

HEALTH CARE REMOTE SYSTEM – SISTEMA DE MONITORAMENTO REMOTO DE SINAIS MÉDICOS

Lucas Benazzi Cestari ¹; Alessandra Dutra Coelho ²

¹ Aluno de Iniciação Científica da Escola de Engenharia Mauá (EEM/CEUN-IMT);

² Professora da Escola de Engenharia Mauá (EEM/CEUN-IMT).

Resumo. *Foi desenvolvido um sistema que permite o monitoramento remoto de sinais vitais de um usuário de modo que esses dados possam ser analisados por um profissional qualificado da área médica por meio de um servidor externo. Junto a isso, é mantido um histórico do paciente para um melhor acompanhamento das informações obtidas pelo sistema de monitoramento, juntamente com um aplicativo de celular que disponibiliza os dados para o usuário. O sistema em questão foi desenvolvido de tal forma a possibilitar a fácil expansão do número de sensores, e possibilitando a integração com diversos sistemas de hospedagem de servidores para armazenamento dos dados obtidos.*

Introdução

A preocupação com a saúde e a qualidade de vida é um tema relevante para a sociedade e o crescente uso de tecnologias em aplicações voltadas para o monitoramento dos sinais vitais bem como para os cuidados com a saúde tornam-se uma tendência. Com os avanços dessas tecnologias, novos métodos e técnicas de medições de sinais vitais vem sendo desenvolvidos possibilitando que pessoas sem experiência médica aprofundada possam medir corretamente os próprios sinais sem auxílio médico, e no próprio ambiente residencial. Entretanto, devido à ausência médica no acompanhamento desses resultados os benefícios desses aparelhos são limitados uma vez que não substituem por completo um atendimento médico pessoal.

No Instituto Mauá de Tecnologia foi desenvolvido uma “Estação Health Care” como Trabalho de Conclusão de Curso (Castagnoli et al., 2015), com o objetivo de promover medições de sinais vitais em tempo real com acompanhamento médico à distância, a fim de proporcionar uma maior frequência no monitoramento de um paciente e diminuindo a necessidade das visitas a um profissional médico, portanto, contribuindo para a mobilidade urbana. Reduzindo o tempo gasto do paciente com transporte, que contribui para o aumento da sua qualidade de vida, melhora a disponibilidade do profissional para auxiliar outros pacientes, e principalmente aprimora a agilidade na resposta a qualquer possível problema de saúde, uma vez que o monitoramento se torna mais constante.

No presente trabalho se pretende aproveitar os conceitos desenvolvidos no trabalho (Castagnoli et al., 2015) e desenvolver um sistema de monitoramento compacto, que permite obter o registro histórico de informações relacionadas à saúde do indivíduo, transmitindo essas informações remotamente de forma que as variáveis possam ser monitoradas por meio de um aplicativo para *smartphones* e *tablets*. Para isso, foi feito uma pesquisa inicial sobre tecnologias de caráter vestível, em especial de monitoramento de sinais vitais, e da metodologia dos sistemas de monitoramento de sinais vitais, a fim de compreender as limitações físicas que um sistema móvel teria, e possibilitar o desenvolvimento dos diversos componentes necessários.

O desenvolvimento do projeto consistiu nas seguintes partes principais: obtenção dos dados dos sensores de medição, transmissão desses dados sem fio para um aplicativo de celular, desenvolvimento do aplicativo, transmissão desses dados para um servidor externo, configurações necessárias do servidor, e desenvolvimento de um protótipo funcional.

Material e Métodos

Como este trabalho apresenta diversos aspectos similares ao Trabalho de Conclusão de Curso (Castagnoli et al., 2015), foi possível aproveitar alguns conceitos e informações. O principal auxílio foi em relação à obtenção dos dados dos sensores, uma vez que os protocolos das transmissões de dados dos sensores não foram disponibilizados ao público pelos seus desenvolvedores. Essas informações possibilitaram a obtenção dos protocolos ausentes, aumentando a agilidade no desenvolvimento do projeto.

