

N. CLASS. M005.5
CUTTER C972p
ANO/EDIÇÃO 2015

CENTRO UNIVERSITÁRIO DO SUL DE MINAS - UNIS-MG
BACHARELADO EM SISTEMAS DE INFORMAÇÃO
RICHARD MIRANDA DA CUNHA

**PROPOSTA DE DESENVOLVIMENTO DE UM APLICATIVO MÓVEL PARA
GERENCIAMENTO DE LOTERIAS**

**Varginha/MG
2015**

FEPESMIG

RICHARD MIRANDA DA CUNHA

**PROPOSTA DE DESENVOLVIMENTO DE UM APLICATIVO MÓVEL PARA
GERENCIAMENTO DE LOTERIAS**

Monografia apresentada ao curso de graduação em Bacharelado em Sistemas de Informação do Centro universitário do Sul de Minas – UNIS/MG, como pré-requisito conclusão do curso, sob orientação do Prof. Rodrigo Franklin Frogeri.

**Varginha/MG
2015**

FEPESMIG

RICHARD MIRANDA DA CUNHA

**PROPOSTA DE DESENVOLVIMENTO DE UM APLICATIVO MÓVEL PARA
GERENCIAMENTO DE LOTERIAS**

Monografia apresentada ao curso de Bacharelado em Sistemas de Informação do Centro Universitário do Sul de Minas – UNIS/MG, como pré-requisito para obtenção do grau de bacharel pela Banca Examinadora composta pelos membros:

Aprovado em 09 / 11 / 2015

Prof. Msc. Rodrigo Franklin Frogeri

Prof. Esp. Rodrigo Gomes da Silva

AGRADECIMENTOS

Agradeço primeiramente a Deus, professores meus pais e minha mãe, meu irmão, minha noiva que esteve em todos os momentos ao meu lado e meus amigos por terem ajudado na construção deste trabalho.

RESUMO

Este projeto tem como objetivo desenvolver um aplicativo para gerenciamento de loterias, chamado RmSfLoterias, afim de facilitar a conferência de resultados para todos os apostadores que encontram dificuldades na conferência de suas apostas. O aplicativo foi desenvolvido através da IDE Android Studio. O FLUIDUI foi utilizado para o desenvolvimento do protótipo das telas do sistema, seguindo as melhores práticas de desenvolvimento e de projetos. O projeto desenvolvido contribuiu com a comunidade acadêmica através do estudo de caso de desenvolvimento de um sistema móvel, proporcionando não só conhecimento técnico, mas também facilitando o acesso aos dados das loterias no Brasil.

Palavras-chave: Android. Loterias. Aplicativo.

ABSTRACT

This project aims to develop an application for managing lotteries, called RmSfLoterias in order to facilitate results conference for all bettors who encounter difficulties in the conference bets. The application was developed by IDE Android Studio. The FLUIDUI was used for the development of the prototype with all screens that the system will display all the best practices in the development and projects. And thus contribute to academic development and make the college as a reference in the course of Information Systems. The project developed has contributed to the academic community through the development of case study of a mobile system, providing not only technical knowledge but also facilitating access to data from lotteries in Brazil.

Keywords: Android. Lotteries. App

SUMÁRIO

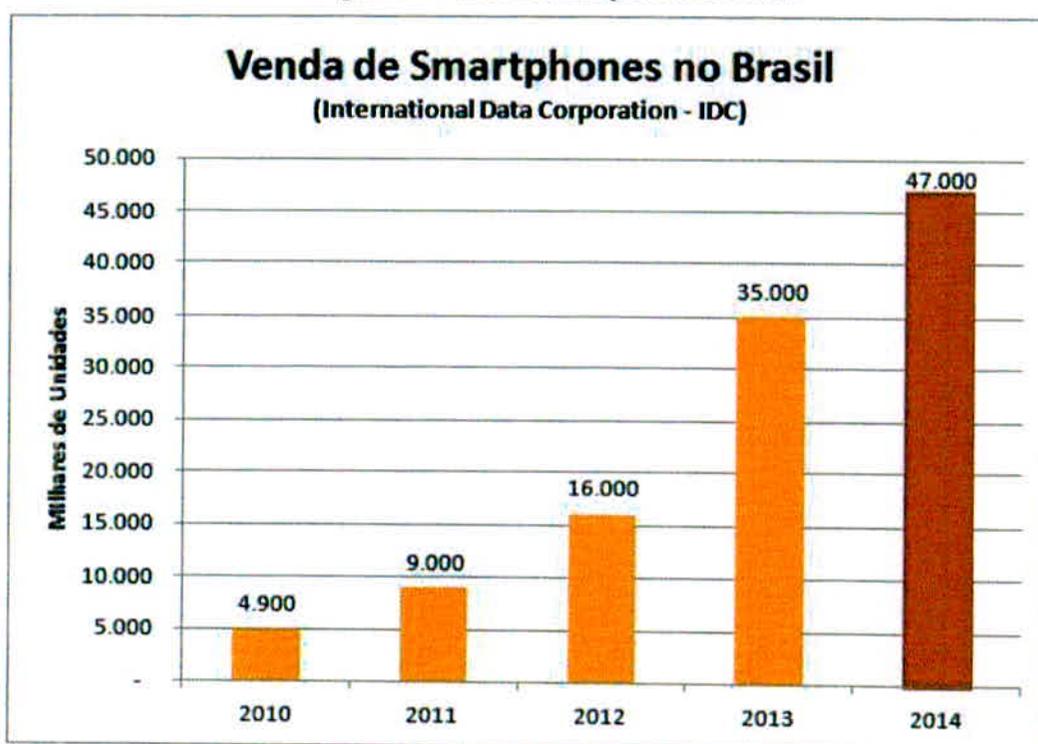
1 INTRODUÇÃO	7
1.1 Problematização.....	9
1.2 Hipóteses	9
1.3 Objetivo geral e Objetivos Específicos	9
1.3.1 Objetivo Geral	9
1.4 Justificativa.....	10
2 REVISÃO DA LITERATURA	11
2.1 Sistemas de Informação.....	11
2.2 Qualidades nos Sistemas de Informação	11
2.3 Levantamento de Requisitos.....	12
2.4 Diagramas de Caso de Uso.....	14
3 FERRAMENTAS TECNOLÓGICAS.....	15
3.1 Java.....	15
3.2 Eclipse	16
3.3 Android Studio	17
3.4 Bancos de Dados	17
4 METODOLOGIA.....	19
5 RESULTADOS	22
6 CONCLUSÃO.....	26
REFERÊNCIAS	27

