

Mario Felipe Raulino

***RaspberryPi e RFID no Monitoramento de
Atividades de Natação***

São José – SC

8 de setembro de 2013

Mario Felipe Raulino

***RaspberryPi e RFID no Monitoramento de
Atividades de Natação***

Monografia apresentada à Coordenação do
Curso Superior de Tecnologia em Sistemas
de Telecomunicações do Instituto Federal de
Santa Catarina para a obtenção do diploma de
Tecnólogo em Sistemas de Telecomunicações.

Orientador:

Marcio Henrique Doniak, M.Eng.

CURSO SUPERIOR DE TECNOLOGIA EM SISTEMAS DE TELECOMUNICAÇÕES
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE SANTA CATARINA

São José – SC

8 de setembro de 2013

Monografia sob o título “*RaspberryPi e RFID no Monitoramento de Atividades de Natação*”, defendida por Mario Felipe Raulino e aprovada em 02 de setembro de 2013, em São José, Santa Catarina, pela banca examinadora assim constituída:

Prof. Marcio Henrique Doniak, M.Eng.
Orientador

Prof. Alexandro Andrade, Dr.
IFSC

Prof. Marcelo Maia Sobral, Dr.
IFSC

Resumo

Grande parte das academias e clubes que oferecem atividades de natação não dispõe de uma forma confiável e automatizada para monitorar seus atletas. Em geral, é utilizado um cronômetro para mensurar o tempo de cada volta realizada em um treino. Assim, é necessário o auxílio de outra pessoa, ou ser feito pelo próprio atleta. Essas duas formas apresentam dificuldades e problemas em obter uma análise apurada. Atualmente a natação carece de um sistema para monitoração de treinos.

O presente trabalho constitui na elaboração de um protótipo para acompanhar treinos de natação usando a tecnologia RFID e a placa Raspberry Pi para processar as informações do treino do nadador. Foi realizado um teste funcional que verificou o comportamento do uso destas tecnologias na atividade de natação em uma piscina. Com os testes realizados foi observado como resultados, ser uma forma eficiente de monitoramento de treinos de natação, pois não necessita estar uma pessoa presente para anotar os valores de tempo de cada volta, proporcionando de forma automatizada o cálculo de tempo de cada volta. Além, inclusive, de obter uma análise completa com os valores de velocidade média, quantidade de voltas, tempo médio e distância percorrida no treino, como será apresentado neste trabalho.

Sumário

| | |
|--|--------|
| Lista de Figuras | p. VI |
| Lista de Tabelas | p. VII |
| 1 Introdução | p. 8 |
| 2 Fundamentação Teórica | p. 11 |
| 2.1 Leitor RFID | p. 14 |
| 2.2 Middleware RFID | p. 16 |
| 2.3 Etiqueta RFID | p. 16 |
| 2.4 Proposta de um sistema de monitoramento de atividades de natação | p. 17 |
| 2.5 Linguagem de programação do <i>software</i> | p. 17 |
| 2.6 Java | p. 18 |
| 3 Raspberry Pi | p. 20 |
| 3.1 Java na placa Raspberry Pi | p. 22 |
| 4 Software para o monitoramento de atividades de natação com RFID | p. 23 |
| 4.1 Extração dos Requisitos | p. 23 |
| 4.2 Arquitetura | p. 24 |
| 4.3 Implementação e Codificação | p. 26 |
| 5 Testes do sistema de monitoramento de natação proposto | p. 31 |
| 5.1 Testes Funcionais | p. 31 |
| 5.2 Testes Funcionais na placa Raspberry Pi | p. 33 |

| | | |
|----------|-----------------------------------|------|
| 5.3 | Teste de Campo | p.33 |
| 6 | Limitações | p.38 |
| 7 | Conclusões | p.39 |
| 8 | Trabalhos Futuros | p.40 |
| | Referências Bibliográficas | p.41 |

Lista de Figuras

| | | |
|-----|---|-------|
| 2.1 | Ilustração do Sistema de Monitoramento de Atividades de Natação (MAN) proposto neste trabalho. | p. 17 |
| 2.2 | API de comunicação Serial | p. 19 |
| 3.1 | Modelo Raspberry Pi (FOUNDATION, 2012). | p. 21 |
| 4.1 | Representação do diagrama de classes do <i>software</i> desenvolvido. | p. 25 |
| 4.2 | Classe Cadastramento do plano de treino do nadador. | p. 27 |
| 4.3 | Classe Interface que disponibiliza os valores obtidos da volta. | p. 28 |
| 4.4 | Descrição do Sistema de Monitoramento de Atividades de Natação (MAN) em forma de Fluxograma (adaptado de DONIAK, 2009). | p. 30 |
| 5.1 | Disposição dos leitores RFID na borda da piscina. | p. 34 |

Lista de Tabelas

| | | |
|-----|---|------|
| 5.1 | Tabela dos valores obtidos do treino de dois nadadores. | p.37 |
|-----|---|------|

1 Introdução

A natação é o terceiro esporte mais praticado no Brasil. Uma pesquisa feita pelo Instituto GPP – Planejamento e Pesquisa, demonstra que no último ciclo olímpico de 2004 a 2008, o esporte teve um crescimento para 44,7% no gosto do brasileiro, ficando atrás somente do futebol e do vôlei (GPP, 2009). A análise feita em 2008 pelo IBGE, juntamente com o Confederação Brasileira de Desportos Aquáticos (CBDA), contabilizou cerca de 14 mil atletas registrados na confederação (MOURA, 2010).

O estudo realizado por Ricardo Moura intitulado Planejamento Estratégico da Natação de 2010 a 2016 (MOURA, 2010), apresenta uma pesquisa realizada pelo IBGE, juntamente com Confederação Inglesa de Natação, onde consta que países como a Rússia e Nova Zelândia possuem cerca de 135 mil pessoas praticantes da natação cada. Em comparação, o Brasil ainda carece no crescimento deste esporte, não apenas de praticantes, mas estrutural. (MOURA, 2010) também lista uma série de necessidades para favorecer o crescimento desse esporte no Brasil, tais como: complementos financeiros, estruturas, sistema de um controle de treinamento, materiais, eventos, parques aquáticos entre outros.

A evolução técnica da natação proporcionou melhorias significativas do esporte em jogos olímpicos. Entre 2004 e 2007 o Brasil teve um aumento de 21% na quantidade de atletas masculinos entre os dez primeiros colocados nas provas (MOURA, 2010). O Dr. Paulo César Marinho, biomecânico da CBDA e da equipe de natação do Serviço Social da Indústria do Estado de São Paulo (Sesi-SP), trabalhando com um dos atletas ícones da natação brasileira, relata a necessidade de obter uma análise técnica do atleta para sua evolução no esporte. “Com a análise são levantados os dados sobre a velocidade média atingida pelos nadadores, além da amplitude e frequência de braçadas. Com isso, buscamos o nado ideal para cada atleta. A tendência é a melhora gradual do tempo de prova e do rendimento. Não há medalha sem uma análise técnica minuciosa.” (SESI-SP, 2013).

Para um atleta de nível profissional que conta com uma equipe multidisciplinar responsável pelo seu treinamento, o uso de tecnologias no seu treinamento é essencial para analisar seus

movimentos e técnicas de forma minuciosa. Porém, este tipo de análise necessita de diversos equipamentos e profissionais o acompanhando. No caso de atletas amadores, que normalmente não contam com uma equipe acompanhando o seu treinamento, a análise do treino não é menos importante. Entretanto, eles necessitam de uma forma mais simples e viável para acompanhar o cotidiano de treinos para, assim como o nadadores profissionais, sua evolução ser monitorada.

O monitoramento de atividades de natação, assim como o de qualquer outra atividade física, provê diversos benefícios ao atleta. Por exemplo, o acompanhamento mostra ao atleta se o desempenho realizado está de acordo com a proposta elaborada pelo técnico. Também é possível mostrar a evolução do atleta ao longo de um ou mais ciclos, ajuda a controlar e evitar *overtraining*¹ e revela ao técnico e ao atleta os pontos fortes e fracos. É importante para o nadador obter informações de seu treino, como a quantidade de voltas e o tempo de cada volta feita em seu treino. Além de ajudar a manter o nadador motivado, ativo e saudável.

Grande parte dos clubes brasileiros não utilizam uma forma adequada em obter uma análise completa e gradual de um treino. Um dos problemas é contar a quantidade de voltas na piscina. Geralmente é feita pelo próprio nadador e depende de sua memória para realizar a contagem. Dessa forma, diminui sua concentração no treino e em seus movimentos. Ao longo do treinamento o cansaço irá tornar a contabilização de voltas cada vez mais difícil (DONIAK, 2009).

Além de contabilizar o número de voltas, outro problema é medir o tempo de cada volta. A forma mais simples de resolver este problema é o nadador iniciar e parar o cronômetro, registrando o tempo de cada nova volta. Este procedimento tem um erro inerente da ação do nadador. Como não há um equipamento para obter remotamente os valores, uma alternativa seria uma pessoa ficar acompanhando o treino, contabilizando as voltas realizadas e registrando o tempo de cada volta, por exemplo.

Ao contrário da corrida, onde os atletas possuem relógios com cronômetro, GPS e pedômetro (equipamento que conta as passadas e indica a distância percorrida), a natação devido ao meio aquático, não existe um equipamento que acompanhe o nadador, que seja economicamente viável e apresente uma baixa margem de erro. Assim, o nadador fica desassistido de uma ferramenta que seja capaz de acompanhar o seu treino.

Este trabalho propõe criar um sistema de monitoramento de atividades de natação, que seja viável economicamente e capaz de acompanhar um treino com eficiência. Também desenvolver um *software* para processar os dados obtidos do treino em um sistema embarcado, chamado Raspberry Pi (FOUNDATION, 2013), utilizando a tecnologia de Identificação por Radiofrequência (RFID).

¹ Condição resultante de se fazer mais exercícios do que seu corpo é capaz de se recuperar.

