

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO RIO
GRANDE DO NORTE
CAMPUS AVANÇADO LAJES
CURSO TÉCNICO INTEGRADO EM INFORMÁTICA

ANA STEPHANE DANTAS FERNANDES

**A INFORMÁTICA ALIADA À COMUNICAÇÃO: PROTÓTIPO DO APLICATIVO
ANDROID “COMUNIC-LAJES”**

LAJES/RN
2020

ANA STEPHANE DANTAS FERNANDES

**A INFORMÁTICA ALIADA À COMUNICAÇÃO: PROTÓTIPO DO APLICATIVO
ANDROID “COMUNIC-LAJES”**

Relatório de Prática Profissional apresentado ao Curso Técnico Integrado em Informática do Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte, Campus Avançado Lajes, em cumprimento às exigências legais como requisito parcial à obtenção do título de Técnico em Informática.

Orientador (a): Prof. Me. Dannilo Martins Cunha

RESUMO

No limiar do século XXI, a tecnologia é a principal ferramenta para propagação de informações pelo mundo. As tecnologias da informação possibilitaram novas maneiras de interagir e se comunicar com a sociedade ao redor. Dessa forma, este trabalho fundamenta-se na aplicabilidade da Informática como instrumento auxiliador no processo de comunicação social, tendo como objetivo principal o desenvolvimento de um aplicativo *Android* responsável por listar as notícias do campus. Para tanto, foi utilizado como objeto de pesquisa o formato de comunicação social proposto pelo Campus Lajes. Dessa forma, foi aplicada uma metodologia que parte de uma pesquisa de caráter exploratório que investiga o formato de comunicação do campus, a fim de averiguar os pontos positivos e negativos desta para a comunidade acadêmica, e como se dá o seu funcionamento. O estudo de casos foi desenvolvido com base nos testes feitos com o aplicativo já implementadas as funções previstas. Portanto, os estudos e resultados obtidos permitiram a construção do aplicativo *mobile Android* Comunic-Lajes, que ainda se encontra na fase de protótipo, mas atende aos objetivos propostos inicialmente neste trabalho.

Palavras-chave: Informática. Comunicação. Aplicativo. Android.

ABSTRACT

At the threshold of the 21st century, technology is the main tool for spreading information around the world. Information technologies have enabled new ways to interact and communicate with the surrounding society. Thus, this work is based on the applicability of Informatics as an aid tool in the social communication process, having as main objective the development of an Android application responsible for listing campus news. For this purpose, the social communication format proposed by Campus Lajes was used as a research object. In this way, a methodology was applied that starts from an exploratory research that investigates the communication format of the campus, in order to ascertain the positive and negative points of this for the academic community, and how it works. The case study was developed based on the tests done with the application, the functions already foreseen already implemented. Therefore, the studies and results obtained allowed the construction of the Android Comunic-Lajes mobile application, which is still in the prototype phase, but meets the objectives initially proposed in this work.

Keywords: Informatics. Communication. App. Android.

LISTA DE ABREVIATURAS E SIGLAS

XML	<i>Extensible Markup Language</i>
IDE	<i>Integrated Development Environment</i>
IFRN	Instituto Federal do Rio Grande do Norte
TIC	Tecnologias da Informação e Comunicação

SUMÁRIO

1	INTRODUÇÃO	6
1.1	OBJETIVO GERAL	6
1.2	OBJETIVOS ESPECÍFICOS	6
1.3	JUSTIFICATIVA	7
1.4	METODOLOGIA	8
1.5	LIMITAÇÕES DO TRABALHO	9
1.6	ESTRUTURA DO TRABALHO	10
2	FUNDAMENTAÇÃO TEÓRICA	11
2.1	AS TECNOLOGIAS DA INFORMAÇÃO (TIC'S) E A REVOLUÇÃO TECNOLÓGICA	11
2.2	A COMUNICAÇÃO SOCIAL INTERATIVA	12
2.3	O DESENVOLVIMENTO <i>ANDROID</i>	12
2.4	<i>ANDROID STUDIO</i>	13
2.5	JAVA	14
2.6	XML	14
3	APLICATIVO DESENVOLVIDO	15
3.1	ESBOÇO DA APLICAÇÃO	15
3.2	FLUXO DE TELAS	19
3.3	PACOTES DO PROJETO, CLASSES E MÉTODOS	20
3.3.1	<i>Pacotes do projeto</i>	20
3.3.2	<i>Classes do projeto</i>	21
3.3.3	<i>Métodos</i>	28
3.4	MODELAGEM DO BANCO DE DADOS	32
4	ESTUDO DE CASO	34
5	CONSIDERAÇÕES FINAIS	38
5.1	<i>TRABALHOS FUTUROS E RECOMENDAÇÕES</i>	39
	REFERÊNCIAS	41

1 INTRODUÇÃO

Para PEREIRA e SILVA (2010), desde o surgimento da Internet, tem-se um mundo cada vez mais conectado e globalizado. O avanço crescente das Tecnologias da Informação e Comunicação despertaram na indústria o desejo e a busca pelo desenvolvimento tecnológico e, dentro desse leque de inovações constantes, fez-se nascer um grande recurso comunicativo: as aplicações *mobile* ou os famosos *App's*.

O uso de aplicativos impulsionou a difusão da manipulação e acesso a dados em diversas áreas de estudo como saúde, educação, segurança e, sobretudo, na comunicação (OLIVEIRA, 2014). Os *Apps's* de notícias facilitam o acesso e entrega de informações ao usuário, visto que se torna mais prático a este utilizar um recurso já instalado e configurado em seu *Smartphone* do que uma página web, por exemplo.

O *WhastApp* e o *Facebook*, que permitem a troca instantânea de mensagens com usuários que podem estar em localizações geográficas distintas, já possuem centenas de *downloads* no mundo inteiro, concorrendo até mesmo com as próprias operadoras telefônicas (COUTINHO, 2014).

Dessa forma, o uso de aplicações *mobile* se torna cada vez mais frequente, gerando novas possibilidades de acesso às informações globais, criando recursos que auxiliem nas práticas diárias e ampliando conhecimentos.

1.1 OBJETIVO GERAL

O objetivo geral deste trabalho é desenvolver aplicativo Android que transmita todas as notícias acerca do IFRN - Campus Avançado Lajes.

1.2 OBJETIVOS ESPECÍFICOS

A partir da proposta de projeto, foram estabelecidos os seguintes objetivos específicos:

- Estabelecer um cronograma de atividades dividido em etapas de desenvolvimento do projeto;
- Estudar a linguagem de programação Java e a linguagem de marcação XML para Android;
- Elaborar esboço do fluxo de telas do aplicativo e da proposta de layout;

- Construir as telas iniciais do *App* com base nos conceitos estudados sobre o sistema Android;
- Construir Banco de Dados;
- Elaborar a modelagem do Banco de Dados proposto;
- Cadastrar notícias;
- Listar as notícias na página de listagem do *App*;
- Acessar uma notícia em específico;
- Realizar estudo de caso.

1.3 JUSTIFICATIVA

A principal motivação para elaboração deste trabalho consiste no auxílio que pode ser fornecido pela informática quando aliada à comunicação social. À princípio, para o desenvolvimento deste projeto, o recurso chave utilizado como ferramenta de estudo foi o atual formato de comunicação do Campus Lajes. Desde o início, as principais ferramentas utilizadas como meio de comunicação extraoficial são as redes sociais do campus, sobretudo, o *Instagram*. Nele, são feitas todas as divulgações necessárias para a comunidade discente, desde acesso a editais como também a divulgação dos eventos que ocorrem na instituição, tais como SEMADEC, EXPOTEC, Mostras Científicas, entre outros.

Outrossim, sabe-se que o desenvolvimento *mobile* cresce a cada dia e apresenta novas possibilidades de uso. Os *smartphones* são, atualmente, o principal mecanismo de acesso à dados e informações globais, via *web*, funcionando como verdadeiros computadores de bolso (COUTINHO, 2014). Assim, com apenas poucos cliques é possível visualizar notícias de qualquer parte do planeta, bem como se comunicar com pessoas distintas em localizações geográficas diferentes.