Os componentes utilizados no desenvolvimento do sistema de monitoramento foram:

-*Arduino Mega 2560* – o Arduino Mega 2560 é um microcontrolador baseado no *ATmega2560*. Possui 54 pinos de *input/output*, 16 pinos analógicos de *input*, 4 *UARTs* (Transmissão e Recepção Assíncrona Universal), um oscilador de cristal de 16 MHz, um conector *USB, ICSP header*, e um botão para *reset*. Foi utilizado no projeto principalmente pelo seu número de portas *UARTs*, pois cada sensor e o módulo *Bluetooth* necessitam de uma conexão distinta.

-*CMS50D* – é um oxímetro de dedo portátil com exibição de curva plestimográfica, com possibilidade de uso pessoal ou hospitalar, da marca *Contec*. Foi utilizado no projeto pela sua capacidade de medir indiretamente a saturação de oxigênio no sangue.

-*HEM-7113* – é um medidor de pressão arterial automático de braço portátil, da marca *Omron*. Foi utilizado no projeto como medidor de pressão sistólica e diastólica, além de medição de batimento cardíaco.

-*RS232 HC-05* – é um módulo para comunicação com *Arduino*. Possui protocolo *Bluetooth v2.0+EDR*, alcance de 10 metros, e suporte para modo mestre ou escravo. Foi utilizado no projeto afim de possibilitar a comunicação entre o *Arduino* e o aplicativo celular de forma sem fio.

-*Placa de circuito impresso* – consiste de uma placa isolante de fenolite, fibra de vidro, fibra e filme de poliéster, com face de cobre simples, e foi utilizado no projeto como ponto de conexão entre *Arduino Mega 2560* e as diversas componentes sem necessidade de fixar o *Arduino Mega 2560*.

-*Pilhas* – foram utilizadas 8 pilhas AAA de tipo alcalinas a fim de suprir 6 volts para o medidor de pressão *HEM-7113*, 3 volts para o oxímetro *CMS50D* e 9 volts para o *Arduino Mega 2560*.

Os softwares utilizados durante o desenvolvimento foram:

-*IDE Arduino* – é um aplicativo multiplataforma escrita na linguagem *Java*, foi originado das linguagens *Processing* e *Wiring*. Inclui um editor de código capaz de compilar e fazer o *upload* de programas para uma placa *Arduino*, sendo utilizado para programar o *Arduino Mega 2560* no caso do projeto.

-*Android Studio* – *Android Studio* é o ambiente de programação oficial para a programação do sistema operacional *Android* do *Google*. Possui linguagem *Java* e foi utilizado para desenvolver o aplicativo para celular no projeto.

-*WampServer* – o aplicativo *WampServer* é um instalador dos softwares *PHP 5, MySQL e Apache*, possibilitando o uso de *scripts PHP* de forma local. Foi utilizado para o auxílio da criação de uma *database* e poder testar as configurações de comunicação com um servidor local, sem a necessidade de utilizar um serviço de *host* externo.

-*NI Multisim* – é um software da *National Instruments* designado para o auxílio e simulação de circuitos elétricos. Possui compatibilidade com o software *NI Ultiboard*, e foi utilizado no projeto para desenvolver as conexões da placa de circuito impresso.

-*NI Ultiboard* – é um software de *design* de circuitos impressos da *National Instruments*. Possui compatibilidade com o software *NI Multisim*, e foi utilizado no projeto para desenvolver os arquivos necessários para a impressão da placa de circuito impresso.

Para a obtenção dos dados dos sensores, como previamente mencionado, foram aproveitados os protocolos de transmissão de dados do trabalho (Castagnoli et al., 2015) e feita a leitura através da plataforma *Arduino*, a fim de armazenar os valores obtidos temporariamente para poderem ser transmitidos. Os sensores foram ligados diretamente às entradas seriais do *Arduino* e decodificados através da programação do *IDE Arduino*, com base no protocolo disponibilizado, e as suas ligações podem ser observadas na figura 1.

