

**UNIVERSIDADE DO EXTREMO SUL CATARINENSE - UNESC**

**CURSO DE CIÊNCIA DA COMPUTAÇÃO**

**JULIANA CHAVES BRAZ DA SILVA**

**COMUNICAÇÃO EM TEMPO REAL COM WEBSOCKET APLICADO AO  
DESENVOLVIMENTO DE UM PROTÓTIPO PARA PRESCRIÇÃO DE TREINO DE  
MUSCULAÇÃO PARA DISPOSITIVOS MÓVEIS ANDROID**

**CRICIÚMA**

**2015**

**JULIANA CHAVES BRAZ DA SILVA**

**COMUNICAÇÃO EM TEMPO REAL COM WEBSOCKET APLICADO AO  
DESENVOLVIMENTO DE UM PROTÓTIPO PARA PRESCRIÇÃO DE TREINO DE  
MUSCULAÇÃO PARA DISPOSITIVOS MÓVEIS ANDROID**

Trabalho de Conclusão de Curso, apresentado para obtenção do grau de Bacharel no curso de Ciência da Computação da Universidade do Extremo Sul Catarinense, UNESC.

Orientador: Prof. Esp. William Bertan da Silva

**CRICIÚMA**

**2015**

**JULIANA CHAVES BRAZ DA SILVA**

**COMUNICAÇÃO EM TEMPO REAL COM WEBSOCKET APLICADO AO  
DESENVOLVIMENTO DE UM PROTÓTIPO PARA PRESCRIÇÃO DE TREINO DE  
MUSCULAÇÃO PARA DISPOSITIVOS MÓVEIS ANDROID**

Trabalho de Conclusão de Curso aprovado pela Banca Examinadora para obtenção do Grau de Bacharel, no Curso de Ciência da Computação da Universidade do Extremo Sul Catarinense, UNESC, com Linha de Pesquisa em Computação Móvel.

Criciúma, 24 de novembro de 2015

**BANCA EXAMINADORA**



Prof. William Bertan da Silva - Esp. - UNESC - Orientador



Prof. Luciano Antunes - MSc. - UNESC



Prof. Fabrício Giordani - Esp. - UNESC

**À Deus e meus pais pela oportunidade e força para a realização deste trabalho.**

## **AGRADECIMENTOS**

Agradeço primeiramente a Deus por ter suprido com tudo que precisei durante o período de realização deste projeto, pela saúde, paciência, força e ânimo nos momentos difíceis.

Aos meus pais, Silvia Cristina de Oliveira Chaves e Milton Braz da Silva, pela oportunidade de estudo, pelo apoio, carinho, confiança, paciência, pelas palavras de conforto e por esta conquista que não seria possível atingir sozinha, vocês são a minha inspiração.

Ao professor Fabricio Giordani que acreditou neste projeto e iniciou como orientador, obrigada pela confiança e paciência durante o período como orientador.

Ao professor e amigo William Bertan da Silva que seguiu como orientador, sempre muito paciente e atencioso, me auxiliando, incentivando, contribuindo e apoiando durante o desenvolvimento e conclusão do projeto.

Ao meu noivo Renato Felicio dos Santos que teve paciência, confiança e me incentivou a concluir o projeto, mesmo em momentos difíceis em que pensei em desistir.

Sou grata a todos os colegas e amigos que me ajudaram e contribuíram de alguma forma com a minha graduação, em especial à minha grande amiga Camila de Farias Crispim que sempre me apoiou, ajudou e incentivou de todas as formas.

E a todos os professores que independente da fase e da matéria ministrada contribuíram com o conhecimento adquirido durante esta fase importante da minha vida.

**“Feliz o homem que acha sabedoria,  
e o homem que adquire conhecimento;  
porque melhor é o lucro que ela dá  
do que o da prata”.**

**Provérbios 3:13-14**

## RESUMO

A tecnologia e uso dos dispositivos móveis estão cada vez mais presentes na cultura popular, espalhados por todo o mundo. Baseado neste fato desenvolvedores buscam criar aplicações com variadas funções para disponibilizar nos dispositivos aplicativos para suprir uma necessidade de forma mais automatizada, além de disponibilizar em vários locais, visto que aparelhos como *tablets* e *smartphones* estão disponíveis em diversas faixas de preços para alcançar um número maior de pessoas. Os dispositivos móveis apresentam um bom funcionamento em relação ao seu tamanho, podendo ser levado a qualquer lugar, permitindo o uso de conexões com a Internet, armazenamento de dados, troca de informações, imagens, além de aplicativos móveis desenvolvidos para o uso nestes aparelhos. Os aplicativos desenvolvidos envolvem diversas áreas como saúde, moda, educação, *fitness*, entre outros. Diante disto, e da maneira atual em que são prescritos os treinos de musculação nas academias que consiste em descrever o treino em uma ficha de papel ou utilizar uma impressora térmica para a impressão do treino que contém o nome do exercício, série, repetição e alguns casos peso ou carga, foi desenvolvido o projeto, que consiste em integrar as áreas de Ciência da Computação com a Educação Física, a fim de apresentar uma solução para a prescrição de treinos realizadas em fichas de papel, que foi a criação de um protótipo de aplicativo *mobile* para dispositivos com o sistema operacional Android, aplicando uma tecnologia chamada WebSocket para a integração e comunicação em tempo real entre duas aplicações, uma *web* e a outra *mobile*, com o objetivo de automatizar a prescrição dos treinos de musculação em academias, buscando facilitar este processo de prescrição e consulta dos treinos de musculação. O desenvolvimento das duas aplicações se deu através do uso dos ambientes de desenvolvimento IDE Netbeans e Android Studio com a linguagem de programação Java, utilização do banco de dados MySQL e SQLite, além das ferramentas Apache Tomcat, Json e o protocolo WebSocket. Por fim, foram aplicados testes em ambiente real de quatro academias de musculação na cidade de Criciúma, Santa Catarina, a fim de validar o uso das aplicações no contexto real e também validar o uso da comunicação em tempo real através do WebSoccket além de verificar o comportamento do aplicativo nestes cenários.

**Palavras-chave:** Android, WebSocket, Aplicativos móveis, Prescrição de treinos, Testes em ambiente real.

## ABSTRACT

The technology and use of mobile devices are increasingly present in popular culture, spread throughout the world. Based on this fact developers looking to create applications with varied functions to provide the application devices to meet a need for more automated, and available at various locations, as devices such as tablets and smartphones are available in various price ranges to reach more of people. Mobile devices present a good run in relation to its size and can be carried anywhere, allowing the use of Internet connections, data storage, exchange of information, images, and mobile applications developed for use in these devices. Developed applications involve various areas such as health, fashion, education, fitness, and more. Given this, and the current way in which weight training in gyms are prescribed which is to describe the training in a paper form or use a thermal printer for training the impression that contains the name of the exercise, series, repetition and some cases weight or load, the project was developed, which consists in integrating the areas of Computer Science with Physical Education in order to present a solution for prescribing training held in paper records, which was the creation of a prototype mobile application for devices with the Android operating system by applying a WebSocket technology called for integration and real-time communication between two applications, a web and other mobile, in order to automate the prescription of weight training in gyms, seeking to facilitate this process of consultation and prescription of weight training. The development of the two applications was through the use of Netbeans IDE development environments and Android Studio with the Java programming language, database use MySQL and SQLite, in addition to Apache Tomcat tools, JSON and the WebSocket protocol. Finally, in a real environment tests were applied four bodybuilding academies in the city of Criciuma, Santa Catarina, in order to validate the use of applications in the real context and also validate the use of real-time communication through WebSoccket and check the application behavior in these scenarios.

**Keywords: Android, WebSocket, Mobile Applications, Training Prescription, Tests in a real environment.**

## LISTA DE ILUSTRAÇÕES

Figura 1 – Qualidades físicas .....	14
Figura 2 – Curva de carga <i>versus</i> número de repetições.....	16
Figura 3 – Programas básicos.....	18
Figura 4 – Estrutura do Android. ....	23
Figura 5 – Emulador do Android.....	25
Figura 6 – Ciclo de vida de uma Activity. ....	27
Figura 7 – Como um navegador interage com um servidor web para fazer requisições. ....	32
Figura 8 – Tela principal do Tomcat Manager.....	33
Figura 9 – Polling versus WebSocket.....	36
Figura 10 – Esquema do funcionamento WebSocket. ....	37
Figura 11 – Diagrama de caso de uso da aplicação web.....	47
Figura 12 – Diagrama de comunicação entre as aplicações.....	48
Figura 13 – Modelo lógico da aplicação. ....	49
Figura 14 – Console do desenvolvedor da Google Play. ....	51
Figura 15 – Pacotes das classes no NetBeans.....	52
Figura 16 – Bibliotecas aplicação <i>web</i> . ....	53
Figura 17 – Uso do <i>framework</i> Foundation. ....	53
Figura 18 – Tela inicial aplicação <i>web</i> .....	54
Figura 19 – Telas de cadastro de usuário e exercício.....	54
Figura 20 – Classe WebSocket do projeto <i>web</i> .....	56
Figura 21 – Diagrama de atividade do aplicativo PersonalFit.....	57
Figura 22 – Estrutura dos pacotes aplicativo PersonalFit. ....	58
Figura 23 – Trecho do código-fonte classe WebSocketManager do aplicativo PersonalFit. ....	59
Figura 24 – Detalhes do aplicativo PersonalFit no console de desenvolvedor.....	60
Figura 25 – Sequência de telas do aplicativo PersonalFit.....	61
Figura 26 – Ficha de prescrição do treino. ....	63
Figura 27 – Tela de cadastro de treino.....	64
Figura 28 – Informações da tabela <i>websocketlog</i> do banco de dados.....	66

Figura 29 – Gráfico das instalações ativas do aplicativo PersonalFit.....	67
Figura 30 – Gráfico de instalações atuais por versão do Android. ....	67
Figura 31 – Gráfico de instalações atuais por tipos de dispositivos. ....	68
Figura 32 – Gráfico de usuários ativos no aplicativo por tipo de rede. ....	69
Figura 33 – Gráfico da soma de usuários por modelos dos dispositivos.....	70
Figura 34 – Gráfico dos erros de conexão por usuários.....	71
Figura 35 – Gráfico da média do tempo de conexão por tipo de rede.....	72
Figura 36 – Gráfico da média do tempo de conexão do usuário mais ativo. ....	72

## LISTA DE TABELAS

Tabela 1 – Parâmetros fisiológicos do treinamento muscular. ....	18
Tabela 2 – Diretrizes sugeridas para o Intervalo de Descanso (ID) entre as séries para várias cargas e suas circunstâncias aplicáveis. ....	19

## LISTA DE ABREVIATURAS E SIGLAS

1RM	Uma Repetição Máxima
ADT	<i>Android Development Tools</i>
AVD	<i>Android Virtual Device</i>
AJAX	<i>Asynchronous Javascript and XML</i>
API	<i>Application Programming Interface</i>
CSS	<i>Cascading Style Sheets</i>
GPS	<i>Global Positioning System</i>
HTML	<i>Hypertext Markup Language</i>
HTTP	<i>Hypertext Transfer Protocol</i>
HTTPS	<i>Hypertext Transfer Protocol Secure</i>
IDE	<i>Integrated Development Environment</i>
IETF	<i>Internet Engineering Task Force</i>
IP	<i>Internet Protocol</i>
JDBC	<i>Java Database Connectivity</i>
JEE	<i>Java Enterprise Edition</i>
JSF	<i>JavaServer Faces</i>
JSON	<i>JavaScript Object Notation</i>
JSP	<i>JavaServer Pages</i>
MVC	<i>Model View Controller</i>
OHA	<i>Open Handset Alliance</i>
RML	Resistência Muscular Localizada
SDK	<i>Software Development Kit</i>
SMS	<i>Short Message Service</i>
SSL	<i>Secure Sockets Layer</i>
TCP	<i>Transmission Control Protocol</i>
URL	<i>Uniform Resource Locator</i>
USB	<i>Universal Serial Bus</i>
UTF-8	<i>8-bit Unicode Transformation Format</i>
VM	<i>Virtual Machine</i>
XML	<i>eXtensible Markup Language</i>
W3C	<i>World Wide Web Consortium</i>

## SUMÁRIO

<b>1 INTRODUÇÃO</b> .....	<b>7</b>
1.1 OBJETIVO GERAL .....	8
1.2 OBJETIVOS ESPECÍFICOS .....	8
1.3 JUSTIFICATIVA .....	9
1.4 ESTRUTURA DO TRABALHO .....	11
<b>2 EXERCÍCIO FÍSICO</b> .....	<b>13</b>
2.1 IMPORTÂNCIAS DO EXERCÍCIO FÍSICO .....	13
2.2 MUSCULAÇÃO .....	15
2.3 METODOLOGIAS DE PRESCRIÇÃO DE TREINO PARA MUSCULAÇÃO .....	16
<b>3 ANDROID</b> .....	<b>21</b>
3.1 DESENVOLVIMENTO ANDROID .....	24
3.2 TECNOLOGIAS ENVOLVIDAS NO DESENVOLVIMENTO ANDROID .....	29
<b>4 APLICAÇÃO WEB</b> .....	<b>32</b>
4.1 TECNOLOGIAS APLICADAS EM DESENVOLVIMENTO <i>WEB</i> .....	32
<b>4.1.1 Java Enterprise Edition</b> .....	<b>32</b>
<b>4.1.2 MySQL</b> .....	<b>33</b>
4.2 SINCRONIZAÇÃO EM TEMPO REAL .....	34
<b>4.2.1 WebSocket</b> .....	<b>35</b>
<b>4.2.2 Funcionamento do WebSocket</b> .....	<b>37</b>
<b>5 TRABALHOS CORRELATOS</b> .....	<b>39</b>
5.1 A UTILIZAÇÃO DA LINGUAGEM SWIFT EM DISPOSITIVOS MÓVEIS BASEADOS NO SISTEMA OPERACIONAL IOS PARA AVALIAÇÃO DA COMPOSIÇÃO CORPORAL .....	39
5.2 PROTÓTIPO DE APLICATIVO COM A UTILIZAÇÃO DA METODOLOGIA ÁGIL SCRUM E DO CONCEITO KANBAN PARA ACOMPANHAMENTO DO DISPÊNDIO ENERGÉTICO EM ATIVIDADES FÍSICAS .....	40
5.3 WEBSOCKET PARA APLICAÇÕES WEB EM ERLANG .....	41
5.4 SIMULAÇÃO DE FUTEBOL EM AMBIENTE WEB: VERSÃO 3.0 .....	42
5.5 UM CHAT PICTOGRÁFICO PARA O SCALA: SISTEMA DE COMUNICAÇÃO ALTERNATIVA PARA O LETRAMENTO DE PESSOAS COM AUTISMO .....	43

5.6 IMPLEMENTAÇÃO DE UM SERVIDOR DE NEGOCIAÇÃO EM BOLSA BASEADO EM WEBSOCKET.....	44
<b>6 COMUNICAÇÃO EM TEMPO REAL COM WEBSOCKET APLICADO AO DESENVOLVIMENTO DO PROTÓTIPO PERSONALFIT PARA PRESCRIÇÃO DE TREINO DE MUSCULAÇÃO PARA DISPOSITIVOS MÓVEIS ANDROID .....</b>	<b>45</b>
6.1 METODOLOGIA.....	46
6.1.1 Levantamento bibliográfico.....	46
6.1.2 Requisitos e funcionamento das aplicações .....	46
6.1.3 Modelagem do banco de dados .....	49
6.1.4 Definição das ferramentas e recursos.....	50
6.1.5 Desenvolvimento web.....	52
6.1.6 Desenvolvimento Android .....	56
6.2 TESTES EM UM CENÁRIO REAL DE ACADEMIA DE MUSCULAÇÃO .....	62
6.3 RESULTADOS OBTIDOS .....	65
6.3.1 Sugestões de melhorias .....	73
<b>7 CONCLUSÃO .....</b>	<b>74</b>
<b>REFERÊNCIAS.....</b>	<b>77</b>

## 1 INTRODUÇÃO

Com a correria do dia-a-dia e todos os seus afazeres é difícil manter uma rotina saudável, com exercícios físicos e uma boa alimentação. Com obstáculos para uma vida saudável, hábitos como a má alimentação e estresse exagerado tornam-se prejudiciais à saúde e trazem doenças como o sedentarismo, obesidade, hipertensão arterial sistêmica e outros distúrbios (BÜNDCHEN et al, 2013).

Uma maneira simples e não farmacêutica para evitar algumas destas doenças é a prática regular de exercícios físicos. Em contrapartida com a evolução dos dispositivos móveis e seu grande uso no dia-a-dia, muitos acabam utilizando-os de maneira exagerada e com isto acabam gastando mais tempo com as tecnologias disponíveis do que com uma atividade física (BÜNDCHEN et al, 2013).

O mercado de dispositivos móveis, em especial celulares, está expandindo cada vez mais e com ele a mobilidade, que tem como objetivo permitir o acesso a dados, internet, e-mail, agenda, mensagens e outras aplicações de qualquer lugar, que por sua vez facilita a troca de informações sem precisar estar em uma estação fixa, como um computador (LECHETA, 2013).

Segundo Deitel et al (2012, tradução nossa) devido ao uso crescente dos dispositivos móveis aumenta também o interesse pelo desenvolvimento mobile, em especial o Android, baseado no grande número de aplicativos disponíveis para *download* na *Play Store*, também conhecida como *Google Play*, loja de aplicativos disponível em aparelhos com o sistema Android, responsável por ter o maior número de aplicativos disponíveis para dispositivos móveis. Em setembro de 2014 foi apontado que cerca de 52,1% dos aplicativos para dispositivos móveis são aplicações Android, seguidos de 41,7% da Apple, 3,6% Microsoft, 2,3% BlackBerry e por fim 0,2% Symbian (COMSCORE, 2014).

Deitel et al (2012, tradução nossa) aponta que um dos grandes benefícios no desenvolvimento de aplicativos Android é a plataforma de código aberto. Com o sistema de código aberto e livre é permitido ver o código fonte e como suas características são implementadas o que torna o desenvolvimento dos aplicativos mais atrativos além da disponibilidade para a publicação na Internet. Entretanto para uma boa interação entre usuários e o aplicativo, este deve ter uma resposta rápida quando conectada a um servidor web. Para o desenvolvimento de aplicações que se

comuniquem em tempo real é utilizada uma tecnologia chamada WebSocket (WANG; SALIM; MOSKOVITS, 2013, tradução nossa).

Desta forma este projeto consiste em aplicar a tecnologia WebSocket em um ambiente real, através de uma aplicação para atender a necessidade da prescrição de treino de musculação de uma forma mais ágil, contando com uma aplicação *web* para que o instrutor insira os treinos de seus alunos, que por sua vez o recebem em seus dispositivos móveis com o sistema Android através de um aplicativo, podendo assim listar seu treino, marcar quais exercícios já foram feitos e visualizar quais ainda faltam para concluir o treino, através de uma conexão em tempo real feita pelo WebSocket. E, por fim validar o uso das duas aplicações disponibilizando-as para testes em um ambiente real de academia de musculação na cidade de Criciúma, Santa Catarina.

## 1.1 OBJETIVO GERAL

Aplicar a comunicação em tempo real por meio do WebSocket em um protótipo de prescrição de treino de musculação para dispositivos móveis com sistema operacional Android.

## 1.2 OBJETIVOS ESPECÍFICOS

Os objetivos específicos deste projeto consistem em:

- a) compreender a metodologia utilizada na prescrição de treinos de musculação;
- b) compreender o desenvolvimento de uma aplicação para dispositivos móveis com sistema operacional Android;
- c) descrever e utilizar a tecnologia do WebSocket para comunicação de aplicações em tempo real;
- d) desenvolver um protótipo *mobile* para prescrição de treino de musculação em dispositivos com sistema operacional Android e sincronização em tempo real com a aplicação *web*;
- e) realizar os testes do protótipo desenvolvido em um cenário real de academia de musculação.

### 1.3 JUSTIFICATIVA

Clarinda (2009) afirma que atualmente um número crescente de indivíduos ingressaram em programas sistematizados de atividades físicas buscando uma melhor composição corporal e, conseqüentemente, uma melhor qualidade de vida. Para que isso seja possível de maneira segura e eficaz são utilizados diferentes métodos de treinamento, em sua grande maioria disponibilizada nas literaturas atuais, o que favorece os praticantes a terem um melhor rendimento frente à prática desta modalidade (WEINHOLD, 2009).

O exercício físico, caracterizado por repetição sistemática e regular, promove no organismo alterações morfológicas, metabólicas, nervosas, hormonais e celulares, contribuindo para uma otimização das funções fisiológicas dos sistemas e órgãos. Contudo, a resposta do organismo frente ao exercício é dependente da natureza, intensidade e duração do estímulo (BARBANTI, 1996). Segundo Guimarães (2010, p. 8) “a musculação na atualidade é uma das práticas esportivas que mais cresce em número de praticantes”.

Os benefícios promovidos pela musculação dependem da manipulação de variáveis do treinamento dentre as quais estão: intensidade, frequência e volume de treinamento (FONSECA, 2013). Para a prescrição do treinamento na musculação o profissional deve ter uma base sólida e alguns conhecimentos científicos como: fisiologia, teoria do treinamento, pedagogia do esporte, cinesiologia<sup>1</sup> e aprendizagem motora (CHAGAS; LIMA, 2004).

Para Fonseca (2013), o número de séries, repetições e exercícios são utilizados para quantificar o volume de treinamento. Tendo o aluno um número determinado dessas variáveis, pode-se adequar a carga e, conseqüentemente, a intensidade do treinamento. Vale lembrar que não se pode generalizar já que não existem valores definitivos de carga para determinada faixa de séries e repetições, pois indivíduos diferentes que realizam um mesmo exercício numa mesma faixa de repetições podem ser submetidos a intensidades também diferentes.

Sobre o desenvolvimento tecnológico, Lecheta (2013) diz que o mercado de celulares está crescendo cada vez mais. Estudos mostram que hoje em dia mais

---

<sup>1</sup> Estudo mecânico dos movimentos do corpo humano (CINESIOLOGIA, 2013).

de 3 bilhões de pessoas possuem um aparelho celular, e isso corresponde a mais ou menos metade da população mundial.

Os dispositivos móveis estão em um nível tecnológico que inclui telas gráficas coloridas com boa resolução, acesso à Internet em alta velocidade, boa capacidade de processamento local e um custo acessível (PEROZZO, 2007).

Segundo Lecheta (2013), aplicativos desenvolvidos para dispositivos móveis, podem estar conectados online para sincronizar informações com o servidor.

Lecheta (2013), ainda afirma que o Android é a primeira plataforma para aplicações móveis completamente livre e de código aberto (*Open Source*), sendo possível utilizar o sistema operacional do Android em seus celulares sem ter que pagar por isso, o que representa uma grande vantagem.

A escolha de desenvolvimento para dispositivos móveis, veio da disseminação da área *mobile* (*smartphones* e *tablets*) conforme citado anteriormente por Lecheta. Um fator importante na escolha da plataforma é o valor a ser pago para a utilização da mesma, a opção escolhida não necessita de taxas para o desenvolvimento e nem para sua utilização. E, conforme o site<sup>2</sup> oficial, o Android oferece uma plataforma de nível mundial para a criação de aplicativos e jogos com usuários em todos os lugares, bem como um mercado de distribuição aberta e instantânea. Além de possuir compatibilidade com a linguagem Java e outras ferramentas para desenvolvimento distribuídas gratuitamente.

Existem mais de uma plataforma disponível para aparelhos móveis, tais como: Android, iOS, Windows Phone, Blackerry e outros. A escolha para o desenvolvimento do protótipo de prescrição do treino de musculação é o Android, que por sua vez é um sistema operacional *Open Source*, ou seja, de código aberto, projetado para dispositivos móveis. O processo de criação de aplicativos para os dispositivos móveis é possível através de um ambiente de desenvolvimento disponibilizado de forma gratuita, além dos recursos como *debug*, que permite encontrar falhas no código de forma mais rápida (GARGENTA; NAKAMURA, 2014, tradução nossa).

Para aplicar e validar o uso da tecnologia WebSocket, cujo objetivo é obter uma conexão de modo que a aplicação e o servidor obtenham uma troca de

---

<sup>2</sup> ANDROID. Disponível em:<developer.android.com>. Acesso em: 27 abr.2014

dados em tempo real, e também pensando em agregar conhecimentos de áreas distintas, no caso a computação e educação física, foi pensado em criar um protótipo de um aplicativo *mobile* para a aplicação do protocolo WebSocket em que seu contexto seja baseado em uma necessidade dentro da área da educação física, no caso, prescrição de treinos de musculação. Visto que a atual forma de consulta dos exercícios a serem feitos dentro de uma academia de musculação são através anotações em fichas de papel com as informações do praticante de musculação e os exercícios que compõem o treino, a série, o número de repetições e a carga (quando houver), a aplicação proposta visa a modernização do esquema de prescrição e consultas de treinos.

Os recursos utilizados para o desenvolvimento deste projeto são os seguintes: o ambiente de desenvolvimento, NetBeans IDE 8.0.2 para a aplicação *web*, Apache Tomcat, banco de dados MySQL e a linguagem Java para o desenvolvimento da aplicação, já para o aplicativo Android utiliza-se o Android Studio em conjunto com o Software Development Kit (SDK), ou seja um kit de desenvolvimento de software ou aplicativos, que permite o desenvolvimento e também a simulação da tela do dispositivo móvel no computador, além do SQLite para armazenar temporariamente poucas informações.

#### 1.4 ESTRUTURA DO TRABALHO

Este trabalho de conclusão de curso é dividido em capítulos de uma maneira que facilite a leitura e a compreensão de todo o conteúdo, esta divisão é feita da seguinte forma: o primeiro capítulo foi utilizado para a introdução, justificativa, objetivos gerais e específico além da estrutura do trabalho.

No segundo capítulo são abordados assuntos relacionados à atividade física até chegar à musculação que é o tópico de maior ênfase da área. O terceiro capítulo entra na área da computação com a plataforma Android, sua estrutura, recursos e ferramentas. O quatro trata-se de uma estrutura *web*, a conexão de uma aplicação *web* com o Android, as tecnologias envolvidas e protocolos de sincronização em tempo real. Os trabalhos correlatos são abordados no sexto capítulo, que contém trabalhos que envolvem as áreas deste trabalho. O sétimo capítulo é composto pelo trabalho desenvolvido, à metodologia utilizada, os recursos

necessários para o seu desenvolvimento, os testes aplicados em ambiente real e seus resultados, além da discussão dos resultados. Por fim, o capítulo de conclusão do projeto com o fechamento do trabalho.

## 2 EXERCÍCIO FÍSICO

O exercício físico é caracterizado por uma atividade com repetições sistemáticas de movimentos orientados, que por sua vez faz aumentar o consumo de oxigênio devido ao esforço exigido do músculo. Um dos objetivos da prática deste tipo de exercício é a melhora do condicionamento físico (MONTEIRO; SOBRAL FILHO, 2004).

Em geral as pessoas que procuram uma atividade para a prática regular de exercícios, estão em busca de um objetivo específico que pode variar entre a estética, condicionamento físico ou até mesmo melhorias na saúde. Os exercícios prescritos no treino mudam de acordo com o objetivo do indivíduo (ABAD et al, 2010).

Para que seja possível alcançar os objetivos do exercício proposto, deve-se seguir uma estrutura programada, repetitiva e regular. Os seguintes pontos também devem ser observados para a prescrição de um treino: frequência dos exercícios (quantidade de vezes executados por semana), intensidade (ritmo de execução e/ou carga) e duração (tempo de execução) (BARBANTI, 2012).

Para a prescrição de exercícios físicos independente do sexo, deve-se considerar o condicionamento cardiorrespiratório, força muscular, composição corporal e flexibilidade (LEITÃO, 2000).

### 2.1 IMPORTÂNCIAS DO EXERCÍCIO FÍSICO

Ferreira (2001) afirma que nos dias de hoje é aceito o conceito de que o exercício físico regular, quando orientado de forma correta por um profissional, traz diversos benefícios, tais como o fortalecimento muscular para enfrentar as situações cotidianas e movimentos repetitivos como puxar, empurrar, subir, fazer levantamentos ou ficar em pé sem fadigar e nem causar lesões devido ao enfraquecimento muscular. Aumento da massa magra, definição corporal e diminuição da gordura corporal e elevação do metabolismo (WILLIAMS; GROVES; THURGOOD, 2010).

A aplicação de exercícios físicos na rotina do ser humano implica em aspectos positivos para a saúde do mesmo. Não somente benefícios físicos, mas

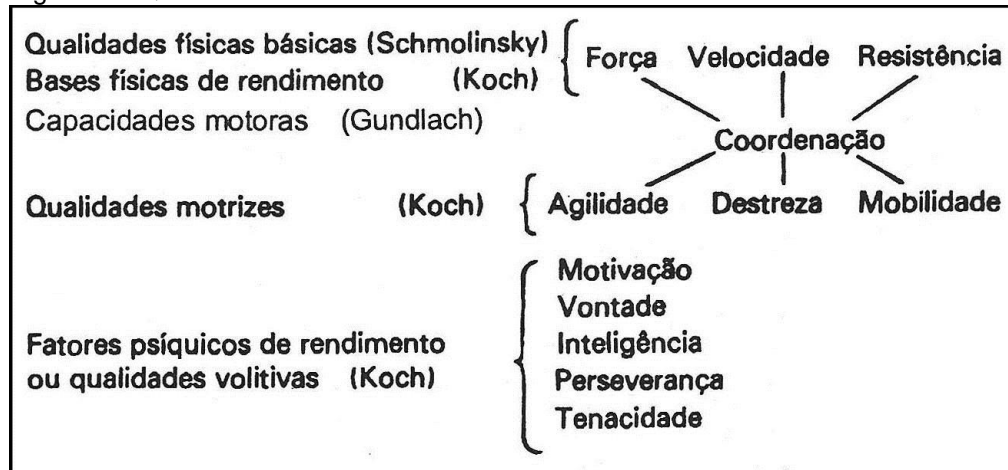
também em questões como o sono e seus possíveis distúrbios, transtornos de humor, depressão, ansiedade, além dos pontos cognitivos, como a memória e a aprendizagem (MELLO et al, 2005).