Diante disso, o aplicativo Comunic-Lajes traria grandes benefícios para a comunidade acadêmica, principalmente por sua praticidade de uso no que diz respeito ao acesso das notícias em primeira mão que, além de representar um novo meio de comunicação, apresenta-se como uma ferramenta de fácil acesso para todos, visto as possibilidades de alcance da tecnologia *Android*.

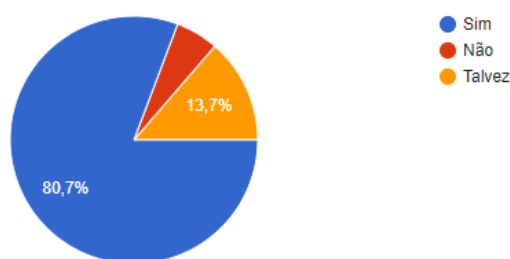
1.4 METODOLOGIA

A metodologia deste trabalho se fundamentou, inicialmente, em uma pesquisa exploratória acerca do repasse de informações entre os setores acadêmicos e os estudantes. No ano de 2018, foi feita uma análise, via questionário Google Forms, acerca do método utilizado para transmitir as notícias do campus, chegando à conclusão de que é necessário um maior canal de comunicação entre os discentes e a instituição. A maioria das notícias envolvendo o Campus Lajes, por exemplo, é transmitida pela página *web* da reitoria, e repassada para a comunidade discente pelo Instagram, não havendo um canal de comunicação específico para tal tarefa. A maioria das perguntas abordava acerca do grau de satisfação da comunidade com o atual formato de comunicação do Campus. No entanto, a última pergunta questionava os estudantes se seria benéfico o desenvolvimento de um aplicativo para *smartphones* no qual estariam expostas todas as notícias do Campus:

Fotografia 1 - Gráfico de resultados via Google Forms

Além da página web citada acima, você gostaria de ter em seu Smartphone um aplicativo (app) que permitisse visualizar as informações do Campus Lajes?

161 respostas



Fonte: Google Forms (2018)

Além disso, foram abertas caixas de diálogo nas quais os estudantes poderiam escrever o que acreditavam que deveria ser revisto na comunicação do campus, como apresentam as imagens 2 e 3, respectivamente:

Fotografia 2 – Caixa de respostas via Google Forms

A criação de páginas e/ou sites mais dinâmicos, atrativos e de fácil navegação.

Fonte: Google Forms (2018)

Fotografia 3 - Caixa de respostas via Google Forms

Mais acessibilidade, outros canais de divulgação.

Fonte: Google Forms (2018)

Dessa forma, constatou-se que, apesar de estarem satisfeitos com o formato de comunicação do campus, seria interesse o desenvolvimento de outros meios para a divulgação de notícias, editais, entre outras informações. Um meio de comunicação que, além de divulgar, exponha tudo que se considera importante para a instituição, como a publicação de editais, chamadas de listas de espera, editais de projetos existentes e demais processos.

Por fim, utilizou-se, ainda, de pesquisas documentais e bibliográficas para o embasamento teórico e levantamento de dados acerca da temática proposta.

1.5 LIMITAÇÕES DO TRABALHO

Uma das limitações apresentadas pelo aplicativo é a ausência de um *chat* de comunicação que dê aos usuários a possibilidade de expor sua opinião sobre o andamento do aplicativo, se ele realmente cumpre com aquilo que foi proposto. Assim, o *chat*, por sua vez, seria útil para sugerir melhorias para o aplicativo, permitindo que o desenvolvedor esteja a par das sugestões do público. Todavia, isso se deve ao fato de que o *app* não foi disponibilizado para testes entre os membros da instituição, visto que a aplicação não possui certificado digital, o qual possibilita a publicação do

aplicativo no *Android Market* (loja de aplicativos para dispositivos *Android*) para seja feito o *download*.

O aplicativo, no geral, faz o que foi destinado a fazer inicialmente, que é o cadastro e a listagem. No entanto, o cadastro, como está implementado, pode ser feito por qualquer usuário, o que é problemático, pois isto prejudica um dos requisitos essenciais para qualquer aplicação, que é a segurança desta.

No mais, a pesquisa e análise dos resultados foi aplicada para uma pequena parcela dos estudantes do campus, o que limita o acesso a uma maior quantidade de respostas da opinião pública acerca do formato de comunicação social do campus.

1.6 ESTRUTURA DO TRABALHO

No capítulo introdutório, tem-se, inicialmente, a contextualização e a retomada histórica acerca dos avanços tecnológicos, que tem por objetivo apresentar ao leitor a temática proposta. Ainda na Introdução, são descritos os seguintes subcapítulos: Objetivos Geral e Específicos; Justificativa; Metodologia, Limitações do Trabalho e Estrutura do Trabalho.

O capítulo seguinte é o da Fundamentação Teórica, no qual são descritos os principais conceitos e ideias trazidos pelos autores, livros, artigos e outros objetos de pesquisa da área que deram embasamento teórico para o desenvolvimento deste trabalho.

No capítulo que leva o nome de Aplicativo Desenvolvido, são detalhadas as etapas de desenvolvimento do *app* Comunic-Lajes, apresentando o fluxo de telas, a montagem e construção dos *layouts* de cada uma delas, além do software e a linguagem de programação e marcação utilizadas para desenvolver a aplicação.

O Estudo de Caso tratará dos testes feitos após o aplicativo pronto. Nas Considerações Finais, bem como no subtópico Trabalhos Futuros e Recomendações, estão as conclusões e recomendações obtidas a partir do desenvolvimento do trabalho e as propostas de desenvolvimento futuras. Por fim, tem-se a sessão de Referências.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 AS TECNOLOGIAS DA INFORMAÇÃO (TIC'S) E A REVOLUÇÃO TECNOLÓGICA

Os avanços técnicos na área da Informática foram de fundamental importância para o surgimento e aprimoramento das Tecnologias da Informação e Comunicação. Dessa forma, atualmente, tem-se grandes resultados na produção de ferramentas capazes de auxiliar o ser humano em diversos contextos: educação, entretenimento, saúde, inclusão social, dentre outros.

A sociedade estabelece contato, direta ou indiretamente, com novas tecnologias quando, por exemplo, assistimos à televisão ou utilizamos serviços bancários on-line etc. Outro ponto de destaque das TICs está relacionado ao processo de ensino. As Tecnologias têm possibilitado a utilização das ferramentas de comunicação no segmento educacional permitindo o início e a ascensão da Educação a Distância (EAD). (PEREIRA E SILVA, 2010, p.5)

Segundo Castells (1996), em sua obra *Sociedade em Rede*, a inovação tecnológica não ocorre como um fator isolado; elas afetam toda uma estrutura socioeconômica existente, contribuindo para o desenvolvimento de outros setores de produção e comunicação. Além disso, Castells (1996, p. 68), na mesma obra, afirma que:

[...] o processo atual de transformação tecnológica expande-se exponencialmente em razão de sua capacidade de criar uma interface entre campos tecnológicos mediante uma linguagem digital comum na qual a informação é gerada, armazenada, recuperada, processada e transmitida.

Assim, o processo evolutivo das novas Tecnologias da Informação, sendo o próprio *smartphone* resultado disto, contribui para que a informação possa ser manipulada de diversas maneiras por meio de “uma linguagem digital”, ou seja, um canal de comunicação propício para a difusão dessas tecnologias.

No entanto, as TIC's não contribuíram somente para o desenvolvimento tecnológico e científico, mas também atuaram como modificadoras do próprio conhecimento humano, sendo responsáveis por alterações de costumes, lazer e,

principalmente, das relações interpessoais (PEREIRA E SILVA, 2010, p.171). Atualmente, as conversas entre pessoas se resumem à tela de bate-papo de alguma rede social. Tudo se apresenta de um jeito mais prático e eficaz.

2.2 A COMUNICAÇÃO SOCIAL INTERATIVA

Para Castells (2009), na obra *Comunicação e Poder*, o ato de comunicar representa um intercâmbio de ideias e de informação, sendo moldado pela tecnologia da comunicação, pelos emissores e receptores desta informação, pela sua conduta cultural e pelo alcance do processo. Dessa forma, a comunicação se faz importante e eficaz no que diz respeito a troca de conhecimento e de informações, definindo quando e como estas serão propagadas.