1 INTRODUÇÃO

Com o rápido avanço da tecnologia surgem, na sociedade, várias necessidades, como, por exemplo, a necessidade de desenvolvimento de sistemas para dispositivos móveis com o fim de facilitar o cotidiano, ganhar tempo, obter mais lucros, tanto para pessoas como também para as empresas.

O mercado tecnológico especificamente para smartphones está mais promissor, tem-se uma quantidade enorme de aplicativos com diversas funcionalidades, gerando maior renda para todos os envolvidos nos processos de desenvolvimento de aplicativos. Abaixo vemos como a venda de Smartphones tem crescido nos últimos anos:

Figura 1 – Venda de Smartphones no Brasil



Fonte: TIPOSDIGITAIS. Disponível em: <<http://www.tiposdigitais.com/2014/04/brasil-n%C3%A3o-%C3%A9-pa%C3%ADs-do-ereader.html>>. Acesso em: 02. Nov. 2015.

Além da tecnologia, o que está tornando muito popular nos dias atuais são as loterias, que estão atraindo mais adeptos, com o intuito de fazer suas apostas e poder mudar de vida com prêmios milionários. Veja no quadro abaixo como tem crescido a arrecadação das loterias nos últimos anos.

Figura 2 – Crescimento Arrecadação das loterias

Ano	Arrecadação	Crescimento
2007	R\$ 5,2 bi	23,0%
2008	R\$ 5,8 bi	10,5%
2009	R\$ 7,3 bi	28,3%
2010	R\$ 8,8 bi	19,8%
2011	R\$ 9,7 bi	10,5%
2012	R\$ 10,4 bi	7,7%
2013	R\$ 11,4 bi	8,82%
2014	R\$ 13,5 bi	18,5%

Fonte: BNLDATA. Disponível em: <<http://www.bnldata.com.br/blogPost.aspx?cod=22954>>
Acesso em 02 de Nov. 2015

Um aplicativo que seja intuitivo, confiável que os ajude a conseguir os resultados das loterias, pode propiciar aos apostadores uma maior comodidade, facilidade na conferência e até mesmo incentivar as apostas.

1.1. Tema específico

Desenvolvimento de um aplicativo móvel para gerenciamento de loterias.

1.2. Problema de pesquisa

É possível desenvolver um aplicativo que forneça resultados confiáveis de loteria, fazendo a leitura do código de barras dos volantes das apostas e oferecendo informações personalizadas ao seu apostador?

1.1 Problematização

Um aplicativo para loterias precisa trazer os resultados de forma confiável, precisa buscar direto da fonte e trazer as informações a cada sorteio, mostrando alertas para que instigue o apostador para que ele se sinta atraído a navegar pelo aplicativo, para que o aplicativo seja considerado o melhor em fornecer esses resultados, com facilidade e comodidade.

1.2 Hipóteses

Um aplicativo que mostre apenas o resultado e nada mais sem nenhum diferencial, torna-se obsoleto, logo não atrai mais usuários. Um aplicativo para apostas ideal, seria aquele que lê o código de barras do volante e mostre para o usuário se o jogo está premiado ou não.

A utilização da linguagem de programação Java possibilitará que o sistema seja multiplataforma, ou seja, possa ser executado em diferentes tipos de sistemas, porém o sistema foi desenvolvido para Android.

1.3 Objetivo geral e Objetivos Específicos

1.3.1 Objetivo Geral

Desenvolver, através do uso das melhores práticas de desenvolvimento de sistemas, uma aplicação móvel que facilite o acesso aos resultados das loterias Federais no Brasil.

1.3.2 Objetivos específicos

- Realizar pesquisas sobre aplicativos existentes no mercado voltados para loterias.
- Analisar todos os requisitos necessários para construção do aplicativo.
- Desenvolver a aplicação seguindo os princípios e melhores práticas da engenharia de software, bem como do desenvolvimento de sistemas computacionais.

1.4 Justificativa

Hoje em dia a maioria das pessoas busca a praticidade que os smartphones com seus milhares de aplicativos podem trazer. Para quem gosta de fazer apostas em loterias, ter um aplicativo que mostre os resultados é muito vantajoso.

Como hoje os resultados da loteria são buscados principalmente pelo site da Caixa e pela TV, o desenvolvimento de aplicativos destinados a mostrar os resultados dos jogos irá tornar a vida dos apostadores mais simples e prática, pois hoje quase todo mundo tem em mãos um smartphone.

Já existem alguns aplicativos desenvolvidos para esse objetivo. Porém, isso não acaba sendo tão prático quanto se pensa, pois, os aplicativos não são tão completos e tornam-se uma dificuldade para os apostadores, que acabam muitas vezes desistindo. Aqueles que não desistem simplesmente esquecem de consultar e nem veem o resultado.