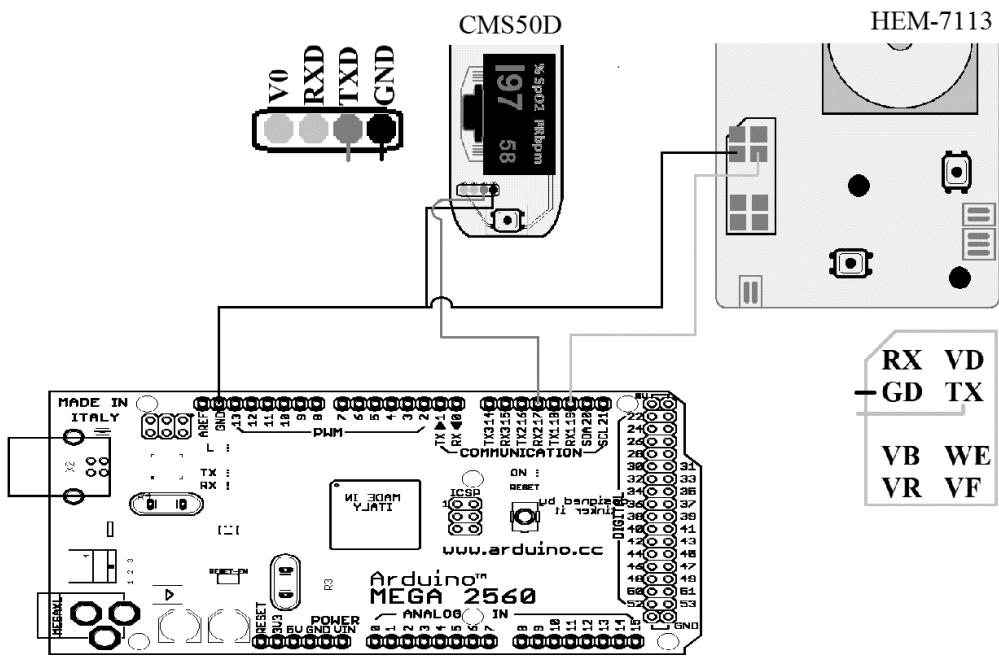


Figura 1 - Esquema elétrico das ligações entre o medidor de pressão HEM-7113 e o oxímetro CMS50D com o Arduino Mega 2560.

As transmissões dos dados armazenados temporariamente no *Arduino* foram transferidas para um aplicativo celular utilizando a tecnologia *Bluetooth*. Entretanto, como a plataforma não possui essa tecnologia integrada em seu sistema, foi necessário utilizar o módulo *RS232 HC-05*, que possui compatibilidade com a plataforma *Arduino* e não necessita de programação específica, sendo apenas necessário definir no programa *IDE Arduino* as portas de conexão que serão utilizadas. Para facilitar a transferência e a leitura dos dados no aplicativo celular, os dados foram agrupados em uma variável e transferidos de forma única. O módulo *RS232 HC-05* possui tensão máxima de 3V nas suas entradas *RXD* e *TXD*, entretanto o sistema *Arduino Mega 2560* trabalha com 5V para níveis lógicos altos; dessa forma é necessário um divisor de tensão antes de ligar ambos diretamente, a fim de reduzir possíveis danos ao módulo. A ligação utilizada no projeto segue o esquema da figura 2.