Dessa forma, o acesso à informação, atualmente, não se resume somente ao rádio ou aos telejornais. Com a popularização das redes sociais, as notícias são propagadas em massa para todos os lugares do mundo, ao passo que, os antigos meios de comunicação foram obrigados a se adaptar aos novos, fazendo com que toda emissora de TV possua, além de um telejornal próprio, sua página web, rede social ou aplicativo para divulgação de notícias. Para Mirian Redin de Quadros (2013, p. 17):

Representantes da lógica colaborativa e participativa da segunda geração da Web 2.0 (O'REILLY, 2005), as redes sociais digitais manifestam-se na rede mundial de computadores por meio de sites como o Facebook, Twitter, Orkut, Google+, entre outros, alcançando expressivos índices de usuários, acessos e faturamento. Ocupando posições cada vez mais centrais na comunicação digital, esses sites atraem desde usuários comuns, que encontram ali um ambiente propício para a manutenção de seus laços sociais, bem como a expressão desintermediada de opiniões, até organizações privadas e públicas, entre elas empresas e veículos de comunicação social que descobriram nas redes sociais na internet (RSIs) poderosas ferramentas com usos diversos, que vão desde o reforço de imagem e marca, passando pela disseminação de conteúdos até a interação com seus públicos-alvo.

2.3 O DESENVOLVIMENTO *ANDROID*

A tecnologia *Android* domina o mercado ao se tornar um dos primeiros sistemas operacionais a desenvolver aplicativos de código-fonte aberto para telefones celulares (ABLESON; SEN; KING; ORTIZ, 2012, p. 3). Dessa forma, tudo que é construído pode

ser acessado e modificado por outros que tenham interesse em melhorar ou conhecer a aplicação desenvolvida.

De certo, a praticidade e expansão do desenvolvimento dessa tecnologia motivou estudantes da Universidade Luterana do Brasil (Ulbra) a construir aplicativo baseado no sistema *Android* que facilite a comunicação de pessoas com transtorno Autista. Para Mello e Sganzerla (2013, p. 231):

Acredita-se que o indivíduo autista desenvolverá sua capacidade de comunicação pelo uso desta ferramenta tanto como fonte de novos conhecimentos (aprendendo novas palavras, (re)conhecendo as letras, ...), quanto como jogo passatempo (utilizando as atividades como reforçadores daquilo que já aprendeu). Em ambos os casos, espera-se que o usuário beneficiasse deste aplicativo.

Em 2012, na Universidade Federal de Minas Gerais, o mecanismo *Android* também foi usado para a construção de aplicativo de referência de vacinação no Brasil, com o intuito de listar toda a sequência de vacinas que devem ser adquiridas pela população desde o primeiro mês de idade.

Um dos pontos da saúde pública que mais efetivamente tem melhorado nas últimas décadas é a vacinação. O programa brasileiro tem obtido sucesso em diversas frentes, e o estímulo à educação para promover conhecimento nessa área é de suma importância. Esse ponto justifica o presente trabalho, que propende relatar a construção de um aplicativo de referência em vacinação utilizando um framework de desenvolvimento para plataformas móveis, com posterior avaliação de usabilidade pelos usuários. (OLIVEIRA RODRIGUES E COSTA, p. 24)

2.4 ANDROID STUDIO

A ferramenta *Android Studio*, disponibilizado gratuitamente pela Google, é uma IDE utilizada para desenvolvimento de aplicações *Android*, promovendo um ambiente de desenvolvimento multiplataforma. (IMASTERS, 2013).

A IDE Android Studio possui algumas vantagens como, por exemplo, o gerenciador de dependências Gradle (vide seção **Links**), também baseado no IntelliJ, muito utilizado fora do Brasil. Este é um dos grandes trunfos do editor da plataforma, pois oferece mais opções ao desenvolvedor na hora de compilar, já que o Eclipse utilizava o jeito clássico de compilação. (DEVMEDIA, 2016)

O *Android Studio* pode ser instalado nos sistemas operacionais Windows, OSX e Linux, necessitando de um certo espaço de memória se ter um funcionamento desejável:

É necessário ter o Java instalado na máquina através do JDK (Java Development Kit) e não a JRE, como normalmente é instalado, pois para desenvolver em Android é necessário que todas as classes de desenvolvimento do Java estejam presentes na máquina. (DEV MEDIA, 2016).

Portanto, o *Android Studio* reúne características e ferramentas de apoio para o desenvolvimento de aplicações *mobile*. (TECMUNDO, 2019).

2.5 JAVA

A linguagem Java utiliza uma sintaxe que deriva do C++, tendo como principal característica o fato de que todo código é escrito dentro de uma classe e tudo é um objeto. As aplicações Java podem ser executadas em qualquer plataforma que possua a *Java Virtual Machine* (JVM), independentemente do tipo de computador utilizado. (TECMUNDO, 2009).

A linguagem Java deriva da linguagem C, portanto suas regras de sintaxe assemelham-se às regras de C. Por exemplo, os blocos de códigos são modularizados em métodos e delimitados por chaves ({ e }) e variáveis são declaradas antes que sejam usadas. (PERRY, 2016)

Em sua estrutura, o Java trabalha com pacotes, que não uma ferramenta de *namespace* da linguagem. Nesses pacotes, são criadas as classes do projeto, nas quais estão os métodos, variáveis, constantes e mais. (PERRY, 2016).

Portanto, a linguagem Java tem como objetivo permitir o desenvolvimento de aplicações portáteis de alto desempenho para a mais ampla variedade possível de plataformas de computação, reduzindo, ainda, todo o custo derivado das aplicações da própria empresa e do consumidor. (JAVA, 2008).

2.6 XML

XML é uma linguagem de marcação que tem como objetivo definir um conjunto de regras para condição de documentos, conteúdo códigos que podem ser aplicados na leitura de dados ou textos feitos por computadores ou pessoas. Assim, a linguagem XML está focada na simplicidade e usabilidade, sendo utilizado para diversos na web. (SOUZA, 2019):

Um arquivo XML é dividido em duas partes: “prolog” e “body”. A parte do “prolog” consiste em metadados administrativos, como declaração XML, instrução de processamento opcional, declaração de tipo de documento e comentários. A parte do “body” é composta por duas partes — estrutural e o conteúdo (presente no texto simples).

Dessa forma, o XML, além disso, tem como objetivo auxiliar os sistemas de informação no compartilhamento de dados, codificando documentos e inserindo seriais nos dados comparando o texto com o de outras linguagens baseadas em serialização. (TECMUNDO, 2009).

3 APLICATIVO DESENVOLVIDO

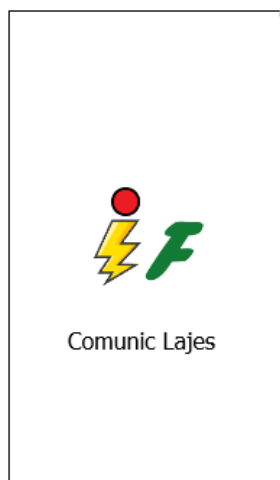
Para a construção do *software*, foi utilizada a IDE *Android Studio*, comumente utilizada para desenvolvimento *Android*. Nesta ferramenta, foi necessário o conhecimento das linguagens de programação Java e a linguagem de marcação XML. Ambas contribuíram para a construção das classes do projeto e montagem dos *layouts*, respectivamente.

3.1 ESBOÇO DA APLICAÇÃO

O primeiro passo para a montagem do aplicativo foi a construção do seu fluxo de telas, bem como a estrutura e modelagem do *layout* de cada uma delas. Para desenvolver o rascunho digital de cada tela, foi usado o programa de edição gráfica *Paint 3D*.