Com o objetivo de auxiliar os apostadores nesse sentido é viável o desenvolvimento de um aplicativo completo capaz de cadastrar o jogo.

A ideia é o próprio aplicativo avisar se o apostador ganhou ou não, e também ler o código de barras do volante e mostrar a pontuação, facilitando, assim, a vida dos milhares de apostadores espalhados pelo nosso país.

2 REVISÃO DA LITERATURA

2.1 Sistemas de Informação

De acordo com Puga (2013), um sistema pode ser definido como um conjunto de elementos (órgão, recurso, funções) interligados de forma organizada, visando atingir um objetivo comum.

Um sistema de informação pode ser definido tecnicamente como um conjunto de componentes inter-relacionados que coletam, processam, armazenam e distribuem informações destinadas a apoiar a tomada de decisões. A coordenação e o controle de uma organização contido no sistema, se tem o que Laudon (2010) chama de visão sistêmica.

Essa visão sistêmica consiste em identificar todos os passos de um sistema, que é a entrada, o processamento a saída e o feedback. Identificando esses processos, é possível conseguir um sistema bem desenvolvido.

Lembrando também que para este sistema ser bem desenvolvido é preciso levar em consideração um ponto muito importante que é prezar na qualidade desse sistema.

2.2 Qualidades nos Sistemas de Informação

Qualidade de informação (QI) torna-se cada vez mais essencial para uma organização, uma vez que as informações de qualidade servem de apoio para a tomada de decisão.

O processo de decisão consiste em analisar um problema, chegando à sua resolução. Para tanto, é necessário que haja um conhecimento específico do problema, o que dependerá das informações que se tem a respeito dele (OLIVEIRA,2008).

Então, o bem mais precioso que temos é a informação, é com ela que vamos conseguir resolver os problemas. Contudo, essa informação precisa vir de forma eficaz sem estar distorcida, nem corrompida, para que haja uma satisfatória conclusão do projeto.

Quando se fala em informação costuma-se pensar em dois pontos: Qualidade e coerência. A informação quando vinda com esses pontos se torna o bem maior de qualquer empresa. Para essa qualidade ser efetiva dentro de um sistema precisa-se seguir as etapas no desenvolvimento de sistemas para o desenvolvimento ficar mais organizado e ter ganhos consideráveis em tempo e qualidade.

No desenvolvimento de um sistema, precisamos nos atentar as etapas a seguir para um desenvolvimento de forma consciente, com um norte a se seguir. Com isso, teremos um

padrão no desenvolvimento e conseqüentemente atenderemos as boas práticas. Ao final de cada etapa é importante que seja feita a revisão, pois esse é um ponto muito importante dentro das etapas.

Assim as principais etapas para o desenvolvimento são:

- Planejamento;
- Construção;
- Desenvolvimento;
- Avaliação;

Seguindo essas etapas, que são as principais, temos de forma organizada o desenvolvimento de qualquer aplicação, para que o resultado seja o mais satisfatório possível.

Para se desenvolver um sistema, é necessário fazer a modelagem e documentação de um sistema. Na maioria dos projetos, os profissionais não dão a devida importância à fase de documentação e desenho do sistema, e assim cometem erros primordiais de análise e de modelagem da aplicação.

UML (*Unified Modeling Language*) é uma linguagem que define uma série de fatos que ajudam a modelar e documentar os sistemas. De acordo com Medeiros (2007), a UML indica apenas as formas que podem ser utilizadas para representar um software em diversos estágios de desenvolvimento.

Ela tem objetivos que são de suma importância no processo de desenvolvimento, podemos citar dentre eles: Documentação; Especificação; Maior visualização lógica do desenvolvimento completo de um sistema de informação.

Conclui-se que a UML ajuda os desenvolvedores a visualizarem o fruto do seu trabalho em forma de diagramas padronizados, facilitando assim sua compreensão perante o sistema.

2.3 Levantamento de Requisitos

No início do desenvolvimento de um sistema, faz-se necessário levantar todos os requisitos que o sistema terá.

De acordo com Somerville (2011), quando fazemos a análise e o levantamento teremos algumas atividades são elas:

- Compreensão do domínio;
- Coleta de requisitos;

- Classificação;
- Resolução de conflitos;
- Definição das prioridades;
- Verificação de requisitos.

Dentro do processo de levantamentos de requisitos, existem os levantamentos de requisitos funcionais e os requisitos não-funcionais.

Os requisitos funcionais definem a função de um sistema, o que ele vai fazer.

Já os requisitos não-funcionais são relacionados a como a aplicação vai se comportar, como será o desempenho dela, como será sua disponibilidade e sua segurança.

Existem várias técnicas de levantamentos de requisitos que tem por objetivo sanar as dificuldades relativas a essa fase de elaboração, lembrando que cada uma das técnicas possui vantagens e desvantagens, e essas técnicas também podem ser utilizadas em conjunto. Algumas das técnicas de levantamento de requisitos são:

- Levantamento orientado a pontos de vista: Esta técnica analisa os pontos de vista de todos os usuários no levantamento de requisitos conhecidos como *stakeholders* e os utilizam no processo de levantamento de requisitos.
- Etnografia: Que é a técnica da observação em que o analista usa da observação para descobrir os requisitos implícitos
- Workshops: Nada mais é do que uma reunião em grupo cujo foco é o trabalho em equipe para determinar os requisitos.
- Prototipagem: Esta técnica tem por objetivo fazer um protótipo a fim de explorar mais as partes críticas do sistema.
- Entrevistas: Essa é a técnica mais comum e simples consiste em fazer uma entrevista seguindo um plano de entrevista e que traz bons resultados.
- Questionários: Se dá na formação de um questionário que os usuários vão responder e através do qual se formam os requisitos.
- Brainstorming: É uma técnica que permite a geração de ideias para os requisitos.

