

UNIVERSIDADE REGIONAL DE BLUMENAU
CENTRO DE CIÊNCIAS EXATAS E NATURAIS
CURSO DE CIÊNCIA DA COMPUTAÇÃO – BACHARELADO

**PROTÓTIPO DE SISTEMA PARA MONITORAMENTO DA
FREQUÊNCIA CARDÍACA DE PACIENTES PORTADORES
DE MIOCARDIOPATIA DILATADA**

ALEXANDER THOMAS GONÇALVES FURTADO

BLUMENAU
2017

ALEXANDER THOMAS GONÇALVES FURTADO

**PROTÓTIPO DE SISTEMA PARA MONITORAMENTO DA
FREQUÊNCIA CARDÍACA DE PACIENTES PORTADORES
DE MIOCARDIOPATIA DILATADA**

Trabalho de Conclusão de Curso apresentado ao curso de graduação em Ciência da Computação do Centro de Ciências Exatas e Naturais da Universidade Regional de Blumenau como requisito parcial para a obtenção do grau de Bacharel em Ciência da Computação.

Prof. Mauro Marcelo Mattos, Doutor - Orientador

**BLUMENAU
2017**

**PROTÓTIPO DE SISTEMA PARA MONITORAMENTO DA
FREQUÊNCIA CARDÍACA DE PACIENTES PORTADORES
DE MIOCARDIOPATIA DILATADA**

Por

ALEXANDER THOMAS GONÇALVES FURTADO

Trabalho de Conclusão de Curso aprovado
para obtenção dos créditos na disciplina de
Trabalho de Conclusão de Curso II pela banca
examinadora formada por:

Presidente: _____
Prof. Mauro Marcelo Mattos, Doutor – Orientador, FURB

Membro: _____
Prof. Miguel Alexandre Wisintainer, Mestre – FURB

Membro: _____
Prof. Francisco Adell Péricas, Mestre – FURB

Blumenau, 04 de Julho de 2017

Dedico este trabalho a minha família que sempre me deu forças e sempre esteve ao meu lado.

AGRADECIMENTOS

À minha família por ter me apoiado e estado presente a todo momento.

Aos meus amigos que incentivaram no desenvolvimento desse trabalho.

Ao Dr. Edson Ramonn Abreu Freitas, cardiologista, por ter contribuído para o desenvolvimento desse trabalho.

Ao meu orientador Mauro Marcelo Mattos pelo estímulo, confiança e contribuições a esse trabalho.

As grandes ideias surgem da observação dos
pequenos detalhes.

Augusto Cury

RESUMO

Este trabalho descreve o desenvolvimento de um protótipo de um aplicativo para o monitoramento da frequência cardíaca de pacientes portadores de insuficiência cardíaca. Para efetuar a coleta da frequência cardíaca foi utilizado o relógio inteligente (*smart watch*), Samsung Gear S2 Classic, como tecnologia vestível e foi desenvolvida uma aplicação móvel para controlar as medições do relógio e armazená-las. Para a aplicação *wearable*, desenvolvida na plataforma Tizen, foi utilizada a linguagem JavaScript, juntamente com HTML e CSS ao invés das linguagens nativas da plataforma, C e C++. A aplicação móvel foi desenvolvida para plataforma Android através da linguagem Java. Para a comunicação entre os dispositivos vestível e móvel foi utilizada a biblioteca Accessory SDK, da fabricante Samsung. Os resultados obtidos demonstram que o sistema cumpre o propósito para o qual foi desenvolvido de registrar o histórico de frequências cardíacas, de forma autônoma ou por iniciativa do usuário, embora o Tizen se apresente limitado em relação a outras alternativas.

Palavras-chave: Vestível. Frequência cardíaca. Tizen. Android.

ABSTRACT

This paper describes the development of a prototype of an application to monitor the heart rate of patients with heart failure. In order to collect the heart rate in the smart watch we used the Samsung watch S2 Classic as wearable technology and a mobile application was developed to control the measurements of the watch and to store them. The wearable application, developed on the Tizen platform, it was used in JavaScript along with HTML and CSS rather than the native platform, C and C ++ languages. The mobile application was developed for Android platform through the Java language. For communication between wearable and mobile devices was done with a library, Accessory SDK, by fabricator Samsung. The results obtained demonstrate that the system fulfills the purpose for which the heart rate historic was developed, either autonomously or at the user's initiative, although Tizen is limited in relation to other alternatives.

Key-words: Wearable. Heart rate. Tizen. Android.

LISTA DE FIGURAS

Figura 1 - À esquerda, um coração normal e, à direita, com MCD.....	15
Figura 2 - Freqüencímetro	17
Figura 3 - Sensor HR	18
Figura 4 - Samsung Gear S2 Classic	19
Figura 5 - Disposição dos componentes do Gear S2 Classic	20
Figura 6 - Aplicação Heart rate	21
Figura 7 - Sensor de frequência cardíaca.....	21
Figura 8 - Visão geral do modelo do FitData	22
Figura 9 - Diagrama de classes do GEOCONFI.....	24
Figura 10 - Tela de consulta do Heart-Control.....	25
Figura 11 - Arquitetura do sistema proposto por Sampaio (2011).....	26
Figura 19 - Diagrama de distribuição do sistema	28
Figura 12 - Diagrama de caso de uso da aplicação wearable	29
Figura 13 - Diagrama de caso de uso da aplicação móvel.....	30
Figura 14 - Diagrama de classes da aplicação wearable	31
Figura 15 - Diagrama de classes da aplicação móvel.....	32
Figura 16 - Diagrama de classes do banco de dados da aplicação móvel	33
Figura 17 - Diagrama de atividades da aplicação wearable	34
Figura 18 - Diagrama de atividades da aplicação móvel.....	35
Figura 20 - Tela inicial da aplicação <i>wearable</i>	45
Figura 21 - Tela da aplicação móvel	46
Figura 22 - Informativo do status do serviço.....	47
Figura 23 - Tela de inicialização do sensor HRM	47
Figura 24 - Tela de aviso de ajuste do relógio.....	48
Figura 25 - Tela de medição	48
Figura 26 - Medição gravada no histórico de leituras	49

LISTA DE QUADROS

Quadro 1 - Função para iniciar o Sensor HRM.....	37
Quadro 2 - Função de retorno do sensor HRM	37
Quadro 3 - Função de parada do sensor HRM	38
Quadro 4 - Perfil do serviço fornecedor	39
Quadro 5 - Função requestSAAgent.....	40
Quadro 6 - Implementação do ouvinte de conexão	41
Quadro 7 - Perfil do serviço consumidor.....	41
Quadro 8 - Classe ServicoConexao	42
Quadro 9 - Utilizando o Framework Samsung Accessory Service para envio de mensagens .	43
Quadro 10 - Implementação do alarme	44
Quadro 11 - Código executado pelo alarme	44
Quadro 12 - Comparativo entre trabalhos correlatos.....	50

LISTA DE ABREVIATURAS E SIGLAS

API – Application Programming Interface

BPM – Batimentos Por Minuto

DAO – Data Access Object

HRM – Heart Rate Monitor

IC – Insuficiência Cardíaca

MCD – Miocardiopatia Dilatada

OMS – Organização Mundial de Saúde

OS – Operational System

RF – Requisito Funcional

RNF – Requisito não Funcional

SAP – Samsung Accessory Protocols

SUMÁRIO

1 INTRODUÇÃO.....	13
1.1 OBJETIVOS.....	14
1.2 ESTRUTURA.....	14
2 FUNDAMENTAÇÃO TEÓRICA	15
2.1 MIOCARDIOPATIA DILATADA.....	15
2.1.1 Sintomas.....	16
2.1.2 Diagnóstico	16
2.2 COLETA DE SINAIS FISIOLÓGICOS.....	16
2.2.1 Frequência cardíaca.....	17
2.2.2 Sensor Heart Rate.....	17
2.3 WEARABLES.....	18
2.4 SAMSUNG GEAR S2 CLASSIC (SM-R732)	19
2.5 TRABALHOS CORRELATOS.....	21
2.5.1 FitData: um sistema para monitoramento de atividade física baseado em dispositivos móveis	21
2.5.2 Um modelo para gerenciamento de histórico de contextos fisiológicos.....	23
2.5.3 Sistema de monitoramento remoto de pacientes implementado em hardware de arquitetura ARM	26
3 DESENVOLVIMENTO	27
3.1 REQUISITOS.....	27
3.2 ESPECIFICAÇÃO	27
3.2.1 Diagrama de distribuição	28
3.2.2 Diagramas de casos de uso.....	28
3.2.3 Diagramas de classes.....	30
3.2.4 Diagramas de atividades	33
3.3 IMPLEMENTAÇÃO	36
3.3.1 Técnicas e ferramentas utilizadas.....	36
3.3.2 Operacionalidade da implementação	45
3.4 ANÁLISE DOS RESULTADOS	49
3.4.1 Comparativo entre os trabalhos correlatos	50
4 CONCLUSÕES.....	51

4.1 EXTENSÕES	52
REFERÊNCIAS	53

1 INTRODUÇÃO

Segundo Organização Mundial de Saúde (OMS), as doenças cardiovasculares, como cardiopatia isquêmica e infartos, continuam sendo a primeira causa de mortes no mundo. Em 2012, 1,7 milhões de pessoas morreram em decorrência desta condição. O dado faz parte de uma lista divulgada pela OMS sobre as principais causas de mortalidade (WHO, 2014, p. 1).

As doenças cardiovasculares, afecções do coração e da circulação, representam a principal causa de mortes no Brasil. No período de 2004 a 2013, foram responsáveis por 3.153.175 óbitos, 29% do total, uma morte a cada 40 segundos em nosso meio. As doenças cardiovasculares causam o dobro de mortes que aquelas devidas a todos os tipos de câncer juntos, 2,3 vezes mais que as todas as causas externas (acidentes e violência), 3 vezes mais que as doenças respiratórias e 6,5 vezes mais que todas as infecções incluindo a AIDS. O alerta, a prevenção e o tratamento adequados podem reverter essa grave situação (SBC, 2015, p. 1).

A Insuficiência Cardíaca (IC), ou Miocardiopatia Dilatada (MCD), tem sido apontada como um importante problema de saúde pública e considerada como uma nova epidemia com elevada mortalidade e morbidade (ALBUQUERQUE, 2014, p. 2). Segundo Valdígem (2014, p. 1), a MCD é uma doença do músculo do coração que impede o bombeamento adequado de sangue para o corpo, ocasionando outras doenças cardiovasculares como arritmias e morte súbita.

Segundo Parameshawar et al. (1992 apud BARRETO et al., 1998, p. 1), “insuficiência cardíaca tem uma prevalência de 1 a 2% na população geral”. Dentro das estatísticas brasileiras, é a maior causa de internação dentre os diagnósticos cardiológicos no Sistema Único de Saúde (SUS) (SUS, 1993 apud BARRETO et al., 1998, p. 1).

No Brasil os pacientes com necessidade de acompanhamento hospitalar sofrem em decorrência ao congestionamento de rede de saúde e aos custos do tratamento. Segundo Thomaz (2016, p. 1), a tecnologia tem estendido as funcionalidades das inovações na área da saúde aos chamados *wearables*, ou seja, tecnologias vestíveis que passam a fazer parte do corpo, como pulseiras, relógios e roupas com mecanismos de detecção inteligente. Uma das principais características que diferencia um dispositivo *wearable* de um computador, ou dispositivo móvel portátil, é a possibilidade deste dispositivo permanecer anexado ao corpo do utilizador durante um longo período de tempo sem que se torne incomodo ou condicionador de movimentos (GODINHO, 2013, p. 2). Para Sumrell (2016, p. 1), os *wearables* têm potencial para revolucionar os custos com a saúde, permitindo que o monitoramento e os cuidados com a saúde possam ser feitos fora de unidades de convencionais.

Diante deste contexto, propõe-se nesse trabalho o desenvolvimento de um protótipo de um sistema para acompanhamento de pacientes diagnosticados com miocardiopatia dilatada

que monitore a frequência cardíaca através de tecnologia vestível, e gere relatórios diários para acompanhamento pelo cardiologista do paciente.

1.1 OBJETIVOS

Esse trabalho objetiva a construção de um protótipo de sistema para monitoramento da frequência cardíaca de pacientes portadores de insuficiência cardíaca.

Os objetivos específicos são:

- a) identificar o sinal de frequência cardíaca utilizando dispositivo vestível;
- b) desenvolver uma aplicação Android que armazene os dados processados da frequência cardíaca do paciente e gere histórico para acompanhamento médico;
- c) desenvolver um conjunto de cenários de testes.

1.2 ESTRUTURA

Este trabalho está dividido em quatro capítulos: introdução, fundamentação teórica, desenvolvimento e conclusão. O primeiro capítulo apresenta a introdução e os objetivos do trabalho. O segundo capítulo detalha a fundamentação teórica sobre a doença miocardiopatia dilatada, coleta de sinais fisiológicos, *wearables*, Samsung Gear S2 Classic e trabalhos correlatos. O terceiro capítulo mostra o desenvolvimento do trabalho, apresentando a especificação, técnicas e ferramentas utilizadas, operacionalidade da aplicação Tizen e aplicação Android e resultados. Por fim, o capítulo quatro apresenta as conclusões e sugestão para extensões do trabalho.

2 FUNDAMENTAÇÃO TEÓRICA

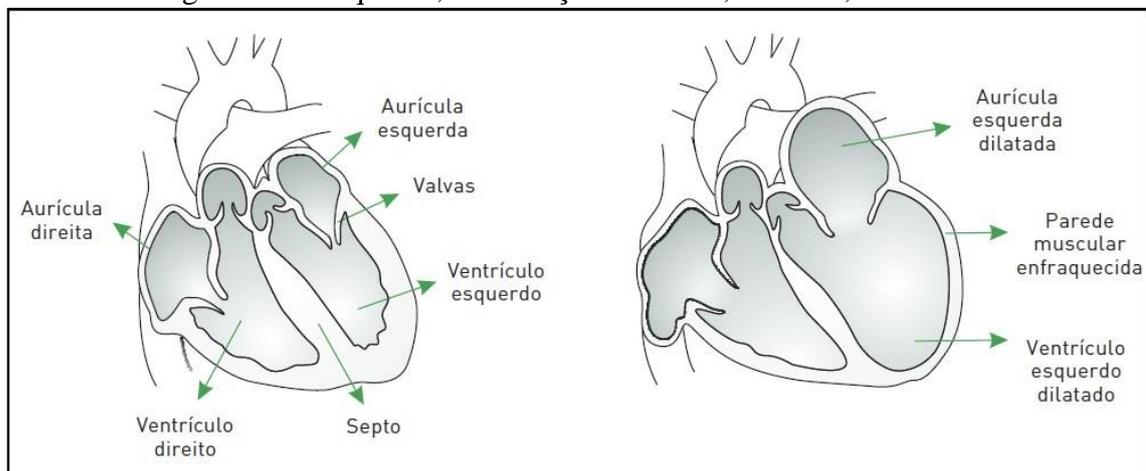
Este capítulo tem como objetivo explorar os principais assuntos necessários para realização deste trabalho. Os assuntos foram subdivididos em quatro partes, onde a seção 2.1 conceitua a doença miocardiopatia dilatada. A seção 2.2, aborda como é feito a leitura dos sinais fisiológicos entre os trabalhos correlatos. Na seção 2.3, é apresentado o conceito de *wearables*. Na seção 2.4 é apresentado características do dispositivo *smartwatch* utilizado nesse trabalho, e, por fim, na seção 2.5 a apresentação dos trabalhos correlatos ao proposto.