A tela inicial (Tela 1) apresenta a logo do aplicativo, que conta com as iniciais do nome da instituição (“IF”), e abaixo o nome, “Comunic-Lajes”, como descreve a imagem 4:

Fotografia 4 – Tela 1



Fonte: autoria própria
(2019)

Na Tela 2, serão listadas todas as notícias referentes ao Campus Lajes, como é descrito na imagem 5:

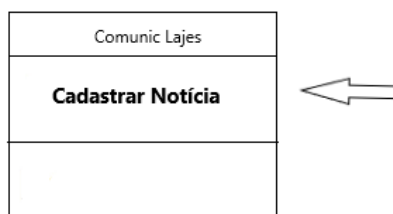
Fotografia 5 – Tela 2

Camunic-Lajes
Alunas da 3º turma do Programa Mulheres Mil colam grau em cerimônia realizada no auditório do Campus Lajes
I EXPOTEC é realizada no Campus Avançado Lajes
NEABI e NAPNE do IFRN Lajes realizam evento sobre "Direitos Humanos e Políticas de Inclusão"

Fonte: autoria própria
(2019)

Além disso, ainda na tela de listagem, o usuário tem a possibilidade de acessar o ícone do menu que contém a opção "Cadastrar Notícia", como está descrito a seguir:

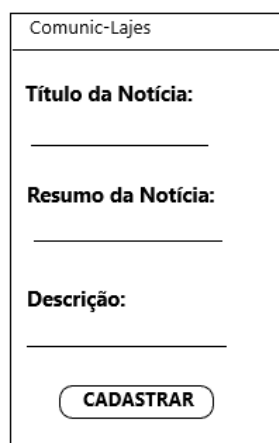
Fotografia 6 – Opção de menu "Cadastrar Notícia"



Fonte: autoria própria
(2019)

Na Tela 3, encontra-se o formulário para cadastro de notícias, que contém as seguintes opções: Título da Notícia, Resumo da Notícia e Descrição da Notícia, como é representado na imagem 7:

Fotografia 7 – Tela 3



Comunic-Lajes

Título da Notícia:

Resumo da Notícia:

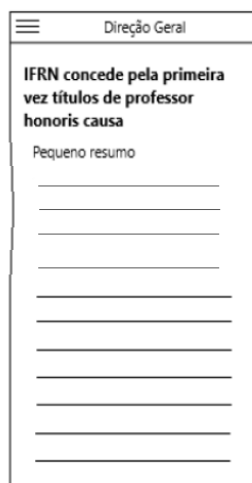
Descrição:

CADASTRAR

Fonte: autoria própria (2019)

Na Tela 4 estará detalhada uma única notícia que foi selecionada pelo usuário na Tela 2 (Listagem). Nela, estará o título, resumo e a descrição da notícia, como está disposto na imagem a seguir:

Fotografia 8 – Tela 4



☰ Direção Geral

IFRN concede pela primeira vez títulos de professor honoris causa

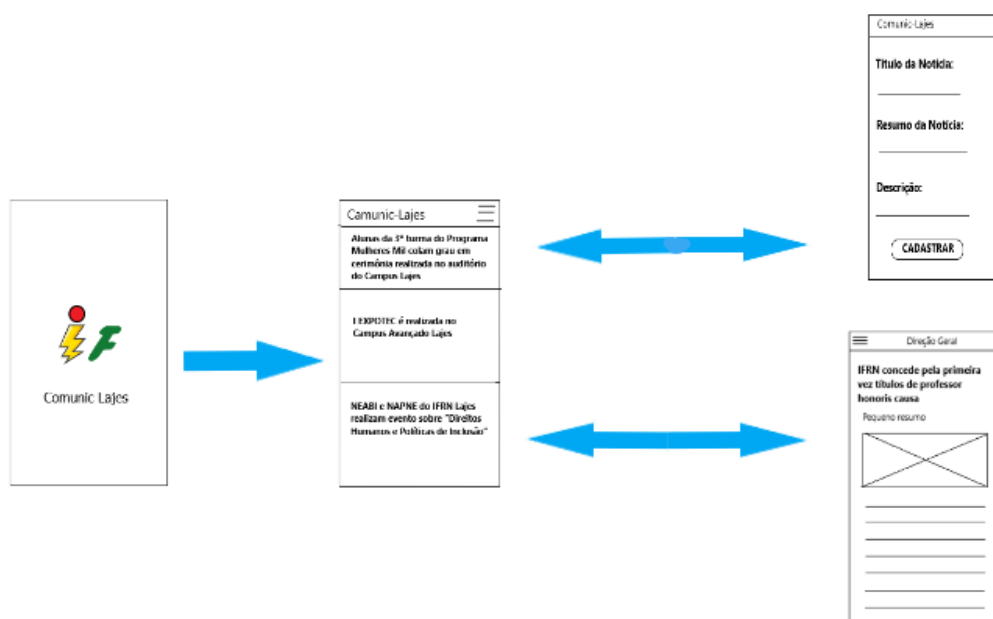
Pequeno resumo

Fonte: autoria própria (2019)

3.2 FLUXO DE TELAS

No fluxo de telas do aplicativo, as telas estão distribuídas de acordo com a sua sequência de acesso, da seguinte maneira:

Fotografia 9 – Fluxo de Telas



Fonte: autoria própria (2019)

Da Tela 1 segue-se para a Tela 2, sem possibilidade de retorno, visto que a tela inicial está programada para ficar ativa durante cinco segundos. Após o fim do tempo determinado, passa-se automaticamente para a Tela 2, na qual estão listadas as notícias. Quanto às Telas 3 (Cadastro de Notícias) e 4 (Detalhamento de Notícias), as setas bidirecionais indicam que, uma vez estando em qualquer uma dessas telas, o usuário pode retornar para a anterior (Listagem de Notícias). É válido destacar que o acesso à Tela 3 se dá pelo ícone presente no *Action Bar* do aplicativo. A Tela 4, por sua vez, pode ser acessada ao clicar em uma notícia.

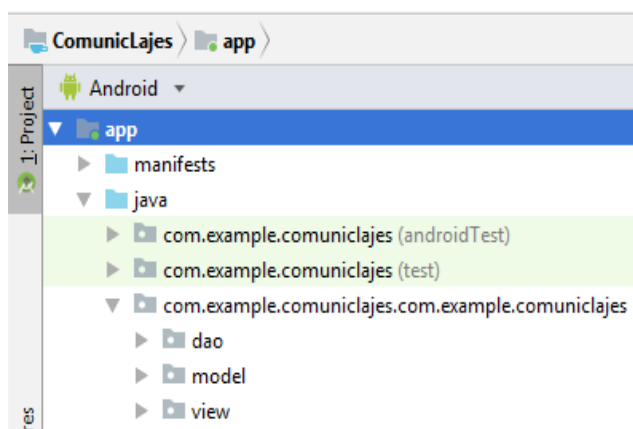
3.3 PACOTES DO PROJETO, CLASSES E MÉTODOS

Neste tópico, serão discutidas as principais estruturas que dão vida e funcionamento ao aplicativo por trás dos *layouts*. Elas foram desenvolvidas com base na linguagem de programação Java, especificamente para o desenvolvimento de aplicações *Android*.

3.3.1 Pacotes do projeto

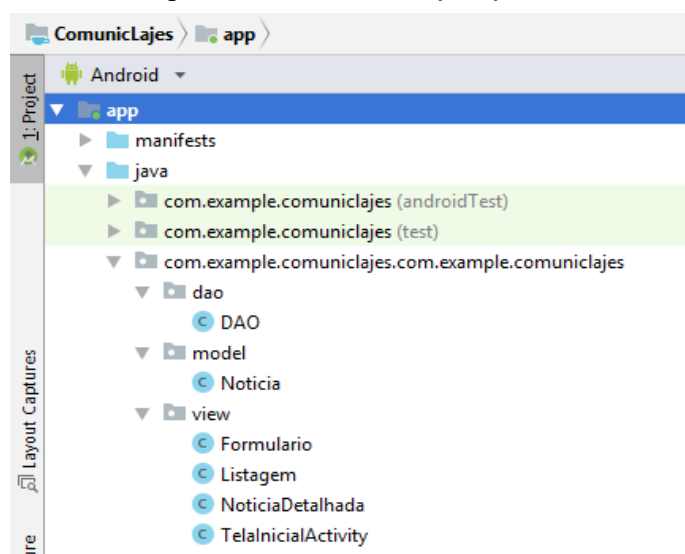
Os pacotes, nos quais estão as classes da aplicação, foram divididos em três tipos: “*dao*”, “*model*” e “*view*”. Esta divisão é semelhante, em partes, ao padrão de arquitetura MVC (*Model*, *View* e *Controller*).