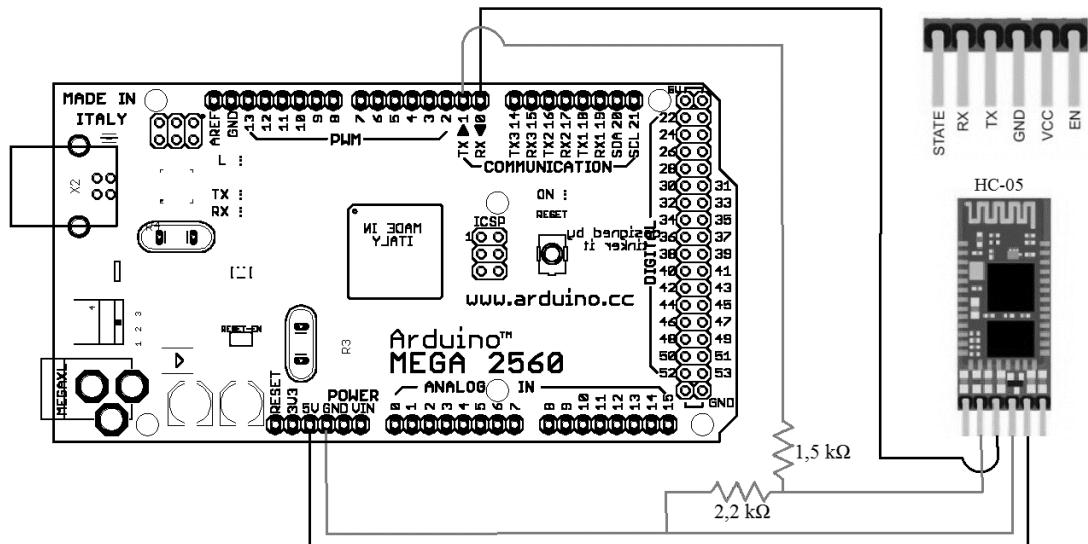


Figura 2 - Esquema para a conexão do módulo *RS232 HC-05* ao *Arduino Mega 2560*.

A transferência de dados sem fio final se dá entre o aplicativo no celular e o servidor. Tendo em vista que todos os *smartphones* possuem alguma forma de comunicação com a Internet, não é necessário conectar algum módulo externo no aplicativo móvel. Desse modo é necessário apenas as configurações do aplicativo celular e do servidor para que ambos possam se comunicar e transferir dados.

O aplicativo de celular em questão foi feito com o *software Android Studio*, e possui as seguintes capacidades: conexão com um dispositivo por *Bluetooth*, decodifica os dados provenientes do *Arduino Mega 2560*, exibe esses dados corretamente e os transfere para um servidor. Sua interface é apresentada na figura 3. A conexão com o dispositivo RS232 *HC-05* é feita de forma padrão. O aplicativo ativa o modo *Bluetooth* e faz com que o celular entre no modo de *descoberta*. É selecionado o dispositivo desejado a fim de realizar o pareamento inicial e tornar possível a conexão. Com a conexão estabelecida, se inicia os processos de leitura e envio de dados de ambos os dispositivos.

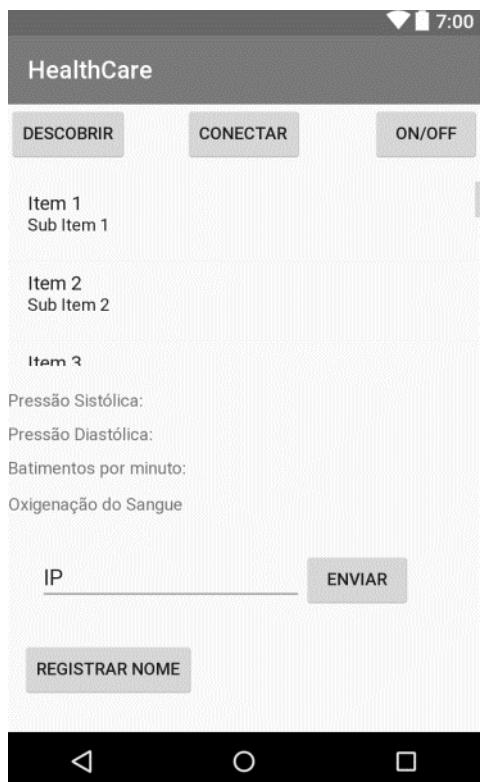


Figura 3 - Interface do aplicativo celular criado com Android Studio.

Para a realização da leitura dos dados, o aplicativo deve receber os dados de forma única, a fim de possibilitar a decodificação dos dados. Quando o *Arduino Mega 2560* recebe os dados provenientes dos sensores, junto a uma conexão bem-sucedida com o aparelho celular, se inicia o processo de envio de dados, através do módulo RS232 *HC-05*. Os dados recebidos pelo celular são separados e armazenados através de um algoritmo, possibilitando a exibição e preparo para a transferência ao servidor.

