

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE INFORMÁTICA
ESPECIALIZAÇÃO EM DESENVOLVIMENTO PARA DISPOSITIVOS MÓVEIS**

VICENTE BERWANGER LLIVI IBANEZ

**DESENVOLVIMENTO DE UM APLICATIVO MÓVEL PARA ANÁLISE TÉCNICA
DE AÇÕES**

MONOGRAFIA DE ESPECIALIZAÇÃO

**Curitiba
2018**

VICENTE BERWANGER LLIVI IBANEZ

**DESENVOLVIMENTO DE UM APLICATIVO MÓVEL PARA ANÁLISE TÉCNICA
DE AÇÕES**

Monografia apresentada ao Departamento de Informática da Universidade Tecnológica Federal do Paraná como requisito parcial à obtenção do título de especialista em desenvolvimento para dispositivos móveis.

Orientador:
Prof. Adriano Francisco Ronszcka

**Curitiba
2018**



Ministério da Educação
UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
Câmpus Curitiba
Diretoria de Pesquisa e Pós-Graduação
Departamento Acadêmico de Informática
*Coordenação do Curso de Especialização em Desenvolvimento
para Dispositivos Móveis*

TERMO DE APROVAÇÃO

“Desenvolvimento de um Aplicativo Móvel para Análise Técnica de Ações”

por

“Vicente Berwanger Lliví Ibanez”

Este Trabalho de Conclusão de Curso foi apresentado às 19:30 do dia 15 de fevereiro de 2018 na sala B201 como requisito parcial à obtenção do grau de Especialista em Desenvolvimento para Dispositivos Móveis na Universidade Tecnológica Federal do Paraná - UTFPR - Campus Curitiba. O(a) aluno(a) foi arguido pela Banca de Avaliação abaixo assinados. Após deliberação, a Banca de Avaliação considerou o trabalho aprovado.

<hr/> <p>Prof. Adriano Francisco Ronszcka (Presidente/Orientador – externo)</p>	<hr/> <p>Profa. Maria Claudia Figueiredo Pereira Emer (Avaliador 1 – UTFPR/Curitiba)</p>
<hr/> <p>Prof. Robson Ribeiro Linhares (Avaliador 2 – UTFPR/Curitiba)</p>	

“A Ata de Aprovação assinada encontra-se na Coordenação do Curso.”

RESUMO

IBANEZ, Vicente B. L. DESENVOLVIMENTO DE UM APLICATIVO MÓVEL PARA ANÁLISE TÉCNICA DE AÇÕES. 2018. 56 f. Monografia (Especialização em Desenvolvimento para Dispositivos Móveis) – Departamento de Informática, Universidade Tecnológica Federal do Paraná. Curitiba, 2018.

Os dispositivos móveis auxiliam e facilitam diversas atividades do dia a dia, tornando-as mais ágeis e permitindo o acesso a informação em diversos lugares. O presente trabalho apresenta o desenvolvimento de um aplicativo para smartphones que auxilia o usuário na tomada de decisão para investimentos em ações negociadas na bolsa de valores. Ademais, o trabalho descreve os principais conceitos de dispositivos móveis, sistemas operacionais, engenharia de software e mercado de ações e suas análises. Por fim, será apresentado o aplicativo desenvolvido, os resultados obtidos e a avaliação do aplicativo por investidores. Em suma, o aplicativo foi avaliado pelos investidores, os quais relataram um aumento no número de operações realizadas. Neste âmbito, o aplicativo está ajudando investidores a aproveitarem mais oportunidades de compra, aumentando o ganho de capital com o mercado de ações.

Palavras-chave: Dispositivos móveis. Mercado de Ações. Análise Técnica.

ABSTRACT

IBANEZ, Vicente B. L. DEVELOPMENT OF A MOBILE APP FOR TECHNICAL ANALYSIS OF STOCKS. 2018. 56 f. Monografia (Especialização em Desenvolvimento para Dispositivos Móveis) – Departamento de Informática, Universidade Tecnológica Federal do Paraná. Curitiba, 2018.

Mobile devices help and simplify variety of day-to-day tasks by making them more agile and allowing the users to access information from various places. The present work presents the development of a mobile app that assists the user in the decision making for investments in stocks traded in the stock exchange. Furthermore, the paper describes the main concepts of mobile devices, operating systems, software engineering and stock market and their analysis. Finally, the developed application, the obtained results and the appraisal of the application by investors will be presented. In short, the application was evaluated by investors, who reported an increase in the number of transactions performed. In this context, the application is helping investors to take advantage of more buying opportunities, increasing capital gain with the stock market.

Keywords: Mobile devices. Stock market. Technical Analysis.

LISTA DE FIGURAS

Figura 1 - Remessa de dispositivos conectados em todo o mundo.	13
Figura 2 - Estrutura de um sistema de computação típico.	14
Figura 3 - Fatia de mercado de SOs de <i>smatphones</i>	15
Figura 4 - <i>Framework</i> Android.	16
Figura 5 - Arquitetura do iOS.	18
Figura 6 - Arquitetura do Windows Phone.	19
Figura 7 - Exemplo de diagrama de caso de uso.	22
Figura 8 - Exemplo de diagrama de classes.	23
Figura 9 - Exemplo de diagrama de atividades.	25
Figura 10 - Exemplo de diagrama de sequências.	26
Figura 11 - Gráfico de <i>candlestick</i> de IBOV de 14/06/2017 a 06/09/2017.	28
Figura 12 - Formação dos <i>candles</i>	28
Figura 13 - Exemplo de Doji em 10/02/2016 em CPLE3.	29
Figura 14 - Exemplo de Martelo em 10/11/2015 em ENGIE.	30
Figura 15 - Exemplo de Marobozu em 03/03/3016 em USIM5.	31
Figura 16 - Exemplo de Bebê Abandonado (a) de fundo e (b) de alta.	31
Figura 17 - Exemplo de Bandas de Bollinger.	32
Figura 18 - Diagrama de Casos de Uso.	36
Figura 19 - Diagrama de Classes.	40
Figura 20 - Diagrama de Atividades - Atualizar Lista.	41
Figura 21 - Diagrama de Atividades - Selecionar Ação.	42
Figura 22 - Diagrama de Sequências.	43
Figura 23 – Resultado da chamada ao <i>web service</i>	45
Figura 24 – Tela inicial do aplicativo com a lista de ações e botão para atualizar. ...	46
Figura 25 – Tela inicial do aplicativo com exemplo de pontos de compra e ação favorita.	47
Figura 26 – Tela de detalhes da ação e botão favorita.	48

LISTA DE SIGLAS

FEBRABAN	Federação Brasileira de Bancos
IDC	<i>International Data Corporation</i>
SO	Sistema Operacional
OHA	<i>Open Hanset Alliance</i>
API	<i>Application Programming Interface</i>
RTOS	<i>Real-Time Operating System</i>
XML	<i>Extensible Markup Language</i>
JSON	<i>Javascript Object Notation</i>
CSV	<i>Comma Separated Values</i>
UML	<i>Unified Modeling Language</i>
OMT	<i>Object Modeling Technique</i>
OOSE	<i>Object-Oriented Software Engineering</i>
OMG	<i>Object Management Group</i>

SUMÁRIO

1 INTRODUÇÃO	9
1 OBJETIVO GERAL	10
1.2 OBJETIVOS ESPECÍFICOS	10
1.3 MOTIVAÇÃO	10
1.4 ESCOPO DO TRABALHO	11
1.5 ESTRUTURA DA MONOGRAFIA	11
2 REFERENCIAL TEÓRICO	12
2.1 DISPOSITIVOS MÓVEIS	12
2.1.1 Sistemas Operacionais	14
2.1.1.1 Android	16
2.1.1.2 iOS	18
2.1.1.3 Windows Phone	19
2.2 TECNOLOGIAS AUXILIARES NO DESENVOLVIMENTO DE APLICATIVOS	20
2.2.1 <i>Web services</i>	20
2.2.1 Banco de dados	20
2.3 MODELAGEM DE SOFTWARE	21
2.3.1 UML	21
2.3.1.1 Diagrama de Casos de Uso	22
2.3.1.2 Diagrama de Classes	23
2.3.1.3 Diagrama de Atividades	24
2.3.1.4 Diagrama de Sequências	25
2.4 MERCADO DE AÇÕES	26
2.4.1 <i>Candlestick</i>	27
2.4.1.1 Doji	29
2.4.1.2 Martelo	30
2.4.1.3 Marobozu	30
2.4.1.4 Bebê Abandonado	31
2.4.1.5 Bandas de Bollinger	32
3 METODOLOGIA	33
4 DESENVOLVIMENTO	34
4.1 LEVANTAMENTO DE REQUISITOS	34
4.1.1 Requisitos Funcionais	34
4.1.2 Requisitos não funcionais	35
4.2 MODELAGEM	35
4.2.1 Diagrama de Casos de Uso	35
4.2.1.1 Consultar cotações	36

4.2.1.2 Atualizar cotações	37
4.2.1.3 Calcular ponto de compra	38
4.2.1.4 Consultar detalhes da ação.....	38
4.2.1.5 Definir ação como favorita.....	39
4.2.2 Diagrama de Classes	39
4.2.3 Diagrama de Atividades	41
4.2.1 Diagrama de Sequências	42
4.3 CONSTRUÇÃO DO APLICATIVO.....	43
4.3.1 Cálculo do ponto de compra	44
4.3.2 Busca de cotações	44
4.3.3 Funcionamento do aplicativo.....	46
4.4 TESTES.....	48
5. RESULTADOS OBTIDOS	50
6. CONCLUSÃO	51
REFERÊNCIAS BIBLIOGRÁFICAS	53

1 INTRODUÇÃO

Atualmente, o Brasil conta com apenas 600 mil investidores cadastrados na bolsa, segundo dados da BM&F BOVESPA (2017). Número ainda pequeno se comparado aos mais de 200 milhões de habitantes do país¹, o que indica um alto potencial de crescimento neste segmento. Estes investidores utilizam diversas fontes de informações e ferramentas computacionais para definir seus investimentos.

Identificar as flutuações e tendências do mercado financeiro e determinar as melhores oportunidades de compra e venda de ativos é uma atividade complexa que demanda conhecimento e disponibilidade de tempo do investidor para analisar uma grande quantidade de informações. Embora existam ferramentas de software que podem tornar esta tarefa menos penosa, garantindo agilidade e também maximizando os ganhos do investidor, a maioria delas ainda necessita de um computador, limitando o usuário que precisa dispor da sua mobilidade para a utilização destes softwares.

Os dispositivos móveis estão a cada dia mais presentes no cotidiano dos brasileiros, segundo projeção da Fundação Getúlio Vargas (2017), o Brasil terá um smartphone para cada habitante até o final de 2017. Estes aparelhos contam com recursos de software e hardware que se assemelham aos encontrados em computadores. Além disso, possuem acesso à internet por redes móveis, o que garante o acesso a informações em diferentes locais. Esta popularização e facilidade de uso está refletida no relatório da Federação Brasileira de Bancos (FEBRABAN), que identificou que o volume de transações via *mobile banking* cresceu 138% em 2015 (FEBRABAN, 2015). No entanto, apesar do número de utilizações de serviços bancários por smartphones ter crescido, a oferta de ferramentas para dispositivos móveis que auxiliam a tomada de decisão para investimentos ainda é escassa.

¹ A população do Brasil segundo dados do IBGE (2018) é de aproximadamente 208 milhões de pessoas.

1 OBJETIVO GERAL

O presente trabalho tem o objetivo de desenvolver um aplicativo para dispositivos móveis que facilite a análise do mercado financeiro em bolsa de valores.

1.2 OBJETIVOS ESPECÍFICOS

Para alcançar o objetivo geral é necessário:

- estudar técnicas de análise de operações na bolsa de valores;
- automatizar a análise técnica de ações;
- aumentar o número de operações realizadas pelo usuário.

1.3 MOTIVAÇÃO

Em contato com investidores do mercado financeiro observou-se uma demanda de ferramentas para quem utiliza a análise técnica em seus investimentos, principalmente de ferramentas que ofereçam maior mobilidade aos usuários.

A análise técnica de ativos e busca de oportunidades na bolsa de valores é uma tarefa que demanda conhecimento do mercado financeiro e tempo para a análise das informações. Além disso, o investidor necessita de um computador para realizar suas avaliações dos ativos. Para investidores que realizam *swing trade* (compra de ação em um dia com o objetivo de venda com lucro no dia seguinte), o melhor horário para realizar uma compra é em até 30 minutos antes do final do pregão do dia, pois neste momento os preços do dia já estão bastante definidos e com menor volatilidade. Devido a isso, os investidores acabam perdendo oportunidades de ganhos por não dispenderem de um computador nos melhores momentos de compra ou por não conseguirem identificar todas as oportunidades em um curto intervalo de tempo.

A automatização dessas análises por meio de um aplicativo para dispositivos móveis auxiliará os usuários a encontrar mais oportunidades de compra com maior eficiência, principalmente para os investidores não profissionais que não possuem tanto conhecimento nem tempo para realizar esta atividade.

1.4 ESCOPO DO TRABALHO

No presente trabalho será desenvolvido um aplicativo para dispositivos móveis que calcule e informe o usuário de uma oportunidade de compra de ações. A aplicação deverá implementar as análises e estratégias de mercado necessárias e escolhidas pelos investidores interessados na aplicação.

1.5 ESTRUTURA DA MONOGRAFIA

Este trabalho está escrito em cinco capítulos. No Capítulo 2 são apresentados os principais conceitos teóricos necessários para a total compreensão do trabalho.