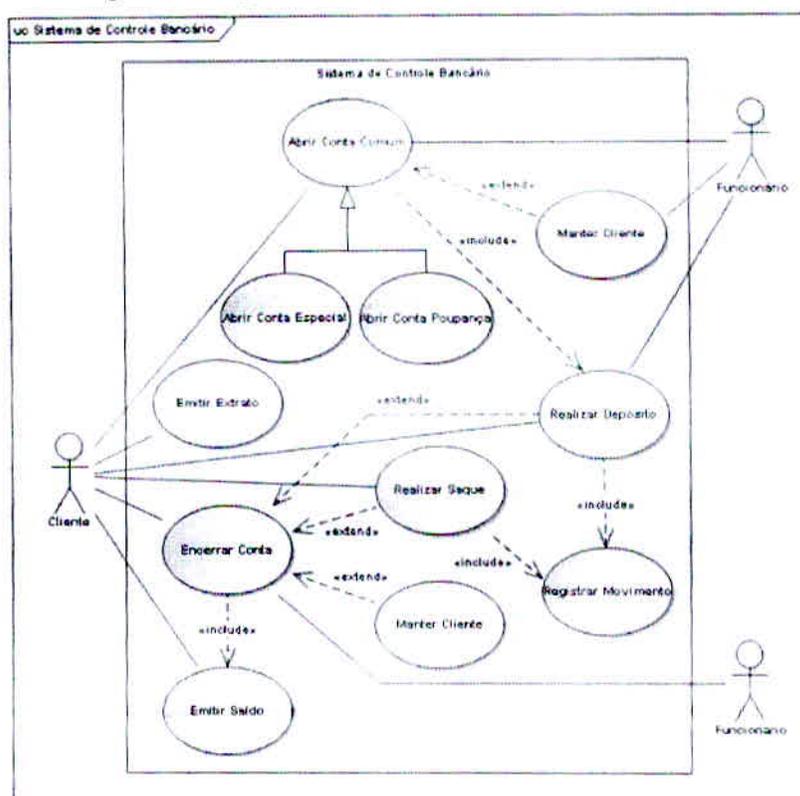
Vimos que se tem várias formas de se fazer um levantamento de requisitos. Através destes requisitos prontos, conseguimos passar para as próximas etapas da engenharia de software e desenvolver os diagramas pois já sabemos o que o aplicativo irá fazer.

2.4 Diagramas de Caso de Uso

No processo de desenvolvimento de um sistema, é necessário considerar a visão do usuário. Para auxiliar com relação a esse aspecto, há diagramas de casos de uso.

De acordo com Guedes (2011), o diagrama de casos de uso mostra o que o sistema faz do ponto de vista do usuário, ou seja, descreve as funcionalidades do sistema e as interações delas com os usuários. Abaixo é mostrado um diagrama de casos de uso de controle bancário.

Figura 3 – Diagrama de Casos de Uso de Controle Bancário



Fonte: Guedes (2011).

Enfim, podemos dizer que o Diagrama de Casos de Uso consegue mostrar como o sistema irá se comportar, como serão os processos, as interações que o sistema irá realizar, tendo o usuário como principal ator.

Por esses motivos que o Diagrama de Caso de Uso é tão importante dentro de um desenvolvimento de um sistema, ela nos auxilia e muito a ter uma visão sistêmica.

3 FERRAMENTAS TECNOLÓGICAS

Este capítulo tem como objetivo demonstrar todas as ferramentas tecnológicas usadas no desenvolvimento de um sistema.

3.1 Java

A linguagem Java começou a ser criada em 1991, e em 1995 foi lançada pela empresa *Sun Microsystems*. Naquela época era uma linguagem de programação que trabalhava em sites produzidos na *World Wide Web*. Hoje essa tecnologia já evoluiu consideravelmente (GONÇALVES, 2006).

O Java consiste em uma linguagem de programação em que é possível se fazer tudo, ou seja, ela não está presa a uma arquitetura ou uma empresa, é muito rápida, estável, é uma linguagem de alto nível.

Para Hostman, Cornell (2010), o Java possui várias características que o tornam uma das melhores linguagens de todo mundo. Essas características são: simples; orientado a objetos; portátil; compatibilidade com redes; robusto; seguro; arquitetura neutra; interpretado; alto desempenho; múltiplos threads; dinâmico.

De acordo com as características citadas acima, pode-se dizer que o Java é simplesmente uma linguagem completa, bem estruturada, com muita segurança e de fácil aprendizado. É uma linguagem que usa o paradigma orientação a objeto.

As linguagens mais comuns são compiladas para código nativo, porém a linguagem Java é exatamente o contrário, ela é compilada para um *bytecode* que é executado por uma máquina virtual. Dessa forma um programa feito nesta linguagem pode rodar em vários sistemas operacionais sem a necessidade de alterá-los para a plataforma do sistema, bastando apenas que uma máquina virtual compatível esteja instalada Deitel (2005).

Uma de suas características é que ela é multiplataforma, ou seja, pode ser utilizada em muitos sistemas operacionais sem que ele seja alterado, basta apenas ter uma máquina virtual instalada (DEITEL, 2005). É uma das linguagens mais utilizadas em todo o mundo.

Um aplicativo em Java é formado por inúmeras partes chamadas de classes. As classes também são formadas por várias partes denominadas métodos, eles realizam tarefas e devolvem informações quando tudo estiver pronto.