2.1 MIOCARDIOPATIA DILATADA

A insuficiência cardíaca é apontada mundialmente como um dos principais problemas de saúde pública. A principal etiologia de IC é a miocardiopatia dilatada, sendo a principal ocasionadora de transplantes cardíacos em adolescentes e adultos jovens (HOROWITZ, 2016, p. 2).

Segundo Valdigem (2014, p. 1), a miocardiopatia dilatada é uma doença do músculo cardíaco que danifica o sistema de bombeamento do sangue para o corpo. A MCD afeta principalmente o ventrículo esquerdo, provocando o dilatamento desse músculo, conforme a Figura 1.

Figura 1 - À esquerda, um coração normal e, à direita, com MCD.



Fonte: Fochesatto Filho e Barros (2013, p. 1).

Na miocardiopatia dilatada o ventrículo esquerdo torna-se dilatado a ponto das fibras musculares se esticarem ao máximo, dificultando o encurtamento dos músculos cardíacos e a compressão do sangue. A dilatação ocasiona o enfraquecimento do musculo, isso devido a maior dificuldade para a contração de um músculo mais esticado (VALDIGEM, 2014, p. 1).

2.1.1 Sintomas

Os pacientes diagnosticados com miocardiopatia dilatada, podem apresentar sinais e sintomas apenas quando a doença causar insuficiência cardíaca ou arritmias (VALDIGEM, 2014, p. 1). Os sintomas incluem:

- a) fadiga e fraqueza;
- b) falta de ar (dispneia), quando o paciente está ativo ou deitado;
- c) vertigens, tonturas ou desmaios;
- d) tosse persistente, especialmente quando deitado;
- e) inchaço (edema) nas pernas, tornozelos e pés;
- f) inchaço do abdome;
- g) ganho de peso repentino devido a retenção de líquidos;
- h) falta de apetite;
- i) sensação de batimentos cardíacos rápidos (palpitações);
- j) pele pálida.

2.1.2 Diagnóstico

Na presença de sintomas associados à insuficiência cardíaca ou arritmia, tais como falta de ar e cansaço, o médico solicita exames ao paciente para verificar a funcionalidade cardíaca e identificar as possíveis causas para doença. Os resultados desses exames auxiliam o médico a diagnosticar a doença e a conduzir o tratamento (VALDIGEM, 2014, p. 1).

Segundo Fochesatto Filho e Barros (2013, p. 1), os principais exames complementares diagnóstico de miocardiopatia dilatada, são:

- a) raio x de tórax: verifica o aumento da silhueta cardíaca;
- b) eletrocardiograma: acompanhamento do ritmo cardíaco;
- c) ecocardiograma: estabelece-se o diagnóstico de insuficiência cardíaca;
- d) cateterismo cardíaco: quando houver suspeita de doença das artérias coronárias;
- e) biópsia endomiocárdica: indicado para suspeita de doenças diagnosticadas por exame histopatológico.

2.2 COLETA DE SINAIS FISIOLÓGICOS

A aquisição de sinais fisiológicos é importante no tratamento e no diagnóstico de determinadas patologias. Entre os sinais fisiológicos mais utilizados pela medicina, destaca-se o uso do eletrocardiograma, que permite através da leitura da frequência cardíaca diagnosticar diversas doenças do coração (RAMINHOS, 2009, p. 5).

2.2.1 Frequência cardíaca

Segundo a definição de Lima (2016, p. 1), a frequência cardíaca é a quantidade de vezes que o coração bate por minuto, mensurado pela unidade de medida batimentos por minuto (bpm). Sendo que a faixa normal de pulsação do coração varia entre 60 e 100 bpm podendo oscilar conforme a idade, exercício físico ou a com o diagnóstico de alguma doença cardíacas.

Um autoexame básico e prático para medir a frequência cardíaca, seria colocar o dedo indicador e médio na parte lateral do pescoço, onde se sente os batimentos cardíacos e contar quantas pulsações no período de 1 minuto. O mesmo exame pode ser realizado contando as pulsações até 15 segundos e multiplicando o resultado por 4 (LIMA, 2016, p. 1).

Conforme afirma Lima (2016, p. 1) uma outra forma de medir a frequência cardíaca e consequentemente mais segura é utilizar um dispositivo chamado frequencímetro que se encontra em lojas de produtos desportivos, conforme a Figura 2. Para Thomaz (2016, p. 1), outra alternativa seria o uso de dispositivos vestíveis capazes de adquirir a frequência cardíaca através de sensores.

Figura 2 - Frequencímetro



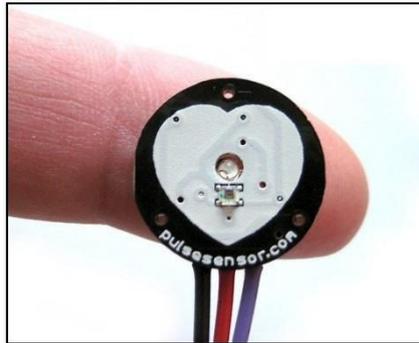
Fonte: Lima (2016, p. 1).

2.2.2 Sensor Heart Rate

Conforme Oliveira (2015, p. 1), a maior parte da luz exposta ao corpo humano é absorvida ou refletida pelos órgãos e tecidos, mas um pouco de luz passa através dos tecidos se eles forem finos o suficiente. Portanto, quando o sangue é bombeado através do organismo humano, ele é comprimido para dentro dos tecidos capilares, fazendo com que o volume dos tecidos aumente ligeiramente e entre os batimentos cardíacos, o volume dos tecidos capilares diminui. Essa mudança de volume afeta a quantidade de luz que passa através do corpo. É uma variação pequena, mas perceptível através de sensores eletrônicos.

O sensor de pulso funciona com uma fonte de luz e um detector de luz. Geralmente é utilizado um LED infravermelho e um sensor de fotodiodo, conforme a Figura 3. É essencial que os dois componentes estejam combinados, de modo que o comprimento de onda de saída de luz do diodo seja detectado pelo fotodiodo. O fotodiodo é essencialmente uma célula solar que gera tensão e corrente quando atingido pelas frequências luminosas (OLIVEIRA, 2015, p. 1).

Figura 3 - Sensor HR



Fonte: Oliveira (2015, p. 1).

2.3 WEARABLES

Conforme Sumrell (2016, p. 1), os chamados dispositivos *wearables* (termo que significa tecnologias para vestir), ainda se limitam a relógios inteligentes ou dispositivos para a prática de esportes, contudo esta é uma área onde existem enormes possibilidades de desenvolvimento e novas tecnologias surgem a cada momento. O autor acredita que nos próximos a tecnologia vestível será ainda mais diversificada e integrada ao cotidiano das pessoas na medida em que alguns especialistas da indústria projetavam vendas de 171 milhões de unidades de dispositivos *wearables* para 2016 (SUMRELL, 2016, p. 1).

Atualmente existem dois mercados bem estabelecidos para os *wearables*. São eles: o segmento *fitness*, englobando dispositivos relacionados ao bem-estar ou controle pessoal de saúde e o segmento de eletrônicos de informação e entretenimento, com dispositivos como *smartwatches*. O mercado desses dispositivos segue impulsionado por uma série de fatores, relacionados a preocupação dos consumidores com questões de saúde até a busca pelo corpo perfeito. Os fabricantes projetavam um crescimento do faturamento com vendas de *wearables*, de US\$ 1.6 bilhão em 2013 para US\$ 5 bilhões em 2016 (SUMRELL, 2016, p. 1).

Para Sumrell (2016, p. 1), existem várias áreas nas quais os *wearables* podem ser empregados, porém é no setor da saúde que se encontra o maior potencial de uso dos dispositivos vestíveis. Os *wearables* seguem como tendência para revolucionar os custos da área de saúde pública, permitindo que cada vez mais que os cuidados com o organismo

possam realizados fora de ambientes hospitalares convencionais. O autor aponta que em 2012, o mercado mundial de *wearables* voltados a área médica atingiu a marca de US\$ 2 bilhões e a expectativa é de que cheguem a US\$ 5,8 bilhões em 2019.

2.4 SAMSUNG GEAR S2 CLASSIC (SM-R732)

O Samsung Gear S2 (Figura 4) é a segunda geração do relógio inteligente da fabricante sul-coreana, o *wearable* conta várias novidades em relação ao seu antecessor. Entre os novos recursos, destaca-se as aplicações voltadas a área de saúde e bem-estar (HARADA, 2015, p. 1).

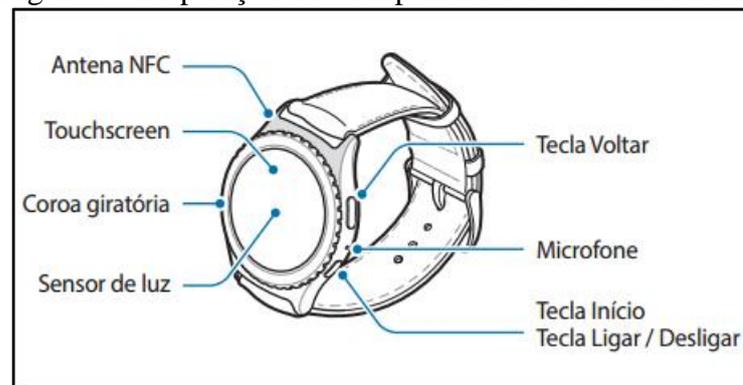
Figura 4 - Samsung Gear S2 Classic



Fonte: Samsung (2016, p. 1).

Afirma Harada (2015, p. 1), que além da mudança visual, diversas funcionalidades do sistema operacional baseado no Tizen Operational System (OS)¹ (em desenvolvimento pela Samsung para seus dispositivos móveis) foram feitas para trabalhar com a coroa giratória, a principal novidade do Gear S2. Os dois botões físicos complementam as ações do relógio, que ainda é equipado com um *display touchscreen* Super AMOLED, microfone e antena NFC (Figura 5).

Figura 5 - Disposição dos componentes do Gear S2 Classic



Fonte: Samsung (2016, p. 1).

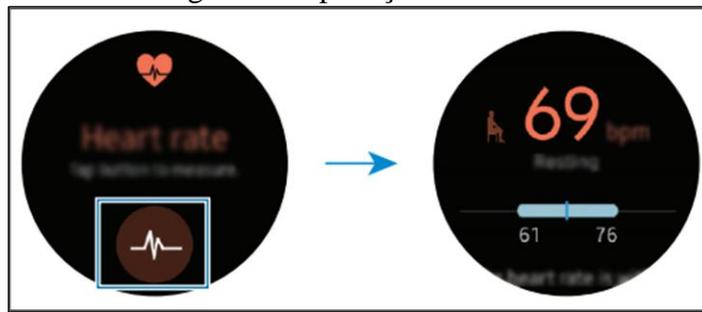
O relógio possui as seguintes especificações (HARADA, 2015, p. 1):

- a) sistema operacional: Tizen OS;
- b) tela: super AMOLED de 1,2 polegadas;
- c) resolução da tela: 360x360 pixels;
- d) densidade de pixels: 302 ppi;
- e) CPU: dual-core de 1 GHz;
- f) memória RAM: 512 MB;
- g) armazenamento interno: 4 GB;
- h) bateria: 250 mAh;
- i) conectividade: Bluetooth 4.1, WiFi, NFC e GPS (A-GPS);
- j) sensores: acelerômetro, pedômetro, giroscópio, monitor cardíaco, barômetro e luz ambiente;
- k) extras: certificação IP68 (resistência à água e poeira).

A Samsung disponibiliza uma plataforma de saúde, denominada de S-Health, a qual em conjunto com o Gear S2 disponibiliza para o relógio diversas aplicações de saúde como acompanhamento de níveis de atividades diárias ou monitoramento da frequência cardíaca. O aplicativo Heart rate (Figura 6) é um exemplo (HARADA, 2015, p. 1).

¹ Tizen OS: Sistema operacional desenvolvido em uma parceria entre a Samsung, Intel, Linux Foundation e outras corporações. Desenvolvido para ser utilizado em *smartphones*, *tablets*, *Smart TVs*, *netbooks*, *wearables* e todos os outros tipos de sistemas embarcados (SILVEIRA, 2016, p. 1).

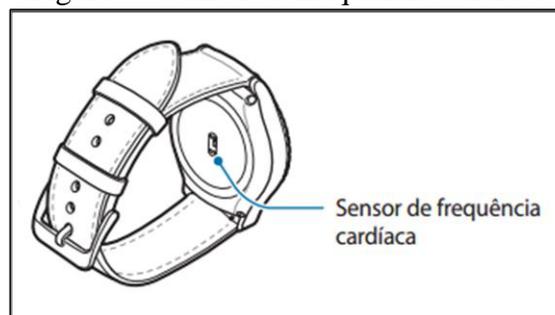
Figura 6 - Aplicação Heart rate



Fonte: Samsung (2016, p. 1).

O sensor de frequência cardíaca do Gear S2, localizado no verso do relógio (Figura 7), possibilita ao *smartwatch* medir automaticamente a frequência cardíaca em intervalos regulares. Ainda é possível alterar o intervalo de medições nas configurações da aplicação. O relógio gera relatórios gráficos diários sobre a frequência cardíaca a quais podem ser visualizados tanto do Gear S2 como no *smartphone* sincronizado com o relógio (SAMSUNG, 2016, p. 1).

Figura 7 - Sensor de frequência cardíaca



Fonte: Samsung (2016, p. 1).

2.5 TRABALHOS CORRELATOS

A seguir estão relacionados três trabalhos correlatos ao proposto. O item 2.1 apresenta um modelo de monitoramento de atividades físicas baseado em dispositivos móveis e wearables (RIGO; RITTER, 2016). O item 2.2 destaca a relevância de um sistema capaz de gerenciar o histórico de contextos fisiológicos dos usuários (OLIVEIRA, 2016). O item 2.3 incorpora o conceito de um sistema acompanhamento dos sinais vitais de pacientes com emissões de alertas sobre os dados coletados (SAMPAIO, 2011).

