

Márcio Silvano de Sousa  
Orientador: Leonardo Pereira de Sousa

# **ALGOEDUC: um aplicativo móvel para apoio no ensino de algoritmos**

Picos - PI  
22 de Novembro de 2019

Márcio Silvano de Sousa  
Orientador: Leonardo Pereira de Sousa

## **ALGOEDUC: um aplicativo móvel para apoio no ensino de algoritmos**

Monografia apresentada ao Curso de Bacharelado em Sistemas de Informação do Campus Senador Helvídio Nunes de Barros como requisito final na aquisição do grau de Bacharel em Sistemas de Informação.  
Orientador: Prof. M.e Leonardo Pereira de Sousa.

Universidade Federal do Piauí  
Campus Senador Heuvídio Nunes de Barros  
Bacharelado em Sistemas de Informação

Picos - PI  
22 de Novembro de 2019

**FICHA CATALOGRÁFICA**  
**Universidade Federal do Piauí**  
**Campus Senador Helvídeo Nunes de Barros**  
**Biblioteca Setorial José Albano de Macêdo**  
**Serviço de Processamento Técnico**

**S725a** Sousa, Márcio Silvano de.  
ALGOEDUC: um aplicativo móvel para apoio no ensino de algoritmos. / Márcio Silvano de Sousa. -- Picos, PI, 2019.  
43 f.  
CD-ROM: 4 ¾ pol.

Trabalho de Conclusão de Curso (Bacharelado em Sistemas de Informação). – Universidade Federal do Piauí, Picos, 2020.  
“Orientador(A): Prof. Me. Leonardo Pereira de Sousa.”

1. Algoritmos. 2. Aplicativo Móvel. 3. Educação – Ensino - Tecnologia. I. Título.

**CDD 005.1**

*Elaborada por Rafael Gomes de Sousa CRB 3/1163*

ALGOEDUC: UM APLICATIVO MÓVEL PARA APOIO NO ENSINO DE  
ALGORITMOS

MÁRCIO SILVANO DE SOUSA

Monografia APROVADA como exigência parcial para obtenção do  
grau de Bacharel em Sistemas de Informação.

Data de Aprovação

Picos – PI, 29 de NOVEMBRO de 2019

  
Prof. Leonardo Pereira de Sousa

  
Prof. Ismael de Holanda Leal

  
Prof. Alan Rafael Ferreira dos Santos

# Agradecimentos

Agradeço primeiramente a Deus pela vida, e em segundo lugar a toda minha família, em especial meus pais, Joaquim Silvano de Sousa e Elisabete de Sousa, e a meus irmãos, Renato Francisco de Sousa e Francisca Maria de Sousa que sempre estiveram comigo nesta caminhada longa e bastante complicada que está chegando ao fim, com bastante paciência e sem exigir nada demais de mim. Agradeço também a todos os professores sobre os quais adquiri conhecimento para levar nesta nova jornada que com fé em Deus irei continuar. Agradecer bastante também ao meu orientador, Professor Leonardo Pereira de Sousa pelo apoio no desenvolvimento desse projeto e também meus amigos e colegas que sempre estiveram comigo principalmente durante a realização dos testes deste projeto.

*A tarefa não é tanto ver aquilo que ninguém viu, mas pensar o que ninguém ainda  
pensou sobre aquilo que todo mundo vê.*

*Arthur Schopenhauer*

# Resumo

A disciplina de algoritmos é considerada uma das mais importantes dentro dos cursos da área de TI e em outras áreas em que se faz necessário o uso da mesma. Por se tratar de uma disciplina ofertada nos primeiros períodos dos cursos da área de computação, esta se constitui como base para muitas outras em que a informática se aplica. Assim como ensinar o indivíduo sobre os principais conceitos de algoritmos, fator que serve de base para disciplinas mais avançadas de programação. Por não estarem acostumados a trabalharem com fatores que formam o pilar desta matéria, grande parte dos alunos apresentam uma certa dificuldade no aprendizado da mesma. O que acaba refletindo em altos índices de reprovação e, conseqüentemente, no mau desempenho dos alunos em outras matérias que possuem programação como base. O presente trabalho busca investigar as principais dificuldades enfrentadas pelos alunos, bem como desenvolver um aplicativo móvel utilizando *android* que visa auxiliar o ensino-aprendizagem de algoritmos através de atividades realizadas com base nos conceitos teóricos e práticos da disciplina. Os resultados apresentados mostram que a aplicação representa uma ferramenta bastante importante no ensino de algoritmos.

**Palavras-chaves:** Ensino, Algoritmos, Aplicativo Móvel, *Android*.

# Abstract

The discipline of algorithms is considered one of the most important within IT courses and other areas where it is needed. Because it is a discipline offered in the early periods of computer courses, it is the basis for many others in which computer science applies. As well as teaching the individual about the main algorithms concepts, which is the basis for more advanced programming disciplines. Because they are not used to working with factors that form the pillar of this subject, most students have some difficulty learning it. This ends up reflecting high rates of failure and, consequently, and the poor performance of students in other subjects that have programming as a basis. The present work seeks to investigate the main difficulties faced by students, as well as develop a mobile application using android that aims to assist the teaching-learning of algorithms through activities performed based on the theoretical and practical concepts of the discipline. The results presented show that the application represents a very important tool in teaching algorithms.



# Lista de ilustrações

Figura 1 – Participação de mercado dos principais Sistemas Operacionais. . . . .	19
Figura 2 – Pilha de software do android. . . . .	20
Figura 3 – Exemplo de digrama de casos de uso. . . . .	23
Figura 4 – Exemplo de Modelo Entidade Relacionamento utilizando a notação <i>Crows Foot</i> . . . . .	24
Figura 5 – Exemplo da conexão entre Android e MySQL. . . . .	29
Figura 6 – Diagrama de casos de uso do aplicativo AlgoEduc. . . . .	31
Figura 7 – Modelo Entidade Relacionamento do Aplicativo AlgoEduc. . . . .	36
Figura 8 – Tela de login do aplicativo AlgoEduc. . . . .	37
Figura 9 – Tela de cadastro do aplicativo AlgoEduc. . . . .	37
Figura 10 – Tela de alteração de senha do aplicativo AlgoEduc. . . . .	37
Figura 11 – Tela principal do aplicativo AlgoEduc. . . . .	38
Figura 12 – Tela de conteúdos do aplicativo AlgoEduc. . . . .	38
Figura 13 – Tela de conceitos do conteúdo do aplicativo AlgoEduc. . . . .	38
Figura 14 – Tela de exemplo do conteúdo do aplicativo AlgoEduc. . . . .	38
Figura 15 – Tela de questões do aplicativo AlgoEduc. . . . .	38
Figura 16 – Tela de pontuação total do aplicativo AlgoEduc. . . . .	38
Figura 17 – Resultados do primeiro questionário. . . . .	41
Figura 18 – Resultados do segundo questionário. . . . .	41
Figura 19 – Questionário aplicado antes e após utilização do aplicativo. . . . .	47

# Lista de tabelas

Tabela 1 – Comparação entre trabalhos relacionados. . . . .	27
Tabela 2 – Requisitos funcionais do aplicativo AlgoEduc. . . . .	30
Tabela 3 – Requisitos não funcionais do aplicativo AlgoEduc. . . . .	30
Tabela 4 – Regras de negócio do aplicativo AlgoEduc. . . . .	31
Tabela 5 – Documentação do caso de uso cadastro de usuário. . . . .	32
Tabela 6 – Documentação do caso de uso alterar senha. . . . .	32
Tabela 7 – Documentação do caso de uso <i>login</i> . . . . .	33
Tabela 8 – Documentação do caso de uso <i>logout</i> . . . . .	33
Tabela 9 – Documentação do caso de uso acessar tópicos. . . . .	34
Tabela 10 – Documentação do caso de uso acessar conteúdos. . . . .	34
Tabela 11 – Documentação do caso de uso responder questionários. . . . .	34
Tabela 12 – Documentação do caso de uso desbloquear tópicos. . . . .	35
Tabela 13 – Documentação do caso de uso obter pontuação mínima. . . . .	35
Tabela 14 – Documentação do caso de uso visualizar pontuação total. . . . .	36
Tabela 15 – Dispositivos utilizados para teste. . . . .	40
Tabela 16 – Pontuação obtida nas questões utilizando o método SUS. . . . .	42

# Lista de abreviaturas e siglas

TI	Tecnologia da Informação
TIMS	Tecnologias de Informação e Comunicação Móveis e Sem Fio
SO	Sistema Operacional
HTML	<i>Hypertext Markup Language</i>
CSS	<i>Cascading Style Sheets</i>
PHP	<i>Hypertext Preprocessor</i>
API	<i>Application Programming Interface</i>
UML	<i>Unified Modeling Language</i>
JSON	<i>JavaScript Object Notation</i>
SQL	<i>Structured Query Language</i>

# Lista de símbolos

%	Porcentagem
---	-------------

# Sumário

<b>1</b>	<b>Introdução</b>	<b>14</b>
1.1	Objetivos	15
1.1.1	Organização do trabalho	15
<b>2</b>	<b>Referencial Teórico</b>	<b>17</b>
2.1	Algoritmos	17
2.1.1	O ensino de algoritmos	17
2.2	Desenvolvimento de aplicativos móveis	18
2.3	<i>Android</i>	18
2.3.1	Arquitetura <i>Android</i>	19
2.4	Requisitos de um sistema	21
2.5	Diagramas da UML	22
2.5.1	Diagrama de casos de uso	23
2.6	Modelo Entidade Relacionamento	24
2.7	System Usability Scale (SUS)	24
<b>3</b>	<b>Trabalhos Relacionados</b>	<b>26</b>
<b>4</b>	<b>Capítulo de Desenvolvimento</b>	<b>28</b>
4.1	Ferramentas	28
4.2	Modelagem do sistema	29
4.2.1	Requisitos do Sistema	29
4.2.1.1	Requisitos Funcionais	30
4.2.1.2	Requisitos Não Funcionais	30
4.2.1.3	Regras de negócio	30
4.2.2	Diagrama de casos de uso	31
4.2.3	Modelo Entidade Relacionamento	36
4.3	Funcionamento do aplicativo	37
4.4	Testes e avaliação	40
4.4.1	Teste de eficiência	40
4.4.2	Teste de campo do aplicativo AlgoEduc	40
4.4.3	Avaliação de usabilidade	41
<b>5</b>	<b>Conclusão</b>	<b>43</b>
5.0.1	Trabalhos Futuros	43
	<b>Referências</b>	<b>44</b>

<b>Apêndices</b>	<b>46</b>
<b>APÊNDICE A Apêndice . . . . .</b>	<b>47</b>
<b>APÊNDICE B Apêndice . . . . .</b>	<b>51</b>

# 1 Introdução

O mercado de trabalho na área de tecnologia da informação vem crescendo bastante nos últimos anos, e de acordo com [blogipog \(2017\)](#), o mercado de TI exige qualificação. Se especializar é o melhor investimento para aproveitar o crescimento da área, os salários atrativos e várias ofertas de emprego. Diante destes fatos, observa-se que existe uma grande atração de alunos para os cursos de computação no Brasil.

Uma das principais disciplinas ofertadas nos cursos da área de TI e que serve de base para várias outras disciplinas é a disciplina de algoritmos. Que também encontra-se presente nas grades curriculares de outras áreas, como por exemplo, exatas, humanas e biológicas. O principal objetivo do ensino de algoritmos é fazer com que os alunos saibam como desenvolver um algoritmo para resolver problemas que possam ser solucionados com o uso de programação. Segundo [Sousa et al. \(2014\)](#) pode-se definir algoritmos como uma sequência finita, ordenada e não ambígua de passos para solucionar determinado problema ou realizar uma tarefa.

Porém, por se tratar de uma matéria que normalmente é ofertada nos períodos iniciais dos cursos da área de computação. Os alunos enfrentam bastante dificuldade em compreender os conceitos da mesma. Devido essa abranger conteúdos que envolvam o uso da interpretação de textos, raciocínio lógico e problemas matemáticos. O que muitas vezes gera grandes índices de reprovações, trancamento na disciplina e o problema mais grave que é a evasão do curso devido as dificuldades de aprendizagem na mesma.