São comprovados que os benefícios da prática de exercícios físicos atingem ambos os sexos e que a prática de exercícios regularmente combinado a uma alimentação adequada se faz importante para o ganho de massa óssea e previne a osteoporose pós-menopáusia. A rotina de exercícios também causa alívios de sintomas pré-menstruais nas mulheres (LEITÃO et al, 2000).

A prática regular de exercícios físicos torna o indivíduo mais ativo e apto a realizar um número maior de atividades devido a uma maior resistência adquirida com a prática de atividade, o que faz com que ele conseqüentemente tenha um risco menor de ter doenças cardiovasculares e até mesmo a obesidade (PONTES; SOUSA; NAVARRO, 2009).

É possível ilustrar as qualidades obtidas pelo exercício físico através da figura 1. Estas qualidades: força, velocidade, resistência, agilidade, destreza e mobilidade, estão todas conectadas por meio da coordenação ao executar os movimentos (BARBANTI, 1997).

Figura 1 – Qualidades físicas



Fonte: Barbanti (1997).

Câmara et al (2007) afirma que exercícios resistidos consistem na utilização de alguma forma externa para realizar contrações em grupos musculares específicos, tais como máquinas, pesos, cordas e elásticos. Este método de exercícios auxilia na manutenção e ganho de massa muscular, na densidade óssea,

fortalecem os músculos e ossos, conhecidos na prática de musculação. Chaves et al (2007) define os exercícios aeróbicos como aqueles que são executados de forma contínua utilizando o oxigênio como fonte principal, como exemplo correr, caminhar, nadar, andar de bicicleta e jogar futebol, geralmente são recomendados por no mínimo 3 vezes por semana em uma sessão de 20 minutos. Exercícios do tipo aeróbico contribuem para a eliminação de peso, gordura corporal, ajudam na capacidade cardiorrespiratória fortalecendo o coração e os pulmões (CIVINSKI; MONTIBELLER; BRAZ, 2011). Uma das classificações dos exercícios aeróbicos é o impacto, podendo ser baixo ou alto. Pode-se citar atividades aeróbicas de baixo impacto, como: a caminhada, ciclismo ou pedalar na bicicleta, natação, a hidroginástica, remo, subir escadas, dançar, ioga e dança aeróbica. E atividades de alto impacto, como: corrida, esportes que envolvam saltos, vôlei e basquete, pular corda e dança aeróbica de alto impacto (MATSUDO; MATSUDO; BARROS NETO, 2001).

## 2.2 MUSCULAÇÃO

A musculação pode ser definida como um tipo de treinamento físico, que se resume em um conjunto de exercícios com carga, onde o aumento dos pesos são realizados de forma progressiva (BARBANTI, 1997).

Atualmente a musculação é uma das atividades mais procuradas e pesquisadas em diversos países, são muitos os motivos que atraem as pessoas para ela, tais como: ganho de força para a manutenção da saúde, treinamento de atletas e até aumento da massa muscular (FERREIRA et al, 2008).

É importante trabalhar com um profissional formado em alguma área de ciência da saúde, como o *personal trainer* ou instrutor de musculação com conhecimento do corpo humano, fisiologia e compreensão dos métodos de treinamento, o que pode trazer um resultado melhor e mais rápido (AABERG, 2001). Junto a isso, é necessário que o profissional da saúde tenha conhecimento das definições, princípios e métodos de desenvolvimento da modalidade, para que oriente a prescrição, organização e periodização de um treinamento com exercícios seguros e de forma correta (MUSSI; DIAS, 2005).

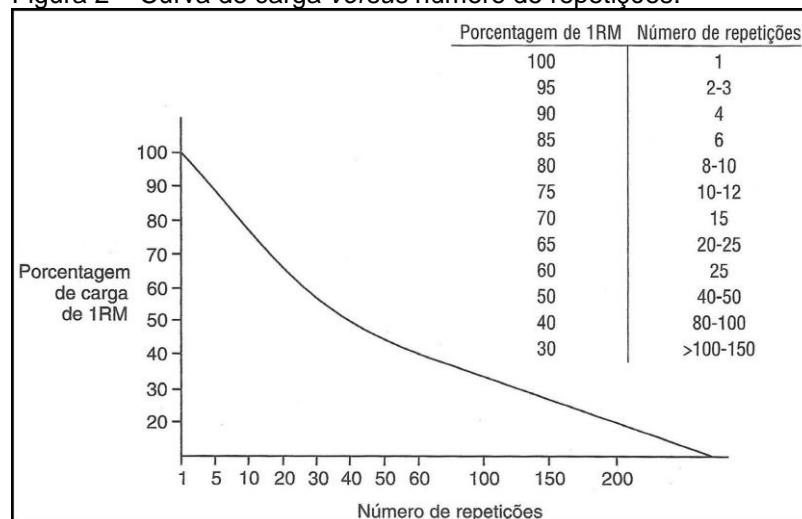
### 2.3 METODOLOGIAS DE PRESCRIÇÃO DE TREINO PARA MUSCULAÇÃO

A prescrição é uma instrução escrita sobre os exercícios físicos que devem ser realizados dentro de um período determinado (BARBANTI, 1997).

Bompa (2001) afirma que os principais pontos a serem observados ao montar um plano de treinamento são descritos por: volume de treinamento, intensidade, número de exercícios, repetições e séries.

- a) volume de treinamento é equivalente ao tempo de duração do treino, como exemplo: 30 minutos;
- b) intensidade é dada pela carga, que é o peso, a ser erguido ou através da repetição máxima, sendo muito importante que seja respeitado o limite individual de cada praticante;
- c) números de exercícios para defini-los não existe uma regra, partindo do ponto em que existem diferenças de objetivos de cada treinador, para os grupos musculares a serem trabalhados, neste caso é observado a idade e nível de desempenho do praticante;
- d) número de repetições e a velocidade em que deve ser executado o movimento é uma consequência da carga utilizada, quanto maior a carga, menor é o número de repetições e mais lentamente deve ser concluído o movimento (BOMPA, 2001). Esta relação pode ser observada na figura 2, que é dada pelo número de repetições versus a porcentagem de uma repetição máxima (1RM).

Figura 2 – Curva de carga *versus* número de repetições.



Fonte: Bompa (2001).

Para aumentar o rendimento físico é realizada uma prescrição com sequências de exercícios e movimentos repetidos de forma sistemática (BARBANTI, 1997). Dentro do treinamento a série é dada ao executar a repetição do exercício seguido por descanso. A quantidade de séries muda conforme o número de exercícios estabelecidos no treino e da combinação de força. As séries são menores quando os números de exercícios são maiores (WILLIAMS; GROVES; THURGOOD, 2010).

Barbanti (1997) afirma que o treinador estipulará quanto tempo se deve ficar em repouso até iniciar a próxima série, este tempo é conhecido como intervalo. O descanso da carga é feito para a recuperação total ou parcial e para o retorno da pulsação inicial. O tempo de pausa para recuperação entre os exercícios são fixados de acordo com:

- a) condicionamento individual;
- b) a intensidade da carga;
- c) tipo de treino exercido.

A figura 3 demonstra a forma em que são prescritos os exercícios dentro de um treino de musculação, contém os seus principais itens para que o praticante da atividade saiba quais deve seguir dentro da academia e como deve ser executado, tais como o nome do exercício, peso, série, repetições (WILLIAMS; GROVES; THURGOOD, 2010).

Figura 3 – Programas básicos.

APARELHO					PESOS LIVRES				
Trabalho de aquecimento (p. 46-47) 10 min.					Trabalho de aquecimento (p. 46-47) 10 min.				
EXERCÍCIO	P.	SÉRIES	REPS.	PESO (RM)	EXERCÍCIO	P.	SÉRIES	REPS.	PESO (RM)
Supino reto no aparelho	118-19	3-6	6-12	12	Supino (mãos aproximadas)	146-47	3-6	6-12	12
Remada sentada com cabo	96-97	3-6	6-12	12	Agachamento com barra	64-65	3-6	6-12	12
Desenvolvimento no aparelho ou Remada em pé com polia	125 126	3-6	6-12	12	Remada inclinada com barra	100-01	3-6	6-12	12
Puxada pela frente	93	3-6	6-12	12	Crucifixo inclinado com halteres	114-15	3-6	6-12	12
Leg press inclinado	78-79	3-6	6-12	12	Tração na barra fixa (pegada afastada)	94 (Var.)	3-6	6-12	12
Flexão plantar (em pé)	84	3-6	6-12	12	Desenvolvimento pela frente com barra (em pé)	124	3-6	6-12	12
Rosca com polia baixa	154-5	3-6	6-12	12	Flexão plantar (em pé)	84	3-6	6-12	12
Tríceps com polia alta ou Mergulho entre barras paralelas	148-49 141	3-6	6-12	12	Rosca direta	150-51	3-6	6-12	12
<b>ROTINA DE ESFRIAMENTO</b>					<b>ROTINA DE ESFRIAMENTO</b>				
Esfriamento 5 min.					Esfriamento 5 min.				
Alongamento de desenvolvimento (p. 208-13) 15 min.					Alongamento de desenvolvimento (p. 208-13) 15 min.				
<b>DURAÇÃO DO PROGRAMA</b> 6 a 8 semanas					<b>DURAÇÃO DO PROGRAMA</b> 6 a 8 semanas				
<b>FREQUÊNCIA DO PROGRAMA</b> 2 a 3 treinos/semana, 1 a 2 dias de descanso entre os treinos					<b>FREQUÊNCIA DO PROGRAMA</b> 2 a 3 treinos/semana, 1 a 2 dias de descanso entre os treinos				
<b>TEMPO DE RECUPERAÇÃO</b> de 30 s a 1,5 min. entre os exercícios					<b>TEMPO DE RECUPERAÇÃO</b> de 30 s a 1,5 min. entre os exercícios				

Fonte: Williams, Groves e Thurgood (2010).

Existem padrões para alguns métodos de treinamento, como específicos para força, explosão e resistência. A tabela 1 mostra a relação entre o método utilizado no treino, o percentual da carga, o número de repetições e séries (MUSSI; DIAS, 2005).

Tabela 1 – Parâmetros fisiológicos do treinamento muscular.

Parâmetro	% Sobrecarga	Repetições	Séries
<b>Força pura</b>	85 a 99	2 a 5	3 a 8
<b>Força dinâmica</b>	70 a 89	6 a 12	3 a 5
<b>Força explosiva</b>	30 a 60	6 a 10	4 a 6
<b>RML</b>	40 a 60	14 a 30	3 a 5
<b>Endurance</b>	25 a 40	Acima de 30	4 a 6

Fonte: Mussi e Dias (2005).

Os dois termos apresentados na tabela 1: Resistência Muscular Localizada (RML) e *endurance* são conhecidos na área de educação física como resistência muscular localizada e resistência muscular não localizada, respectivamente (MUSSI; DIAS, 2005).

Outra relação também é expressa entre a aplicabilidade do exercício, porcentagem de carga, velocidade do desempenho e intervalo de descanso, como descrita na tabela 2.

Tabela 2 – Diretrizes sugeridas para o Intervalo de Descanso (ID) entre as séries para várias cargas e suas circunstâncias aplicáveis.

Porcentagem de carga	Velocidade do desempenho	ID (em minutos)	Aplicabilidade
<b>105 (excêntrico)</b>	Lenta	4 – 5	Melhorar força máxima e tônus muscular
<b>80 – 100</b>	Lenta a média	3 – 5	Melhorar força máxima e tônus muscular
<b>60 – 80</b>	Lenta a média	2	Melhorar hipertrofia muscular
<b>50 – 80</b>	Rápida	4 – 5	Melhorar potência
<b>30 – 50</b>	Lenta a média	1 – 2	Melhorar RM

Fonte: Bompa (2001).

Em geral os métodos de treinamento, são divididos em duas categorias: alternado por segmento e localizados por articulação. Estes treinos podem ser realizados com pesos livres ou aparelhos, normalmente ambos estão disponíveis em salas de musculação. O primeiro é indicado para quem ainda não possui experiência com a musculação e possui um nível baixo de condição física, o objetivo é fadigar o musculo localizado, conforme a montagem do treino (COSSENZA, 1995).

Cossenza (1995) ainda afirma que ao iniciar o treino prescrito, devem-se levar em conta os músculos que serão trabalhados, começando por exercícios que atuam em grandes grupos musculares, visto que a força é um dos fatores importantes. Além de exigir mais do músculo antes da fadiga, o que maximiza o resultado.

Ao prescrever um treinamento de musculação é importante observar os seguintes pontos: O estado médico-esportivo, quando existe alguma observação

médica como lesões e deficiências, avaliação de treinos anteriores (quando já é um praticante de musculação), a condição física atual, a motivação e os objetivos principais, para que seja possível então estabelecer os exercícios principais do treino e por fim elaborar as séries de repetições (BARBANTI, 1997).

### 3 ANDROID

O Android é um sistema operacional desenvolvido para a utilização em sistemas móveis. A base de sua estrutura é feita por um *kernel* do Linux, que por sua vez é *Open Source*, ou seja, possui seu código aberto, tornando possível o acesso do código aos desenvolvedores para que estes possam realizar colaborações, correções e desenvolvimento de novos aplicativos (JACKSON, 2011, tradução nossa).

Não desenvolvido por uma única empresa e sim por um grupo chamado *Open Handset Alliance (OHA)*, onde estão envolvidas organizações mundialmente conhecidas como Dell, Intel, Motorola, Samsung e outros, além da Google que lidera o grupo, a plataforma de desenvolvimento para aplicativos móveis Android possui uma interface rica, *Global Positioning System (GPS)*, várias aplicações já instaladas, um poderoso ambiente de desenvolvimento, além de utilizar a linguagem de programação Java (LECHETA, 2013).

Desenvolvedores Android podem desenvolver e distribuir seus aplicativos na *Google Play*, também conhecida como *Play Store*, loja de aplicativos da plataforma Android, nela é possível disponibilizar aplicações gratuitas ou pagas para todo o mundo, visto que a loja atende todos os usuários de dispositivos com sistema Android (DEVELOPER, 2015, tradução nossa).

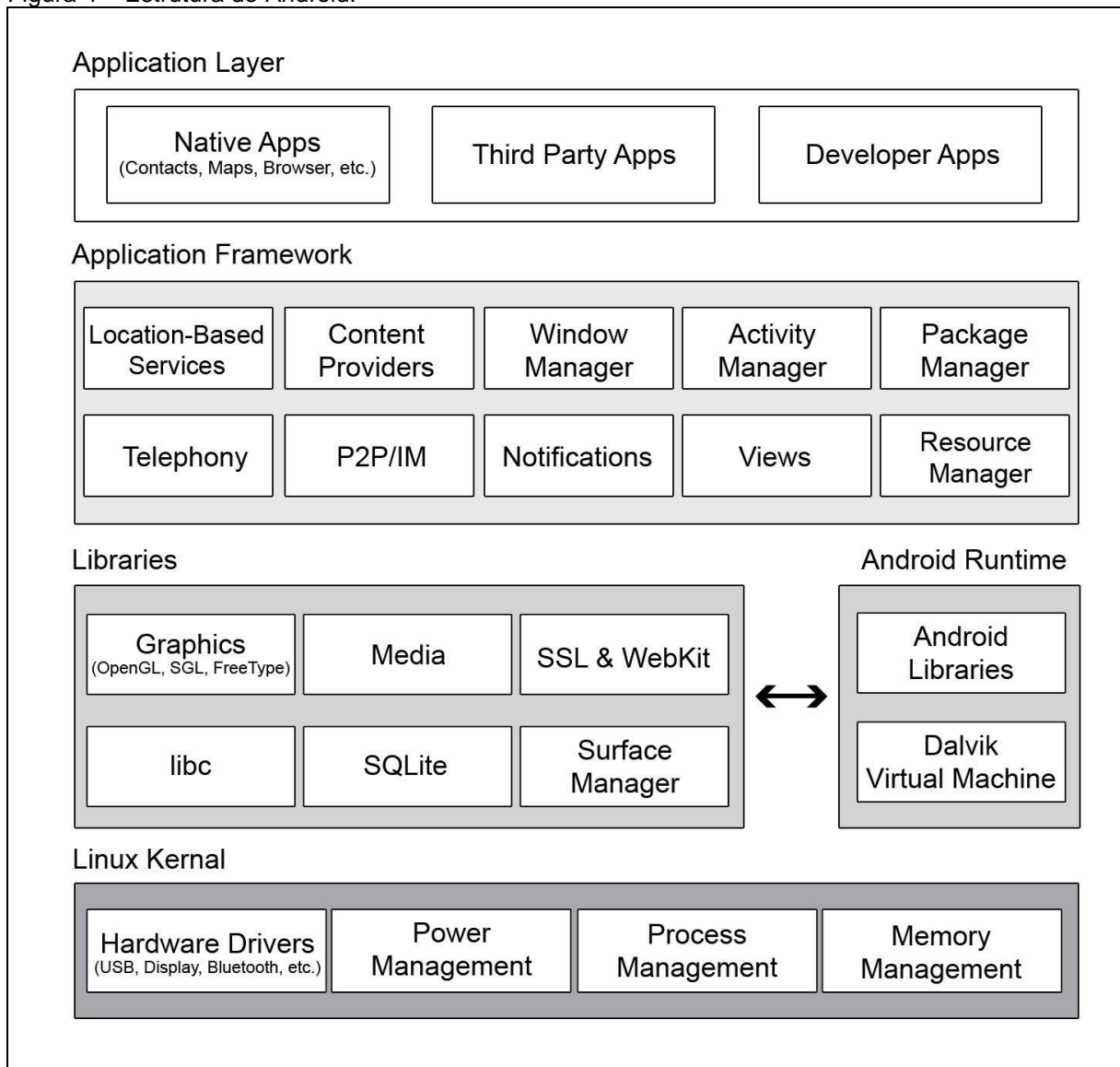
Gok e Khanna (2013, tradução nossa) afirmam que a classificação dos aplicativos Android pode ser dividida em quatro categorias, são elas:

- a) aplicações nativas: estão disponíveis em lojas de aplicativos como a *Play Store*, e são os mais comuns. Geralmente são utilizadas linguagens de programação de nível superior, tais como Java. *Application Programming Interface (APIs)* nativas, são fornecidas junto com o *Software Development Kit (SDK)* da plataforma desta forma é possível acessar os recursos de hardware, câmera e demais componentes do dispositivo. Outro ponto é que não precisa estar conectado à Internet para utilizar o aplicativo;
- b) genéricas são sites projetados com o foco em dispositivos móveis, geralmente utilizado em várias plataformas sem a utilização de APIs nativas;

- c) web dedicada estão aplicações que são web e foram adaptadas para uma plataforma móvel. A forma de acesso é através do endereço digitado na *Uniform Resource Locator* (URL) do navegador do dispositivo móvel, porém tem uma desvantagem que é a dependência da rede, podendo ter latência e prejudicando o acesso ao usuário;
- d) aplicações híbridas tem a forma de acesso semelhante com a web, dada através de um endereço na URL, porém a tela fica cheia, sem visualizar a barra de endereços e algumas ações são realizadas por comandos do navegador ou pelo WebKit. Esta última categoria usa uma camada conhecida como ponte, que permite o acesso do JavaScript a vários recursos específicos do dispositivo e APIs nativas que geralmente não são acessíveis do navegador web.

Sua estrutura é composta por cinco camadas, sendo elas: aplicação, *framework*, bibliotecas, execução e *kernel* Linux (MEIER, 2009, tradução nossa), camadas estas ilustradas na figura 4.

Figura 4 – Estrutura do Android.



Fonte: Meier (2009).

Segundo Meier (2009, tradução nossa) as características de cada camada podem ser descritas como:

- aplicação:** Todos os aplicativos nativos ou não, são construídos sobre a camada de aplicação e usam as mesmas bibliotecas e APIs. Faz uso das classes e serviços disponibilizados a partir da estrutura de aplicação em tempo de execução do Android;
- framework:** Fornece classes utilizadas para criar aplicativos Android e também o acesso ao hardware, gerenciamento da interface do usuário e aos recursos de aplicação;

- c) bibliotecas: Algumas bibliotecas rodam em cima do *kernel* e incluem outras em C/C++, bem como: bibliotecas de mídia para reprodução de áudio e vídeo, gerenciamento de exibição, gráficas que incluem SGL e OpenGL para gráficos 2D e 3D, SQLite para suporte de banco de dados, *Secure Sockets Layer* (SSL) e WebKit para navegador *web* e segurança na Internet;
- d) execução: Também chamada de *Android Runtime*, é o que diferencia um celular Android de um celular comum. A implementação em Linux com tempo de execução do Android, alimentam as aplicações junto com as bibliotecas *core* e a máquina virtual (VM) *Dalvik*. A máquina virtual *Dalvik* não é feita em Java, ela é baseada em registradores e otimizadas para garantir que o dispositivo execute vários aplicativos simultâneos. A biblioteca *core* é desenvolvida em Java, e ao contrário de bibliotecas específicas, elas estão disponíveis na linguagem C;
- e) *kernel* Linux: inclui drivers de hardware, gerenciamento de memória e processos, segurança, rede e gerenciamento de energia.

Para iniciar o desenvolvimento na plataforma Android é importante conhecer a estrutura do sistema operacional, suas principais funções e componentes e outras questões sobre as ferramentas e recursos disponíveis, que serão abordados no próximo capítulo.

### 3.1 DESENVOLVIMENTO ANDROID

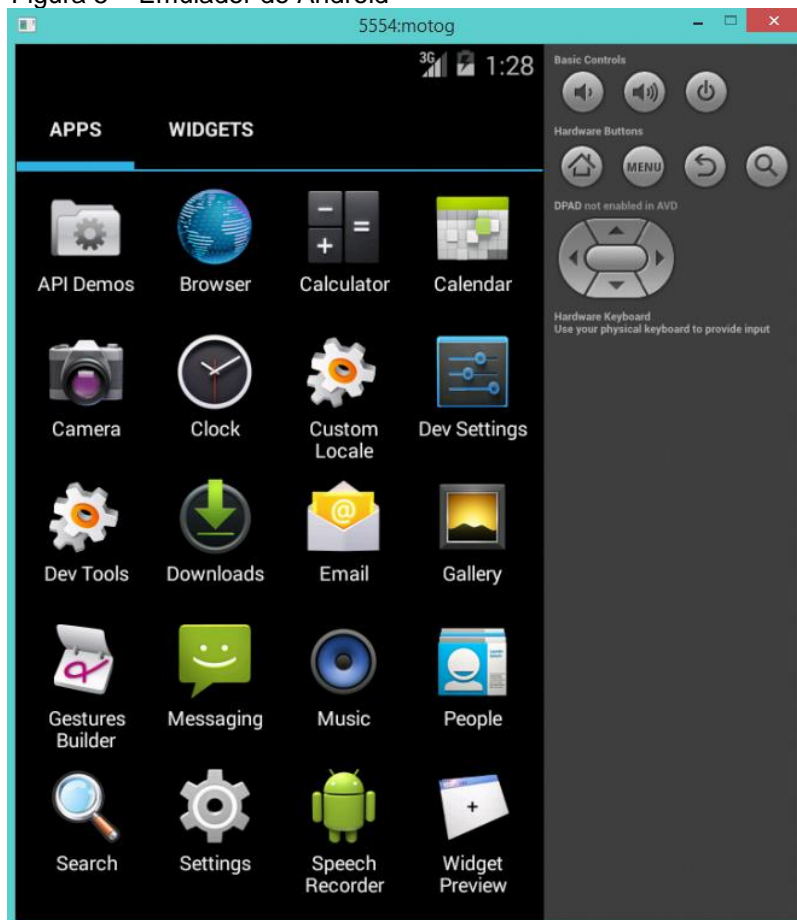
Ao desenvolver um aplicativo é importante levar em consideração alguns pontos, que mudam conforme a empresa que disponibiliza o dispositivo móvel, sendo eles: tamanho da tela do dispositivo, fragmentação e capacidade do *hardware* (WOLFSON; FELKER, 2013, tradução nossa).

- a) tamanho da tela: devido ao grupo OHA, existe mais de uma empresa que fornecem aparelhos com o sistema operacional Android, por este fato os tamanhos de tela mudam bastante e uma aplicação deve se adaptar para que a aparência continue boa;

- b) fragmentação: é referente as atualizações de versão do sistema operacional, cada aplicação será compatível com uma versão, cabe o usuário atualizar e o distribuidor do dispositivo liberar para o hardware;
- c) capacidade do *hardware*: são vários os recursos disponibilizados pelo fabricante e que limitam a capacidade do dispositivo, como memória, processador e outros recursos como câmera, GPS, que podem ser crucial para uma aplicação que dependa destes recursos.

É possível visualizar a tela do celular no computador através do emulador, para que isto seja possível é utilizado o *Android Virtual Device (AVD)*, que irá ter o comportamento de um celular real com as suas configurações, conforme ilustra a figura 5. É importante verificar as APIs usadas e se o celular comporta a versão em que foi desenvolvida a aplicação. Em casos de APIs recentes, estas não serão compatíveis com versões anteriores do Android, o que pode não garantir o funcionamento da aplicação (LECHETA, 2013).

Figura 5 – Emulador do Android



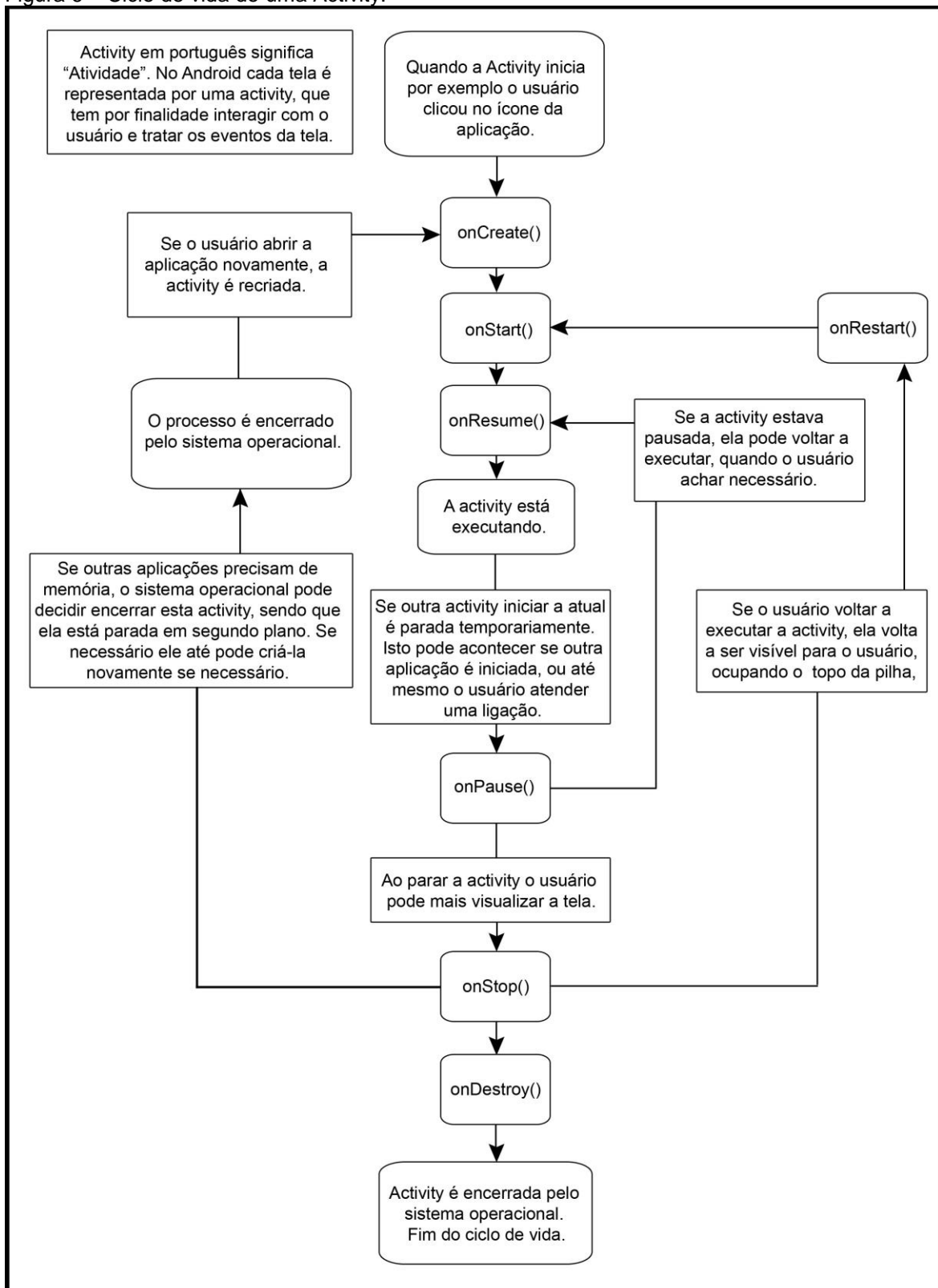
Fonte: Do autor.

Além dos conceitos citados anteriormente, é muito importante ter conhecimento das principais classes Android, cada uma tem uma função similar ao padrão *Model View Controller (MVC)* onde uma classe para o modelo que é a estrutura, outra para a visualização na tela e a última para ser o controlador com as funções da aplicação (LECHETA, 2013).

O comportamento de um *JFrame* do *Swing* em Java é a aplicação de uma tela composta de elementos visuais, como botão, caixa de texto e outros componentes. A classe que representa isto em Android é *android.app.Activity* e os elementos da tela são feitos pela classe *android.view.View*. Pelo fato da classe *Activity* definir a existência de uma tela, controlar seu estado, a troca de parâmetros entre as telas e os métodos que são chamados ao clicar em algum item da tela, é considerada o *Controller* do modelo MVC. A *View* como o nome em inglês já sugere tem a finalidade de demonstrar um componente gráfico na tela, como um botão, *checkbox*, imagem e atua no *layout* da tela, sendo considerada a *View* do modelo MVC, ou seja, a visualização (LECHETA, 2013).