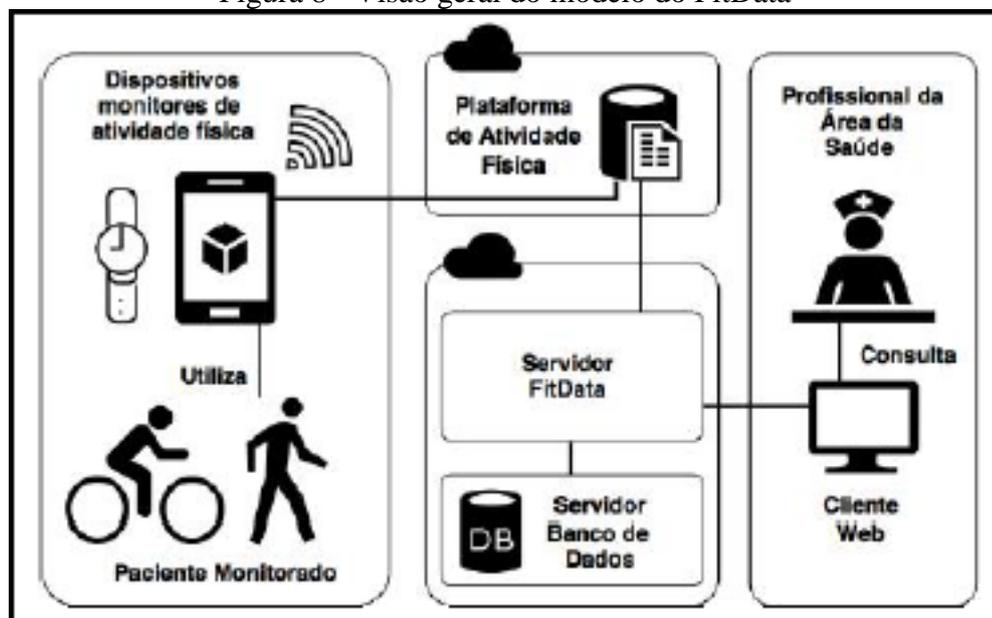
2.5.1 FitData: um sistema para monitoramento de atividade física baseado em dispositivos móveis

O trabalho consiste em um sistema de monitoramento e acompanhamento de atividade física, direcionado a profissionais da área da saúde, utilizando dispositivos móveis comercialmente disponíveis no mercado. Os sinais fisiológicos são obtidos pelos sensores

embutidos em dispositivos como *smartphones* ou *wearables*, interligados a uma plataforma online de monitoramento das atividades físicas (RIGO; RITTER, 2016, p. 550).

O modelo proposto por Rigo e Ritter (2016, p. 552), apresentado na Figura 8, é baseado no conceito de computação em nuvem que garante amplo acesso às informações por qualquer pessoa autorizada, em qualquer lugar e a qualquer momento. O monitoramento das atividades do paciente é realizado por meio de sensores embutidos em dispositivos móveis ou *wearables*, interligados a plataforma de monitoramento e armazenamento de informações na nuvem (RIGO; RITTER, 2016, p. 553).

Figura 8 - Visão geral do modelo do FitData



Fonte: Rigo e Ritter (2016, p. 552).

O Sistema utiliza a plataforma de atividade física do Google, a Google Fit. Para acessar as informações dos pacientes armazenadas nessa plataforma, foi utilizada a Application Programming Interface (API) Web disponibilizada. Essa API utiliza o protocolo OAuth 2.0 para autenticação e autorização. Este é um protocolo de padrão aberto que permite que terceiros acessem os dados do usuário sem ser necessário conhecer a senha de acesso do mesmo. A conexão entre o FitData e o Google Fit foi possível através dos serviços REST disponibilizado por meio de requisições HTTPS (RIGO; RITTER, 2016, p. 554). Segundo os autores o sistema mostra-se adequado aos propósitos e foi bem avaliado pelos especialistas da área de saúde e também pelos usuários. O FitData apresenta potencial de aplicação para fins de análise dos dados, monitoramento de pacientes e suporte a especialistas, além de ser uma ferramenta de estímulo a promoção de um estilo de vida saudável (RIGO; RITTER, 2016, p. 556).

Embora o FitData tenha obtido resultados satisfatórios, a limitação no conjunto de cenários de testes e a compatibilidade com apenas a plataforma Google Fit, limita o FitData aos sistemas Android e Android Wear, inviabilizando o uso de outros dispositivos móveis.

2.5.2 Um modelo para gerenciamento de histórico de contextos fisiológicos

O trabalho desenvolvido por Oliveira (2016), e denominado GEOCONFI, consiste em um sistema capaz de gerenciar o histórico de contextos fisiológicos de usuários, associando a relação dos sinais corporais com as características do ambiente e da situação em que eles se encontram no momento da coleta de dados. Segundo Oliveira (2016, p. 17), “sinais fisiológicos são importantes métricas para avaliar a saúde do usuário, no entanto, uma das dificuldades enfrentadas é a forma de armazenar e analisar os dados gerados pelos dispositivos de captura”.

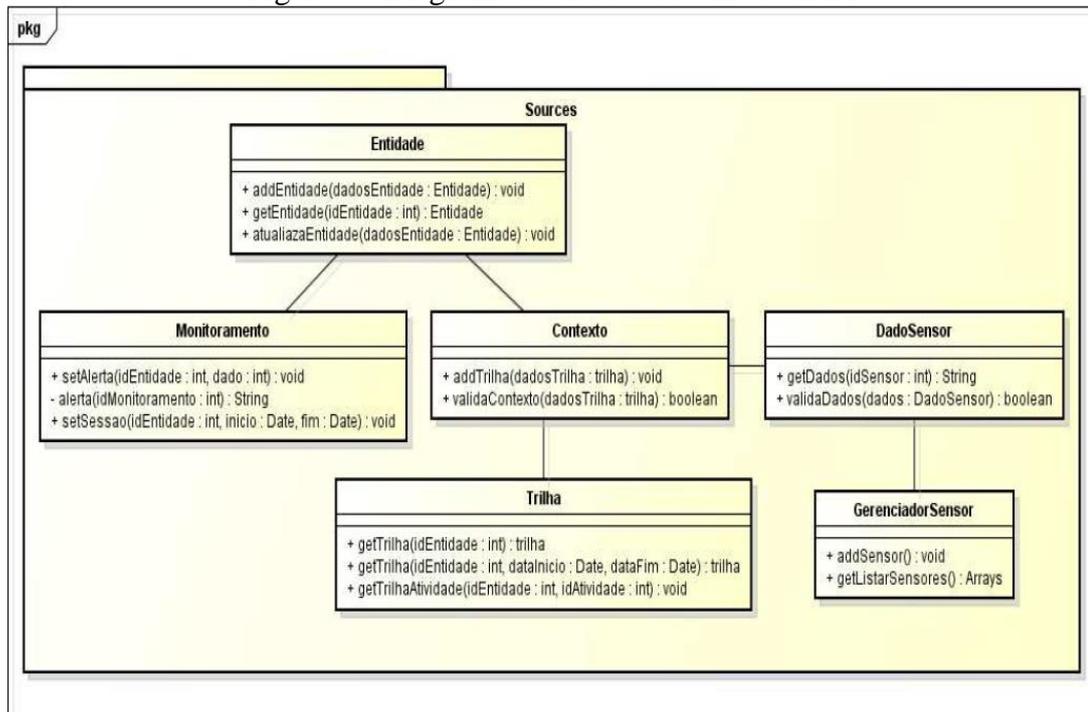
As principais características do modelo proposto por Oliveira (2016), são o gerenciamento de contextos fisiológicos de usuários, a coleta de dados através de dispositivos e sensores de monitoramento e a definição de uma ontologia para o domínio desses contextos. O foco do trabalho está no armazenamento e gerenciamento de contextos fisiológicos.

Um diferencial da proposta de Oliveira (2016, p. 38), é abordar o conceito de histórico de contextos ao invés de apenas um histórico de sinais fisiológicos. Seu modelo se baseia nos seguintes conceitos:

- a) contexto: caracterização da situação de uma entidade em um determinado momento;
- b) atividade: representa a ação/evento que foi executado pelo usuário;
- c) localização: representa a localização onde ocorreu o registro do contexto;
- d) entidade: representa uma pessoa ou usuário que deseja criar ou gerenciar trilhas;
- e) recurso: é qualquer sensor ou aparelho de monitoramento que possa gerar um contexto fisiológico;
- f) cliente: são aplicações que utilizam os serviços do servidor.

Foi modelado um sistema baseado na arquitetura cliente-servidor, onde os clientes são representados pelas Aplicações Clientes (AP) e o servidor pelo Servidor de Contextos Fisiológicos (SCF). Os clientes utilizam os mesmos serviços disponibilizados pelo servidor e ficam responsáveis por solicitar e tratar a informação conforme sua necessidade (OLIVEIRA, 2016, p. 41). As principais entidades do servidor de contexto fisiológicos são apresentadas no diagrama de classes representado na Figura 9.

Figura 9 - Diagrama de classes do GEOCONFI



Fonte: Oliveira (2016, p. 51).

Para construir o servidor, Oliveira (2016, p. 51) utilizou a linguagem de programação PHP no IDE NetBeans e implementou o banco de dados utilizando o Sistema Gerenciador de Banco de Dados (SGBD) MySQL. Para a execução do servidor web, o autor utilizou o Apache, responsável por disponibilizar os serviços do sistema. A comunicação com os clientes foi feita por meio de uma conexão utilizando o protocolo HTTP. A troca de dados entre o servidor e os clientes utilizou o formato JSON de maneira serializada e a estruturação dos serviços foi construída utilizando-se os fundamentos da arquitetura REST (OLIVEIRA, 2016, p. 52).

A implementação das Aplicações Clientes se deu a fim de avaliar o uso dos serviços ofertados pelo SCF, e para isto foram construídos três subsistemas: o SiCuide, o FitBurn e o Heart-Control, cada um com um segmento de monitoração diferente. Essas aplicações foram codificadas com a linguagem de programação PHP e suas páginas web construídas com a biblioteca bootstrap possibilitando a utilização de plug-ins JavaScript, CSS e HTML5 (OLIVEIRA, 2016, p. 53).

Uma das aplicações cliente do GEOCONFI, o módulo Heart-Control (Figura 10), verifica a frequência cardíaca do usuário localmente e sincroniza os dados com uma base remota. O Heart-Control cria os alertas para os dados informados usando a funcionalidade no módulo de monitoramento de modo que, se os batimentos cardíacos superarem o valor

máximo previamente estabelecido, o servidor envia uma notificação para a aplicação Heart-Control (OLIVEIRA, 2016, p. 64).

Figura 10 - Tela de consulta do Heart-Control

The screenshot shows a web browser window with the URL localhost/Avaliacao2/HeartControl.php. The page title is "Heart-Control". Below the title, there is a search form with the following fields: "Usuário:" with the value "João", "Atividade:" with a dropdown menu showing "Todas", "Inicio:" with the value "24/02/2016 12:26", and "Fim:" with the value "24/02/2016 13:25". A blue "Pesquisar" button is located to the right of the "Fim:" field. Below the search form is a table with 4 columns: "#", "Bpm", "Tempo", and "Atividade/Situação". The table contains 20 rows of data. The 15th row is highlighted in red.

#	Bpm	Tempo	Atividade/Situação
1	108	24/02/16 12:26	Caminhada
2	115	24/02/16 12:27	Caminhada
3	133	24/02/16 12:28	Caminhada
4	130	24/02/16 12:29	Caminhada
5	115	24/02/16 12:30	Caminhada
6	98	24/02/16 12:31	Caminhada
7	109	24/02/16 12:32	Caminhada
8	125	24/02/16 12:33	Caminhada
9	137	24/02/16 12:34	Caminhada
10	113	24/02/16 12:35	Caminhada
11	95	24/02/16 12:40	Repouso
12	80	24/02/16 12:45	Repouso
13	83	24/02/16 12:50	Repouso
14	82	24/02/16 12:55	Repouso
15	144	24/02/16 13:00	Repouso
16	113	24/02/16 13:05	Repouso
17	90	24/02/16 13:10	Repouso
18	82	24/02/16 13:15	Repouso
19	75	24/02/16 13:20	Repouso
20	74	24/02/16 13:25	Repouso

Fonte: Oliveira (2016, p. 64).

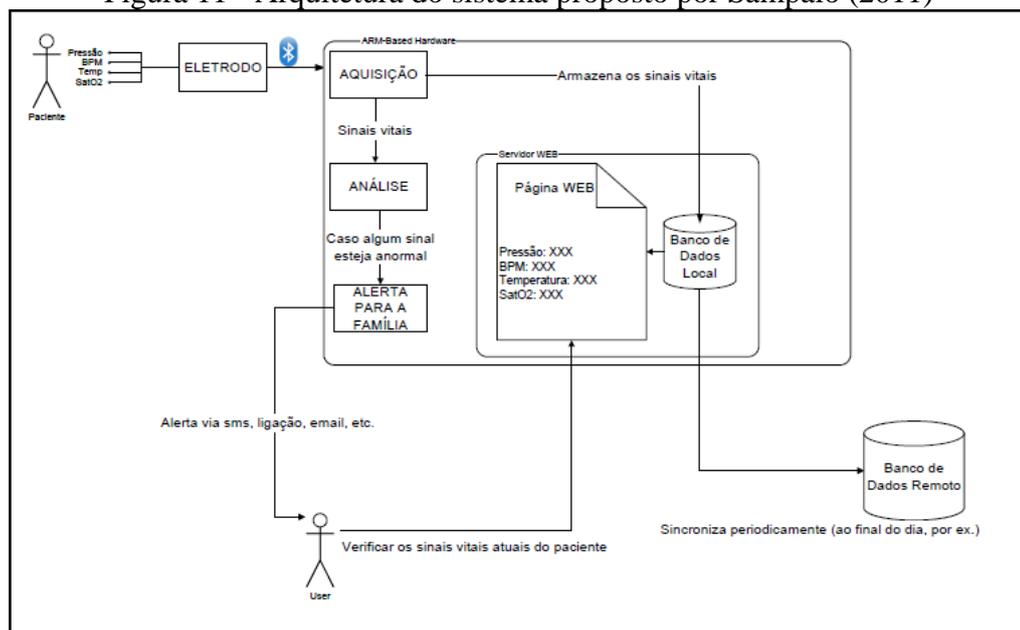
Com base nos resultados alcançados na avaliação aplicada por Oliveira (2016, p. 66), com pacientes reais, 86% deles consideram que as aplicações clientes são úteis para o monitoramento fisiológico. Dessa forma o autor constatou que o modelo proposto foi adequado aos propósitos do projeto.

2.5.3 Sistema de monitoramento remoto de pacientes implementado em hardware de arquitetura ARM

O protótipo elaborado no trabalho de Sampaio (2011), permite o monitoramento de sinais vitais de um paciente usando tecnologias sem fio e a exibição dessas informações através de uma página WEB. O protótipo alerta e concede o acesso das informações coletadas do paciente aos seus familiares e médicos a partir de qualquer lugar.

O sistema proposto por Sampaio (2011, p. 2) (Figura 11), se insere dentro no contexto de sistemas chamados *home care*, ou seja, sistemas elaborados para cuidados e/ou monitoramento de pacientes dentro dos seus ambientes domiciliares. O sistema é composto por um dispositivo de saúde chamado de Health Device Profile (HDP), que age como coletor dos sinais vitais do paciente, por uma central de monitoramento que recebe os sinais vitais, analisa e os transmite para a WEB, e um serviço WEB que apresenta os sinais vitais do paciente e os limiares estabelecidos para que o médico analise a condição de seus pacientes.