De acordo com [Barcelos et al. \(2009\)](#) as causas para este alto nível de insucesso não são específicas da área. Pois de modo geral muitos autores apontam que os alunos não apresentam desenvoltura na organização de raciocínios, elaboração de estratégias de resolução de problemas, atenção, concentração, estímulo ao processo de cálculo mental. Sendo assim, as habilidades envolvidas nesse processo, tais como, tentar, observar, conjecturar e deduzir que compõem o que chamamos de raciocínio lógico. Não estando apropriadamente desenvolvidas, interferem na aprendizagem de praticamente todas as áreas do conhecimento, mas em especial afetam a área da computação.

Por esse motivo que, de acordo com [Júnior e Boniati \(2015\)](#) evidencia-se que o ensino de algoritmos requer metodologias diferenciadas. Especialmente no que diz respeito ao desenvolvimento do raciocínio lógico, tão importante para a organização do pensamento. Cada vez mais se percebe a necessidade de adoção de métodos de ensino diferenciados, que possam quebrar algumas barreiras de aprendizado e que possam atrair/reter a atenção e concentração dos alunos.

Atualmente, grande parte desses alunos possuem bastante familiaridade com as novas tecnologias, sendo que uma das que mais vem sendo utilizada nos dias de hoje são os *smartphones*. Onde de acordo com pesquisa realizada no ano de 2019 pela Fundação Getúlio

Vargas de São Paulo (FGV-SP), existem 230 milhões de celulares ativos no Brasil [Estadão \(2019\)](#).

O uso das Tecnologias da Informação e Comunicação Móveis e Sem Fio(TIMS) aumentam os desafios da realidade escolar. Os educadores precisam se adequar a realidade desenhada pelas TIMS. Entre estas, temos o celular, um aparelho popular, com aplicativos que podem vir a ser utilizados em sala de aula como recurso pedagógico [Cristina e Bento \(2009\)](#).

Visando a grande utilização dos alunos com esse tipo de ferramenta e a dificuldade destes no aprendizado de algoritmos, este trabalho tem por objetivo desenvolver uma aplicação *mobile* para o ensino dos principais conteúdos desta. Com o intuito de apoiar os mesmos no aprendizado da disciplina, buscando reforçar os conteúdos apresentados por professores em sala de aula e aplicar métodos que faça com que estes busquem o aprendizado por meio de algo onde eles considerem mais interativo.

## 1.1 Objetivos

O objetivo geral deste trabalho consiste em desenvolver um aplicativo para dispositivos móveis utilizando a linguagem de programação *Android* destinado a auxiliar no ensino-aprendizado dos conteúdos da disciplina de algoritmos.

Este trabalho tem como objetivos específicos:

1. Pesquisar e avaliar métodos existentes para o ensino de algoritmos que possam ser incorporados a aplicação;
2. Desenvolver um aplicativo com base nos conceitos da disciplina de algoritmos de modo a tornar mais interativo o aprendizado da mesma com o uso de um aplicativo;
3. Reduzir as dificuldades dos alunos em entender os principais conceitos da disciplina.

### 1.1.1 Organização do trabalho

Este trabalho está organizado em cinco capítulos. Logo após esta introdução, mais quatro capítulos serão apresentados, sendo que estes tratam respectivamente da revisão bibliográfica, trabalhos relacionados, desenvolvimento e conclusão.

- O capítulo 2 traz uma revisão bibliográfica abordando os principais componentes utilizados para o desenvolvimento do trabalho, permitindo uma compreensão sobre cada um deles.



- O capítulo 3 apresenta os principais trabalhos relevantes e relacionados a este, apresentado o que estes possuem em comum e mostrando o que este trabalho apresenta de diferente em relação aos demais.
- O capítulo 4 apresenta a estrutura do sistema desenvolvido, bem como as etapas para a construção do mesmo e os métodos utilizados para avaliação.
- O capítulo 5 traz a conclusão do trabalho, apresentado os resultados do trabalho e o que será feito com este trabalho no futuro em relação a melhorias que possam trazer uma maior interação na utilização do mesmo.

## 2 Referencial Teórico

### 2.1 Algoritmos

Segundo [Ziviani \(1999\)](#) os algoritmos fazem parte do dia-a-dia das pessoas. As instruções para o uso de medicamentos, as indicações de como montar um aparelho qualquer, uma receita de culinária são alguns exemplos de algoritmos. Um algoritmo pode ser visto como uma sequência de ações executáveis para a obtenção de uma solução para um determinado tipo de problema.

[Cormen et al. \(2002\)](#) afirma que informalmente, um algoritmo é um procedimento computacional bem definido que toma um conjunto de valores como entrada e produz algum conjunto de valores como saída. Portanto, um algoritmo é uma sequência de passos computacionais que transforma a entrada na saída.

Ainda de acordo com [Cormen et al. \(2002\)](#), também podemos visualizar um algoritmo como uma ferramenta para resolver um problema computacional bem especificado. O enunciado do problema especifica em termos gerais o relacionamento entre a entrada e a saída desejada. O algoritmo descreve um procedimento computacional específico para se alcançar esse relacionamento da entrada com a saída.

#### 2.1.1 O ensino de algoritmos

A área de computação vem evoluindo de forma bastante rápida, encontrar profissionais capacitados no mercado ainda é uma tarefa que possui um certo nível de dificuldade. Os cursos da área ainda formam poucas pessoas, sendo que as taxas de evasão ainda são bastante elevadas, onde um dos principais motivos deste alto índice são as dificuldades dos alunos em acompanhar determinadas disciplinas, podendo citar a disciplina de algoritmos como uma delas.

[Santos e Costa \(2006\)](#) afirmam em seu trabalho que, atualmente, o ensino de algoritmos busca nas Ciências Exatas seu pilar de sustentação, pois disciplinas nessa área despertam o raciocínio matemático-lógico para resolução de problemas. A título de exemplo, alguns autores preconizam a necessidade de domínio de habilidades matemáticas prévias ou ao menos integradas, ou pelo menos desejável, sendo considerado até um bom indicador de sucesso para o processo. Para outros, o próprio nível de conhecimento prévio de lógica matemática é discutido. Mas, isso deve ser mais uma das ferramentas para o estudante e não a sua única fonte, pois a sua formação futura será muito ampla.

## 2.2 Desenvolvimento de aplicativos móveis

Atualmente com a grande evolução dos *smartphones*, o desenvolvimento de aplicativos que possam ser executados nesse tipo de dispositivo, está se tornando a cada dia uma grande oportunidade de negócio para desenvolvedores e empresas. Sendo que muito destes são essenciais para o cumprimento de diversas atividades importantes no cotidiano das pessoas.

Existem três tipos de aplicações que podem ser desenvolvidas, a ferramenta desenvolvida neste trabalho consiste de um aplicativo nativo. Mas além deste, serão apresentados os conceitos de *web apps* e aplicativos híbridos para que se possa observar a diferença entre esses tipos, bem como em qual situação determinada aplicação melhor se adequa.

Com base no trabalho desenvolvido por [Prezotto e Boniati \(2014\)](#) será apresentado o conceito desses três tipos de aplicativos *mobile*:

- **Aplicações Nativas:** as aplicações nativas são as que foram desenvolvidas especialmente para aquela plataforma e somente para ela, utilizando para isso as ferramentas disponibilizadas para a mesma, como a linguagem de desenvolvimento, ambiente e emulador.
- **Web App:** além das aplicações nativas de cada plataforma existe também as *Web Apps*. Basicamente são aplicações *Web* que rodam no *browser* do dispositivo, dessa forma não é necessário se preocupar com as diferentes linguagens, pois ela utiliza padrões *web* como HTML, CSS e JavaScript para o lado do cliente e PHP, Java ou outras linguagens para o lado servidor.
- **Híbridas:** estas são aplicações que possuem como finalidade funcionar em qualquer que seja o dispositivo, sendo que para as diferentes plataformas, será utilizado o mesmo código-fonte. Aplicações híbridas ficam instaladas no dispositivo e podem funcionar independentemente de se ter ou não conexão com internet. Partindo do princípio de utilizar uma mesma aplicação para diferentes plataformas, normalmente as aplicações híbridas são desenvolvidas em linguagens *Web* que são interpretadas pelo *browser* nativo do sistema. Possuem tanto características de aplicações nativas como de *web apps*.

## 2.3 Android

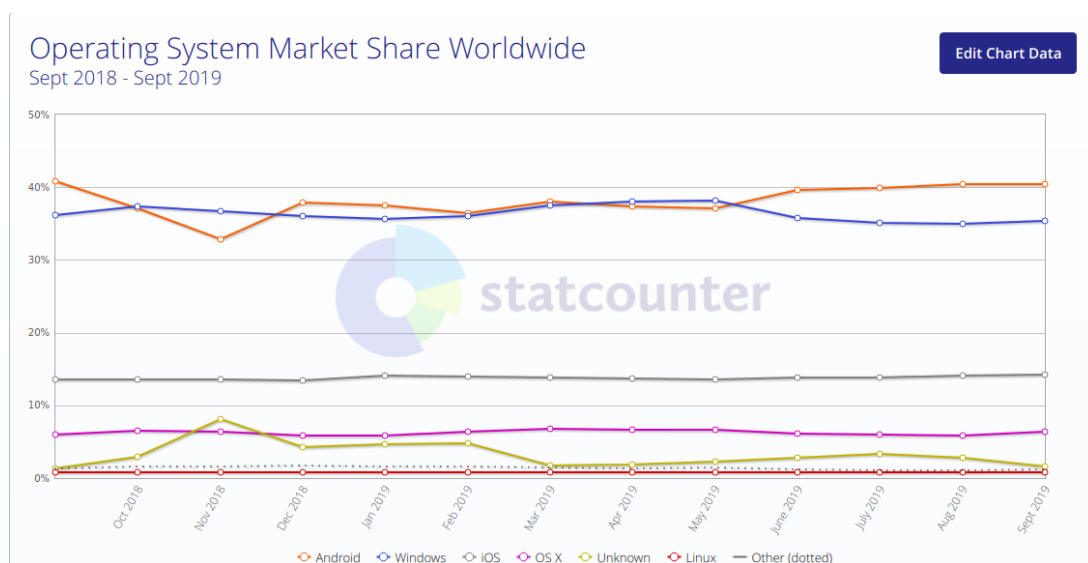
De acordo com [Lecheta \(2013\)](#), o *android* é um sistema operacional para dispositivos móveis desenvolvido pela *google*. O mesmo foi desenvolvido baseado no *linux*, SO de código aberto utilizado em computadores pessoais, o primeiro aparelho celular a utilizar o android foi o *T-Mobile G1* lançado em 2008. Os aplicativos projetados para serem utilizados nesse

tipo de sistema, são desenvolvidos utilizando o java, uma das linguagens de programação mais utilizadas no mundo.

O *Android* causou um grande impacto quando foi anunciado, atraindo a atenção de muita gente. Isso aconteceu porque por trás está nada mais nada menos do que o *Google*, a empresa que está revolucionando a *internet*. Entretanto, não é apenas o *Google* que está na jogada, e sim um grupo formado por empresas líderes do mercado de telefonia como a Motorola, LG, Samsung, Sony Ericson e muitas outras. Esse grupo, chamado de *Open Handset Alliance (OHA)* foi criado com a intenção de padronizar uma plataforma de código aberto e livre para celulares, justamente para atender todas as expectativas e tendências do mercado atual [Lecheta \(2013\)](#).

De acordo com dados do [GlobalStats \(2019\)](#) o *Android* é o sistema operacional com maior participação de mercado em todo o mundo durante o período de setembro de 2018 a setembro de 2019 com 40,44% de participação, passando o *windows* que possui 35.32%. A Figura 1 apresenta os de participação de mercado dos principais Sistemas Operacionais.

Figura 1: Participação de mercado dos principais Sistemas Operacionais.