Uma *Activity* deve ser declarada dentro do arquivo *AndroidManifest.xml* da mesma forma que um *Servlet* deve ser declarado no *web.xml* de uma aplicação *web*. A classe *Activity*, ou atividade, possui um ciclo de vida ilustrada na figura 6, passando por mudanças de estado: executando, temporariamente interrompida, em segundo plano ou completamente destruída. Primeiro é preciso pensar na seguinte situação: o usuário está executando o seu aplicativo Android no celular e recebe uma ligação, o que será feito com o aplicativo ao atender a ligação. É preciso definir se ao interromper a ação do aplicativo ele deve salvar os dados, fechar a aplicação ou deixar em *pause* para que seja retomado ao encerrar a ligação (LECHETA, 2013).

Figura 6 – Ciclo de vida de uma Activity.



Fonte: Lecheta (2013).

A figura 6 apresenta a vida completa de uma *Activity* da sua criação até a sua destruição. Ela é iniciada quando o usuário clica no ícone da aplicação e chama

o método *onCreate* que é obrigatório e executado uma única vez para a criação da *Activity*. No *onStart* ela fica visível ao usuário e chama os métodos para exibir os elementos na tela, o próximo é o *onResume*, que é chamado quando a *Activity* estiver no topo da pilha, neste momento o aplicativo está executando. Caso outra função do celular seja acionada, como uma ligação, é então chamado o método *onPause* e a *Activity* será interrompida temporariamente e colocada na pilha para voltar neste ponto futuramente. É possível executar o *onResume* após ter sido interrompido temporariamente para voltar à execução, ou o método *onStop* para encerrar a *Activity*. Se for preciso também é possível reiniciar a *Activity* com o método *onRestart* ou finalizar ela com o *onDestroy*, este último é executado quando o usuário encerra a *Activity* ou quando ela está em segundo plano e o sistema operacional precisa de mais memória, podendo ser encerrada por ele para liberar o espaço (LECHETA, 2013).

Durante este ciclo existem três sub-níveis que podem ficar repetindo durante a execução do aplicativo, que são: *entire lifetime*, *visible lifetime* e *foreground lifetime* (LECHETA, 2013).

- a) *entire lifetime* é o ciclo completo entre o início e a destruição da *Activity*, que ocorre somente uma vez e define o tempo de execução, os métodos presentes neste sub-nível estão entre *onCreate* e *onDestroy*;
- b) *visible lifetime* neste sub-nível a *Activity* já foi iniciada, podendo estar no topo da pilha, em execução e interagindo com o usuário ou em uma pausa temporária em segundo plano. Está presente entre os métodos *onStart* e *onStop*;
- c) *foreground lifetime* sub-nível executado entre os métodos *onResume* e *onPause*, está no topo da pilha e está interagindo com o usuário.

A classe responsável por enviar uma mensagem da aplicação solicitando a execução de uma tarefa é chamada de *Intent*, da classe *android.content.Intent*. Após o envio da solicitação o sistema operacional interpreta o conteúdo da mensagem e toma a decisão que pode ser abrir uma tela, uma página *web* ou outra função (LECHETA, 2013).

A comunicação da *Intent* é feita com a classe *BroadcastReceiver* e tem como finalidade a de reagir aos eventos gerados pela *Intent* em segundo plano, sem que o usuário tenha conhecimento. Para dar ênfase a uma mensagem ao usuário é possível fazer uma notificação que aparecerá na barra de status chamando a atenção. Esta ação é feita através da classe *Notification (android.app.Notification)* (LECHETA, 2013).

Além de notificações enviadas ao usuário é possível enviar uma mensagem através da *Intent* para o sistema operacional informando a data e hora que a aplicação deve ser executada, recurso utilizado pela classe *AlarmManager (android.app.AlarmManager)*. Este agendamento só é cancelado quando o aparelho é reiniciado (LECHETA, 2013).

Para o desenvolvimento de uma aplicação com todos os recursos citados existem ferramentas e tecnologias disponíveis, linguagem de programação, banco de dados SQLite e emuladores, todos estes citados no próximo capítulo.

### 3.2 TECNOLOGIAS ENVOLVIDAS NO DESENVOLVIMENTO ANDROID

As aplicações Android são desenvolvidas para dispositivos da área móvel utilizando o Android *SDK*, ferramenta que está disponível sem custos e taxas para os desenvolvedores. A linguagem pode ser escolhida conforme a experiência de cada um, partindo da ideia que existe mais de uma linguagem compatível com o Android, sendo elas: Java, C/C++, Python, Ruby, *Hypertext Markup Language* (HTML) e Javascript (GOK; KHANNA, 2013, tradução nossa).

A linguagem de programação Java contém uma sintaxe considerada amigável, com recursos de orientação de objetos - OO, gerenciamento de memória e o melhor: portabilidade (SIERRA; BATES, 2009). O diferencial desta linguagem é

que pode ser executada em qualquer hardware desde que tenha uma máquina virtual (LUCKOW; MELO, 2010).

Uma das possíveis interfaces para o desenvolvimento de uma aplicação em Java para Android é o Ambiente Integrado de Desenvolvimento, do inglês *Integrated Development Environment (IDE) Eclipse*. Que é uma plataforma de código aberto para desenvolvimento, sendo possível utilizar outras linguagens de programação nesta IDE Eclipse, bastando adicionar alguns *plug-ins* (LUCKOW; MELO, 2010).

Para facilitar o desenvolvimento Android o Google criou no ambiente de desenvolvimento *Eclipse* um plug-in conhecido como *Android Development Tools (ADT)* que também ajuda nos testes e na compilação do projeto. Através dele é possível utilizar o emulador direto da IDE Eclipse e todos os seus recursos, como debug que executa passo a passo o código, controlar o emulador, ter acesso aos logs, simular uma chamada ou uma mensagem chamada de *Short Message Service (SMS)*, além de enviar e visualizar arquivos através do emulador (LECHETA, 2013).

Outra IDE utilizada para o desenvolvimento Android é o Android Studio, esta ferramenta foi adotada como o ambiente de desenvolvimento oficial do Android. Que dispõe de todos os recursos necessários, como o editor de código inteligente, capaz de refatorar e analisar os códigos apontando os possíveis erros e sugerindo o que deve ser alterado, permite as simulações do aplicativo em diversos tamanhos de telas e tipos de dispositivos, como *Tablets, Smartphones, Android Wear, Android TV* e *Google Glass*. Pode ser feito o download da ferramenta de forma gratuita (DEVELOPER, 2015).

Lecheta (2013) também diz que uma integração de dados entre o Android e um banco de dados é possível através do SQLite que é um banco de dados gratuito, poderoso e leve, característica importante ao pensar na capacidade do dispositivo. O gerenciamento pode ser feito através de ferramentas disponíveis gratuitamente como *SQLite Expert Personal* e *SQLite Plus*.

O SQLite é de código aberto e já vem embutido na plataforma Android o que torna mais fácil a sua utilização, feita através APIs nativas. O SQLite implementa um banco de dados SQL que geralmente é utilizado em aplicativos para armazenar uma pequena ou média quantidade de dados devido a capacidade dos dispositivos.

Ele suporta dados dos tipos: *text*, *integer* e *real* similares em Java aos respectivos tipos de dados: *String*, *long* e *double* (LEE, 2012, tradução nossa).

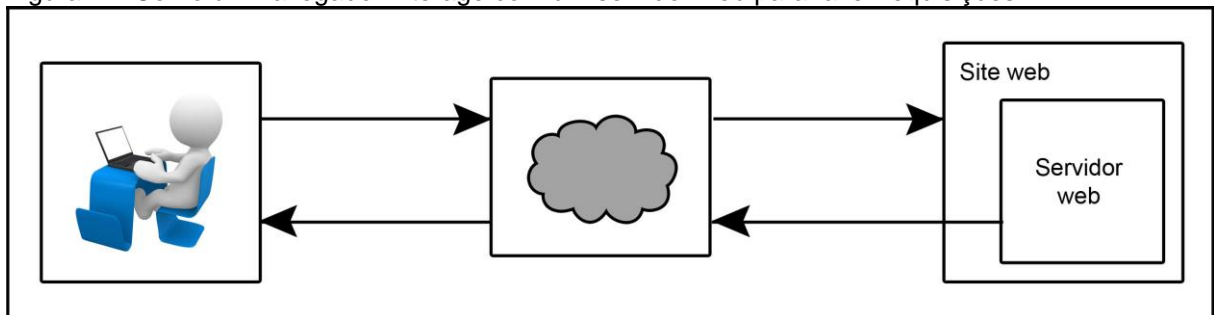
A possibilidade de conexão Android com o banco de dados é muito importante para o armazenamento de conteúdo dos aplicativos, devido à capacidade limitada dos dispositivos. Uma solução é utilizar uma conexão com um servidor para gravar os dados em um banco de dados diretamente no servidor e utilizá-los sempre que necessário (LECHETA, 2013).

Para o desenvolvimento de sistemas *web*, *desktops* e Android pode-se utilizar a plataforma Java que oferece recursos como: *Servlet*, *Java Server Pages* (JSP), *Java Server Faces* (JSF) além de opções para o desenvolvimento de ambientes *desktops* como *swing* (MENDES, 2009).

## 4 APLICAÇÃO WEB

Aplicações *web* ou sites são projetados para fornecer o retorno de uma requisição feita pelo usuário, a pessoa informa um endereço URL para acessar uma página *web*, a requisição é enviada para o servidor que processa a solicitação e retorna uma resposta para a requisição do usuário, como ilustra a figura 7 (POTTS; KOPACK, 2003).

Figura 7 – Como um navegador interage com um servidor web para fazer requisições.



Fonte: Potts e Kopack (2003).

### 4.1 TECNOLOGIAS APLICADAS EM DESENVOLVIMENTO WEB

Para o desenvolvimento de uma aplicação *web* é necessário escolher as ferramentas e tecnologias que permitam a criação, armazenamento, manipulação e a comunicação de dados. As seleções destas ferramentas podem envolver vários aspectos, como a popularidade e a experiência (LUCKOW; MELO, 2010).

#### 4.1.1 Java Enterprise Edition

*Java Enterprise Edition (JEE)* pode ser definida como um conjunto de especificações detalhadas de como deve ser implementado um software. Possui API para desenvolvimento de aplicações, incluindo serviços de rede e *web*. As principais APIs disponibilizadas pelo JEE voltadas para o desenvolvimento *web* são: *JavaServer Pages (JSP)*, *Java Servlets* e *Java Server Faces (JSF)* (CAELUM, 2012).

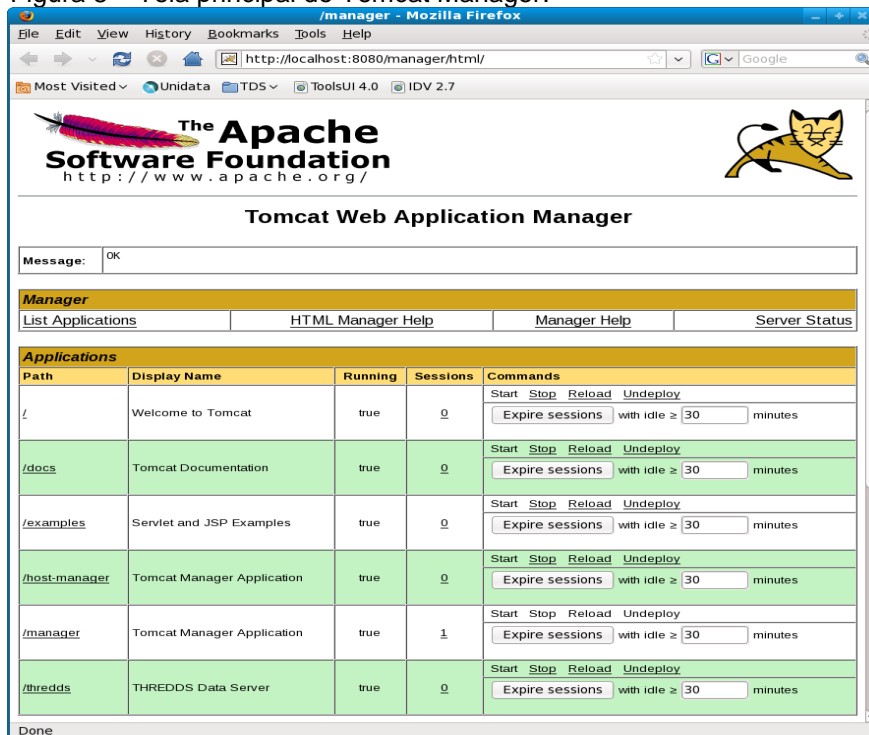
A plataforma JEE fornece um ambiente de tempo de execução e API para desenvolvimento e execução em grande escala, multi-camadas, aplicações de rede escaláveis, confiáveis e seguras, além da compatibilidade com servidores, como:

*GlassFish* e também a ferramenta *Apache Tomcat* (ORACLE, 2014, tradução nossa).

O *JSF* é um *framework web* padrão da plataforma *Java Enterprise Edition*, que permite a utilização da linguagem de programação *Java* para desenvolvimentos de páginas *web* (LUCKOW; MELO, 2010). Seu principal foco é simplificar e facilitar o desenvolvimento das telas de uma aplicação *web*, também conhecida como interfaces, por meio de componentes reutilizáveis (FARIA, 2013).

O *Apache Tomcat* pode ser definido com duas funções: contêiner *Java* e servidor *web* simultaneamente. É compatível com as tecnologias *Java Servlet* e *JSP*, que possibilitam o funcionamento do *Java* em um ambiente *web*. Além da sua capacidade de execução do *JSP* e *Java Servlet*, o *Apache Tomcat* possui um recurso para o gerenciamento das sessões *web*, como é possível visualizar na figura 8 (LUCKOW; MELO, 2010).

Figura 8 – Tela principal do Tomcat Manager.



Path	Display Name	Running	Sessions	Commands
/	Welcome to Tomcat	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/docs	Tomcat Documentation	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/examples	Servlet and JSP Examples	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/host-manager	Tomcat Manager Application	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/manager	Tomcat Manager Application	true	1	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/thredds	THREDDS Data Server	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes

Fonte: Unidata (2014).

#### 4.1.2 MySQL

O *MySQL* é um banco de dados, utilizado para o armazenamento dos dados de um software. É o banco de dados mais conhecido que possui código-aberto, grandes empresas utilizam este banco, como Amazon.com, Google,

Motorola e outros. Além de possuir recursos como o *MySQL Workbench*, uma ferramenta visual para o desenvolvimento da estrutura do banco de dados (LUCKOW; MELO, 2010).

A ferramenta utilizada para conexão entre um banco de dados e a linguagem Java é conhecida como *Java Database Connectivity (JDBC)*, que consiste em um conjunto de classes e interfaces *API* escritas em Java que possibilitam a conexão com um banco de dados ou o acesso a qualquer fonte de dados tabular, além do envio de instruções SQL e a possibilidade de processar os resultados (ORACLE, 2014, tradução nossa).

#### 4.2 SINCRONIZAÇÃO EM TEMPO REAL

Existe mais de uma forma de comunicação para que seja possível transmitir os dados de um dispositivo para a *web* e vice-e-versa, sendo eles: utilização do protocolo *Hypertext Transfer Protocol (HTTP)*, *sockets* ou *WebSockets*. Cada uma possui particularidades, porém todas funcionam com a finalidade da comunicação dos dados. O acesso através de um *WebService* é uma das tecnologias disponíveis para a comunicação, porém não existe uma *API* nativa para acessar um *WebService* de uma aplicação Android (LECHETA, 2013).

Um aplicativo *web* baseado em *HTML* não pode fazer uma comunicação *full-duplex*, nos dois sentidos, com o servidor, pois a comunicação *HTTP* só começa a partir de um cliente, isto significa que um servidor não pode enviar qualquer informação a um cliente sem acesso a partir do cliente. Diferente do protocolo *WebSocket* que possui a comunicação *full-duplex* e é de fácil desenvolvimento através do *Javascript* (FURUKAWA, 2011, tradução nossa).

Uma das formas de conexão para a comunicação dos dados é feito por *sockets*, que funcionam da seguinte maneira: para a conexão com o *socket* é preciso conectar-se em um determinado endereço com uma porta liberada, feito pelo pacote *java.net* e a classe *java.net.socket*. A aplicação conecta no servidor através do *socket*, que permanece ativa até que seja realizada uma ação que tenha uma resposta para retornar à aplicação. Devido a forma de processamento do *socket* a comunicação entre a aplicação do dispositivo e o servidor é feito em tempo real (LECHETA, 2013).

Ao pesquisar sobre as possíveis formas de comunicação entre as aplicações *web* e Android e compreender as limitações de cada uma foi escolhida a tecnologia WebSocket, com o objetivo de permitir que o início da conexão seja feito de ambos os lados e mantenha as aplicações funcionando em tempo real.

#### 4.2.1 WebSocket

O protocolo que tem sido utilizado na *web* é conhecido como *HTTP*, porém algumas limitações fazem com que ele seja trocado por um novo protocolo chamado WebSocket, que não apresenta ineficiência quando um aplicativo *web* é hospedado em um navegador e permanece conectado com o servidor em uma conexão persistente, como acontece com o protocolo *HTTP*. A comunicação bidirecional do WebSocket entre o aplicativo e o servidor *web* é feita por um único soquete *Transmission Control Protocol (TCP)*, que permite a comunicação em tempo real (ESPOSITO, 2012).

Além de ser um protocolo para a sincronização em tempo real, o WebSocket também é uma API padronizada pela *World Wide Web Consortium (W3C)* enquanto a padronização do protocolo é realizada pela *Internet Engineering Task Force (IETF)*, WebSocket também pode ser definida como uma conexão *full-duplex*, ou seja, pode se comunicar em ambas direções, sobre os quais mensagens podem ser enviadas entre cliente e o servidor. O padrão WebSocket simplifica muito a complexidade em torno da comunicação *web* bidirecional e de gerenciamento de conexão (WEB SOCKET, 2013, tradução nossa).

A tecnologia aplicada em um WebSocket proporciona a criação de aplicações mais performáticas com a comunicação em tempo real. Podem ser aplicados em casos de bate-papo, jogos *multiplayer online*, mapas interativos, notificações, ferramentas de colaboração entre outras opções (CAMPANINI, 2012).

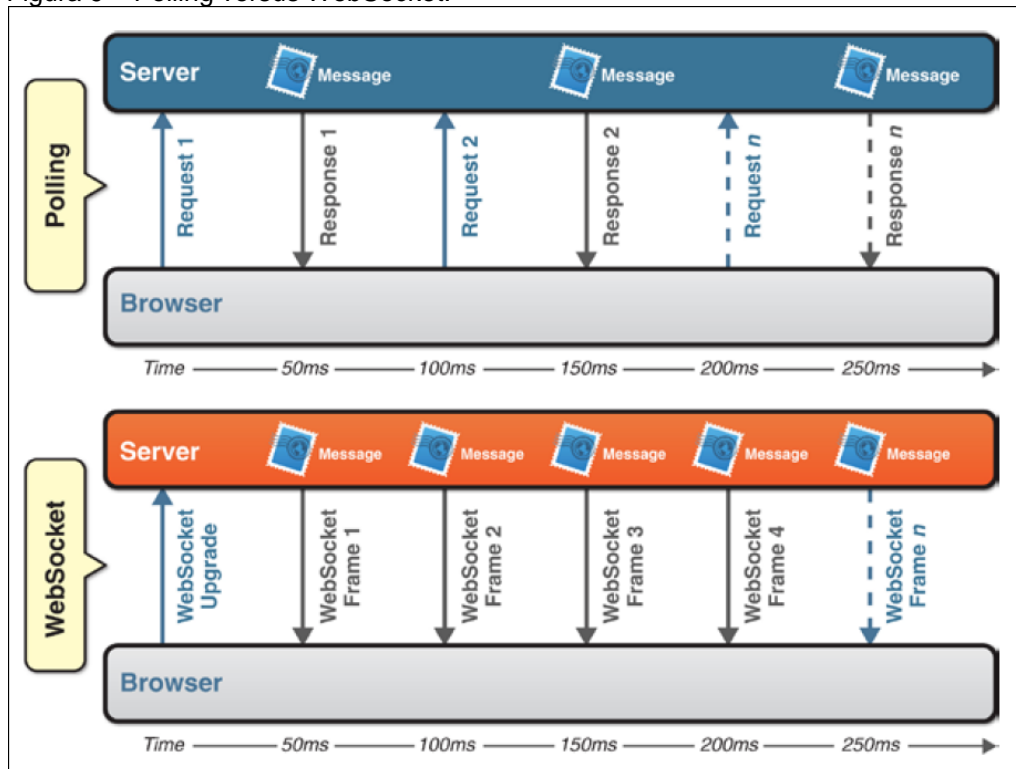
Entretanto, a tecnologia WebSocket é nova e não é suportada em todos os browsers, existem alternativas para estas situações como as tecnologias de *polling*, *long-polling* e *streaming* (WEBSOCKET, 2014, tradução nossa).

- a) *polling*: são enviadas solicitações pelo navegador através do *HTTP* com intervalos regulares até que receba uma resposta, como ilustra a figura 9. É uma forma de comunicação mais próxima ao tempo real anterior ao

WebSocket, porém é necessário prever quando o servidor terá a resposta para enviar, caso contrário serão abertas e fechadas várias conexões sem uso real (WEBSOCKET, 2014, tradução nossa);

- b) *long-polling*: o navegador envia uma solicitação ao servidor que mantém o pedido em aberto por um período definido, caso tenha uma resposta durante a conexão aberta é enviada uma resposta, quando não obter a resposta e terminar o tempo definido é encaminhada uma mensagem para encerrar a conexão. O que não possui vantagem quando o número de requisições for alto (WEBSOCKET, 2014, tradução nossa);
- c) *streaming*: o navegador envia uma requisição completa ao servidor que retorna a resposta e a mantém aberta, é constantemente atualizado e enviadas as respostas, porém o servidor não sinaliza quando o retorno está completo que mantém a conexão em aberta para requisições futuras, consequentemente causa uma alta carga no servidor (WEBSOCKET, 2014, tradução nossa).

Figura 9 – Polling versus WebSocket.



Fonte: Wang, Salim e Moskovits (2013, tradução nossa).

As tecnologias alternativas ao WebSocket são indicadas quando o *browser* não suporta a nova tecnologia, pois o WebSocket tem vantagem de ter o serviço em tempo real e não gerar muita carga como as outras opções (WEBSOCKET, 2014, tradução nossa).

#### 4.2.2 Funcionamento do WebSocket

O protocolo WebSocket é baseado em uma conexão persistente e possui um comportamento de envio de mensagens, com o conteúdo de requisições e respostas a estas requisições. O cliente inicia enviando uma requisição ao servidor com o cabeçalho “*upgrade*” para que seja possível alterar do modo HTTP para WebSocket e o servidor deve retornar a resposta da solicitação do cliente, este procedimento é nomeado de *Open Handshake*, como ilustra a figura 10 (CAMPANINI, 2012).

Figura 10 – Esquema do funcionamento WebSocket.



Fonte: Adaptado de WebSocket (2014, tradução nossa).

Quando o servidor reconhece o protocolo WebSocket ele retorna a requisição realizada pelo browser com um cabeçalho de resposta informando que o *HTTP* será substituído pelo WebSocket sobre a mesma conexão *TCP/IP* assim como as portas que eram utilizadas pelo *HTTP* (80) e *HyperText Transfer Protocol Secure* (HTTPS) (443), que é o padrão. Ao estabelecer a conexão com o WebSocket ela é mantida e pode ser enviadas mensagens através de quadros de ida e volta tanto do cliente como do servidor, esta tecnologia é conhecida como *full-duplex*, comunicação ambas direções. Os dados são colocados no quadro com apenas dois

bytes, cada um com um byte 0x00, termina com um *byte* de 0xFF, e contém dados *8-bit Unicode Transformation Format* (UTF-8) no meio. Os quadros de envio de dados em forma de texto usam uma terminação enquanto quadros binários usam o tamanho de comprimento (WEBSOCKET, 2014, tradução nossa).

Por ser uma tecnologia nova o WebSocket possui algumas incompatibilidades com servidores e *browsers*. O servidor *GlassFish* 3.1 é compatível com o WebSocket (CAMPANINI, 2012) bem como o Apache Tomcat a partir da versão 7 (TOMCAT, 2014). Os *browsers* que possuem suporte para o WebSocket, são: *Chrome* 4.0+, *Safari* 5.0+, *Firefox* 4.0+, *Opera* 11 e *Internet Explorer* 9+ Beta (DRYSDALE, 2011).

A maior vantagem para a utilização do *WebSocket* é ter uma conexão entre servidor e cliente com o tempo real de baixa latência. Além de gerar pouca carga no servidor. Com isso empresas grandes utilizam a tecnologia *WebSocket*, como o *Twitter* e *Facebook*, já que é ideal para aplicar em serviços de bate-papo, jogos online com vários jogadores, transmissões ao vivo via Internet e atualizações em tempo real nas redes sociais (UBL; KITAMURA, 2010).

## 5 TRABALHOS CORRELATOS

Durante o levantamento bibliográfico e pesquisas ao decorrer deste trabalho foram encontrados artigos e trabalhos que abordam temas presentes neste projeto.

### 5.1 A UTILIZAÇÃO DA LINGUAGEM SWIFT EM DISPOSITIVOS MÓVEIS BASEADOS NO SISTEMA OPERACIONAL IOS PARA AVALIAÇÃO DA COMPOSIÇÃO CORPORAL

Este projeto de conclusão de curso foi apresentado para a obtenção de grau de bacharel no curso de Ciência da Computação na Universidade do Extremo Sul Catarinense em 2014 pelo acadêmico Guilherme Waltricke Freitas. O trabalho tem como objetivo desenvolver uma aplicação móvel utilizando a linguagem de programação Swift para dispositivos com o sistema operacional iOS. O aplicativo envolve a área de educação física, visto que este é voltado para a utilização dentro de uma academia para a avaliação da composição corporal. Entretanto o projeto integra duas áreas distintas, sendo elas a Ciência da Computação e a Educação Física.

Para o desenvolvimento da aplicação foram realizadas pesquisas e reuniões com profissionais da Educação Física, a fim de obter conhecimento do processo para efetuar a avaliação da composição corporal, bem como pesquisas relacionadas as ferramentas necessárias para o desenvolvimento do aplicativo. As ferramentas utilizadas foram: IDE de desenvolvimento Xcode, banco de dados SQLite, experiência de usuário (UX) e principalmente a linguagem de programação Swift juntamente com o sistema operacional iOS 8. Para testes e simulações foi utilizado o iOS Simulator (FREITAS, 2014).

Com o desenvolvimento deste projeto foi possível chegar aos resultados esperados, uma aplicação funcional para os profissionais da educação física que aplicam a avaliação da composição corporal disponibilizando na plataforma iOS através da utilização de uma linguagem relativamente nova e pouco abordada em projetos de pesquisa e conclusão de curso. Além da obtenção de certificado disponibilizado pela Apple por ter desenvolvido a aplicação na linguagem Swift.

## 5.2 PROTÓTIPO DE APLICATIVO COM A UTILIZAÇÃO DA METODOLOGIA ÁGIL SCRUM E DO CONCEITO KANBAN PARA ACOMPANHAMENTO DO DISPÊNDIO ENERGÉTICO EM ATIVIDADES FÍSICAS

Projeto de conclusão de curso para a obtenção de grau de Bacharel no curso de Ciência da Computação pela Universidade do Extremo Sul Catarinense em 2015, pelo acadêmico Thalles Jacobs Vieira. Com o objetivo de aplicar a metodologia ágil *Scrum* em conjunto com o conceito *Kanban* a fim de desenvolver um protótipo de aplicativo *mobile* baseado no sistema operacional Android para o monitoramento do dispêndio energético em atividades físicas.

Neste trabalho as metodologias pesquisadas e utilizadas para o desenvolvimento do protótipo permitiram monitorar melhor os passos de desenvolvimento da aplicação bem como os erros, permitindo assim a correção de uma maneira mais rápida. O aplicativo recebe informações diretamente dos usuários que possibilita o monitoramento do dispêndio energético obtido em atividades praticadas, estes dados são retroalimentados, permitindo o acompanhamento visual destas informações posteriormente. As ferramentas utilizadas para o desenvolvimento foram: IDE do Eclipse, máquina virtual Java bem como a linguagem de programação Java e XML, além do ADT *plugin* e banco de dados SQLite.

Por fim chegou-se ao resultado de um protótipo de aplicativo *mobile* Android com as funções estabelecidas para o acompanhamento do dispêndio energético em atividades físicas desenvolvido com os métodos *Scrum* e *Kanban* utilizadas durante o desenvolvimento da aplicação, contudo estes métodos foram importantes para conseguir atingir o objetivo do desenvolvimento, corrigindo erros apontados já no início da implementação através da visualização dos mesmos em cada etapa seguida (VIEIRA, 2015).

### 5.3 WEBSOCKET PARA APLICAÇÕES WEB EM ERLANG

Trabalho de conclusão de curso desenvolvido por Emiliano Carlos de Moraes Firmino em 2011, apresentado na Universidade do Estado do Amazonas. O trabalho tem como objetivo descrever sobre a implementação em *Erlang* do *internet-draft WebSocket Hixie 76* do *HTML5*, a tecnologia WebSocket foi utilizada para atender o problema de comunicação *Asynchronous Javascript and XML (AJAX)* que não possui comunicação bidirecional, o *HTML5* é utilizado pois é ele que tem suporte para a *API WebSocket* e a linguagem de programação escolhida foi a *Erlang* que é funcional de código aberto, criada no laboratório de Ciência da Computação da Ericsson para desenvolver aplicações de telecomunicações, que possuem um alto número de transações simultâneas e alta disponibilidade.

Para a realização de testes foi utilizado o software *Echo* que permite a verificação de envio das mensagens enviadas e recebidas do servidor, para constatar a aplicação do WebSocket.