O Capítulo 3, por sua vez, apresenta a metodologia utilizada para o trabalho, contextualizando detalhadamente todas as abordagens utilizadas.

O Capítulo 4 apresenta o desenvolvimento do trabalho.

O Capítulo 5 se destina a análise dos resultados obtidos.

Por fim, o Capítulo 6 apresenta as conclusões do trabalho e sugere possíveis melhorias e trabalhos futuros.

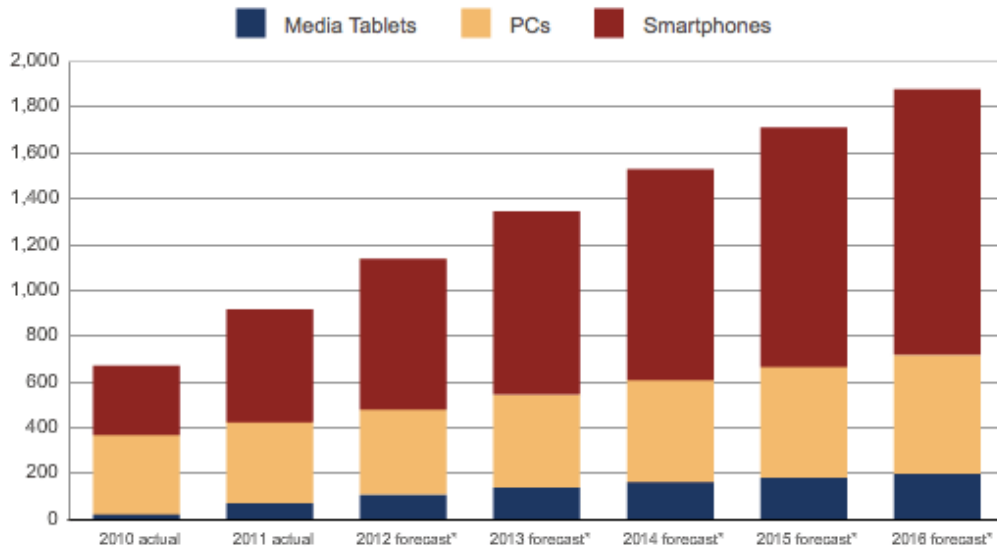
2 REFERENCIAL TEÓRICO

Este capítulo apresenta o embasamento teórico estudado pertinente para o desenvolvimento do aplicativo móvel. Nesse âmbito, a Seção 2.1 apresenta os principais conceitos sobre dispositivos móveis e os dispositivos mais usados. A Seção 2.2 mostra as principais tecnologias auxiliares ao desenvolvimento de aplicativos que serão utilizadas no trabalho. Já a Seção 2.3 apresenta conceitos de engenharia de software, necessários para o desenvolvimento do trabalho. E, por fim, a Seção 2.4 explica conceitos fundamentais do mercado de ações.

2.1 DISPOSITIVOS MÓVEIS

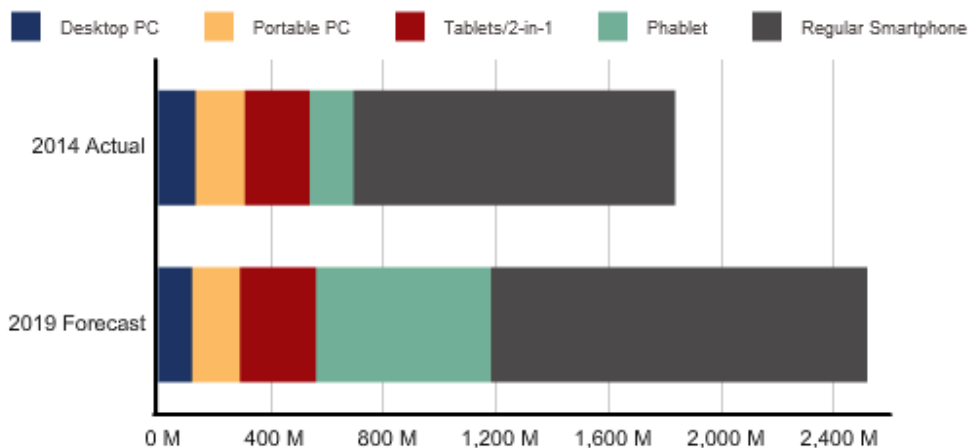
Dispositivos móveis podem ser definidos como equipamentos que permitem ao usuário ter acesso a serviços, funções e aplicações independentes da sua localização, garantindo que o usuário possa se deslocar facilmente caso necessário. Existem diversos tipos de dispositivos móveis disponíveis no mercado - PDAs (computadores de dimensões reduzidas), *E-readers* (leitores de livros), telefones celulares e smartphones são alguns exemplos destes dispositivos. Pelos relatórios do *International Data Corporation* (IDC, 2017) podemos observar que o número de smartphones a cada ano se torna mais significativo em relação aos computadores e telefones celulares convencionais, como pode ser observado na Figura 1.

Worldwide Smart Connected Device Shipments, 2010-2016 (Unit Millions)



(a) De 2010 a 2016

WW Smart Connected Device Shipments, 2014 and 2019



Source : IDC WW Quarterly Smart Connected Device Tracker, March 2015

(b) De 2014 a 2019

Figura 1 - Remessa de dispositivos conectados em todo o mundo.

Fonte: *International Data Corporation* IDC.

Na Figura 1 podemos observar que os smartphones são os eletrônicos com maior crescimento nas vendas, dentre os chamados dispositivos conectados, superando até mesmo dispositivos com maior capacidade de hardware como notebooks. Isso pode ser explicado porque os smartphones oferecerem uma grande diversidade de funções aliada a conveniência de ser um dispositivo de tamanho reduzido. Para oferecer esta gama de funcionalidades os smartphones precisam de um sistema operacional.

2.1.1 Sistemas Operacionais

Sistema operacional (SO) é o software responsável pelo gerenciamento de recursos e criação do ambiente de trabalho de um equipamento. Segundo Maziero (2017), um sistema operacional é uma camada intermediária de software que opera entre os programas e aplicativos voltados ao usuário final e o hardware. O sistema operacional abstrai aspectos de baixo nível para serem utilizados por outras aplicações, como *drivers* por exemplo, além de realizar gerenciamento dos recursos físicos do dispositivo como, por exemplo, a memória. Em alto nível, ele realiza o tratamento da interface gráfica e também de aplicativos utilitários. A Figura 2 demonstra a arquitetura de um sistema computacional típico.

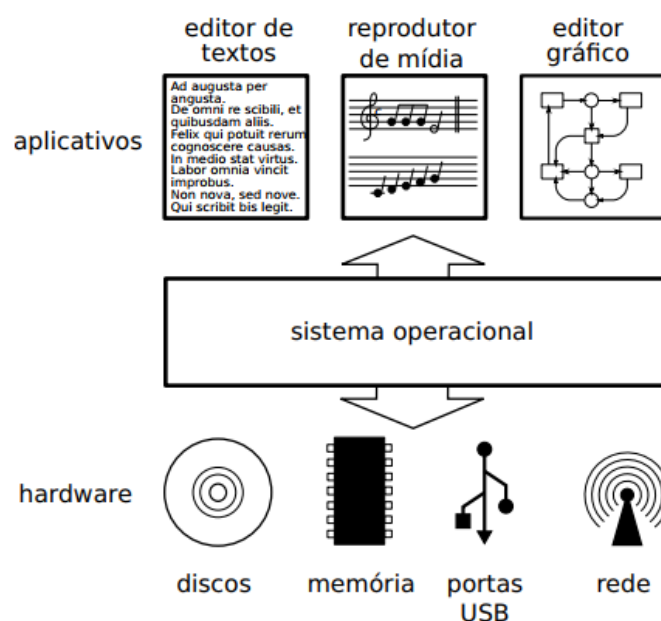


Figura 2 - Estrutura de um sistema de computação típico.

Fonte: Maziero (2017).

Na Figura 2 podemos observar o sistema operacional funcionando como a parte intermediária do sistema de computação, sendo a conexão entre os elementos de hardware (discos, memória, portas de entrada e saída e rede) e os aplicativos destinados ao usuário (editor de texto, reproduutor de mídia e editor gráfico).

Em se tratando de dispositivos móveis três sistemas operacionais dominam o mercado de smartphones, obtendo juntos mais de 98% do mercado, segundo pesquisas do IDC, Figura 3. Por isso, as próximas subseções são dedicadas a detalhar cada um dos três principais sistemas, o Android, o iOS e o Windows Phone.

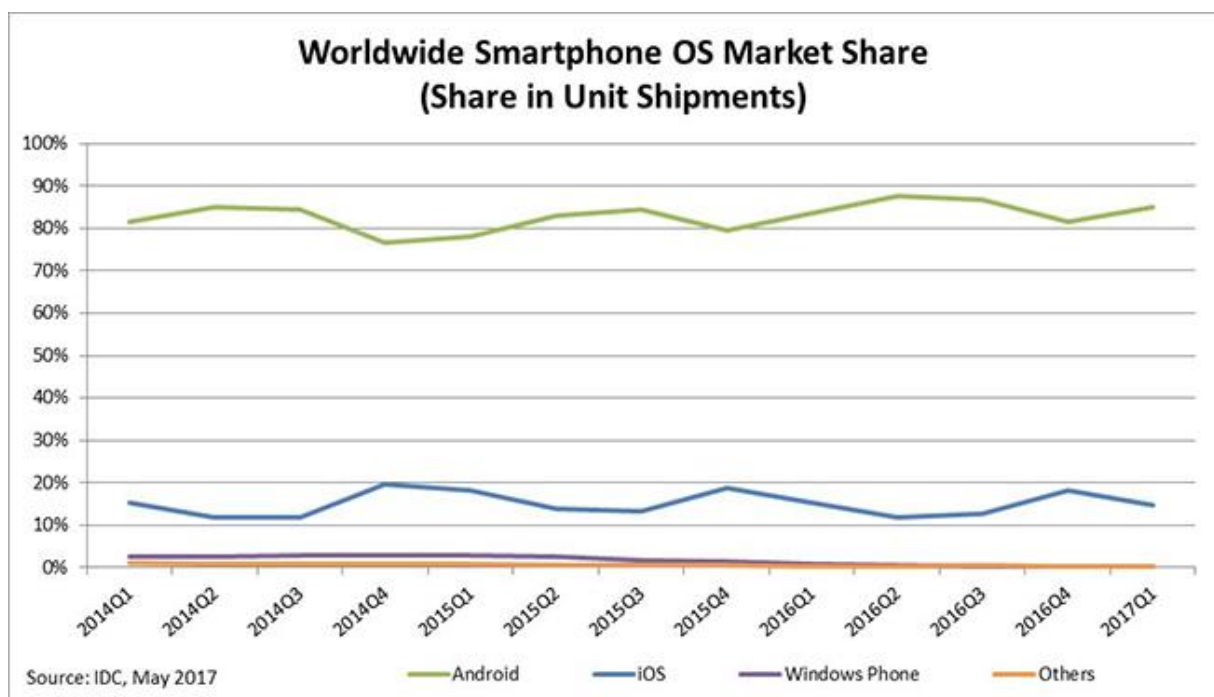


Figura 3 - Fatia de mercado de SOs de *smatphones*.

Fonte: IDC.

Pelas pesquisas do IDC, ilustradas através da Figura 3, verificamos que o Android é o sistema operacional que detém a maior fatia do mercado sendo utilizado em aproximadamente 85% dos dispositivos durante os 4 anos pesquisados. Além disso, pode se observar que o iOS é o segundo SO mais utilizado porém ele detém apenas aproximadamente 15% do mercado.

2.1.1.1 Android

O Android representa um conjunto de aplicativos que inclui um SO *open source* baseado no *kernel* do Linux desenvolvido para dispositivos móveis e é atualmente coordenado pela Open Handset Alliance (OHA). A Android Inc. foi fundada em 2003 com o objetivo de desenvolver softwares para dispositivos móveis. Ela foi adquirida pela Google em 2005 que continuou o desenvolvimento da plataforma com o intuito de ingressar neste mercado. Posteriormente o controle do projeto foi entregue para a OHA (*Open Handset Alliance*), um consórcio de tecnologia entre empresas do ramo de software, fabricantes de telefones, operadoras de telefonia e fabricantes de *chipsets*. Porém, foi apenas em 2007 que a OHA e a plataforma Android foram anunciados.

Além do sistema operacional, o Android conta com *middleware* e um conjunto de aplicações chave. As camadas da arquitetura do sistema podem ser observadas na Figura 4.



Figura 4 - *Framework* Android.

Fonte: Google (2017).

A arquitetura do Android é dividida em camadas, conforme observado na Figura 4, contendo os seguintes componentes:

- a) **Aplicações:** é a camada mais utilizada pelos desenvolvedores de aplicações para a plataforma. Nela estão os aplicativos básicos como alarme, navegador de internet, gerenciador de contatos, câmera, relógio, SMS, mapas e ligações por exemplo.
- b) **Framework Android:** camada contendo as principais *APIs (Application Programming Interface)* utilizadas pelos principais aplicativos do sistema e utilizadas pelos desenvolvedores. Estas *APIs* são o *ActivityManager*, *LocationManager*, *PackageManager*, *NotificationManager*, *ResourceManager*, *TelephonyManager*, *WindowManager* e *View*.
- c) **Bibliotecas Nativas:** é um conjunto de bibliotecas em C/C++ que são utilizadas pelos componentes do sistema. Estas bibliotecas fornecem diversas instruções para áudio, vídeo, gráficos, banco de dados e navegador que são abstraídas para os desenvolvedores através da camada de *framework* do Android.
- d) **Tempo de execução:** é a camada responsável pela execução das aplicações. Todo aplicativo Android tem um processo próprio que é executado por uma instância da máquina virtual Dalvik. A máquina virtual Dalvik garante a execução dos aplicativos, ela abstrai a camada de hardware do dispositivo sem que os desenvolvedores precisem se preocupar com os diversos tipos de hardware.
- e) **HAL:** é a camada de abstração de hardware (*HardwareAbstractionLayer*). O *HAL* define o tipo de interface padrão que os desenvolvedores de hardware devem implementar. Ela permite adicionar funcionalidades sem afetar os níveis superiores.
- f) **Linux Kernel:** esta camada funciona como uma camada de abstração entre o hardware e o software. Sendo a camada gerencial do sistema ela implementa os *drivers* de hardware, gerenciamento de memória e energia, segurança e rede, foi feita baseada no *kernel* do Linux versão 2.6.