O Raspberry Pi é um computador de pequeno porte, aproximadamente com dimensões de um cartão de crédito, que segue a seguinte estrutura: um microcontrolador, unidade de processamento gráfico, memória de acesso aleatório (Memória RAM), saídas de áudio e vídeo (FOUNDATION, 2013). Essa placa possui um *hardware* bem estruturado o que permite uma fácil instalação de um sistema operacional. Sua facilidade pode-se comparar a uma instalação em um computador pessoal, podendo também escolher variados sistemas operacionais. Neste trabalho será utilizado o sistema operacional Linux devido a familiaridade e facilidade com o mesmo.

Foi desenvolvido um *software* para obter de forma eficiente os valores de tempo de cada volta, o número de voltas, a velocidade média e a distância percorrida. O *software* também irá proporcionar a visualização dos valores ao vivo em uma interface gráfica e ao final produzir um relatório total do treino.

O texto deste trabalho aborda a fundamentação teórica, que descreve uma visão geral da natação, a infraestrutura dos centros de treinamento e as formas mais utilizadas em obter valores sobre o treino de natação. Com a descrição os elementos e equipamentos envolvidos no sistema, ou seja, RFID, o *hardware* Raspberry Pi, linguagem de programação e descrição do *software* desenvolvido, além dos testes mostrando a usabilidade do sistema de monitoramento de natação e suas possíveis limitações.

2 *Fundamentação Teórica*

A natação é um esporte aquático que pode ser praticado no mar, rio, lago ou piscina. Nos centros urbanos é praticado essencialmente em piscinas. A natação é um esporte olímpico desde a realização dos primeiros Jogos Olímpicos da Era Moderna em Atenas, Grécia, em 1896. Na primeira participação brasileira nos Jogos Olímpicos, em 1920, na Antuérpia, o Brasil competiu com cinco atletas as provas de natação (OLYMPIC, 2013). Um bom desempenho em Jogos Olímpicos e em campeonatos mundiais resulta em uma maior exposição na mídia e, conseqüentemente, uma maior divulgação da modalidade para a população. Essa exposição na mídia chamou a atenção de patrocinadores que permitiu uma melhor formação dos atletas através da profissionalização do esporte. Com patrocínio, a confederação e os atletas puderam se aperfeiçoar e passar a viver apenas do esporte. E os resultados começaram a surgir. As competições de natação, assim como, os registros de recordes e as regras da natação são definidos pela Federação Internacional de Natação (FINA, 2013) e cada país possui sua entidade associada a FINA, sendo no Brasil a Confederação Brasileira de Desportos Aquáticos (CBDA, 2013).

No Brasil, a natação figura como um dos esportes mais praticados sendo estimado mais de 11 milhões de adeptos (DACOSTA, 2006). Todo esporte para ser bem sucedido necessita de apoio político e financeiro, assim, ele pode ser levado mais próximo às comunidades para que estas façam uso da atividade. Desfrutando, dessa forma, dos seus benefícios diretos e indiretos. Com o intuito de ajudar ainda mais a prática desportiva no Brasil e a realização dos Jogos Olímpicos no Rio de Janeiro em 2016, o Governo Federal lançou em 2012 o Plano Brasil Medalha 2016 (MINISTERIO DO ESPORTE, 2012) sendo investido R\$ 1 bilhão em atletas e infraestrutura para treinamento. Esse projeto selecionou 21 modalidades olímpicas e 15 paralímpicas que possuem maiores chances de conquista de medalha. Com isso, além de apoiar atletas olímpicos, o país também estará difundindo a prática esportiva dentre sua população, incentivando cada vez mais a cultivar o esporte no seu cotidiano.

No desenvolvimento tecnológico, a fabricante de relógios Omega é a pioneira em desenvolver equipamentos para natação. A marca trabalha em diversos eventos, frequentemente com

a FINA e a CBDA entre outras confederações dos demais países. Em 1968 a Omega foi inserida nos Jogos da Cidade do México para cronometrar as variadas modalidades, iniciando uma parceria com o Comitê Olímpico Internacional. Já em 2008 a marca começou a utilizar câmeras de vídeo usadas nos eventos esportivos e, em 2012 esta tecnologia recebe uma melhora na precisão e resolução de vídeo, solidificando a sua utilização em eventos internacionais de natação (OMEGA-WATCHES, 2013).

Um dos principais auxílios que a CBDA possui é um sistema básico da Omega para monitoração, sendo utilizado principalmente em competições. O Ares 21 é um equipamento utilizado em variados esportes aquáticos, pois é flexível nas configurações dos seus requisitos. Tendo uma entrada e saída serial permitindo ser ligado a um computador, um *software* também o acompanha para gerenciar os dados e salvá-los em disco ou fazer impressão (OMEGA, 2009).

Um fator importante para o desenvolvimento físico do atleta é a obtenção dos dados sobre seus treinos. O atleta obtendo as informações perante seu treino faz com que possa aprimorar suas técnicas e melhorar o seu desempenho. Assim consegue alcançar suas metas sem causar lesões, tendo uma evolução gradativa de seu desempenho. Quando tem informações sobre o seu desenvolvimento físico, e consegue alcançar suas metas sentindo-se saudável, ele vai à busca de mais aprimoramento para o seu treino. Ou seja, o atleta passa a se preocupar com sua alimentação, seu estado emocional, sua fisiologia e seu psicológico. Isso pode ser observado em diversas assessorias esportivas que oferecem auxílio aos seus atletas em todas essas áreas.

É através do treinamento que os atletas alcançam suas melhores formas e técnicas em seu esporte. São necessários alguns fatores que viabilizam o aprimoramento nos centros de treinamento, como financeiros, de infraestrutura e tecnológicos. Para estar em constante treinamento o atleta normalmente se vincula a um clube ou academia. Os centros de treinamentos e estrutura organizacional da natação brasileira são compostos de clubes, juntamente com as federações estaduais e a CBDA.

São nos clubes de natação que o atleta encontra a infraestrutura adequada para seu treinamento. Tomando como exemplo o Clube Curitibano, um dos principais clubes de esporte da cidade de Curitiba, Paraná, sua infraestrutura é composta por técnicos, preparadores físicos, psicólogos, fisiologistas e a biomecânica. Sua equipe competitiva tem um planejamento anual em ciclos bem definidos, com fases onde são desenvolvidas as capacidades físicas e motoras e os estilos de nado. Portanto, é um clube que tem uma estrutura bem desenvolvida para prática da natação.

Porém a grande parte das academias e centros de treinamentos aquáticos não possui uma infraestrutura que proporcione a obtenção dos dados de uma maneira confiável, como a quan-

tidade de voltas e tempo de cada volta de treino dos atletas. Por apresentarem um alto valor comercial, o investimento em equipamentos acaba ficando fora do orçamento das academias.

Nas academias de natação que não possuem uma forma de contabilizar a quantidade de voltas e nem de registrar o tempo de cada volta dos nadadores, a obtenção das informações do treino realizado pelo nadador ou são feitas pelo próprio nadador ou por um observador. Normalmente quando o nadador conta a quantidade de voltas, ele não consegue registrar o tempo de cada volta. Quando um observador fica acompanhando o treino de um atleta, ele pode anotar o tempo de cada volta em uma planilha e calcular a velocidade. Entretanto, existe o erro inerente de medida do observador que varia de medida para medida e de observador para observador. Para registrar o tempo das voltas é utilizado o cronômetro. Devido ao erro humano no parar e iniciar o cronômetro, este equipamento não faz uma medida precisa adicionando um erro no tempo das voltas. Se forem mensuradas pelo próprio atleta ocorre uma descontinuidade no treinamento do atleta, devido ao tempo manuseando o cronômetro e anotando os valores.

Uma vez conhecido o tempo de cada volta, e sabendo o tamanho da piscina, pode-se calcular a distância total percorrida, o tempo total de treino, além da velocidade média de cada volta e total do treino. De posse destas informações os nadadores poderão planejar melhor os seus treinos, resultando em evoluções mais consistentes e rápidas. Podendo até calcular a quantidade de calorias gastas durante um treinamento.

Para que haja uma evolução no desempenho do nadador, ele deve respeitar a prescrição do seu treino. Quando o treinador determina que uma volta seja realizada em um determinado tempo, o nadador não pode ser nem muito mais rápido nem muito mais lento, existindo um limiar aceito pelo treinador. E com o treinador normalmente tem diversos atletas treinando simultaneamente. Ele não consegue acompanhar exclusivamente um nadador. Desta forma, um produto que acompanhe volta a volta a performance do atleta, mostrando o desempenho em tempo real e entregando um relatório ao seu final é muito importante para que o treinador saiba sobre a efetividade do treino prescrito. Portanto, um acompanhamento é muito importante na avaliação do atleta.

Para fazer esse acompanhamento de treino algumas soluções já estão disponíveis atualmente aos nadadores e academias de natação. Uma delas é a instalação de sensores de toque nas bordas da piscina. Este produto é fabricado por diversas empresas, dentre elas, a Colorado Time Systems (COLORADO, 2013) e a Omega. Porém, estes equipamentos são encontrados em apenas algumas das principais competições nacionais, estaduais e em alguns dos principais clubes brasileiros. Geralmente estando presentes nos blocos de saída da piscina e nas bordas. Assim, quando o atleta inicia a volta saindo do bloco, inicia-se a contagem de tempo. Exis-

tem outros fabricantes de sensores, como a Seiko (SEIKO-WATCH, 2007-2011), parceira da FINA entre 1999 à 2001 e a Daktronics (DAKTRONICS, 2013) popular nos Estados Unidos (SWIMITUP!, 2013). Para aquisição destes equipamentos é necessário um alto investimento, inviabilizando a popularização destes sensores. Em academias, onde em determinados horários, os nadadores compartilham uma mesma raia, muitas vezes chegando a nadar até três atletas em uma mesma raia, este tipo de sensor de toque não é viável, pois, não tem uma identificação personalizada do nadador.

No estudo realizado por (GREFF, 2009) e (DONIAK, 2009) foi analisado e proposto uma forma de contar as voltas do nadador na piscina sem o uso de sensores de toque, mas através de identificação por Radiofrequência (RFID). O atleta utiliza etiqueta para identificação. Cada etiqueta apresenta uma série de números e letras chamados de identificação (ID) da TAG¹. Um leitor é posicionado na piscina de tal forma que o nadador portando a etiqueta entre dentro da área de cobertura.