Com a realização deste trabalho foi possível concluir que o protocolo WebSocket é simples e permite criar muitas aplicações ao ser utilizado juntamente com o *HTML5* e serviços na *web*. Os resultados dos testes obtiveram um retorno positivo, foi possível obter a solução para o caso da comunicação que permite a conexão a um servidor por uma *URL* ou ser usada por outro servidor de aplicação *Erlang* que pode prover serviços pelo protocolo a qualquer cliente (FIRMINO, 2011).

#### 5.4 SIMULAÇÃO DE FUTEBOL EM AMBIENTE WEB: VERSÃO 3.0

Trabalho de conclusão de curso apresentando por Thyago Schleuss na Universidade Regional de Blumenau do curso de Ciência da Computação no ano de 2011. O trabalho consiste em uma adaptação de um sistema *web* capaz de realizar simulações de uma partida de futebol através de *Applet* para uma plataforma padrão utilizando HTML5 e sincronização utilizando WebSocket. Também, um motor de inteligência utilizando redes neurais foi criado para permitir tomadas de decisão mais dinâmicas dos jogadores durante uma simulação (SCHLEUSS, 2011).

As tecnologias envolvidas para a visualização das simulações utilizadas foram padrões para *web*, como o *HTML5*, a versão 3 do *Cascading Style Sheets (CSS3)* e a versão 1.8.1 do *Javascript*, além da comunicação via WebSocket e servidor *Jetty* (SCHLEUSS, 2011).

Foi possível aplicar a tecnologias citadas junto com a Inteligência Artificial criando a simulação das partidas de futebol em ambiente *web*. Um fator importante no desenvolvimento do trabalho foi a utilização do WebSocket, no início a proposta era utilizar na comunicação o *AJAX*, porém com o estudo das ferramentas foi possível concluir que a comunicação do WebSocket é superior pois possui um canal de comunicação cliente e servidor bidirecional onde não há necessidade de ficar solicitando diversas requisições ao servidor (SCHLEUSS, 2011).

Conclui-se do trabalho que foram atendidos os objetivos esperados através das funcionalidades tecnológicas aplicadas (SCHLEUSS, 2011).

## 5.5 UM CHAT PICTOGRÁFICO PARA O SCALA: SISTEMA DE COMUNICAÇÃO ALTERNATIVA PARA O LETRAMENTO DE PESSOAS COM AUTISMO

Realizado como Trabalho de Conclusão de Curso de Leonardo Batecini Ramos em 2013 na Universidade Federal do Rio Grande do Sul em Porto Alegre, este trabalho tem em foco propiciar uma interação entre usuários de tecnologia assistiva inseridos no sistema *SCALA* por meio de um chat (RAMOS, 2013).

Para possibilitar esta realização foi feito um estudo de acessibilidade na *web* através de tecnologias assistivas que utilizassem pictogramas, permitindo o aumento da autonomia de usuários com uma necessidade de compreensão do sistema (RAMOS, 2013).

As tecnologias envolvidas foram o protocolo *WebSocket*, que permite a interação entre cliente-servidor sem o atraso nas transações e necessidade de abrir uma nova conexão. Na implementação foi utilizada a *API Socket.IO* integrante da ferramenta *Node.js* pioneira na utilização do *Javascript V8* para este tipo de desenvolvimento (RAMOS, 2013).

O autor conclui que é muito importante um sistema de comunicação alternativa para usuários com necessidades especiais, e foi possível concluir este projeto devido ao estudo das tecnologias e de *softwares* já existentes de Comunicação Aumentada e Alternativa. Foram feitas as análises de requisitos para montar a estrutura do *software* além do desafio da utilização do protocolo *WebSocket* que precisava para a comunicação do chat em tempo real (RAMOS, 2013).

## 5.6 IMPLEMENTAÇÃO DE UM SERVIDOR DE NEGOCIAÇÃO EM BOLSA BASEADO EM WEBSOCKET

Este trabalho foi realizado para o Mestrado em Ciência de Computadores em Porto, Portugal no ano de 2014 por Vasco Morais Themudo. Com o objetivo de criar uma ferramenta robusta para uso financeiro com comunicação em tempo real e segura com o servidor *ServerDealWeb* visto que é um recurso disponibilizado na *web* (THEMUDO, 2014).

As movimentações financeiras e este mercado estão em uma mudança constante e rápida, devido a importância dos dados e das transações, é necessária uma aplicação de comunicação em tempo real, que não seja prejudicada pela demora dos dados (THEMUDO, 2014).

Para suprimir as limitações da tecnologia HTTP de pedidos (requisições) e respostas foi usada a tecnologia do protocolo WebSocket, compatível com o HTML. Os testes criados de comparativos entre o *ServerDealWeb* e o protótipo desenvolvido com a comunicação de WebSocket demonstram que este protocolo permite uma utilização mais eficiente da largura de banda e um menor número de troca de mensagens para a mesma quantidade de informação (THEMUDO, 2014).

Após o desenvolvimento e testes concluiu-se que o protocolo de comunicação bidirecional WebSocket é aplicável e nesta aplicação possibilitou com que o objetivo principal do trabalho fosse atendido (THEMUDO, 2014).

## **6 COMUNICAÇÃO EM TEMPO REAL COM WEBSOCKET APLICADO AO DESENVOLVIMENTO DO PROTÓTIPO PERSONALFIT PARA PRESCRIÇÃO DE TREINO DE MUSCULAÇÃO PARA DISPOSITIVOS MÓVEIS ANDROID**

O presente trabalho consiste no desenvolvimento de duas aplicações a fim de possibilitar a aplicação do protocolo WebSocket para uma comunicação em tempo real, visando a usabilidade em um ambiente real através de testes aplicados em academias de musculação na cidade de Criciúma, Santa Catarina.

As aplicações serão utilizadas para auxiliar a prescrição de treinos de musculação, suprimindo a necessidade do uso das fichas ou folhas de papel, atual forma de consulta aos treinos dos praticantes da atividade de musculação. A ideia consiste em disponibilizar uma aplicação *web* para que o instrutor de musculação prescreva os treinos de musculação para o seu aluno, que por sua vez tenha acesso ao aplicativo desenvolvido para a plataforma Android nomeado PersonalFit, cuja função é a consulta dos exercícios que compõe o treino de forma prática e clara, o que facilita a rotina dentro da academia em horários de maior movimento, visto que a consulta dos treinos no aplicativo *mobile* conta com o nome do exercício, uma breve descrição em que o instrutor pode informar como executar o movimento do exercício bem como o número da máquina em que o exercício deve ser feito, além da série e repetição de cada exercício. De forma que facilite o entendimento do que deve ser feito e como deve ser executado o exercício, diminuindo assim o fluxo de pessoas com dúvidas relacionadas ao treino durante o horário de maior movimento na academia.

Entretanto, este projeto faz a integração de duas áreas distintas sendo elas a Ciência da Computação e a Educação Física, com o objetivo de atender uma necessidade da área da Educação Física através do desenvolvimento das aplicações especificadas citadas anteriormente bem como a aplicação da tecnologia WebSocket, o projeto conta também com testes aplicados em um ambiente real. Estes são de suma importância para que se obtenha noção da usabilidade dentro de um cenário real de academia de musculação, bem como o comportamento das aplicações com as tecnologias envolvidas.

## 6.1 METODOLOGIA

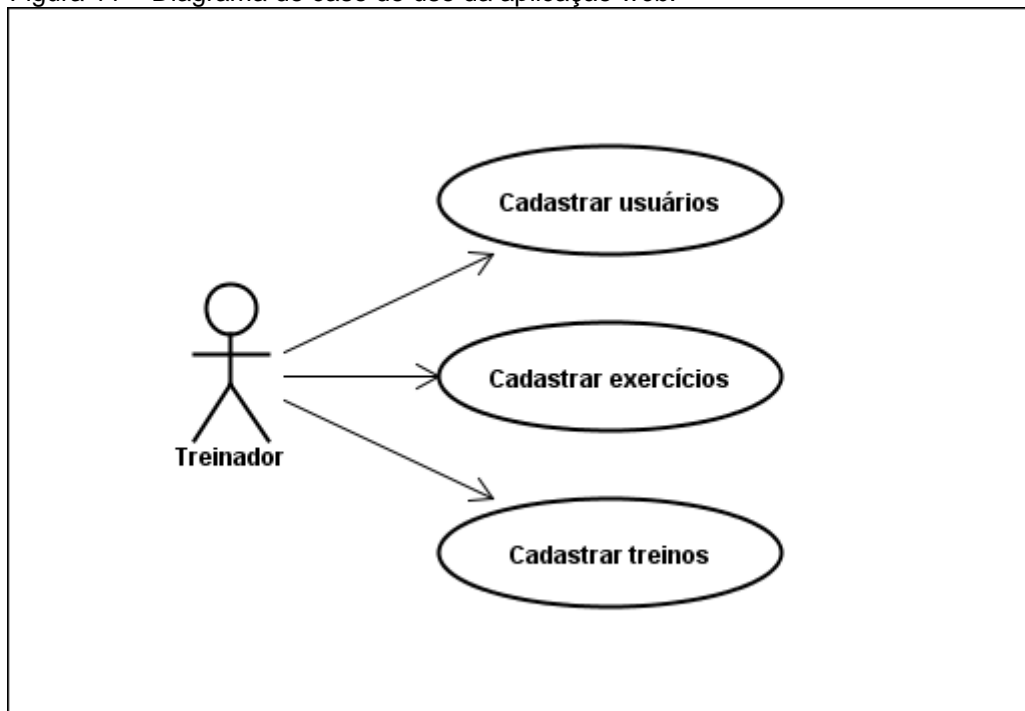
Para atingir os objetivos deste trabalho de conclusão de curso se fez necessário seguir, algumas etapas e estabelecer o que era necessário em cada etapa até a conclusão do projeto. Estas etapas foram abordadas nas sessões a seguir.

### 6.1.1 Levantamento bibliográfico

O levantamento bibliográfico foi realizado a partir de livros, artigos, pesquisas na Internet, *e-books* e publicações, a fim de coletar dados para a fundamentação teórica deste projeto, que teve início com a pesquisas relacionadas à musculação e atividade física para ter a base de como é realizado uma prescrição dos treinos. Após o estudo da área relacionada à educação física foi iniciada a pesquisa que envolve as áreas da Ciência da Computação: Plataforma Android, como é feito o desenvolvimento para dispositivos móveis e *web*, quais são as ferramentas disponíveis para o desenvolvimento de forma gratuita e as tecnologias que estão envolvidas para a integração dos dados, banco de dados e o foco na tecnologia WebSocket para a comunicação em tempo real entre as aplicações.

### 6.1.2 Requisitos e funcionamento das aplicações

Com o conhecimento dos requisitos necessários para a prescrição do treino foi possível definir o funcionamento do aplicativo e as suas principais funções a fim de atender os objetivos estabelecidos no projeto. Entretanto foi definido que a aplicação *web* é de uso exclusivo do instrutor de musculação, profissional qualificado para prescrever o treino de forma correta. Na aplicação o treinador terá acesso aos cadastros como ilustra o diagrama de caso de uso na figura 11, cadastro de usuário, exercícios e treino.

Figura 11 – Diagrama de caso de uso da aplicação *web*.

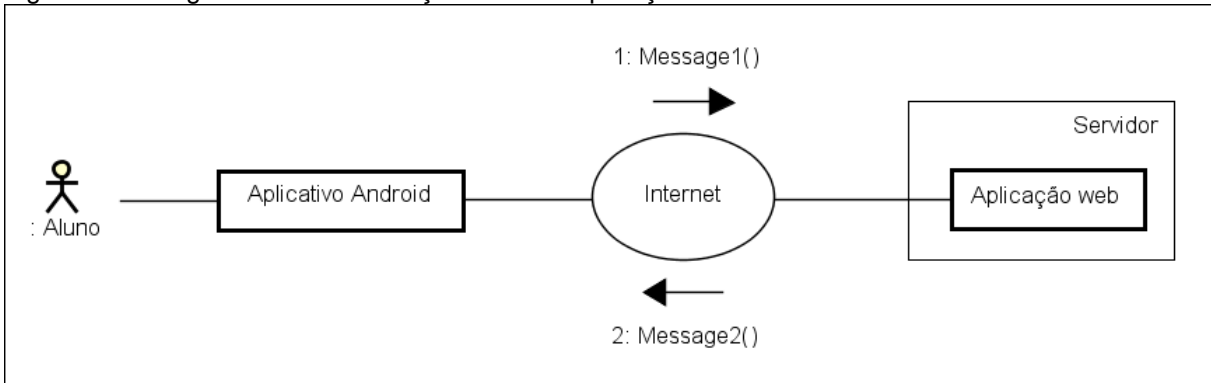
Fonte: Do autor.

Na aplicação o treinador terá acesso a cadastrar o usuário praticante da atividade física, de uma forma simples, com o nome, o *login* e a senha. Este *login* será informado para o aluno acessar o aplicativo *mobile* PersonalFit. Além do cadastro de usuário, a aplicação possui o cadastro de exercícios e treinos. Os exercícios são utilizados para prescrever o treino, sendo que os campos deste cadastro são compostos pelo nome do exercício e descrição, onde o instrutor pode colocar o modo correto de executar o movimento do exercício, bem como a numeração da máquina em que o exercício deve ser feito. Após cadastrar estes itens é possível cadastrar o treino para o aluno, selecionando o aluno cadastrado, uma descrição do treino que aparecerá na lista dos treinos do usuário no aplicativo *mobile*, como exemplo *treino A*, *treino B*, etc, além dos exercícios, série e repetições. Com estas informações cadastradas o usuário consegue receber os treinos no celular para a consulta durante o período do treino.

A aplicação para a plataforma Android foi nomeada PersonalFit, devido a publicação na Google *Play*, este nome é utilizado para a sua identificação na loja dos aplicativos. O aplicativo é baseado na consulta dos dados informados pelo instrutor na aplicação *web*, entretanto as aplicações se comunicam através da

Internet e utilizam o protocolo WebSocket para a comunicação em tempo real como é ilustrado na figura 12.

Figura 12 – Diagrama de comunicação entre as aplicações.



Fonte: Do autor.

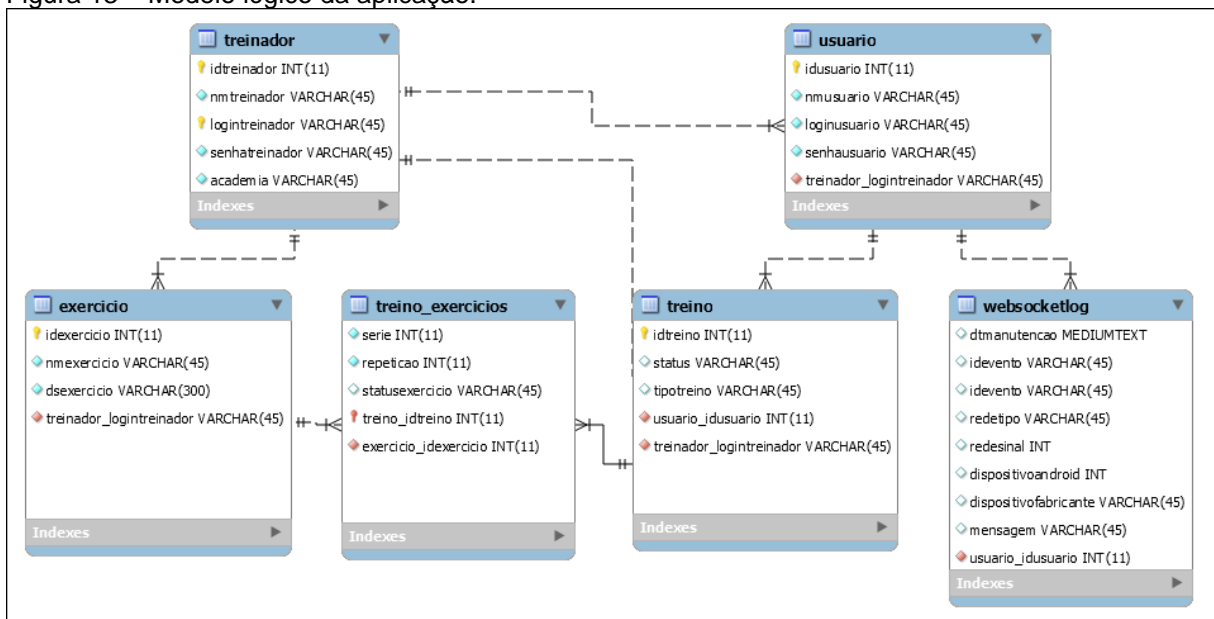
Qualquer pessoa pode fazer o *download* do aplicativo, visto que ficará disponível na Google *Play* por ser uma loja de aplicativos para a plataforma Android tanto de aplicativos pagos, como gratuitos, que é o caso da aplicação PersonalFit, porém só terá usuário e senha para o acesso ao aplicativo quem for cadastrado pelo instrutor de musculação, pois o objetivo é que o aluno tenha a consulta dos treinos no celular, desde que tenha um profissional da área de educação física que oriente quais são os exercícios corretos para aquela pessoa.

Ao entrar na aplicação é enviado o usuário e senha para o servidor afim de que realize a validação do *login*, ao retornar um usuário já cadastrado o aplicativo envia um *log* com as seguintes informações: código do usuário que está acessando o aplicativo, data, tipo do evento, o tipo de rede utilizada para a conexão, o sinal da rede, a versão do Android, o modelo do dispositivo, fabricante e a mensagem, que é a informação que envia ou recebe do servidor. Seguido de uma tela com uma mensagem de “bem vindo, *nome do aluno(a)*”, um botão para a consulta dos treinos e uma opção para sair da aplicação, ao clicar em consulta de treinos é verificado no servidor se o usuário possui treinos que retorna uma lista com os treinos disponíveis para consulta, ao selecionar o treino que realizará no dia, abre uma nova tela com os exercícios, séries e repetições. Na tela de consulta de treino é possível realizar o *check-in* de exercício por exercício, desta forma o aluno pode marcar o exercício como finalizado e este ficará de outra cor para a melhor visualização do que ainda falta fazer.

### 6.1.3 Modelagem do banco de dados

A etapa de modelagem de banco de dados consiste em analisar as informações necessárias que foram levantadas nos requisitos de funcionamento do sistema para criar um esquema lógico antes iniciar o desenvolvimento das aplicações, evitando erros de programação e o desperdício de tempo durante o desenvolvimento utilizado para analisar quais os itens devem ser salvos no banco de dados e de que forma os dados serão organizados dentro dele. A figura 13 mostra o modelo lógico criado para o desenvolvimento da aplicação *web* e Android, partindo do ponto que os dados serão compartilhados entre as aplicações utilizando o banco de dados MySQL.

Figura 13 – Modelo lógico da aplicação.



Fonte: Do autor.

Para a criação do modelo lógico foi utilizado a ferramenta MySQL *Workbench* versão 6.2 disponível<sup>3</sup> de forma gratuita para *download*. É possível criar o modelo lógico e a partir dele o banco de dados, como também o contrário, neste caso é chamado de engenharia reversa, ou seja, partindo da criação das tabelas, atributos e chaves primárias para depois criar o modelo lógico.

O modelo lógico apresentado foi utilizado para definir as tabelas, os atributos, chaves primárias e as relações entre as tabelas e as suas dependências.

<sup>3</sup> Disponível em: < <http://dev.mysql.com/downloads/workbench/>>. Acesso em: 29 de out. 2015.

#### 6.1.4 Definição das ferramentas e recursos

Para definir as ferramentas e recursos foi pesquisado durante o levantamento bibliográfico quais são distribuídas de forma gratuita e quais atendem a necessidade para o desenvolvimento das duas aplicações. Como citado na sessão de modelagem de banco de dados foi utilizado o banco de dados MySQL<sup>4</sup> para a aplicação *web* que possui compatibilidade com o MySQL Workbench que facilita durante o processo de modelagem do banco. Além da biblioteca Autobahn<sup>5</sup> utilizada para aplicar o WebSocket.

Como ambiente de desenvolvimento para a aplicação *web* foi definido o NetBeans<sup>6</sup> 8.0.2 pela facilidade de uso, além de disponibilizar como servidor o Tomcat<sup>7</sup> 8.0.18, utilizar linguagem de programação Java e recursos como JSP e *Servlet*. Para disponibilizar uma *interface* mais amigável ao usuário foi utilizado o *framework Foundation*<sup>8</sup> que possui códigos prontos para criar botões e itens mais bonitos além de tabelas responsivas que acompanham telas menores e suas dimensões sem que as informações fiquem muito pequenas.

O Android Studio 1.4<sup>9</sup> foi escolhido como ferramenta para o desenvolvimento Android, visto que é o ambiente de desenvolvimento oficial do Android e possui todos os recursos necessários para a implementação do aplicativo, desde recursos inteligentes para sugerir modificações no código quando está incorreto até simulações em diversos tamanhos de telas e dispositivos. E para armazenar poucos dados foi utilizado o SQLite que já vem na plataforma, ele implementa um banco de dados SQL através das classes *SQLiteOpenHelper* e *SQLiteDatabase*.

Entre as ferramentas e recursos utilizados no projeto o único que teve custo foi a criação de uma conta de desenvolvedor, necessário para ter acesso ao console<sup>10</sup> de desenvolvedor da Google ferramenta que permite publicar o aplicativo PersonalFit na loja de aplicativos Android Google Play, como mostra na figura 14.

---

<sup>4</sup> Disponível em: <<https://www.mysql.com/downloads/>>. Acesso em: 29 out. 2015.

<sup>5</sup> Disponível em: <<http://autobahn.ws/android/>>. Acesso em 30 out. 2015.

<sup>6</sup> Disponível em: <<https://netbeans.org/downloads/index.html>>. Acesso em: 29 out. 2015.

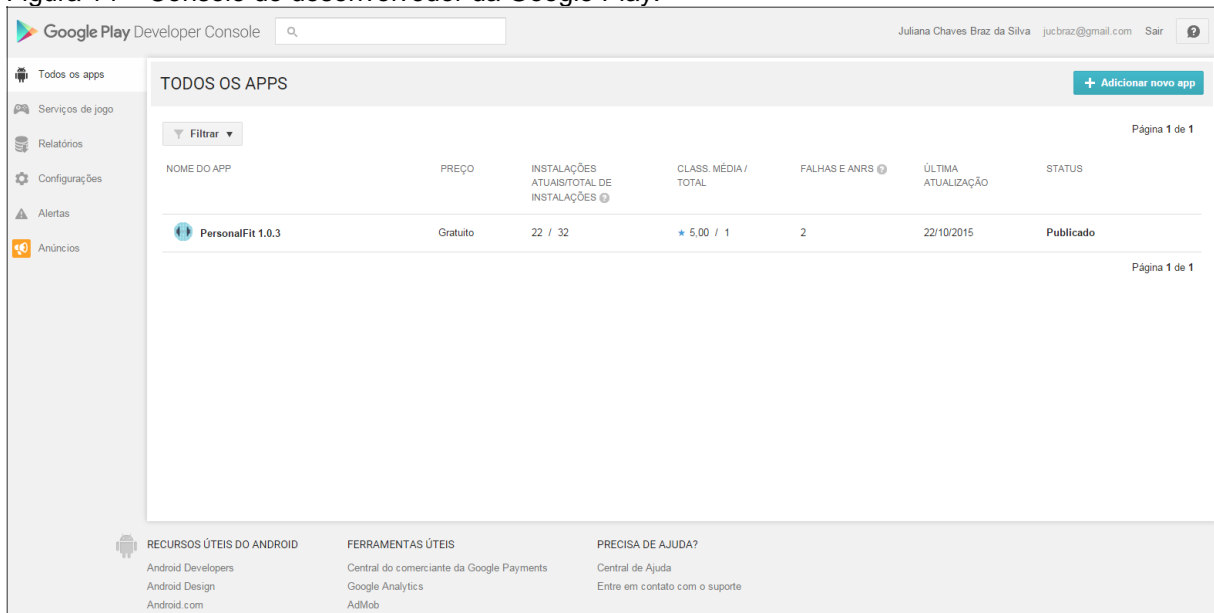
<sup>7</sup> Disponível em: <<http://tomcat.apache.org/download-80.cgi>>. Acesso em: 29 out. 2015.

<sup>8</sup> Disponível em: <<http://foundation.zurb.com/develop/download.html>>. Acesso em: 29 out. 2015.

<sup>9</sup> Disponível em: <<https://developer.android.com/intl/pt-br/sdk/index.html>>. Acesso em: 29 out. 2015.

<sup>10</sup> Disponível em: <<https://console.developers.google.com>>. Acesso em: 29 out. 2015.

Figura 14 – Console do desenvolvedor da Google Play.



Fonte: Google (2015).

O console utilizado por usuários com contas de desenvolvedor dispõe de estatísticas de uso do aplicativo, apontando gráficos e percentuais de *downloads*, com total de instalações, instalações atuais, número de usuários por versões do Android entre outras estatísticas que serão abordadas nos resultados obtidos com os testes no ambiente real das academias.

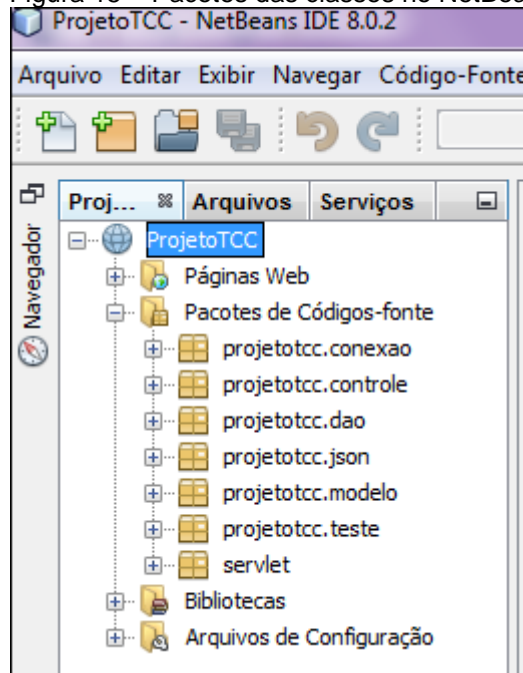
E, por fim foi selecionado o software *Astah Community*<sup>11</sup> para criar os diagramas apresentados no projeto, bem como casos de uso e modelagens. A ferramenta possui versão para teste com uma licença temporária, porém como estudante foi possível solicitar via e-mail a licença de estudante, que neste caso foi adquirida de forma gratuita apenas com a comprovação de estudante, a licença é válida por um ano.

<sup>11</sup> Disponível em: <<http://astah.net/download>>. Acesso em: 29 out. 2015.

### 6.1.5 Desenvolvimento web

Conforme apresentado na sessão de definição de ferramentas e recursos foi utilizado a IDE NetBeans para a criação da aplicação *web*, que consiste na aplicação que o instrutor utiliza para prescrever o treino do aluno. A organização do projeto *web* se faz através de pacotes para facilitar o controle e alterações das classes, como é mostrado na figura 15, bem como o padrão *Model View Controller* (MVC), composto por classes que são utilizadas como modelo em que são definidos os atributos, visão que é composta pelos itens da *interface* e controle que são as funções ou ações utilizadas em uma tela.

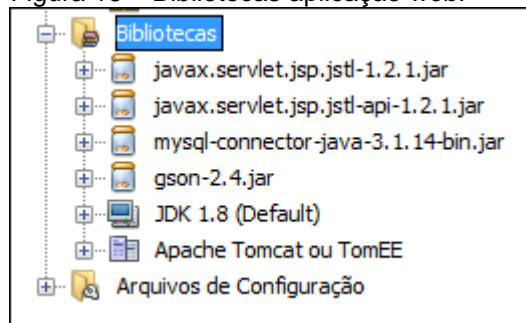
Figura 15 – Pacotes das classes no NetBeans.



Fonte: Do autor.

Em conjunto ao ambiente de desenvolvimento NetBeans foi utilizado o Apache Tomcat na versão 8.0.18 como servidor do projeto *web* juntamente com o banco MySQL. Além do uso do JSP para criação das telas da aplicação e *Servlet* para o controle das ações das páginas. Para a utilização dos componentes foram adicionadas as bibliotecas necessárias para o funcionamento destes recursos, que são: *mysql-connector*, *javax.servlet*, *gson-2.4*, mostrados na figura 16.

Figura 16 – Bibliotecas aplicação web.



Fonte: Do autor.

Para o *design* as páginas web foi utilizado o *framework Foundation* que possui recursos como botões, barra de menu, tabelas responsivas, links entre outros componentes para melhorar a *interface* da aplicação além de facilitar o desenvolvimento do projeto, neste caso utilizou-se as tabelas de forma responsiva para que se enquadrem em diversos tamanhos de tela, a figura 17 mostra como foi declarado o uso do *framework* dentro das classes, permitindo que ao criar as tabelas nas classes já estejam definidas como responsivas.

Figura 17 – Uso do *framework* Foundation.

```

10 <%@page import="projetotcc.modelo.Usuario"%>
11 <%@page contentType="text/html" pageEncoding="UTF-8" session="true"%>
12 <%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
13
14 <!DOCTYPE html>
15 <html class="no-js" lang="en">
16   <head>
17     <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
18     <meta charset="utf-8" />
19     <meta name="viewport" content="width=device-width, initial-scale=1.0" />
20     <title>Treino</title>
21     <link rel="stylesheet" href="css/foundation.css" />
22     <script src="js/vendor/modernizr.js"></script>
23
24   </head>
25
149 </script>
150
151     <script src="js/vendor/jquery.js"></script>
152     <script src="js/foundation.min.js"></script>
153     <script>
154       $(document).foundation();
155     </script>
156
157   </body>
158 </html>

```

Fonte: Do autor.

A aplicação *web* abrange os cadastros básicos para a prescrição do treino, inicialmente foi criado uma tela para efetuar o *login*, com a ideia de restringir o acesso somente ao profissional responsável por prescrever os treinos, como mostra a figura 18.

Figura 18 – Tela inicial aplicação *web*.