Graças à orientação a objeto o qual permite a reutilização das classes, os desenvolvedores podem utilizar das classes existentes nas bibliotecas de classe Java, conhecidas como APIs. Ela possui duas bibliotecas gráficas são elas: Swing e SWT. Por tudo que foi dito anteriormente o Java é uma das aplicações que mais cresceu nos últimos anos. Deitel (2005).

3.2 Eclipse

Eclipse é uma IDE de desenvolvimento de programação, ou seja, é um programa de computador que reúne características e ferramentas de apoio ao desenvolvimento de software com o objetivo de agilizar este processo. Criada primeiramente pela IBM, que gastou muito antes desta ferramenta se tornar *Open Source*.

O Eclipse fornece um ambiente integrado para a execução dos plug-ins, ou seja, o Eclipse não desenvolve apenas sistemas na linguagem Java. Ele pode desenvolver em qualquer uma desde que tenha um plug-in desenvolvido para ela, caso não tenha nenhuma, qualquer pessoa poderá desenvolver (GONÇALVES, 2006).

Para entendermos melhor o Eclipse precisamos conhecer a sua arquitetura, que é composta da seguinte forma:

- Runtime que tem a seguinte tarefa descobrir quais plug-ins existem disponíveis no diretório do Eclipse;
- Workspace tem a responsabilidade de gerenciar os recursos do usuário que são organizados em um ou em vários projetos;
- Workbench é a interface gráfica do Eclipse; Help que nada mais é do que o componente de ajuda do Eclipse o qual possui uma documentação muito extensa. (GONÇALVES, 2006)

Esses são os componentes básicos que compõem a arquitetura do Eclipse, uma ferramenta fantástica que se pode ser utilizada em qualquer projeto de desenvolvimento com Java. Ele oferece, ainda, uma gama fantástica de plug-ins que irá auxiliar qualquer programador no processo de desenvolvimento da sua aplicação, por estes motivos o Eclipse é uma ferramenta tão utilizada em todo o mundo.

3.3 Android Studio

O Google lançou uma nova IDE Android, o Android Studio, que tem como base a versão do IntelliJ IDEA e possui muitas funcionalidades se comparada à sua concorrente Eclipse.

O Android Studio conta com todos os pacotes para o desenvolvimento da plataforma Android, incluindo o SDK, os emuladores e ferramentas.

Alguma das características do Android Studio são:

- Pré-visualização do layout: Enquanto você desenvolve o projeto já pode ter uma ideia de como ele vai ficar em qualquer resolução de tela suportadas o que facilita muito.
- Um construtor de layout excelente do tipo arrastar e soltar.
- Análise de seus códigos baseadas na API do google.

Ele possui, ainda, uma interface muito atraente bem intuitiva e conta com vários temas.

Seu auto-complete dos códigos é completo e muito rápido, como vimos nas características a pré-visualização do layout está muito completa.

É um recurso que ajuda e muito na hora de desenvolver algum layout de um aplicativo.

Além que falando em desenvolver o layout o arrastar e soltar elementos na tela o torna muito mais fácil e prazeroso o desenvolvimento de um aplicativo para Android.

3.4 Bancos de Dados

Em nosso cotidiano existem vários Banco de Dados (BD). Date (2004) afirma que um banco de dados é uma coleção de dados persistentes utilizada pelos sistemas de aplicação de uma empresa. Um exemplo simples de banco de dados seria uma lista telefônica.

Certos princípios devem ser considerados para se obter um BD eficiente. São eles: redundância, inconsistência e integração. Ter um Banco de Dados é de suma importância, pois precisamos dele para que a aplicação funcione de forma eficaz e consiga salvar informações e interações.

Um banco de dados conta um sistema gerenciador de banco de dados (SGBD) que é uma coleção de permite aos usuários criar e manter um banco de dados (ELMASRI, 2010).

Existem vários SGBDs no mercado, dentre eles podemos citar: Oracle Database da Oracle; SQL Server da Microsoft; PostgreSQL Group e o MySQL da MySQL AB.

Para que possamos guardar a informação dentro de um banco de dados, podemos usar tipos diferentes de armazenamento de dados. Temos o banco de dados multidimensionais e relacionais. O multidimensional ainda é novo no mercado, o seu crescimento se deve em parte pelo crescimento da *datawarehouse*, já o relacional é o mais utilizado, pode-se dizer que é o mais comum para o desenvolvimento de sistemas (Dennis & Wixon, 2005).

Para a utilização de um banco de dados, iremos utilizar o *sqlite* que é um banco de dados que o próprio Android tem. Ele gera um banco de dados que pode ser entregue junto com a aplicação, ou seja, quando se instala a aplicação ele fornece o banco de dados junto com a aplicação.

4 METODOLOGIA

Primeiramente foi feita uma pesquisa na internet e em sites relacionados a aplicativos, para entender a real necessidade dos usuários, pesquisas também foram feitas no Google Play (para ver se existem aplicativos voltados para loterias) foram encontrados aplicativos que tem como objetivos mostrar os resultados de loterias, como o próprio aplicativo da Caixa Econômica Federal e também o aplicativo Loteria do Brasil mais nenhum deles pode se conferir o resultado através do leitor de código de barras.

A pesquisa bibliográfica foi feita sobre os temas principais que são as loterias e aplicativos Android, e também sobre as ferramentas utilizadas no desenvolvimento, suas características, as vantagens e desvantagens sobre seus concorrentes.

Com os dados previamente colhidos, foram analisados e filtrados somente os que forem de alguma utilidade para o desenvolvimento do sistema.