2.1.1.2 iOS

O iOS é um sistema operacional Unix baseado no *kernel* Darwin, proprietário da Apple Inc. Inicialmente desenvolvido para o iPhone, hoje este SO também é utilizado no iPad e iPod. Ele foi apresentado em 2007 em conjunto com o iPhone. Os aplicativos para esta plataforma podem ser desenvolvidos em Objective-C ou em Swift.

No nível mais alto o iOS funciona como uma camada intermediária entre os aplicativos e o hardware do sistema. A arquitetura do iOS é dividida em quatro camadas, como pode ser observado na Figura 5.

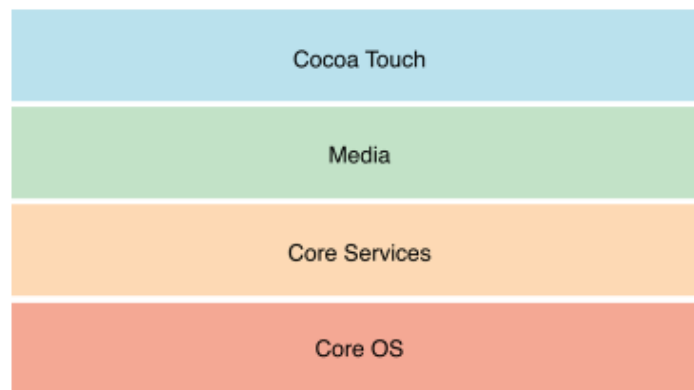


Figura 5 - Arquitetura do iOS.

Fonte: Apple (2014).

As funcionalidades de cada camada da arquitetura do iOS apresentada na Figura 5 são:

- a) *Cocoa Touch*: é a mais alta camada da arquitetura. Ela oferece o *framework* para a criação dos aplicativos para iOS. Ademais, ela fornece a infraestrutura básica para a aplicação como multitarefa, interface *touch*, notificações e demais serviços de alta camada.
- b) *Media*: a camada de mídia implementa todos os recursos gráficos, de áudio e vídeo, para serem utilizados pelos aplicativos.
- c) *Core Services*: esta camada incorpora todos os serviços fundamentais do sistema que são utilizados por todos os aplicativos, sendo a localização, *iCloud*, serviços de rede *SQLite* e suporte *XML* suas principais tecnologias.

- d) *Core OS*: é a camada mais baixa do sistema, na qual todas as demais camadas são construídas em cima. Ela faz todo o gerenciamento de hardware do sistema, tornando-o transparente e independente de hardware para os aplicativos.

2.1.1.3 Windows Phone

O Windows Phone é um sistema operacional para dispositivos móveis desenvolvido pela Microsoft. Ele foi lançado apenas em 2010, três anos mais tarde que o Android e o iOS. Uma das principais diferenças do Windows Phone é a sua interface de usuário baseada em uma nova linguagem visual, que a própria Microsoft chamou de Metro. Ela define o *design* utilizando mais tipografia e menos elementos gráficos. Segundo a empresa o Metro é uma linguagem de *design* que tem este nome porque é moderna e limpa, é rápida e em movimento, é sobre conteúdo e tipografia, como apresentado por Pahkala (2012).

Além disso, outro diferencial do Windows Phone é ele ser um RTOS (*real-time operating system*) o que significa que as respostas aos eventos tem um tempo de resposta predefinido. Os aplicativos para este sistema são usualmente escritos em C# e XAML. A arquitetura do sistema pode ser observada na Figura 6.

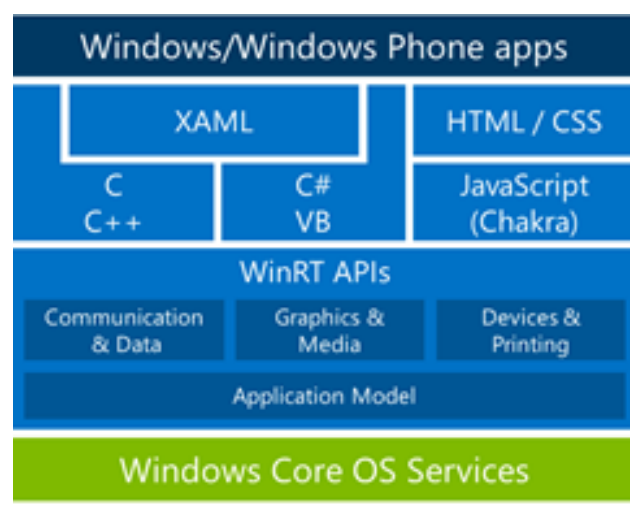


Figura 6 - Arquitetura do Windows Phone.

Fonte: Microsoft (2014).

Em 2017 a Microsoft descontinuou o nome Windows Phone e adotou o nome Windows 10 Mobile para seu novo sistema operacional, com o objetivo de melhorar a integração dos seus sistemas operacionais.

2.2 TECNOLOGIAS AUXILIARES NO DESENVOLVIMENTO DE APLICATIVOS

Além das particularidades de cada sistema operacional, outras tecnologias também são utilizadas no desenvolvimento de aplicativos para dispositivos móveis. Nas próximas seções serão descritas as principais tecnologias utilizadas no trabalho.

2.2.1 *Web services*

Os *web services* são uma tecnologia desenvolvida para a comunicação entre máquinas utilizando a internet. Eles tornam os sistemas independentes fazendo com que softwares desenvolvidos em diferentes linguagens de programação sejam compatíveis e consigam se comunicar sem intervenção humana (LECHETA, 2015).

Os *web services* fornecem informações por meio da transferência de dados em formatos conhecidos, como por exemplo XML (*Extensible Markup Language*), JSON (*Javascript Object Notation*) e CSV (*Comma Separated Values*), a uma aplicação cliente. Esta pode ser uma aplicação móvel que implementa uma interface para apresentar os dados ao usuário de uma forma amigável ou ainda pode ser outro servidor.

Uma forma de criar um *web service* é utilizando REST (*Representational State Transfer*). O REST é uma abstração da arquitetura web que define a arquitetura de componentes, conectores e elementos de um sistema distribuído. Ele utiliza o protocolo HTTP na sua comunicação, sendo cada comunicação independente (LECHETA, 2015).

2.2.1 Banco de dados

Um banco de dados é um conjunto de dados inter-relacionados que definem informações sobre um determinado assunto, como descrito por Silberschatz (2012).

Ele promove a abstração dos dados isolando dos usuários e das aplicações os detalhes internos do armazenamento de informações.

Para o controle dos bancos de dados, são utilizados sistemas de gerenciamento de banco de dados. O SQLite é um exemplo de biblioteca que implementa um software de banco de dados muito utilizado em dispositivos de baixa capacidade devido ao seu projeto compacto. Entre as vantagens do SQLite para sistemas embarcados se destacam principalmente o código fonte de domínio público, não exigir nenhuma instalação ou pré-configuração para o uso, o tamanho reduzido e não possuir dependências externas.

2.3 MODELAGEM DE SOFTWARE

Para o desenvolvimento de um software robusto e de qualidade, que permita escalabilidade e manutenção simples, é necessário seguir práticas e padrões de projeto que garantam a correta especificação do software.

Uma forma de auxiliar o desenvolvimento do programa é especificar os requisitos do software e utilizar modelos que abstraíam o sistema a ser criado. Um método comum para modelagem de software é a UML (*Unified Modeling Language*).

2.3.1 UML

A UML é uma linguagem de modelagem que surgiu em meados da década de 90 pela união de três metodologias de modelagem, o método de Booch, o OMT (*Object Modeling Technique*) e o OOSE (*Object-Oriented Software Engineering*). Em 1997 o OMG (*Object Management Group*) adotou a UML como linguagem padrão de modelagem (OMG, 2017).

A UML permite representar um sistema para facilitar a especificação, desenvolvimento, documentação e visualização de artefatos de software orientado a objeto (OMG, 2017). Ela é composta por itens (componentes, classes, interfaces, casos de uso), relacionamentos (dependência, associação, generalização, realização) e diagramas (classes, objetos, sequência, casos de uso). Cada diagrama é uma visão específica do sistema que facilita a visualização sobre determinada perspectiva. Estes diagramas são divididos em dois grupos que são os estáticos e os dinâmicos (OMG, 2017). Os diagramas estáticos definem as características estruturais do sistema e

suas partes, como classes, métodos e atributos. Por sua vez, os diagramas dinâmicos apresentam as características comportamentais do sistema, eles demonstram a comunicação entre os objetos e suas respostas a determinados eventos. A UML possui no total 14 tipos de diagramas, neste trabalho serão apresentados quatro dos principais diagramas, sendo dois de cada grupo.

2.3.1.1 Diagrama de Casos de Uso

O diagrama de casos de uso é um dos diagramas estáticos. Ele tem a função de apresentar os requisitos funcionais necessários para o sistema e representá-los graficamente pelos casos de uso, mostrando as interações dos atores com o sistema. Através desse diagrama é possível observar as principais funcionalidades do sistema. Os principais elementos de um diagrama de casos de uso são os atores e os casos de uso. Os atores são representações de entidades externas que interagem com o sistema em algum momento de sua execução. Os casos de uso são uma sequência de ações que o sistema realiza ao interagir com os atores.

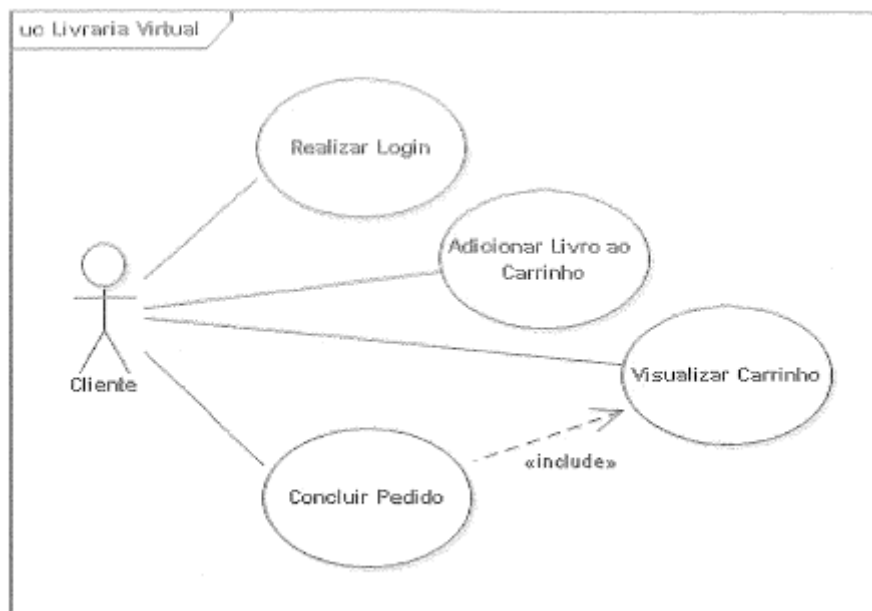


Figura 7 - Exemplo de diagrama de caso de uso.

Fonte: GUEDES (2011).

A Figura 7 mostra um exemplo de diagrama de casos de uso onde o ator cliente interage com o sistema, podendo realizar *login*, adicionar livro ao carrinho, visualizar o carrinho ou concluir o pedido.

2.3.1.2 Diagrama de Classes

O diagrama de classes também faz parte do grupo de diagramas estáticos da UML. Este diagrama apresenta graficamente a estrutura dos objetos do sistema, projetada para atender os requisitos funcionais do programa seguindo o paradigma de orientado a objeto. Ele é utilizado para auxiliar na análise de requisitos do sistema e durante todo o seu projeto. Os principais elementos de um diagrama de classes são as classes, seus relacionamentos e cardinalidades.