Figura 11 - Arquitetura do sistema proposto por Sampaio (2011)



Fonte: Sampaio (2011, p. 2).

Quando o sistema detecta que algum sinal vital do paciente ultrapassa o limiar estabelecido previamente por um especialista de saúde para aquele paciente em questão, o sistema emite alertas que podem variar entre SMS, e-mails e inclusive ligações em formato voz sobre IP (VoIP), para familiares, médico ou plano de saúde do paciente. Dessa forma a proposta do autor possibilita o acompanhamento remoto dos pacientes em monitoramento (SAMPAIO, 2011, p. 2).

3 DESENVOLVIMENTO

Neste capítulo são apresentadas as etapas de desenvolvimento do aplicativo Tizen e Android. A seção 3.1 apresenta os requisitos funcionais e não funcionais. A seção 3.2 descreve a especificação através de diagramas da Unified Modeling Language (UML). A seção 3.3 apresenta os detalhes da implementação, mostrando técnicas e ferramentas utilizadas, assim como a operacionalidade do sistema. Por fim, a seção 3.4 apresenta os resultados obtidos com a elaboração deste trabalho.

3.1 REQUISITOS

A seguir estão listados os requisitos funcionais (RF) e requisitos não funcionais (RNF) do sistema proposto, que deverá:

- a) ler os dados do sensor Heart Rate Monitor (HRM), incorporado no Samsung Gear S2 Classic, a cada 10 minutos (RF).
- b) enviar os dados da coleta de frequência cardíaca para a aplicação Android (RF);
- c) armazenar os sinais de frequência cardíaca na aplicação Android (RF);
- d) apresentar o histórico de frequência cardíaca na aplicação Android (RF);
- e) utilizar o dispositivo *wearable*, Samsung Gear S2 Classic, para coleta da frequência cardíaca (RNF);
- f) utilizar as linguagens de programação JavaScript, CSS e HTML para desenvolvimento da aplicação *wearable* (RNF);
- g) utilizar o dispositivo *smartphone*, com Android 5.0 ou superior, para armazenar o histórico de frequência cardíaca (RNF);
- h) utilizar a linguagem de programação Java para o desenvolvimento da aplicação móvel (RNF);
- i) utilizar a API Samsung Accessory Protocols (SAP), para implementação da comunicação entre o *smarthphone* e *wearable* (RNF).

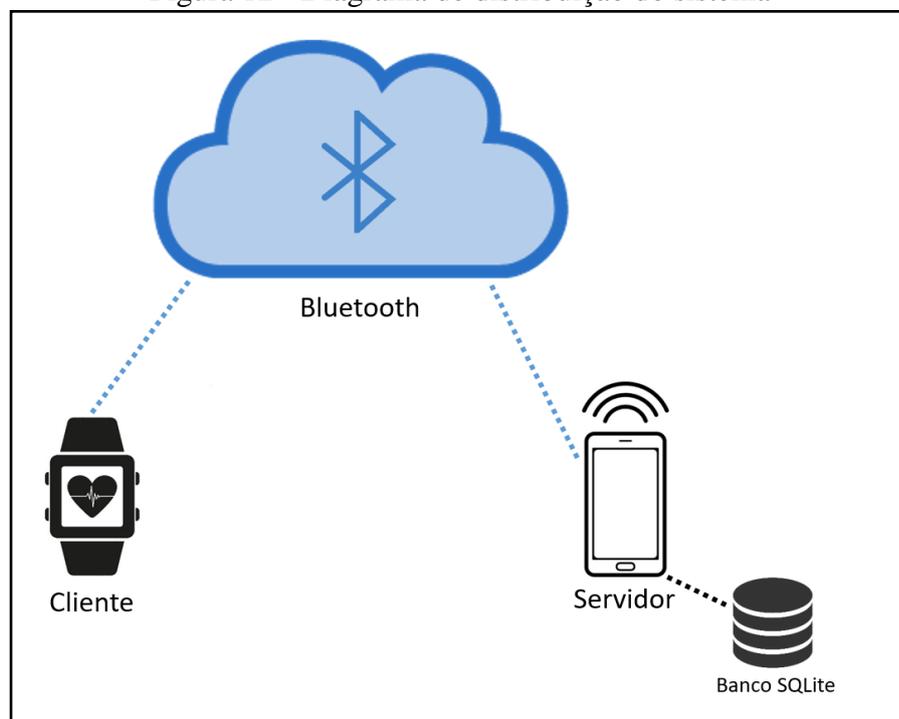
3.2 ESPECIFICAÇÃO

Para a especificação dos aplicativos foram utilizados diagramas de casos de uso, de classes e de atividades da UML, através da ferramenta Enterprise Architect (EA). Também foi especificado um diagrama de distribuição para o sistema.

3.2.1 Diagrama de distribuição

A Figura 12 apresenta o diagrama de distribuição do sistema desenvolvido. O *smartwatch* contém o serviço cliente, que recebe os comandos do servidor e retorna à frequência cardíaca para o celular que exerce o serviço de servidor, realizando requisições ao relógio, recebendo as medições e armazenando-as no banco de dados do sistema. A comunicação entre cliente e servidor é realizada via conexão *Bluetooth*, onde foi utilizado a API SAP que oferece protocolos de comunicação para facilitar a comunicação entre dispositivos Samsung e acessórios.

Figura 12 - Diagrama de distribuição do sistema

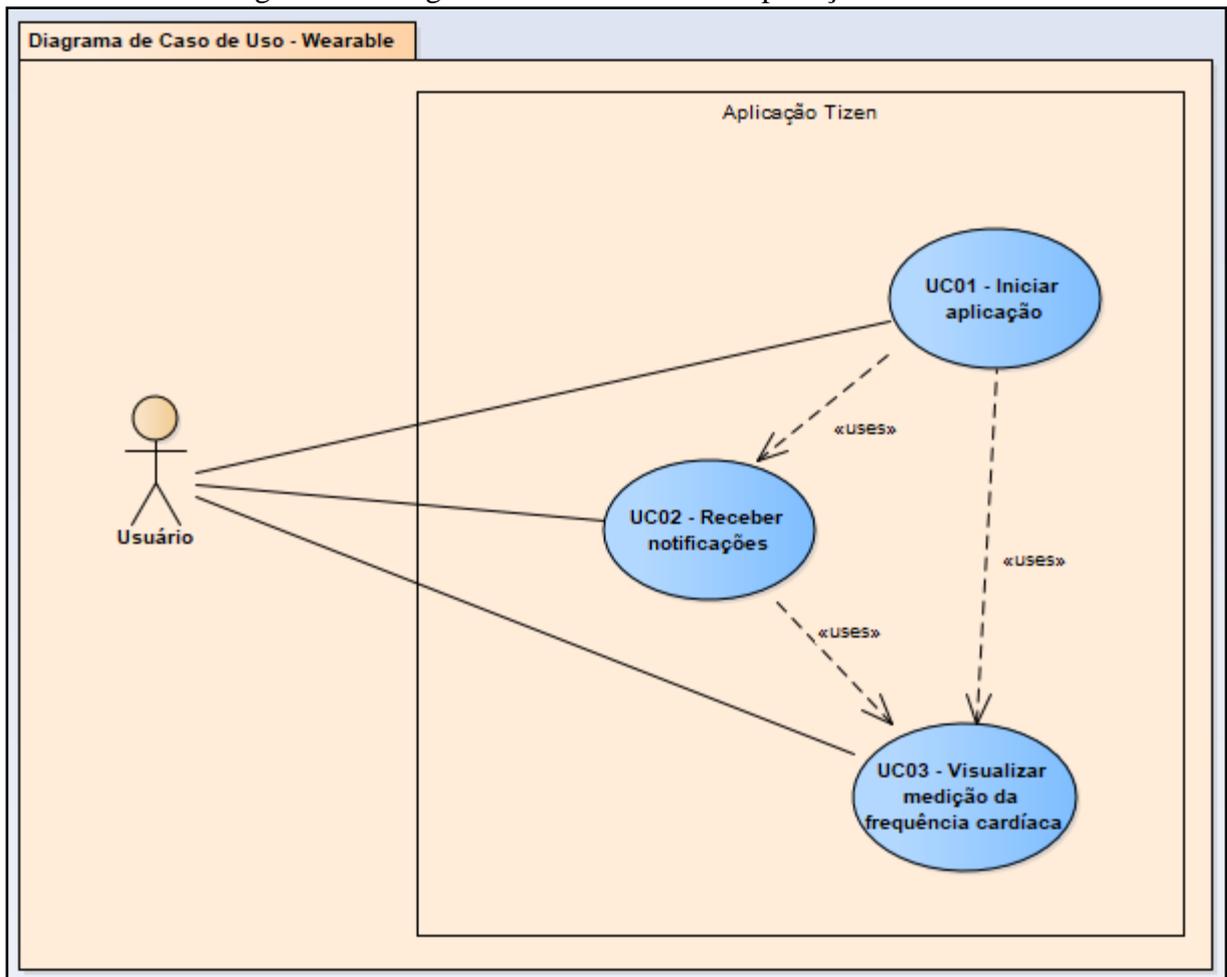


Fonte: elaborado pelo autor.

3.2.2 Diagramas de casos de uso

A Figura 13 exibe o diagrama de casos de uso com as funções disponibilizadas pela aplicação *wearable* para o ator Usuário. Nesse diagrama, o Usuário pode iniciar a aplicação, receber as notificações da aplicação e visualizar a medição da frequência cardíaca. No caso de uso UC01 - Iniciar aplicação, o Usuário pode iniciar a aplicação wearable no relógio. No caso de uso UC02 - Receber notificações, o Usuário pode receber as notificações da aplicação, início de leitura do sensor, parada de leitura do sensor e alerta sobre a falha na leitura do sensor. No caso de uso UC03 - Visualizar medição da frequência cardíaca, o Usuário pode visualizar a leitura da frequência cardíaca realizada pelo sensor do relógio.

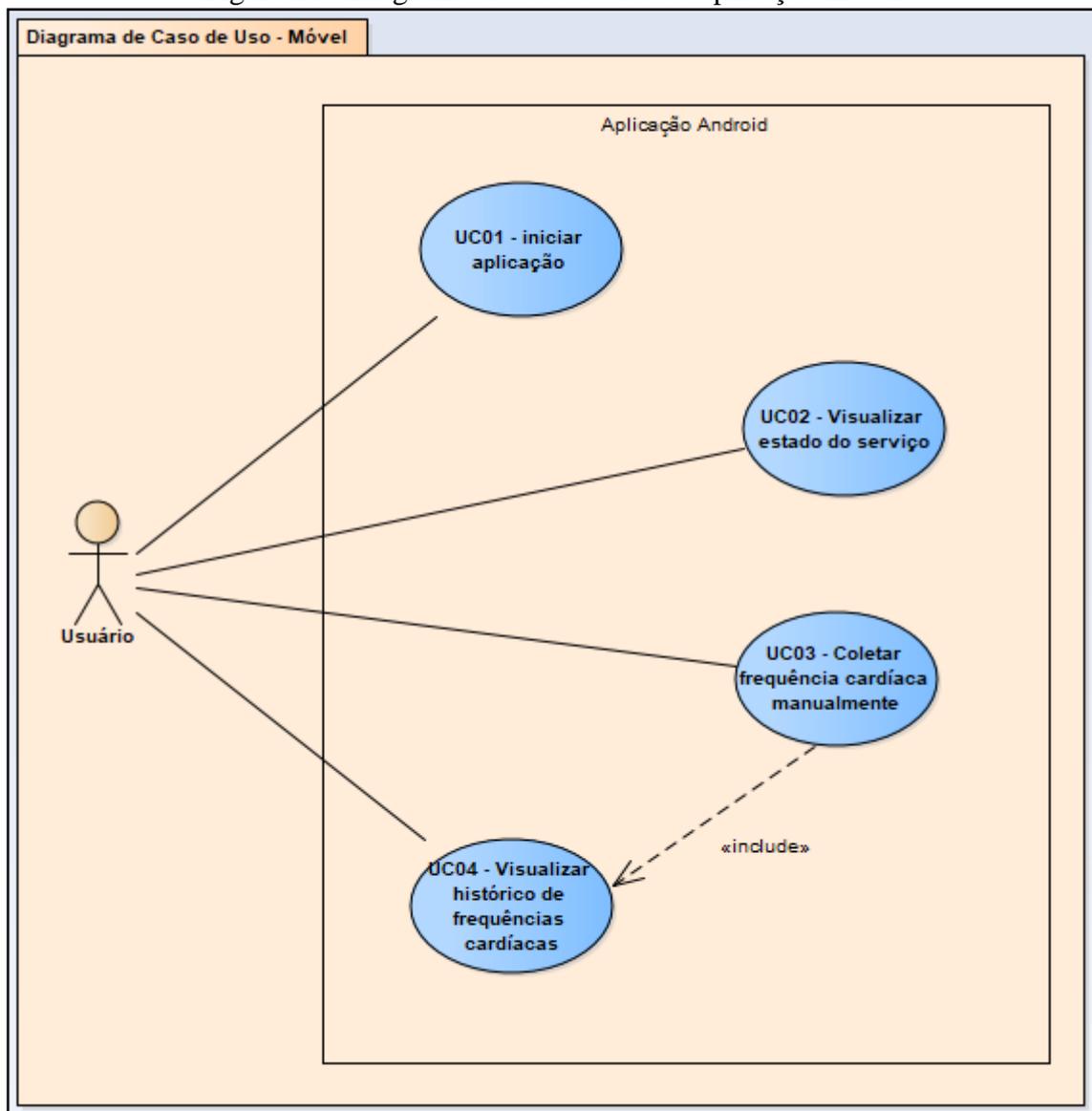
Figura 13 - Diagrama de caso de uso da aplicação wearable



Fonte: elaborado pelo autor.

A Figura 14 exibe o diagrama de casos de uso com as funções disponibilizadas pela aplicação móvel para o ator Usuário. Nesse diagrama, o Usuário pode iniciar a aplicação, visualizar o estado do serviço, coletar a frequência cardíaca manualmente e visualizar o histórico de frequências cardíacas. No caso de uso UC01 - Iniciar aplicação, o Usuário pode iniciar o aplicativo Android no celular. No caso de uso UC02 - Visualizar estado do serviço, o Usuário pode visualizar o estado do serviço de comunicação na tela da aplicação móvel, que pode variar em: Serviço Ativo, Conectado ou Desconectado. No caso de uso UC03 - Coletar frequência cardíaca manualmente, o Usuário pode solicitar a leitura da frequência cardíaca de forma manual através do botão COLETAR FREQUÊNCIA CARDÍACA na tela aplicação móvel. No caso de uso UC04 - Visualizar histórico de frequências cardíacas, o Usuário pode visualizar o histórico das frequências cardíacas registradas pelo sistema.