Fonte: Do autor.

Após o *login* o instrutor deve cadastrar os alunos da academia para que tenham acesso ao aplicativo Android, bem como o exercício e a sua descrição como mostra a figura 19.

Figura 19 – Telas de cadastro de usuário e exercício.

Consultar usuários	Cadastrar Exercícios	Cadastrar Treino	Voltar	Consultar/Editar exercícios	Cadastrar Usuários	Cadastrar Treino	Voltar
<p><b>Cadastro Usuário</b></p> <p>Nome: Informe o nome</p> <p>Login: Informe o login</p> <p>Senha: Informe a senha</p> <p>Salvar</p>				<p><b>Cadastro Exercício</b></p> <p>Exercício: Informe o nome do exercício</p> <p>Descrição: Informe a descrição do exercício</p> <p>Salvar</p>			

Fonte: Do autor.

Além de funções de cadastro a aplicação possui a consulta de todos os cadastros de usuário, exercício, tipo de treino e o treino prescrito. Estas informações são gravadas no banco de dados e são utilizadas nas consultas *web* e principalmente na consulta de treino através do aplicativo *mobile*, a classe no projeto *web* responsável pela conexão WebSocket é chamada de *WSEndpointTreino*, nela é possível receber o usuário que está usando o aplicativo bem como a ação executada por ele e através disto executar a ação para retornar a mensagem para o aplicativo.

A troca de informações é realizada através de mensagens, primeiro é iniciada a conexão pela função *onOpen* que adiciona o usuário na sessão, sendo possível receber a mensagem através da função *onMessage* junto com a sessão do usuário, esta função é utilizada para executar o que é enviado do aplicativo que passa na mensagem uma *TAG* que é a instrução do que fazer, como a autenticação do *login* do aluno no aplicativo, listar os tipos de treino do aluno bem como a lista de cada tipo de treino com os exercícios, e a ação de marcar o exercício como realizado e também deixar ele disponível para executar novamente. Por fim, esta classe pode finalizar a conexão tirando o usuário da sessão por meio da função *onClose*. A figura 20 mostra a classe descrita.

Figura 20 – Classe WebSocket do projeto *web*.

```

@ServerEndpoint("/treinoendpoint")
public class WSEndpointTreino {

    private static Set<Session> sessions = Collections.synchronizedSet(new HashSet<Session>());

    @OnOpen
    public void onOpen(Session session) {
        sessions.add(session);
        System.out.println(session.getId() + " has opened a connection");
    }

    @OnMessage
    public void onMessage(String message, Session session) {
        WebSocketMessage websocketMessage = JsonManager.fromJson(message, new TypeToken<WebSocketMessage>() {
        }.getType());
        String tag = websocketMessage.getTag();
        message = websocketMessage.getMessage();
        System.out.println("Message from: " + session.getId() + ": " + tag + " - " + message);
        //Verifica conforme TAG para saber qual ação executar.
        try {
            WebSocketMessage websocketMessageRetorno = new WebSocketMessage(tag);
            switch (tag) {
                case "LoginActivity": {
                    int indexPipe = message.indexOf("|");
                    String txtLogin = message.substring(0, indexPipe);
                    String txtPassword = message.substring(indexPipe + 1);
                    System.out.println("txtLogin: " + txtLogin);
                    System.out.println("txtPassword: " + txtPassword);
                    //TODO Validar Login
                    Usuario usuario = UsuarioDAO.verificaUsuario(txtLogin, txtPassword);
                    if (usuario == null) {
                        websocketMessageRetorno.setMessage("-1");
                    } else {
                        websocketMessageRetorno.setMessage(JsonManager.toJson(usuario));
                    }
                }
                break;
                case "TreinoActivity": {
                    int indexPipe = message.indexOf("|");
                }
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    @OnClose
    public void onClose(Session session) {
        sessions.remove(session);
        System.out.println(session.getId() + " fechou a conexao");
    }
}

```

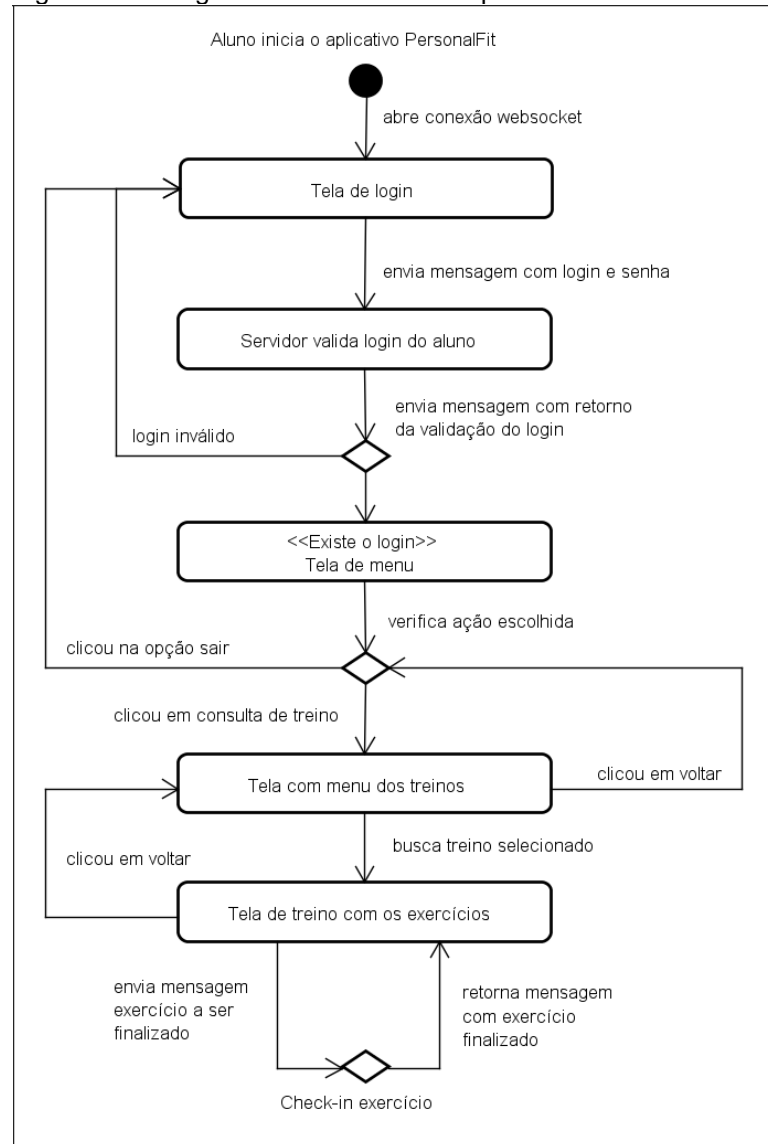
Fonte: Do autor.

Após a implementação da aplicação *web* foi iniciada a instalação do ambiente para desenvolvimento Android para o desenvolvimento do aplicativo PersonalFit, como descrito na próxima sessão.

### 6.1.6 Desenvolvimento Android

Realizada a instalação do ambiente de desenvolvimento Android Studio, foi iniciado o planejamento das telas e de suas funções, como deveria ser o fluxo das informações com base nos requisitos de funcionamento das aplicações. Desta forma chegou-se ao diagrama de atividade mostrado na figura 21.

Figura 21 – Diagrama de atividade do aplicativo PersonalFit.



Fonte: Do autor.

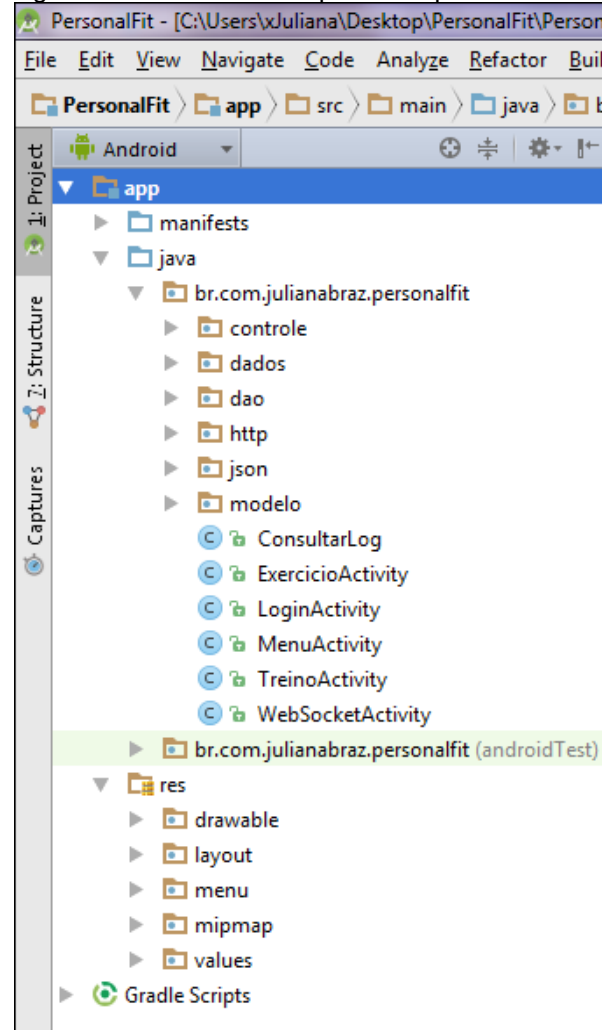
A comunicação entre as aplicações inicia com a execução do aplicativo PersonalFit, que abre uma conexão utilizando WebSocket. O usuário que neste caso é o aluno precisa informar o *login* e senha para que seja enviado uma *TAG* com estes dois atributos, onde a classe *WSEndpointTreino* da aplicação *web* recebe estes dados e adiciona o usuário na sessão, identifica que a *TAG* é referente a validação do *login* e executa a função que retorna se existe ou não o *login*, caso não exista é informado que o *login* não é válido, caso contrário carrega a tela de menu com as ações de consultar ou sair da aplicação. Neste ponto também é enviado o *log* com as informações gravadas temporariamente no celular, que são os dados do aparelho como fabricante, modelo, versão do Android, data, tipo de rede que está

usando, o sinal, o evento e a mensagem que é o conteúdo da TAG. O log com estas informações só é enviado quando o usuário usa a aplicação novamente, porém a conexão persiste até o final da aplicação, estes dados foram utilizados nos resultados obtidos com os testes.

A integração das aplicações para as demais telas de tipo de treino e consulta de treino são realizadas da mesma maneira pela mensagem enviada pela TAG com os dados para a consulta no servidor e o retorno da mesma.

A estrutura do projeto Android contempla classes em Java como o modelo, *activity* que são as telas e *xml* definido no pacote *layout* para o desenho das telas conforme mostra a figura 22.

Figura 22 – Estrutura dos pacotes aplicativo PersonalFit.



Fonte: Do autor.

Para aplicar o uso do WebSocket no aplicativo foi utilizada uma biblioteca chamada de Autobahn, esta possui código aberto e implementa o protocolo WebSocket como mostra a figura 23 com um trecho do código-fonte utilizando a biblioteca através da importação *de.tavendo.autobahn* e os métodos descritos anteriormente.

Figura 23 – Trecho do código-fonte classe WebSocketManager do aplicativo PersonalFit.

```

WebSocketManager.java x
import de.tavendo.autobahn.WebSocketConnection;
import de.tavendo.autobahn.WebSocketException;
import de.tavendo.autobahn.WebSocketHandler;
import de.tavendo.autobahn.WebSocketOptions;

public class WebSocketManager extends WebSocketHandler{

    private Context ctx;
    private static final String TAG = "WebSocketManager";
    private static final String WS_URI = "ws://191.101.224.207/treinoendpoint";

    private static WebSocketConnection mConnection;
    private static HashMap<String, WebSocketListener> hmListener = new HashMap<>();

    public WebSocketManager() {}

    public WebSocketManager init(Context ctx) {...}

    private static class SingletonHolder {...}

    @ public static WebSocketManager getInstance() { return SingletonHolder.INSTANCE; }

    private void connect() throws WebSocketException {
        WebSocketOptions options = new WebSocketOptions();
        options.setSocketConnectTimeout(30000);
        options.setSocketReceiveTimeout(30000);
        mConnection.connect(WS_URI, this, options);
        WebSocketLogManager.log(ctx, "Connect");
    }

    @Override
    public void onOpen() {
        Log.d(TAG, "Conexão iniciada - Android.");
        WebSocketLogManager.log(ctx, "Open");
    }

    @Override
    public void onClose(int code, String reason) {
        Log.d(TAG, "Conexão finalizada - Android. Motivo: " + code + " - " + reason);
        Toast.makeText(ctx, "Conexão com o servidor finalizada.", Toast.LENGTH_SHORT).show();
        WebSocketLogManager.log(ctx, "Close", code + " - " + reason);
        Intent i = ctx.getPackageManager().getLaunchIntentForPackage(ctx.getPackageName());
    }
}

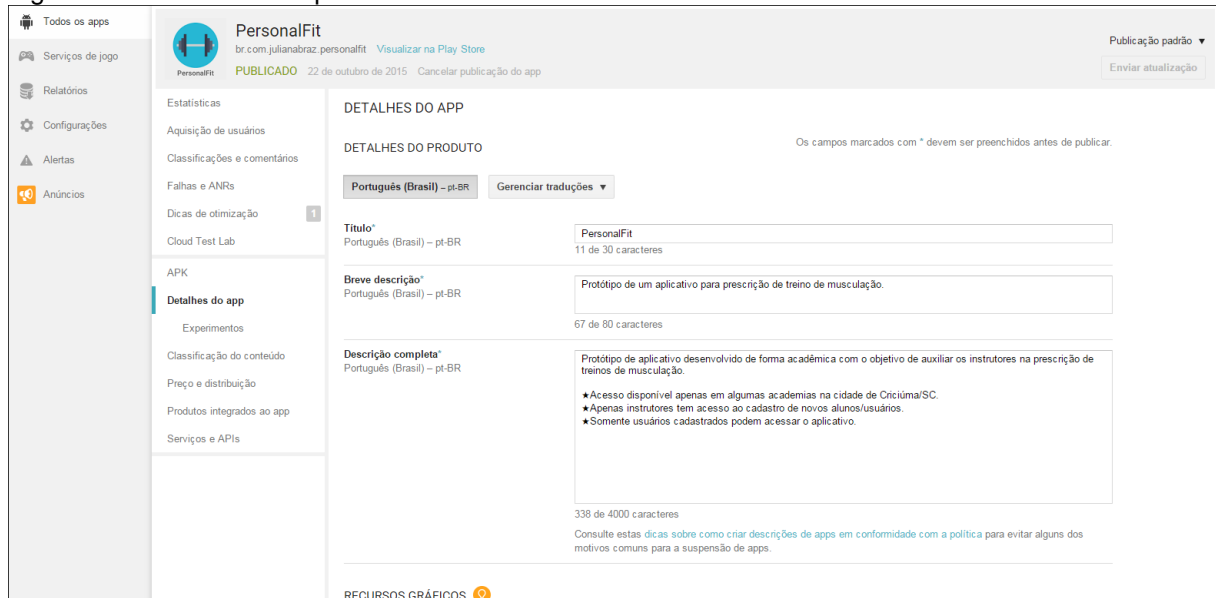
```

Fonte: Do autor.

Foi utilizado o banco de dados SQLite que permite a manipulação do banco de dados já embutido na aplicação, ou seja, não é preciso fazer a instalação de outra ferramenta, alternativa usada para gravar o *log* com os dados da conexão WebSocket, este *log* é enviado somente uma vez por acesso ao aplicativo, que é no início logo após a verificação do *login* do usuário. Para a transformação na troca de informações entre as aplicações foi usado o *Javascript Object Notation* (JSON), que permite o envio de atributos e objetos de uma aplicação para a outra através da biblioteca *com.google.gson*.

Após o desenvolvimento do aplicativo foi associada a conta de e-mail do Gmail como conta de desenvolver Android, permitindo assim a publicação da aplicação na Google *Play* através do console do desenvolvedor. Para a publicação deve ser seguido os passos e as regras que a Google solicita, como a aplicação ter um nome, bem como o pacote criado no desenvolvimento com um nome único neste caso foi usado *br.com.julianabraz.personalfit*, além de selecionar em que sessão a aplicação se enquadra, como saúde e fitness, deve também conter um título, descrição da aplicação e informar se existe censura de idade para o uso do aplicativo, como mostra a figura 24.

Figura 24 – Detalhes do aplicativo PersonalFit no console de desenvolvedor.



Fonte: Google (2015).

Contudo, deve-se apresentar alguns recursos gráficos para facilitar a distribuição na loja de aplicativos, que são na verdade imagens das telas da aplicação, como é ilustrado na figura 25.

Figura 25 – Sequência de telas do aplicativo PersonalFit.



Fonte: Do autor.

Com o aplicativo disponível na Google *Play* é possível que os usuários façam o *download* em dispositivos que possuam o sistema Android com versão igual ou superior a 4.3. A versão foi limitada pois o método utilizado para verificar o tipo de conexão realizada pelo dispositivo está disponível apenas da versão 4.3 em diante. Desta forma, também foi publicada a aplicação *web* em um servidor previamente configurado com o Tomcat, que disponibiliza a ferramenta Tomcat *Web Application Manager* para implantar a aplicação, ou seja, torná-la disponível através de uma URL.

Entretanto foi escolhido um ambiente real para que fossem efetuados os testes a fim de obter um resultado mais próximo do real possível, em que os recursos utilizados para executar as aplicações fossem mais variados como a conexão dos aparelhos em redes *WIFI* e 3G, além de obter um maior número de celulares, versões do sistema Android e até mesmo opiniões de uso da aplicação, sugestões de melhorias para que as aplicações fiquem mais utilizáveis dentro deste ambiente.

## 6.2 TESTES EM UM CENÁRIO REAL DE ACADEMIA DE MUSCULAÇÃO

Antes de colocar as aplicações disponíveis para testes em um ambiente real, neste caso as academias, foram simulados os testes através do próprio Android Studio via cabo *Universal Serial Bus* (USB) para executar diretamente em um aparelho celular Motorola, com versão do Android 5.1 e modelo XT1033. Após executar os testes e analisar que o comportamento das aplicações estava de acordo com o esperado, foram liberados os testes no ambiente real.

Os testes das aplicações foram feitos em quatro cenários diferentes, ou seja, em quatro academias de musculação na cidade de Criciúma, Santa Catarina, sendo elas: Via Corpo I, Via Corpo II, Enforce Fitness e Academia Super Action. Em cada academia foi realizada uma reunião com os responsáveis para a apresentação do projeto e autorização da realização dos testes dentro de cada ambiente, visto que além de disponibilizar as aplicações, também é preciso a colaboração de instrutores e alunos dispostos a instalarem a aplicação *mobile*. Os termos de consentimento assinados pelos responsáveis das academias Enforce Fitness, Via Corpo II, Super Action e Via Corpo I estão nos apêndices A, B, C e D, respectivamente.

O primeiro ambiente a iniciar os testes das aplicações foi a Academia Enforce Fitness no dia vinte (20) de outubro até o dia seis (6) de novembro. O projeto foi apresentado para dois (2) responsáveis, sendo um deles também instrutor, contudo foi apresentado para três (3) instrutores. Durante a demonstração da aplicação mostraram-se interessados pela ideia, porém foi apontado que para que fosse realmente abrangente deveria estar disponível para celular com o sistema operacional iOS, baseado em seus alunos, que optam por usar aparelhos desta plataforma. Desta forma, também foram apresentados dois banners para o incentivo e divulgação dos testes dentro da academia, o primeiro é para uso do instrutor, para lembrá-lo dos passos a serem seguidos na aplicação *web*, entende-se que por mais simples que seja o uso da aplicação o instrutor também precisa adaptar-se ao uso dela. Além de um banner informando do que se trata o aplicativo PersonalFit e de que maneira ele deve ser usado, efetuando o *download* na *Play Store* e solicitando um usuário para o instrutor, estes banners estão nos apêndices E e F, respectivamente. Mesmo com poucos alunos que dispunham dos dispositivos

necessários para teste, os instrutores conseguiram utilizar a aplicação *web* e fazer com que alunos utilizassem o aplicativo *mobile*.

A segunda academia a adotar uso das aplicações para os testes deste projeto foi a Academia Via Corpo II, que inicialmente mostrou-se muito interessada na ideia, apontando que é uma boa sugestão para a troca do atual modo de prescrever o treino e disponibilizar para o seu aluno. O atual método consiste em realizar prescrever o treino em uma ficha de papel com informações como o nome do exercício, a série, repetições e em alguns casos o peso, conforme mostra a figura 26.

Figura 26 – Ficha de prescrição do treino.

PROGRAMA DE TREINAMENTO															Sua mensalidade vence sempre								
INÍCIO: / /		CODINOME:													dia _____ de cada mês.								
OBJETIVO:					MÉTODO:					SISTEMA:													
	Nº	EXERCÍCIO	S	REP	KG		Nº	EXERCÍCIO	S	REP	KG		Nº	EXERCÍCIO	S	REP	KG						
A B D O M E N		FLEX. COLUNA				T R O N C O		ELEVAÇÃO LATERAL				M E M B R O S		LEG PRESS									
		FLEX. DO QUADRIL						ELEVAÇÃO FRONTAL							AGACHAMENTO								
		FLEX. LAT. TRONCO							DESENV. P/ FRENTE							AGACH. ABERTO							
		CRUNCH							DESENV. P/ TRÁS							EXT. DOS JOELHOS							
P E I T O		SUPINO RETO				M E M B R O S		DESENV. C/ HALTERES				I N F E R I O R E S		FLEX. DOS JOELHOS									
		SUPINO INCLINADO						REMADA EM PÉ							STIFF								
		SUPINO FECHADO							ENC. DOS OMBROS							ADUÇÃO DO QUADRIL							
		CRUCIFIXO RETO					B I C E P S		ROSCA SCOTT					T R I C E P S									
		CRUCIFIXO INCLINADO							ROSCA DIRETA														
		PECK DECK							ROSCA ALT. EM PÉ														
		CROSSOVER							ROSCA ALT. SENTADO														
		PULL-OVER							ROSCA CONCENTRADA														
	T R O N C O		REMADA BAIXA						S U P E R I O R E S		ROSCA MARTELO						T R I C E P S		ABDUÇÃO DO QUADRIL				
			REMADA ALTA								ROSCA NA POLIA BXA										EXT. DO QUAD. POLIA		
		REM. BI. ABDUZIDA								TRICEPS POLIA ALTA								EXT. DO QUAD. EM PÉ					
		REM. CURVADA								ROSCA FRANCESA								EXT. QUAD. 4 APOIOS					
		PUXADA P/ FRENTE								TRICEPS TESTA								EXT. DO QUADRIL MÁQ.					
C O S T A S			PUXADA P/ TRÁS				A N T E B R			KICK BECK				P E R N A S		AVANÇO SUSPENSO							
			SERROTE							TRICEPS NO BANCO								AVANÇO					
			BARRA FIXA								ROSCA INVERSA								FLEX. PLANTAR EM PÉ				
			CRUCIFIXO INVERSO								ROSCA PUNHO								FLEX. PLANTAR SENT.				
			LEVANT. TERRA								ROSCA PUNHO INV.												
		EXTENSÃO DA COLUNA																					
	CRONOGRAMA DE TREINO		SEGUNDAS	TERÇAS	QUARTAS	QUINTAS		SEXTAS	ESTEIRA:		BICICLETA:		ELIPTICO:										

Fonte: Academia Via Corpo<sup>12</sup> (2015).

Contudo foi apresentado a aplicação *web* que é utilizada pelo instrutor para realizar os cadastrados informados anteriormente além do principal que é o cadastro de treino como mostra a figura 27, este método foi apresentado para um instrutor na academia Via Corpo II que ficou responsável de conversar com os demais, este participou da reunião juntamente com as responsáveis pela academia e

<sup>12</sup> Documento não publicado e de autoria da academia Via Corpo, 2015.

mostrou bastante interesse, apontando também a sugestão de implantação para a plataforma iOS, além de uma opção para informar a ordem em que deve ser executado os exercícios. Informou também que é uma ótima opção para quem é *personal trainer*, sendo que o instrutor que acompanha o aluno de perto e prescreve o treino mais específico, e também para profissionais que prescrevem o treino a distância, até mesmo de outra cidade. Os testes tiveram início no dia vinte e um (21) de outubro até o dia seis (6) de novembro de 2015.

Figura 27 – Tela de cadastro de treino.

Exercício	Descrição	Séries		Repetições
Agachamento Sumo	Afaste as pernas e abra os joelhos e pes para fora, permitindo que o quadril vá para trás ao agachar.	<input type="text"/>	x	<input type="text"/>
Supino	Deite-se no banco do aparelho levante a barra em direção ao teto..	<input type="text"/>	x	<input type="text"/>
Agachamento	Abaixe-se como se fosse sentar em um banco até que os joelhos fiquem em um ângulo de 90 graus.	<input type="text"/>	x	<input type="text"/>
Abdominal	Deite-se de barriga para cima erga os ombros em direção ao teto, tirando-os do chão.	<input type="text"/>	x	<input type="text"/>
Esteira	Caminhar ou correr na esteira	<input type="text"/>	x	<input type="text"/>

Fonte: Do autor.

A academia Super Action foi a terceira a iniciar os testes a partir do dia vinte e dois (22) de outubro até o dia seis (6) de novembro. Foi realizada a reunião com os proprietários da academia que também são os instrutores, estes apresentaram bastante interesse no método de prescrição apresentado, principalmente por não ter que utilizar folhas com a descrição dos treinos, que é a atual maneira de uso. Durante a apresentação de funcionamento das aplicações alguns alunos passaram pelo local e mostraram interesse em utilizar as aplicações. Entretanto, também foi sugerido como nas outras academias o funcionamento do aplicativo *mobile* na plataforma iOS.

A quarta e última academia a realizar os testes foi a Via Corpo I que iniciou no dia vinte e oito (28) de outubro até o dia seis (6) de novembro. A reunião iniciou com a apresentação do funcionamento para a responsável pela academia e logo após para dois (2) instrutores, estes ficaram responsáveis de mostrar para os outros instrutores da academia devido ao horário dos turnos. Tendo em vista que a apresentação se fez no horário mais cômodo para os instrutores cadastrarem os alunos, exercícios e treinos, que foi estabelecido pela proprietária da academia. O método de prescrição de treino é a mesma da academia Via Corpo II utilizando a mesma ficha de treinamento mostrado anteriormente na figura 26. Os instrutores acharam a ideia interessante, porém apontaram o uso da plataforma iOS de alguns alunos, além da dificuldade de ter um computador disponível para o uso da aplicação *web*, o computador disponível fica na recepção o que torna inviável o uso durante o movimento intenso, mesmo que a aplicação seja *web* sendo possível o uso em aparelhos de celulares.

Portanto para a aplicação dos testes em um ambiente real foi disponibilizado as aplicações de forma gratuita para as academias envolvidas, bem como sem custos acadêmicos. As academias disponibilizaram a conexão *WIFI* para seus alunos, que por sua vez também não geraram custos para nenhuma parte envolvida. O período de testes no ambiente de academia de musculação foi do dia vinte de outubro (20) até o dia seis (6) de novembro de 2015, buscou-se mais de uma academia para obter um número maior de usuários envolvidos, entendendo que nem todos estão dispostos a utilizar as aplicações que precisa de uma atenção para fazer o *download* do aplicativo além da prescrição do treino pelo instrutor.

### 6.3 RESULTADOS OBTIDOS

Contando com todas as apresentações de funcionamento nas academias dispostas a utilizarem a aplicação para os testes deste projeto foi possível chegar em alguns números apontados nesta sessão.

O número de cenários ou ambientes em que foram realizados os testes foram quatro (4) academias, o projeto foi apresentado para oito (8) instrutores e sete (7) responsáveis e proprietários dos locais. A aplicação *mobile* PersonalFit teve trinta e dois (32) *downloads* dentro do período entre dia vinte (20) de outubro a seis (06)

de novembro de 2015. Porém nem todos que fizeram o *download* conseguiram usar o aplicativo, devido a regra que foi estabelecida que só teria um usuário e senha para realizar o *login* no aplicativo quem o instrutor da academia cadastrasse através da aplicação *web*, pois a intenção é deixar somente o profissional qualificado responsável por prescrever os treinos, então o usuário que não frequenta nenhuma das academias na região de Criciúma: Enforce Fitness, Via Corpo I, Via Corpo II ou Academia Super Action, não possui um *login*. Sendo assim cinco (5) pessoas tentaram efetuar o *login* e não possuíam usuários cadastrados. Esta informação foi retirada da tabela *websocketlog* que contém as informações da conexão do aplicativo, como ilustra a figura 28, um usuário que tenta entrar no aplicativo e não tem um *login* na tabela, o número do usuário permanece como zero (0) e o servidor retorna uma mensagem com o código “-1”, ou seja, não encontrou o usuário e senha informado.