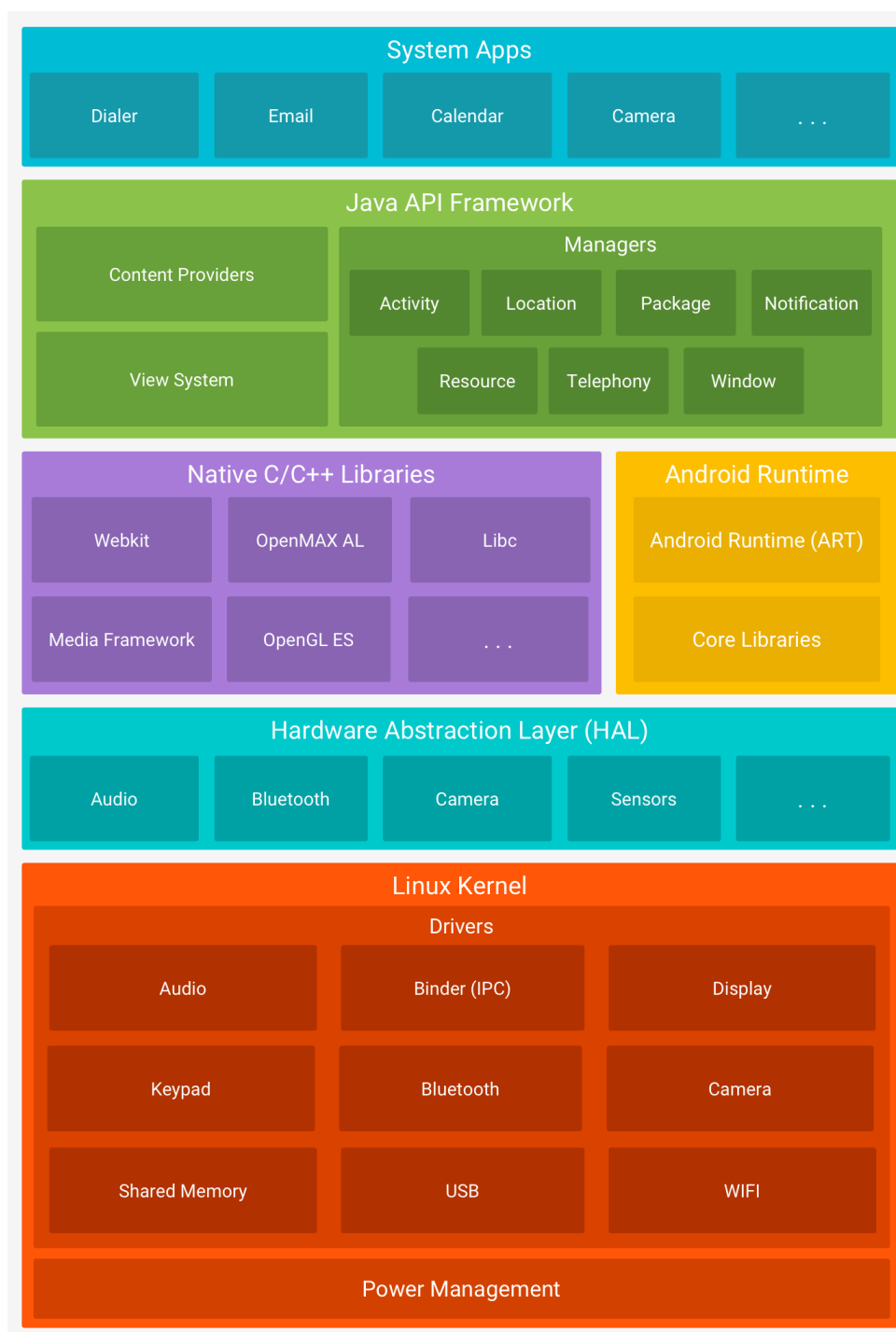


Fonte: [GlobalStats \(2019\)](#)

### 2.3.1 Arquitetura *Android*

Como é colocado em [Developers \(2019\)](#), O *Android* é uma pilha de *software* com base em *Linux* de código aberto criada para diversos dispositivos e fatores de forma. A Figura 2 apresenta as principais camadas do SO.

Figura 2: Pilha de software do android.



Fonte: [Developers \(2019\)](#)

A fundação da plataforma *Android* é o *kernel* do *linux*. Por exemplo: o *Android Runtime (ART)* confia no *kernel* do *Linux* para cobrir funcionalidades como encadeamento e gerenciamento de memória de baixo nível. Usar um *kernel* do *Linux* permite que o *Android* aproveite os recursos de segurança principais e que os fabricantes dos dispositivos desenvolvam *drivers* de *hardware* para um *kernel* conhecido [Developers \(2019\)](#).

A camada de abstração de *hardware (HAL)* fornece interfaces padrão que expõem as

capacidades de *hardware* do dispositivo para a estrutura da Java API de maior nível. A HAL consiste em módulos de biblioteca, que implementam uma interface para um tipo específico de componente de *hardware*, como o módulo de câmera ou *Bluetooth*. Quando uma *Framework API* faz uma chamada para acessar o *hardware* do dispositivo, o sistema *Android* carrega o módulo da biblioteca para este componente de *hardware* [Developers \(2019\)](#).

Para dispositivos com *Android* versão 5.0 (API nível 21) ou mais recente, cada aplicativo executa o próprio processo com uma instância própria do *Android Runtime (ART)*. O *ART* é projetado para executar várias máquinas virtuais em dispositivos de baixa memória executando arquivos DEX, um formato de *bytecode* projetado especialmente para *Android*, otimizado para oferecer consumo mínimo de memória [Developers \(2019\)](#).

Vários componentes e serviços principais do sistema *Android*, como *ART* e *HAL*, são implementados por código nativo que exige bibliotecas nativas programadas em C e C++. A plataforma *Android* fornece as Java *Framework APIs* para expor a funcionalidade de algumas dessas bibliotecas nativas aos aplicativos. Por exemplo, é possível acessar *OpenGL ES* pela *Java OpenGL API* da estrutura do *Android* para adicionar a capacidade de desenhar e manipular gráficos 2D e 3D no seu aplicativo [Developers \(2019\)](#).

O conjunto completo de recursos do SO *Android* está disponível pelas APIs programadas na linguagem Java. Essas APIs formam os blocos de programação que você precisa para criar os aplicativos *Android* simplificando a reutilização de componentes e serviços de sistema modulares e principais [Developers \(2019\)](#).

O *Android* vem com um conjunto de aplicativos principais para *e-mail*, envio de SMS, calendários, navegador de *internet*, contatos etc. Os aplicativos inclusos na plataforma não têm status especial entre os aplicativos que o usuário opta por instalar. Portanto, um aplicativo terceirizado pode se tornar o navegador da *Web*, o aplicativo de envio de SMS ou até mesmo o teclado padrão do usuário (existem algumas exceções, como o aplicativo Configurações do sistema) [Developers \(2019\)](#).

## 2.4 Requisitos de um sistema

De acordo com [Sommerville \(2011\)](#) os requisitos de um sistema são as descrições do que o sistema deve fazer, os serviços que oferece e as restrições a seu funcionamento. Esses requisitos refletem as necessidades dos clientes para um sistema que serve a uma finalidade determinada, como controlar um dispositivo, colocar um pedido ou encontrar informações.

Requisitos de sistema são descrições mais detalhadas das funções, serviços e restrições operacionais do sistema de *software*. O documento de requisitos do sistema (às vezes, chamado especificação funcional) deve definir exatamente o que deve ser implementado. Pode ser parte do con-

trato entre o comprador do sistema e os desenvolvedores de *software* Sommerville (2011).

Durante o processo de projeto de desenvolvimento de *software* os projetistas separam os requisitos classificando-os em dois grupos, os requisitos funcionais e os requisitos não funcionais.

Para Sommerville (2011) os requisitos funcionais descrevem o que um sistema computacional deve fazer. Eles dependem do tipo de *software* a ser desenvolvido, de quem são seus possíveis usuários e da abordagem geral adotada pela organização ao escrever os requisitos. Quando expressos como requisitos de usuário, os requisitos funcionais são normalmente descritos de forma abstrata, para serem compreendidos pelos usuários do sistema. No entanto, requisitos de sistema funcionais mais específicos descrevem em detalhes as funções do sistema, suas entradas e saídas, exceções etc.

Ainda segundo Sommerville (2011) os requisitos não funcionais são requisitos que não estão diretamente relacionados com os serviços específicos oferecidos pelo sistema a seus usuários. Eles podem estar relacionados às propriedades emergentes do sistema, como confiabilidade, tempo de resposta e ocupação de área. Uma alternativa a esse cenário seria os requisitos definirem restrições sobre a implementação do sistema, como as capacidades dos dispositivos de E/S ou as representações de dados usadas nas interfaces com outros sistemas.

## 2.5 Diagramas da UML

Os diagramas da UML tornaram-se ferramentas indispensáveis durante a elaboração do projeto de desenvolvimento de *software*. Trantando-se de uma linguagem para realizar toda a modelagem de como deverá funcionar um determinado sistema computacional, trazendo maior flexibilidade para a equipe no momento do desenvolvimento e melhorando a qualidade do produto final para o cliente.

Silva e Videira (2001) afirmam que a UML é uma linguagem para especificação, construção, visualização e documentação de artefactos de um sistema de *software*. É promovido pelo *Object Management Group (OMG)*, com contribuições e direitos de autoria das seguintes empresas: *Hewlett-Packard*, *IBM*, *ICON Computing*, *i-Logix*, *IntelliCorp*, *Electronic Data Services*, *Microsoft*, *ObjecTime*, *Oracle*, *Platinum*, *Ptech*, *Rational*, *Reich*, *Softeam*, *Sterling*, *Taskon A/S* e *Unisys*.

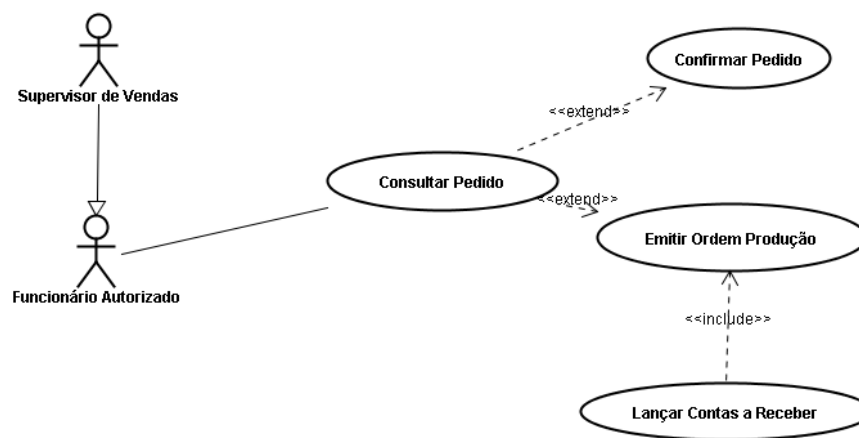
Durante a fase de projeto desse trabalho utilizado o digrama de casos de uso, que será descrito logo abaixo.

### 2.5.1 Diagrama de casos de uso

De acordo [Guedes \(2011\)](#) o Diagrama de Casos de Uso é o diagrama mais geral e informal da UML, utilizado normalmente nas fases de Levantamento e Análise de Requisitos do sistema, embora venha a ser consultado durante todo o processo de modelagem e possa servir de base para outros diagramas. Apresenta uma linguagem simples e de fácil compreensão para que os usuários possam ter uma idéia geral de como o sistema irá se comportar. Procura identificar os atores (usuários, outros sistemas ou até mesmo algum *hardware* especial), que utilizarão de alguma forma o software, bem como os serviços, ou seja, as opções que o sistema disponibilizará aos atores, conhecidas neste diagrama como Casos de Uso.

[Silva e Videira \(2001\)](#) afirmam que o diagrama de casos de uso descreve a relação entre atores e casos de utilização de um dado sistema. Este é um diagrama que permite dar uma visão global e de alto nível do sistema, sendo fundamental a definição correcta da sua fronteira. Ainda de acordo com os autores, estes diagramas são utilizados preferencialmente na fase de especificação de requisitos e na modelação de processos de negócio.

Figura 3: Exemplo de digrama de casos de uso.