Figura 14 - Diagrama de caso de uso da aplicação móvel

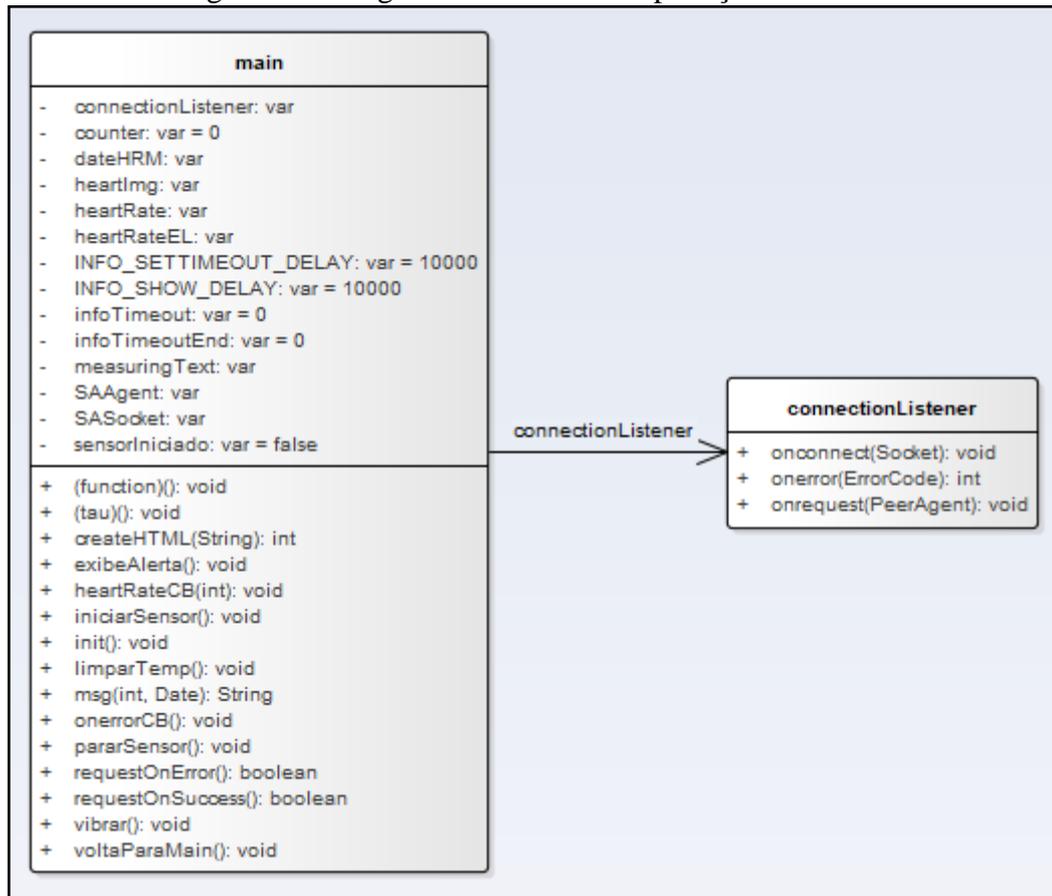


Fonte: elaborado pelo autor.

3.2.3 Diagramas de classes

A Figura 15 exibe as classes que compõem o aplicativo wearable, que são: (a) `main`, responsável por controlar a interface da aplicação, acionar o sensor de frequência cardíaca e instanciar os objetos que implementam os métodos do Framework Accessory Services, da Samsung, responsáveis por estabelecer a comunicação entre o relógio e o celular; (b) `connectionListener`, estrutura da classe `main`, responsável por implementar os métodos dos objetos `SAAgent` e `SASocket` do Framework Accessory Services.

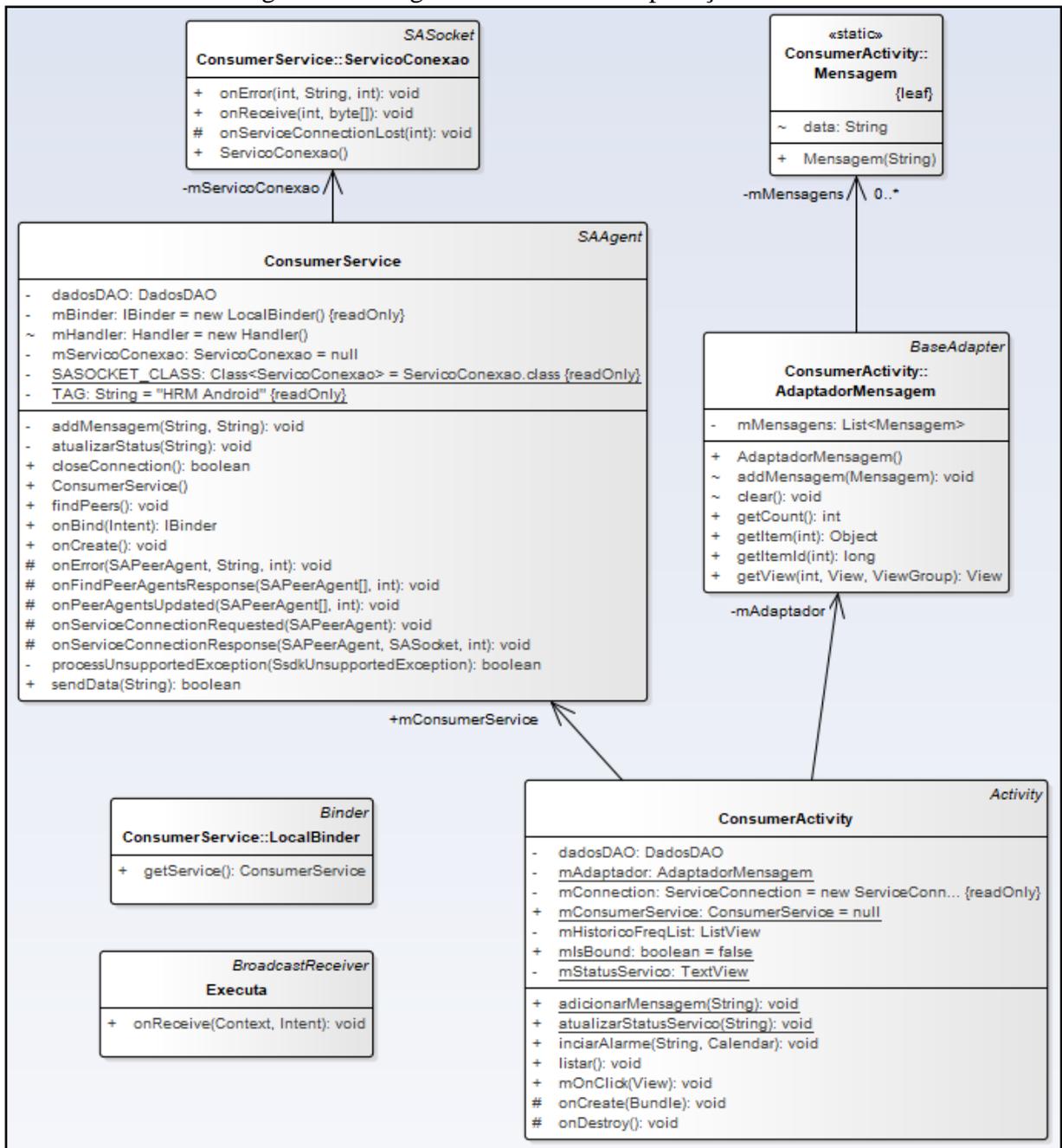
Figura 15 - Diagrama de classes da aplicação wearable



Fonte: elaborado pelo autor.

A Figura 16 exibe as classes lógicas que compõem o aplicativo móvel, que são: (a) *ConsumerActivity*, gerencia a tela da aplicação, controla o alarme da aplicação, e instancia os objetos responsáveis pelos serviços de comunicação da aplicação móvel; (b) *Mensagem*, classe responsável receber a informação das mensagens trocadas entre as aplicações. (c) *AdaptadorMensagem*, adapta a mensagem para a lista com o histórico de frequência cardíaca apresenta na tela da aplicação móvel; (d) *ConsumerService*, estende a classe *SAAgent* do Framework Accessory Service responsável por procurar e requisitar conexão a serviços agentes disponíveis; (e) *ServicoConexao*, estende a classe *SASocket* do Framework Accessory Service responsável por estabelecer a conexão entre um serviço provedor e um serviço consumidor; (f) *LocalBinder*, cria um vínculo da classe *ConsumerService* com o serviço de comunicação implementado na classe *ServicoConexão*; (g) *Executa*, classe que armazena os métodos que serão executados quando o alarme da aplicação móvel disparar.

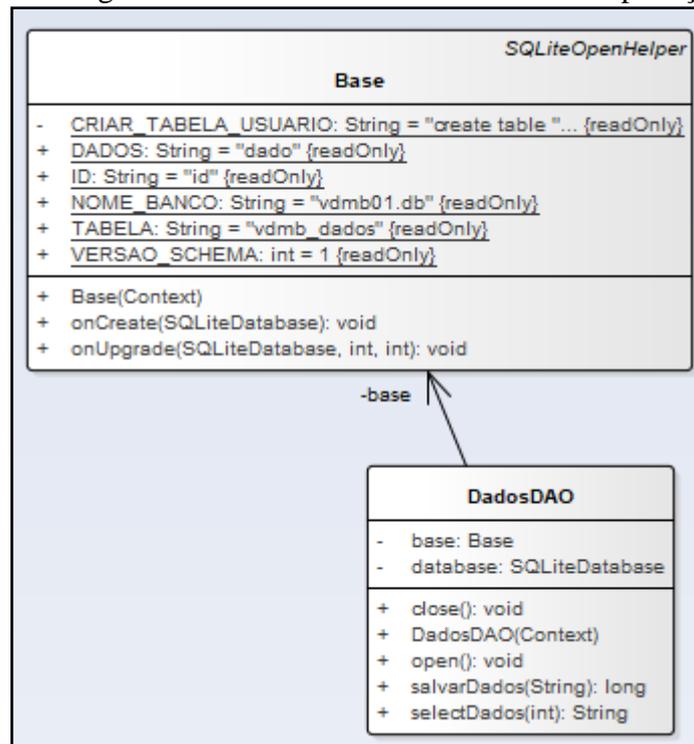
Figura 16 - Diagrama de classes da aplicação móvel



Fonte: elaborado pelo autor.

A Figura 17 exibe as classes persistência que compõem o aplicativo móvel, que são (a) Base, classe responsável por criar o banco de dados SQLite da aplicação móvel; (b) DadosDAO, classe que implementa o padrão para persistência de dados Data Access Object (DAO), responsável por obter as conexões e executar os comandos SQL.

Figura 17 - Diagrama de classes do banco de dados da aplicação móvel

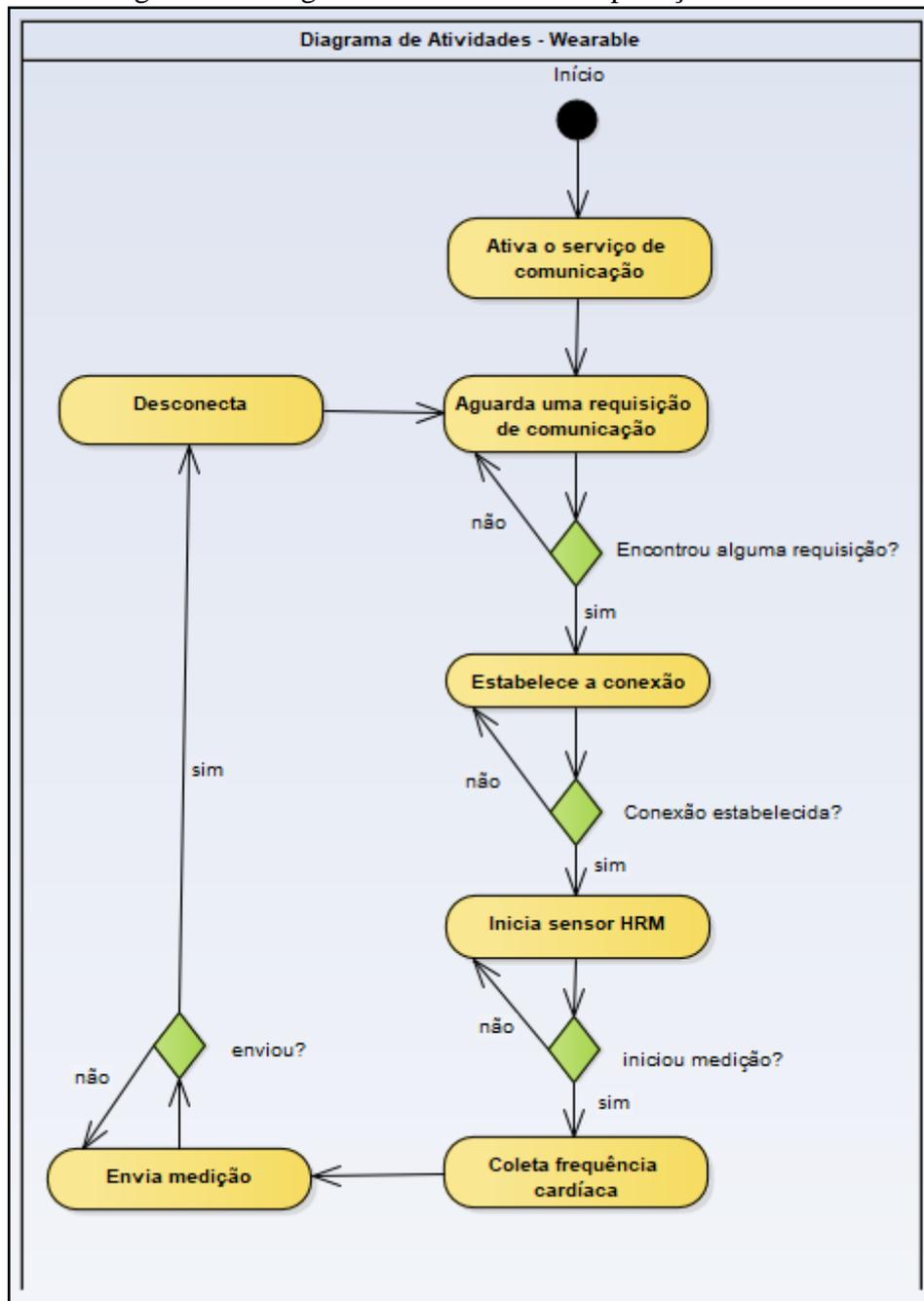


Fonte: elaborado pelo autor.

3.2.4 Diagramas de atividades

Na Figura 18 é apresentado o diagrama de atividades da aplicação wearable. Quando a aplicação é iniciada, inicia-se o serviço do agente de comunicação, que busca pares disponíveis para comunicação. Após a aplicação ter sido iniciada, a mesma fica à espera de uma requisição de comunicação oriunda da aplicação móvel. Ao receber uma requisição, as aplicações estabelecem um canal de comunicação para troca de mensagens, quando a conexão for estabelecida a aplicação wearable inicia o sensor de frequência cardíaca. Após efetuada a coleta da frequência cardíaca por intermédio do sensor HRM do relógio, a aplicação wearable retorna uma mensagem para a aplicação móvel com o valor da frequência cardíaca, hora e data da medição. Após o envio da medição a conexão entre os dois dispositivos é desfeita e a aplicação wearable volta ao estado de espera de uma nova requisição de conexão, sem a necessidade de estar executando em primeiro plano.