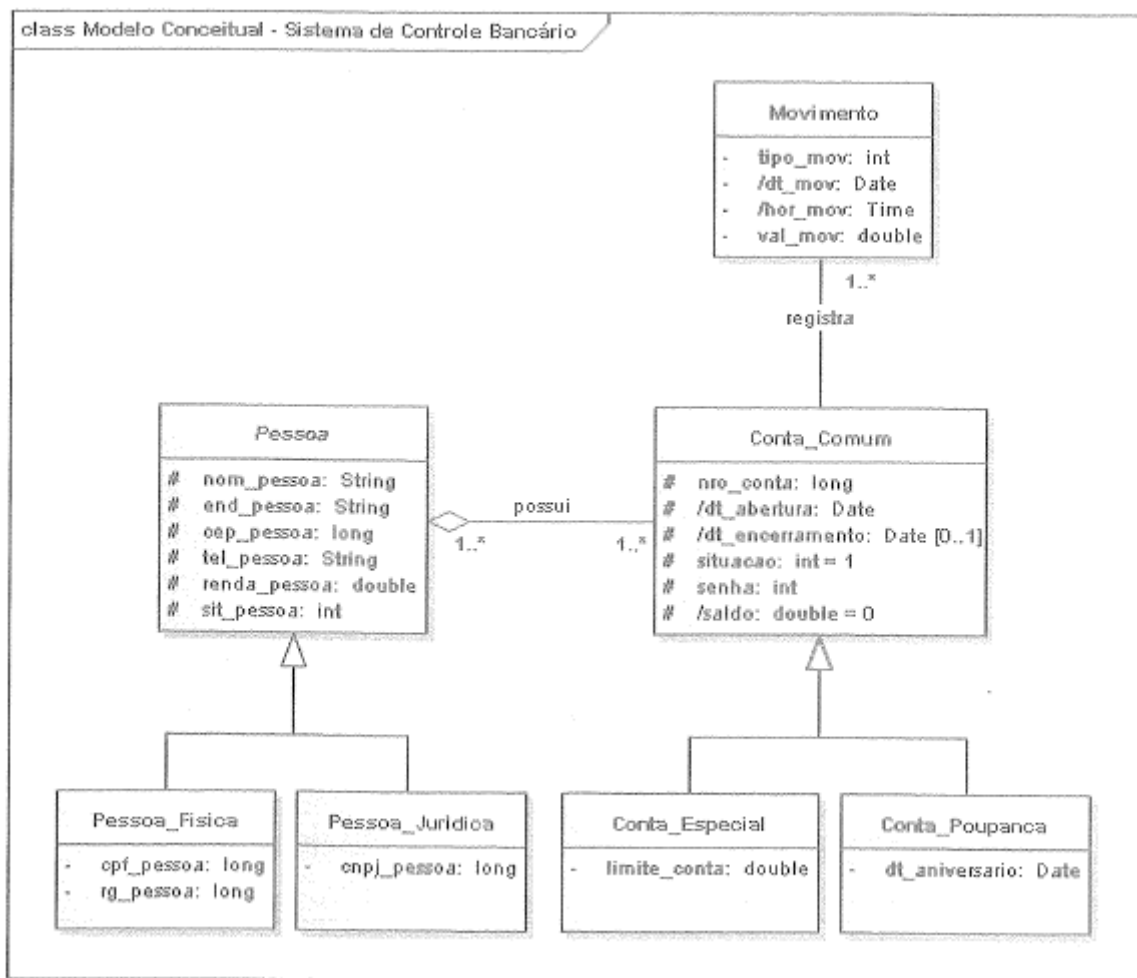


Figura 8 - Exemplo de diagrama de classes.

Fonte: GUEDES (2011).

Na Figura 8 observamos um exemplo de diagrama de classes para um sistema de controle bancário. Nele existem classes para definir os tipos de pessoa, física ou jurídica e seus dados pessoais e atributos. O diagrama também apresenta classes para os diferentes tipos de conta bancária e outra classe para definir as movimentações financeiras.

2.3.1.3 Diagrama de Atividades

O diagrama de atividades por sua vez é um dos diagramas dinâmicos da UML. Este diagrama exhibe o fluxo das atividades do sistema necessárias para executar determinado processo, mostrando as dependências entre cada atividade. Ele pode ser analisado como um fluxograma de processo do sistema. Os principais elementos deste diagrama são as atividades, ações, controle e arestas.

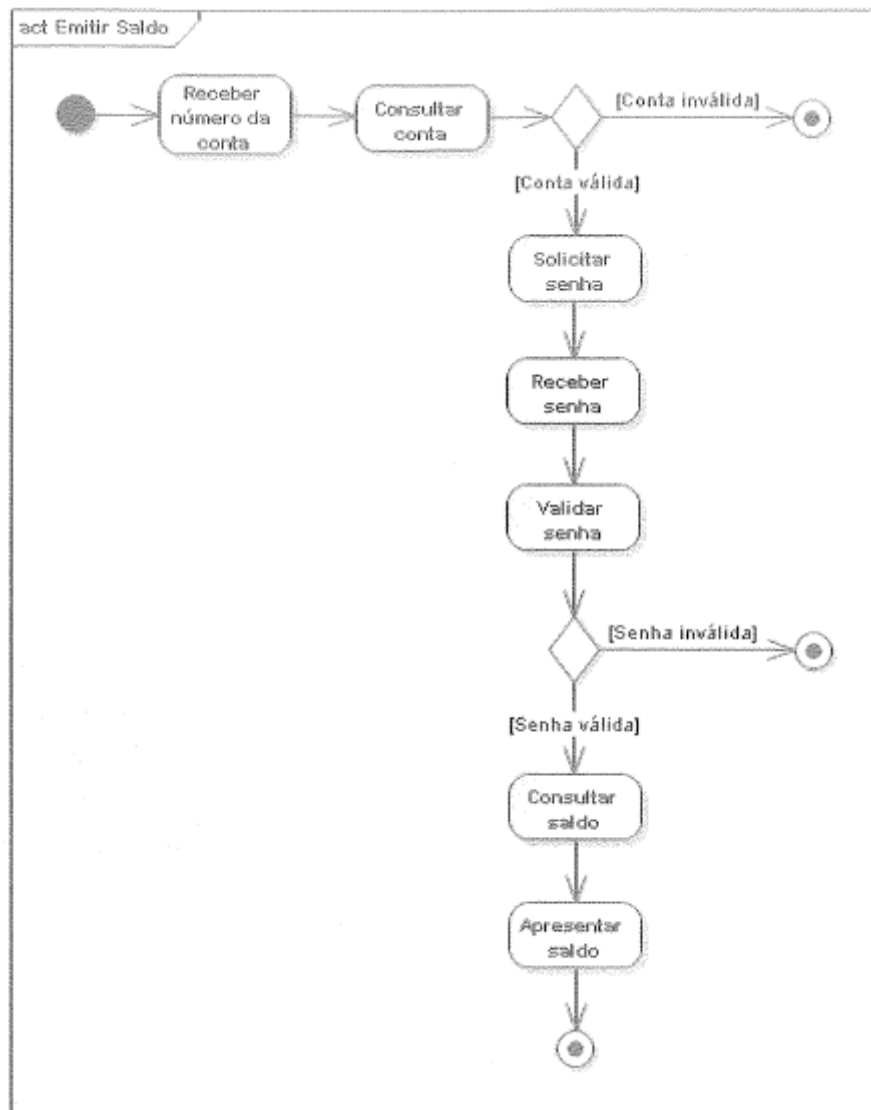


Figura 9 - Exemplo de diagrama de atividades.

Fonte: GUEDES (2011).

A Figura 9 exibe um exemplo de diagrama de atividades para um emitir o saldo de um usuário de um sistema bancário.

2.3.1.4 Diagrama de Sequências

O diagrama de sequências também é um diagrama dinâmico. Ele apresenta a troca de mensagens entre os objetos e a sequência dos processos no decorrer do tempo de uma operação. Os elementos de um diagrama de sequência são a linha de vida, as mensagens, interações e os fragmentos.

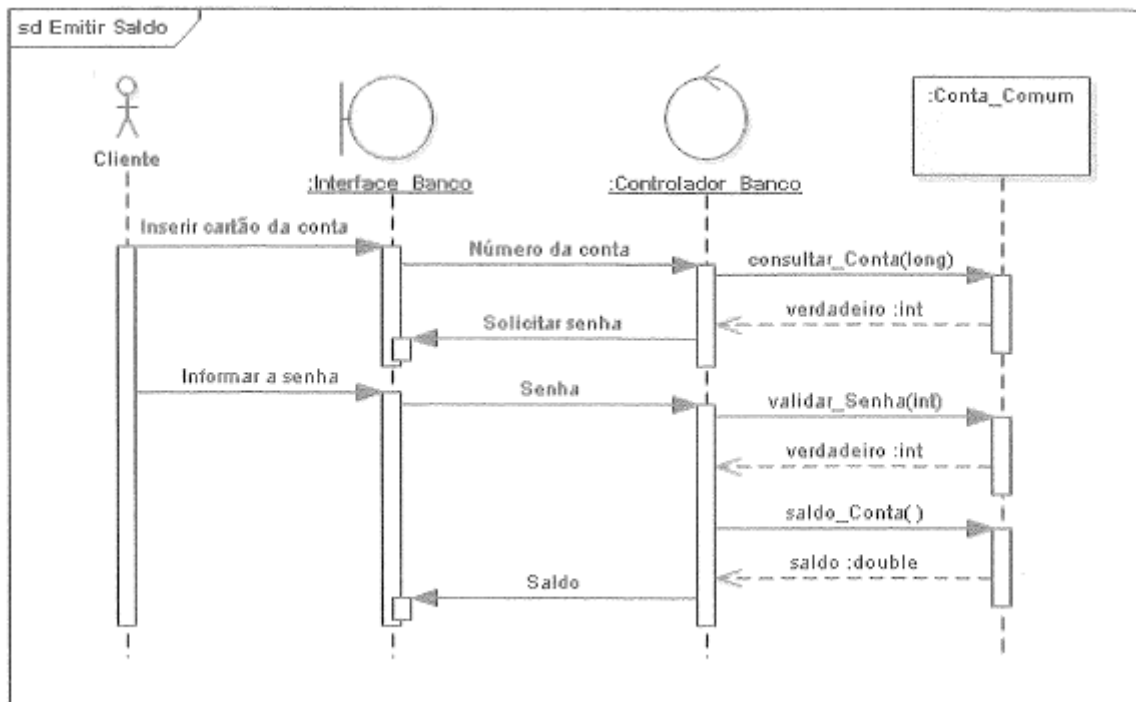


Figura 10 - Exemplo de diagrama de seqüências.

Fonte: GUEDES (2011).

A Figura 10 apresenta um exemplo de fragmento de diagrama de seqüências que mostra a troca de mensagens realizada para a emissão de saldo em um sistema bancário.

2.4 MERCADO DE AÇÕES

Os investimentos no mercado de ações podem ser feitos seguindo duas vertentes de análises, a análise fundamentalista e a análise técnica.

Por meio da primeira, o investidor verifica os últimos balanços da empresa, seus ativos, estuda seus produtos e o mercado na qual está inserida. Após esta análise, o investidor tem embasamento para decidir se quer investir na companhia e ser sócio dela por um período de médio a longo prazo. Segundo Vieira (2016), esta forma de análise é normalmente utilizada por investidores mais conservadores.

Por outro lado, a análise técnica, ou análise gráfica, visa identificar as oscilações e tendências de alta ou de baixa do mercado baseando-se na interpretação dos gráficos de preços, volumes, médias e demais indicadores. Por ser uma análise de tendências grafistas baseada na projeção de padrões e de expectativas de preços

futuros, o horizonte de investimento é mais curto, podendo ser de semanas, dias, horas ou até mesmo minutos.

Segundo Vieira (2016), a análise gráfica possui três premissas. A primeira define que o preço desconta tudo, ou seja, qualquer notícia, relatório ou atividade referente a determinado ativo já está refletida em seu preço através de uma ordem de compra ou de venda. A segunda define que o mercado se move em tendências e é importante identificar quando elas chegam ao fim. A terceira define que os padrões de movimentos se repetem ao longo do tempo. Nesse âmbito, existem diversas ferramentas e técnicas de análises gráficas para nortear os investidores em suas decisões, sendo uma delas o *candlestick*.

2.4.1 *Candlestick*

O gráfico de *candlestick* tem este nome, pois o desenho das representações gráficas de cada período se assemelha ao corpo de uma vela com pavio. Esta técnica teve origem no Japão nas bolsas de arroz de Osaka e foi descoberta e trazida para o ocidente pelo operador Steve Nilson na década de 1980, conforme apresentado por Debastiani (2016). Este tipo de gráfico se tornou muito utilizado pelos operadores do mercado devido as diversas informações que são apresentadas em um mesmo gráfico, diferentemente dos gráficos de barra. Os gráficos de *candlestick* mostram a evolução dos preços no período, pontos de volatilidade (alongamento das velas), máximas e mínimas do período, preço inicial e final. Na Figura 11 pode-se observar um exemplo de gráfico de *candlestick*.



Figura 11 - Gráfico de *candlestick* de IBOV de 14/06/2017 a 06/09/2017.

Fonte: BM&FBOVESPA (2017).

Para a interpretação deste tipo de gráfico é importante entender a leitura de cada *candle*. Primeiramente a cor da vela representa se o movimento dos preços foi de alta ou de baixa. Quando o preço de fechamento é maior que o de abertura, este é representado por um *candle* branco ou verde, que indica um movimento de alta. Por outro lado, quando o preço de abertura é maior que o de fechamento, este é representado neste caso por um *candle* preto ou vermelho, que indica um movimento de baixa. O pavio na parte superior da figura indica o preço máximo que o ativo atingiu no período e o na parte inferior o preço mínimo. A Figura 12 ilustra os elementos dos *candles*.

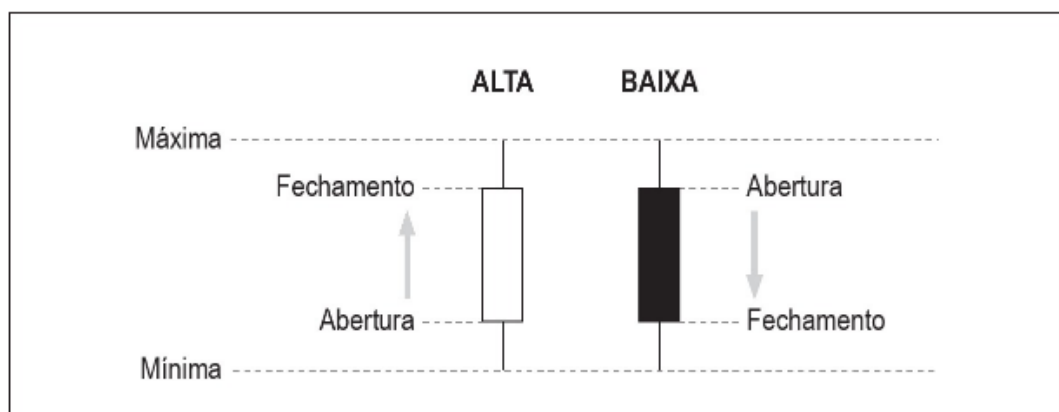


Figura 12 - Formação dos *candles*.