A etiqueta deve ser capturada pelo leitor de forma rápida no início e virada da volta, por isso uma solução é utilizar duas etiquetas. Uma etiqueta presente ao tornozelo e outra presente ao pulso. O fato dos sensores estarem presentes nas bordas faz com que seja lido a TAG no início, na virada ou em uma parada de cada volta feita pelo nadador.

Com isso este sistema permite mensurar a quantidade de voltas, registrar e calcular o tempo de cada volta na piscina. Conhecendo-se também, o tamanho da piscina, é possível calcular a velocidade média na volta, a distância total percorrida, a velocidade média do treino, entre outras informações, que também podem ser exploradas.

Em continuidade aos estudos realizado por (GREFF, 2009) e (DONIAK, 2009), surge o tema deste projeto, a criação do protótipo do Sistema de Monitoramento de Atividades de Natação (MAN). Com o sistema de RFID, o middleware, a placa Raspberry Pi e um *software* com ambiente gráfico que disponibilize as informações para o usuário final.

2.1 Leitor RFID

A Identificação por radiofrequência (RFID) é uma tecnologia que pode ser utilizada em diversos segmentos, como para identificação, controle e rastreamento. Itens marcados, como parcelas, joias, roupas ou qualquer outro objeto podem ser identificados através de ondas de radiofrequência com leitores de RFID. As principais aplicações que a utilizam são antirroubos e sistemas de controle de acesso, além de atuais como para a medicina, com um microchip in-

¹TAG, ou em português etiqueta, é uma palavra-chave ou termo associado com uma informação

terno em um paciente coletando informações, pode-se obter com leitor de RFID as informações fisiológicas do paciente (VIZINEX-RFID, 2013). Um sistema de RFID é composto por uma antena, um transceptor e também um transponder ou etiqueta de radiofrequência (RF), que deverá conter o circuito e a informação a ser transmitida. Estas etiquetas podem estar presentes em pessoas, animais, produtos, embalagens, enfim, em equipamentos diversos.

Um sistema de RFID pode trabalhar em diversas faixas de frequências, podendo variar nas baixas frequências (LF), de 125 KHz por exemplo, alta frequência (HF), de 13,56 MHz, e ultra alta frequência (UHF), operando na faixa de 860 a 960MHz. No geral, a frequência está ligada a velocidade de transferência de informações entre TAG e o leitor. Porém, esta velocidade não é o aspecto mais importante para um sistema de RFID. Deve-se levar em conta o alcance da onda eletromagnética para leitura entre a TAG e o leitor de RFID, o tamanho e o projeto das antenas das etiquetas e dos leitores são diferentes para as diferentes frequências. Podemos assim dizer que cada frequência tem características diferentes e, por isso, aplicações diferentes (BALADEI, 2013).

Contundo o leitor de RFID utilizado no sistema de Monitoramento de Atividades de Natação (MAN), estará presente no ambiente aquático dificultando ainda mais a propagação da onda eletromagnética. Por isso, o IFSC - Campus São José adquiriu um equipamento de baixa frequência, de 125 kHz, para testar esta aplicação. Este leitor foi escolhido devido as ondas de radiofrequência ter dificuldades para se propagar na água e quanto mais baixa a frequência melhor é a propagação. Entretanto, ainda não foi encontrado um leitor de RFID dedicado para a aplicação do MAN. Para utilizar o leitor de RFID conectado a placa de desenvolvimento, será necessária um conversor serial RS232 para interface USB. Há no mercado conversores prontos que saem com custo muito baixo. A fim de obter facilidade, neste projeto será utilizado este conversor.

A identificação coletada pelo leitor de RFID entregará as informações a um *middleware* utilizando uma interface serial no padrão RS-232. Este padrão é utilizado em conexões curtas, pois tem pouco controle aos níveis de tensão podendo ocasionar problemas na comunicação. As tensões de sinais variam de 3 a 15 volts, sendo o nível lógico 0 é tensão positiva e o nível lógico 1 é tensão negativa. Os bits são entregues em fila, por isto o termo serial, fisicamente ligado ao *middleware* (FOUNDATION, 2012).

2.2 Middleware RFID

Um *middleware* serve para transportar informações entre *softwares* de diferentes linguagens, ou ligando *softwares* a diferentes plataformas. Mascando a heterogeneidade e assim, obtendo uma única aplicação facilitando o desenvolvimento de um sistema distribuído.

Há no mercado empresas especializadas na criação de *middleware* RFID, além de empresas fabricantes de sistemas integrados de gestão empresarial, sistemas conhecidos como ERP ou SIGE. Geralmente estas empresas além de criarem os módulos a serem adicionados ao sistema de RFID, também incluem suporte.

Existem sistemas RFID que são capazes de ler inúmeros eventos por segundo, assim surge a necessidade de abstrair esses fluxos de dados gerados (*streams*). Como solução haveria a necessidade de criar um *middleware* integrando cada *software*, para que o processamento dos *streams* transmitidos seja possível. Desta forma, surge o conceito de *middleware* RFID, o *middleware* é responsável pela troca de dados entre a unidade de processamento e o leitor. Ou seja, em sua forma mais simples, pelo envio do ID lido pelo leitor RFID para a unidade de processamento. Mas, quando pode trocar conteúdo, para ser armazenado na etiqueta, o *middleware* também faz este gerenciamento.

As etiquetas RFID possuem um chip e memória em seu interior. A principal função do *middleware* é enviar para a unidade de processamento o ID da etiqueta lida. Simples assim, tão logo o ID seja recebido é disponibilizado na interface de comunicação com a unidade de processamento. No projeto, o *middleware* transportará o valor do ID das TAG coletadas para serem processadas no *software* desenvolvido. Tendo estas características, um *software* é desenvolvido para processar e disponibilizar os dados na forma desejada.

2.3 Etiqueta RFID

Em um sistema de RFID podem ser utilizados três tipos de etiquetas: ativa, passiva e semi-passiva. A etiqueta RFID ativa utiliza uma bateria que pode ser utilizada como uma fonte de energia parcial ou completa para o circuito e para a antena da etiqueta. Algumas etiquetas ativas contêm baterias substituíveis para anos de uso e outras são unidades seladas. A etiqueta passiva utiliza as ondas eletromagnéticas do leitor de RFID como fonte de energia. Quando as ondas eletromagnéticas do leitor chegam à etiqueta passiva, a antena dentro da etiqueta forma um campo magnético, criando uma corrente induzida na etiqueta. Assim, ela terá potência para transmitir seu ID ao leitor, retirando a energia do leitor, transmitindo energia aos seus circuitos.

A etiqueta então envia as informações presentes na sua memória para o leitor (IBM, 2013).

2.4 Proposta de um sistema de monitoramento de atividades de natação

Como definido nos objetivos, este projeto implementará a proposta definida por (DONIAK, 2009) e (GREFF, 2009), que é um Sistema de Monitoramento de Atividades de Natação com o uso de RFID e a placa Raspberry Pi. Para criação deste sistema, será necessário o leitor de RFID, juntamente com as etiquetas contendo os ID's das TAG's, um *middleware* e um *software* que calcule e disponibilize a quantidade, tempo e velocidade de cada volta. Obtendo assim uma análise do desempenho do atleta, de acordo com a Figura *ilustMAN*.



Figura 2.1: Ilustração do Sistema de Monitoramento de Atividades de Natação (MAN) proposto neste trabalho.

2.5 Linguagem de programação do *software*

Para este projeto podem ser escolhidas variadas plataformas e linguagens de programação como, C, C#, C++, Python, PHP e Java. Cada qual com suas características tendo seus prós e contras. Não existe uma linguagem perfeita e, sim, fatores que servem para escolher a linguagem adequada à aplicação.

Um fator importante para o desenvolvimento do *software* é a continuidade de seu desenvolvimento. Ao longo do crescimento do programa e o aumento na demanda de desenvolvedores, o *software* deve fornecer condições para sua continuidade. Outro fator é a velocidade de desenvolvimento, fator correspondente ao esforço e horas gastas pelo desenvolvedor para implementar determinada funcionalidade no sistema. A portabilidade e agregação do *software* em variados ambientes e tecnologias também devem ser levados em conta.

Devido às horas de aprendizado, afinidade do desenvolvedor e continuidade do projeto aos demais desenvolvedores, neste projeto a linguagem de programação escolhida é o Java.

2.6 Java

A linguagem de Programação Java foi criada em 1993 por um grupo de engenheiros da Sun Microsystems (CAELUM, 2004). Inicialmente sua usabilidade seria na TV digital, mas ao longo do caminho seguiu em outro rumo. Começou em aplicações para clientes web, sendo hoje utilizado na parte de servidor web e também em variados projetos de sistemas computacionais.

A virtude de utilizar o Java é que seu código executável não é atrelado a cada sistema operacional, ou seja, o Java é independente do sistema operacional. O Java utiliza o conceito de máquina virtual, seria uma camada entre o código binário Java (Bytecode Java) e o sistema operacional, denominando-se Java Virtual Machine (JVM). O conceito de máquina virtual é bem amplo, pois utiliza a estratégia de “computador de mentira”, ele gerencia memória, *threads*, pilha de execução e etc.

O uso de máquina virtual é bem vinda aos sistemas computacionais, como também uma linguagem de programação que utiliza orientação a objeto. Tendo também inúmeras bibliotecas que auxiliam em diversas aplicações, facilidades de criação de interface gráfica e entre outros. Por estas peculiaridades o por quê desta adoção da linguagem de programação Java ao Sistema de monitoramento de natação.

Para o sistema de monitoramento de atividades de natação proposto será necessário a utilização da biblioteca para comunicação serial. A API Java Communications juntamente com a JVM, possui uma interface de programação de aplicativos (API) com bibliotecas prontas para o uso na comunicação serial. Podendo ser utilizada não só na comunicação serial como comunicação paralela. A API é idealizada por uma classe abstrata chamada *CommPort* e seguida de duas subclasses *SerialPort*, para utilizar em comunicação serial, e *ParallelPort*, para utilizar em comunicação paralela. Pode-se observar um diagrama pela Figura 2.2. Está API é disponibilizada gratuitamente pelo site da SUN, pode-se também importá-la para uma IDE (Ambiente de Desenvolvimento Integrado) para desenvolvimento do *software*. Com isso facilita a criação do *software* que irá compor o sistema.