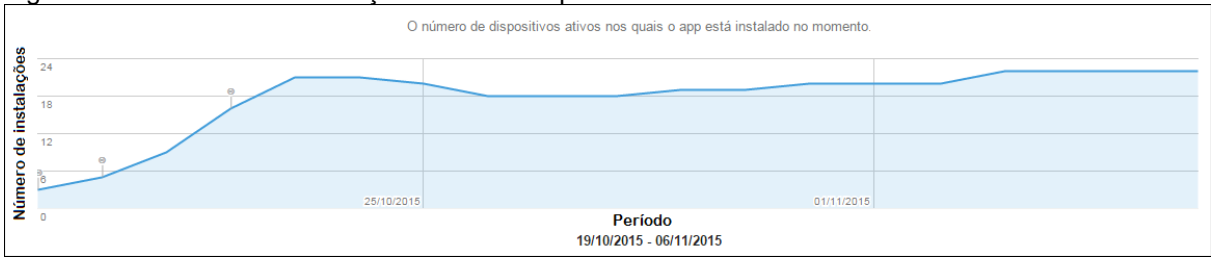
Figura 28 – Informações da tabela *websocketlog* do banco de dados.

idusuario	dtmanutencao	idevento	redetipo	redesinal	dispositivoandroid	dis dispositivomodelo	dis dispositivofabricante	mensagem
0	1446369039718	Send Init	WIFI	-50	18	SM-G3502T	samsung	{"message":"sakanoshiro2006@outlook.com c123sa04si..."}
0	1446369039731	Send Finish	WIFI	-50	18	SM-G3502T	samsung	NULL
0	1446369040444	Receive	WIFI	-51	18	SM-G3502T	samsung	{"tag":"LoginActivity","message":"-1"}
0	1446369057937	Connect	WIFI	-62	18	SM-G3502T	samsung	NULL
0	1446369058350	Open	WIFI	-62	18	SM-G3502T	samsung	NULL
0	1446369063994	Send Init	WIFI	-65	18	SM-G3502T	samsung	{"message":"","tag":"LoginActivity"}
0	1446369064007	Send Finish	WIFI	-65	18	SM-G3502T	samsung	NULL
0	1446369064103	Send Init	WIFI	-65	18	SM-G3502T	samsung	{"message":"","tag":"LoginActivity"}
0	1446369064113	Send Finish	WIFI	-65	18	SM-G3502T	samsung	NULL
0	1446369064276	Receive	WIFI	-65	18	SM-G3502T	samsung	{"tag":"LoginActivity","message":"-1"}

Fonte: Do autor.

Baseado neste fato, nem todos que realizaram o *download* do aplicativo permaneceram com ele instalado durante o período de testes este número de usuários ativos mudou praticamente todos os dias, até pelo fato de que cada academia começou a utilizar as aplicações em dias diferentes, o gráfico apresentado na figura 29 mostra o número de usuários que permanecem com o aplicativo instalado em seus dispositivos por dia.

Figura 29 – Gráfico das instalações ativas do aplicativo PersonalFit.



Fonte: Google (2015).

O gráfico aponta que os dias com maior número de instalações ativas foram do dia três (3) ao dia (6) de novembro com o total de vinte e duas (22) instalações.

Destes usuários que permanecem com a instalação do aplicativo podemos saber quais as versões do Android que seus dispositivos possuem, como mostra a figura 30. Do total de instalações atuais, que é de vinte e dois (22) dispositivos, estão divididos nas versões: Android 4.4 com nove (9) instalações, Android 5.1 em (5) dispositivos e em terceiro lugar as versões do Android 4.3 e 5.0 ambos com quatro (4) instalações cada.

Figura 30 – Gráfico de instalações atuais por versão do Android.

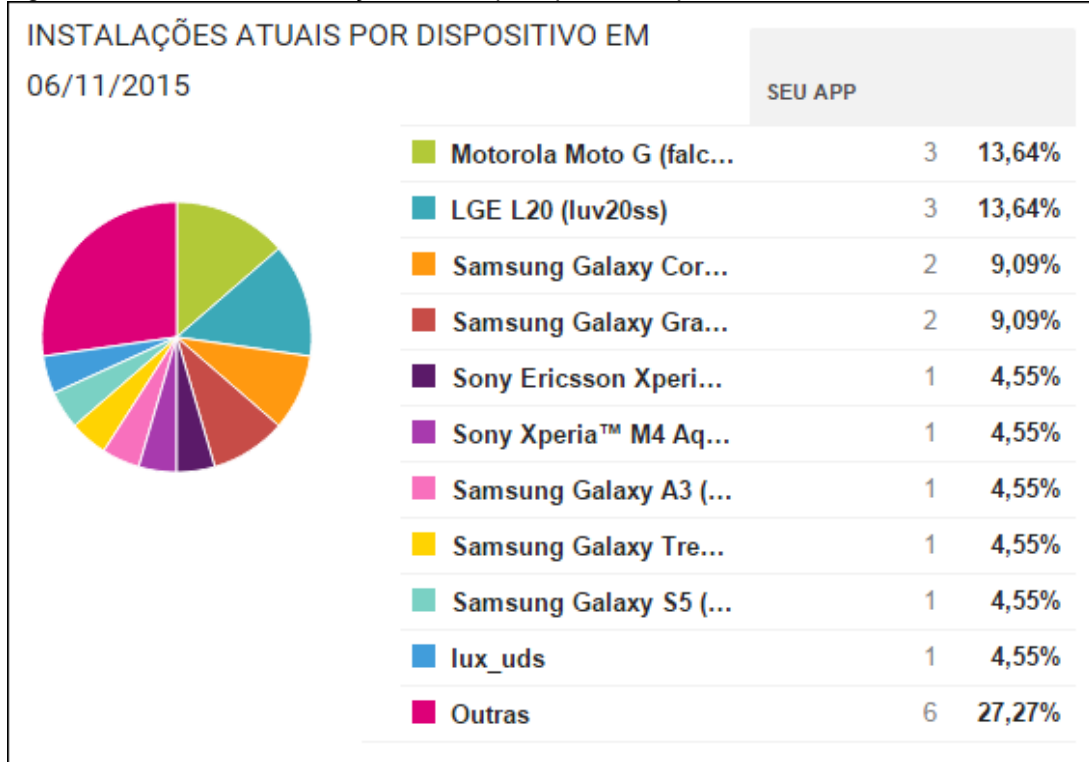


Fonte: Google (2015).

Também é possível visualizar cada aparelho de celular que foi utilizado para fazer o *download* do aplicativo como mostra a figura 31. Sendo que o maior número de usuários possui dispositivos do fabricante Samsung com sete (7) usuários, logo após vem aparelhos da Motorola com três (3) usuários juntamente

com LG que possui o mesmo número de três (3) usuários, seguido da fabricante Sony com dois (2) usuários, as demais fabricantes somam sete (7) usuários.

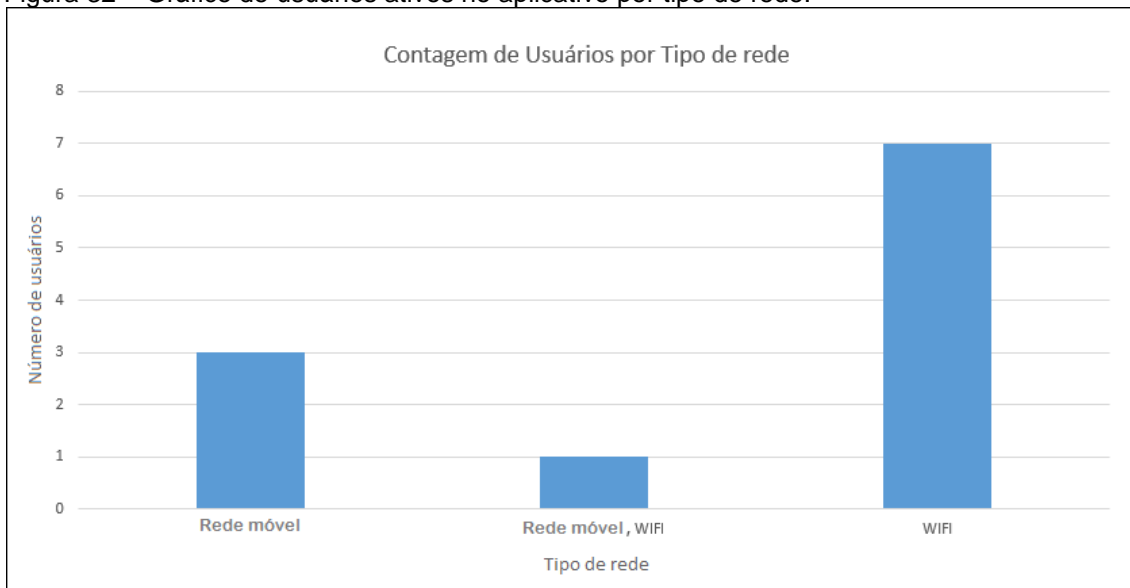
Figura 31 – Gráfico de instalações atuais por tipos de dispositivos.



Fonte: Google (2015).

Os dados informados anteriormente foram retirados das estatísticas apresentadas pelo console de desenvolvedor da Google *Play* que representa usuários que tem o aplicativo em seus dispositivos, que não é necessariamente ativo no uso da aplicação. Porém ao analisar a tabela *websocketlog* podemos chegar a alguns números que serão representados por usuários que utilizam realmente a aplicação. Como mostra a figura 32, representada pelo número de usuários que utilizaram a aplicação com os tipos de rede *WIFI*, 3G ou ambas as conexões durante o período de testes, ou seja, não utilizou sempre a mesma conexão, um dia utilizou *WIFI* e no outro 3G.

Figura 32 – Gráfico de usuários ativos no aplicativo por tipo de rede.

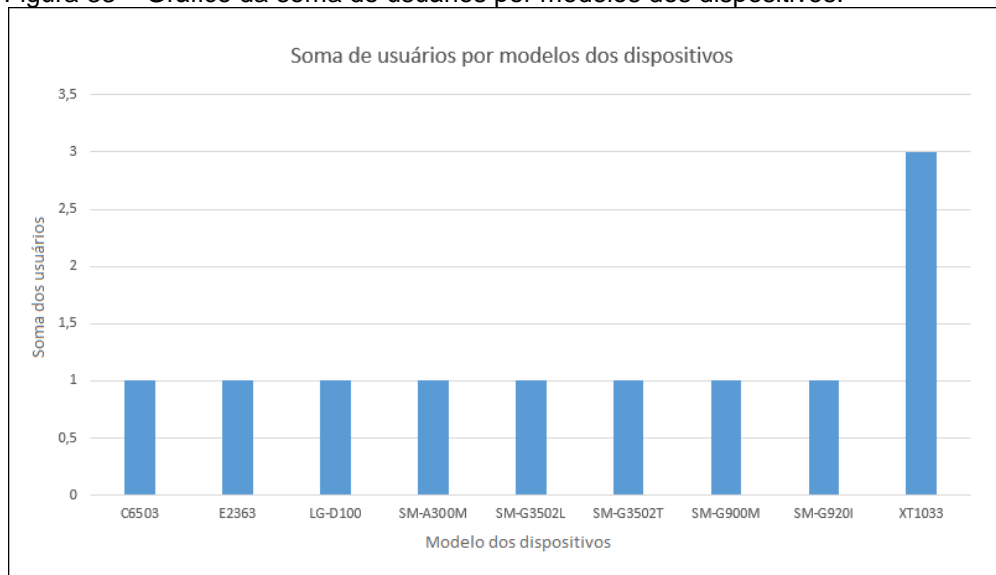


Fonte: Do autor.

O número de usuários que utilizaram o aplicativo PersonalFit foi de onze (11) durante o período de teste, a partir do gráfico da figura 32 foi possível visualizar que a maior parte utilizou a rede *WIFI* para conectar-se à Internet e ter acesso aos treinos, cerca de sete (7) pessoas, três (3) pessoas utilizaram redes móveis e por fim uma (1) pessoa utilizou ambas conexões. Partindo do ponto que houveram pessoas que optaram por utilizarem em redes móveis e *WIFI*, os dois modos de rede apresentaram um bom funcionamento, porém quando o usuário estava conectado através de uma rede móvel, o aplicativo demora um pouco mais para carregar as informações, mas mesmo assim funcionando e conseguindo trazer todas as informações do servidor.

A experiência com os testes nas academias proporcionou que usuários utilizassem aparelhos de celulares diferentes assim como as conexões de redes apontadas anteriormente, o aplicativo teve um bom funcionamento em todos os nove (9) modelos de celulares apresentados no gráfico da figura 33, os aparelhos apresentados são dos fabricantes Samsung, Sony, LG e Motorola.

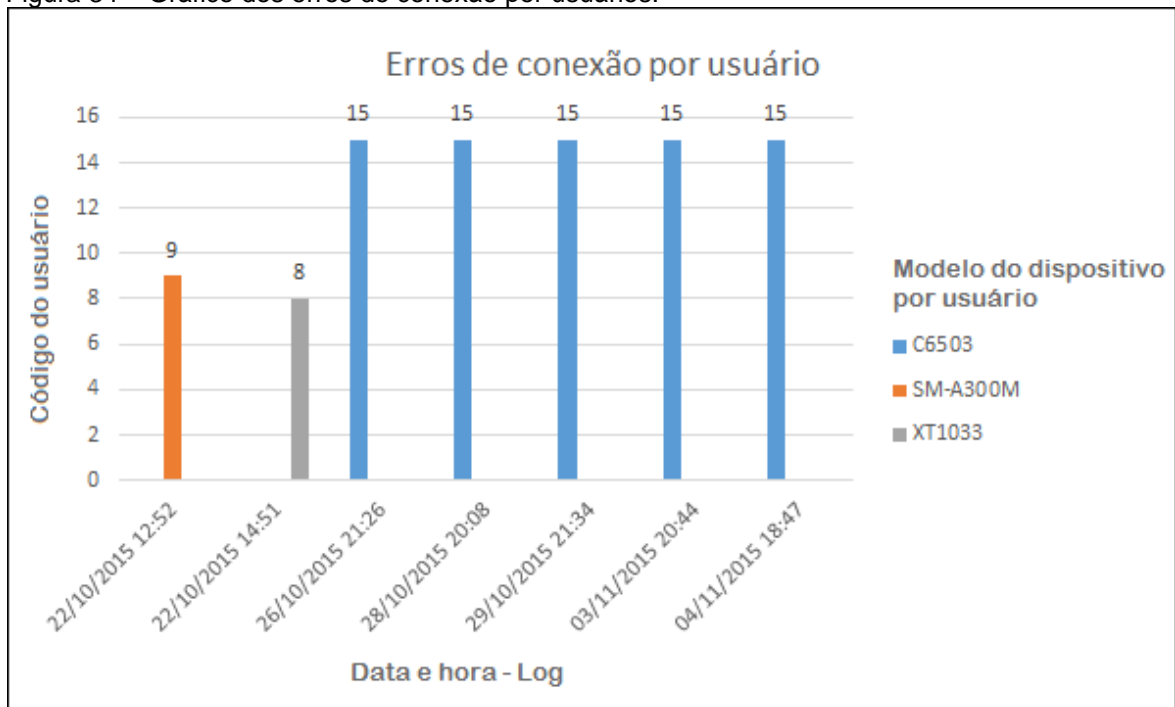
Figura 33 – Gráfico da soma de usuários por modelos dos dispositivos.



Fonte: Do autor

Durante o período de testes nas academias foi possível visualizar que três (3) usuários perderam a conexão com o servidor, pois ficaram sem rede para o acesso à Internet, sendo esta informação gravada no banco de dados possibilitando assim a geração do gráfico demonstrado na figura 34. Apesar de terem perdido a conexão, que neste caso a aplicação volta para a tela de *login* para que seja possível entrar novamente e continuar do ponto que estava antes de perder a conexão, os usuários conseguiram utilizar a aplicação. No caso destes usuários que perderam a conexão foram utilizados para executar o aplicativo os dispositivos XT1033 do fabricante Motorola, o aparelho SM-A300M da Samsung e C6503 da Sony.

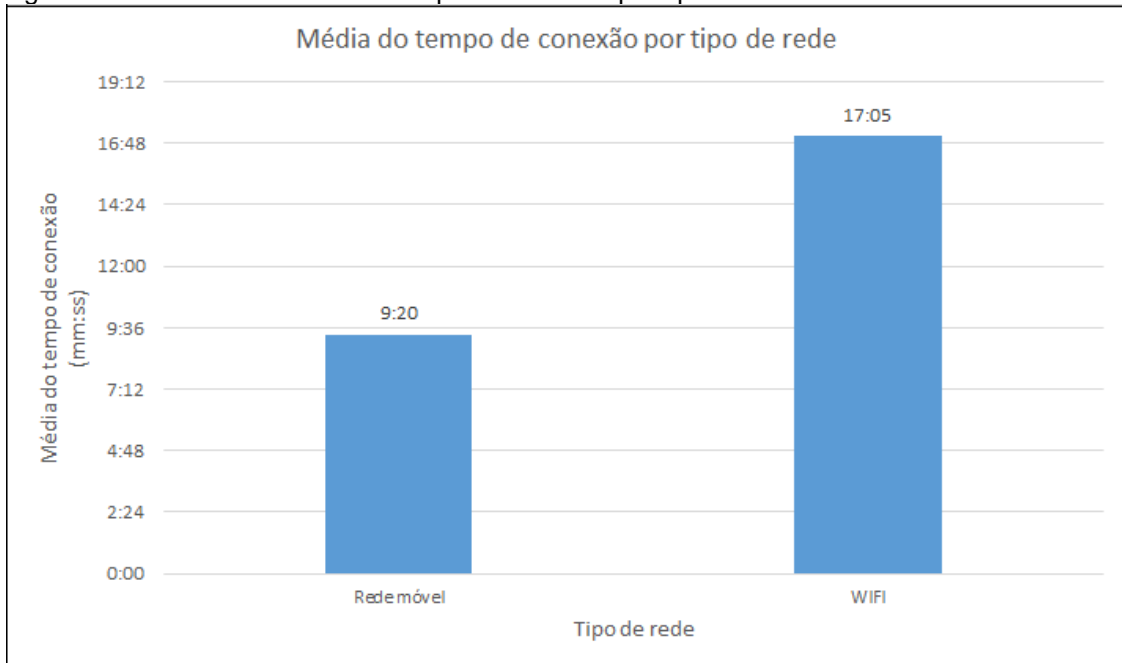
Figura 34 – Gráfico dos erros de conexão por usuários.



Fonte: Do autor.

As informações de data e hora do início e fim da conexão feita pelo usuário são importantes para definir se o usuário está conseguindo manter a conexão ativa durante seu treino, além de conseguir criar uma média do tempo de conexão juntamente com outras informações como o tipo de rede utilizada, como demonstra a imagem 35 a média de tempo das conexões por tipo de rede. A rede mais utilizada foi *WIFI* visto que as academias disponibilizam para seus alunos de forma gratuita, além de possuir uma velocidade melhor, facilitando o uso do protótipo, com base nesta informação e nos dados do servidor é possível afirmar que a média de conexão da rede *WIFI* é maior sendo de aproximadamente dezessete (17) minutos e nove (9) minutos e vinte (20) segundos nas redes móveis.

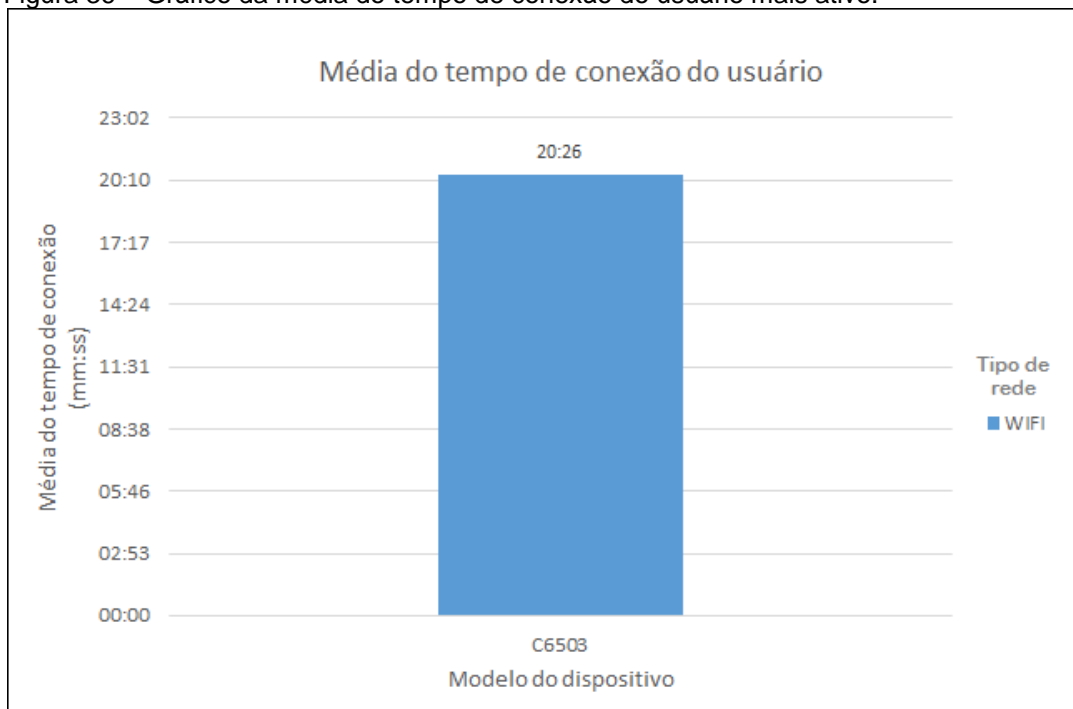
Figura 35 – Gráfico da média do tempo de conexão por tipo de rede.



Fonte: Do autor.

Para obter uma média por usuário foi verificado qual aluno utilizou mais o aplicativo e realizado o levantamento de dados das suas conexões, bem como o modelo do dispositivo, fabricante e o tipo de rede para conexão com a Internet. Permitindo assim a criação do gráfico da figura 36.

Figura 36 – Gráfico da média do tempo de conexão do usuário mais ativo.



Fonte: Do autor.

Através do gráfico da figura 36 é possível afirmar que a média de conexão do usuário mais ativo na aplicação é de vinte (20) minutos e vinte e seis (26) segundos utilizando a rede *WIFI* do dispositivo C6503, cujo fabricante é Sony.

Entretanto o aplicativo obteve o comportamento esperado mantendo a conexão ativa através das redes móveis e *WIFI*, bem como o funcionamento em dispositivos com modelos e versões do Android diferentes. Porém quando o celular perde o sinal do *WIFI* ou rede móvel a aplicação finaliza a conexão com o servidor via WebSocket causando no aplicativo o retorno para a tela inicial de *login* e o tratamento do fechamento da conexão pelo servidor.

Contudo para o melhor funcionamento dentro do ambiente real de musculação foram sugeridas melhorias em relação ao aplicativo *mobile* que estão na próxima sessão.

### **6.3.1 Sugestões de melhorias**

Durante a fase de testes do protótipo dentro do ambiente das academias de musculação na cidade de Criciúma/SC, os instrutores das academias e seus respectivos responsáveis mostraram interesse no funcionamento da prescrição de treino através das aplicações, contudo sugeriram algumas melhorias para que o aplicativo ficasse mais adequado à realidade destes ambientes, são elas:

- a) inclusão de um campo para ordenar os exercícios dentro do treino;
- b) possibilidade de relacionamento de mais de um exercício por vez, no caso dos treinos chamados *bi-sets* em que o aluno executa dois exercícios intercalados até terminar a série do exercício;
- c) disponibilizar o aplicativo *mobile* em mais de uma plataforma, principalmente para o sistema operacional iOS.

## 7 CONCLUSÃO

Este projeto apresentou duas aplicações para o auxílio na prescrição de treino de musculação, sendo uma *web*, utilizada pelo instrutor de musculação e outra para a consulta dos treinos prescritos através de um aplicativo *mobile* para dispositivos com o sistema operacional Android utilizada pelos alunos praticantes das atividades físicas, a fim de aplicar a tecnologia WebSocket em um ambiente real com o objetivo de testar a solução apresentada e obter dados do comportamento do WebSocket.

Durante o desenvolvimento do projeto surgiram algumas dificuldades, como ao pesquisar sobre a comunicação WebSocket, visto que existe mais *blogs* explicando o que é o WebSocket do que exemplos reais de funcionamento e bibliografias mais confiáveis, como livros, artigos e outros projetos já realizados, porém com maior esforço conseguiu-se obter mais materiais de referências sobre o assunto. Ao desenvolver a aplicação *mobile* não foi possível executar o emulador do Android no ambiente de desenvolvimento pelo motivo de incompatibilidade com o computador utilizado, portanto durante o desenvolvimento foi realizado o uso do cabo USB junto com um aparelho celular do modelo XT1033 para executar os testes e simulações de funcionamento. Após o desenvolvimento do protótipo foram iniciados os testes em um ambiente real que ocasionou o encontro de mais algumas dificuldades como a disponibilidade de tempo do instrutor para realizar os cadastros da aplicação e prescrição de treino bem como alunos dispostos a instalarem o aplicativo PersonalFit em dispositivos móveis com sistema operacional Android, pois nem todos tinham aparelhos com este sistema operacional, bem como algumas academias também não dispunham de um computador para a utilização da aplicação *web*.

Mesmo com as dificuldades apontadas anteriormente foi possível chegar aos objetivos estabelecidos no projeto, que por sua vez foram atendidos através de pesquisas realizadas para obter o conhecimento necessário para desenvolver as aplicações *web*, Android e também disponibilizar o protótipo desenvolvido em um ambiente real de academia de musculação para a aplicação dos testes. E principalmente chegar na integração das duas aplicações, que foram estabelecidas

com sucesso, permitindo a troca de dados entre elas com a utilização da tecnologia WebSocket conforme proposto no projeto.

A pesquisa realizada para compreender a metodologia de uso na prescrição dos treinos de musculação foi importante para estabelecer o funcionamento das aplicações, sendo uma para o instrutor informar os dados necessários para a prescrição dos treinos dos usuários e outra para que cada aluno consulte os seus treinos. Foram realizadas também pesquisas sobre o Android a fim de entender como funciona o desenvolvimento dos aplicativos para a plataforma, o ciclo de uma *activity*, como criar telas, funções e estabelecer a conexão com o banco de dados, além de entender quais são as ferramentas necessárias para o seu desenvolvimento e o que é preciso para a publicação de aplicativos na Google Play, a fim de utilizar este recurso para facilitar nos testes em um ambiente real. Desta forma também foram realizadas pesquisas sobre a tecnologia WebSocket a fim de buscar o conhecimento para aplicar o uso do protocolo WebSocket nas aplicações com o objetivo iniciar uma conexão e permanecer com ela ativa para a troca de informações entre as aplicações, diminuindo o tempo de resposta das requisições.

A conexão utilizando o protocolo WebSocket foi estabelecida com sucesso ao executarem as aplicações quando a conexão com a Internet estava ativa, por meio de redes móveis ou *WIFI*. A comunicação feita pelo WebSocket foi perdida somente quando os usuários se desconectavam das redes móveis ou *WIFI*, visto que em alguns algumas academias o sinal da *WIFI* não funciona em todo o ambiente.

Através dos testes aplicados em quatro (4) academias na cidade de Criciúma, Santa Catarina, chegaram-se ao número de onze (11) usuários que utilizaram o aplicativo dentro das academias com o uso das redes móveis e/ou *WIFI*, além de modelos de dispositivos diferentes, e fabricantes distintos. Durante o período de testes, que ocorreram do dia vinte (20) de outubro até seis (6) de novembro, o aplicativo obteve um bom funcionamento mantendo a conexão e a sua funcionalidade como era esperado. Contudo, durante a fase de testes alguns instrutores de musculação apontaram sugestões e melhorias que poderiam acrescentar na funcionalidade dentro de uma academia de musculação, o que mais foi sugerido foi a disponibilidade da aplicação em dispositivos de outras plataformas, principalmente com o sistema operacional iOS.

Por fim, através do estudo e conhecimento adquirido durante o desenvolvimento do projeto pode-se sugerir como extensões deste trabalho para que seja realizada novas pesquisas e trabalhos futuros:

- a) realizar testes em larga escala para verificar o comportamento da comunicação WebSocket e com serviços que precise de mais demanda de informações;
- b) realizar testes com WebSocket em serviços considerados críticos que a comunicação em tempo real seja quase que um item essencial;
- c) utilização de um *framework* multiplataforma para disponibilizar o aplicativo para dispositivos além da plataforma Android;
- d) aplicar o uso de ferramentas gráficas como imagens e vídeos para utilizar na explicação dos exercícios;
- e) aplicar o uso de bibliotecas para geração de gráficos e relatórios sobre os treinos.

## REFERÊNCIAS

AABERG, Everett. **Musculação, biomecânica e treinamento**. São Paulo: Manole Ltda, 2001. 217 p.

ABAD, César Cavinato Cal et al. Efeito do exercício aeróbico e resistido no controle autonômico e nas variáveis hemodinâmicas de jovens saudáveis. **Revista brasileira de educação física do esporte**, São Paulo, v.24, n.4, p.535-544, out./dez.2010. Disponível em: <[http://www.scielo.br/scielo.php?pid=S1807-55092010000400010&script=sci\\_arttext](http://www.scielo.br/scielo.php?pid=S1807-55092010000400010&script=sci_arttext)>. Acesso em: 14 set. 2014.

BARBANTI, Valdir J. **Treinamento físico: bases científicas**. 3.ed. São Paulo: CLR Balieiro, 1996.

\_\_\_\_\_. **Teoria e prática do treinamento esportivo**. 2.ed. São Paulo: Editora Edgard Blücher Ltda, 1997.

\_\_\_\_\_. Prescrição de exercícios físicos. In: Escola de Educação Física e Esporte de Ribeirão Preto da Universidade de São Paulo, 2012. **Anais eletrônicos...** São Paulo: EEFERP, 2012. Disponível em: <<http://www.eeferp.usp.br/paginas/docentes/Valdir/pef.pdf>>. Acesso em: 15 set.2014.

BOMPA, Tudor O. **A periodização no treinamento esportivo**. São Paulo: Manole Ltda, 2001. 259 p.

BÜNDCHEN, Daiana Cristine et al. Exercício físico controla pressão arterial e melhora qualidade de vida. **Revista brasileira de medicina do esporte**, São Paulo, v. 2, n. 19, p.91-95, mar/abr. 2013. Disponível em: <[http://www.scielo.br/scielo.php?pid=S1517-86922013000200003&script=sci\\_arttext](http://www.scielo.br/scielo.php?pid=S1517-86922013000200003&script=sci_arttext)>. Acesso em: 03 nov. 2014.

CAELUM. **O que é Java EE?** 2012. Disponível em: <<http://www.caelum.com.br/apostila-java-web/o-que-e-java-ee/>>. Acesso em: 20 nov. 2014.

CÂMARA, Lucas Caseri et al. Exercícios resistidos terapêuticos para indivíduos com doença arterial obstrutiva periférica: evidências para a prescrição. **Jornal Vascular Brasileiro**. Porto Alegre, p. 247-257. set. 2007. Disponível em: <<http://www.scielo.br/pdf/jvb/v6n3/v6n3a08>>. Acesso em: 22 out. 2015.