Fonte: [Fernando Godóy \(2013\)](#)

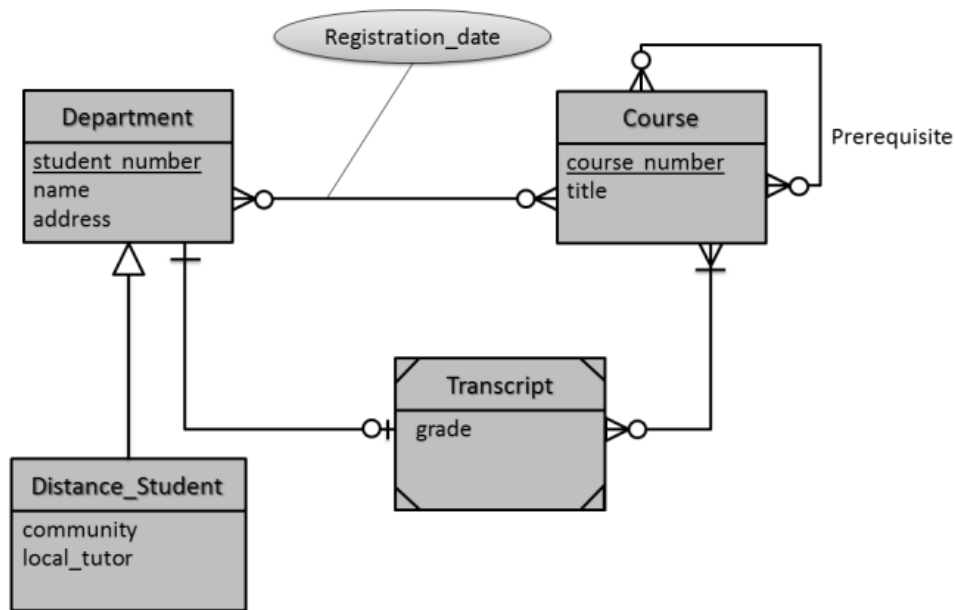
A Figura 3 apresenta um exemplo de utilização de diagrama de casos de uso. Os bonecos representam os atores do sistema, os casos de usos são representados sob a forma de elipse com um rótulo dentro que representa uma funcionalidade do sistema e as linhas sólidas apresentam o relacionamento entre os atores e os casos de uso. As linhas pontilhadas podem representar dois tipos de relacionamento. O de inclusão que é utilizado quando a relação de um caso de uso depende de outro caso de uso para executar uma funcionalidade. E o de extensão onde o caso de uso estendido irá funcionar igual ao caso de uso base, só que alguns passos novos inseridos no caso de uso estendido.



## 2.6 Modelo Entidade Relacionamento

De acordo com [Elmasri e Navathe \(2005\)](#), o modelo entidade relacionamento é um modelo de dados conceitual de alto nível, além de muito popular. Esse modelo e suas variações são normalmente empregados para o projeto conceitual de aplicações de um banco de dados, e muitas ferramentas de projeto de um banco de dados aplicam seus conceitos.

Figura 4: Exemplo de Modelo Entidade Relacionamento utilizando a notação *Crows Foot*.



Fonte: [Rosa et al. \(2012\)](#)

Para o projeto de banco de dados deste trabalho foi utilizado uma variação do modelo entidade relacionamento chamada de notação *crows foot*. Nesta notação, os retângulos representam as entidades do banco, as linhas representam os relacionamentos entre as entidades e os símbolos sobre as linhas representam as cardinalidades. A Figura 4 apresenta um exemplo de Modelo Entidade Relacionamento utilizando a notação *Crows Foot*.

## 2.7 System Usability Scale (SUS)

De acordo com [Fabricio Teixeira \(2015\)](#) o System Usability Scale é um dos mais conhecidos e mais simples métodos de averiguação do nível de usabilidade de um sistema. A popularidade do método se deve, entre outros motivos, ao fato dele apresentar um balanço interessante entre ser cientificamente apurado e ao mesmo tempo não ser extremamente longo para o usuário nem para o pesquisador.

O método foi criado por John Brooke em 1986, e pode ser usado para avaliar produtos, serviços, hardware, software, websites, aplicações [Fabricio Teixeira \(2015\)](#).

O questionário consiste de 10 perguntas, e para cada uma delas o usuário pode responder em uma escala de 1 a 5, onde 1 significa Discordo Completamente e 5 significa Concordo Completamente [Fabricio Teixeira \(2015\)](#).

Depois de colher os resultados, você precisa fazer algumas contas para chegar até a pontuação final [Fabricio Teixeira \(2015\)](#):

- Para as respostas ímpares (1, 3, 5), subtraia 1 da pontuação que o usuário respondeu;
- Para as respostas pares (2 e 4), subtraia a resposta de 5. Ou seja, se o usuário respondeu 2, contabilize 3. Se o usuário respondeu 4, contabilize 1;
- Agora some todos os valores das dez perguntas, e multiplique por 2.5;
- Essa é sua pontuação final, que pode ir de 0 a 100.

A média do System Usability Score é 68 pontos. Se você fez menos pontos do que isso, você provavelmente está enfrentando problemas sérios de usabilidade em seu produto [Fabricio Teixeira \(2015\)](#).

### 3 Trabalhos Relacionados

Nesta seção serão apresentados alguns trabalhos que tem por finalidade resolver um problema semelhante a este, tratando-se de facilitar o aprendizado de algoritmos utilizando como meio algum tipo de ferramenta tecnológica disponível no mercado.

Moreira (2014) desenvolveu uma ferramenta que permite o gerenciamento de turmas e criação de questionários sobre algoritmos. Onde o professor pode cadastrar questões disponibilizando-as de forma pública ou privada, além de poder associar gabarito em diversas linguagens de programação. Aos alunos cadastrados no sistema são apresentados questionários para a avaliação, os questionários anteriormente avaliados com o resultado da avaliação (de cada avaliador) e as questões que ele precisa avaliar.

Oliveira et al. (2014) criaram uma ferramenta onde atribuíram o nome de GrubiBots Educacional que consiste em uma ferramenta para o ensino de algoritmos na educação básica. O GrubiBots Educacional é uma plataforma de desenvolvimento que apresenta um método de programação inovador em relação aos tradicionais métodos de ensino e aprendizagem desta área. Uma vez que, tal ferramenta possibilita a construção de algoritmos a partir da interconexão entre blocos de programação formados por instruções e estruturas de programação representadas por ícones

O trabalho desenvolvido por Santos et al. (2017) consiste em um ambiente de aprendizado sobre o qual os autores atribuíram o nome de *Algorithm.h*. Onde este tem por objetivo permitir a aprendizagem de algoritmos a partir da notação textual de algoritmos, o Portugol.

Amaral et al. (2017) desenvolveram em seu trabalho uma ferramenta para o apoio no ensino de algoritmos para alunos iniciantes chamada de Algo+. O aplicativo possui duas interfaces, uma de coordenação e outra de uso comum, entre aluno e professor. Em ambos os casos, as informações ficam centralizadas no banco de dados próprio. Para o uso do Portal, inicialmente, o professor envia um e-mail para a coordenação solicitando o acesso à ferramenta. Posteriormente, com o acesso liberado, o professor pode criar turmas e colaborar com o repositório de questões. Após essas etapas, o aluno se inscreve em alguma das turmas disponíveis e a matrícula é validada pelo professor responsável. Finalmente, é disponibilizado ao aluno o conteúdo e as avaliações de cada módulo. A ferramenta obtém o resultado da avaliação e o disponibiliza para visualização pelo professor e pelo aluno, nesse caso, na forma de um certificado.

O HelpBlock é uma ferramenta para o ensino de algoritmos que foi desenvolvido com o intuito de ser um ambiente intuitivo e de fácil compreensão por parte dos usuários. Contendo um sistema de cadastros de exercícios classificados por tipo e dificuldade e disponibilizando um local de programação visual todo feito em blocos conectáveis, isso através da utilização da biblioteca *blockly* Gomes (2017).

Com o objetivo de exercitar a prática no planejamento e construção de algoritmos, [Silva e Knop \(2018\)](#) desenvolveram em seu projeto um jogo onde desafiavam o aluno a planejar a movimentação e interação de um personagem através de um labirinto. Esse controle é realizado na programação de ações de um personagem pela ordenação de comandos pré definidos. O jogador é desafiado solucionar as salas do labirinto através da escolha de comandos em uma paleta, buscando eficiência no uso dos comandos.

Tabela 1: Comparação entre trabalhos relacionados.

Trabalhos Relacionados	Principais Conteúdos Teóricos de Algoritmos	Ensino Através de Codificação	Plataforma
<a href="#">Moreira (2014)</a>	Não	Sim	Web
<a href="#">Oliveira et al. (2014)</a>	Não	Sim	Web
<a href="#">Santos et al. (2017)</a>	Não	Sim	Android
<a href="#">Amaral et al. (2017)</a>	Não	Não	Android
<a href="#">Gomes (2017)</a>	Não	Sim	Web
<a href="#">Silva e Knop (2018)</a>	Não	Sim	Android
Este trabalho	Sim	Não	Android

Este trabalho difere-se dos demais por buscar ensinar algoritmos por meio uma metodologia voltada para o ensino dos principais conteúdos teóricos da disciplina. Onde entende-se que é mais importante os alunos aprenderem primeiramente os conteúdos teóricos para só posteriormente começarem a trabalhar com a parte de codificação.

A maioria dos trabalhos relacionados a este, buscam ensinar algoritmos já voltado para a parte de codificação, sendo que para isso é importante que os alunos já possuam uma boa base teórica que existe por trás do desenvolvimento de algoritmos. Somente o trabalho desenvolvido por [Amaral et al. \(2017\)](#) traz uma abordagem mais voltada para teoria, mas este apresenta um ensino por meio de questionários sem uma apresentação de conceitos e exemplos sobre o assunto.

## 4 Capítulo de Desenvolvimento

Este capítulo irá descrever todo o processo de desenvolvimento da aplicação proposta, desde a fase de projeto do sistema até a finalização do mesmo, descrevendo as principais tecnologias utilizadas nesse processo.

### 4.1 Ferramentas

Para o desenvolvimento da aplicação foi utilizado o *Android*, Sistema Operacional utilizado principalmente em *smartphones*, todo o processo de construção do projeto foi feito utilizando o SO *Linux Ubuntu* na versão 18.04 de 64-bit. Para realizar a codificação, foi utilizado o ambiente de desenvolvimento *Android Studio* na versão 3.2.

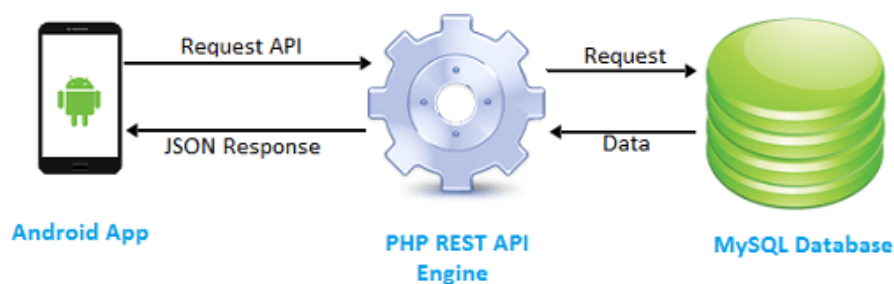
A aplicação deverá armazenar alguns dados, os dados de cadastro de usuário, a pontuação das questões respondidas e a pontuação total de todas as questões. Para realizar a criação do banco de dados foi utilizado o *MySQL*, sistema de gerenciamento de banco de bastante utilizado no mundo.

Por questões de segurança principalmente, o *Android* não permite uma conexão direta entre uma aplicação e um sistema de banco de dados externo. Para realizar tal atividade recomenda-se o desenvolvimento de uma API própria utilizando uma determinada linguagem de programação, onde nesse projeto foi utilizado um *webservice* com PHP.

O PHP fica responsável por receber as requisições do aplicativo android por meio de JSON, que é um formato leve de troca de informações entre sistemas, e devolver algo como resposta. Por outro lado o PHP faz a conexão com servidor de banco de dados por meio de consultas SQL.

Como a aplicação deverá ficar online, para realizar a hospedagem do *webservice* foi utilizado um servidor de hospedagem gratuito chamado *000webhost*. Para gerenciar o banco de dados foi feita a utilização da ferramenta *phpmyadmin*, que é uma plataforma para administração de banco de dados *MySQL web* livre. A Figura 5 apresenta o processo de troca de dados entre *Android* e *MySQL*.