As linguagens *Java*, utilizada na programação *Android*, e o sistema *MySQL* de banco de dados estão extensivamente implementados em diversos aparelhos e serviços, de forma que são compatíveis com uma grande variedade de sistemas e programas. Entretanto, não é possível realizar a comunicação direta entre um aparelho celular *Android* com uma *database MySQL*, sendo necessário um sistema intermediário. A linguagem em *PHP (Hypertext Preprocessor)* auxilia nessa comunicação. Dessa forma, o aplicativo faz um requerimento para a *database MySQL* através de um arquivo escrito em linguagem *PHP*. Para tanto, a programação dos requerimentos do aplicativo celular foi feita em duas partes, o desenvolvimento do código para o próprio celular em linguagem *Java*, e o seu complemento de forma externa em um arquivo escrito com linguagem *PHP*. Finalizado

isso foi possível desenvolver todos os comandos necessários para a transmissão dos dados para o servidor.

O servidor foi desenvolvido de forma local com auxílio do aplicativo *WampServer*, facilitando as interações entre o sistema de *database MySQL*, os comandos escritos com linguagem *PHP*, e o sistema de *host* com o aplicativo *Apache*. Os dados principais que foram armazenados dentro do banco de dados foram: a identidade (*id*) de cada armazenamento, o nome do usuário, as informações provenientes dos sensores, e a data e a hora na qual os dados foram transferidos para o servidor. No projeto atual, os dados dos sensores transmitidos foram: batimentos por minuto, pressão sistólica, pressão diastólica e oxigenação do sangue. Uma imagem dos dados armazenados no servidor pode ser observada na figura 4.

+ Options										
	← →	▼	Id	Nome	Pressao Sistolica	Pressao Diastolica	Bpm	Oxigenacao	Tempo	
<input type="checkbox"/>				1	Pessoa 1	135	76	68	97	2017-11-06 08:33:20
<input type="checkbox"/>				2	Pessoa 1	132	78	75	97	2017-11-07 08:51:28
<input type="checkbox"/>				3	Pessoa 2	142	86	82	96	2017-11-07 15:48:14
<input type="checkbox"/>				4	Pessoa 1	130	72	72	98	2017-11-08 08:16:51
<input type="checkbox"/>				5	Pessoa 2	145	90	85	96	2017-11-08 15:31:10

Figura 4 - Imagem dos dados armazenados no servidor local.

Uma vez armazenados os dados, é possível organizar os dados do histórico do paciente de forma gráfica, a fim de auxiliar o seu acompanhamento médico. Uma ilustração desses dados em forma gráfica pode ser observada na figura 5. Devido a ampla aplicação do sistema *MySQL*, é possível exportar os dados do servidor para programas externos, como por exemplo *Microsoft Excel*, caso seja necessária alguma análise mais específica.

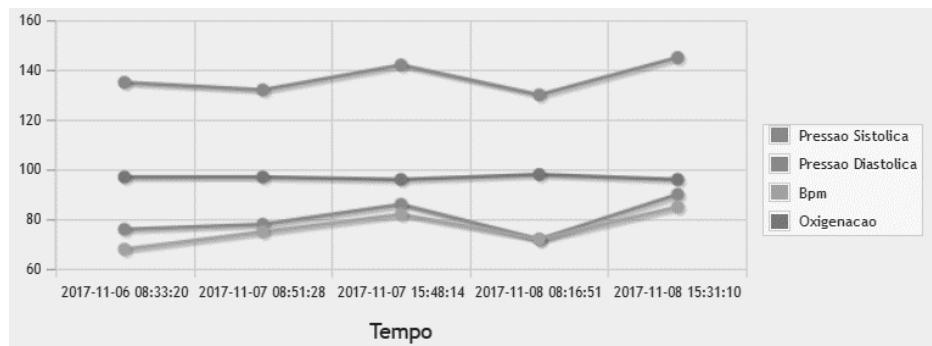


Figura 5 - Gráfico exemplar dos dados de um paciente.

Após o correto funcionamento de todas as partes do projeto, a obtenção e transmissão dos dados para um servidor, se iniciou o processo da criação de um protótipo, com função principal de unir todas os elementos, junto com a capacidade de se posicionar e retirar os componentes com facilidade à medida que forem necessários. O protótipo inicial serviu para armazenar todos os elementos, e ainda mantendo o acesso aos botões e o *display* do medidor de pressão *HEM-7113*. Se optou pela construção utilizando principalmente madeira por possuir relativa leveza e força, suficientes para o projeto em questão, junto a maior facilidade de moldagem.