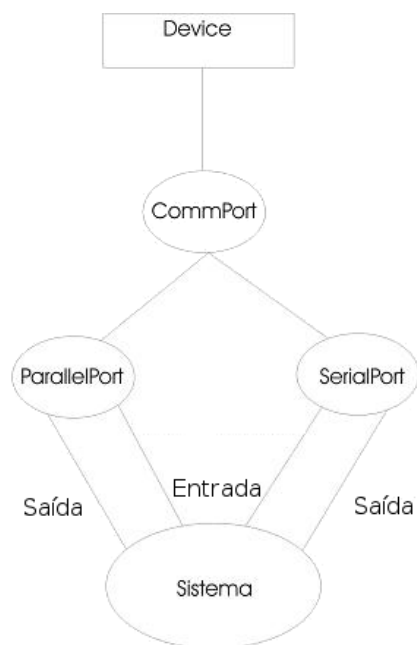


Figura 2.2: API de comunicação Serial

3 *Raspberry Pi*

A unidade de processamento escolhido para este projeto será um computador de pequeno porte chamado Raspberry Pi. A escolha desta placa foi feita pelo seu tamanho e funcionalidade. Seu tamanho tem proporção de um cartão de crédito e suas funcionalidades são como de um computador pessoal (FOUNDATION, 2012).

O Raspberry Pi, chamado também de RasPi ou RPI, é um computador de pequeno porte inicialmente criado com intuito de ensinar programação para crianças. Foi criado e registrado pela Raspberry Pi Foundation, uma instituição sem fins lucrativos do Reino Unido (FOUNDATION, 2013). Contendo custo baixo, entre \$25 e \$35 dólares, com isso podendo chegar a lugares com alto índice de pobreza. Porém, seu caminho foi muito além da educação, surgindo inúmeras aplicações, como: multimídias, jogos, e até centrais telefônicas.

Sua estrutura de *hardware* passou por modificações ao longo do tempo. Um primeiro protótipo chamado *Raspberry Pi Concept 2006 edition*, criado em 2006, uma placa utilizando um microcontrolador da Atmel ATMEGA644 com clock de 22.1 MHz, e uma SRAM (Memória de acesso aleatório estática) 512 KB para armazenamento de dados e com saída de vídeo de 320 por 240 linhas.

Em maio de 2011, o programador de jogos David Braben e alguns colegas realizaram uma matéria para BBC, apresentando o *Raspberry Pi USB Prototype Board*. Segundo protótipo do Raspberry Pi em formato de um cartão USB, contendo câmera e saída de vídeo HDMI. Em agosto de 2011 então foi feito o protótipo em formato do atual Raspberry Pi. Logo em dezembro do mesmo ano, foi criado o *Raspberry Pi Beta Production Board*. Primeiro protótipo de Raspberry Pi em escala de vendas, sendo vendido pela eBay. O grupo de desenvolvimento da empresa NokiaQt doou um valor para produção de 400 unidades de RPI's.

O Raspberry Pi possui dois modelos, modelo A e modelo B, para o projeto será utilizado o modelo B. Atualmente apenas o modelo A é produzido. Portanto, será utilizado o modelo antigo, que não é mais usado atualmente. Sua estrutura é composta por um microcontrolador ARM11 de 700 MHz, unidade de processamento gráfico, memória RAM com 256 MB no

modelo B e 512 MB no modelo A, placa ethernet, entrada para cartão SD, saídas de vídeo RCA e HDMI, saída de áudio, duas portas USB 2.0 e utiliza padrão USB de 5 volts e 1 Ampere para seu funcionamento elétrico. Sua estrutura pode ser vista na Figura 3.1.

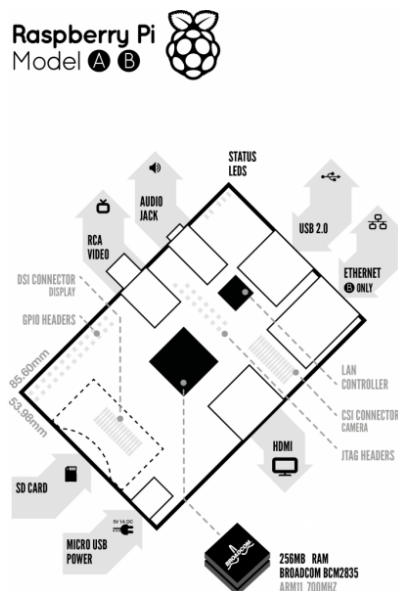


Figura 3.1: Modelo Raspberry Pi (FOUNDATION, 2012).

O Raspberry Pi funciona como um computador, portanto é necessário ser instalado um Sistema Operacional (SO), como Linux ou Windows. Neste projeto será utilizado o Sistema Operacional Linux. Há também variadas distribuições do SO Linux que podem ser instaladas no Raspberry Pi. Para utilização no projeto foi escolhido o GNU/Linux Debian versão Wheezy (MURDOCK, 1993), devido a sua facilidade de utilização. Pode-se obter uma imagem do SO pronta para utilização no site oficial do Raspberry Pi, a imagem vem estruturada para ser salva em algum padrão de memória flash, *Pen drive* ou *cartão SD*.

Para entender o funcionamento do Raspberry Pi é necessário entender o seu processo de boot. O *bootloader* é o processo de iniciação do computador que carrega o sistema operacional quando a máquina é ligada. Ao ligar o Raspberry Pi, a GPU (Graphics Processing Unit, ou Unidade de Processamento Gráfico) assume, carrega um arquivo chamado *bootcode.bin* para a memória *cache L2* (Dispositivo de acesso rápido) e o executa. O *bootcode.bin* inicializa a SDRAM (Memória de Acesso Aleatório dinâmica síncrona) e carrega um outro arquivo chamado *start.elf* para a RAM (Memória de Acesso Aleatório) externa e o executa. O *start.elf* é responsável por configurar o *hardware* de acordo com o arquivo *config.txt*, carregar a imagem do kernel contida no arquivo *kernel.img* ler a linha de comandos do kernel no arquivo *cmdline.txt*, iniciando assim, o sistema operacional Linux. Todos estes arquivos devem estar na primeira partição do cartão SD ou Pen drive. Estes arquivos podem ser manipulados, obtendo um funcionamento específico para um projeto utilizando o RPI. Os parâmetros para configuração es-

pecífica do arquivo *config.txt*, pode-se obter no sítio http://elinux.org/RPi_config.txt. Para este projeto será utilizado a forma padrão, ou seja, o arquivo *config.txt* pré-configurado da imagem obtida no sítio oficial do Raspberry (FOUNDATION, 2013).

A imagem do sistema operacional é estruturada dentro da memória flash. Não é necessário instalação, sendo após a finalização do processo de boot o SO GNU/Linux Debian estará em produção. O GNU/Linux Debian segue a estrutura dos variados sistemas operacionais Linux, um sistema multiusuário e multitarefa.

O SO Linux é um sistema composto por elementos que formam uma plataforma, o Kernel, que é o núcleo do sistema, ferramentas e *softwares* de aplicações. O Kernel é o elemento base do sistema, possui inúmeros códigos de programação essenciais ao sistema. Como por exemplo, acesso a memória RAM, ao disco rígido e demais *hardwares* ligados ao computador.

3.1 Java na placa Raspberry Pi

Em uma aplicação Java é necessário um *software*, chamado *Java Runtime Environment*, instalado no Sistema Operacional para conseguir executá-lo. No Raspberry Pi utilizando Linux, será necessário a instalação deste *software*. Este *software* é disponibilizado pelo sistema de pacotes do GNU/Linux Debian, *APT*. Para sua instalação é necessário estar com acesso a internet e utilizando um terminal, com o comando `# apt-get install openjdk-jre`, o GNU/Linux Debian estará pronto para rodar o *software* desenvolvido em Java.

O Java é uma linguagem que possui alto consumo de memória RAM em um sistema. Por utilizar uma máquina virtual, chamada JVM, para ser independente dos sistemas operacionais. É de suma importância ao desenvolvedor do *software* procurar gerenciar o uso da memória no código fonte do *software*, com lógicas que tenham poucos códigos e o uso adequado de alocação de memória. Apesar do alto consumo de memória RAM, a própria JVM é constituída de ferramentas que auxiliam nesse gerenciamento de memória, com isso facilitará no desenvolvimento do *software* neste projeto.

Um planejamento no ambiente do sistema operacional do Raspberry Pi também auxilia na disponibilidade de memória RAM ao *software* desenvolvido. A utilização de um gerenciador de janelas leve decrementa o uso de memória RAM por parte da GPU. Também deixar em execução apenas os *softwares* essenciais para o sistema operacional, faz com que disponibilize memória RAM para o uso do Java.

4 Software para o monitoramento de atividades de natação com RFID

Tendo a finalidade de auxílio ao atleta de natação, o Sistema de Monitoramento de Atividades de Natação (MAN) é fundamentado e estruturado para ser um equipamento de baixo custo e estar sendo utilizados em academias, escolas ou em qualquer clube que ofereça atividades de natação. A fim de demonstrar sua usabilidade, este projeto também visa o desenvolvimento de um *software* para o monitoramento de atividades de natação com RFID.

Nos capítulos anteriores foram descritos os componentes do sistema de monitoramento de atividades de natação proposto: o RFID, a placa Raspberry Pi e a linguagem de programação Java. Neste capítulo é descrito o processo de desenvolvimento do *software* que será responsável por adquirir e calcular os parâmetros envolvidos na atividade de natação.

Neste projeto foram definidas quatro etapas para o desenvolvimento do software. Primeiro a extração dos requisitos desejados para o usuário final e clientes, arquitetura, implementação ou codificação e os testes. A extração dos requisitos consiste em: reconhecer o problema, avaliar o problema, modelar e especificar os requisitos e, por fim, fazer a revisão. A arquitetura consiste em direcionar o projeto a sua linguagem e estrutura. A implementação é codificar os requisitos especificados para o código fonte executável do projeto. Por fim, o teste funcional e de campo do software e do sistema.