Fotografia 10 – Organização dos pacotes



Fonte: autoria própria (2020)

Fotografia 11 –Classes por pacote



Fonte: autoria própria (2020)

O pacote “*dao*” armazena a classe, de mesmo nome, “*DAO*”, na qual está organizado o banco de dados do projeto. Como o banco o projeto ainda não possui conexão com servidor de banco de dados *MySQL*, então a classe “*DAO*” realiza as suas funções. Nela, estão descritos todos os métodos que representam as funcionalidades do banco dentro da aplicação.

No pacote “*model*”, está a classe Java que é a principal entidade do sistema: “Noticia”. Segundo a arquitetura MVC, na camada modelo estará descrito tudo que representa, de fato, um modelo para a aplicação.

O pacote intitulado “*view*” armazena todas as classes que representam cada tela do aplicativo. Partindo dessa ideia, o próprio nome “*view*”, em português, significa “visualização”. Dessa forma, nesse pacote estão armazenadas todas as classes fundamentais no diz respeito a tudo que é visual (telas) na aplicação.

3.3.2 Classes do projeto

O projeto, em sua totalidade, possui seis classes: “Listagem.java”, “Formulario.java”, “TelaInicialActivity.java”, “NoticiaDetalhada.java”, as quais representam cada *Activity* (tela) da aplicação, e por fim, “Noticia.java” e “DAO.java”, que descrevem a entidade e o banco de dados, respectivamente.

A classe “Formulario.java” representa a tela que cadastra uma nova notícia ao banco:

Fotografia 12 – Classe “Formulario.java”

```
1 package com.example.comuniclajes.com.example.comuniclajes.view;
2
3 import ...
4
5 public class Formulario extends Activity {
6
7     private Noticia noticia = new Noticia();
8
9     @Override
10    public void onCreate(Bundle savedInstanceState) {
11        super.onCreate(savedInstanceState);
12        setContentView(R.layout.activity_cadastro);
13    }
14 }
```

Fonte: autoria própria (2020)

É perceptível que a classe estende a classe “Activity”. Esta herança é necessária visto que “Formulario.java” representa uma tela da aplicação e, para se comportar como tal, necessita de métodos e atributos que estão presentes na classe “Activity”, como por exemplo, o método “onCreate”, necessário para interligar a classe de formulário ao *layout* correspondente.

Ainda na classe “Formulario.java”, para descrever cada componente presente no *layout* “activity_cadastro.xml”, foi feito o seguinte:

Fotografia 13 – Descrição dos componentes do *layout*

```
Button voltar = (Button) findViewById(R.id.voltar);
voltar.setOnClickListener((v) -> {
    Intent intent = new Intent( packageContext: Formulario.this, Listagem.class);
    startActivity(intent);
    finish();
});
Button botao = (Button) findViewById(R.id.botao);
botao.setOnClickListener((v) -> {

    Intent intent = new Intent( packageContext: Formulario.this, Listagem.class);
    startActivity(intent);
    finish();

    EditText titulo = (EditText) findViewById(R.id.titulo);
    EditText resumo = (EditText) findViewById(R.id.resumo);
    EditText descricao = (EditText) findViewById(R.id.descricao);

    noticia.setTitulo(titulo.getEditableText().toString());
    noticia.setResumo(resumo.getEditableText().toString());
    noticia.setDescricao(descricao.getEditableText().toString());

    DAO dao = new DAO( context: Formulario.this);
    dao.inserir(noticia);
    dao.close();

    Toast.makeText( context: Formulario.this, text: "Notícia adicionada!", Toast.LENGTH_LONG).show()
    finish();
});
```

Fonte: autoria própria (2020)

A página de cadastro precisa ter um retorno para a página de listagem, o que pode ser feito por meio da ação de um botão. Isso requer uma comunicação entre as duas *activities*, e na programação *Android* isso deve ser feito passando-se o comando de processamento de uma tela para a outra. A instância da classe “Intent” possibilita isso, fazendo com o processamento seja transferido da atual tela (“Formulario.this”) para a tela seguinte (“Listagem.class”). Isso deve ser feito para que a ação vinculada ao botão de fato funcione, como é demonstrado na imagem a seguir:

Fotografia 14 – Intent

```
no.setOnClickListener((v) -> {
    Intent intent = new Intent( packageContext: Formulario.this, Listagem.class);
    startActivity(intent);
    finish();
    EditText titulo = (EditText) findViewById(R.id.titulo);
```

Fonte: autoria própria (2020)

A página “Listagem.java” representa a tela com a listagem de notícias que são cadastradas:

Fotografia 15 – Classe “Listagem.java”

```
package com.example.comuniclajes.com.example.comuniclajes.view;
import ...
public class Listagem extends AppCompatActivity {
    private Noticia noticia;
    private Noticia noticiaSelecionada;
    private ListView lista;
    private AlertDialog alerta;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.act_lista);
```

Fonte: autoria própria (2020)

Cada notícia é cadastrada no banco de dados DAO. Dessa forma, na tela de listagem, é necessário chamar cada uma dessas notícias que foram cadastradas e atribuí-las a um item da *ListView*. Isso é feito da seguinte maneira:

Fotografia 16 – *ListView*

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.act_lista);

    lista = (ListView) findViewById(R.id.lista);
    carregaLista();
}

```

Fonte: autoria própria (2020)

Portanto, o objeto “lista” é relacionado ao componente XML *ListView* por meio do comando *findViewById*, dentro do qual é chamado o identificador atribuído ao componente.

A classe “*TelaInicialActivity*” representa a tela de abertura do aplicativo, também conhecida como “*Splash Screen*”, que está programada para aparecer durante cinco segundos quando o aplicativo é iniciado:

Fotografia 17 – Classe “*TelaInicialActivity*”

```

public class TelaInicialActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.act_tela_inicial);

        getSupportActionBar().hide();

        getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN, WindowManager.LayoutParams.FLAG_FULLSCREEN);

        new Handler().postDelayed(new Runnable() {
            @Override
            public void run() {
                Intent i = new Intent(packageContext, TelaInicialActivity.this, Listagem.class);
                startActivity(i);

                finish();
            }
        }, delayMillis: 5000);
    }
}

```

Fonte: autoria própria (2020)

Antes de passar o comando de processamento de uma tela para outra, a tela fica exposta durante cinco segundos, como é feito no comando da imagem 18:

Fotografia 18 – *Delay*

```
@Override
public void run() {
    Intent i = new Intent( packageContext: TelaInicialActivity.this, Listagem.class);
    startActivity(i);

    finish();
}
}, delayMillis: 5000);
```

Fonte: autoria própria (2020)

A classe “Notícia.java”, como já descrito anteriormente, representa uma entidade. Nela que estão descritos os atributos que fazem jus a uma notícia propriamente dita (título, resumo e descrição), bem como seus respectivos métodos *getters* e *setters*, descritos nas imagens 20 e 19, respectivamente:

Fotografia 19 – Métodos dos atributos

```
public Long getId() { return id; }

public void setId(Long id) { this.id = id; }

public String getTitulo() { return titulo; }

public void setTitulo(String titulo) { this.titulo = titulo; }

public String getResumo() { return resumo; }

public void setResumo(String resumo) { this.resumo = resumo; }

public String getDescricao() { return descricao; }

public void setDescricao(String descricao) { this.descricao = descricao; }

@Override
```

Fonte: autoria própria (2020)

Fotografia 20 – Atributos da classe Noticia.java

```
public class Noticia extends BaseAdapter {  
  
    private List<Noticia> noticias;  
    private Context context;  
    private Long id;  
    private String titulo;  
    private String resumo;  
    private String descricao;
```

Fonte: autoria própria (2020)

Como *view*, tem-se também a classe *NotíciaDetalhada*, que representa a tela que contém a notícia específica selecionada pelo usuário.

Fotografia 21 – Classe “NoticiaDetalhada”

```
public class NoticiaDetalhada extends AppCompatActivity {  
  
    private Noticia noticia = new Noticia();  
  