Com o planejamento de uma configuração geográfica que adequasse todos os elementos finalizado, se iniciou o processo de desenvolvimento das diversas peças responsáveis por assegurar de forma fixa todos os elementos. Com intuito de possibilitar o encaixe do *Arduino Mega 2560*, e facilitar as conexões com o mesmo, foi desenvolvido uma placa de circuito impresso, com conexões para ambos os sensores de oxigenação e pressão sanguínea, e o módulo *Bluetooth HC-05* juntamente com o divisor de tensão. Uma ilustração do esquema da placa desenvolvida segue na figura 6.

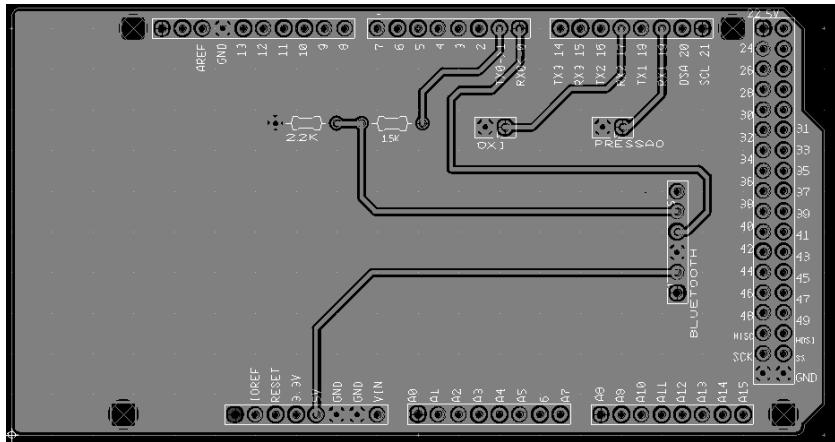


Figura 6 - Layout da placa de circuito impresso desenvolvido com NI Ultiboard.

O protótipo desenvolvido pode ser observado na figura 7, constituído do involucro de madeira com dois botões de controle, um *display*, encaixe para o *Arduino Mega 2560*, e necessidade de 6 pilhas de tipo AAA para o seu correto funcionamento.

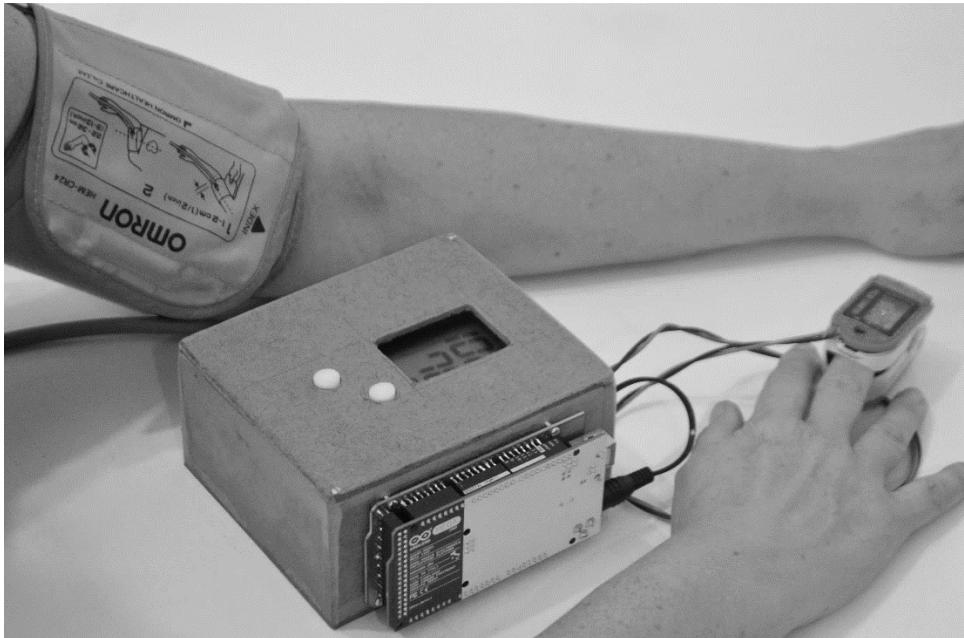


Figura 6 - Imagem exemplar do protótipo em uso.