4.1 Extração dos Requisitos

A extração dos requisitos envolve o reconhecimento do problema a ser atingido. No capítulo 2, foram analisados os problemas enfrentados pelos atletas na obtenção e disponibilidade dos valores em um treinamento de natação. Como solução descrita neste projeto, a utilização de um sistema em uma plataforma computacional. Contendo um *software* possibilitando o cálculo e obtenção dos valores de tempo e quantidade de voltas de forma eficiente em um treinamento de natação. Ao informar o tamanho da piscina o software pode calcular a velocidade média,

guardar quantidade de repetições do treino, abrir um tempo de descanso e ao final apresentar um relatório com estes dados em uma interface gráfica.

4.2 Arquitetura

Arquitetura de software é a análise da estrutura no qual o software é construído e moldado. Segundo (HILL, 2011), uma arquitetura de software pode ser definida como: “A arquitetura de software de um programa ou sistema computacional é a estrutura ou as estruturas do sistema, que abrange os componentes de software, as propriedades externamente visíveis desses componentes e as relações entre eles.”

Há três abstrações básicas para entender a arquitetura de software, que levam os nomes de componentes, conectores e configuração. Os componentes neste projeto são os objetos construídos das classes utilizadas no sistema, sendo objetos e classes os termos utilizados na orientação a objeto da linguagem de programação Java. Utiliza-se também uma API contendo uma série de classes já desenvolvidas. As classes utilizadas são autônomas, ou seja, uma não depende da outra.

Os conectores são como conexões que transmitem informações entre os objetos das classes. As classes podem ter comunicação uma com a outra, quando se instância um objeto, cria-se um vínculo de comunicação entre eles. A configuração é a estrutura do software de como estão vinculado às classes, assim criando visualmente o sistema.

Para visualizar a arquitetura do *software* em Java, será utilizado a representação de classe na forma de UML¹. Esta representação auxilia na implementação do código fonte, observa-se as classes com seus devidos atributos e métodos. Assim, facilita na lógica para solução dos requisitos do sistema.

A Figura 4.1 representa o diagrama de classe do *software* desenvolvido. Neste *software*, foram criados cinco classes Java nomeadas de: Cadastramento, SerialCommCadastro, Interface, SerialCommPrograma e Relatório. A classe Cadastramento tem a finalidade de cadastrar o nadador para seu atual treino. Nela serão preenchidos informações como o nome do nadador, o tamanho da piscina, tempo de descanso, quantidade de voltas, quantidade de repetições e valores dos ID's das TAG's. A classe SerialCommCadastro tem como função fornecer para a classe Cadastramento os valores dos ID's das TAG's. A classe Interface tem a finalidade de apresentar os dados obtidos no treino em uma interface gráfica para o nadador visualizar o andamento do seu treino. A classe SerialCommPrograma obtém os valores de ID das TAG's

¹Unified Modeling Language.

durante o treino, calcula o tempo de cada volta e a quantidade de voltas do treino atual. Por fim, a classe Relatório fornece o relatório final com todas as informações do treino atual: quantidade de voltas total, tempo total e de cada volta, distância total percorrida, quantidade de repetições feitas e velocidade média de cada volta e total do treino.

Para haver comunicação entre as classes do *software* são utilizados os objetos, conforme está ilustrado na Figura 4.1. A classe Cadastramento instância um objeto da classe SerialCommCadastro afim de obter os valores dos ID's das TAG's. Como também, instância um objeto da classe Interface para fornecer os valores do cadastro e serem utilizados ao longo da execução do *software*. A classe SerialCommPrograma instância objetos das classes Interface e Relatório, afim de fornecer o tempo e quantidade de cada volta do treino monitorado. O relatório recebe os valores obtidos durante todo o treino monitorado e, assim, apresenta todas as informações obtidas durante o seu treino em interface gráfica.

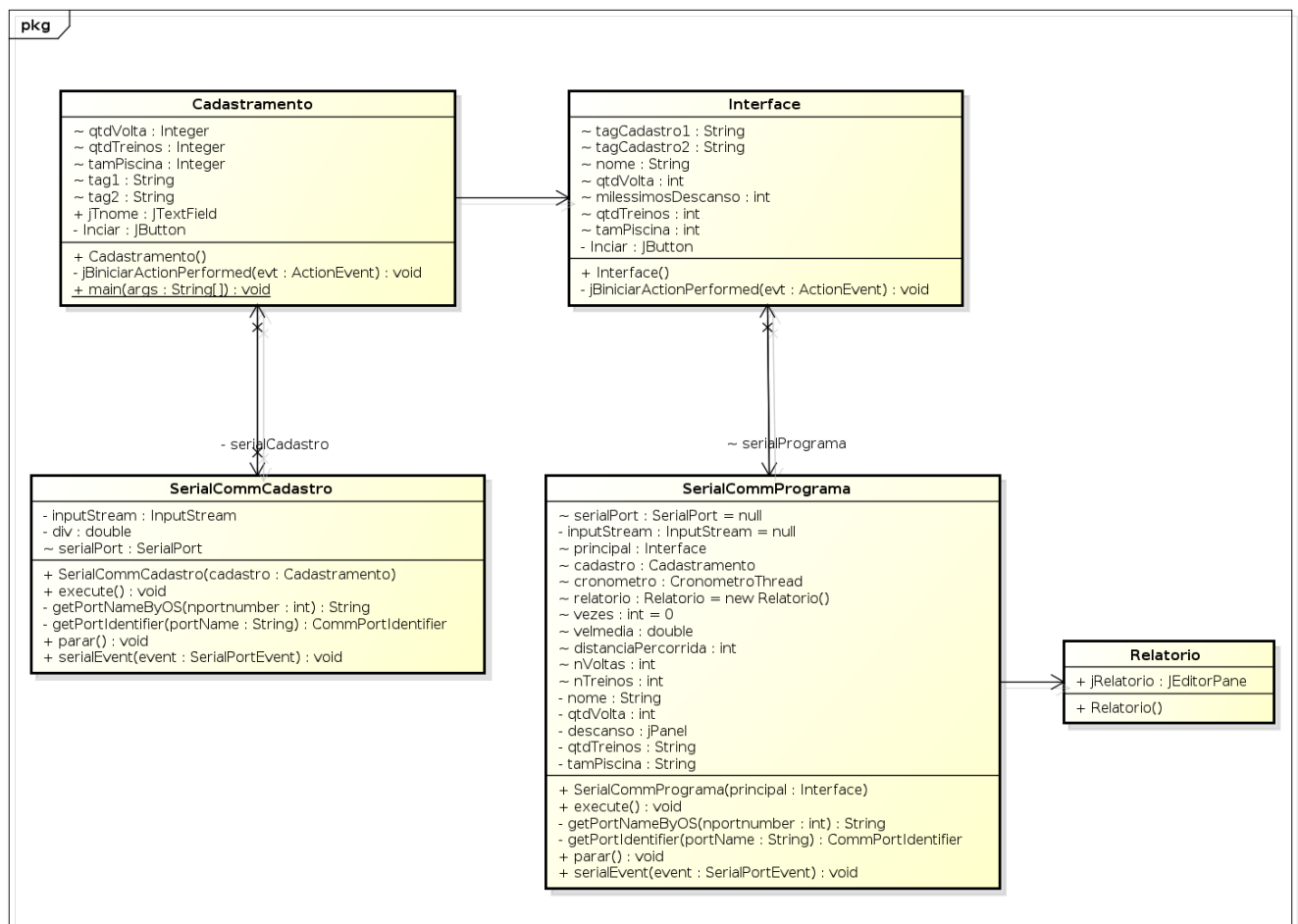


Figura 4.1: Representação do diagrama de classes do *software* desenvolvido.

4.3 Implementação e Codificação

A parte que demanda mais esforço do desenvolvimento do projeto é a implementação do código fonte. É neste estágio que o código executável do *software* é criado. O código deve ser abordado de maneira adequada para os requisitos do Sistema de monitoramento de atividades de natação utilizando RFID e a placa Raspberry Pi.

Assim, o *software* terá como requisitos o recebimento do ID de TAG, registro de tempo das voltas baseadas na leitura do ID, calcular o tempo relativo entre as leituras sucessivas, calcular a quantidade de voltas dadas na piscina, calcular o tempo de cada volta, calcular a velocidade média de cada volta, calcular a distância total percorrida, calcular o tempo total do treino e calcular a velocidade média durante o treino. Portanto, os valores obtidos destes parâmetros formam um monitoramento adequado em um treino de natação.

Para entender como são obtidos os valores, esta seção descreve o desenvolvimento do *software* em etapas. A primeira etapa do *software* é o cadastramento do plano de treino que o nadador irá realizar, conforme ilustra a Figura 4.2. O nadador adiciona nome, quantidade de voltas do treino atual, quantidade de repetições do treino atual, tempo de descanso, tamanho da piscina e os ID's das TAG's vinculadas ao nadador. Estas informações definem as características do treino do nadador, ou seja, quantas voltas serão dadas na piscina até o período de descanso, define o período que o nadador irá descansar após uma sequência de voltas, e quantas vezes isso vai se repetir. O nadador precisa cadastrar os ID's das TAG's que ele está portando durante o treino. Assim, ao clicar no botão Coletar ele passa cada TAG na frente do leitor para que esse identifique cada ID. A classe Cadastramento é a encarregada de receber os dados, utilizando uma interface gráfica, onde receberá as entradas destes dados e guardados em atributos.

Os atributos que recebem os ID's das TAG's são gerados pela SerialCommCadastro. Esta classe utiliza a API de comunicação serial (SerialComm), contendo a lógica para receber o ID da TAG enviada ao leitor via comunicação serial. Esta classe caracteriza-se por estar em espera de eventos gerados pela interface serial. Quando o leitor de RFID envia o ID da TAG para o Raspberry Pi, este disponibiliza para a SerialComm, gerando um evento na classe, assim recebendo o valor de ID da TAG e o guardando em um atributo. Este atributo é repassado à classe Cadastramento, que o guarda no atributo local da classe, vinculando os ID's das TAG's ao nadador.