Figura 18 - Diagrama de atividades da aplicação wearable

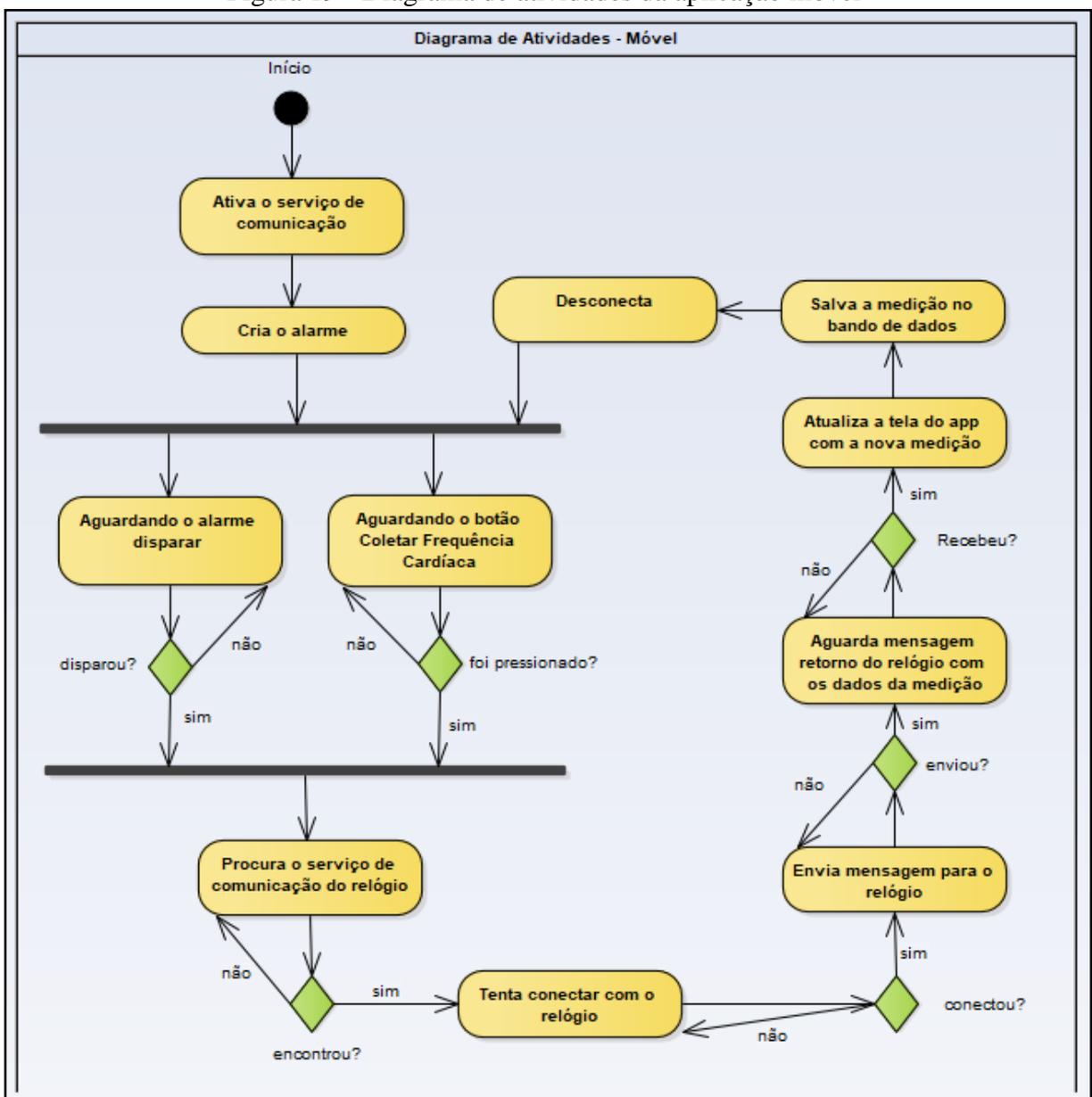


Fonte: elaborado pelo autor.

A Figura 19 apresenta o diagrama de atividades da aplicação móvel. Quando a aplicação é iniciada ela ativa o serviço do agente de comunicação e cria o alarme para fazer a aplicação solicitar a medição da frequência cardíaca de tempo em tempo. Após a aplicação ter sido iniciada, com serviço de comunicação ativo e alarme criado, a aplicação fica à espera do evento de disparo do alarme, executando a medição da frequência cardíaca de forma autônoma, ou a espera do click do botão COLETAR FREQUÊNCIA CARDÍACA, quando o usuário solicita uma coleta manualmente, fora do agendamento do alarme da aplicação. Após a ocorrência do disparo do alarme ou do click do botão de coleta, a aplicação móvel procura o

serviço do agente de comunicação da aplicação *wearable*. Se o agente do relógio estiver disponível, a aplicação móvel estabelece a comunicação e envia uma mensagem para o relógio aguardando o retorno com a medição da frequência cardíaca coletada na aplicação *wearable*. Quando a mensagem com a informação da coleta da frequência é recebida, a aplicação móvel atualiza o histórico de frequências cardíacas com a nova medição e faz a persistência dessa nova medição no banco de dados da aplicação. Enfim a comunicação é desfeita e a aplicação móvel retorna ao estado de espera do disparado do alarme ou ação do botão de coleta da aplicação.

Figura 19 - Diagrama de atividades da aplicação móvel



Fonte: elaborado pelo autor.

3.3 IMPLEMENTAÇÃO

A seguir são mostradas as técnicas e ferramentas utilizadas, descrito a obtenção de dados do sensor HRM e a comunicação entre as aplicações Tizen e Android, bem como apresentada a operacionalidade da implementação.

3.3.1 Técnicas e ferramentas utilizadas

O aplicativo para a plataforma Tizen foi desenvolvido como uma aplicação Web utilizando as linguagens de programação CSS, HTML e JavaScript juntamente com o ambiente de desenvolvimento Tizen Studio 1.2.

O aplicativo Android foi desenvolvido utilizando a linguagem Java, através da IDE Android Studio 2.3.3. Para a tela do aplicativo foi utilizada a classe `Activity`, que é a classe que gerencia a interface com o usuário. Para o armazenamento dos dados, foi utilizado o banco de dados SQLite. As trocas de informações da interface com o banco de dados são realizadas através de classes que utilizam o padrão DAO, que determina que toda a interação com o banco de dados seja feita por essa classe sem influenciar as demais classes do sistema. Para as operações do aplicativo móvel que realizam comunicação com a aplicação Tizen, foi utilizada a biblioteca Accessory SDK, mantida pela Samsung, que tem a proposta de facilitar a conexão de acessórios aos dispositivos da Samsung abstraindo preocupações relacionadas a conectividade ou protocolos de rede.

3.3.1.1 Obtenção de dados do sensor HRM do Gear S2

Para acessar o sensor HRM do relógio foi utilizado a API Human Activity Monitor (HAM) que disponibiliza acesso aos sensores que monitoram a atividade humana nos dispositivos Tizen. Para utilizar a API é preciso solicitar a permissão adicionando o privilégio: `<tizen:privilege name="http://tizen.org/privilege/healthinfo"/>` que foi adicionado no arquivo `config.xml` do projeto. O Quadro 1, apresenta a função `iniciarSensor` que inicia o sensor HRM do relógio através chamada da função `tizen.humanactivitymonitor.start`, disponibilizada pela HAM API.

Quadro 1 - Função para iniciar o Sensor HRM

```
function iniciarSensor() {
    vibrar();
    sensorIniciado = true;
    limparTemp();
    measuringText.classList.remove('hide');
    measuringText.classList.add('show');

    tizen.humanactivitymonitor.start('HRM', heartRateCB, onerrorCB);
}
```

Fonte: elaborado pelo autor.

A função `tizen.humanactivitymonitor.start`, recebe três parâmetros. O primeiro identifica o sensor a ser utilizado. O segundo define a função de retorno que receberá os dados coletados pelo sensor, a função `heartRateCB` é apresentada no Quadro 2. E o terceiro define uma função de retorno em caso de erro ao tentar acessar o sensor de frequência cardíaca.

Quadro 2 - Função de retorno do sensor HRM

```
function heartRateCB(heartRateInfo) {
    if (!sensorIniciado){
        counter = 0;
    }
    counter++;
    var rate = heartRateInfo.heartRate;
    var activePage = document.getElementsByClassName('ui-page-active')[0];
    var activePageId = activePage ? activePage.id : '';
    if (rate < 1) {
        counter = 0;
        rate = 0;
        heartRateEl.innerHTML = '';
        measuringText.classList.remove('hide');
        measuringText.classList.add('show');
        if (activePageId === 'main' && infoTimeout === 0) {
            infoTimeout = window.setTimeout(showMeasuringInfo, INFO_SETTIMEOUT_DELAY);
        }
    } else {
        limparTemp();
        hideMeasuringInfo();
        if (!heartImg.classList.contains('animate')) {
            measuringText.classList.remove('show');
            measuringText.classList.add('hide');
        }
        heartRateEl.innerHTML = rate;
        heartRate = rate;
        dateHRM = new Date();
        if (counter > 100)
        {
            pararSensor();
        }
    }
}
```

Fonte: elaborado pelo autor.

A função `heartRateCB` foi implementada para receber os dados de frequência cardíaca coletados pelo sensor e atualizar a informação da frequência apresentada na tela da aplicação

do relógio. Essa função recebe dados do sensor continuamente durante 100 iterações, para o sensor coletar uma frequência estabilizada, ou quando o botão parar da aplicação tiver sido acionado, parando o sensor através da execução da função `pararSensor` apresentada no Quadro 3.

Quadro 3 - Função de parada do sensor HRM

```
function pararSensor () {  
    counter = 0;  
  
    vibrar();  
    sensorIniciado = false;  
    measuringText.classList.remove('show');  
    measuringText.classList.add('hide');  
  
    limparTemp();  
  
    tizen.humanactivitymonitor.stop("HRM");  
}
```

Fonte: elaborado pelo autor.

A função `pararSensor` atualiza a tela da aplicação, limpa os temporizadores, altera a informação do botão de controle da interface do usuário e para o sensor de frequência cardíaca do relógio com a chamada da função `tizen.humanactivitymonitor.stop` da HAM API.

3.3.1.2 Comunicação entre os aplicativos Tizen e Android

A comunicação entre os aplicativos do sistema proposto nesse trabalho utiliza a SAP API, disponibilizada pela Samsung através da sua biblioteca Accessory SDK. Essa biblioteca possibilita a comunicação entre agentes fornecedores e consumidores através de um serviço de troca de mensagens baseados em *Sockets*. O aplicativo Tizen executa o serviço de fornecedor, para enviar os dados do sensor ao aplicativo Android que implementa o serviço consumidor, recebendo as informações enviadas pelo relógio.

O perfil do serviço fornecedor foi configurado no código Tizen através do arquivo `accessoryservices.xml`, apresentado no Quadro 4. Esse arquivo configura o perfil do serviço para seja possível o estabelecimento da comunicação entre os agentes fornecedores e consumidores.

Quadro 4 - Perfil do serviço fornecedor

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE xml>
<resources>
  <application name="HRM">
    <serviceProfile
      id="/tcc/hrm"
      name="hrmtizen"
      autoLaunchAppId="3otE3CXdrw.HRM"
      role="provider"
      version="1.0"
      serviceLimit="ANY"
      serviceTimeout="30">
      <supportedTransports>
        <transport type="TRANSPORT_BT" />
      </supportedTransports>
      <serviceChannel
        id="104"
        dataRate="HIGH"
        priority="LOW"
        reliability="ENABLE"/>
      </serviceProfile>
    </application>
  </resources>

```

Fonte: elaborado pelo autor.

A tag <application> define o nome da aplicação que executa o serviço. A tag <serviceProfile> define o id do serviço que deve ser o mesmo no para o fornecedor e consumidor, o parâmetro `role` define que o serviço vai ser um fornecedor, o parâmetro `serviceLimit` define que o serviço poderá se conectar com mais de um consumidor e a o parâmetro `serviceTimeout` define um tempo de espera para o estabelecimento da conexão de 30 segundos. A tag <supportedTransports> define que a troca de mensagens entre os agentes ocorre via Bluetooth. E por fim, a tag <serviceChannel> define o canal de envio das mensagens onde foi estabelecido o id do canal, a frequência de informações enviadas, a prioridade de envio, e através do parâmetro `reliability` a tentativa de reenvio de mensagens.

A aplicação Tizen requisita os agentes de conexão através da função `webapis.sa.requestSAAgent`, disponibilizada pela SAP API, apresentada no Quadro 5. Essa função possui como parâmetros as funções `requestOnSuccess` que seta o ouvinte do serviço de conexão para cada agente fornecedor encontrado e `requestOnError` que retorna uma mensagem de erro no caso de falha na procura por agentes.

Quadro 5 - Função requestSAAgent

```

webapis.sa.requestSAAgent(requestOnSuccess, requestOnError);

function requestOnSuccess (agents) {
    var i = 0;
    for (i; i < agents.length; i += 1) {
        if (agents[i].role === "PROVIDER") {
            SAAgent = agents[i];
            break;
        }
    }
    SAAgent.setServiceConnectionListener(connectionListener);
}

function requestOnError (e) {
    createHTML("requestSAAgent Erro" +
        "Nome: " + e.name + "<br />" +
        "Mensagem de erro: " + e.message);
}

```

Fonte: elaborado pelo autor.

A função `SAAgent.setServiceConnectionListener`, disponibilizada pela SAP API, seta os ouvintes para todos os agentes fornecedores, aguardando requisições de agentes consumidores. O seu parâmetro, `connectionListener`, implementado conforme apresentado no Quadro 6, é responsável por aceitar a requisição do agente consumidor se o nome da aplicação Android coincidir com `"HRM_Android"`, se positivo inicia o sensor HRM do relógio.

Definido o agente parceiro, o `connectionListener` estabelece uma conexão entre os dois pares via Socket, utilizando a classe `SASocket` da SAP. Com a conexão estabelecida a função `dataOnReceive` verifica o recebimento de alguma mensagem destinada a aplicação Tizen. Ao receber essa mensagem é pego o id do canal do remetente para então o agente fornecedor enviar a mensagem de volta, com a frequência cardíaca e data e hora em que foi mensurada.

Quadro 6 - Implementação do ouvinte de conexão