    @SuppressWarnings("UseValueOf")  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.noticia_detalhada);
```

Fonte: autoria própria (2020)

Como é perceptível na imagem acima, a classe é direcionada para o arquivo XML *noticia_detalhada*. Este, por sua vez, possui cada um de seus componentes declarados dentro do método *onCreate*, a fim de que sejam expostos no ato de processamento:

Fotografia 22 – Descrição dos componentes

```
noticia = (Noticia) getIntent().getSerializableExtra( name: "NOTICIA");  
  
TextView titulo = (TextView) findViewById(R.id.tituloNoticia);  
TextView resumo = (TextView) findViewById(R.id.resumoNoticia);  
TextView descricao = (TextView) findViewById(R.id.descricaoNoticia);  
  
titulo.setText(noticia.getTitulo());  
resumo.setText(noticia.getResumo());  
descricao.setText(noticia.getDescricao());
```

Fonte: autoria própria (2020)

Além disso, é válido salientar que, para cada objeto *TextView* que está sendo declarado, nas três últimas linhas de código, cada um deles faz uso do método *setText*, que utiliza como parâmetro os atributos do objeto “noticia” (instância da classe *Noticia*), por meio do método *get*, que sempre retornará o próprio objeto. Assim, é possível fazer uso das informações que foram cadastradas na *activity* “Formulario”. Por exemplo, o parâmetro *noticia.getTitulo()* retornará o objeto do tipo *String* “titulo” que foi informado no ato de cadastro de uma notícia.

Por fim, tem-se a classe “DAO”, que simula um servidor de banco de dados. Essa classe estende outra chamada *SQLiteOpenHelper*, que possibilita o uso de métodos *insert*, *delete*, entre outros.

Fotografia 23 – Classe DAO

```
public class DAO extends SQLiteOpenHelper {
    //private Noticia noticiaEscolhida;
    private static final int VERSAO = 1;
    private List<Noticia> lista;
    private static final String TABELA = "Noticia";
    private static final String[] COLS = {"id", "titulo", "resumo", "descricao"};
    private SQLiteDatabase db;
    private DAO dao;
```

Fonte: autoria própria (2020)

Os objetos “TABELA” e “COLS”, representam, respectivamente, a tabela da entidade “Noticia” e as colunas desta (“id”, “titulo”, “resumo”, “descricao”).

3.3.3 Métodos

Neste tópico, serão declarados os métodos essenciais para o desempenho de cada uma das funcionalidades do aplicativo.

3.3.3.1 onItemClick()

Este método é utilizado para direcionar uma ação ou evento quando o usuário clica em algum item da lista:

Fotografia 24 – Método *onItemClick*

```
lista.setOnItemClickListener(new AdapterView.OnItemClickListener() {
    @Override
    public void onItemClick(AdapterView<?> adapter, View view, int position, long id) {
        Noticia noticia = (Noticia) adapter.getItemAtPosition(position);
        Intent intent = new Intent( packageContext: Listagem.this, NoticiaDetalhada.class);
        intent.putExtra( name: "NOTICIA", noticia);
        startActivity(intent);
    }
});
```

Fonte: autoria própria (2020)

Como dito anteriormente, a classe “Intent” possibilita que sejam passadas informações de uma *activity* para outra. O método *getItemAtPosition* faz com que, ao clicar em um item da lista, seja passado o inteiro que representa a posição daquele item. Esse valor é passado como parâmetro do método *putExtra*, por meio do objeto “noticia”, declarado anteriormente.

Portanto, ao realizar selecionar um item da lista, que representa uma notícia, é gerada a posição desse item e por meio do objeto da classe “Intent” são passadas informações da classe “Listagem” para a classe “NoticiaDetalhada”. É válido atentar que, para realizar esse procedimento, a classe “Noticia” precisa implementar a interface *java.io.Serializable*.

3.3.3.2 CarregaLista()

O método “carregaLista()” é responsável, de fato, por exibir a lista na tela toda vez que ela é chamada ou atualizada:

Fotografia 25 – Classe “NoticiaDetalhada”

```
private void carregaLista(){
    DAO dao = new DAO( context: this);
    List<Noticia> noticias = dao.getLista();
    dao.close();

    ArrayAdapter<Noticia> adapter = new ArrayAdapter<>( context: this, android.R.layout.simple_list_item_1, noticias);
    lista.setAdapter(adapter);
}
```

Método *carregaLista* – Figura 18

Dessa forma, sempre que a tela de listagem é chamada pelo usuário, o método se encarrega por buscar a lista de registros que está no banco e organizá-los dentro de um *ArrayAdapter*, no *layout* “*simple_list_item_1*”.

3.3.3.3 ValidarCampos()

Este método é responsável por realizar a validação dos campos do formulário de cadastro. Logo, caso o usuário não insira informação em pelo menos um dos campos, ele será notificado de que deve preenchê-los:

Fotografia 26 – Método *validarCampos()*

```
private boolean validarCampos() {  
  
    boolean retorno = false;  
  
    if (!TextUtils.isEmpty(titulo.getText().toString())) {  
        retorno = true;  
    } else {  
        titulo.setError("*");  
        titulo.requestFocus();  
    }  
  
    if (!TextUtils.isEmpty(resumo.getText().toString())) {  
        retorno = true;  
    } else {  
        resumo.setError("*");  
        resumo.requestFocus();  
    }  
  
    if (!TextUtils.isEmpty(descricao.getText().toString())) {  
        retorno = true;  
    } else {  
        descricao.setError("*");  
        descricao.requestFocus();  
    }  
  
    return retorno;  
}
```

Fonte: autoria própria (2020)

A variável “retorno” assume, inicialmente, o valor *false*. Dentro da condição *if*, por meio da negação estabelecida pelo símbolo de exclamação (!), sabe-se que se o campo “titulo” não estiver vazio (*Empty*), o retorno será *true*. Caso contrário, o *else* estabelece que no campo do título deve ser exibida uma mensagem de erro e em seguida, por meio do método *requestFocus* a linha pertencente àquele campo deve

ficar destacada na cor vermelha, acompanhada de um sinal de exclamação ao lado. A mesma lógica é aplicada para os demais campos. Estes testes estão descritos no capítulo Estudo de Caso.

3.3.3.4 Inserir e getLista

Na classe DAO, o método *inserir* é responsável por registrar as notícias cadastradas no banco de dados, enquanto o método *getLista*, por sua vez, retorna a listagem das notícias que foram cadastradas, com base em cada dos registros da tabela “Noticia”, como está descrito nas imagens 27 e 28, respectivamente:

Fotografia 27 – Método *inserir*

```
public void inserir (Noticia noticia){
    ContentValues values = new ContentValues();
    values.put("titulo", noticia.getTitulo());
    values.put("resumo", noticia.getResumo());
    values.put("descricao", noticia.getDescricao());
    getWritableDatabase().insert(TABELA, nullColumnHack: null, values);
}
```

Fonte: autoria própria (2020)

Fotografia 28 – Método *getLista*

```
public List<Noticia> getLista(){
    List<Noticia> noticias = new ArrayList<>();