Tendo preenchido os campos do cadastro, inicia-se a segunda etapa do *software* pressionando o botão iniciar. A segunda etapa do software concentra-se o principal objetivo do projeto, contabilizar e extrair tempo e quantidade de voltas em um treino. Ou seja, o monitoramento da

Monitoramento de Natacao

MAN - Cadastramento

Nadador:

Quantidade de Voltas:

Tempo de Descanso: Segundos

Quantidade de Repetições:

Tamanho da Piscina: Metros

TAG 1:

TAG 2:

Figura 4.2: Classe Cadastramento do plano de treino do nadador.

atividade de natação. Para isso, são utilizadas as classes Interface e SerialCommPrograma.

A Interface é onde se concentra a disponibilidade dos valores obtidos da volta, em uma interface gráfica, representada na Figura 4.3. Ao pressionar o botão iniciar, o *software* aguarda a etiqueta com ID entrar na área de cobertura do leitor de RFID para que ela transmita seu ID. Quando a TAG entra nesta área de cobertura, o leitor recebe o ID, o *middleware* se encarrega de disponibilizá-lo na interface serial. Assim, o Raspberry Pi, fica monitorando sua porta serial para ver se o *middleware* postou algum dado nesta interface. E é a classe SerialCommPrograma que fica encarregada de receber o valor de ID da TAG.

O treino inicia quando o nadador clicar no botão Iniciar e o ID de uma de suas TAG's for lida, marcando o horário de início da volta. O próximo recebimento do ID é caracterizado pelo final da primeira volta e início da segunda, registrando o horário atual em um atributo. Com a diferença entre os valores dos atributos do horário atual e o horário inicial, é obtido o tempo da volta, que também é guardado em um atributo. Depois de registrado o tempo da volta, o atributo de horário inicial recebe o valor do atributo de horário final, ou seja, registra o horário inicial da segunda volta. Para as próximas voltas, seguem neste mesmo fluxo: recebe o ID, registra o valor do horário atual, faz a diferença entre horário atual e horário inicial, registrando o tempo da volta e o valor do horário atual é novamente atualizado pelo horário inicial. E segue repetindo este cálculo para cada volta dada até o final do treino. Pois, o tempo de cada volta é determinado entre o horário relativo do início da volta e do seu final. Para obter o valor de quantidade de voltas é simples, o atributo que representa a quantidade de voltas é somado ao valor um em cada final de volta.

Com o atributo de quantidade voltas sendo incrementado a cada volta, chegando ao valor

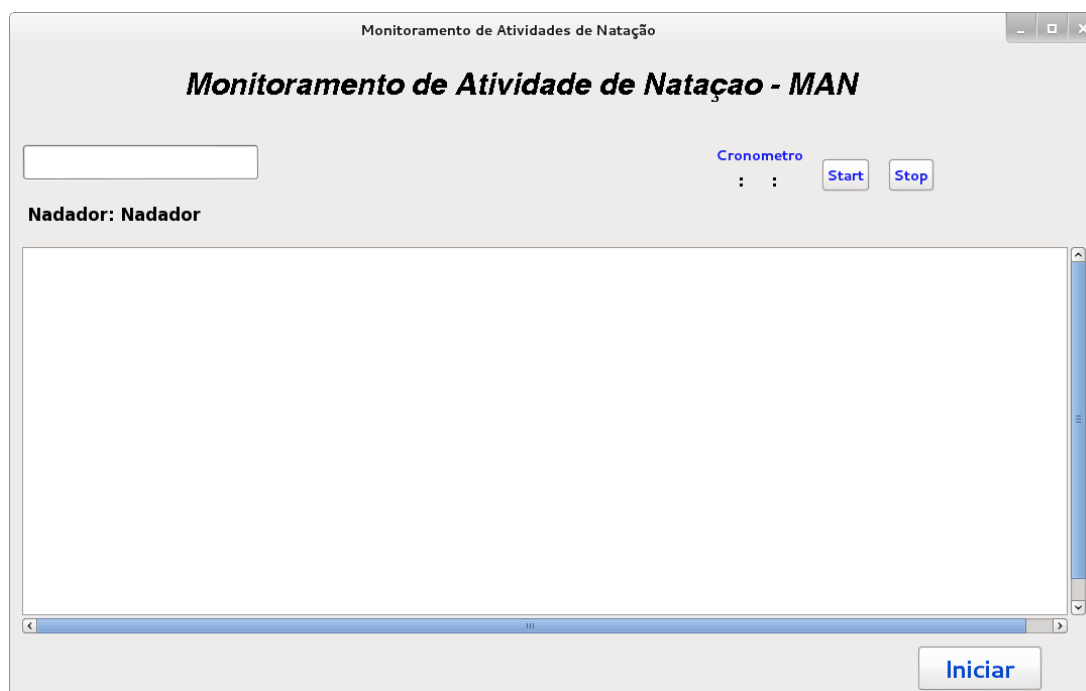


Figura 4.3: Classe Interface que disponibiliza os valores obtidos da volta.

da quantidade de voltas cadastradas, inicia-se o descanso. O descanso é o período de tempo necessário para o nadador se recuperar dos desgastes das voltas feitas na repetição. Este período é importante ser respeitado, pois este processo ajuda no desempenho do atleta. Portanto, o tempo é pré-definido no cadastro e o nadador ficará a espera deste tempo acabar. Após a finalização deste tempo pré-determinado o nadador inicia a próxima repetição ao registrar o seu ID para abrir uma nova volta. No *software* o descanso é simples, apenas um *sleep*² da execução do *software* com valor de tempo cadastrado no início.

Com o término do tempo de descanso é retornado a repetição do treino, seguindo o mesmo fluxo de contabilização de voltas do *software*. Outro atributo contabiliza a quantidade de vezes em que ocorre o descanso, este atributo ao chegar ao valor da quantidade de repetições do treino cadastrada, chega-se ao fim de treino.

Ao chegar ao fim de treino, inicia o último estágio do software, o relatório do treino, que utiliza a classe Relatório. Esta classe é uma interface gráfica em que apresenta todos os valores obtidos no treino, o número e o tempo de cada volta, além de disponibilizar a velocidade média em cada volta e a distância acumulada a cada volta, que foram calculadas com o valor do tamanho da piscina cadastrada. Nos testes este relatório serve para conferir se as informações registradas no treino estão de acordo com o que foi realizado pelo nadador. Não estava entre os objetivos de este trabalho exportar o relatório de treino em um arquivo, por exemplo, em um

²Na programação *sleep* é o ato de deixar o programa parado, ou seja, dormindo.

arquivo de texto.

É possível também ao nadador, observar os valores obtidos em cada volta em tempo real pela interface gráfica da classe Interface. Sendo assim o *software* disponibiliza o valor obtido de cada volta ao longo do treino e gera um relatório final de todo o treino.

Para entender melhor o processo de coleta das informações em um treino utilizando o sistema de monitoramento de atividades de natação proposto, a Figura 4.4 descreve o processo em forma de diagrama. O sistema inicia-se no processo de aguardar a etiqueta entrar na área de cobertura do leitor. Se caso não houver etiqueta na área de cobertura do leitor, continuará o processo de aguardo. Caso a etiqueta entre na área de cobertura do leitor, a TAG transmite seu ID para o leitor que o entrega para o Raspberry Pi via o *middleware*. O Raspberry Pi ao receber o ID em sua porta serial notifica a aplicação, esta analisa e verifica se o ID lido é o mesmo do nadador. Se for início de treino, registra a hora inicial e retorna ao início do processo de aguardar a etiqueta na área de cobertura do leitor. Se não for início de treino, registra a hora final para calcular o tempo da volta e contabiliza uma nova volta. Se o número registrado da volta for igual a quantidade de voltas registrada no cadastro, significa que o nadador entrou no período de descanso. Caso não for igual, volta ao processo de aguardar a etiqueta na área de cobertura do leitor, iniciando a próxima volta. Um atributo contabiliza a quantidade de vezes que o *software* entra no processo de descanso. Com este número sendo igual ao número de repetições registrado no cadastro, caracteriza-se pelo final de treino e o *software* disponibiliza um relatório da análise total do treino em uma interface gráfica dos valores: quantidade de voltas dadas na piscina, tempo de cada volta, velocidade média de cada volta, distância total percorrida, tempo total do treino e a velocidade média durante o treino.

Para analisar cada etapa do *software* foram feitos dois tipos de testes: teste funcional, servindo como principal teste da funcionalidade do código fonte do *software* e o teste de campo, realizado em uma piscina com nadadores, para testar a funcionalidade de todo o sistema em geral, como será apresentado no próximo capítulo.

5 *Testes do sistema de monitoramento de natação proposto*

Ao longo deste projeto foram descritas as necessidades e os elementos que são utilizados para alcançar o objetivo do sistema. A utilização de um sistema de RFID com a placa Raspberry Pi e um *software* desenvolvido em Java para obter um sistema de monitoramento de atividades de natação.

Neste capítulo serão descritos os resultados do sistema como um todo. Com o resultado obtido, é possível identificar as limitações do *software* e dos equipamentos envolvidos no sistema. Neste projeto é criado um protótipo do sistema de monitoramento de atividades de natação, mostrando sua viabilidade técnica. O desenvolvimento do *software* proporcionou o mínimo de *design* visual para a interação de seu uso, como visto no *layout* do Cadastramento e Interface do *software*.

Para validar o desempenho do *software* desenvolvido, na placa Raspberry Pi, é necessário observar a relação entre o desempenho do *software* desenvolvido e os requisitos presentes na placa. Verificar também se o código é bem estruturado e consegue utilizar poucos recursos presente na placa e, assim, definir se a placa possui ou não os requisitos necessários para este sistema.

Para verificar o fator desempenho e validar a sua usabilidade, os testes funcionais do *software* foram realizados em um computador com alto desempenho de processamento, até a validação dos resultados obtidos. Após validação, os testes passaram a ser na placa Raspberry Pi, que serão descritos nas próximas seções.

5.1 Testes Funcionais

Os testes funcionais servem para avaliar o comportamento do *software*. Realizar os testes funcionais é incluir entradas de dados, observar o processamento e obter uma resposta. Com

a resposta pode ser analisado e modificado até ao ponto do *software* atender aos requisitos: quantidade e tempo de volta em um monitoramento de atividades de natação. Portanto, os testes realizados neste projeto visam alcançar a meta traçada no projeto, ou seja, em obter os dados de tempo de volta, quantidade de volta, velocidade média, distância percorrida, tempo médio e quantidade de repetições no treino de um nadador.