Resultados e Discussão

Foi concluído com sucesso um protótipo capaz de medir os sinais médicos de um paciente e transmitir remotamente esses dados, sendo esses sinais: pressão sistólica e diastólica, ritmo cardíaco, e oxigenação sanguínea.

O protótipo possui dimensões relativamente reduzidas, entretanto ainda sendo possível o seu aprimoramento, a fim de ter dimensões mais compactas, com maior resistência e leveza. Isso ainda não foi realizado visto que o objetivo do protótipo era a prova de conceito do sistema, e servir como base para futuras expansões. Dessa forma, alguns aspectos do protótipo diferem de um modelo final, como a implementação de um *Shield* para encaixe do *Arduino Mega 2560*, e certas funcionalidades do aplicativo móvel.

Tendo em vista que foi utilizado um servidor local com endereço de *IP* variado, foi necessário implementar um campo para designar o *IP* desejado, algo dispensável caso seja utilizado um serviço de *host* para servidor. Para um invólucro mais aprimorado, seria possível adotar materiais distintos

da madeira, como termoplásticos a fim de facilitar a moldagem e descartar a utilização do *Arduino Mega 2560* em troca de um microcontrolador específico. Certos aspectos do aplicativo móvel também podem ser expandidos, visto que o aplicativo desenvolvido possui capacidade apenas de ler e transmitir os dados remotamente, sendo possível uma maior automatização do aplicativo, uma maior facilidade de uso da interface, e a implementação de um maior número de funções, como a exibição gráfica do histórico do paciente. Além disso a expansão do aplicativo para diferentes sistemas operacionais pode ser realizada, visto que o aplicativo é compatível apenas com o sistema operacional *Android*.

Conclusões

O desenvolvimento do sistema de monitoramento remoto de sinais médicos envolve a utilização de diversos elementos de áreas diferentes, sendo necessário desenvolver programas em diversas linguagens de programação, bem como o trabalho com diversos materiais para desenvolver um invólucro para todo o sistema. O avanço imediato ao projeto seria a continuação do protótipo e a elaboração de um modelo mais aprimorado, eliminando os artifícios implementados para facilitar os testes.

Com o desenvolvimento do projeto, foi possível criar uma base para a comunicação entre um microcontrolador e um servidor de forma remota com entremedio de um aplicativo celular, podendo ser usufruído para futuras expansões com um maior número de sensores de sinais médicos ou em diferentes aplicações que fariam proveito de tal sistema.

Referências Bibliográficas

ANAESTHESIA UK. “Principles of pulse oximetry”, 11 set. 2004.

ANDROID STUDIO. “The Official IDE for Android”. Disponível em:
<<https://developer.android.com/studio/index.html>> Acesso em: 7 de julho de 2017.

Castagnoli, G.; Toyota, L. T. S.; Veríssimo, R. R.; Theodoro, V. O. (2015). Tecnologia Assistiva – Estação Health Care para idosos, Trabalho de Conclusão de Curso do Curso de Engenharia de Controle e Automação do Centro Universitário do Instituto Mauá de Tecnologia, São Caetano do Sul, SP.

OGEDEGBE, G.; PICKERING, T. Principles and techniques of blood pressure measurement. *Cardiology Clinics*, v. 28, n. 4, p. 571-586, 30 abr. 2013.

OLIVEIRA, P. “Base para a medição: Oximetria de pulso”. Disponível em:
<<http://www.discombobulate.me/workshop/arquetipos/e-health/spo2/>> Acesso em: 10 de março de 2017.

Palladino, V. “How wearable heart-rate monitor works, and which is best for you”. Disponível em:
<<https://arstechnica.com/gadgets/2017/04/how-wearable-heart-rate-monitors-work-and-which-is-best-for-you/>> Acesso em: 10 de março de 2017.

Vishik, I. “How do optical heart rate sensors work?”. Disponível em:
<<https://www.quora.com/How-do-optical-heart-rate-sensors-work>> Acesso em: 10 de novembro de 2017.

W3SCHOOLS. “PHP 5 Tutorial”. Disponível em: <<https://www.w3schools.com/php/>> Acesso em: 30 de agosto de 2017.