Chega a hora de levantar os requisitos do sistema, para isso foi utilizado a técnica de observação que permite coletar informações de acordo com o cotidiano dos usuários e das operações por eles executadas, afim de descobrir todos os requisitos funcionais e não funcionais do sistema.

Esta técnica de levantamentos foi escolhida porque ela tem vantagens significativas em relação a real necessidade dos usuários, porque ela é capaz de captar o comportamento natural das pessoas, e o melhor o nível de intromissão é muito baixo, o que faz com que seja mais fácil de capturar as informações necessárias dos usuários sem que eles se sintam incomodados com isso.

Logo após a coleta de dados foi possível destacar os requisitos funcionais, que são mostrados no quadro abaixo:

Tabela 1 – Requisitos funcionais do aplicativo.

Requisitos Funcionais
<ul style="list-style-type: none"> ➤ O sistema deverá permitir ao usuário escolher a loteria desejada. ➤ O sistema deverá disponibilizar o resultado do último sorteio de cada loteria. ➤ O sistema deverá ler o código de barras do volante e mostrar se o volante está premiado ou não. ➤ O sistema deverá consumir uma API de um web service para retornar os resultados das loterias.

Além dos requisitos funcionais também foram encontrados os requisitos não funcionais como pode ser visto no quadro abaixo:

Tabela 2 – Requisitos não funcionais do aplicativo

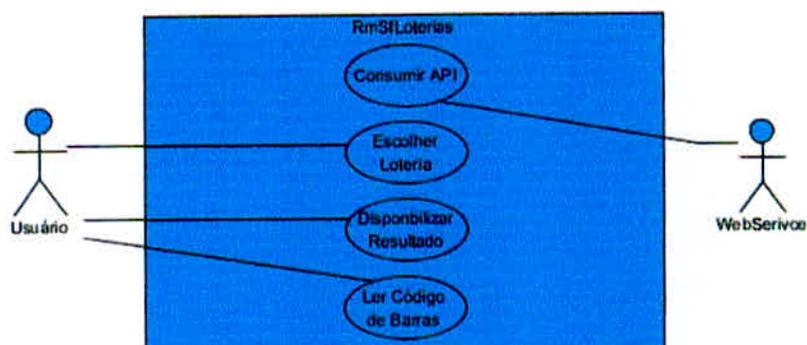
Requisitos Não Funcionais
<ul style="list-style-type: none"> ➤ O sistema deverá ser desenvolvido em Java. ➤ O sistema deverá utilizar a IDE Android Studio para o seu desenvolvimento. ➤ O sistema deverá ser composto por telas amigáveis, para melhor navegação do usuário. ➤ O sistema deverá mostrar o resultado após ler o código de barras no volante em no máximo 5 segundos. ➤ O Aplicativo deve funcionar em dispositivos Android, com versão igual ou superior à 4.4.2.

Já com os requisitos em mãos se tem um ponto muito importante que é a modelagem, utilizaremos um diagrama da UML que no caso será o diagrama de casos de uso.

Que segundo Rezende (2005), o principal propósito do Diagrama Entidade Relacionamento (DER) é representar os objetos de dados e suas relações, sendo que cada entidade termina representada pelo menos por uma tabela de dados.

Podemos ver abaixo o diagrama de casos de uso do aplicativo:

Figura 4 – Diagrama de casos de uso do aplicativo



Fonte: Desenvolvido pelo autor.

O próximo passo foi o desenvolvimento do aplicativo seguindo padrões e boas práticas, como qualidade de informação e usabilidade para que a experiência do usuário seja a melhor possível. E enfim o aplicativo resolverá o problema que é desenvolver um aplicativo que possa ler o código de barras do volante e dizer ao usuário se o volante se encontra premiado ou não.

Abaixo temos a tabela com o cronograma de todas as etapas do desenvolvimento do aplicativo.

Tabela 03 - Cronograma

Etapas da pesquisa	Período de realização
Pesquisa bibliográfica sobre aplicativos voltados para loterias	Julho/agosto
Levantamento requisitos necessários para desenvolvimento do aplicativo	Julho/agosto
Desenvolvimento do Escopo do projeto	Primeira quinzena de setembro
Implementação do aplicativo	Setembro/outubro
Teste do aplicativo	Setembro/outubro
Revisão de partes do projeto para adequação e escrita final da monografia	Outubro
Escrita de monografia final	Novembro

5 RESULTADOS

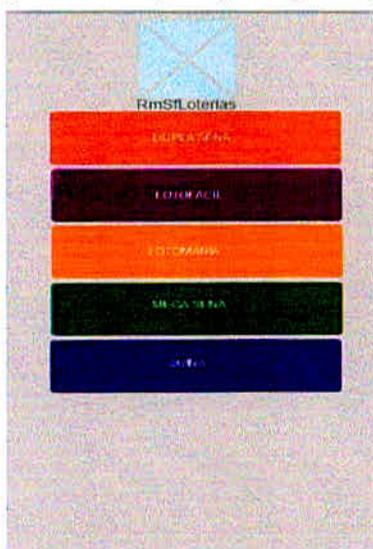
Primeiramente o passo mais importante logo após a pesquisa feita, em sites, blogs, o levantamento de requisitos foi feito, a técnica utilizada foi a de observação.

Nesta técnica podemos ver o comportamento das pessoas sem que essas pessoas se sintam constrangidas. Foi feita uma observação em casas lotéricas de pessoas que tem o hábito de fazer apostas, tanto individuais, quanto em bolões.

Depois dessa observação descrevemos, como foi possível observar no item 5 Metodologia que o sistema irá apresentar, diante disso foi desenvolvido um protótipo que mostra como será todas as telas do sistema, seguindo os requisitos funcionais apresentados na metodologia.