Os primeiros testes funcionais foram feitos em um computador, fora da placa Raspberry Pi. Este teste serviu para verificar as entradas de dados no cadastramento, erros na interface, comportamentos no geral, erros de iniciação e término deste *software*. Tendo o leitor de RFID ligado ao computador, este *software* pode ser executado com a coleta de identificação da TAG e o processamento de sua lógica.

Esses testes foram realizados ao longo do desenvolvimento do *software*. Iniciando pelas entradas de dados nos campos do Cadastramento: nadador, quantidade de voltas, tempo de descanso em segundos, quantidade de repetições e tamanho da piscina. Estes campos recebendo números inteiros e strings¹. Também foi testada a coleta de identificação das TAG's, servindo para verificar o uso do RFID na aplicação. É repetido este teste até a execução não apresentar erros seguindo para a Interface.

A Interface é a classe onde apresenta os valores obtidos no treino do nadador. Para testar, foi simulado juntamente com o auxílio de um cronômetro, para comparar o tempo obtido das voltas do cronômetro manual com o tempo obtido do *software*. Ao decorrer dos testes foram modificados códigos do *software* até chegarem aos valores equivalentes aos obtidos pelo cronômetro. O código inicial possuía um cronômetro desenvolvido ao *software* para contabilizar o tempo de volta. Porém, isto levou a um alto uso de processamento devido estar constantemente contabilizando o tempo. Surgindo a modificação em utilizar a marcação de horário inicial e horário atual, e a diferença entre esses valores atribui o tempo total da volta. Com isso, concede ao cálculo exato de tempo e uma utilização baixa de processamento.

Com a correta contabilização de tempo da volta, proporcionou ao correto cálculo da velocidade média. Validando as amostragens feitas pelo *software*: tempo de volta, quantidade de volta, velocidade média, distância percorrida, tempo médio e quantidade de repetições. Após esta validação o *software* passa a ser testado na placa Raspberry Pi.

¹ Sequência ordenada de caracteres (símbolos).

5.2 Testes Funcionais na placa Raspberry Pi

Com o funcionamento da coleta de identificação das TAG's pelo RFID, os valores obtidos do treino e a disposição dos valores no ambiente gráfico adequadamente em funcionamento. Seguiu-se para o próximo estágio de teste, o funcionamento do *software* desenvolvido na placa Raspberry Pi.

Com a adequação do sistema operacional, instalação do pacote *Java Runtime Environment* necessário para executar aplicações Java, na placa. O *software* desenvolvido foi copiado e executado na placa Raspberry Pi. Em análise foi verificado o uso entre 50 e 60 por cento no processador, por parte do *software*. Também utilizado entre 90 MB e picos de uso em 125 MB dos 256 MB disponíveis de memória RAM deste modelo de Raspberry Pi. Ou seja, o *software* está pronto para o próximo estágio de teste, o teste de campo, no ambiente de natação.

5.3 Teste de Campo

A Associação dos Empregados da Eletrosul (ELASE) disponibilizou o Parque Aquático Wálmore Pereira Siqueira para o teste do protótipo do Sistema de Monitoramento de Atividades de Natação (MAN). A piscina utilizada no parque possui 25 metros de comprimento e largura de 12,5 metros, contendo seis raias. Como cada raia possui 2,083 metros dificultou o uso de um leitor de RFID, utilizando dois leitores expostos na borda da piscina.

Um fator importante na verificação da usabilidade é a disposição física do *hardware*, leitor de RFID, no ambiente de natação. O leitor faz a leitura da etiqueta no início do treino, na virada de volta e ao fim do treino. Nesses três momentos de leitura o *hardware* deve apresentar uma área de cobertura garantindo a recepção do ID da etiqueta. Caso uma identificação não seja capturada pelo leitor de RFID, a inicialização ou finalização da volta não é anotada. Assim não há marcação correta do número e nem do tempo da volta. Com isto ocorrendo não será uma falha de *software*, mas pode ser uma falha do equipamento não adequado a este ambiente e aplicação.

O leitor RFID utilizado nos testes foi desenvolvido para aplicações de controle de acesso de pessoas, onde as suas etiquetas são crachás. Este leitor opera na frequência de 125 KHz e possui um alcance frontal máximo de 55 cm, na lateral seu alcance reduz para menos da metade do frontal. Mas como não foi encontrado no mercado um equipamento de RFID que satisfizesse plenamente a aplicação de contar voltas de um nadador na piscina, este foi o equipamento encontrado que melhor satisfez as exigências para esta aplicação até a presente data. Logo o

fato de cada raia possui um pouco mais de 2 m de largura, não é possível cobrir toda a largura da raia com este leitor, desta forma foi utilizado dois leitores para tentar minimizar as zonas de sombras na borda da raia. A disposição dos equipamentos na piscina pode ser vista na Figura 5.1.

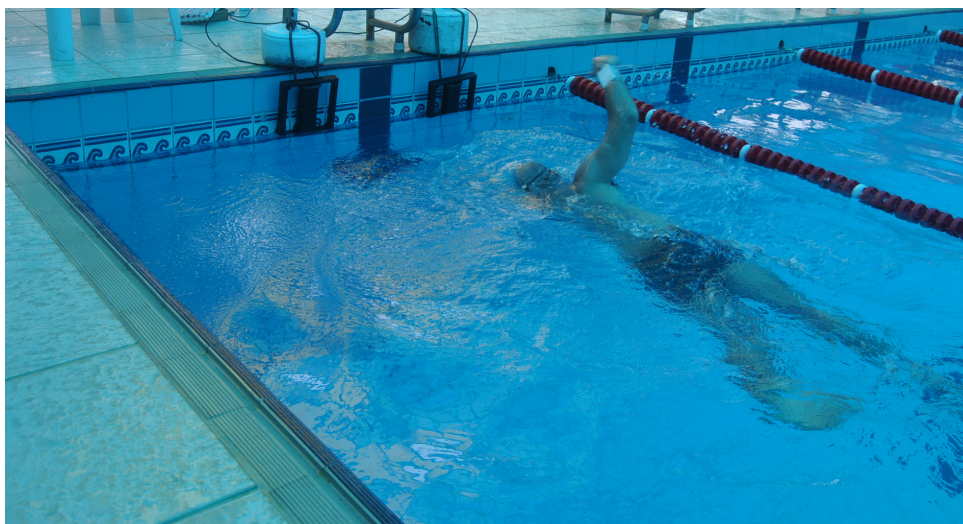


Figura 5.1: Disposição dos leitores RFID na borda da piscina.

Para este teste foram convidados dois nadadores, chamados de Nadador 1 e Nadador 2, respectivamente. Cada nadador nadou usando o protótipo proposto neste projeto por aproximadamente 25 min. Utilizando duas etiquetas cada um, uma posicionada no pulso e outra no tornozelo.

O treino do Nadador 1 foi configurado da seguinte forma:

- Nome: Nadador 1
- Quantidade de voltas: 2
- Tempo de descanso: 30 s
- Número de repetições: 4
- Tamanho da piscina: 25 m
- TAG 1: 2200D879C7
- TAG 2: 2200D879C8

O treino do Nadador 2 foi configurado da seguinte forma:

- Nome: Nadador 2

- Quantidade de voltas: 4
- Tempo de descanso: 60 s
- Número de repetições: 2
- Tamanho da piscina: 25 m
- TAG 1: 2200D879C7
- TAG 2: 2200D879C8

O Nadador 1 realizou um treino com quatro repetições, contendo duas voltas em cada repetição. O Nadador 2 realiza dois treinos com duas repetições, contendo quatro voltas em cada repetição. A fim de obter uma conclusão, a Tabela 5.1 reporta os valores de tempo de cada volta e o tempo médio de cada repetição em segundos e milésimos. Também apresenta de forma comparativa a de um cronômetro manuseado por uma pessoa. Como não há outra forma confiável para comparação, a única forma de saber se os testes estão convergindo é comparar com o cronômetro, que é acessível ao teste. Contudo, desta forma não se pode obter a conclusão de qual valor de tempo é exato. Porém obtendo a diferença de tempo entre o sistema e o cronômetro, pode-se ter uma análise de usabilidade.

O Nadador 1 obteve uma média geral do tempo em 57 segundos e 39 milésimos, calculada no sistema. Pelo cronômetro manuseado por uma pessoa obteve a média geral do tempo em 57 segundos e 31 milésimos. No primeiro treino o Nadador 2, obteve uma média geral do tempo em 61 segundos e 41 milésimos, calculada no sistema. Sendo que pelo cronômetro obteve a média geral do tempo em 61 segundos e 34 milésimos. Em seu segundo treino e último do teste, o Nadador 2 obteve uma média geral no tempo do treino em 64 segundos e 77 milésimos, calculada no sistema. Obtidas pelo cronômetro obteve a média geral do tempo em 64 segundos e 71 milésimos em seu treino. O valor de desvio no tempo médio do Nadador 1 na primeira repetição foi de 6 milésimos. Na segunda repetição obteve o valor de 9 milésimos, sendo na terceira de 6 milésimos e na quarta e última de 7 milésimos. Obtendo a média de 7 milésimos de desvio em todo o treino. O Nadador 2 obteve os valores de desvio no tempo médio em seu primeiro treino de 7 milésimos nas duas repetições. Já em seu segundo treino obteve 6 milésimos na primeira repetição e na segunda 5 milésimos. Obtendo a média de 7 milésimos do primeiro treino e 5,5 milésimos do segundo treino. Em consideração ao tempo de perda que ocorre no manuseio do cronômetro, a obtenção dos valores tanto do sistema e do cronômetro são valores bem próximos.