Figura 5: Exemplo da conexão entre Android e MySQL.



Fonte: [Ramos De Souza Janones \(2019\)](#)

## 4.2 Modelagem do sistema

Nesta seção serão apresentados os modelos que foram criados durante a fase de projeto com o intuito de explicar as características e o comportamento do aplicativo desenvolvido.

### 4.2.1 Requisitos do Sistema

Com intuito de buscar as principais funcionalidades a serem utilizadas na aplicação, foi feito o processo de levantamento de requisitos, onde a partir das informações levantadas nessa fase o desenvolvimento do projeto ocorresse de forma mais fácil e ágil. Nesta seção serão descritos os requisitos funcionais, não funcionais e as regras de negócio.

## 4.2.1.1 Requisitos Funcionais

Tabela 2: Requisitos funcionais do aplicativo AlgoEduc.

Identificador	Descrição	Depende de
RF 01	Cadastrar dados do usuário.	RFN01
RF 02	Alterar senha do usuário.	RF01 RFN01
RF 03	O usuário deve conseguir acessar os tópicos desbloqueados.	RF01 RFN02 RFN05
RF 04	Cadastrar pontos de cada tópico.	RF01 RFN01
RF 05	O sistema deverá desbloquear os tópicos de acordo com a pontuação do tópico anterior.	RF01 RF04
RF 06	O usuário deve conseguir acessar os conteúdos dos tópicos desbloqueados.	RF01 RF03 RF04 RF05 RFN05
RF 07	O usuário deve conseguir responder os questionários de cada conteúdo.	RF01 RF03 RF04 RF05 RF06 RFN05
RF 08	O sistema deverá apresentar mensagens ao final dos questionários de cada tópico.	RF01 RF03 RF04 RF05 RF06 RF 07
RF 09	O sistema deverá cadastrar a pontuação total das questões respondidas.	RF01 RF03 RF04 RF05 RF06 RF07 RFN01
RF 10	O usuário poderá visualizar a pontuação total das questões respondidas.	RF01 RF09 RFN05

## 4.2.1.2 Requisitos Não Funcionais

Tabela 3: Requisitos não funcionais do aplicativo AlgoEduc.

Identificador	Descrição	Categoria	Depende de
RFN01	Consistência e persistência dos dados sendo feita por meio de um SGBD.	Manutenibilidade	
RFN02	Acesso ao aplicativo por meio de login e senha.	Segurança de acesso	RF01
RFN03	Sigilo das informações armazenadas no aplicativo.	Segurança	RF01
RFN04	Suporte a dispositivos com android a partir da versão 4.0 e diferentes tamanhos de tela.	Portabilidade	
RFN05	O aplicativo deve ser de fácil utilização, visando um fácil aprendizado em relação as funcionalidade do mesmo.	Usabilidade	
RFN06	O aplicativo deve estar disponível 24 horas por dia.	Disponibilidade	

## 4.2.1.3 Regras de negócio

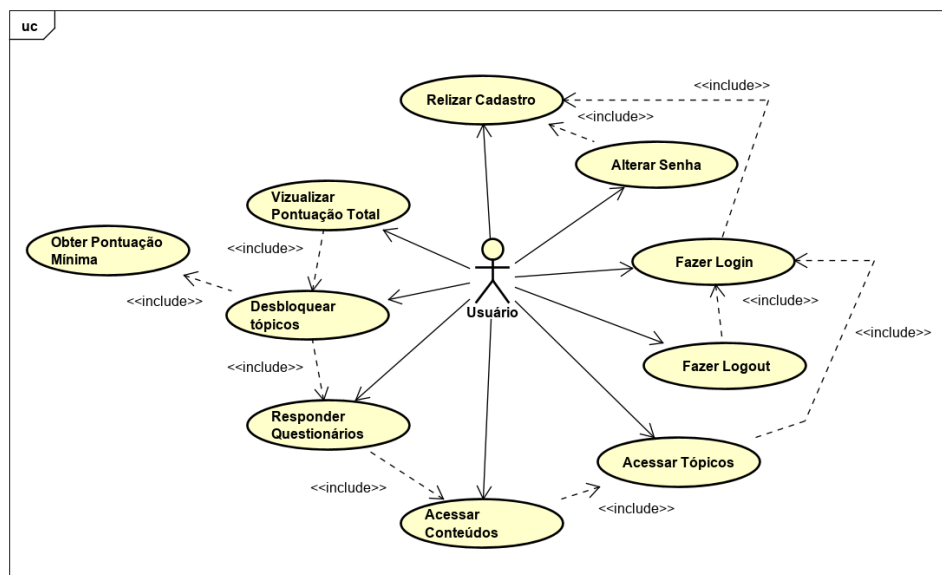
As regras de negócio, como o próprio nome indica, são instruções ou restrições específicas sobre como determinadas ações devem ser executadas dentro de um sistema. A Tabela 4 apresenta as principais regras de negócio presentes no aplicativo.

Tabela 4: Regras de negócio do aplicativo AlgoEduc.

Identificador	Descrição	Depende de
RN01	O usuário deverá possuir conexão com a internet para usar o aplicativo.	
RN02	O usuário só terá acesso as funcionalidades do aplicativo após realizar cadastro.	RF01 RFN01
RN03	A senha deverá conter no mínimo 8 caracteres.	
RN04	O próximo tópico só será desbloqueado caso o usuário atinja uma pontuação mínima no tópico anterior.	
RN05	A pontuação final só será exibida após a conclusão de todos os tópicos.	

### 4.2.2 Diagrama de casos de uso

Figura 6: Diagrama de casos de uso do aplicativo AlgoEduc.



Fonte: O autor

A Figura 6 traz o diagrama de casos de uso, que apresenta uma relação entre o usuário do aplicativo e suas principais funcionalidades, desenvolvido durante a fase de projeto do mesmo.

Os casos de uso apresentados na Figura 6 serão melhor descritos nas tabelas abaixo. As funcionalidades da aplicação serão detalhadas seguindo uma ordem de execução do sistema.



Tabela 5: Documentação do caso de uso cadastro de usuário.

Nome do Caso de Uso	Realizar Cadastro
Caso de Uso Geral	
Ator Principal	Usuário
Ator Secundário	
Resumo	Apresenta os dados necessários para o usuário realizar o cadastro.
Pré-Condições	
Pós-Condições	
Fluxo Principal	
<b>Ações do Autor</b>	<b>Ações do Sistema</b>
	1. O aplicativo solicita os dados do usuário a serem cadastrados.
2. O usuário insere os dados.	
	3. Verifica se já existe usuário cadastrado com o mesmo email.
	4. O aplicativo insere os dados no banco de dados.
Restrições/Validações	Os dados do usuário só são inseridos caso o usuário não tenha sido cadastrado anteriormente.

Tabela 6: Documentação do caso de uso alterar senha.

Nome do Caso de Uso	Alterar Senha
Caso de Uso Geral	
Ator Principal	Usuário
Ator Secundário	
Resumo	Apresenta os dados necessários para o usuário alterar a senha.
Pré-Condições	Possuir cadastro no aplicativo.
Pós-Condições	
Fluxo Principal	
<b>Ações do Autor</b>	<b>Ações do Sistema</b>
	1. O aplicativo solicita os dados do usuário para realizar a alteração.
2. O usuário insere os dados.	
	3. Verifica se existe usuário cadastrado com o email informado.
	4. O aplicativo altera a senha no banco de dados.
Restrições/Validações	A senha do usuário só será alterada caso o usuário possua cadastro.

Tabela 7: Documentação do caso de uso *login*.

Nome do Caso de Uso	<i>Login</i>
Caso de Uso Geral	
Ator Principal	Usuário
Ator Secundário	
Resumo	O usuário insere seus dados de cadastro e ao realizar <i>login</i> o mesmo terá acesso as funcionalidades do sistema.
Pré-Condições	Possuir cadastro no aplicativo.
Pós-Condições	
Fluxo Principal	
<b>Ações do Autor</b>	<b>Ações do Sistema</b>
	1. O aplicativo solicita os dados do usuário para realizar <i>login</i> .
2. O usuário insere os dados de <i>login</i> .	
	3. Verifica se os dados informados correspondem aos armazenados no banco de dados.
	4. O aplicativo libera o acesso as funcionalidades do sistema.
Restrições/Validações	Caso o usuário informe dados que não correspondem aos armazenados no sistema, não irá efetuar <i>login</i>

Tabela 8: Documentação do caso de uso *logout*.

Nome do Caso de Uso	<i>Logout</i>
Caso de Uso Geral	
Ator Principal	Usuário
Ator Secundário	
Resumo	O usuário após realizar <i>login</i> poderá sair do sistema efetuando <i>logout</i> .
Pré-Condições	Ter efetuado <i>login</i> no aplicativo.
Pós-Condições	
Fluxo Principal	
<b>Ações do Autor</b>	<b>Ações do Sistema</b>
1. O usuário realiza <i>login</i> no sistema.	
2. O usuário clica no botão sair para realizar <i>logout</i> .	
	3. Fecha a tela com as funcionalidades.
Restrições/Validações	

Tabela 9: Documentação do caso de uso acessar tópicos.

Nome do Caso de Uso	Acessar Tópicos
Caso de Uso Geral	
Ator Principal	Usuário
Ator Secundário	
Resumo	O usuário após realizar <i>login</i> poderá acessar os tópicos desbloqueados.
Pré-Condições	Ter efetuado <i>login</i> no aplicativo.
Pós-Condições	
Fluxo Principal	
<b>Ações do Autor</b>	<b>Ações do Sistema</b>
1. O usuário realiza <i>login</i> no sistema.	
	2. Apresenta a tela com os tópicos
3. Acessa os tópicos desbloqueados.	
Restrições/Validações	O usuário só poderá acessar um tópico caso este esteja desbloqueado.

Tabela 10: Documentação do caso de uso acessar conteúdos.

Nome do Caso de Uso	Acessar Conteúdos
Caso de Uso Geral	
Ator Principal	Usuário
Ator Secundário	
Resumo	O usuário após acessar um tópico poderá visualizar os conteúdos presentes nesse.
Pré-Condições	Ter acessado um tópico desbloqueado.
Pós-Condições	
Fluxo Principal	
<b>Ações do Autor</b>	<b>Ações do Sistema</b>
1. Abre um tópico desbloqueado.	
	2. Apresenta os conteúdos do tópico.
3. Acessa os conteúdos do tópico.	
Restrições/Validações	

Tabela 11: Documentação do caso de uso responder questionários.

Nome do Caso de Uso	Responder Questionários
Caso de Uso Geral	
Ator Principal	Usuário
Ator Secundário	
Resumo	O usuário após acessar e visualizar um conteúdo terá de responder as questões.
Pré-Condições	Ter acessado um conteúdo.
Pós-Condições	
Fluxo Principal	
<b>Ações do Autor</b>	<b>Ações do Sistema</b>
1. Acessa um conteúdo.	
	2. Apresenta as questões.
3. Responde as questões.	
Restrições/Validações	

Tabela 12: Documentação do caso de uso desbloquear tópicos.

Nome do Caso de Uso	Desbloquear Tópicos
Caso de Uso Geral	
Ator Principal	Usuário
Ator Secundário	
Resumo	O usuário após responder todas as questões dos conteúdos de um tópico deverá desbloquear o próximo tópico.
Pré-Condições	Ter respondido todas as questões do tópicos anterior.
Pós-Condições	
Fluxo Principal	
<b>Ações do Autor</b>	<b>Ações do Sistema</b>
1. Responde as questões.	
	2. Apresenta a pontuação das respostas.
	3. Desbloqueia o tópico.
Restrições/Validações	O tópico só será desbloqueado caso o usuário obtenha uma pontuação mínima determinada para cada tópico.

Tabela 13: Documentação do caso de uso obter pontuação mínima.