    Cursor c = getWritableDatabase().query(TABELA, COLS, selection: null, selectionArgs: null,
    orderBy: null );

    try {

        while (c.moveToNext()) {
            Noticia noticia = new Noticia();
            noticia.setId(c.getLong( columnIndex: 0));
            noticia.setTitulo(c.getString( columnIndex: 1));
            noticia.setResumo(c.getString( columnIndex: 2));
            noticia.setDescricao(c.getString( columnIndex: 3));

            noticias.add(noticia);
        }
    }finally {
        c.close();
    }

    return noticias;
}
```

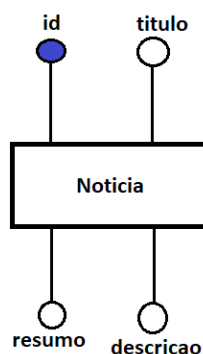
Fonte: autoria própria (2020)

3.4 MODELAGEM DO BANCO DE DADOS

A modelagem do Banco de Dados descreve, por meio de tabelas e diagramas, como os dados estão dispostos internamente no banco. Ainda, nele é possível delinear todos os relacionamentos que ocorrem entre cada uma das entidades existentes, bem como os atributos pertencentes a cada uma delas.

Para o aplicativo, só há uma única entidade, que é “Noticia”. Dessa forma, não é possível estabelecer um relacionamento com outras entidades. O Modelo de Dados Conceitual está representado na imagem a seguir:

Fotografia 29 – Modelo de Dados Conceitual



Fonte: autoria própria (2020)

Como dito anteriormente, há uma única entidade no sistema, que é “Noticia”. Esta, por sua vez, dispõe dos atributos “resumo”, “titulo”, “descricao” e “id”, sendo este último a chave primária ou identificador da tabela do banco intitulada “Noticia”.

A partir do Modelo Conceitual, é possível construir o Modelo Lógico, no qual as informações do Banco estão dispostas na tabela a seguir:

Fotografia 30 – Modelo de Dados Lógico da entidade Noticia

NOTICIA

NOME DO CAMO	TIPO DE DADOS	TAMANHO
id	INT	4
titulo	VARCHAR	20
resumo	VARCHAR	30
descricao	VARCHAR	50

Fonte: autoria própria (2020)

Nas tabelas, estão dispostas três colunas:

- Nome do Campo: onde estão elencados os nomes dos atributos da entidade em questão;
- Tipo de Dados: o tipo primitivo do respectivo atributo. Por exemplo, Inteiro, Caractere, Booleano, Double, Float, entre outros;
- Tamanho: quanto de espaço aquele dado ocupará na memória. Pode ser representado em quantidade de bits ou de caracteres.

Por fim, tem-se o Modelo de Dados Físico, que representa o código *SQL* utilizado para criar a tabela “Noticia” no banco:

Fotografia 31 – Modelo de Dados Físico da entidade Noticia