Com isso pode-se concluir que neste teste o cronômetro não ocasionou uma perda significativa em um monitoramento de natação. Porém é necessário outra pessoa estar manuseando o equipamento. Já o sistema proposto, obteve os valores utilizando RFID. Que proporciona uma forma automatizada em coletar os tempos das voltas. Com quanto que o leitor esteja com sua área de cobertura disponível para coleta de etiqueta tanto submersa e fora da água. Portanto, este sistema mostra-se funcional e serão discutidas suas possíveis limitações no próximo capítulo.

| Nadador 1 | Sistema | Cronômetro | Desvio dos valores |
|------------------------|---------|------------|--------------------|
| Repetição 1 | | | |
| Volta 1 | 46:45 | 46:39 | 00:06 |
| Volta 2 | 55:49 | 55:43 | 00:06 |
| Tempo Médio | 50:97 | 50:91 | 00:06 |
| Repetição 2 | | | |
| Volta 1 | 53:36 | 53:29 | 00:10 |
| Volta 2 | 55:11 | 55 | 00:11 |
| Tempo Médio | 54:23 | 54:14 | 00:09 |
| Repetição 3 | | | |
| Volta 1 | 62:29 | 62:23 | 00:06 |
| Volta 2 | 61:14 | 61:07 | 00:07 |
| Tempo Médio | 61:71 | 61:65 | 00:06 |
| Repetição 4 | | | |
| Volta 1 | 63:92 | 63:86 | 00:06 |
| Volta 2 | 61:35 | 61:27 | 00:08 |
| Tempo Médio | 62:63 | 62:56 | 00:07 |
| Nadador 2 | Sistema | Cronômetro | Desvio dos valores |
| <i>Primeiro Treino</i> | | | |
| Repetição 1 | | | |
| Volta 1 | 59:64 | 59:57 | 00:07 |
| Volta 2 | 61:04 | 60:96 | 00:08 |
| Volta 3 | 62:63 | 62:57 | 00:06 |
| Volta 4 | 61:09 | 61:05 | 00:04 |
| Tempo Médio | 61:10 | 61:03 | 00:07 |
| Repetição 2 | | | |
| Volta 1 | 61:29 | 61:22 | 00:07 |
| Volta 2 | 61:74 | 61:68 | 00:06 |
| Volta 3 | 61:97 | 61:91 | 00:06 |
| Volta 4 | 61:84 | 61:77 | 00:07 |
| Tempo Médio | 61:71 | 61:64 | 00:07 |
| <i>Segundo Treino</i> | | | |
| Nadador 2 | Sistema | Cronômetro | Desvio dos valores |
| Repetição 1 | | | |
| Volta 1 | 63:39 | 63:34 | 00:05 |
| Volta 2 | 67:56 | 67:50 | 00:06 |
| Volta 3 | 66:38 | 66:32 | 00:06 |
| Volta 4 | 65:18 | 65:11 | 00:07 |
| Tempo Médio | 65:63 | 65:57 | 00:06 |
| Repetição 2 | | | |
| Volta 1 | 62:32 | 62:28 | 00:04 |
| Volta 2 | 64:49 | 64:45 | 00:04 |
| Volta 3 | 66:30 | 66:26 | 00:04 |
| Volta 4 | 62:49 | 62:42 | 00:07 |
| Tempo Médio | 63:90 | 63:85 | 00:05 |

Tabela 5.1: Tabela dos valores obtidos do treino de dois nadadores.

6 *Limitações*

O *software* desenvolvido é limitado ao treino cadastrado. Se houver alguma interrupção no atual treino, o software não compensa o tempo decorrido pela interrupção. Caso o nadador queira parar algum momento entre cada volta, isto irá prejudicar no tempo coletado e assim ocasionando um erro na medição de tempo em sua volta.

Um outro fator limitante deste projeto é o *hardware*, leitor de RFID, que não é o adequado para esta aplicação, visto que sua área de cobertura é muito limitada. Isto significa que pode não ocorrer a leitura no momento exato, ou seja, no início de treino, na virada de volta ou ao fim do treino prejudicando o resultado do relatório final.

Outro fator limitante é a necessidade de trabalhar com múltiplos nadadores, que este sistema não provê tal utilização. Assim, pensando em um produto, deve ser levado em conta a criação de um novo projeto submetendo a mudanças que visam monitorar diversos atletas, além de utilizar um leitor que apresente uma área de coleta de etiqueta adequada ao ambiente de natação.

A fim de continuidade ao sistema proposto, algumas tópicos serão sugeridos no próximo capítulo, que possibilitará a construção de um produto comercialmente viável do Sistema de Monitoramento de Atividades de Natação (MAN).

7 *Conclusões*

O sistema de monitoramento de atividades de natação proposto mostra ser uma forma eficiente de monitoramento de treinos de natação. O próprio *software* que obtém os valores de tempo de cada volta e quantidade de volta não necessitando estar uma pessoa presente para anotar os valores. Com a utilização de um sistema de RFID para identificar o nadador, proporciona de forma automatizada o cálculo de tempo de cada volta, além de obter uma análise completa com os valores de velocidade média, quantidade de voltas, tempo médio e distância percorrida no treino.

Levando-se em consideração os resultados obtidos no teste de campo, conclui-se que é possível obter o tempo de cada volta feita por um nadador com o sistema proposto neste projeto, impactando de forma positiva a utilização da placa Raspberry Pi e RFID no Sistema de Monitoramento de Atividades de Natação (MAN).

Atualmente, na maioria das academias e clubes não existe nenhum sistema para realizar um monitoramento de seus atletas, utilizando apenas maneiras rudimentares. Como a utilização de cronômetro manuseado pelo próprio atleta ou por outra pessoa. Este sistema proposto, utilizando RFID, mostra trazer benefícios para o monitoramento de atividades de natação em academias e clubes.

O desenvolvimento deste *software*, utilizando um sistema de RFID e a placa Raspberry Pi, demonstrou ser utilizável para monitorar um nadador. Em apresentação deste protótipo para técnicos e nadadores, todos apreciaram e gostaram da solução desenvolvida. E também com a atual necessidade da natação em ter tecnologias auxiliares aos treinos, concluindo assim, que possui um grande espaço mercadológico e atual.

8 *Trabalhos Futuros*

Este projeto criou um sistema utilizando a placa Raspberry Pi e RFID no Monitoramento de Atividades de Natação (MAN). Com base nos resultados obtidos, é necessário a sua continuação. Os itens abaixo indicam tópicos de propostas para futuros trabalhos:

1. Modificar o *software* deste projeto ou criar um *software* com conexão remota a um banco de dados, possibilitando o uso de treinos pré-cadastrados;
2. Projetar um equipamento RFID com uma antena específica para o uso no ambiente de natação;
3. Criar um *software* que faça o Monitoramento de Atividades de Natação com RFID, baseado na internet, para acompanhar o desempenho do atleta no treino e ao longo dos treinamentos realizados, em tempo real;
4. Projetar um *software* específico para sistema embarcado com a finalidade de Monitoramento de Atividades de Natação com RFID.

Através destes tópicos sugeridos é possível chegar na criação de um produto comercial e estável do Sistema de Monitoramento de Atividades de Natação (MAN).

Referências Bibliográficas

- BALADEI, R. R. de Freitas Dias e S. D. P. Diferenças entre as frequências do sistema rfid passivo. In: *RFID Journal Brasil*. [s.n.], 2013. Disponível em: <<http://brasil.rfidjournal.com/artigos>>.
- CAELUM. *Apostila Caelum FJ-11 Java e Orientações a Objeto*. 2004. Disponível em: <<http://www.caelum.com.br/apostila-java-orientacao-objetos/>>.
- CBDA. 2013. Disponível em: <<http://www.cbda.org.br>>.
- COLORADO. *Aquatic Products*. 2013. Disponível em: <<http://www.coloradotime.com/products/>>.
- DACOSTA, L. P. *Cenário de tendências gerais dos esportes e atividades físicas no Brasil*. 2006. Disponível em: <<http://cev.org.br/biblioteca/cenario-tendencias-gerais-dos-esportes-atividades-fisicas-brasil/>>.
- DAKTRONICS. *Scoreboards Timing Systems - Aquatics*. 2013. Disponível em: <<http://www.daktronics.com>>.
- DONIAK, M. *Patente de Monitoramento de atividade de natação com RFID*. 2009. Disponível em: <<http://www.inpi.gov.br>>.
- FINA. *FEDERATION INTERNATIONALE DE NATATION*. 2013. Disponível em: <<http://www.fina.org/>>.
- FOUNDATION, R. P. *Raspberry Pi Wiki*. 2012. Disponível em: <<http://elinux.org/RaspberryPiBoard>>.
- FOUNDATION, R. P. *Rasperry Pi*. 2013. Disponível em: <<http://www.raspberrypi.org>>.
- GPP. *Demonstração do crescimento dos esportes na preferência do brasileiro*. 2009. Disponível em: <www.gpp.com.br>.
- GREFF, P. de A. *Especificação de um Sistema para Monitoramento de Atividades de Natação usando RFID*. Outubro 2009. Disponível em: <http://wiki.sj.ifsc.edu.br/wiki/images/0/06/ProjetoFinal_Ponciano.pdf>.
- HILL, M. (Ed.). *Engenharia de Software*. [S.l.]: Roger S. Pressman, 2011.
- IBM. Controlando o rfid. In: . [s.n.], 2013. Disponível em: <<http://www.ibm.com/br/ibm>>.
- MINISTERIO DO ESPORTE. *Programa Bolsa Atleta*. [S.l.]: LEI Nº 12.395, 2012.
- MOURA, R. de. *Planejamento Estratégico da Natação 2010/2016*. 2010. Disponível em: <www.cbda-pbn.org.br>.

MURDOCK, I. *Debian GNU/Linux*. 1993.

OLYMPIC, O. W. to M. 2013. Disponível em: <<http://www.olympic.org/>>.

OMEGA. *Ares 21*. 2009. Disponível em:
<<http://www.htsgroup.co.nz/Pdf/Omega/ARES21.pdf>>.

OMEGA-WATCHES. *Swimming*. 2013. Disponível em: <<http://www.omegatiming.com/>>.

SEIKO-WATCH. 2007–2011. Disponível em: <<http://www.seiko.com.br/>>.

SESI-SP. *Biomecânica a serviço da equipe de natação do Sesi-SP*. 2013. Disponível em:
<<http://www.sesisp.org.br/noticias/biomecanica-a-servico-da-equipe-de-natacao-do-sesi-sp>>.

SWIMITUP! Disponível em: <<http://www.regrasdenatacao.com.br/>>.

VIZINEX-RFID. *Medical Uses for RFID Products*. 2013. Disponível em:
<<http://www.vizinexrfid.com/medical-uses-for-rfid-products/>>.