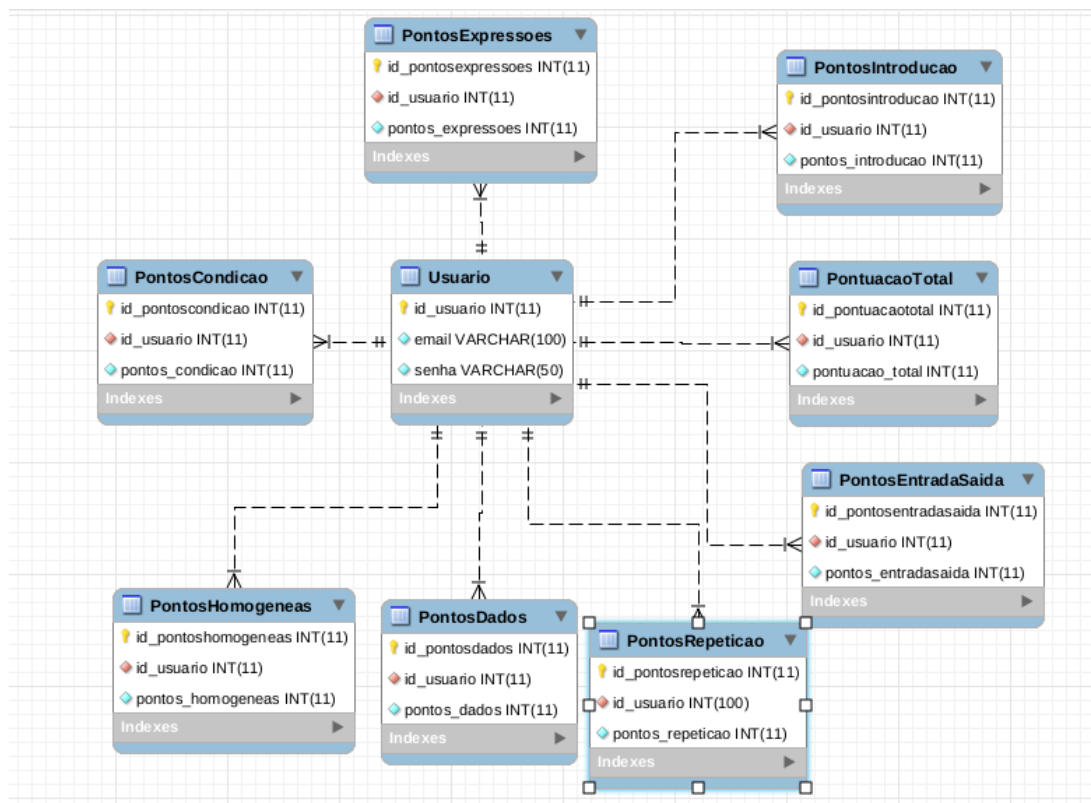
Nome do Caso de Uso	Obter Pontuação Mínima
Caso de Uso Geral	
Ator Principal	Usuário
Ator Secundário	
Resumo	O usuário após responder todas as questões deverá obter uma pontuação mínima determinada pelo sistema para desbloquear o próximo tópico.
Pré-Condições	Ter respondido todas as questões.
Pós-Condições	
Fluxo Principal	
<b>Ações do Autor</b>	<b>Ações do Sistema</b>
1. Responde as questões.	
	2. Determina uma pontuação mínima para o tópico.
	3. Desbloqueia o próximo tópico.
Restrições/Validações	

Tabela 14: Documentação do caso de uso visualizar pontuação total.

Nome do Caso de Uso	Vizualisar Pontuação Total
Caso de Uso Geral	
Ator Principal	Usuário
Ator Secundário	
Resumo	O usuário após realizar as atividades de todos os tópicos poderá vizualisar a pontuação total obtida.
Pré-Condições	Ter realizado todas as atividades de todos os tópicos.
Pós-Condições	
Fluxo Principal	
<b>Ações do Autor</b>	<b>Ações do Sistema</b>
1. Realiza todas as atividades de todos os tópicos.	
	2. Calcula a pontuação total obtida.
	3. Mostra a pontuação total.
Restrições/Validações	

### 4.2.3 Modelo Entidade Relacionamento

Figura 7: Modelo Entidade Relacionamento do Aplicativo AlgoEduc.



Fonte: O autor

A Figura 7 apresenta o Diagrama Entidade Relacionamento do banco de dados da aplicação, neste são apresentadas as tabelas criadas, bem como os atributos da mesma, e a relação entre estas que se relacionam principalmente com a tabela de usuários.

De acordo com a imagem observa-se que o banco de dados da aplicação apresenta 9 tabelas, uma para armazenar os dados do usuário, 7 para guardar a pontuação obtida pelo usuário em cada um dos tópicos e por fim uma tabela para armazenar a pontuação total obtida nos questionários.

### 4.3 Funcionamento do aplicativo

Ao acessar o aplicativo, o usuário irá se deparar com a tela de login mostrada na figura 8, onde nessa irá conter os campos para inserir o *email* e senha cadastrada e realizar login no aplicativo. Caso o usuário não possua cadastro este deverá clicar sobre a opção CADASTRE-SE ou caso queira alterar a senha ir na opção ALTERAR SENHA.

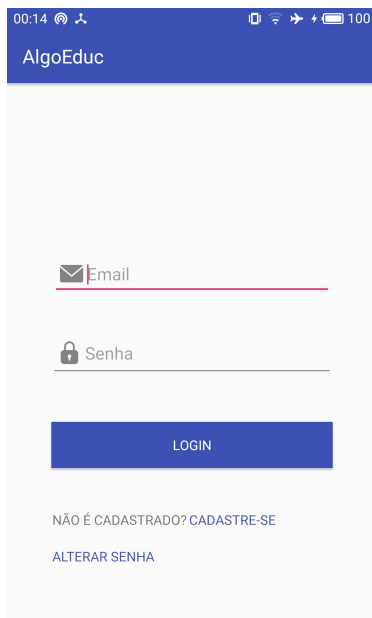


Figura 8: Tela de login do aplicativo AlgoEduc.

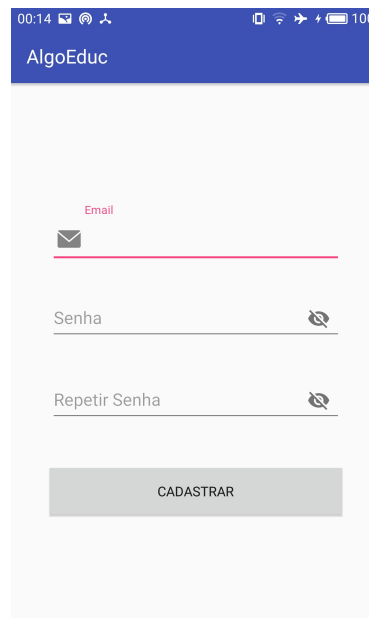


Figura 9: Tela de cadastro do aplicativo AlgoEduc.

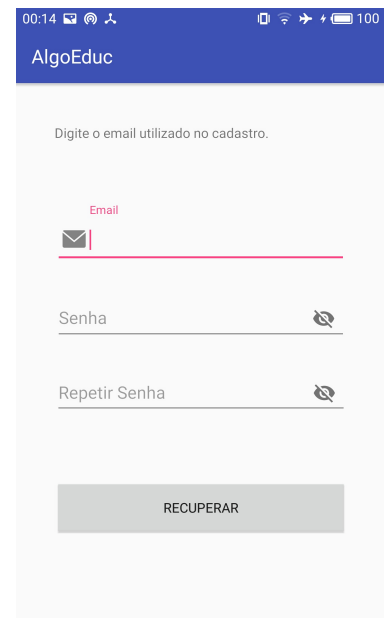


Figura 10: Tela de alteração de senha do aplicativo AlgoEduc.

Para realizar o cadastro no aplicativo, o usuário deverá inserir os dados nos campos exigidos pelo aplicativo e selecionar o botão CADASTRAR, como é apresentado na Figura 9.

Caso o usuário deseje alterar sua senha de cadastro este deverá preencher os campos apresentados na tela de alteração de senha e selecionar o botão RECUPERAR para realizar tal operação, como é mostrado na Figura 10.

Após realizar login no aplicativo o usuário será direcionado para a página principal da aplicação, nela contém os tópicos a serem estudados, como podemos ver na Figura 11. Ao acessar pela primeira vez, somente o tópico introdução estará desbloqueado, para desbloquear os demais o usuário deverá obter uma determinada pontuação nos questionários de cada tópico.

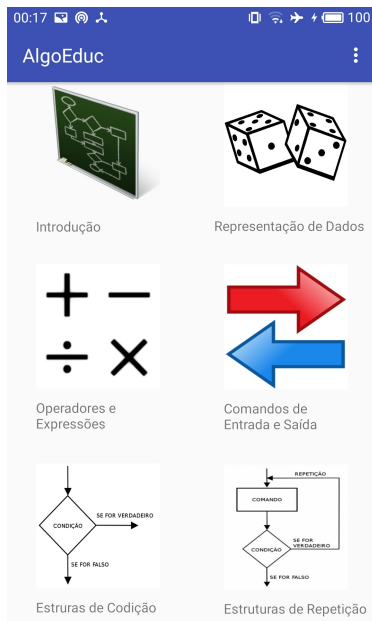


Figura 11: Tela principal do aplicativo AlgoEduc.



Figura 12: Tela de conteúdos do aplicativo AlgoEduc.

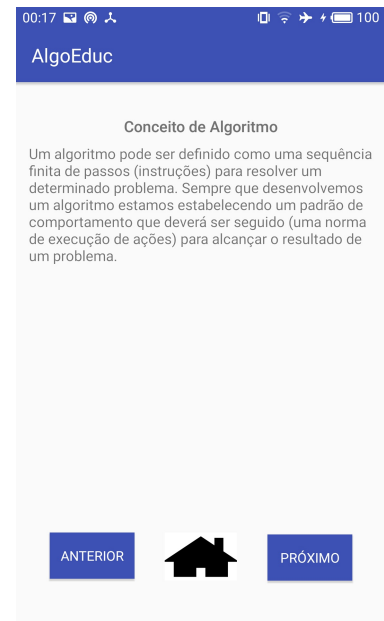


Figura 13: Tela de conceitos do conteúdo do aplicativo AlgoEduc.

Ao selecionar um determinado tópico o usuário será redirecionado para a página com os conteúdos a serem estudados naquele tópico, como é apresentado na Figura 12. Quando for selecionado um conteúdo ao clicar no botão iniciar, será apresentado uma tela com conceito sobre aquele conteúdo, como mostra a Figura 13. Ao selecionar o botão PRÓXIMO será aberto uma tela com um exemplo sobre o conteúdo abordado onde pode ser visto na Figura 14.

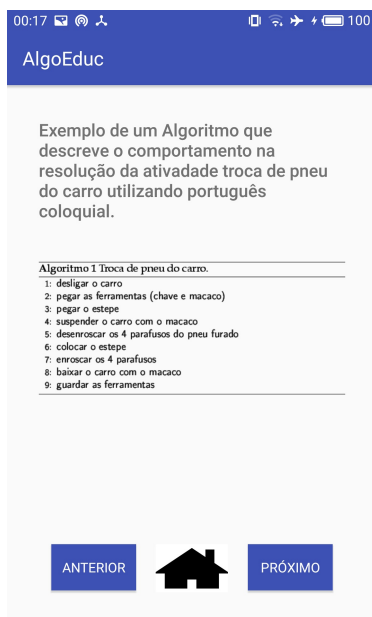


Figura 14: Tela de exemplo do conteúdo do aplicativo AlgoEduc.

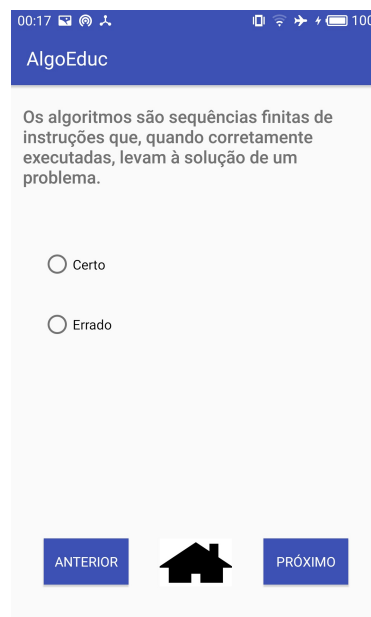


Figura 15: Tela de questões do aplicativo AlgoEduc.



Figura 16: Tela de pontuação total do aplicativo AlgoEduc.