Fonte: Candlestick, um método para ampliar lucros na bolsa de valores (2007).

No lado esquerdo da Figura 12 observamos um *candle* branco de alta no qual o preço de fechamento foi maior que o de abertura. Já ao lado direito, observamos um *candle* preto de baixa onde o valor de fechamento foi menor que o de abertura.

Além das informações obtidas pela leitura de cada *candle* no gráfico, esta ferramenta fornece diversos padrões de *candles* que aparecerem com certa frequência nos gráficos e que representam uma reação do mercado à uma tendência estabelecida. Por isso, estes padrões de *candles* podem alertar o investidor de possíveis reversões no mercado (DEBASTIANI, 2016). Alguns desses padrões são *doji*, martelo, marobozu e bebê abandonado, que serão apresentados a seguir.

2.4.1.1 Doji

O Doji é um padrão representado por um *candle*, no qual o preço de abertura e de fechamento são idênticos, ele pode ter pavios ou não. O Doji representa um momento de incerteza no mercado (DEBASTIANI, 2016).

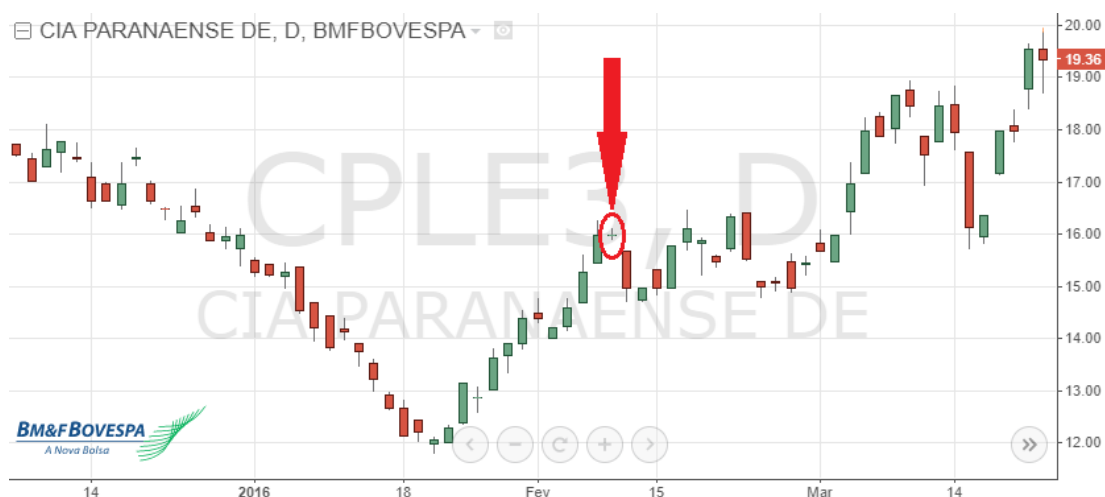


Figura 13 - Exemplo de Doji em 10/02/2016 em CPLE3.

Fonte: Adaptado de BM&FBOVESPA (2017).

A Figura 13 mostra um exemplo de Doji encontrado em 10/02/2016 no gráfico da Companhia Paranaense de Energia Elétrica.

2.4.1.2 Martelo

O Martelo é uma figura formada por apenas um *candle* de corpo pequeno de qualquer cor e com um longo pavio que deve ter ao menos duas vezes o tamanho do corpo. O Martelo deve ser encontrado ao final de uma tendência de baixa ou após uma queda nos preços. Ele representa uma tendência de reversão, mostrando que logo os preços devem começar a subir novamente (DEBASTIANI, 2016).

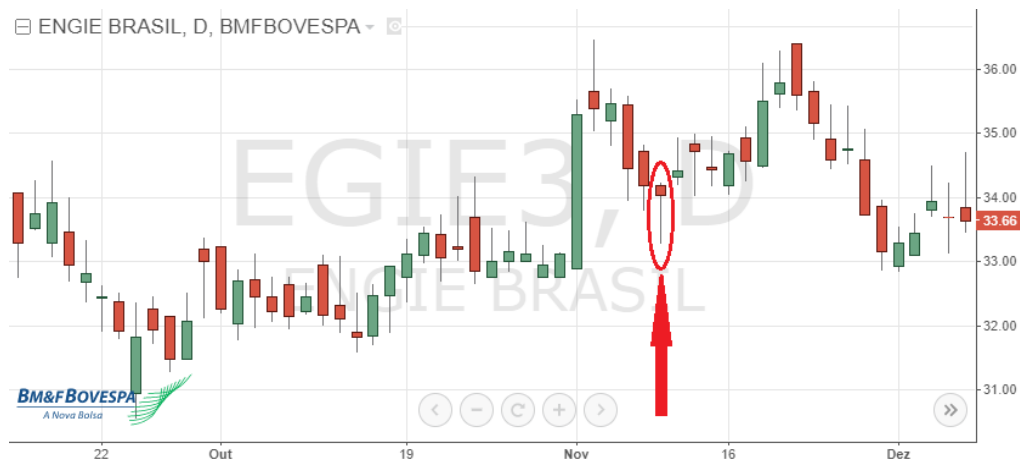


Figura 14 - Exemplo de Martelo em 10/11/2015 em ENGIE.

Fonte: Adaptado de BM&FBOVESPA (2017).

A Figura 14 mostra um exemplo de Martelo em 10/11/2015 no gráfico da ENGIE.

2.4.1.3 Marobozu

O Marobozu é um padrão representado por um *candle* sem pavios ou praticamente sem pavios e com um corpo muito longo se comparado aos demais *candles*. Ele representa uma tendência de movimento de alta ou de baixa, dependendo da sua cor (VIEIRA, 2016).

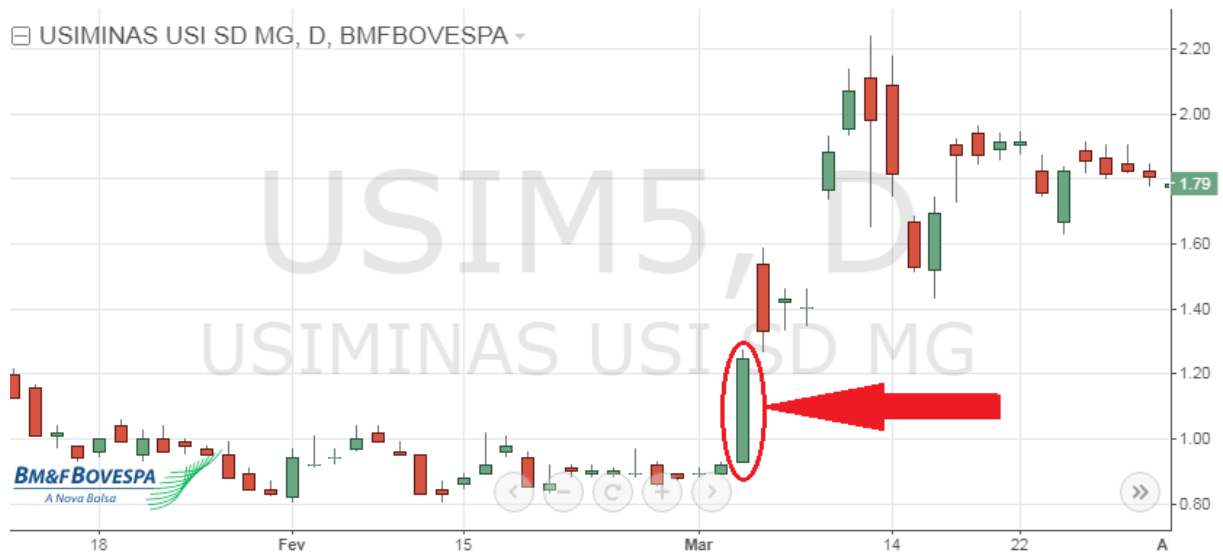


Figura 15 - Exemplo de Marobozu em 03/03/2016 em USIM5.

Fonte: Adaptado de BM&FBOVESPA (2017).

A Figura 15 mostra um exemplo de Marobozu de alta, encontrado na ação da Usiminas em 03/03/2016.

2.4.1.4 Bebê Abandonado

O bebê abandonado é um forte padrão de reversão, podendo ser de fundo ou de alta. Ele é representado por três *candles* a segunda figura deve ser um Doji e deve apresentar um *gap* entre o seu peço máximo ou mínimo e as outras duas figuras. Este padrão representa o esgotamento de tendência (VIEIRA, 2016).

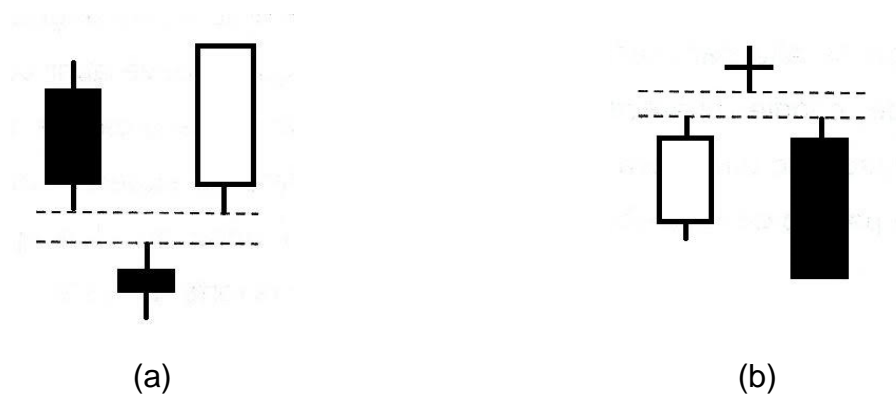


Figura 16 - Exemplo de Bebê Abandonado (a) de fundo e (b) de alta.

Fonte: Adaptado de Vieira (2016).

A Figura 16 mostra dois exemplos de bebê abandonado, um de fundo e outro de alta.

2.4.1.5 Bandas de Bollinger

As bandas de Bollinger foram introduzidas na década de 1980 por John Bollinger, elas são uma ferramenta para análise técnica de ações. As bandas de Bollinger são representadas no gráfico por duas linhas, uma acima dos preços do papel (banda superior) e outra abaixo do preço do papel (banda inferior). Elas são calculadas a partir da média móvel simples e do desvio padrão dos preços em um dado período. Quanto mais abertas as bandas estiverem, maior é a volatilidade dos preços. Já quando as bandas estão mais fechadas, menor será a tendência do movimento (VIEIRA, 2016).

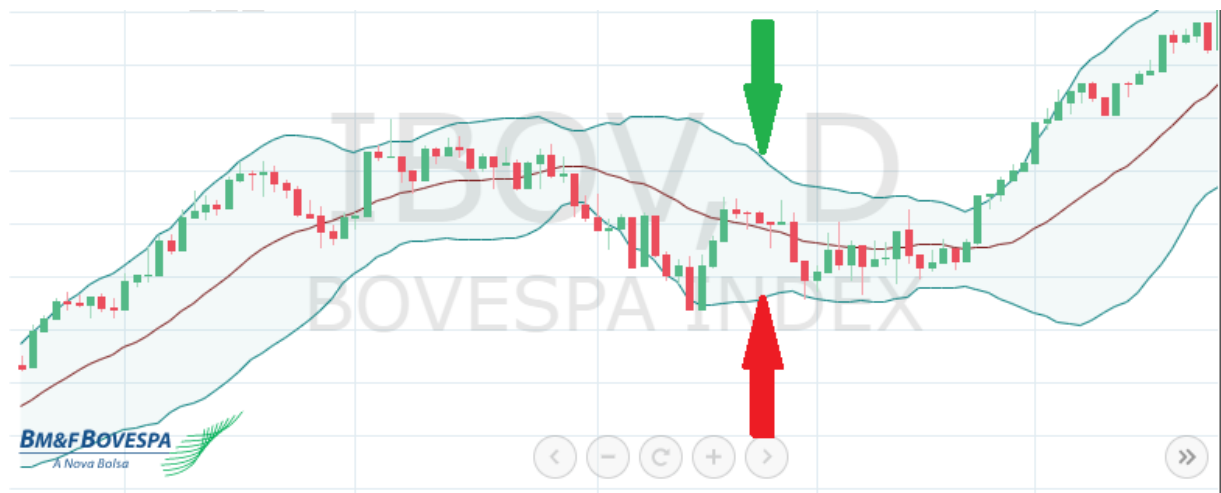


Figura 17 - Exemplo de Bandas de Bollinger.

Fonte: Adaptado de BM&FBOVESPA (2017).

A Figura 17 mostra o exemplo de um gráfico do índice Ibovespa no qual as setas indicam o desenho das linhas em azul que representam as bandas de bollinger. A seta em verde indica a banda de bollinger superior e a seta vermelha indica a banda de bollinger inferior.