CAMPANINI, André. **WebSocket no Java e Java EE: da especificação a exemplos**. 2012. Disponível em: <<http://www.infoq.com/br/articles/websocket-java-javaee>>. Acesso em: 20 nov. 2014.

CHAGAS, M.H; LIMA, F.V. **Variáveis estruturais: elementos primários para sistematização do treinamento em musculação**. Belo Horizonte: Editora GráficaSilveira, 2004.

CHAVES, Celia Regina Moutinho de Miranda et al. Exercício aeróbico, treinamento de força muscular e testes de aptidão física para adolescentes com fibrose cística: revisão da literatura. **Revista Brasileira de Saúde Materno Infantil**, Recife, v. 3, n. 7, p.245-250, jul. 2007. Disponível em: <<http://www.scielo.br/pdf/rbsmi/v7n3/03.pdf>>. Acesso em: 23 out. 2015.

CINESIOLOGIA. In: **Dicionário Priberam da Língua Portuguesa**. 2013. Disponível em: <<http://www.priberam.pt/DLPO/cinesiologia>>. Acesso em: 16 nov.2014.

CIVINSKI, Cristian; MONTIBELLER, André; BRAZ, André Luiz de Oliveira. A importância do exercício físico no envelhecimento. **Unifebe**, Brusque, v. 3, n. 9, p.163-175, jan. 2011. Disponível em: <<http://www.unifebe.edu.br/revistadaunifebe/2011/artigo028.pdf>>. Acesso em: 29 out. 2014.

CLARINDA, Deivid. **Avaliação do padrão de recrutamento de fibras do mesmo grupo muscular, através da eletromiografia em diferentes movimentos de musculação**. 2009. 36 f. Monografia (Especialista em Fisiologia do Exercício para o magistério superior) - Universidade do Extremo Sul Catarinense, Criciúma.

COMSCORE. **ComScore Reports September 2014 U.S. Smartphone Subscriber Market Share**. 2014. Disponível em: <<http://www.comscore.com/Insights/Market-Rankings/comScore-Reports-September-2014-US-Smartphone-Subscriber-Market-Share>>. Acesso em: 08 nov. 2015.

COSENZA, Carlos Eduardo. **Musculação: métodos e sistemas**. Rio de Janeiro: Sprint, 1995. 119 p.

DEITEL, Paul et al. **Android for programmers: An App-Driven Approach**. New Jersey: Prentice Hall, 2012. 481 p.

DEVELOPER, Android. **Como baixar o Android Studio e o SDK Tools**. 2015. Disponível em: <<https://developer.android.com/intl/pt-br/sdk/index.html>>. Acesso em: 26 out. 2015.

\_\_\_\_\_. **The Google Play Opportunity**. 2015. Disponível em: <<http://developer.android.com/intl/pt-br/distribute/googleplay/about.html>>. Acesso em: 23 out. 2015.

\_\_\_\_\_. **The world's most popular mobile platform**. 2014. Disponível em: <<http://developer.android.com/about/index.html>>. Acesso em: 17 maio 2014.

DRYSDALE, Brad. **WebSockets: France**: Kaazing, 2011. 68 slides, color. Disponível em: <<http://www.infoq.com/presentations/WebSockets-The-Web-Communication-Revolution#downloadPdf>>. Acesso em: 20 nov. 2014.

ECLIPSE. **IDE Eclipse**. Disponível em: <<https://eclipse.org/ide/>>. Acesso em: 20 nov. 2014.

ESPOSITO, Dino. **Compreendendo o poder dos WebSockets**. 2012. Disponível em: <<http://msdn.microsoft.com/pt-br/magazine/hh975342.aspx>>. Acesso em: 18 nov. 2014.

FARIA, Thiago. **Java EE 7: com JSF, PrimeFaces e CDI**. Minas Gerais: Algaworks, 2013. 197 p.

FERREIRA, Alan de Carvalho Dias et al. Musculação: Aspectos fisiológicos, neurais, metodológicos e nutricionais. In: ENCONTRO DE INICIAÇÃO À DOCÊNCIA, XI, 2008, Paraíba. **Anais eletrônicos...** Paraíba: UFPB, 2008. Disponível em: <[http://www.prac.ufpb.br/anais/xenex\\_xienid/xi\\_enid/prolicen/ANAIS/Area6/6CCSDE FPLIC04.pdf](http://www.prac.ufpb.br/anais/xenex_xienid/xi_enid/prolicen/ANAIS/Area6/6CCSDE FPLIC04.pdf)>. Acesso em: 15 set.2014.

FERREIRA, Marcos Santos. Aptidão física e saúde na educação física escolar: Ampliando o enfoque. **Revista brasileira de ciências do esporte**, Florianópolis, v.22, n.2, p.41-54, jul.2001. Disponível em: <<http://rbceonline.org.br/revista/index.php/RBCE/article/view/411/336>>. Acesso em: 08 set. 2014.

FIRMINO, Emiliano Carlos de Moraes. **WebSocket para aplicações web em Erlang**. 2011. 60 f. TCC (Graduação) - Curso de Engenharia de Computação, Universidade do Estado do Amazonas, Manaus, 2011. Disponível em: <<http://www.tcc-computacao.tiagodemelo.info/monografias/2011/tcc-emiliano-firmino.pdf>>. Acesso em: 20 nov. 2014.

FONSECA, Karoline Shienna. **Musculação e sua atuação na redução do percentual de gordura em alunos iniciantes do sexo feminino**. 2013. 41 f. Dissertação (Graduação em Educação Física) - Centro Universitário de Formiga, Formiga.

FRASSETTO, Susana. **Musculação para mulheres: Revisão Bibliográfica**. 2005. 42 f. Monografia (Especialista em Treinamento Esportivo para o magistério superior) - Universidade do Extremo Sul Catarinense, Criciúma.

FREITAS, Guilherme Waltricke. **A utilização da linguagem swift em dispositivos móveis baseados no sistema operacional iOS para avaliação da composição corporal**. 2014. 98 f. TCC (Graduação) - Curso de Ciência da Computação, Universidade do Extremo Sul Catarinense, Criciúma, 2014. Disponível em: <<http://tcc.kironunes.net.br/arquivos/trabalhos/406.pdf>>. Acesso em: 01 nov. 2015.

FURUKAWA, Y.. Web-based control application using WebSocket. In: ICALEPCS, 13., 2011, Grenobla. **Proceedings...** Grenobla: Esrf, 2011. p. 673 - 675. Disponível em: <<https://accelconf.web.cern.ch/accelconf/icalepcs2011/papers/wemau010.pdf>>. Acesso em: 17 nov. 2014.

GARGENTA, Marko; NAKAMURA, Masumi. **Learning Android: Develop mobile apps using Java and Eclipse**. 2. ed. Sebastopol: O'reilly Media, 2014. 288 p.

GOK, Nizamettin; KHANNA, Nitin. **Building Hybrid Android Apps with Java and JavaScript**. Sebastopol: O'reilly Media, 2013. 156 p.

GOOGLE. **Google Developers Console**. 2015. Disponível em: <<https://console.developers.google.com/>>. Acesso em: 29 out. 2015.

GUIMARÃES, Luigi Ferreira. **Análise de programas de treinamento na musculação prescritos na internet**. 2010. 76 f. Monografia (Especialização) - Curso de Educação Física, Universidade Federal de Minas Gerais, Belo Horizonte, 2010. Disponível em: <[https://www.ufmg.br/online/arquivos/anexos/monografia\\_guimaraes.pdf](https://www.ufmg.br/online/arquivos/anexos/monografia_guimaraes.pdf)>. Acesso em: 24 nov. 2014.

JACKSON, Wallace. **Android Apps for Absolute Beginners**. Nova York: Apress, 2011. 344 p.

LECHETA, Ricardo R. **Google Android: Aprenda a criar aplicações para dispositivos móveis com o Android SDK**. 3ª.ed., São Paulo: Novatec Editora, 2013.

LEE, Sunguk. Creating and Using Databases for Android Applications. **International Journal Of Database Theory And Application**. Pohang, p. 99-106. jun. 2012. Disponível em: <[http://www.sersc.org/journals/IJDTA/vol5\\_no2/8.pdf](http://www.sersc.org/journals/IJDTA/vol5_no2/8.pdf)>. Acesso em: 06 nov. 2015.

LEITÃO, Marcelo Bichels et al. Posicionamento oficial da sociedade brasileira de medicina do esporte: Atividade física e saúde na mulher. **Revista brasileira de medicina do esporte**, Niterói, v.6, n.6, p.215-220, nov./dec. 2000. Disponível em: <[http://www.scielo.br/scielo.php?pid=S1517-86922000000600001&script=sci\\_arttext](http://www.scielo.br/scielo.php?pid=S1517-86922000000600001&script=sci_arttext)>. Acesso em: 08 set. 2014.

LUCKOW, Décio Heinzemann; MELO, Alexandre Altair de. **Programação Java para a web**. São Paulo: Novatec, 2010. 638 p.

MATSUDO, Sandra Mahecha; MATSUDO, Victor Keihan Rodrigues; BARROS NETO, Turíbio Leite. Atividade física e envelhecimento: aspectos epidemiológicos. **Revista Brasileira de Medicina no Esporte**, São Paulo, v. 1, n. 7, p.2-13, jan./fev. 2001. Disponível em: <<http://www.scielo.br/pdf/rbme/v7n1/v7n1a02.pdf>>. Acesso em: 29 out. 2014.

MELLO, Marco Túlio de et al. O exercícios físico e os aspectos psicobiológicos. **Revista brasileira de medicina do esporte**, Niterói, v.11, n.3, p.203-207, jun. 2005. Disponível em: <[http://www.scielo.br/scielo.php?script=sci\\_arttext&pid=S1517-86922005000300010&lng=pt&nrm=iso](http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1517-86922005000300010&lng=pt&nrm=iso)>. Acesso em: 04 set. 2014.

MEIER, Reto. **Professional Android Application Development**. Indianapolis: Wiley Publishing, Inc, 2009. 409 p.

MENDES, Douglas Rocha. **Programação Java com Ênfase em Orientação a Objetos**. São Paulo: Novatec, 2009. 456 p.

MONTEIRO, Maria de Fátima; SOBRAL FILHO, Dário C. Exercício físico e o controle da pressão arterial. **Revista Brasileira de Medicina do Esporte**, Niterói, v. 10, n. 6, p.513-516, nov./dez. 2004. Disponível em: <<http://www.scielo.br/pdf/rbme/v10n6/a08v10n6.pdf>>. Acesso em: 8 set. 2014.

MUSSI, Ricardo Franklin de Freitas, DIAS, Marcos Aurélio Ferreira. **Repetição máxima e prescrição na musculação**. 2005. 11 f. Monografia (Especialização) - Curso de Treinamento Desportivo Avançado, Universidade Gama Filho, Salvador, 2005. Disponível em: <[http://www.educadores.diaadia.pr.gov.br/arquivos/File/2010/artigos\\_teses/EDUCACAO\\_FISICA/monografia/BoletimEF.org\\_Repeticao-maxima-e-prescricao-na-musculacao.pdf](http://www.educadores.diaadia.pr.gov.br/arquivos/File/2010/artigos_teses/EDUCACAO_FISICA/monografia/BoletimEF.org_Repeticao-maxima-e-prescricao-na-musculacao.pdf)>. Acesso em: 21 set. 2014.

ORACLE. **Java EE**. Disponível em: <<https://docs.oracle.com/javase/6/firstcup/doc/gkhoy.html>>. Acesso em: 20 nov. 2014.

\_\_\_\_\_. **Java EE Compatibility**. Disponível em: <<http://www.oracle.com/technetwork/java/javase/overview/compatibility-jsp-136984.html>>. Acesso em: 24 nov. 2014.

\_\_\_\_\_. **JDBC**. Disponível em: <<http://www.oracle.com/technetwork/java/javase/jdbc/index.html>>. Acesso em: 24 nov. 2014.

PEROZZO, ReinerFanthesco. **Framework para construção de sistemas supervisórios em dispositivos móveis**. 2007. 107 f. Dissertação (Mestrado em Engenharia Elétrica) - Universidade Federal do Rio Grande do Sul, Porto Alegre.

PONTES, Alexandro Lima Cordeiro; SOUSA, Iara Alves de; NAVARRO, Antonio Coppi. O tratamento da obesidade através da combinação dos exercícios físicos e terapia nutricional visando o emagrecimento. **Revista Brasileira de Obesidade, Nutrição e Emagrecimento**, São Paulo, v. 3, n. 14, p.124-135, mar. 2009. Disponível em: <<http://www.rbone.com.br/index.php/rbone/article/view/139/136>>. Acesso em: 19 out. 2015.

POTTS, Stephen; KOPACK, Mike. **Aprenda em 24 horas web services**. Rio de Janeiro: Elsevier, 2003. 367 p.

OLIVEIRA, Leandro Ramos de, MEDINA, Roseclea Duarte. **Desenvolvimento de objetos de aprendizagem para dispositivos móveis: uma nova abordagem que contribui para a educação**. Santa Maria: 2007. Disponível em: <<http://www.cinted.ufrgs.br/ciclo9/artigos/4aLeandro.pdf>>. Acesso em: 14 maio 2014.

RAMOS, Leonardo Batecini. **Um chat pictográfico para o SCALA: Sistema de Comunicação Alternativa para o Letramento de pessoas com Autismo**. 2013. 59 f.

TCC (Graduação) - Curso de Ciência da Computação, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2013. Disponível em: <<http://www.lume.ufrgs.br/bitstream/handle/10183/86431/000910038.pdf?sequence=1>>. Acesso em: 24 nov. 2014.

TOMCAT, Apache. **Which version**. Disponível em: <<http://tomcat.apache.org/whichversion.html>>. Acesso em: 25 nov. 2014.

SANTOS, Jefferson. **Interfaces para dispositivos móveis**. 2013. 99 f. Dissertação (Mestrado em Tecnologia da Inteligência e Design Digital) - Pontifícia Universidade Católica de São Paulo, São Paulo.

SCHLEUSS, Thyago. **Simulação de Futebol em Ambiente Web: Versão 3.0**. 2011. 102 f. TCC (Graduação) - Curso de Ciência da Computação, Fundação Universidade Regional de Blumenau, Blumenau, 2011. Disponível em: <<http://dsc.inf.furb.br/arquivos/tccs/monografias/TCC2011-2-30-VF-ThyagoSchleuss.pdf>>. Acesso em: 24 nov. 2014.

SIERRA, Kathy; BATES, Bert. **Use a cabeça! Java**. 2. ed. Rio de Janeiro: Alta Books, 2009. 484 p.

THEMUDO, Vasco Morais. **Implementação de um servidor de negociação em bolsa baseado em WebSocket**. 2014. 77 f. Tese (Doutorado) - Curso de Ciência de Computadores, Ciência de Computadores, Faculdade de Ciências da Universidade do Porto, Porto, 2014. Disponível em: <<http://repositorio-aberto.up.pt/bitstream/10216/75373/2/98622.pdf>>. Acesso em: 24 nov. 2014.

UBL, Malte; KITAMURA, Eiji. **Apresentando WebSockets: trazendo soquetes para a web**. 2010. Disponível em: <<http://www.html5rocks.com/pt/tutorials/websockets/basics/>>. Acesso em: 20 nov. 2014.

UNIDATA. **Tomcat Web Application Manager**. Disponível em: <<http://www.unidata.ucar.edu/software/thredds/v4.3/tds/images/manager.png>>. Acesso em: 17 nov. 2014.

VIEIRA, Thalles Jacobs. **Protótipo de aplicativo com a utilização da metodologia ágil scrum e do conceito kanban para acompanhamento do dispêndio energético em atividades físicas**. 2015. 100 f. TCC (Graduação) - Curso de Ciência da Computação, Universidade do Extremo Sul Catarinense, Criciúma, 2015. Disponível em: <<http://tcc.kironunesec.net.br/arquivos/trabalhos/441.pdf>>. Acesso em: 01 nov. 2015.

WANG, Vanessa; SALIM, Frank; MOSKOVITS, Peter. **The definitive guide to HTML5 WebSocket: Build real-time applications with HTML5**. New York: Apress, 2013. 208 p.

WEB SOCKET. **What is web socket?**. Disponível em: <<http://www.websocket.org/index.html>>. Acesso em: 17 nov. 2014.

\_\_\_\_\_. **HTML5 Web Sockets: A Quantum Leap in Scalability for the Web.**  
Disponível em: <<http://www.websocket.org/quantum.html>>. Acesso em: 17 nov. 2014.

WILLIAMS, Len; GROVES, Derek; THURGOOD, Glen. **Treinamento de força: Guia completo passo a passo para um corpo mais forte e definido.** São Paulo: Manole, 2010. 256 p.

WOLFSON, Mike; FELKER, Donn. **Android Developer Tools Essentials.** Sebastopol: O'reilly Media, 2013. 250 p.

**APÉNDICE(S)**

## APÊNDICE A – Termo de consentimento academia Enforce Fitness.



### TERMO DE CONSENTIMENTO

A academia **ENFORCE FITNESS** está sendo convidada para participar da aplicação dos testes do projeto de conclusão de curso intitulado:

Comunicação em tempo real com WebSocket aplicado ao desenvolvimento de um protótipo para prescrição de treino de musculação para dispositivos móveis Android, que tem como objetivo: Aplicar a tecnologia WebSocket para comunicação em tempo real entre uma aplicação web e um aplicativo Android (PersonalFit).

Os testes serão aplicados dentro da academia, onde o instrutor fica responsável por cadastrar o usuário (aluno) que vai usar o aplicativo em seu dispositivo móvel com o sistema *Android* e também, cadastrar os exercícios e suas especificações, bem como a sequência dos exercícios, suas repetições e séries (treino).

Os testes serão realizados a partir do dia 21 outubro até 6 de novembro de 2015.

Aceitando participar dos testes fica esclarecido que, por ser uma participação voluntária e sem interesse financeiro, a academia não terá direito a nenhuma remuneração. E também está permitindo a divulgação e publicação de seu nome no projeto, além dos resultados obtidos, que são os dados coletados através do uso das aplicações. Bem como a divulgação do nome da academia em redes sociais como de forma de incentivo os alunos para utilizarem o protótipo para a consulta dos treinos.

#### Procedimentos detalhados que serão utilizados na pesquisa

- Apresentação do funcionamento das aplicações para o responsável pela academia e também para o instrutor responsável pela prescrição dos treinos;
- Liberado o link da aplicação para o instrutor assim como um usuário e senha para acesso;
- Será explicado o uso das telas da aplicação web para o instrutor;
- Será explicado como efetuar o download do aplicativo do usuário/aluno nos dispositivos móveis Android, bem como o funcionamento das consultas.
- A coleta dos dados é feita via internet e gravada no servidor a cada acesso do usuário ao aplicativo.

Riscos/Alertas: Responsabilidade do instrutor de prescrever os exercícios corretos para cada aluno.

Benefícios: Prescrição do treino de maneira mais ágil, sem a utilização de papéis ou fichas de treino.

A coleta de dados será realizada pela acadêmica **Juliana Chaves Braz da Silva** (fone: 48 - 9921 2708) da 9ª fase da Graduação de Ciência da Computação da UNESC e orientado pelo professor responsável **William Bertan da Silva** (fone: 48 - 9943 6344).

Criciúma (SC) 21 de outubro de 2015.

「 22 501 859/0001-17 」

ENFORCE FITNESS LTDA. - ME

R. MARCELO LOBETTI, 33 - SALA: 05;  
Academia  
CENTRO - CEP 88.901-510  
CRICIÚMA - SC

*Pl. Danielle Michel*

Responsável:  
CPF: 044.273.509-09

*Juliana Chaves Braz da Silva*

Pesquisador Responsável:  
CPF: 075.125.789-38

## APÊNDICE B – Termo de consentimento academia Via Corpo II.



### TERMO DE CONSENTIMENTO

A academia **VIA CORPO II** está sendo convidada para participar da aplicação dos testes do projeto de conclusão de curso intitulado:

Comunicação em tempo real com WebSocket aplicado ao desenvolvimento de um protótipo para prescrição de treino de musculação para dispositivos móveis Android, que tem como objetivo: Aplicar a tecnologia WebSocket para comunicação em tempo real entre uma aplicação web e um aplicativo Android (PersonalFit).

Os testes serão aplicados dentro da academia, onde o instrutor fica responsável por cadastrar o usuário (aluno) que vai usar o aplicativo em seu dispositivo móvel com o sistema *Android* e também, cadastrar os exercícios e suas especificações, bem como a sequência dos exercícios, suas repetições e séries (treino).

Os testes serão realizados a partir do dia 22 outubro até 6 de novembro de 2015.

Aceitando participar dos testes fica esclarecido que, por ser uma participação voluntária e sem interesse financeiro, a academia não terá direito a nenhuma remuneração. E também está permitindo a divulgação e publicação de seu nome no projeto, além dos resultados obtidos, que são os dados coletados através do uso das aplicações. Bem como a divulgação do nome da academia em redes sociais como de forma de incentivo os alunos para utilizarem o protótipo para a consulta dos treinos.

#### Procedimentos detalhados que serão utilizados na pesquisa

- Apresentação do funcionamento das aplicações para o responsável pela academia e também para o instrutor responsável pela prescrição dos treinos;
- Liberado o link da aplicação para o instrutor assim como um usuário e senha para acesso;
- Será explicado o uso das telas da aplicação web para o instrutor;
- Será explicado como efetuar o download do aplicativo do usuário/aluno nos dispositivos móveis Android, bem como o funcionamento das consultas.
- A coleta dos dados é feita via internet e gravada no servidor a cada acesso do usuário ao aplicativo.

Riscos/Alertas: Responsabilidade do instrutor de prescrever os exercícios corretos para cada aluno.

Benefícios: Prescrição do treino de maneira mais ágil, sem a utilização de papéis ou fichas de treino.

A coleta de dados será realizada pela acadêmica **Juliana Chaves Braz da Silva** (fone: 48 - 9921 2708) da 9ª fase da Graduação de Ciência da Computação da UNESC e orientado pelo professor responsável **William Bertan da Silva** (fone: 48 - 9943 6344).

Criciúma (SC) 22 de outubro de 2015.



MARINA TISSI

Responsável:  
CPF: 39 4023 800 78

Juliana Chaves Braz da Silva

Pesquisador Responsável:  
CPF: 075.125.789-38

## APÊNDICE C – Termo de consentimento academia Super Action.



### TERMO DE CONSENTIMENTO

A academia **SUPER ACTION** está sendo convidada para participar da aplicação dos testes do projeto de conclusão de curso intitulado:

Comunicação em tempo real com WebSocket aplicado ao desenvolvimento de um protótipo para prescrição de treino de musculação para dispositivos móveis Android, que tem como objetivo: Aplicar a tecnologia WebSocket para comunicação em tempo real entre uma aplicação web e um aplicativo Android (PersonalFit).

Os testes serão aplicados dentro da academia, onde o instrutor fica responsável por cadastrar o usuário (aluno) que vai usar o aplicativo em seu dispositivo móvel com o sistema *Android* e também, cadastrar os exercícios e suas especificações, bem como a sequência dos exercícios, suas repetições e séries (treino).

Os testes serão realizados a partir do dia 23 outubro até 6 de novembro de 2015.

Aceitando participar dos testes fica esclarecido que, por ser uma participação voluntária e sem interesse financeiro, a academia não terá direito a nenhuma remuneração. E também está permitindo a divulgação e publicação de seu nome no projeto, além dos resultados obtidos, que são os dados coletados através do uso das aplicações. Bem como a divulgação do nome da academia em redes sociais como de forma de incentivo os alunos para utilizarem o protótipo para a consulta dos treinos.

#### Procedimentos detalhados que serão utilizados na pesquisa

- Apresentação do funcionamento das aplicações para o responsável pela academia e também para o instrutor responsável pela prescrição dos treinos;
- Liberado o link da aplicação para o instrutor assim como um usuário e senha para acesso;
- Será explicado o uso das telas da aplicação web para o instrutor;
- Será explicado como efetuar o download do aplicativo do usuário/aluno nos dispositivos móveis Android, bem como o funcionamento das consultas.
- A coleta dos dados é feita via internet e gravada no servidor a cada acesso do usuário ao aplicativo.

Riscos/Alertas: Responsabilidade do instrutor de prescrever os exercícios corretos para cada aluno.

Benefícios: Prescrição do treino de maneira mais ágil, sem a utilização de papéis ou fichas de treino.

A coleta de dados será realizada pela acadêmica **Juliana Chaves Braz da Silva** (fone: 48 - 9921 2708) da 9ª fase da Graduação de Ciência da Computação da UNESC e orientado pelo professor responsável **William Bertan da Silva** (fone: 48 - 9943 6344).

Criciúma (SC) 23 de outubro de 2015.

CNPJ: 21.358.745/0001-05

Super Action Academia

Academia

Antonio Luiz da Matta Junior

Responsável:

CPF: 074.710.669-03

Juliana Chaves Braz da Silva

Pesquisador Responsável:

CPF: 075.125.789-38

## APÊNDICE D – Termo de consentimento academia Via Corpo I.



### TERMO DE CONSENTIMENTO

A academia **VIA CORPO I** está sendo convidada para participar da aplicação dos testes do projeto de conclusão de curso intitulado:

Comunicação em tempo real com WebSocket aplicado ao desenvolvimento de um protótipo para prescrição de treino de musculação para dispositivos móveis Android, que tem como objetivo: Aplicar a tecnologia WebSocket para comunicação em tempo real entre uma aplicação web e um aplicativo Android (PersonalFit).

Os testes serão aplicados dentro da academia, onde o instrutor fica responsável por cadastrar o usuário (aluno) que vai usar o aplicativo em seu dispositivo móvel com o sistema *Android* e também, cadastrar os exercícios e suas especificações, bem como a sequência dos exercícios, suas repetições e séries (treino).

Os testes serão realizados a partir do dia 26 outubro até 6 de novembro de 2015.

Aceitando participar dos testes fica esclarecido que, por ser uma participação voluntária e sem interesse financeiro, a academia não terá direito a nenhuma remuneração. E também está permitindo a divulgação e publicação de seu nome no projeto, além dos resultados obtidos, que são os dados coletados através do uso das aplicações. Bem como a divulgação do nome da academia em redes sociais como de forma de incentivo os alunos para utilizarem o protótipo para a consulta dos treinos.

#### Procedimentos detalhados que serão utilizados na pesquisa

- Apresentação do funcionamento das aplicações para o responsável pela academia e também para o instrutor responsável pela prescrição dos treinos;
- Liberado o link da aplicação para o instrutor assim como um usuário e senha para acesso;
- Será explicado o uso das telas da aplicação web para o instrutor;
- Será explicado como efetuar o download do aplicativo do usuário/aluno nos dispositivos móveis Android, bem como o funcionamento das consultas.
- A coleta dos dados é feita via internet e gravada no servidor a cada acesso do usuário ao aplicativo.

Riscos/Alertas: Responsabilidade do instrutor de prescrever os exercícios corretos para cada aluno.

Benefícios: Prescrição do treino de maneira mais ágil, sem a utilização de papéis ou fichas de treino.

A coleta de dados será realizada pela acadêmica **Juliana Chaves Braz da Silva** (fone: 48 - 9921 2708) da 9ª fase da Graduação de Ciência da Computação da UNESC e orientado pelo professor responsável **William Bertan da Silva** (fone: 48 - 9943 6344).

Criciúma (SC) 26 de outubro de 2015.

Zaniboni & Zaniboni Ativ. Físicas  
CNPJ: 10.306.246/0001-40  
Fone: (48) 3045-5312

Academia

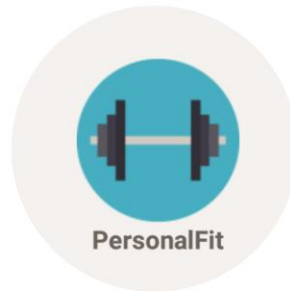
MARIA ZANIBONI

Responsável:  
CPF: 551.288.159.00

Juliana Chaves Braz da Silva

Pesquisador Responsável:  
CPF: 075.125.789-38

APÊNDICE E – Banner com dicas para o instrutor sobre a aplicação web.



## **Dicas para o instrutor:**

- \* Cadastre seu aluno;**
- \* Cadastre os exercícios;**
- \* Cadastre o treino para o aluno;**
  
- \* Informe o login e senha para que aluno entre no aplicativo;**

**Pronto! O treino de seus alunos  
está disponível no aplicativo  
PersonalFit!**

**Para o instrutor:**

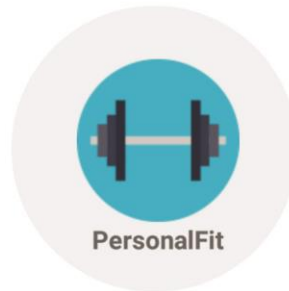
**Acesse o link <http://191.101.224.207/ProjetoTCC/login.jsp>**

**Para o aluno:**

**Faça o download do app PersonalFit**



APÊNDICE F – Banner sobre o aplicativo PersonalFit.