```
CREATE TABLE NOTICIA  
(  
  id int PRIMARY KEY,  
  titulo VARCHAR (20),  
  resumo VARCHAR (30),  
  descricao VARCHAR (50)  
);
```

Fonte: autoria própria (2020)

4 ESTUDO DE CASO

Inicialmente, é válido destacar que os testes feitos com o aplicativo foram realizados no emulador 3.3 WQVGA API 26, com o sistema operacional Android na versão 8.0. À princípio, tem-se a tela de apresentação do aplicativo, que fica exposta durante cinco segundos antes da próxima tela:

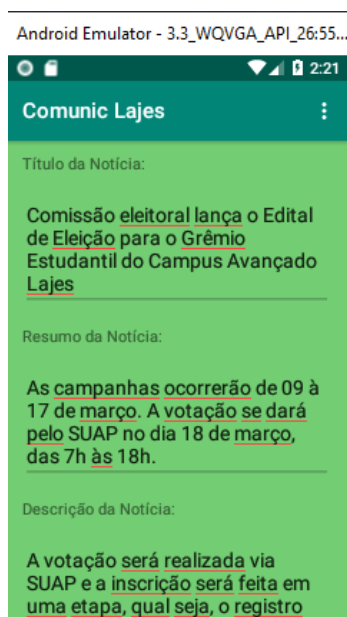
Fotografia 32 – Tela Inicial



Fonte: autoria própria (2020)

O aplicativo possibilita o cadastro de uma notícia, solicitando aos usuários as seguintes informações: o título da notícia, o resumo e a descrição desta. Tudo está disposto na tela como é ilustrado na imagem seguir:

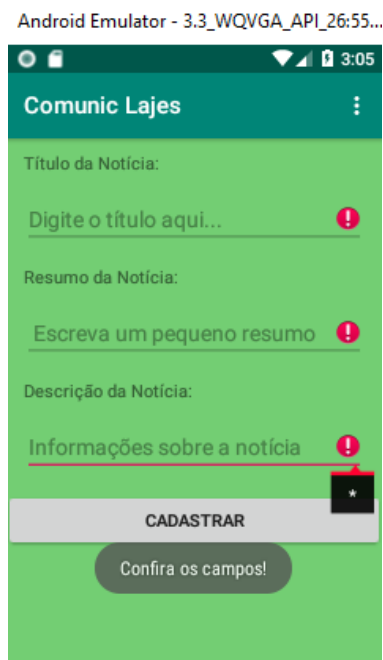
Fotografia 33 – Cadastro de notícias



Fonte: autoria própria (2020)

No ato de cadastro, a fim de que o usuário não cadastre no banco valores nulos, é feita a validação de cada campo, como apresenta a imagem 34:

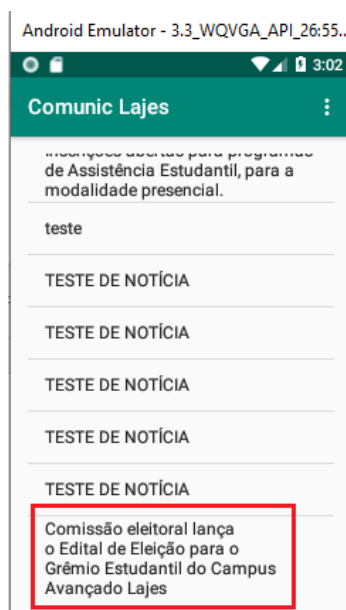
Fotografia 34 – Validação dos campos



Fonte: autoria própria (2020)

A tela de listagem já contava com algumas notícias que já haviam sido cadastradas somente para fins de teste da funcionalidade de cadastro. Assim, ao cadastrar uma notícia, o usuário é direcionando para a listagem, onde só é mencionado o título da notícia, como está ilustrado na imagem 35:

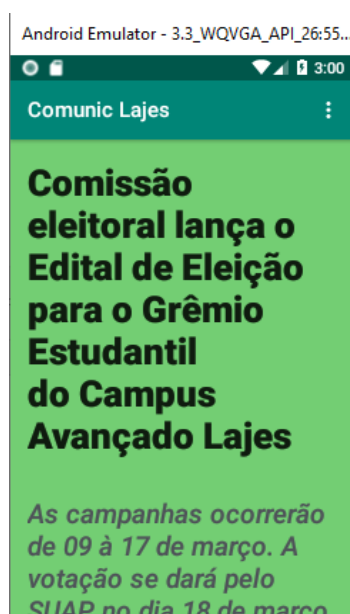
Fotografia 35 – Listagem de notícias



Fonte: autoria própria (2020)

Dessa forma, ao clicar no item de uma notícia, no qual é mencionado somente o título desta, o usuário é direcionando para a página que contém somente as informações acerca da notícia selecionada. Em seguida, aparecem, além do título, o resumo e a descrição da notícia, como na imagem a seguir:

Fotografia 36 – Notícia detalhada



Fonte: autoria própria (2020)

5 CONSIDERAÇÕES FINAIS

Sabe-se que a Informática é uma área com aplicabilidade nos mais diversos ramos, entre eles, a comunicação social. No que tange à questão da comunicação, a produção deste trabalho, teve como motivação o formato de comunicação social do Campus Avançado Lajes. Tal formato de comunicação se fez como principal objeto de estudo, possibilitando que fosse possível delinear os principais pontos para a construção da aplicação *Android*. Sendo assim, o aplicativo Comunic-Lajes é o produto da aplicação dos conhecimentos da Informática como ferramenta tecnológica que auxilia os procedimentos de comunicação social do Campus Lajes, promovendo uma maior praticidade e formalidade no que diz respeito ao acesso dessas informações.

Constata-se que o objetivo geral foi atendido pois, de fato, o trabalho conseguiu atender a meta estabelecida inicialmente: o desenvolvimento do aplicativo *Android* Comunic-Lajes, possibilitando a exposição das notícias do campus. Entretanto, ao longo do processo, o estudo acerca das linguagens de programação e marcação utilizadas se apresentou, inicialmente, como um obstáculo, visto que a programação para desenvolvimento *Android*, especificamente para aplicativos, não foi abordada ao longo do curso. Dessa forma, foi necessário buscar materiais externos, como apostilas, conteúdo *online*, entre outros.

A fim de construir o trabalho proposto, foi utilizada uma metodologia que se fundamentou, inicialmente, em uma pesquisa de caráter exploratório a fim de coletar dados baseados na opinião pública acerca do atual formato de comunicação do campus. Para tanto, foi desenvolvido um formulário via *Google Forms* que foi aplicado, exclusivamente, para a comunidade discente do Campus Avançado Lajes.

Todavia, a pesquisa não representa a opinião da maioria dos estudantes do campus, e sim de uma pequena parcela, visto que ela foi aplicada somente para os

alunos do turno matutino, pois em virtude da disponibilidade e acesso, o questionário não conseguiu alcançar a todos como o esperado.

O estudo acerca das linguagens de programação e marcação Java e XML, respectivamente, foi realizado, possibilitando o desenvolvimento do *front-end* e *back-end* do aplicativo. Dessa forma, foi possível desenvolver cada uma das telas e classes que permitem o funcionamento do *software* como um todo.

Os esboços do aplicativo foram construídos, exemplificando o posicionamento e a estrutura dos botões, das listas, do *layout* das telas e, sobretudo, do ícone presente na tela de abertura do aplicativo. Além do esboço, foi estabelecido o fluxo de telas do aplicativo, que mostra como se dá a transição de uma tela para outra.

As funcionalidades de cadastro, listagem e acesso a uma notícia específica foram atendidas. Dessa forma, o usuário, no aplicativo, pode cadastrar uma notícia, visualizar a listagem de notícias que foram cadastradas e acessar a notícia que desejar. Além disso, o aplicativo conta com uma tela de abertura que fica exposta durante cinco segundos até passar para a tela de listagem. Ainda assim, não há um controle de acesso para cada uma das funções e, dessa forma, todas elas podem ser exercidas por qualquer usuário dentro do aplicativo.

O estudo de caso foi realizado, expondo os testes feitos com o aplicativo. Foi testada a funcionalidade de cadastro de uma notícia e, em seguida, a exposição desta na listagem. Já na listagem, a notícia pode ser clicada, direcionando o usuário para a tela que expõe somente as informações da notícia desejada.

Em suma, o aplicativo, com as suas implementações, ainda se encontra na fase de protótipo, e não foi disponibilizado para testes entre os estudantes do campus, visto que não foi concedido para *download* em *smartphones*. Portanto, isto impossibilitou coletar a satisfação do público acerca do funcionamento do próprio aplicativo.

5.1 TRABALHOS FUTUROS E RECOMENDAÇÕES

Como o aplicativo ainda se encontra na fase de protótipo, a melhoria de algumas funcionalidades é importante para que ele se torne, de fato, um aplicativo mais completo. Recomenda-se que, para trabalhos futuros, sejam implementadas funções que permitam editar e excluir uma notícia em específico. Para tanto, o ideal é

que a edição, a exclusão e o cadastro dessas notícias não possam ser feitos por todos os usuários, e sim por aqueles que possuem a permissão para tal, o que requer a implementação de funções de autorização e autenticação de usuários.

Para a autorização, é necessário a criação de, basicamente, dois perfis de usuário: o estudante e/ou servidor e o administrador, que seria a equipe da assessoria de comunicação do campus, responsável por disponibilizar as notícias para a comunidade acadêmica. Dessa forma, o objetivo é fazer com que os estudantes e servidores possam apenas listar as notícias, enquanto o administrador tem acesso total para cadastrar, excluir, editar e listar as notícias.

Por conseguinte, no que tange a modelagem do banco de dados, terá de ser mais complexa, visto que devem ser criadas, além da entidade “Notícia”, as entidades que representem um usuário comum (aluno e/ou servidor) e o administrador (assessor de comunicação), estabelecendo um novo relacionamento entre elas.

Já em relação ao *design* do aplicativo como um todo também está suscetível a melhorias, podendo se tornar mais dinâmico, aperfeiçoando a interação do usuário com o *layout* proposto. Por exemplo, na opção de cadastro pode ser inserido um ícone que faz referência à ação do botão, que neste caso, é uma nova notícia. Pode ser feito o mesmo para a edição e exclusão das notícias já cadastradas.

Por fim, espera-se que o aplicativo seja disponibilizado para *download*, a fim de que os estudantes, servidores e demais interessados o utilizem como plano principal de acesso às notícias do campus, deixando o *Instagram* e o *Facebook* como plano de contingência. Dessa forma, sempre que o aplicativo estiver em fase de manutenção ou reparo de erros, as notícias podem ser divulgadas pelos outros dois meios de comunicação.

REFERÊNCIAS

- ABLESON, W. Frank et al. **Android em Ação**. 3. ed. Rio de Janeiro: Elsevier, 2012.
- CASTELLS, Manuel. **A Sociedade em Rede**. 8. ed. São Paulo: Paz e Terra, 1999.
- COUTINHO, Gustavo Leuzinger. **A Era dos Smartphones: Um estudo exploratório sobre o uso dos Smartphones no Brasil**. 2014. 67 f. Monografia (Graduação em Publicidade e Propaganda) – Universidade de Brasília. Brasília, 2014.
- CORREIA, Elizabeth Saad. Comunicação digital: uma questão de estratégia e de relacionamento com público. **Organicom**, v. 2, n. 3, 2005. Disponível em: <<http://www.periodicos.usp.br/organicom/article/view/138900>>. Acesso em: 20 set. 2019.
- CALEUM. **Desenvolvimento Moveel com Google Android**. São Paulo, 2013.
- DEVMEDIA. **XML Tutorial**, 2012. Disponível em: <<https://www.devmedia.com.br/xml-tutorial/24936>>. Acesso em: 11 de mar. 2020.
- FILHO, Porfirio Amarilla. Educação e a cultura da informática. **Revista Eletrônica de Educação**, v. 2, n. 1, 2008. Disponível em: <<http://www.reveduc.ufscar.br/index.php/reveduc/article/view/14>>. Acesso em: 20 set. 2019.
- MELLO, C. M. C.; SGANZERLA, R. A. M. Aplicativo Android para auxiliar no desenvolvimento da comunicação de autistas. In: NUEVAS IDEAS EN INFORMÁTICA EDUCATIVA, 18., 2013, Porto Alegre. **Anais eletrônicos...** Porto Alegre: PUCRS, 2013. Disponível em: <<http://www.tise.cl/volumen9/TISE2013/231-239.pdf>>. Acesso em: 20 set. 2019.
- MENEZES, João Paulo. **Protótipo de Sistema para Monitoramento em tempo real de Produtos e Veículos**. 2017. 74 f. Monografia (Graduação em Ciência da Computação) – Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais, Formiga, 2017.
- OLIVEIRA, R. T.; COSTA, R. M. F. Desenvolvimento de aplicativo móvel de referência sobre vacinação no Brasil. **Journal of Health Informatics**. São Paulo, 4 mar. 2012. Caderno A, p. 2.
- OLIVEIRA, R. Y. O Instagram como uma nova ferramenta para estratégias publicitárias. In: CONGRESSO DE CIÊNCIAS DA COMUNICAÇÃO NA REGIÃO NORDESTE, 16., 2014, João Pessoa. **Anais eletrônicos...** João Pessoa: Intercom, 2014. Disponível em: <<http://portalintercom.org.br/anais/nordeste2014/resumos/R42-0112-1.pdf>>. Acesso em: 20 set. 2019.

PEREIRA, M. D.; SILVA, S. G. **As Tecnologias de Informação e Comunicação (TICs) como aliadas para o desenvolvimento**. Vitória da Conquista: Cadernos de Ciências Sociais Aplicadas, 2010. Disponível em: <<http://periodicos2.uesb.br/index.php/ccsa/article/download/1935/1652/>>. Acesso em: 18 set. 2019.

PERRY, STEVEN. **Fundamentos da Linguagem Java**. Disponível em: <<https://www.ibm.com/developerworks/br/java/tutorials/j-introjava1/index.html>>. Acesso em: 11 de mar. 2020.

QUADROS, Mirian Redin. **As redes sociais no jornalismo radiofônico: as estratégias interativas adotadas pelas rádios Gaúcha e CBN**. 2013. 250 f. Dissertação (Programa de Pós-Graduação em Comunicação) – Universidade Federal de Santa Catarina, 2014.

SOUZA, Ivan. **Entenda o que é XML e quais as utilidades dessa linguagem de marcação**. Disponível em: <<https://rockcontent.com/blog/o-que-e-xml/>>. Acesso em: 11 de mar. 2020.