3 METODOLOGIA

Para iniciar o desenvolvimento do trabalho foi necessário definir as etapas de construção e definição do trabalho. Os seguintes passos foram definidos para o trabalho:

- definir os requisitos funcionais e não funcionais do aplicativo;
- escolher quais serão os sistemas móveis para os quais a aplicação será desenvolvida;
- definir qual *web service* será utilizado;
- desenvolver uma funcionalidade que faça uma busca e atualização local das cotações dos principais papéis da BM&FBOVESPA;
- estudar e definir quais técnicas de análise de mercado o aplicativo deverá implementar;
- implementar as técnicas de análise definidas;
- criar uma interface simples;
- interagir com os usuários para verificar a usabilidade da ferramenta.

4 DESENVOLVIMENTO

O trabalho foi desenvolvido essencialmente em duas etapas, a primeira foi a etapa de levantamento de requisitos feita em uma reunião inicial em conjunto com os interessados no sistema, seguida da modelagem do aplicativo. A segunda, fase por sua vez, se constituiu pela escolha das tecnologias e pela construção do aplicativo. Estas etapas são detalhadas a seguir.

4.1 LEVANTAMENTO DE REQUISITOS

O processo de levantamento de requisitos foi realizado em conjunto com o grupo de oito investidores interessados no aplicativo, visando descrever todas as funcionalidades indispensáveis para a correta modelagem do sistema a fim de o aplicativo atender todas as necessidades dos especialistas no mercado financeiro.

4.1.1 Requisitos Funcionais

Os requisitos funcionais do aplicativo são:

- **Cotações:** Apresentar a lista das cem ações mais negociadas na BM&FBOVESPA e suas cotações no momento da consulta obtida através de consulta a um *web service*.
- **Favoritos:** O aplicativo deve ter um método que permita cadastrar determinadas ações como favoritas a fim de facilitar a consulta de informações das mesmas.
- **Detalhes:** O aplicativo deve permitir ao usuário consultar mais detalhes e informações de um determinado papel.
- **Ponto de compra:** O aplicativo deve implementar os métodos para o cálculo da análise de um ponto de compra através dos critérios pré-definidos pelos clientes a fim de alertar o usuário caso identifique algum ponto de compra.

4.1.2 Requisitos não funcionais

Os requisitos não funcionais do aplicativo são:

- **Sistema Operacional:** O aplicativo deve funcionar nos smartphones dos investidores.
- **Rapidez:** O sistema deve apresentar as informações ao usuário da forma mais rápida possível para tornar mais ágil as tomadas de decisão do usuário.
- **Interface:** O aplicativo deve possuir uma interface simples apresentando somente as informações necessárias para a tomada de decisão.
- **Dados:** O aplicativo deve armazenar os dados em um banco de dados local, evitando chamadas repetidas aos *web services*.
- **Conectividade:** O aplicativo deve funcionar tanto através de redes *Wi-Fi* quanto por redes móveis.

4.2 MODELAGEM

A modelagem do aplicativo foi feita utilizando a linguagem UML e é apresentada nas próximas seções. A criação dos diagramas foi feita utilizando o software Astah Community.

4.2.1 Diagrama de Casos de Uso

O diagrama de casos de uso tem a função de apresentar os requisitos funcionais levantados e representá-los graficamente pelos casos de uso, mostrando as interações dos atores com o sistema. A figura 18 apresenta o diagrama de casos de uso da aplicação.

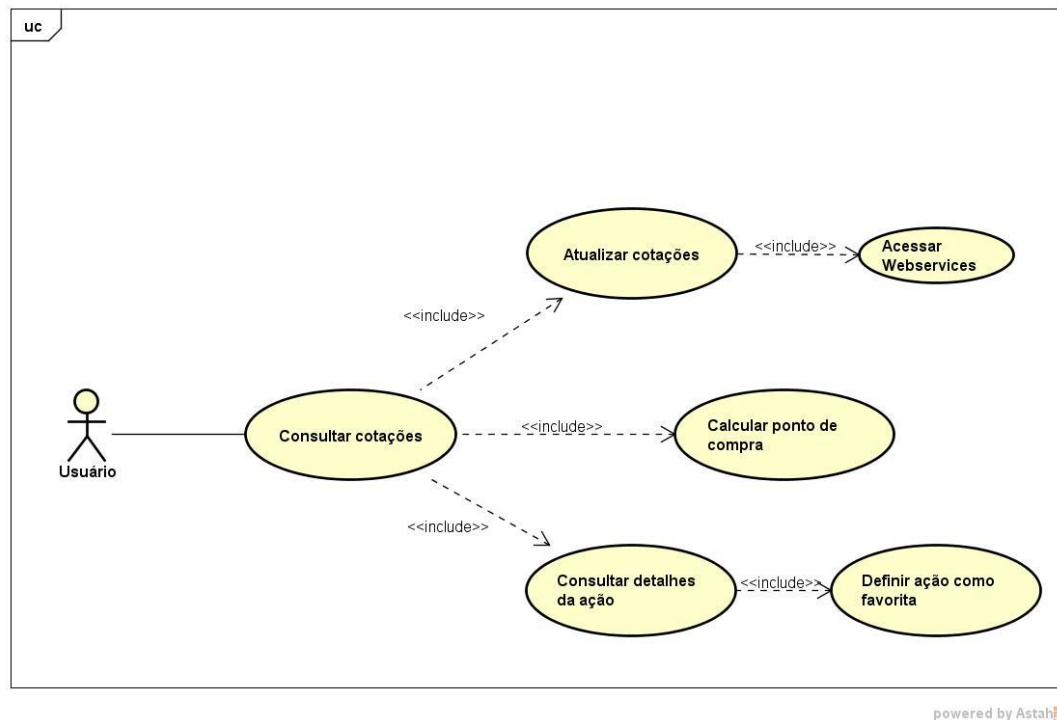


Figura 18 - Diagrama de Casos de Uso.

Fonte: Autoria Própria.

No diagrama da Figura 18 podemos observar os casos de uso do aplicativo e as funcionalidades que ele oferece ao usuário.

4.2.1.1 Consultar cotações

Este caso de uso tem como finalidade apresentar ao usuário a lista de todas as cotações.

- Atores: usuário.
- Pré-condições: iniciar o aplicativo.
- Pós-condições: a lista é apresentada com as últimas cotações disponíveis.
- Fluxo principal:
 1. Este caso de uso é iniciado quando o usuário abre o aplicativo.
 2. A tela é atualizada com as últimas cotações disponíveis para cada ação da lista.
 3. O caso de uso é encerrado.

4.2.1.2 Atualizar cotações

Este caso de uso tem como finalidade atualizar os preços das cotações da lista de ações.

- Atores: usuário.
- Pré-condições: caso de uso "consultar cotações" e o usuário deve ter clicado no botão de atualizar. O smartphone precisa de conexão com a rede.
- Pós-condições: os preços das ações são atualizados, a base de dados é atualizada e os novos pontos de compra são calculados.
- Fluxo principal:
 4. Este caso de uso é iniciado quando o usuário clica sobre o botão atualizar (A1).
 5. O aplicativo informa ao usuário que está buscando as cotações através de uma mensagem (M1).
 6. O caso de uso acessar *web services* é chamado e o aplicativo solicita as informações ao *web service*.
 7. O aplicativo informa ao usuário que concluiu a atualização (M2).
 8. O caso de uso calcular ponto de compra é iniciado.
 9. O caso de uso é encerrado.
- Fluxo alternativo: A1.
 1. O aplicativo informa ao usuário que não há conexão com a rede (M3).
 2. O caso de uso é encerrado.
- Mensagens:
 - M1. "Atualizando".
 - M2. "Concluído".
 - M3. "Sem conexão de rede".

4.2.1.3 Calcular ponto de compra

Este caso de uso tem como finalidade atualizar as sugestões de investimento para o usuário.

- Atores: usuário.
- Pré-condições: caso de uso "atualizar cotações".
- Pós-condições: os novos pontos de compra são calculados e a base de dados é atualizada.
- Fluxo principal:
 1. Este caso de uso é iniciado quando o aplicativo termina o caso de uso "atualizar cotações".
 2. O aplicativo calcula os novos pontos de compra para as novas cotações.
 3. O caso de uso é encerrado.

4.2.1.4 Consultar detalhes da ação

Este caso de uso tem como finalidade apresentar ao usuário mais detalhes da ação selecionada.

- Atores: usuário.
- Pré-condições: caso de uso "consultar cotações" e o usuário deve ter clicado sobre a ação que deseja verificar mais detalhes.
- Pós-condições: a tela é atualizada para apresentar as demais informações sobre a ação.
- Fluxo principal:
 1. Este caso de uso é iniciado quando o usuário clica sobre uma ação da lista.
 2. O aplicativo atualiza a tela e apresenta ao usuário mais informações sobre a ação selecionada.
 3. O caso de uso é encerrado.
- Fluxo alternativo: A1.

1. O caso de uso definir ação como favorita é iniciado.
2. O caso de uso é encerrado.

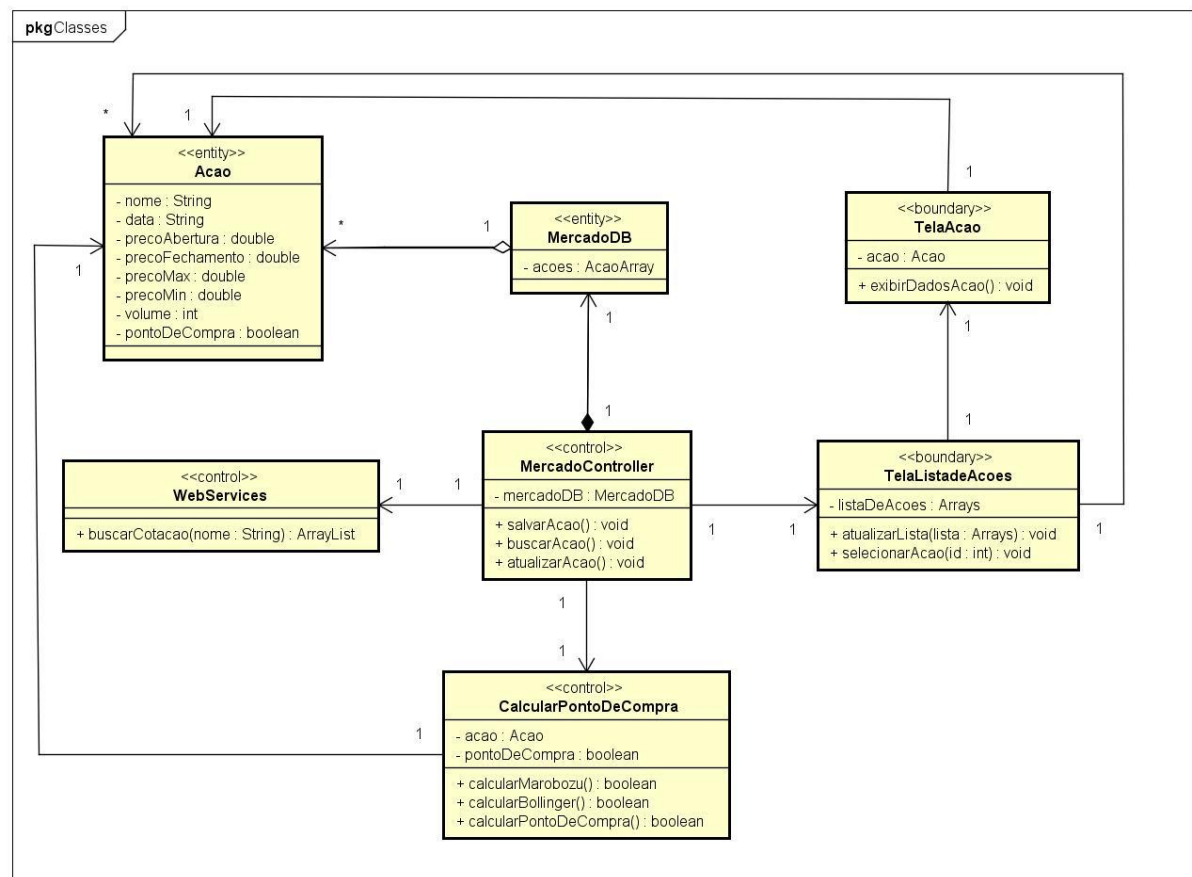
4.2.1.5 Definir ação como favorita

Este caso de uso tem como finalidade definir uma ação como favorita e apresentá-la no início da lista de ações.

- Atores: usuário.
- Pré-condições: caso de uso "consultar detalhes da ação" e o usuário deve ter clicado no botão de definir ação como favorita.
- Pós-condições: a ação é identificada como favorita para ser apresentada antes das demais ações não favoritas.
- Fluxo principal:
 1. Este caso de uso é iniciado quando o usuário clica sobre o botão de definir ação como favorita.
 2. O aplicativo identifica a ação como favorita.
 3. O caso de uso é encerrado.

4.2.2 Diagrama de Classes

O diagrama de classes do aplicativo pode ser observado na Figura 19. Nele está representada graficamente a estrutura das classes do sistema projetada para atender os requisitos funcionais do programa e os principais métodos e atributos do sistema.



powered by Astah

Figura 19 - Diagrama de Classes.

Fonte: Autoria Própria.

Pelo diagrama de classes apresentado na Figura 19 podemos observar que a classe principal do aplicativo é a *MercadoController*, esta classe controla as principais funcionalidades do aplicativo e chamadas as demais classes. A classe *TelaListaDeAções* de ações implementa os métodos de atualizar a lista, selecionar uma ação e definir uma ação como favorita. A classe *CalcularPontoDeCompra* implementa todos os métodos necessários para realizar os cálculos para a definição de um ponto de compra. A *MercadoDB* por sua vez controla a base de dados do aplicativo e a *Acao* define todas as características de uma ação. A *TelaAcao* apresenta ao usuário mais informações sobre a ação selecionada na tela de lista de ações.