```

connectionListener = {
    onrequest: function (peerAgent) {
        if (peerAgent.appName === "HRM_Android") {
            SAAgent.acceptServiceConnectionRequest(peerAgent);
            iniciarSensor();
        } else {
            SAAgent.rejectServiceConnectionRequest(peerAgent);
        }
    },
    onconnect: function (socket) {
        var onConnectionLost,
            dataOnReceive;
        SASocket = socket;
        onConnectionLost = function onConnectionLost (reason) {
            createHTML("Conexão desconectada devido:<br />" + reason);
        };
        SASocket.setSocketStatusListener(onConnectionLost);
        dataOnReceive = function dataOnReceive (channelId, data) {
            if (!SAAgent.channelIds[0]) {
                createHTML("ERRO: Canal sem ID!");
                return;
            }
            SASocket.sendData(SAAgent.channelIds[0], msg(heartRate, dateHRM));
        };
        SASocket.setDataReceiveListener(dataOnReceive);
    },
    onerror: function (errorCode) {
        createHTML("Erro no serviço de conexão<br />erro: " + errorCode);
    }
};

```

Fonte: elaborado pelo autor.

O perfil do serviço consumidor foi configurado no código Android através do arquivo `accessoryservices.xml`, apresentado no Quadro 7. Esse arquivo configura o perfil do consumidor para seja possível o estabelecimento da comunicação com agentes fornecedores.

Quadro 7 - Perfil do serviço consumidor

```

<resources>
  <application name="HRM_Android" >
    <serviceProfile
      id="/tcc/hrm"
      name="hrandroid"
      role="consumer"
      serviceImpl="br.furb.tcc.android.hrm.consumer.ConsumerService"
      version="1.0"
      serviceLimit="ANY"
      serviceTimeout="30">
      <supportedTransports>
        <transport type="TRANSPORT_BT" />
      </supportedTransports>
      <serviceChannel
        id="104"
        dataRate="low"
        priority="low"
        reliability="enable"/>
    </serviceProfile>
  </application>
</resources>

```

Fonte: elaborado pelo autor.

Os parâmetros do arquivo `accessoryservices.xml` do aplicativo Android foram configurados conforme o do Tizen. Salvo os parâmetros `name` que identifica o serviço, o parâmetro `role` que para a aplicação Android foi setado como "consumer" e o parâmetro `serviceImpl` que aponta para a classe do código fonte na qual o serviço foi implementado.

A classe `ConsumerService` estende a classe `SAAgent`, que é a interface principal entre as aplicações fornecedor, consumidor e o Framework Samsung Accessory Service. A subclasse `servicoConexao`, conforme o quadro, estende a classe `SASocket`, que representa uma instância de um serviço de conexão entre o serviço fornecedor e o serviço consumidor. O método `onReceive` recebe as mensagens oriundas do agente fornecedor, atualiza a tela da aplicação Android e armazena as informações da mensagem no banco de dados.

Quadro 8 - Classe `ServicoConexao`

```
public class ServicoConexao extends SASocket {
    public ServicoConexao() {
        super(ServicoConexao.class.getName());
    }

    @Override
    public void onError(int channelId, String errorMessage, int errorCode) {
    }

    @Override
    public void onReceive(int channelId, byte[] data) {
        final String mensagem = new String(data);
        addMensagem("", mensagem);
        //--> Acesso ao banco
        dadosDAO.open();
        dadosDAO.salvarDados(mensagem);
        dadosDAO.close();
    }

    @Override
    protected void onServiceConnectionLost(int reason) {
        atualizarStatus("Desconectado");
        closeConnection();
    }
}
```

Fonte: elaborado pelo autor.

A comunicação entre a aplicação Android e Tizen ocorrem por intermédio de algumas etapas. Primeira etapa, a busca dos agentes pares, consumidor e fornecedor disponíveis para conexão. Segunda etapa, o estabelecimento de uma conexão entre os pares de agentes. Terceira etapa, a troca de mensagens entre as aplicações, iniciada pela aplicação Android que envia uma mensagem para aplicação Tizen, que quando a recebe inicia o sensor HRM, faz a

coleta da frequência cardíaca e retorna uma mensagem para aplicação Android com a frequência coletada com hora e data. A última etapa, a aplicação Android recebe o retorno do relógio, adiciona os dados recebidos no histórico de registros de frequências cardíacas e encerra a conexão entre os dois dispositivos.

A função `findPeerAgents`, estendida da classe `SAAgent` procura agentes pares disponíveis e se encarrega de estabelecer a conexão entre os serviços fornecedores e consumidores. Conforme o Quadro 9, o objeto `mServicoConexao`, instância da classe `ServicoConexao` que estende a classe `SASocket`, invoca o método `send` que envia a mensagem do celular para o relógio iniciando o sensor HRM. O mesmo objeto se encarrega de finalizar a conexão após o recebimento dos dados enviados pelo relógio com a chamada do método `close`.

Quadro 9 - Utilizando o Framework Samsung Accessory Service para envio de mensagens

```

public void findPeers() {
    findPeerAgents();
}

public boolean sendData(final String data) {
    boolean retvalue = false;
    if (mServicoConexao != null) {
        try {
            mServicoConexao.send(getServiceChannelId(0), data.getBytes());
            retvalue = true;
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
    return retvalue;
}

public boolean closeConnection() {
    if (mServicoConexao != null) {
        mServicoConexao.close();
        mServicoConexao = null;
        return true;
    } else {
        return false;
    }
}
}

```

Fonte: elaborado pelo autor.

3.3.1.3 Alarme na aplicação Android

A aplicação Android controla a aplicação Tizen, para tanto foi implementado um alarme na aplicação Android para que a mesma execute periodicamente, requisitando a coleta da frequência cardíaca ao sensor do relógio. O alarme foi implementado conforme a

AlarmManager API nativa do Android que implementa alarmes que acordam as aplicações conforme sua programação, conforme o Quadro 10.

Quadro 10 - Implementação do alarme

```
public void iniciarAlarme(String action, Calendar dt_hr_inic){
    AlarmManager alarmMgr = (AlarmManager) this.getSystemService(Context.ALARM_SERVICE);
    Intent intent = new Intent (action);
    PendingIntent alarmIntent = PendingIntent.getBroadcast(this, 1, intent, 0);

    long inicio = dt_hr_inic.getTimeInMillis();
    long intervalo = 0;
    if(action == "EXECUTA"){
        intervalo = 10 * 60 * 1000; //--> intervalo das medições
    }
    alarmMgr.setRepeating(AlarmManager.RTC, inicio, intervalo, alarmIntent);
}
```

Fonte: elaborado pelo autor.

O alarme foi programado para despertar a cada 10 minutos. Quando disparado, o alarme executa o método onReceive da classe Executa, que estabelece a comunicação entre o relógio e celular para envio dos dados do sensor HRM, conforme implementação no Quadro 11. Dessa forma as aplicações coletam e armazenam os dados das coletas de frequência cardíaca sem a necessidade de interação do usuário.

Quadro 11 - Código executado pelo alarme

```
public class Executa extends BroadcastReceiver {

    @Override
    public void onReceive(Context context, Intent intent) {

        if (ConsumerActivity.mIsBound == true && ConsumerActivity.mConsumerService != null) {

            /**
             * ConsumerActivity.mConsumerService.findPeers();