Abaixo é apresentada a figura da tela principal do protótipo, a logo é mostrada acima e abaixo os botões do menu, um para cada loteria:

Figura 5 – Tela principal do protótipo.



Ao clicar no botão de *Dupla sena*, o usuário é redirecionado para a tela da dupla sena onde temos 2 botões: um para voltar ao *Menu Principal* e outro para poder acionar o *Código de Barras*. Logo abaixo temos o resultado do último concurso:

Figura 6 – Tela Dupla Sena Protótipo



Ao clicar no botão *Código de Barras* da tela dupla sena é chamada a leitura de código de barras e depois de feita é redirecionada para uma outra tela onde é mostrado se o volante foi premiado ou não e mostra também as dezenas lidas, nesta tela ainda encontramos um botão para voltar a *tela anterior*:

Figura 7 – Tela Dupla Sena Código de Barras (Volante Premiado ou não)



O mesmo procedimento foi realizado para as outras categorias de jogos. Depois de feito o protótipo, foi realizado o projeto utilizando o Android Studio seguindo como base o diagrama de casos de uso e tendo como referência o protótipo recém-criado.

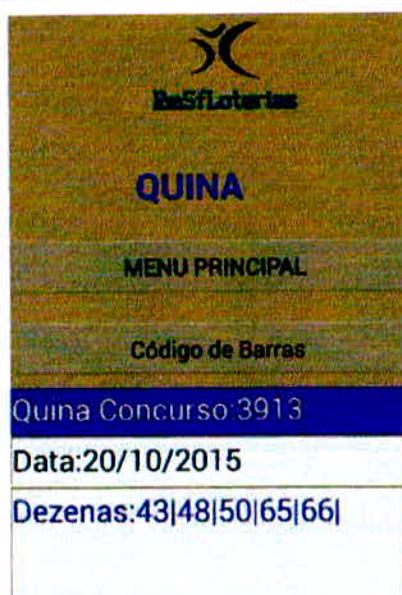
Foi utilizado para o desenvolvimento do projeto a API 19 do Android, com a target Android 4.4.2, a primeira tela criada, com o nosso menu onde temos os botões para escolha da loteria criada, são elas:

- *Dupla Sena;*
- *Loto fácil;*
- *Lotomania;*
- *Mega Sena;*
- *Quina;*

O leitor de código de barras será em uma versão futura do aplicativo.

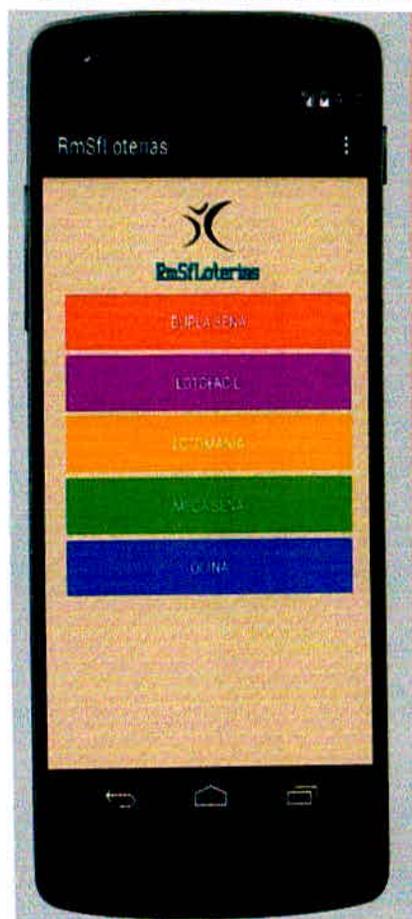
Na parte que mostra o resultado do último sorteio os números são consumidos via Json, colocado manualmente, ou seja, foi pego um resultado fictício e colocado em uma String JSON_STRING para mostrar o resultado desejado, foi criado um novo objeto Json para exibir as informações, que futuramente será substituído por uma API de resultados de loterias. Foi desenvolvido um protótipo com todas as telas do sistema, e foi desenvolvido uma implementação básica que é o resultado consumido pelo Json, como mostrado na tela a seguir:

Figura 8 – Tela Resultado da Quina via Json.



A seguir é mostrado a tela principal do sistema no emulador configurado no Android, nele é possível ver a os botões de menu e também a logomarca, é um botão para cada loteria.

Figura 9 - Tela principal do sistema



6 CONCLUSÃO

Os motivos para o desenvolvimento deste projeto foram a paixão pela loteria, a necessidade em ajudar as pessoas, e a forma que foi encontrada é ajudar os apostadores que tem dificuldade em consultar os resultados.

Vários benefícios foram vistos com a criação deste projeto, além do amplo aprendizado, ajudar a todos os apostadores que tem dificuldades em conferir os seus volantes, o projeto já consegue ser de valia, pois através dele já se tem uma fonte com o último resultado de cada loteria.

O objetivo geral foi cumprido através do desenvolvimento de um aplicativo que torne mais fácil a conferência dos resultados das loterias.

As hipóteses foram negadas, pois o aplicativo se encontra com algumas limitações que deverão ser sanadas em uma versão futura.

Para a versão futura, o aplicativo consumirá uma API que mostrará os últimos resultados das loterias, e também será implementada a leitura do código de barras para o total funcionamento do sistema e assim poder colocá-lo a disposição para que os usuários façam o download e utilizem da forma como desejar.

O projeto trouxe um desafio enorme de poder desenvolver um aplicativo que fizesse algo que nenhum outro aplicativo no mercado faz, ou seja, que tenha um diferencial que atraia os usuários e atenda às suas necessidades, que é o mais importante, mesmo com as limitações que o aplicativo apresenta, já consegue tornar mais fácil a consulta aos resultados das loterias.