Por fim, a classe *WebServices* faz a chamada de busca de ações para baixar os dados atualizados do servidor.

4.2.3 Diagrama de Atividades

O fluxo de controle das atividades do aplicativo pode ser observado na Figura 20 e na Figura 21.

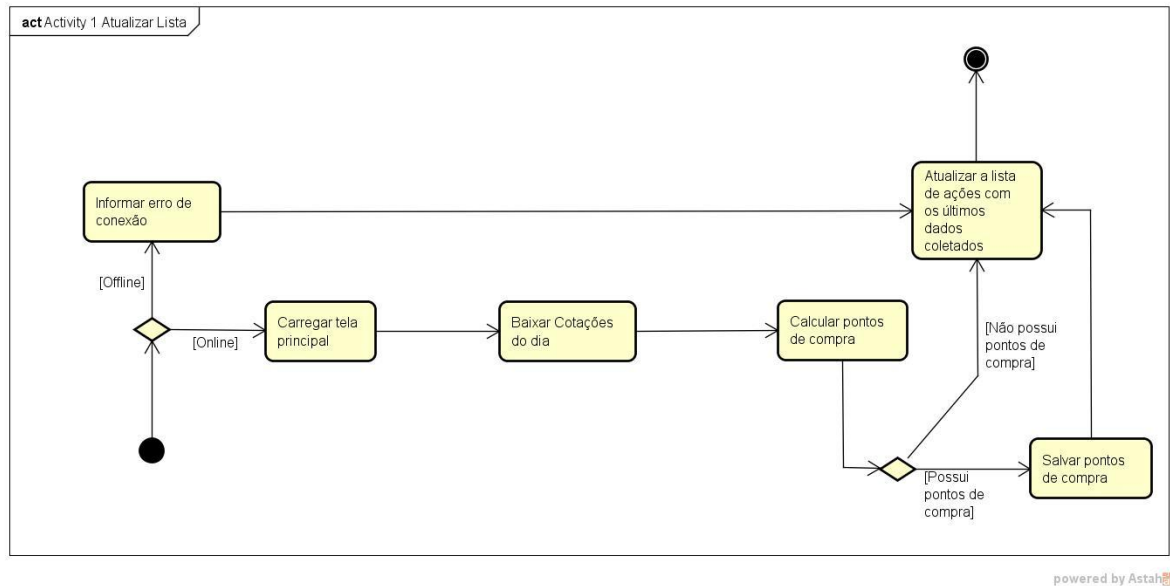
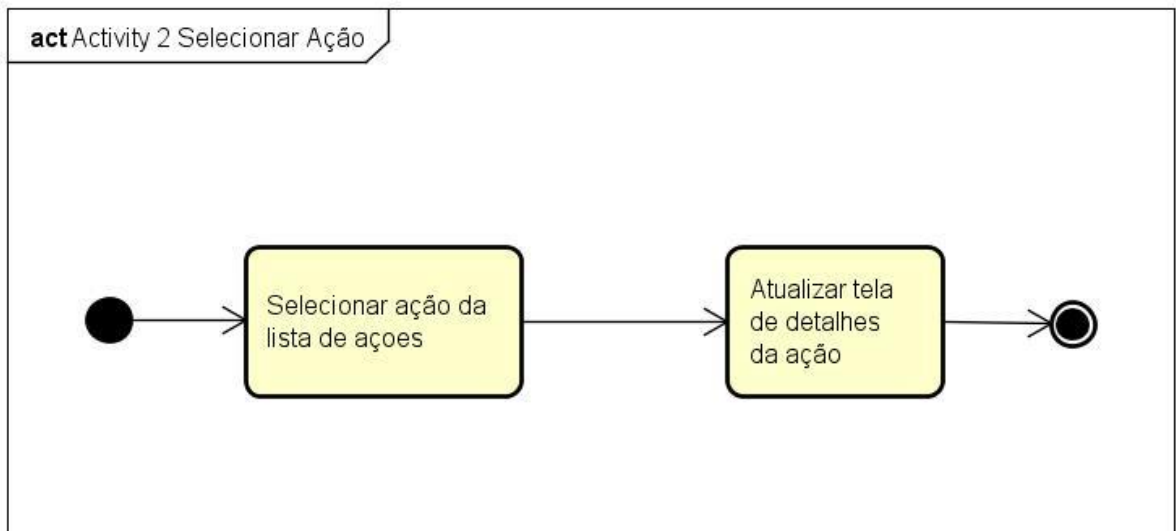


Figura 20 - Diagrama de Atividades - Atualizar Lista.

Fonte: Autoria Própria.

O diagrama de atividades da Figura 20 apresenta como a lista é atualizada. Quando o usuário clica sobre o botão de atualizar, o sistema verifica se existe ou não conexão de rede e informa o usuário. Caso não exista conexão a lista de ações é atualizada com os últimos dados disponíveis e a atividade é encerrada. Caso exista conexão, o aplicativo verifica se a base de dados está atualizada, caso não esteja ele baixa um histórico de cotações e caso a base de dados esteja atualizada ele apenas baixa as cotações atuais. Em seguida, o aplicativo salva os dados e calcula os pontos de compra para as novas cotações. Após isto, a lista é atualizada com as novas informações e a atividade é encerrada.



powered by Astah

Figura 21 - Diagrama de Atividades - Selecionar Ação.

Fonte: Autoria Própria.

A Figura 21 apresenta o diagrama de atividade de selecionar uma ação. Ela inicia quando o usuário seleciona uma ação da lista de ações, em seguida a tela do aplicativo é atualizada com os detalhes da ação selecionada e a atividade é encerrada.

4.2.1 Diagrama de Sequências

O diagrama de sequências que apresenta a troca de mensagens entre os objetos e a sequência dos processos do aplicativo está apresentada na figura 22.

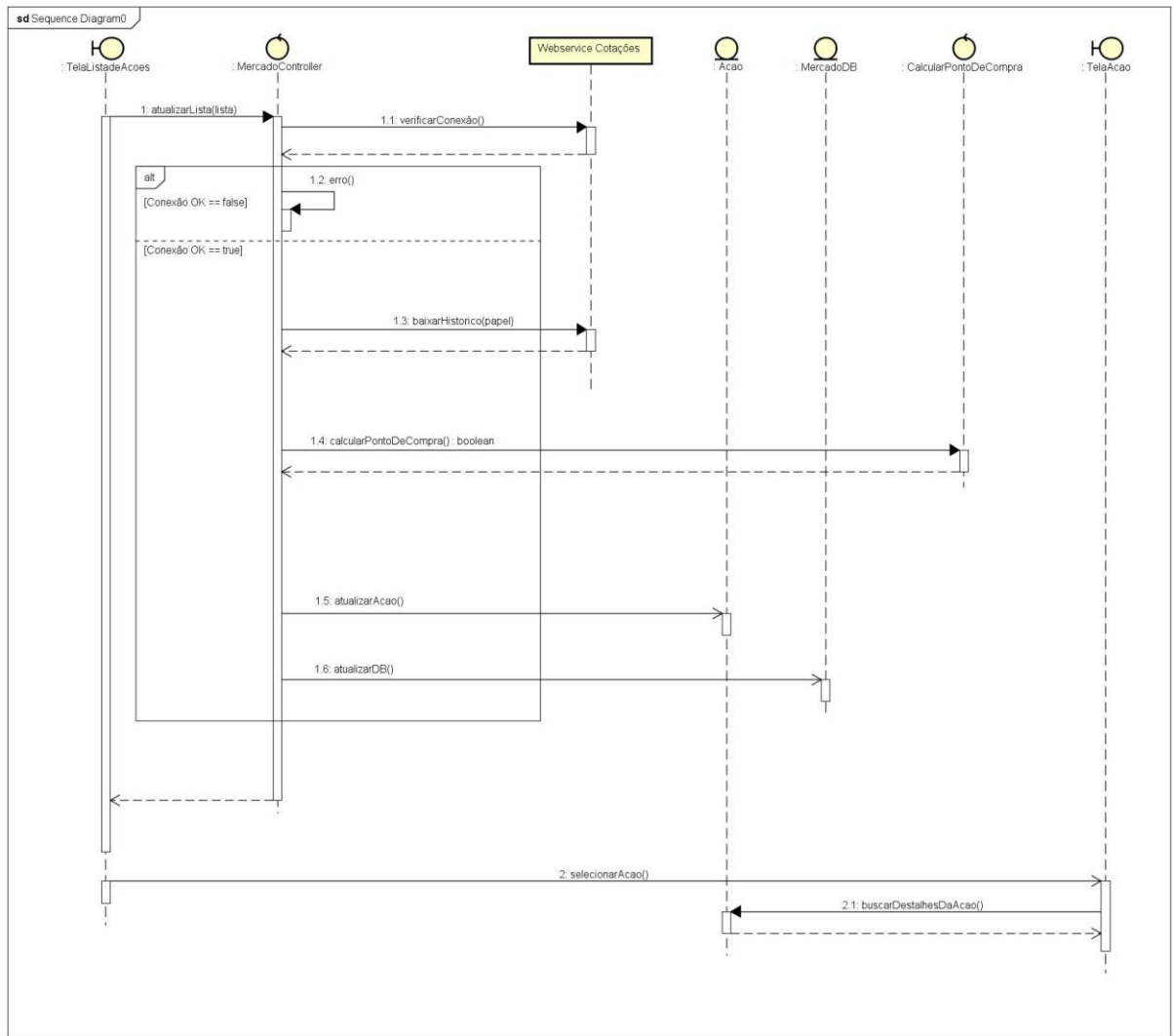


Figura 22 - Diagrama de Sequências.

Fonte: Autoria Própria.

Na figura 22 é possível observar a sequência das requisições das mensagens em dois momentos. No primeiro quando, a partir da tela inicial com a lista de ações, o usuário clica sobre o botão de atualização. O segundo momento retrata o fluxo do aplicativo quando o usuário clica sobre uma ação da lista e a tela com os detalhes da ação é exibida.

4.3 CONSTRUÇÃO DO APLICATIVO

Nesta seção serão apresentados os métodos utilizados para o desenvolvimento do aplicativo. Também serão mostrados o aplicativo e suas funcionalidades.

Para iniciar o desenvolvimento do aplicativo foi necessário definir para quais sistemas operacionais o aplicativo seria desenvolvido, qual seria o método para o cálculo do ponto de compra de um ativo e pesquisar qual seria o melhor *web service* para realizar a coleta das cotações. Como todos os integrantes do grupo de investidores possuíam smartphones Android, optou-se pela aplicação atender apenas este SO. As definições do cálculo para o ponto de compra e do *web service* são descritas a seguir.

4.3.1 Cálculo do ponto de compra

A fim de aproximar os especialistas na área financeira do desenvolvimento do aplicativo, a definição do cálculo do ponto de compra foi feita em conjunto. Desta forma, pode-se garantir que as funcionalidades do aplicativo seriam úteis no dia a dia dos investidores.

Os clientes solicitaram que fossem utilizados dois critérios para a definição do ponto de compra. O primeiro critério para a definição de um ponto de compra seria o cálculo do marobozu, no momento da atualização das cotações. O segundo critério seria o preço atual estar acima da banda de Bollinger superior utilizando uma média de 8 períodos. As fórmulas para o cálculo podem ser observadas na equação 1 e 2 (onde σ representa o desvio padrão do conjunto de dados).

$$Preco_{atual} - Abertura_{atual} \geq (Max_{anterior} - Min_{anterior}) \quad (1)$$

$$Preco_{atual} \geq \frac{\sum_0^7 Preco_{fechamento}}{7} + 2\sigma \quad (2)$$

4.3.2 Busca de cotações

Devido ao valor econômico das informações de cotações das ações da bolsa, a maioria dos *web services* disponíveis são fechados e pagos. No primeiro momento não achou-se conveniente a utilização de um serviço pago. Durante a pesquisa, foram encontradas três opções sem custo e amplamente utilizadas, o *web service* do Yahoo Finance, o do Google Finance e o do Alpha Vantage. Porém, durante o

desenvolvimento do trabalho os serviços fornecidos pelo Yahoo e pelo Google foram descontinuados. Portanto, para realizar a busca de cotações do aplicativo foi utilizado o *web service* de cotações fornecido pelo Alpha Vantage.

O serviço oferece cotações *intraday*, diárias, semanais e mensais nos formatos JSON e CSV através de uma chamada REST. Para o aplicativo foi utilizada a chamada para cotações diárias de cada ativo para retorno em formato CSV. A figura 23 exibe um exemplo do resultado da chamada https://www.alphavantage.co/query?function=TIME_SERIES_DAILY&symbol=MSFT&apikey=demo&datatype=csv.

```
timestamp,open,high,low,close,volume
2017-12-08,83.6300,84.5800,83.3300,84.1600,23825056
2017-12-07,82.5400,82.8000,82.0000,82.4900,20378114
2017-12-06,81.5500,83.1400,81.4300,82.7800,24821403
2017-12-05,81.3400,82.6800,80.9801,81.5900,25512120
2017-12-04,84.4200,84.4299,80.7000,81.0800,37977732
2017-12-01,83.6000,84.8100,83.2200,84.2600,29387325
2017-11-30,83.5100,84.5200,83.3400,84.1700,32998408
2017-11-29,84.7100,84.9172,83.1750,83.3400,27358477
2017-11-28,84.0700,85.0600,84.0200,84.8800,21162639
2017-11-27,83.3100,83.9800,83.3000,83.8700,17603760
2017-11-24,83.0100,83.4300,82.7800,83.2600,7425503
2017-11-22,83.8300,83.9000,83.0400,83.1100,20213704
2017-11-21,82.7400,83.8400,82.7400,83.7200,21237454
2017-11-20,82.4000,82.5900,82.2500,82.5300,16072495
2017-11-17,83.1200,83.1200,82.2400,82.4000,21715498
2017-11-16,83.1000,83.4200,82.9400,83.2000,20659209
2017-11-15,83.4700,83.6900,82.6900,82.9800,19097333
2017-11-14,83.5000,84.1000,82.9800,84.0500,18604034
2017-11-13,83.6600,83.9400,83.4600,83.9300,14080820
2017-11-10,83.7900,84.1000,83.2300,83.8700,19340435
2017-11-09,84.1100,84.2700,82.9000,84.0900,20924172
2017-11-08,84.1400,84.6100,83.8300,84.5600,18034170
2017-11-07,84.7700,84.9000,83.9300,84.2700,17152583
```

Figura 23 – Resultado da chamada ao *web service*.

