistas.htm> (acesso em 11 de novembro de 2012).
- Dornelles, Adalberto A. “Fundamentos de Linguagem C.” *Departamento de Engenharia de Computação e Automação - UFRN*. 1997. http://www.dca.ufrn.br/~xamd/dca0800/apostila_C.pdf (acesso em 09 de novembro de 2012).
- Inovcity. *Casas mais Inteligentes*. 2010. <http://www.inovcity.pt/pt/Pages/casas-inteligentes.aspx> (acesso em 11 de novembro de 2012).
- Lecheta, Ricardo R. *Google Android Aprenda a criar aplicações para dispositivos móveis com o Android SDK*. São Paulo: Novatec, 2010.
- Loyola, Darshany. *'Casas inteligentes' atraem cada vez mais clientes no mercado capixaba*. 16 de setembro de 2011. <http://g1.globo.com/espírito-santo/mercado-imobiliario/noticia/2011/09/casas-inteligentes-atraem-cada-vez-mais-clientes-no-mercado-capixaba.html> (acesso em 11 de novembro de 2012).
- Mass, FRAMINGHAM. *Android Marks Fourth Anniversary Since Launch with 75.0% Market Share in Third Quarter, According to IDC*. 1 de novembro de 2012. <https://www.idc.com/getdoc.jsp?containerId=prUS23771812> (acesso em 2012 de novembro de 15).
- Quines, Rafael. *EthernetShield Oficial do Arduino*. 8 de novembro de 2008. <http://www.arduino.net/blog/tag/ethernet-shield-arduino-internet-servidor-web/> (acesso em 10 de outubro de 2012).