REFERÊNCIAS

- BATEBYTE. **Processo de Desenvolvimento de Sistemas com Qualidade**. Disponível em <<http://www.batebyte.pr.gov.br/modules/conteudo/conteudo.php?conteudo=1492>>
- CATUSO, Jaime. **Utilizando android SDK no Desenvolvimento de aplicações para dispositivos móveis**. Medianeira. 2011.
- DATE, C. J. **Introdução a Sistemas de Banco de Dados**; tradução[da 7. ed. Americana]. Rio de Janeiro: Campus, 2004.
- DEITEL, H. M.; DEITEL, P. J. **Java - Como Programar**. 6.ed. São Paulo: Pearson Prentice Hall, 2005.
- DENNIS, Alan & WIXOM, Barbara Haley. **Análise e Projeto de Sistemas**. Rio de Janeiro: LTC, 2005, 2ª. Ed.
- DEVMEDIA. **Sqlite. O pequeno Notável**. Disponível em <<http://www.devmedia.com.br/sqlite-o-pequeno-notavel/7249>>.
- DEVMEDIA. **Engenharia de Software 2 – Técnicas para levantamento de requisitos** Disponível em <<http://www.devmedia.com.br/engenharia-de-software-2-tecnicas-para-levantamento-de-requisitos/9151>>.
- DICAS DE PROGRAMAÇÃO. **O que é Banco de Dados?** Disponível em <<http://www.dicasdeprogramacao.com.br/o-que-e-um-banco-de-dados/>>.
- ELMASRI, Ramez. **Sistemas de banco de dados / Ramez Elmasri e Shamkant B. Navathe**; revisor técnico Luis Ricardo de Figueiredo. – São Paulo: Addison Wesley, 2010.
- GONÇALVES, Edson. **Dominando o Eclipse – Tudo que o Desenvolvedor precisa para Criar Aplicativos para Desktop**. Ciência Moderna. 2006.
- GUEDES, G. T. **UML - Uma abordagem prática** 2º Edição. 2011. Novatec.
- HOSTMAN, CORNELL, CAY S, GARY. **Core java**, volume I: fundamentos / Cay S. Horstmann, Gary Cornell; tradução Carlos Schafranski e Edson Furmankiewicz; revisão técnica Nivaldo Foresti.-i.ed.-São Paulo: Pearson Prentice Hall, 2010.
- INFOQBRASIL. **Android Studio: Uma nova IDE do google baseada no IntelliJ IDEA** Disponível em <<http://www.infoq.com/br/news/2013/05/android-studio> Acesso em 28/09/2015 às 23h33min.
- IMASTERS. **Android Studio: Vantagens e desvantagens com relação ao eclipse**. Disponível em <<http://imasters.com.br/mobile/android/android-studio-vantagens-e-desvantagens-com-relacao-ao-eclipse/>>.
- JAVAFREE.ORG. **Tutorial Java: O que é java?** Disponível em <<http://javafree.uol.com.br/artigo/871498/Tutorial-Java-O-que-e-Java.html>>

LAUDON, Kenneth C.; LAUDON, Jane P. **Sistemas de informação gerenciais**. 9. ed. São Paulo: Pearson Prentice Hall, 2010.

LOPES, José Euclides da Silva Lopes. **SISJOL – Sistema de controle acadêmico para escola jovem online**. Monografia (Graduação em Sistemas de Informação) – Faculdade de Balsas, Maranhão, 2013.

MAKESYSBLOG. Definindo seu software: O que são requisitos funcionais e não funcionais? Disponível em < <http://blog.makesys.com.br/definindo-seu-software-o-que-sao-requisitos-funcionais-e-nao-funcionais>>.

MEDEIROS, LUCIANO FRONTINO DE. **Banco de Dados: princípios e prática** [livro eletrônico] / Luciano Frontino de Mediros. - Curitiba: InterSaberes, 2004.

OLIVEIRA, Alexsandro Juliano. **Qualidade da Informação do Sistema Integrado de Gestão Utilizado na Força Aérea Brasileira**. Monografia (Graduação em Administração) – Escola de Administração, UFRGS, Porto Alegre, 2008.

PALMBRASIL. **Quais ferramentas são necessárias para desenvolver aplicativos para android**. Disponível em < <http://www.palmbrasil.com.br/android/faq/129-desenvolvimento-de-aplicacoes-para-android/1940-quais-ferramentas-sao-necessarias-para-desenvolver-aplicativos-para-android>>.

PORTAL SIS. **Etapas do processo de desenvolvimento de sistemas**. Disponível em < <https://portalsis.wordpress.com/2011/06/20/etapas-do-processo-de-desenvolvimento-de-sistemas/>>.

PUGA, SANDRA. Banco de Dados: implantação em SQL, PL/SQL e Oracle 11g / Sandra Puga, Edson França, Milton goya. – São Paulo: Pearson Education do Brasil, 2013.

REZENDE, Denis Alcides ; Denis Alcides Rezende , **Engenharia de software e sistemas de informação**. 3. Ed. rev. Ampl -- Rio de Janeiro: Brasport. 2005.

SISTEMAS DE INFORMAÇÃO. **Qualidade em Sistemas de Informação**. Disponível em < <http://sisdinf.blogspot.com.br/2007/06/qualidade-em-sistemas-de-informao.html>>.

SOMERVILLE, I. **Engenharia de software**. 9º ed. Tradução Maurício de Andrade. São Paulo: Ed Addison-Wesley, 2011.