Fonte: Autoria Própria.

Na figura 23 podemos observar que o servidor fornece informações diárias dos preços de abertura, máxima, mínima e fechamento de uma ação, bem como o volume de papéis negociados no dia. O aplicativo foi desenvolvido para funcionar como uma

aplicação cliente do servidor Alpha Vantage, buscando as cotações diárias no servidor e realizando localmente o cálculo dos pontos de compra.

4.3.3 Funcionamento do aplicativo

Assim que o aplicativo é executado é apresentada a tela inicial com a lista das ações e suas cotações baixadas na última atualização realizada, as ações favoritas e os pontos de compra calculados na última atualização. Além disso, a tela inicial fornece um botão para a atualização dos papéis. As figuras 24 e 25 mostram a tela inicial do aplicativo.



Figura 24 – Tela inicial do aplicativo com a lista de ações e botão para atualizar.

Fonte: Autoria Própria.

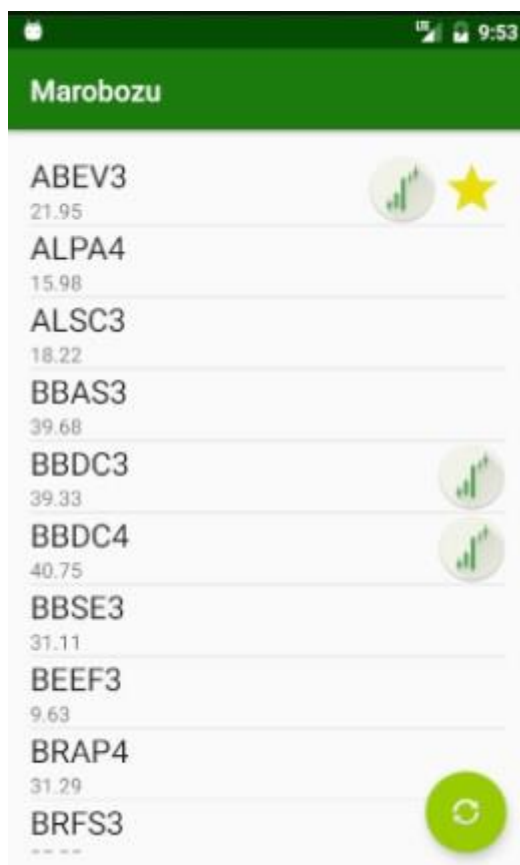


Figura 25 – Tela inicial do aplicativo com exemplo de pontos de compra e ação favorita.

Fonte: Autoria Própria.

A estrela ao lado de uma ação, que pode ser observada na Figura 25, é o símbolo utilizado pelo aplicativo para mostrar que ela foi definida como uma ação favorita. Por sua vez o símbolo redondo com *candles* verdes em seu interior apresentado ao lado de algumas ações na Figura 25 é o ícone que o aplicativo utiliza para mostrar ao usuário que um ponto de compra foi identificado para o papel nesta atualização.

Ao clicar no botão de atualização a aplicação irá verificar se o smartphone do usuário tem conexão com a internet. Caso não tenha conexão com a rede, é exibida uma mensagem de alerta ao usuário “Sem conexão com a rede”. E caso exista conexão, o aplicativo irá realizar a chamada REST para o servidor externo Alpha Vantage e exibirá a mensagem “Atualizando”. Após baixar as novas informações de cotação, o aplicativo salva os dados no banco de dados local. Em seguida, o aplicativo calcula se existe algum ponto de compra para as novas cotações, em caso afirmativo a informação é atualizada no banco de dados.

Quando o usuário clica sobre uma ação apresentada na lista da tela inicial é apresentada a tela de detalhes desta ação, que pode ser vista na figura 26. Nesta tela é possível observar mais informações sobre a ação, tais como preço atual, máxima, mínima, preço de abertura, volume e fechamento do dia da última atualização e do dia anterior. Nesta tela também é exibido um botão para o usuário adicionar a ação como favorita ou remove-la dos favoritos.



2018-01-31		2018-01-30	
Preço:	49.41	Fechamento:	50.49
Abertura:	50.69	Abertura:	50.57
Max:	50.86	Max:	50.69
Min:	49.41	Min:	49.13
Vol:	958100.0	Vol:	1179300.0

Figura 26 – Tela de detalhes da ação e botão favorita.

Fonte: Autoria Própria.

4.4 TESTES

Após o desenvolvimento do aplicativo, a fim de testar suas funcionalidades, ele foi distribuído para os oito investidores que participaram do levantamento de requisitos. Foi explicado como o aplicativo mostrava um ponto de compra e foi solicitado que os

usuários utilizassem o aplicativo durante 1 mês para preencherem o seguinte questionário:

1. Como você avalia a facilidade de usar o aplicativo? (De uma nota de 0 a 10).
2. De um modo geral, como você avalia a interface do aplicativo? (De uma nota de 0 a 10).
3. O aplicativo melhorou a busca por oportunidades de compra? (Sim ou não).
4. O número de operações realizadas foi maior com o auxílio do aplicativo durante o mês? (Sim ou não).
5. O que poderia ser melhorado no aplicativo? (Pergunta aberta).

5. RESULTADOS OBTIDOS

Com a participação dos investidores interessados na ferramenta, foi possível criar um aplicativo que atendesse as necessidades reais do grupo. Era necessário criar um aplicativo simples e enxuto para uma rápida visualização que auxiliasse na tomada de decisão para um investimento. Após testarem o aplicativo durante um mês os usuários responderam o questionário de avaliação do aplicativo e as respostas podem ser observadas no Quadro 1.

	Pergunta 1	Pergunta 2	Pergunta 3	Pergunta 4	Pergunta 5
Usuário 1	9	8	Sim	Sim	O aplicativo poderia ter notificações.
Usuário 2	7	7	Sim	Sim	
Usuário 3	8	6	Sim	Sim	
Usuário 4	10	8	Sim	Sim	Notificações para os pontos de compra.
Usuário 5	8	7	Sim	Sim	
Usuário 6	8	8	Sim	Sim	Alertas de pontos de compra.
Usuário 7	7	6	Sim	Sim	Buscas automáticas com notificações.
Usuário 8	9	8	Sim	Sim	Notificações.

Quadro 1 – Respostas do questionário de avaliação

Fonte: Autor.

Ao observar as respostas dos clientes no Quadro 1, pode se verificar que todos relataram um aumento no número de operações realizadas. A automação do cálculo permitiu que eles conseguissem analisar todas as opções de investimento e tomar uma decisão ainda com o mercado aberto para operações.

6. CONCLUSÃO

O uso de dispositivos móveis já faz parte do cotidiano das pessoas, auxiliando nas mais diversas atividades diárias. Estes dispositivos proporcionam maior comodidade e liberdade de locomoção durante o uso, o desenvolvimento de novas ferramentas para smartphones é de grande interesse para os usuários e para fabricantes. Já existem alguns aplicativos destinados a investidores que fornecem grande quantidade de informação para auxiliar a tomada de decisão, porém ainda é escassa oferta de aplicativos para dispositivos móveis que automatizem a análise gráfica.

Neste sentido, no trabalho foi apresentado o desenvolvimento de um aplicativo para dispositivos móveis com o intuito de auxiliar os investidores do mercado financeiro na tomada de decisão por meio da automação de uma análise técnica das ações. O trabalho e a construção do aplicativo proporcionaram o aprendizado e a prática dos conceitos e técnicas de programação para dispositivos móveis vistos durante o curso. Com o trabalho foi possível consolidar e aperfeiçoar os conceitos de *web services*, bancos de dados e UML do autor deste trabalho.

O maior desafio durante o desenvolvimento do aplicativo foi encontrar um serviço confiável para a busca das cotações de forma gratuita. Inicialmente foi implementada a versão do Yahoo Finance, a qual foi descontinuada. Na sequência, optou-se por implementar a versão do Google Finance, que também foi descontinuado meses mais tarde. Neste sentido, foi necessária a implementação de uma terceira versão, a Alpha Vantage, que foi a definitiva para a versão do aplicativo.

A eficácia do aplicativo pode ser comprovada pelos investidores que participaram dos testes. Eles relataram um aumento no número de operações realizadas após o uso da ferramenta, devido ao aplicativo os permitir realizar a análise diariamente, com maior rapidez e sem a necessidade de um computador.

Além disso, a eficácia do aplicativo pôde ser comprovada pelos investidores que participaram dos testes. Eles relataram um aumento no número de operações realizadas após o uso da ferramenta, uma vez que o aplicativo permitiu realizar a análise diariamente, com maior rapidez e sem a necessidade de um computador.

Como trabalhos futuros, poderia ser criado um *widget* para facilitar a verificação das cotações e dos pontos de compra. Também poderia ser implementada a atualização automática das cotações e alerta dos pontos de compra por notificações.

Tais melhorias aumentariam a eficácia das tomadas de decisão dos investidores em um ambiente que necessita de precisão e agilidade para maximizar os ganhos com os investimentos.

REFERÊNCIAS BIBLIOGRÁFICAS

ALPHA VANTAGE. **Alpha Vantage API Documentation**. Disponível em: <https://www.alphavantage.co/documentation/>. Acesso em dez/2017.

ANDROID. **About Android**. Disponível em: <https://source.android.com/>. Acesso em jul/2017.

APPLE. **About the iOS Technologies**. Disponível em: https://developer.apple.com/library/content/documentation/Miscellaneous/Conceptual/iPhoneOSTechOverview/Introduction/Introduction.html#//apple_ref/doc/uid/TP40007898-CH1-SW1/. Acesso em jul/2017.

BMFBOVESPA. **Histórico pessoas físicas**. Disponível em: http://www.bmfbovespa.com.br/pt_br/servicos/market-data/consultas/historico-pessoas-fisicas/. Acesso em set/2017.

DEBASTIANI, Carlos. **Candlestick Um método para ampliar lucros na Bolsa de Valores**. São Paulo: Editora Novatec, 2016.

FOSSE, Gustavo; LEO, Sergio; SANCHEZ, Cleide; GRATÃO, Nilton. **Pesquisa FEBRABAN de Tecnologia Bancária**. São Paulo: Federação Brasileira de Bancos, 2015.

GUEDES, Gilleanes T. A. **UML Uma abordagem prática**. São Paulo: Editora Novatec, 2011.

IBGE. **Projeção da população do Brasil e das Unidades da Federação**. Disponível em: <https://ww2.ibge.gov.br/apps/populacao/projecao/>. Acesso em jan/2018.

IDC. **Smartphone OS Market Share**. Disponível em: <https://www.idc.com/promo/smartphone-market-share/os/>. Acesso em jun/2017.

IDC. **Worldwide Smartphone Forecast.** Disponível em: <https://www.idc.com/getdoc.jsp?containerId=US41084216/>. Acesso em jun/2017.

Lecheta, Ricardo R. **Web Services Restful.** São Paulo: Editora Novatec, 2015.

MAZIERO, Carlos. **Sistemas Operacionais: Conceitos e Mecanismos.** Curitiba: Universidade Federal do Paraná, 2017.

MEIRELLES, Fernando. **Pesquisa Anual do Uso de TI.** Rio de Janeiro: Fundação Getúlio Vargas, 2017.

MICROSOFT. **Windows Phone 8.1 for Developers.** Disponível em: <https://blogs.msdn.microsoft.com/thunbrynt/2014/03/31/windows-phone-8-1-for-developersconverging-the-app-models/>. Acesso em jul/2017.

MICROSOFT. **Windows Embedded Products.** Disponível em: <https://www.microsoft.com/windowseembedded/en-us/products-solutions-overview.aspx/>. Acesso em jul/2017.

OMG. **What is UML.** Disponível em: <http://www.uml.org/what-is-uml.htm>. Acesso em dez/2017.

OMG. **Unified Modeling Language.** Object Management Group Publication, 2017.

OPEN HANSET ALLIANCE. **Android Overview.** Disponível em: https://www.openhandsetalliance.com/android_overview.html/. Acesso em jul/2017.

PAHKALA, Jan. **Introduction to Windows Phone 8.** Oulu: Oulu University of Applied Sciences, 2012.

ROCHA, Adriano; NETO, Roberto. **Introdução à Arquitetura Apple iOS.** Catalão: Universidade Federal de Goiás ENACOMP, Encontro Nacional de Computação , 2011.

SILBERSCHATZ, Abraham. **Sistema de Banco de Dados**. Rio de Janeiro: Editora Campus, 2012.

SQLite. **AboutSQLite**. Disponível em :<https://www.sqlite.org>. Acesso em ago/2017.

VIEIRA, Rafael; GUIMARÃES, Vanderlei. **Golden Traders - Curso de Análise Técnica**. Curitiba: 2016.