            new Handler().postDelayed(new Runnable() {
                @Override
                public void run() {
                    ConsumerActivity.mConsumerService.sendData("Olá Gear!");
                    new Handler().postDelayed(new Runnable() {
                        @Override
                        public void run() {
                            ConsumerActivity.mConsumerService.closeConnection();
                        }
                    }, 2 * 1000); //--> intervalo de fim de conexão
                }
            }, 60 * 1000); //--> intervalo do envio de requisição de retorno de msg
            /**/
        }
    }
}
```

Fonte: elaborado pelo autor.

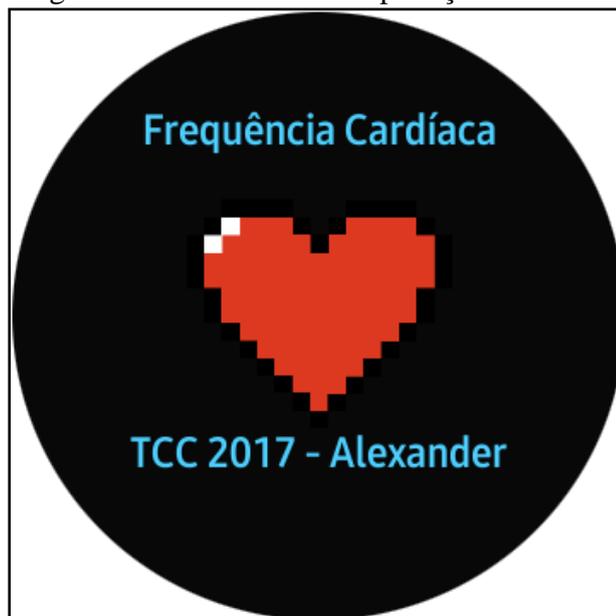
O método `onReceive` foi implementado para iniciar a comunicação entre os dois dispositivos por meio do método `findPeers`, assim que a conexão for estabelecida o sensor HRM do relógio é iniciado, pois o sensor demora um tempo para começar a ler a frequência cardíaca do usuário. Dessa forma o método `sendData` foi implementado dentro de uma `Thread` que inicia após 60 segundos para garantir que a mensagem de retorno do relógio traga uma informação válida do sensor HRM.

3.3.2 Operacionalidade da implementação

Nesse trabalho foi desenvolvido o HRM, um sistema coletor de sinais de frequência cardíaca, composto por um aplicativo *wearable* responsável por fazer a coleta da frequência cardíaca, bem como uma aplicação móvel que coordena e armazena um registro das coletas dos sinais de frequência cardíaca. A seguir é descrita a operacionalidade da aplicação *wearable* e do aplicativo móvel.

A aplicação *wearable* é controlada pela aplicação móvel, dessa forma a mesma não apresenta nenhum botão de interação com usuário do relógio, contudo possui interface para apresentar o estado atual da leitura do sensor de frequência cardíaca. Ainda que haja interface a aplicação executa em segundo plano sem a necessidade de interação com o usuário. Salvo quando o sensor não consegue realizar a leitura da frequência, aplicação alerta o usuário. A Figura 20 apresenta a tela inicial da aplicação, de quando ela é iniciada e ainda não houve nenhuma leitura realizada pelo sensor HRM.

Figura 20 - Tela inicial da aplicação *wearable*



Fonte: elaborado pelo autor.

A aplicação móvel possui uma única tela, apresentada na Figura 21. O sistema não exige a interação do usuário pois executa em primeiro ou segundo plano em ambas aplicações. A aplicação móvel opera requisitando as leituras de frequência cardíaca à aplicação *wearable* de tempos em tempos, recebendo-as e armazenando-as gerando dessa forma um histórico com das leituras de frequência cardíaca do usuário.

Cada medição apresenta três informações sobre a leitura: a frequência cardíaca em batimentos por minuto, a hora e a data na qual foi coletada. Ainda que o sistema realize as medições de forma autônoma, a interface da aplicação Android possui o botão `COLETAR FREQUÊNCIA CARDÍACA`, para que o usuário possa fazer uma medição de forma síncrona.

Figura 21 - Tela da aplicação móvel

Frequência	Horário	Data
078 bpm	14:11	20/06/2017
074 bpm	14:20	20/06/2017
076 bpm	14:21	20/06/2017
066 bpm	14:30	20/06/2017
086 bpm	14:40	20/06/2017
077 bpm	14:50	20/06/2017
094 bpm	15:00	20/06/2017
079 bpm	15:10	20/06/2017
072 bpm	15:20	20/06/2017
076 bpm	15:34	20/06/2017
076 bpm	15:44	20/06/2017
078 bpm	15:54	20/06/2017
071 bpm	16:04	20/06/2017
083 bpm	16:13	20/06/2017

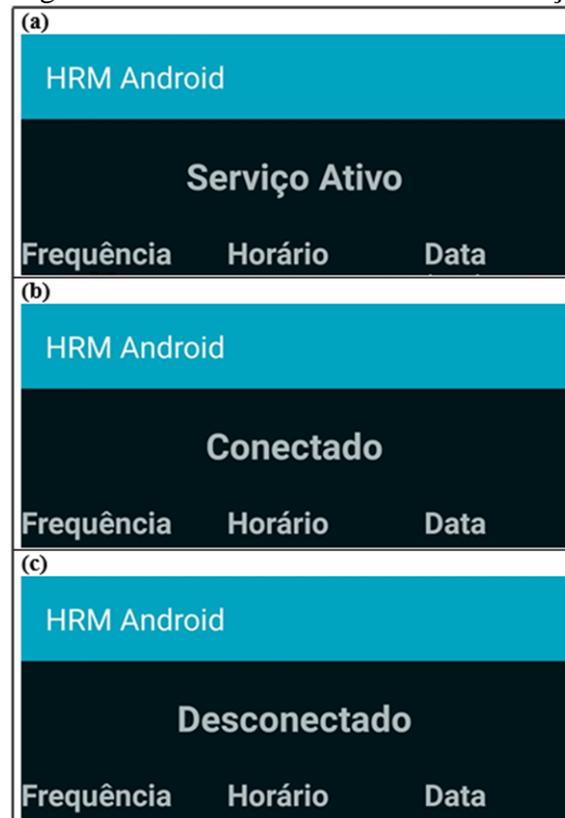
COLETAR FREQUÊNCIA CARDÍACA

TCC 2017 - Alexander T. G. Furtado

Fonte: elaborado pelo autor.

A tela da aplicação móvel apresenta o status do serviço de conexão entre as duas aplicações. O status pode informar `Serviço Ativo` Figura 22 (a), quando a aplicação é iniciada, informa que os serviços de comunicação estão disponíveis nos dois dispositivos. Pode informar `Conectado` Figura 22 (b), quando a comunicação entre o relógio e celular foi estabelecida. E ainda `Desconectado` Figura 22 (c), após ter havido o a troca de mensagens entres as aplicações informando que a comunicação entre relógio e celular foi encerrada.

Figura 22 - Informativo do status do serviço



Fonte: elaborado pelo autor.

Quando a aplicação móvel estabelece uma conexão com a aplicação *wearable*, o sensor do relógio é iniciado. Dessa forma a interface da aplicação do relógio é alterada informando que o sensor está sendo iniciado, conforme a Figura 23.

Figura 23 - Tela de inicialização do sensor HRM



Fonte: elaborado pelo autor.

O sensor HRM demora alguns segundos para se estabilizar e começar a ler a frequência cardíaca do usuário. A aplicação Tizen, espera o sensor iniciar a leitura em dentro de 10 segundos, não ocorrendo essa condição a aplicação mostra a tela de aviso para usuário verificar o relógio ou sensor, conforme a Figura 24.

Figura 24 - Tela de aviso de ajuste do relógio



Fonte: elaborado pelo autor.

Quando o sensor começa a leitura da frequência cardíaca do usuário do relógio a aplicação *wearable* entra na tela de medição (Figura 25). A tela de medição é atualizada de acordo com as leituras do sensor HRM, alternando o valor da frequência na tela por algumas iterações até o sensor concluir a leitura e definir o valor daquela medição.

Figura 25 - Tela de medição



Fonte: elaborado pelo autor.

O processo de medição se encerra quando a leitura chega na aplicação móvel sendo adicionada no histórico de leituras anteriores, conforme mostra a Figura 26. Neste estado o sensor está desligado para não consumir a bateria do relógio e ambas aplicações aguardam a próxima medição que será realizado ao disparar o alarme configurado na aplicação Android.

Figura 26 - Medição gravada no histórico de leituras



Fonte: elaborado pelo autor.

3.4 ANÁLISE DOS RESULTADOS

Conforme apresentado ao longo do texto, constata-se que foi possível alcançar o objetivo geral proposto de desenvolver um protótipo de sistema para acompanhamento da frequência cardíaca de pacientes com insuficiência cardíaca. Entretanto algumas dificuldades encontradas durante o desenvolvimento do mesmo demandaram esforço de adequação dos objetivos específicos tendo em vista adequá-los ao cumprimento do objetivo principal.

Foram encontradas dificuldades no início das atividades com o sistema operacional Tizen, tendo em vista que o mesmo é um sistema novo e que ainda não possui muito conteúdo disponível. Além disso, ainda há poucas iniciativas para desenvolvimentos de aplicações. Não obstante isto, a estrutura da organização da API é confusa, pois alguns exemplos fornecidos apresentavam erros e há uma pequena comunidade envolvida no desenvolvimento de soluções para esta plataforma.

Outro obstáculo superado esteve associado a problemas com atualizações do sistema operacional do relógio e da aplicação que gerencia o relógio no celular o que causou alguns transtornos como, perdas de comunicação entre relógio e celular, onde só era possível restabelecer a comunicação entre os dois dispositivos manualmente.

Contudo, foi interessante desenvolver o código do Tizen como uma aplicação Web, utilizando linguagens apropriadas para isso como JavaScript, CSS e HTML, o que tornou o processo de desenvolvimento da aplicação mais fácil e rápido. O fabricante disponibilizou essa opção na expectativa de atrair desenvolvedores para a plataforma Tizen. As aplicações Web facilitam a programação na plataforma Tizen, e possuem desempenho similar a soluções desenvolvidas nas linguagens nativas do Tizen, C e C++, pois existem interfaces que dão

privilégios às aplicações Web de acessarem conteúdo do sistema operacional, como os sensores.

3.4.1 Comparativo entre os trabalhos correlatos

O Quadro 12 apresenta um comparativo entre as funcionalidades presentes nos trabalhos correlatos e o sistema desenvolvido.

Quadro 12 - Comparativo entre trabalhos correlatos

trabalhos características	FitData (RIGO; RITTER, 2016)	GEOCONFI (OLIVEIRA, 2016)	(SAMPAIO, 2011)	Sistema Desenvolvido
utiliza dispositivo vestível	X	X		X
utilidade médica	acompanhamento	acompanhamento	monitoramento	acompanhamento
histórico de sinais fisiológicos	X	X	X	X
análise de contexto fisiológico		X		

Fonte: elaborado pelo autor.

O sistema FitData, o GEOCONFI e o sistema desenvolvido nesse trabalho possuem utilidade médica de acompanhamento, pois coletam os sinais fisiológicos e os armazenam gerando históricos para acompanhamento pessoal ou de um especialista. A proposta do trabalho de Sampaio (2011) diverge, pois, se trata de um sistema de monitoramento em tempo real com emissão de alertas sobre o estado do paciente.

Em relação aos demais trabalhos, Oliveira (2016) aplica a análise de contexto fisiológico, que além da captura dos sinais de frequência cardíaca, leva em consideração a caracterização da situação do usuário no momento da aferição dos sinais. O sistema desenvolvido nesse trabalho, diferente dos trabalhos correlatos, coleta apenas um tipo de sinal fisiológico, a frequência cardíaca, porem independe de algum tipo de conexão externa para armazenar o histórico de medições pois consegue captar os sinais de frequência cardíaca durante o cotidiano do usuário sem haver necessidade de interação do mesmo, ainda que possibilite a leitura manual da frequência cardíaca quando solicitada pelo usuário.

4 CONCLUSÕES

O objetivo de desenvolver um sistema de monitoramento da frequência cardíaca de pacientes com miocardiopatia dilatada utilizando tecnologia vestível foi atendido adequadamente. O sistema atende aos conceitos básicos que caracterizam dispositivos *wearables*, como a capacidade de computar alguma informação sobre o usuário sem a necessidade de interação direta do mesmo com dispositivo.

O uso da linguagem Java supriu todas as necessidades do desenvolvimento da aplicação móvel, desenvolvida para a plataforma Android. No caso da aplicação *wearable*, foi interessante desenvolver o código do relógio como uma aplicação Web, utilizando as linguagens JavaScript, CSS e HTML, pois o Tizen OS oferece suporte para tal abstração, o que tornou o processo de desenvolvimento da aplicação vestível mais rápida, simples e suprimindo a necessidades do sistema, como acesso ao sensor HRM do relógio.

Em relação às técnicas de desenvolvimento utilizadas, as mesmas foram satisfatórias. Destacando o uso da biblioteca Accessory SDK, que contribuiu sendo uma facilitadora da comunicação entre o relógio e o celular, abstraindo qualquer preocupação com relação a camada de comunicação do sistema desenvolvido.

As ferramentas utilizadas para o desenvolvimento e para a especificação atenderam bem as necessidades que surgiram. Porém a execução de testes durante o desenvolvimento da aplicação *wearable* foram limitadas devido a IDE Tizen Studio possuir um emulador muito limitado e pesado e os testes diretamente no relógio tornaram o processo mais lento e demorado.

O Samsung Gear S2 Classic cumpriu seu objetivo como tecnologia vestível, do qual foi possível extrair as leituras de frequência cardíacas. Porém o Tizen SO se apresentou como sistema operacional limitado, com documentação pouco elaborada e APIs confusas, com poucas explicações e pouco detalhamento. Além disso, o desenvolvimento desse trabalho sofreu atrasos no cronograma devido a atualizações oficiais da fabricante do relógio, que geravam falhas na comunicação entre relógio e o celular.

Entre os pontos positivos da tecnologia Tizen se pode citar: a facilidade de uso do sistema e a aplicação do uso da tecnologia vestível dentro do cenário que esse trabalho se insere. O uso da tecnologia *wearable* se justifica devido a necessidade de especialistas em cardiologia encontram em acompanhar a frequência cardíaca de pacientes portadores de miocardiopatia dilatada por um período de tempo maior, já que os procedimentos tradicionais são limitados e em muitos casos pouco acessíveis.

A partir dos resultados pode-se concluir que embora tenha havido reajustes no cronograma devido a empecilhos no decorrer do desenvolvimento, os objetivos foram alcançados e o sistema como foi desenvolvido se apresenta adequado as necessidades para qual foi desenvolvido.

4.1 EXTENSÕES

As sugestões de extensões para trabalhos futuros são:

- a) utilizar outras plataformas de dispositivos *wearables*;
- b) integrar o sistema a um serviço web;
- c) permitir a captura de outros sinais fisiológicos;
- d) elaborar relatórios dos sinais fisiológicos mais detalhados.

REFERÊNCIAS

- ALBUQUERQUE, Denilson C. et al. Registro Brasileiro de Insuficiência Cardíaca – Aspectos Clínicos, Qualidade Assistencial e Desfechos Hospitalares. **BREATHE**, São Paulo, v. 1, n. 1, 2014. Disponível em: <http://www.scielo.br/pdf/abc/2015nahead/pt_0066-782X-abc-20150031.pdf>. Acesso em: 10 set. 2016.
- BARRETO, Antonio C. P. et al. Insuficiência Cardíaca em Grande Hospital Terciário de São Paulo. In: ARQUIVOS BRASILEIROS DE CARDIOLOGIA, vol. 71 n.1, 1998, São Paulo. **Anais eletrônicos...** São Paulo: SciELO Brasil, 1998. Disponível em:< http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0066-782X1998000700004> . Acesso em: 8 set. 2016.
- CORAÇÃO ALERTA: Holter 24 horas. [S.l.], 2014. Disponível em: <<https://coracaoalerta.com.br/tratamento-do-infarto/holter-24-horas/>>. Acesso em: 10 nov. 2016.
- FOCHESATTO FILHO, L.; BARROS, E. (Org.). **Medicina interna na prática clínica**. Porto Alegre: Artmed, 2013.
- GODINHO, P. M. A. S. **Pulseira inteligente para monitorização de funções vitais**. 2013. 121 f. Dissertação (Mestrado em Engenharia Eletrotécnica e de Computadores) - Curso de Mestrado em Engenharia Eletrotécnica e de Computadores, Instituto Superior de Engenharia do Porto, Porto.
- HOROWITZ, E. S. K. Miocardiopatia Dilatada: Manejo Clínico. **Revista da Sociedade Brasileira de Cardiologia do Rio Grande da Sul**, Porto Alegre, Ano XIII n° 01, abr. 2014. Disponível em:<<http://sociedades.cardiol.br/sbc-rs/revista/2004/01/artigo09.pdf>> . Acesso em: 10 set. 2016.
- Disponível em: <<http://www.ibict.br/cionline/>>. Acesso em: 18 maio 1998.
- HARADA, Eduard. **Tecmundo – Review: smartwatch Samsung Gear S2 [vídeo]**. [S.I.], 2015. Disponível em:< <http://www.tecmundo.com.br/samsung-gear-s2/91209-analise-smartwatch-samsung-gear-s2-video-review.htm>>. Acesso em: 14 set. 2016
- HU, Feiyan; SMEATON, Alan F.; NEWMAN, Eamonn. 2014. Periodicity Detection in Lifelog Data with Missing and Irregularly Sampled Data. **Bioinformatics and Biomedicine**, Belfast, 2014. Disponível em: <<http://dx.doi.org/10.1109/BIBM.2014.6999284>>. Acesso em: 29 out. 2016.
- LAMKIN, Paul. **Best smartwatch 2016: Apple, Pebble, Samsung, Sony, Garmin, Tag and more**. Pernambuco, [S.I.], 2016. Disponível em:< <http://www.wearable.com/smartwatches/the-best-smartwatches-in-the-world>>. Acesso em: 2 nov. 2016.
- LIMA, A. L. **O que altera e como normalizar a frequência cardíaca**. Pernambuco, [2016]. Disponível em:< <https://www.tuasaude.com/frequencia-cardiaca/>>. Acesso em: 2 nov. 2016.
- MALOFF, J. A internet e o valor da "internetização". **Ciência da Informação**, Brasília, v. 26, n. 3, 1997.
- OLIVEIRA, G. A. **Um Modelo Para Gerenciamento De Histórico De Contextos Fisiológicos**. 2016. 71 f. Dissertação (Mestrado em Pós-Graduação em Computação Aplicada) - Mestrado em Pós-Graduação em Computação Aplicada, Universidade do Vale do Rio dos Sinos, São Leopoldo.

OLIVEIRA, P. **Tecnologias e afetos: pulse sensor** [2015]. Disponível em: <<http://www.discombobulate.me/workshop/arquetipos/plano-de-aula-arquetipos/sensores/pulse-sensor/>>. Acesso em: 31 out. 2016.

PARAMESHAWAR et al. apud BARRETO, Antonio C. P. et al. Insuficiência Cardíaca em Grande Hospital Terciário de São Paulo. In: ARQUIVOS BRASILEIROS DE CARDIOLOGIA, vol. 71 n.1, 1998, São Paulo. **Anais eletrônicos...** São Paulo: SciELO Brasil, 1998. Disponível em:<http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0066-782X1998000700004> . Acesso em: 8 set. 2016.

RAMINHOS, J. P. B. D. V. **Aquisição de Sinais Fisiológicos: Aplicação ao controlo de uma plataforma móvel a partir do EOG.** 2009. 106 f. Dissertação (Mestrado em Engenharia Electrotécnica e de Computadores) - Mestrado em Engenharia Electrotécnica e de Computadores, Universidade Técnica de Lisboa, Lisboa.

RIGO, Sandro J.; RITTER, Eduardo W. FITDATA: Um sistema para monitoramento de atividade física baseado em dispositivos móveis. In: XII BRAZILIAN SYMPOSIUM ON INFORMATION SYSTEMS. 2016, Florianópolis. **Anais eletrônicos...** Minas Gerais: UFMG, 2016. Disponível em:<<http://www.lbd.dcc.ufmg.br/colecoes/sbsi/2016/072.pdf>> Acesso em: 12 set. 2016.

SAMSUNG: **Smartwatch Gear S2 Classic.** [S.I.], 2016. Disponível em: <<http://www.samsung.com/br/consumer/mobile-devices/wearables/gear-s/SM-R7320ZKAZTO>>. Acesso em: 18 set. 2016.

SAMPAIO, Í. C. **Sistema de Monitoramento Remoto de Pacientes Implementado em Hardware de Arquitetura ARM.** 2011. 70 f. Monografia (Bacharelado em Engenharia de Teleinformática) – Curso de Graduação em Engenharia de Teleinformática, Universidade Federal do Ceará, Ceará.

SBC. **Cardiômetro - Mortes por doenças cardiovasculares no Brasil.** [S.I.], 2015. Disponível em: <<http://www.cardiometro.com.br>>. Acesso em: 30 ago. 2016.

SILVEIRA, D. P. **Tudo o que você precisa saber sobre o Tizen, SO da Samsung.** [S.I.], 2016. Disponível em:<<https://www.oficinadanet.com.br/post/16719-tudo-o-que-voce-precisa-saber-sobre-o-tizen-so-da-samsung>>. Acesso em: 10 nov. 2016.

SUMRELL, Mariano. **Canaltech – Os Wearables são a evolução da tecnologia móvel.** [S.I.], 2016. Disponível em:<<http://canaltech.com.br/coluna/mobile/os-wearables-sao-a-evolucao-da-tecnologia-movel/>>. Acesso em: 12 set. 2016.

SUS apud BARRETO, Antonio C. P. et al. Insuficiência Cardíaca em Grande Hospital Terciário de São Paulo. In: ARQUIVOS BRASILEIROS DE CARDIOLOGIA, vol. 71 n.1, 1998, São Paulo. **Anais eletrônicos...** São Paulo: SciELO Brasil, 1998. Disponível em:<http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0066-782X1998000700004> . Acesso em: 8 set. 2016.

THOMAZ, M. **4 tendências de tecnologia na área da saúde em 2016.** São Paulo, [2016]. Disponível em: <<http://blog.iclinic.com.br/4-tendencias-de-tecnologia-na-area-da-saude-em-2016/>>. Acesso em: 15 set. 2016.

VALDIGEM, B. **Minha Vida - Miocardiopatia dilatada: sintomas, tratamentos e causas.** [S.I.], 2014. Disponível em: <<http://www.minhavidacom.br/saude/temas/miocardiopatia-dilatada>>. Acesso em: 11 set. 2016.

WEISE, M. The Computer for the 21st Century. **Scientific American**, v. 265 n. 3, p. 94-104, 1991.

WHO. **The top 10 causes of death**. [S.l.], 2014. Disponível em: < <http://www.who.int/mediacentre/factsheets/fs310/en/>>. Acesso em: 15 set. 2016.