# Seu treino agora está disponível\* no seu celular!

**Fale com seu instrutor**



**Faça o download do app PersonalFit**

**\*Disponível somente para dispositivos  
Android. Requer versão 4.3 ou superior**



## APÊNDICE G – Artigo

# Comunicação em tempo real com WebSocket aplicado ao desenvolvimento de um protótipo para prescrição de treino de musculação para dispositivos móveis Android

Juliana Chaves Braz da Silva<sup>1</sup>, William Bertan da Silva<sup>1</sup>

<sup>1</sup>Ciência da Computação – Universidade do Extremo Sul Catarinense (UNESC)  
Criciúma – SC – Brasil

jucbraz@gmail.com, w\_toco@hotmail.com

**Abstract.** *The technology and use of mobile devices are increasingly present in popular culture, spread throughout the world. This fact makes developers create applications using the technology to supply automated routines, on this and the current way of prescribing workouts in gyms paper sheets with the necessary information about the workout, was created this project in order to present a solution automated for prescription workouts with real-time communication through the use of WebSocket between a web application and an Android application.*

**Resumo.** *A tecnologia e uso dos dispositivos móveis estão cada vez mais presentes na cultura popular, espalhados por todo o mundo. Com isto desenvolvedores buscam criar aplicativos para suprir uma necessidade de forma mais automatizada, diante disto e da atual maneira de prescrever treinos de musculação nas academias em fichas de papel com as informações necessárias sobre o treino, foi criado este projeto a fim de apresentar uma solução automatizada para prescrição dos treinos de musculação com a comunicação em tempo real através do uso de WebSocket entre uma aplicação web e um aplicativo Android.*

## 1. Introdução

Hoje em dia com todos os afazeres é difícil manter uma rotina saudável, com exercícios físicos e uma boa alimentação. Com os obstáculos para uma vida saudável, hábitos como a má alimentação e estresse exagerado tornam-se prejudiciais à saúde e trazem doenças como o sedentarismo, obesidade, hipertensão arterial sistêmica e outros distúrbios [Bündchen et al, 2013].

O mercado de dispositivos móveis, em especial celulares, está expandindo cada vez mais e com ele a mobilidade, que tem como objetivo permitir o acesso a dados, Internet, e-mail, agenda, mensagens e outras aplicações de qualquer lugar, que por sua vez facilita a troca de informações sem precisar estar em uma estação fixa, como um computador [Lecheta, 2013].

Com o uso crescente dos dispositivos móveis aumenta também o interesse pelo desenvolvimento *mobile*, em especial o Android, baseado no grande número de aplicativos disponíveis para *download* na Google Play Store, loja de aplicativos disponíveis para dispositivos móveis [Comscore, 2014].

Entretanto para uma boa interação entre usuários e o aplicativo, este deve ter uma resposta rápida quando conectada a um servidor *web*. Para o desenvolvimento de aplicações que se comuniquem em tempo real é utilizada uma tecnologia chamada WebSocket [Wang; Salim; Moskovits, 2013, tradução nossa].

Desta forma o artigo consiste em mostrar a aplicação da tecnologia WebSocket em um ambiente real, através de uma aplicação para atender a necessidade da prescrição de treino de musculação de uma forma mais ágil, contando com uma aplicação *web* para que o instrutor insira os treinos de seus alunos, que por sua vez o recebem em seus dispositivos móveis com o sistema Android através de um aplicativo, podendo assim listar seu treino, marcar quais exercícios já foram feitos e visualizar quais ainda faltam para concluir o treino, através de uma conexão em tempo real feita pelo WebSocket. E, por fim validar o uso das aplicações disponibilizando-as para testes em um ambiente real de academia de musculação na cidade de Criciúma, Santa Catarina.

## 2. Exercício físico

O exercício físico é caracterizado por uma atividade com repetições sistemáticas de movimentos orientados, que por sua vez faz aumentar o consumo de oxigênio devido ao esforço exigido do músculo. Um dos objetivos da prática deste tipo de exercício é a melhora do condicionamento físico [Monteiro; Sobral Filho, 2004].

Em geral as pessoas que procuram uma atividade para a prática regular de exercícios, estão em busca de um objetivo específico que pode variar entre a estética, condicionamento físico ou até mesmo melhorias na saúde. Os exercícios prescritos no treino mudam de acordo com o objetivo do indivíduo [Abad et al, 2010]. Para que seja possível alcançar os objetivos do exercício proposto, deve-se seguir uma estrutura programada, repetitiva e regular. Os seguintes pontos também devem ser observados para a prescrição de um treino: frequência dos exercícios (quantidade de vezes que são executados por semana), intensidade (ritmo de execução e/ou carga) e duração (tempo de execução) [Barbanti, 2012].

### 2.1 Musculação

A musculação pode ser definida como um tipo de treinamento físico, que se resume em um conjunto de exercícios com carga, onde o aumento dos pesos são realizados de forma progressiva [Barbanti, 1997].

É importante trabalhar com um profissional formado em alguma área da ciência da saúde, como o *personal trainer* ou instrutor de musculação com conhecimento do corpo humano, fisiologia e compreensão dos métodos de treinamento, o que pode trazer um resultado melhor e mais rápido [Aaberg, 2001].

### 2.2 Metodologias de prescrição de treino para musculação

A prescrição é uma instrução escrita sobre os exercícios físicos que devem ser realizados dentro de um período determinado [Barbanti, 1997]. Os principais pontos a serem observados ao montar um plano de treinamento são descritos por: volume de treinamento, intensidade, número de exercícios, repetições e séries [Bompa, 2001].

## 3. Android

O Android é um Sistema operacional desenvolvido para a utilização em sistemas móveis. A base de sua estrutura é feita por um *kernel* do Linux, que por sua vez é *Open Source*, ou seja, possui seu código aberto, tornando possível o acesso do código aos desenvolvedores para que

estes possam realizar colaborações, correções e desenvolvimento de novos aplicativos [JACKSON, 2011, tradução nossa].

Desenvolvedores Android podem desenvolver e distribuir seus aplicativos na *Google Play*, também conhecida como *Play Store*, loja de aplicativos da plataforma Android, nela é possível disponibilizar aplicações gratuitas ou pagas para todo o mundo, visto que a loja atende todos os usuários de dispositivos com sistema Android [DEVELOPER, 2015, tradução nossa].

### 3.1 Desenvolvimento Android

Ao desenvolver um aplicativo é importante levar em consideração alguns pontos, que mudam conforme a empresa que disponibiliza o dispositivo móvel, sendo eles: tamanho da tela do dispositivo, fragmentação e capacidade do *hardware* [WOLFSON; FELKER, 2013, tradução nossa].

É possível visualizar a tela do celular no computador através do emulador, para que isto seja possível é utilizado o *Android Virtual Device* (AVD), que irá ter o comportamento de um celular real com as suas configurações. É importante verificar as APIs usadas e se o celular comporta a versão em que foi desenvolvida a aplicação. Em casos de APIs recentes, estas não serão compatíveis com versões anteriores do Android, o que pode não garantir o funcionamento da aplicação [LECHETA, 2013].

### 3.2 Tecnologias envolvidas no desenvolvimento Android

As aplicações Android são desenvolvidas para dispositivos da área móvel utilizando o *Android SDK*, ferramenta que está disponível sem custos e taxas para os desenvolvedores. A linguagem pode ser escolhida conforme a experiência de cada um, partindo da ideia que existe mais de uma linguagem compatível com o Android, sendo elas: Java, C/C++, Python, Ruby, *Hypertext Markup Language* (HTML) e Javascript [GOK; KHANNA, 2013, tradução nossa].

Um recurso disponível para o desenvolvimento Android é o *Android Studio*, esta ferramenta foi adotada como o ambiente de desenvolvimento oficial do Android. Que dispõe de todos os recursos necessários, como o editor de código inteligente, capaz de refatorar e analisar os códigos apontando os possíveis erros e sugerindo o que deve ser alterado, permite as simulações do aplicativo em diversos tamanhos de telas e tipos de dispositivos, como *Tablets*, *Smartphones*, *Android Wear*, *Android TV* e *Google Glass*. Pode ser feito o download da ferramenta de forma gratuita [DEVELOPER, 2015].

Lecheta (2013) também diz que uma integração de dados entre o Android e um banco de dados é possível através do *SQLite* que é um banco de dados gratuito, poderoso e leve, característica importante ao pensar na capacidade do dispositivo. O gerenciamento pode ser feito através de ferramentas disponíveis gratuitamente como *SQLite Expert Personal* e *SQLite Plus*.

## 4. Aplicação web

Aplicações *web* ou sites são projetados para fornecer o retorno de uma requisição feita pelo usuário, a pessoa informa um endereço *URL* para acessar uma página *web*, a requisição é enviada para o servidor que processa a solicitação e retorna uma resposta para a requisição do usuário, como ilustra a figura 2 [POTTS; KOPACK, 2003].

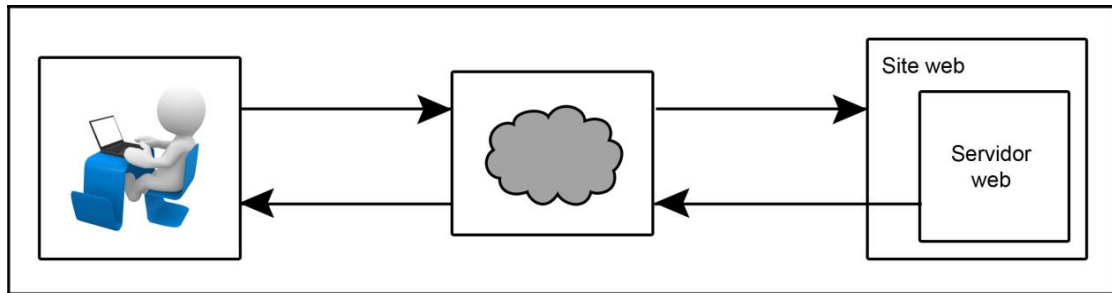


Figure 1. Como um navegador interage com um servidor *web* para fazer requisições.

#### 4.1. Tecnologias aplicadas em desenvolvimento *web*

Para o desenvolvimento de uma aplicação *web* é necessário escolher as ferramentas e tecnologias que permitam a criação, armazenamento, manipulação e a comunicação de dados. As seleções destas ferramentas podem envolver vários aspectos, como a popularidade e a experiência [LUCKOW; MELO, 2010].

#### 4.2. Java Enterprise Edition

*Java Enterprise Edition (JEE)* pode ser definida como um conjunto de especificações detalhadas de como deve ser implementado um software. Possui API para desenvolvimento de aplicações, incluindo serviços de rede e *web*. As principais APIs disponibilizadas pelo JEE voltadas para o desenvolvimento *web* são: *JavaServer Pages (JSP)*, *Java Servlets* e *Java Server Faces (JSF)* [CAELUM, 2012].

O *JSF* é um *framework web* padrão da plataforma *Java Enterprise Edition*, que permite a utilização da linguagem de programação Java para desenvolvimentos de páginas *web* [LUCKOW; MELO, 2010]. Seu principal foco é simplificar e facilitar o desenvolvimento das telas de uma aplicação *web*, também conhecida como interfaces, por meio de componentes reutilizáveis [FARIA, 2013].

O *Apache Tomcat* pode ser definido com duas funções: contêiner Java e servidor *web* simultaneamente. É compatível com as tecnologias *Java Servlet* e *JSP*, que possibilitam o funcionamento do Java em um ambiente *web*. Além da sua capacidade de execução do *JSP* e *Java Servlet*, o *Apache Tomcat* possui um recurso para o gerenciamento das sessões *web* [LUCKOW; MELO, 2010].

#### 4.3. MySQL

O *MySQL* é um banco de dados, utilizado para o armazenamento dos dados de um software. É o banco de dados mais conhecido que possui código-aberto, grandes empresas utilizam este banco, como Amazon.com, Google, Motorola e outros. Além de possuir recursos como o *MySQL Workbench*, uma ferramenta visual para o desenvolvimento da estrutura do banco de dados [LUCKOW; MELO, 2010].

A ferramenta utilizada para conexão entre um banco de dados e a linguagem Java é conhecida como *Java Database Connectivity (JDBC)*, que consiste em um conjunto de classes e interfaces *API* escritas em Java que possibilitam a conexão com um banco de dados ou o acesso a qualquer fonte de dados tabular, além do envio de instruções *SQL* e a possibilidade de processar os resultados [ORACLE, 2014, tradução nossa].

#### 4.4. Sincronização em tempo real

Existe mais de uma forma de comunicação para que seja possível transmitir os dados de um dispositivo para a *web* e vice-e-versa, sendo eles: utilização do protocolo *Hypertext Transfer*

*Protocol* (HTTP), *sockets* ou *WebSockets*. Cada uma possui particularidades, porém todas funcionam com a finalidade da comunicação dos dados. O acesso através de um *WebService* é uma das tecnologias disponíveis para a comunicação, porém não existe uma API nativa para acessar um *WebService* de uma aplicação Android [LECHETA, 2013].

Um aplicativo *web* baseado em *HTML* não pode fazer uma comunicação *full-duplex*, nos dois sentidos, com o servidor, pois a comunicação *HTTP* só começa a partir de um cliente, isto significa que um servidor não pode enviar qualquer informação a um cliente sem acesso a partir do cliente. Diferente do protocolo *WebSocket* que possui a comunicação *full-duplex* e é de fácil desenvolvimento através do *Javascript* [FURUKAWA, 2011, tradução nossa].

#### 4.4.1 WebSocket

A comunicação bidirecional do *WebSocket* entre o aplicativo e o servidor *web* é feita por um único soquete *Transmission Control Protocol (TCP)*, que permite a comunicação em tempo real [ESPOSITO, 2012].

Além de ser um protocolo para a sincronização em tempo real, o *WebSocket* também é uma API padronizada pela *World Wide Web Consortium (W3C)* enquanto a padronização do protocolo é realizada pela *Internet Engineering Task Force (IETF)*, *WebSocket* também pode ser definida como uma conexão *full-duplex*, ou seja, pode se comunicar em ambas direções, sobre os quais mensagens podem ser enviadas entre cliente e o servidor. O padrão *WebSocket* simplifica muito a complexidade em torno da comunicação *web* bidirecional e de gerenciamento de conexão [WEB SOCKET, 2013, tradução nossa].

#### 4.4.2 Funcionamento do WebSocket

O protocolo *WebSocket* é baseado em uma conexão persistente e possui um comportamento de envio de mensagens, com o conteúdo de requisições e respostas a estas requisições. O cliente inicia enviando uma requisição ao servidor com o cabeçalho “*upgrade*” para que seja possível alterar do modo *HTTP* para *WebSocket* e o servidor deve retornar a resposta da solicitação do cliente, este procedimento é nomeado de *Open Handshake*, como ilustra a figura 2 [CAMPANINI, 2012].



Figure 2. Esquema do funcionamento WebSocket

A maior vantagem para a utilização do *WebSocket* é ter uma conexão entre servidor e cliente com o tempo real de baixa latência. Além de gerar pouca carga no servidor. Com isso empresas grandes utilizam a tecnologia *WebSocket*, como o *Twitter* e *Facebook*, já que é ideal

para aplicar em serviços de bate-papo, jogos online com vários jogadores, transmissões ao vivo via Internet e atualizações em tempo real nas redes sociais [UBL; KITAMURA, 2010].

## **5. Comunicação em tempo real com WebSocket aplicado ao desenvolvimento do protótipo PersonalFit para prescrição de treino de musculação para dispositivos móveis Android**

O projeto consiste no desenvolvimento de duas aplicações a fim de possibilitar a aplicação do protocolo WebSocket para uma comunicação em tempo real, visando a usabilidade em um ambiente real através de testes aplicados em academias de musculação na cidade de Criciúma, Santa Catarina.

As aplicações serão utilizadas para auxiliar a prescrição de treinos de musculação, suprimindo a necessidade do uso das fichas ou folhas de papel, atual forma de consulta aos treinos dos praticantes da atividade de musculação. A ideia consiste em disponibilizar uma aplicação *web* para que o instrutor de musculação prescreva os treinos de musculação para o seu aluno, que por sua vez tenha acesso ao aplicativo desenvolvido para a plataforma Android nomeado PersonalFit, cuja função é a consulta dos exercícios que compõe o treino de forma prática e clara, o que facilita a rotina dentro da academia em horários de maior movimento, visto que a consulta dos treinos no aplicativo *mobile* conta com o nome do exercício, uma breve descrição em que o instrutor pode informar como executar o movimento do exercício bem como o número da máquina em que o exercício deve ser feito, além da série e repetição de cada exercício. De forma que facilite o entendimento do que deve ser feito e como deve ser executado o exercício, diminuindo assim o fluxo de pessoas com dúvidas relacionadas ao treino durante o horário de maior movimento na academia.

Entretanto, este projeto faz a integração de duas áreas distintas sendo elas a Ciência da Computação e a Educação Física, com o objetivo de atender uma necessidade da área da Educação Física através do desenvolvimento das aplicações especificadas citadas anteriormente bem como a aplicação da tecnologia WebSocket, o projeto conta também com testes aplicados em um ambiente real. Estes são de suma importância para que se obtenha noção da usabilidade dentro de um cenário real de academia de musculação, bem como o comportamento das aplicações com as tecnologias envolvidas.

### **5.1 Metodologia**

Para a realização do projeto foi realizado o levantamento bibliográfico a partir de livros, artigos, pesquisas na Internet, *e-books* e publicações, a fim de coletar dados para iniciar o desenvolvimento das aplicações. A partir disto, foram estabelecidos os requisitos e funcionamento das aplicações, uma aplicação *web* para o instrutor prescrever os exercícios do treino para o seu aluno, este com acesso ao aplicativo Android para a consulta do treino através do uso da Internet e da conexão WebSocket para a troca de dados. Somente alunos cadastrados pelo instrutor têm acesso ao aplicativo Android nomeado PersonalFit.

Com os requisitos e funcionamento estabelecidos foi realizada a modelagem do banco de dados a fim de iniciar o desenvolvimento das aplicações, o modelo lógico do banco permite definir quais as tabelas e atributos devem ser criados e de que forma são os relacionamentos entre elas. Foram utilizados os ambientes de desenvolvimento IDE NetBeans 8.0.2 e Android Studio, bem como a linguagem de programação Java, servidor Apache Tomcat 8.0.18, *framework* Foundation, banco de dados MySQL e SQLite, JSON para a troca de dados entre as plataformas diferentes e o *framework* Autobahn que permite a utilização do WebSocket e sua estrutura como demonstra a figura 3.

A estrutura do WebSocle é dada por quatro principais funções onOpen para iniciar a conexão, onMessage para enviar as mensagens, onTextMessage para receber a mensagem e por fim onClose para fechar a conexão WebSocket.

```

@ServerEndpoint("/treinoendpoint")
public class WSEndpointTreino {

    private static Set<Session> sessions = Collections.synchronizedSet(new HashSet<Session>());

    @OnOpen
    public void onOpen(Session session) {
        sessions.add(session);
        System.out.println(session.getId() + " has opened a connection");
    }

    @OnMessage
    public void onMessage(String message, Session session) {
        WebSocketMessage webSocketMessage = JsonManager.fromJson(message, new TypeToken<WebSocketMessage>() {
        }.getType());
        String tag = webSocketMessage.getTag();
        message = webSocketMessage.getMessage();
        System.out.println("Message from: " + session.getId() + ": " + tag + " - " + message);
        //Verifica conforme TAG para saber qual ação executar.
        try {
            WebSocketMessage webSocketMessageRetorno = new WebSocketMessage(tag);
            switch (tag) {
                case "LoginActivity": {
                    int indexPipe = message.indexOf("|");
                    String txtLogin = message.substring(0, indexPipe);
                    String txtPassword = message.substring(indexPipe + 1);
                    System.out.println("txtLogin: " + txtLogin);
                    System.out.println("txtPassword: " + txtPassword);
                    //TODO Validar Login
                    Usuario usuario = UsuarioDAO.verificaUsuario(txtLogin, txtPassword);
                    if (usuario == null) {
                        webSocketMessageRetorno.setMessage("-1");
                    } else {
                        webSocketMessageRetorno.setMessage(JsonManager.toJson(usuario));
                    }
                }
                break;
                case "TreinoActivity": {
                    int indexPipe = message.indexOf("|");
                }
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    @OnClose
    public void onClose(Session session) {
        sessions.remove(session);
        System.out.println(session.getId() + " fechou a conexao");
    }
}

```

Figure 3. Classe WebSocket do projeto *web*.

## 5.2 Testes em um cenário real de academia de musculação

A realização dos testes teve início no dia vinte (20) de outubro e fim no dia seis (6) de novembro, aplicado em quatro (4) academias na cidade de Criciúma. Foram apresentadas as aplicações aos responsáveis pelas academias juntamente com os instrutores de musculação. Todas as academias disponibilizaram o uso do *WIFI* para a conexão das aplicações e também incentivaram os alunos ao uso do aplicativo PersonalFit.

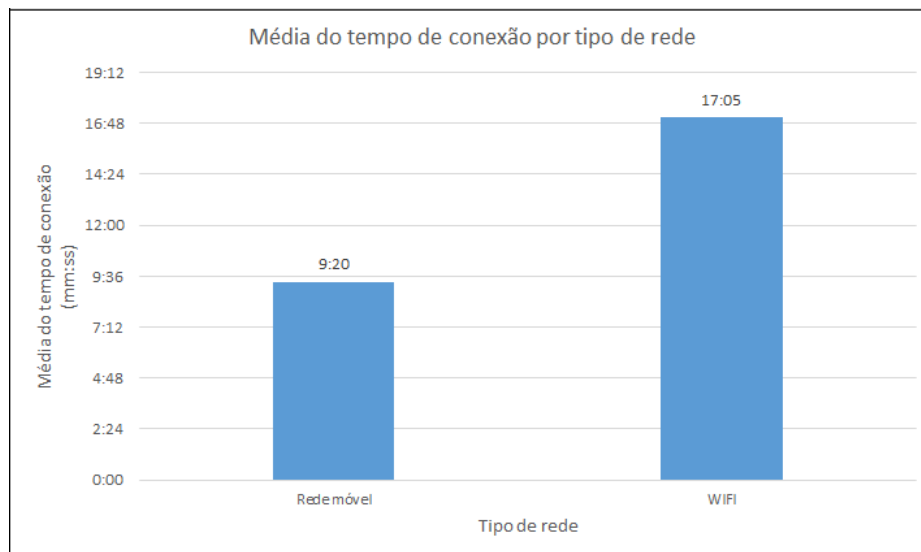
## 5.3 Resultados obtidos

Entre as quatro academias que se disponibilizaram para a realização dos testes foram apresentadas as aplicações para oito (8) instrutores e sete (7) responsáveis e proprietários dos locais. O aplicativo *mobile* PersonalFit teve trinta e dois (32) *downloads* dentro do período de testes, porém nem todos que fizeram o *download* conseguiram usar, visto que só tem acesso o aluno que o instrutor cadastra a fim de restringir a prescrição dos treinos somente ao instrutor

de musculação ou *personal*, com isto o total de usuários ativos dentro do período foi de onze (11) alunos.

Dos onze (11) alunos que utilizaram a aplicação sete (7) conectaram-se através da rede *WIFI*, três (3) através da rede móvel e um (1) utilizou as duas redes, uma em cada dia de uso. Além das conexões por redes diferentes também foram feitas utilizações em aparelhos de marcas e versões do sistema Android distintas. Os principais fabricantes foram Samsung, Sony, LG e Motorola.

O tempo médio de conexão por tipo de rede *WIFI* e rede móvel baseado no número de onze (11) alunos foi de 17min de conexão ativa por meio do *WIFI* e 9min e 20s por rede móvel como ilustra a figura.



**Figure 4. Gráfico da média do tempo de conexão por tipo de rede.**

A média do aluno mais ativo do aplicativo PersonalFit foi de 20min e 26s de conexão, utilizando a rede *WIFI* e um dispositivo da fabricante Sony, modelo C6503.

## 6. Conclusão

Entretanto o projeto apresentou duas aplicações para o auxílio na prescrição de treino de musculação, sendo uma *web*, utilizada pelo instrutor de musculação e outra para a consulta dos treinos prescritos através de um aplicativo *mobile* para dispositivos com o sistema operacional Android utilizada pelos alunos praticantes das atividades físicas, a fim de aplicar a tecnologia WebSocket em um ambiente real com o objetivo de testar a solução apresentada e obter dados do comportamento do WebSocket.

A pesquisa realizada para compreender a metodologia de uso na prescrição dos treinos de musculação foi importante para estabelecer o funcionamento das aplicações, bem como suas restrições, e também adquirir conhecimento para o desenvolvimento das aplicações e o uso das tecnologias envolvidas nelas.

Após o desenvolvimento das aplicações foi possível aplicar os testes em um cenário real a fim de validar o uso da conexão WebSocket e das aplicações em seu ambiente real de musculação. Com os testes foi possível obter o comportamento esperado mantendo a conexão ativa através das redes móveis e *WIFI* em modelos de dispositivos diferentes bem como as versões do sistema operacional Android.

A conexão utilizando o protocolo WebSocket foi estabelecida com sucesso ao executarem as aplicações quando a conexão com a Internet estava ativa, por meio de redes

móveis ou *WIFI*. A comunicação feita pelo WebSocket foi perdida somente quando os usuários se desconectavam das redes móveis ou *WIFI*, visto que em alguns algumas academias o sinal da *WIFI* não funciona em todo o ambiente.

Contudo para o melhor funcionamento dentro do ambiente real de musculação foram sugeridas melhorias em relação ao aplicativo *mobile* PersonalFit, principalmente como disponibilizar a aplicação para outros sistemas operacionais, principalmente para o sistema iOS.

## Referências

- AABERG, Everett. **Musculação, biomecânica e treinamento**. São Paulo: Manole Ltda, 2001. 217 p.
- ABAD, César Cavinato Cal et al. Efeito do exercício aeróbico e resistido no controle autonômico e nas variáveis hemodinâmicas de jovens saudáveis. **Revista brasileira de educação física do esporte**, São Paulo, v.24, n.4, p.535-544, out./dez.2010. Disponível em: <[http://www.scielo.br/scielo.php?pid=S1807-55092010000400010&script=sci\\_arttext](http://www.scielo.br/scielo.php?pid=S1807-55092010000400010&script=sci_arttext)>. Acesso em: 14 set. 2014.
- BARBANTI, Valdir J. **Treinamento físico: bases científicas**. 3.ed. São Paulo: CLR Balieiro, 1996.
- \_\_\_\_\_. **Teoria e prática do treinamento esportivo**. 2.ed. São Paulo: Editora Edgard Blücher Ltda, 1997.
- BOMPA, Tudor O. **A periodização no treinamento esportivo**. São Paulo: Manole Ltda, 2001. 259 p.
- BÜNDCHEN, Daiana Cristine et al. Exercício físico controla pressão arterial e melhora qualidade de vida. **Revista brasileira de medicina do esporte**, São Paulo, v. 2, n. 19, p.91-95, mar/abr. 2013. Disponível em: <[http://www.scielo.br/scielo.php?pid=S1517-86922013000200003&script=sci\\_arttext](http://www.scielo.br/scielo.php?pid=S1517-86922013000200003&script=sci_arttext)>. Acesso em: 03 nov. 2014.
- CAELUM. **O que é Java EE?** 2012. Disponível em: <<http://www.caelum.com.br/apostila-java-web/o-que-e-java-ee/>>. Acesso em: 20 nov. 2014.
- CAMPANINI, André. **WebSocket no Java e Java EE: da especificação a exemplos**. 2012. Disponível em: <<http://www.infoq.com/br/articles/websocket-java-javaee>>. Acesso em: 20 nov. 2014.
- COMSCORE. **ComScore Reports September 2014 U.S. Smartphone Subscriber Market Share**. 2014. Disponível em: <<http://www.comscore.com/Insights/Market-Rankings/comScore-Reports-September-2014-US-Smartphone-Subscriber-Market-Share>>. Acesso em: 08 nov. 2015.
- DEVELOPER. **The Google Play Opportunity**. 2015. Disponível em: <<http://developer.android.com/intl/pt-br/distribute/googleplay/about.html>>. Acesso em: 23 out. 2015.
- ESPOSITO, Dino. **Compreendendo o poder dos WebSockets**. 2012. Disponível em: <<http://msdn.microsoft.com/pt-br/magazine/hh975342.aspx>>. Acesso em: 18 nov. 2014.
- FARIA, Thiago. **Java EE 7: com JSF, PrimeFaces e CDI**. Minas Gerais: Algaworks, 2013. 197 p.
- FURUKAWA, Y.. Web-based control application using WebSocket. In: ICALEPCS, 13., 2011, Grenobla. **Proceedings...** Grenobla: Esrf, 2011. p. 673 - 675. Disponível em: <<https://accelconf.web.cern.ch/accelconf/icalepcs2011/papers/wemau010.pdf>>. Acesso em: 17 nov. 2014.
- GOK, Nizamettin; KHANNA, Nitin. **Building Hybrid Android Apps with Java and JavaScript**. Sebastopol: O'reilly Media, 2013. 156 p.

- JACKSON, Wallace. **Android Apps for Absolute Beginners**. Nova York: Apress, 2011. 344 p.
- LECHETA, Ricardo R. **Google Android: Aprenda a criar aplicações para dispositivos móveis com o Android SDK**. 3ª.ed., São Paulo: Novatec Editora, 2013.
- LUCKOW, Décio Heinzemann; MELO, Alexandre Altair de. **Programação Java para a web**. São Paulo: Novatec, 2010. 638 p.
- MONTEIRO, Maria de Fátima; SOBRAL FILHO, Dário C. Exercício físico e o controle da pressão arterial. **Revista Brasileira de Medicina do Esporte**, Niterói, v. 10, n. 6, p.513-516, nov./dez. 2004. Disponível em: <<http://www.scielo.br/pdf/rbme/v10n6/a08v10n6.pdf>>. Acesso em: 8 set. 2014.
- ORACLE. **JDBC**. Disponível em: <<http://www.oracle.com/technetwork/java/javase/jdbc/index.html>>. Acesso em: 24 nov. 2014.
- POTTS, Stephen; KOPACK, Mike. **Aprenda em 24 horas web services**. Rio de Janeiro: Elsevier, 2003. 367 p.
- UBL, Malte; KITAMURA, Eiji. **Apresentando WebSockets**: trazendo soquetes para a web. 2010. Disponível em: <<http://www.html5rocks.com/pt/tutorials/websockets/basics/>>. Acesso em: 20 nov. 2014.
- WANG, Vanessa; SALIM, Frank; MOSKOVITS, Peter. **The definitive guide to HTML5 WebSocket**: Build real-time applications with HTML5. New York: Apress, 2013. 208 p.
- WEB SOCKET. **What is web socket?**. Disponível em: <<http://www.websocket.org/index.html>>. Acesso em: 17 nov. 2014.
- WOLFSON, Mike; FELKER, Donn. **Android Developer Tools Essentials**. Sebastopol: O'reilly Media, 2013. 250 p.