Selecionando o botão PRÓXIMO na tela apresentada na Figura 14, o usuário será

direcionado para as telas com as questões, como mostra a Figura 15. A quantidade de questões dos conteúdos variam de no mínimo 3 questões a no máximo 5 questões.

Para desbloquear o próximo tópico o usuário deverá obter uma pontuação mínima estabelecida para cada tópico, de todas as questões respondidas naquele tópico.

Após ter passado por todos os tópicos, o usuário poderá verificar sua pontuação total obtida ao responder todos os questionários do aplicativo como pode ser visto na Figura 16.



## 4.4 Testes e avaliação

Nesta seção serão apresentados os testes que foram realizados para verificar a eficiência da aplicação com relação ao desempenho no uso bem como na parte de ensino. Será apresentado também a aplicação do método realizado para avaliação da usabilidade do aplicativo.

### 4.4.1 Teste de eficiência

O aplicativo AlgoEduc foi testado em vários dispositivos com configurações diferentes, com o intuito de analisar o funcionamento deste perante distintas disposições de recursos. A tabela apresenta alguns desses dispositivos e suas respectivas configurações.

Tabela 15: Dispositivos utilizados para teste.

Modelo	Versão do SO	Processador	Memória RAM
Meizu MX4	Android 5.1	8 Core 2.0 GHZ	2 GB
Moto G4	Android 7.0	8 Core 1.4 GHZ	2 GB
Xiaomi Redmi Note 7	Android 9 MIUI 10 Pie	8 Core 2.0 GHZ	3 GB
Xiaomi Mi 9 SE	Android 9 MIUI 10 Pie	8 Core 2.4 GHZ	6 GB

O aplicativo funcionou perfeitamente com os dispositivos testados, com relação as diferentes configurações de telas, principalmente o tamanho, a aplicação não apresentou nenhuma diferença muito grande de um dispositivo para outro. O mesmo foi desenvolvido para aparelhos com *Android* a partir da versão 4.0.

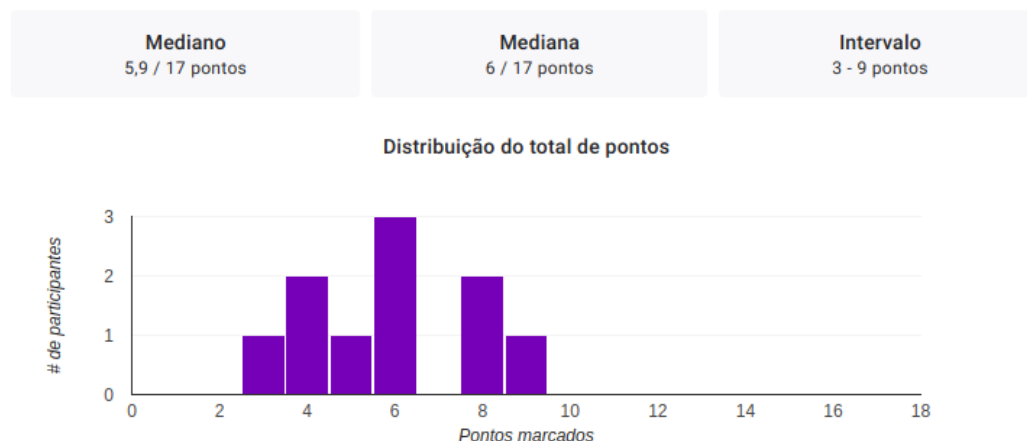
### 4.4.2 Teste de campo do aplicativo AlgoEduc

Com o intuito de levantar o nível de conhecimento das pessoas em relação a algoritmos, foi realizado um questionário com 17 questões (Apêndice A) sobre o tema a ser aprendido utilizando o *google forms*, ferramenta da *google* que permite a criação de questionários para pesquisa.

O teste foi feito com 10 pessoas com pouco ou nenhum conhecimento a respeito do assunto, onde a média de acertos das questões respondidas foi bem baixa, apenas 6 pontos de 17, sendo que o intervalo de acertos foi de 3 a 9 pontos. A Figura 17 apresenta os resultados obtidos no primeiro questionário.

Figura 17: Resultados do primeiro questionário.

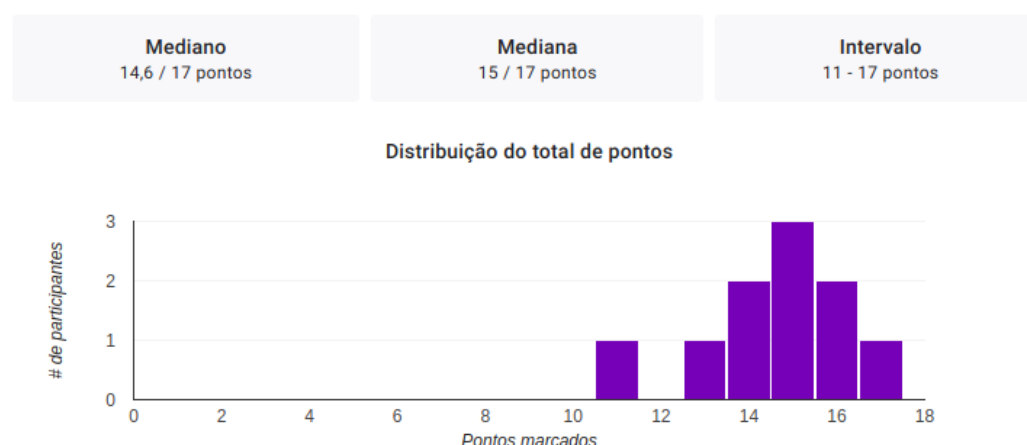
### Informações



Após terem respondido o questionário sem nenhum apoio, os mesmos utilizaram o aplicativo e logo após o uso foi passado o mesmo questionário com o intuito de verificar o nível de aprendizado após a utilização da aplicação. Os resultados obtidos foram bem significativos em relação ao primeiro questionário, a média passou de 6 para 15 e o intervalo de questões corretas subiu de 3 - 9 pontos para um intervalo de 11 - 17 pontos.

Figura 18: Resultados do segundo questionário.

### Informações



#### 4.4.3 Avaliação de usabilidade

Após a realização dos questionários para testar o aplicativo, foi feito outro questionário (Apêndice B) com o intuito de verificar a usabilidade do aplicativo, utilizando o método *System Usability Scale (SUS)*.

Tal questionário possuía 10 questões, onde as pontuações calculadas com base no método, são apresentadas na Tabela 16.

Tabela 16: Pontuação obtida nas questões utilizando o método SUS.

Questões	Pontuação
1	28
2	25
3	22
4	28
5	30
6	23
7	31
8	26
9	30
10	31

Após a obtenção da pontuação de cada questão, foi realizado o cálculo da média de todas as pontuações obtidas, onde o resultado da mesma foi de 68,5 sendo que a média do *System Usability Score* é 68 pontos. O resultado obtido no teste de usabilidade da aplicação alcançou um resultado um pouco acima do recomendado pelo SUS, mostrando que a aplicação não apresentou nenhum problema com relação a usabilidade.

## 5 Conclusão

O trabalho desenvolvido mostrou a importância do uso de ferramentas que auxiliem na aprendizagem de algoritmos, por se tratar de uma disciplina ofertada em praticamente todos os cursos da área de computação. Devido o fato desta trazer um certo grau de dificuldade para alunos iniciantes nos cursos, causando altos índices de reprovação e evasão.

Neste documento foram apresentados os principais conceitos por trás do desenvolvimento da aplicação, bem como os recursos utilizados para a construção do mesmo. Para entender como a aplicação seria desenvolvida para atender as necessidades dos usuários, foram elaborados as descrições dos requisitos funcionais, não funcionais e as regras de negócio, bem como o diagrama de casos de uso e o diagrama entidade relacionamento.

Através dos testes com os usuários, pode-se notar a importância da ferramenta no ensino-aprendizado de algoritmos. Os questionários aplicados mostraram que antes de usarem a aplicação a taxa de acertos nas questões foi de aproximadamente 35% da pontuação total, após o uso do aplicativo essa taxa subiu para aproximadamente 88%. Representando uma taxa de aprendizado bastante expressiva.

Diante dos fatos apresentados conclui-se que a ferramenta representa algo bastante interessante na aprendizagem de algoritmos, disciplina que representa um certo grau de dificuldade aos alunos dos cursos da área de computação.

### 5.0.1 Trabalhos Futuros

Visando melhorar ainda mais o aplicativo, propõe-se como trabalhos futuros:

- Adicionar um *ranking* com as pontuações dos usuários;
- Adicionar conquistas ao fim dos tópicos;
- Melhorar o *layout* visando uma melhor interatividade no uso da aplicação.

## Referências

- AMARAL, Érico et al. Algo+ uma ferramenta para o apoio ao ensino de algoritmos e programação para alunos iniciantes. *Anais do XXVIII Simpósio Brasileiro de Informática na Educação (SBIE 2017)*, p. 1677–1686, 2017. Citado 2 vezes nas páginas 26 e 27.
- BARCELOS, R. et al. O uso de mobile learning no ensino de algoritmos. *RENOTE - Revista Novas Tecnologias na Educação*, v. 7, n. 2, 2009. Citado na página 14.
- BLOGIPOG. *Tecnologia da Informação: conheça as tendências e perspectivas de mercado para 2017*. 2017. Disponível em: <<https://blog.ipog.edu.br/tecnologia/tecnologia-da-informacao-conheca-as-tendencias-e-perspectivas-de-mercado-para-2017/>>. Acesso em: 27 abr. 2018. Citado na página 14.
- CORMEN, T. H. et al. *Algoritmos: teoria e prática*. [S.l.]: Rio de Janeiro - RJ, 2002. Citado na página 17.
- CRISTINA, M.; BENTO, M. Tecnologias móveis em educação: o uso do celular na sala de aula. *ECCOM - Educação, Cultura e Comunicação*, v. 4, n. 7, p. 113–120, 2009. Citado na página 15.
- DEVELOPERS, A. *Arquitetura da plataforma*. 2019. Disponível em: <<https://developer.android.com/guide/platform?hl=pt-br>>. Acesso em: 23 out. 2019. Citado 3 vezes nas páginas 19, 20 e 21.
- ELMASRI, R.; NAVATHE, S. B. *Sistemas de Banco de Dados*. [S.l.]: Pearson Education, 2005. 798 p. (4ª edição). ISBN 85-88639-17-3. Citado na página 24.
- ESTADÃO. *Brasil tem 230 mi de smartphones em uso*. 2019. Disponível em: <[https://www.em.com.br/app/noticia/economia/2019/04/26/internas\\_economia,1049125/brasil-tem-230-mi-de-smartphones-em-uso.shtml](https://www.em.com.br/app/noticia/economia/2019/04/26/internas_economia,1049125/brasil-tem-230-mi-de-smartphones-em-uso.shtml)>. Acesso em: 09 dez. 2019. Citado na página 15.
- FABRICIO TEIXEIRA. *O que é o SUS (System Usability Scale) e como usá-lo em seu site*. 2015. Disponível em: <<https://brasil.uxdesign.cc/o-que-%C3%A9-o-sus-system-usability-scale-e-como-us%C3%A1-lo-em-seu-site-6d63224481c8>>. Acesso em: 01 dez. 2019. Citado 2 vezes nas páginas 24 e 25.
- FERNANDO GODÓY. *TCC – Casos de Uso*. 2013. Disponível em: <<https://fernandogodoy.wordpress.com/2011/07/06/tcc-casos-de-uso/>>. Acesso em: 15 nov. 2019. Citado na página 23.
- GLOBALSTATS, S. *Operating System Market Share Worldwide*. 2019. Disponível em: <<https://gs.statcounter.com/os-market-share>>. Acesso em: 25 out. 2019. Citado na página 19.
- GOMES, E. R. Helpblock: Uma ferramenta web baseada na biblioteca blockly para apoio ao ensino de algoritmos. 2017. Citado 2 vezes nas páginas 26 e 27.
- GUEDES, G. T. A. *UML 2 - Uma Abordagem Prática*. São Paulo: Novatec, 2011. 488 p. (2ª edição). ISBN 978-85-7522-281-2. Citado na página 23.

- JÚNIOR, R.; BONIATI, B. Logicblocks: Uma ferramenta para o ensino de lógica de programação. *Anais do EATI*, n. 1, p. 63–70, 2015. Citado na página 14.
- LECHETA, R. R. *Google Android: aprenda a criar aplicações para dispositivos móveis com o Android SDK*. [S.l.]: São Paulo - SP, 2013. Citado 2 vezes nas páginas 18 e 19.
- MOREIRA, B. G. Desenvolvimento de uma ferramenta de avaliação por pares para disciplinas de algoritmos e programação. *3º Congresso Brasileiro de Informática na Educação (CBIE 2014)*, p. 68–75, 2014. Citado 2 vezes nas páginas 26 e 27.
- OLIVEIRA, M. et al. Ensino de lógica de programação no ensino fundamental utilizando o scratch: um relato de experiência. *XXXIV Congresso da Sociedade Brasileira de Computação*, 2014. Citado 2 vezes nas páginas 26 e 27.
- PREZOTTO, E. D.; BONIATI, B. B. Estudo de frameworks multiplataforma para desenvolvimento de aplicações mobile híbridas. *Anais do EATI - Encontro Anual de Tecnologia da Informação e Semana Acadêmica de Tecnologia da Informação*, v. 4, n. 1, p. 72–79, 2014. Citado na página 18.
- RAMOS DE SOUZA JANONES. *Como usar MySQL com Android Java no Webservice*. 2019. Disponível em: <<https://www.ramosdainformatica.com.br/programacao/mobile-programacao-2/como-usar-mysql-com-android-java-no-webservice/>>. Acesso em: 15 nov. 2019. Citado na página 29.
- ROSA, A. et al. Uma abordagem orientada a modelos para modelagem conceitual de banco de dados. *Anais SULCOMP*, v. 6, 10 2012. Citado na página 24.
- SANTOS, A. S. et al. UtilizaÇÃo de um ambiente de aprendizagem como proposta de auxílio ao ensino de algoritmos e programaÇÃo. *XVII SEPA - SEMINÁRIO ESTUDANTIL DE PRODUÇÃO ACADÊMICA*, v. 6, p. 430–448, 2017. Citado 2 vezes nas páginas 26 e 27.
- SANTOS, R. P. dos; COSTA, H. A. X. Análise de metodologias e ambientes de ensino para algoritmos, estruturas de dados e programação aos iniciantes em computação e informática. *INFOCOMP*, v. 5, n. 1, p. 41–150, 2006. Citado na página 17.
- SILVA, A. M. R. da; VIDEIRA, C. A. E. *UML, Metodologias e Ferramentas CASE*. [S.l.]: Lisboa - LX, 2001. Citado 2 vezes nas páginas 22 e 23.
- SILVA, L. de Rezende da; KNOP, I. de O. Um jogo multiplataforma para prática de algoritmos. *X Workshop do DCC*, 2018. Citado na página 27.
- SOMMERVILLE, I. *Software engineering*. [S.l.]: Pearson Education, 2011. 790 p. (9ª edição). ISBN 978-0-13-703515-1. Citado 2 vezes nas páginas 21 e 22.
- SOUSA, B. J. de et al. *Introdução a Programação*. João Pessoa, 2014. Citado na página 14.
- ZIVIANI, N. *Projeto de Algoritmos Com Implementações em Pascal e C*. [S.l.]: São Paulo - SP, 1999. Citado na página 17.

## Apêndices

# APÊNDICE A – Apêndice

Figura 19: Questionário aplicado antes e após utilização do aplicativo.

## Questões sobre algoritmos

\*Obrigatório

### 1. Marque a alternativa que apresenta o conceito de algoritmos. \*

Marcar apenas uma oval.

- ☐ Um algoritmo é uma sequência lógica e infinita de instruções, que devem ser seguidas para a resolução de um problema ou execução de uma tarefa.
- ☐ Um algoritmo é uma sequência de passos documentados que devem ser seguidos para resolver diversos problemas.
- ☐ Um algoritmo é uma sequência lógica e finita de instruções, que devem ser seguidas para a resolução de um problema ou execução de uma tarefa.
- ☐ Um algoritmo é uma espécie de receita de bolo onde as instruções devem ser seguidas para resolver um problema específico.

### 2. Marque a opção que apresenta os principais tipos de representação de algoritmos. \*

Marque todas que se aplicam.

- ☐ Pseudocódigo, Portunhol e Fluxograma
- ☐ Portugol, K-Means, Matriz
- ☐ Java, Descrição Narrativa, Pseudocódigo
- ☐ Descrição Narrativa, Fluxograma, Pseudocódigo

### 3. A descrição narrativa é uma forma de algoritmo que utiliza linguagem estruturada e, embora sua estrutura e simbologia apresentem pequenas diferenças em relação à língua portuguesa, também é chamada de português estruturado. \*

Marcar apenas uma oval.

- ☐ Certo
- ☐ Errado

### 4. Existem diversas formas de expressar um algoritmo. Em particular, nos fluxogramas, os símbolos, da esquerda para a direita, \*



Marcar apenas uma oval.

- ☐ Armazenamento e Exibição
- ☐ Decisão e Processo
- ☐ Decisão e Documento
- ☐ Transporte e Documento



**5. A estrutura básica de um pseudocódigo é formada pelas palavras: \****Marcar apenas uma oval.*

- ☐ Algoritmo, var, inicio, fim
- ☐ pseudocódigo, inicio, var, fim
- ☐ Algoritmo, inicio, var, fim
- ☐ Algoritmo, variaveis, inicio, fim

**6. Marque a alternativa que apresenta os tipos primitivos de dados. \****Marcar apenas uma oval.*

- ☐ Inteiro, Real, Vetor, Matriz
- ☐ Real, Numérico, Constante, Lógico
- ☐ Inteiro, Real, Caractere, Lógico
- ☐ Fracionários, Texto, Real, Ponteiro

**7. Com relação a constantes e variáveis marque a alternativa incorreta. \****Marcar apenas uma oval.*

- ☐ Uma constante são valores fixos que um programa não pode alterar.
- ☐ Uma variável é uma posição de memória usada para guardar um valor.
- ☐ Uma variável corresponde a uma posição de memória, cujo conteúdo não pode variar ao longo do tempo durante a execução de um programa.

**8. Marque a alternativa que apresenta o comando que é utilizado para entrada de dados em algoritmos. \****Marcar apenas uma oval.*

- ☐ entrada
- ☐ leia
- ☐ escreva
- ☐ var

**9. Marque a alternativa que apresenta o comando de saída em algoritmos. \****Marcar apenas uma oval.*

- ☐ saída
- ☐ leia
- ☐ escreva
- ☐ var

**10. A estrutura de decisão SE/ENTÃO/SENÃO, ou IF/THEN/ELSE, permite que seja sempre executado um comando. Isso porque, caso a condição seja verdadeira, o comando da condição SE/ENTÃO será executado; caso contrário, o comando da condição SENÃO (falsa) será executado. \****Marcar apenas uma oval.*

- ☐ Certo
- ☐ Errado

11. A condição, na estrutura de repetição enquanto-faça, é testada antes de se executar o conjunto de instruções. \*

Marcar apenas uma oval.

- ☐ Certo  
☐ Errado

12. O laço faça-enquanto será executado sempre que a condição for falsa e terminará quando esta for verdadeira. \*

Marcar apenas uma oval.

- ☐ Certo  
☐ Errado

13. Nos laços de repetição para-faça, a condição é verificada no princípio do laço, antes da entrada nesse laço. \*

Marcar apenas uma oval.

- ☐ Certo  
☐ Errado

14. Em algoritmos é também conhecido (a) como array uni-dimensional. \*

Marcar apenas uma oval.

- ☐ Grafos  
☐ Matriz  
☐ Vetor  
☐ Pilha

15. Dado um vetor X com zeros em todas as variáveis e o algoritmo. Serão digitados na sequência os números 34, 81, 79, 25 e 60. Após a digitação do segundo número, o vetor conterá: \*

```
(1)  PARA i ← 1 ATÉ 5 FAÇA
(2)  INICIO
(3)      ESCRIVA "Digite o número ", i
(4)      ESCRIVA X[i]
(5)  FIM
```

Marcar apenas uma oval.

- ☐ 0 - 81 - 0 - 0 - 0  
☐ 34 - 81 - 0 - 0 - 0  
☐ 34 - 81 - 79 - 0 - 0  
☐ 34 - 81 - 79 - 25 - 0  
☐ 34 - 81 - 79 - 25 - 60

16. A estrutura de uma matriz não permite que o armazenamento de valores possa ser referenciado e associado a outros em duas ou mais dimensões. \*

Marcar apenas uma oval.

- ☐ Certo  
☐ Errado

17. Selecione qual das seguintes configurações de linhas e colunas de matriz possui maior número de elementos: \*

*Marcar apenas uma oval.*

- ☐ 10 linhas e 1 coluna
- ☐ 1 linha e 15 colunas.
- ☐ 4 linhas e 3 colunas
- ☐ 2 linhas e 7 colunas.
- ☐ 4linhas e 4colunas

Fonte: O autor

# APÊNDICE B – Apêndice

## Questionário de usabilidade

\*Obrigatório

1. Eu acho que gostaria de usar esse aplicativo com frequência. \*

*Marcar apenas uma oval.*

- ☐ 1  
☐ 2  
☐ 3  
☐ 4  
☐ 5

2. Eu acho o aplicativo desnecessariamente complexo.

*Marcar apenas uma oval.*

- ☐ 1  
☐ 2  
☐ 3  
☐ 4  
☐ 5

3. Eu achei o aplicativo fácil de usar.

*Marcar apenas uma oval.*

- ☐ 1  
☐ 2  
☐ 3  
☐ 4  
☐ 5

4. Eu acho que precisaria de ajuda de uma pessoa com conhecimentos técnicos para usar o aplicativo. \*

*Marcar apenas uma oval.*

- ☐ 1  
☐ 2  
☐ 3  
☐ 4  
☐ 5

5. **Eu acho que as várias funções do aplicativo estão muito bem integradas. \***

*Marcar apenas uma oval.*

- ☐ 1  
☐ 2  
☐ 3  
☐ 4  
☐ 5

6. **Eu acho que o aplicativo apresenta muita inconsistência. \***

*Marcar apenas uma oval.*

- ☐ 1  
☐ 2  
☐ 3  
☐ 4  
☐ 5

7. **Eu imagino que as pessoas aprenderão como usar esse aplicativo rapidamente. \***

*Marcar apenas uma oval.*

- ☐ 1  
☐ 2  
☐ 3  
☐ 4  
☐ 5

8. **Eu achei o aplicativo atrapalhado de usar. \***

*Marcar apenas uma oval.*

- ☐ 1  
☐ 2  
☐ 3  
☐ 4  
☐ 5

9. **Eu me senti confiante ao usar o aplicativo. \***

*Marcar apenas uma oval.*

- ☐ 1  
☐ 2  
☐ 3  
☐ 4  
☐ 5

**10. Eu precisei aprender várias coisas novas antes de conseguir usar o aplicativo.**

*Marcar apenas uma oval.*

- ☐ 1
- ☐ 2
- ☐ 3
- ☐ 4
- ☐ 5

Fonte: O autor





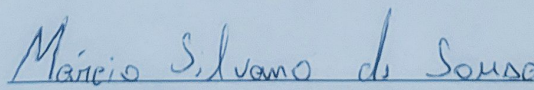
**TERMO DE AUTORIZAÇÃO PARA PUBLICAÇÃO DIGITAL NA BIBLIOTECA  
“JOSÉ ALBANO DE MACEDO”**

**Identificação do Tipo de Documento**

- ☐ Tese
- ☐ Dissertação
- ☒ Monografia
- ☐ Artigo

Eu, **Márcio Silvano de Sousa**, autorizo com base na Lei Federal nº 9.610 de 19 de Fevereiro de 1998 e na Lei nº 10.973 de 02 de dezembro de 2004, a biblioteca da Universidade Federal do Piauí a divulgar, gratuitamente, sem ressarcimento de direitos autorais, o texto integral da publicação **ALGOEDUC: um aplicativo móvel para apoio no ensino de algoritmos** de minha autoria, em formato PDF, para fins de leitura e/ou impressão, pela internet a título de divulgação da produção científica gerada pela Universidade.

Picos-PI 17 de Fevereiro de 2020.

  
Assinatura