**PROTÓTIPO DE APLICATIVO PARA DISPOSITIVOS MÓVEIS PARA AUXILIAR NO APRENDIZADO DOS CONCEITOS BÁSICOS DE LÓGICA DE PROGRAMAÇÃO**

**Introdução**

O alto grau de desenvolvimento que a tecnologia apresenta, remete ao pensamento que a programação de computadores é uma área exclusiva para profissionais de informática, que apenas as pessoas que trabalham diretamente com tecnologia é que deveriam aprender a programar. Porém, de acordo com Celes (2016), a presença dos computadores é tão massiva que dificilmente algum profissional não esteja envolvido com seu manuseio, mesmo como usuários de aplicações específicas. Para ele, existe uma carência de profissionais como engenheiros, químicos, matemáticos, geofísicos, etc. que tenham conhecimentos avançados em programação, de forma que possam ter certa autonomia em suas atividades.

A programação nada mais é que uma sequência de etapas que são realizadas para se alcançar determinado objetivo. Independentemente da área em que uma pessoa atua, essas sequências são determinadas cotidianamente para identificar soluções para os problemas que surgem. Ao transferir essas etapas para uma máquina, está sendo feita apenas uma tradução para um formato compreensível para o computador, assim um mesmo problema pode ser tratado de forma mais rápida e eficiente.

Em meados de 2016 os Estados Unidos celebraram a Semana de Ensino em Ciência da Computação (Computer Science Education Week). De acordo com o site do evento, o que se pretendia, era conseguir o apoio da população ao assinarem uma petição concordando que “todos os estudantes em todas as escolas deveriam ter a oportunidade de aprender ciência da computação” (CsedWeek.org, 2016).

A premissa para todo esse movimento, são os dados estatísticos apresentados pelo Code.Org, que reúne um grupo estratégico de profissionais da computação interessados na disseminação da base da Ciência da Computação em toda a Educação Básica americana. Esses dados apontam, por exemplo, que em 2020 existirão nos Estados Unidos quatrocentos mil alunos de Ciência da Computação para um milhão e quatrocentas mil vagas de emprego. Apontam ainda, que atualmente, apenas 2% dos estudantes escolhem a área de Ciência da Computação (Code.Org, 2016). As estatísticas apontam ainda números relacionados ao gênero e à etnia em relação à escolha da área de Ciência da Computação.

Se a sociedade atual é baseada na economia digital, a preocupação do governo norte-americano e de qualquer outra nação que perceba uma queda na formação de mão de obra no segmento de Tecnologia da Informação é bastante pertinente. Por que uma área tão estratégica e imprescindível para qualquer nação, que apresenta constante perspectiva de crescimento e remuneração atrativa, enfrenta dificuldades de formação mão de obra qualificada? Por que alguns países desenvolvidos como a Inglaterra e a China já introduziram em seus currículos de educação básica o ensino de Ciência da Computação, em especial, a programação de computadores? Já no Brasil, os projetos envolvendo a programação ainda envolvem um número limitado de escolas e dependem da iniciativa de educadores. PIVA (2016)

Resnick (2013), acredita que a programação é uma das habilidades do século XXI e deve ser tratada com a mesma importância que a leitura e a escrita. Para ele, é importante aprender a codificar, não só pelas oportunidades de trabalho, mas pela possibilidade de ver o mundo de novas maneiras.

De acordo com Barker-Plummer (2013), não existe nenhum requisito conceitual necessário para que um aluno do Ensino Médio possa aprender lógica de programação. Estando eles aptos a desenvolver o raciocínio voltado à computação.

Um levantamento realizado pelo Sindicato das Entidades Mantenedoras de Estabelecimentos de Ensino Superior no Estado de São Paulo (Semesp, 2012) mostrou que os cursos da área de tecnologia da informação são os que apresentam a maior taxa de desistência de alunos. De acordo com a pesquisa, a cada três alunos que entram no curso de Sistemas de Informação, apenas um recebe o diploma. Na maioria das vezes, isso ocorre porque os alunos iniciam o curso sem nenhum conceito de programação, e apresenta dificuldades em entender conteúdos abstratos.

Para aprender lógica de programação, existem alguns métodos diferentes que incluem estudo teórico e prática de exercícios, onde se escreve comandos em algum ambiente de programação. Ainda existem jogos e ambientes que se utilizam de recursos visuais para facilitar a compreensão, como por exemplo, encaixando-se uma sequência de blocos para representar determinada atividade.

A Internet abriga algumas ferramentas que auxiliam o aprendizado da lógica de programação. Entre elas, destacam-se:

* Hour of Code: Uma plataforma web com diversos tutorias e vídeos sobre programação.
* Codeacademy: Uma plataforma web, onde você se cadastra e tem acesso a conteúdos sobre programação.
* Scratch: Uma plataforma web que ensina programação de forma visual, usando blocos com comandos específicos, do estilo arrastar e soltar.
* Blockly: Outra plataforma que também ensina programação utilizando blocos de códigos.

Blikstein (2008), define a importância do pensamento computacional:

“... é saber usar o computador como um instrumento de aumento do poder cognitivo e operacional humano – em outras palavras, usar computadores, e redes de computadores, para aumentar nossa produtividade, inventividade e criatividade. ”

Para ele, o pensamento computacional é uma união entre as habilidades humanas e as habilidades tecnológicas. Da mesma forma que um computador é capaz de realizar operações matemáticas complexas e fazer tarefas repetitivas, os humanos conseguem se utilizar da intuição para reconhecimento de determinadas situações. Os dois lados se completam, trabalhando juntos para alcançar um objetivo. Aprender a pensar de maneira computacional, significa transferir para a máquina o raciocínio desenvolvido e assim, consegue-se então, melhores resultados.

**Objetivo**

O objetivo deste projeto é o desenvolvimento de um protótipo de aplicativo para dispositivos móveis, visando ensinar conceitos iniciais de lógica de programação que estimulem o desenvolvimento do pensamento computacional, de forma que pessoas que nunca tiveram contato com lógica aprendam com maior facilidade os tópicos iniciais de programação que são objeto de estudo nos semestres iniciais dos cursos de computação. O Aplicativo irá utilizar os conceitos de "gamificação" de forma a estimular o usuário a continuar o aprendizado com o recebimento de pontos e Badges.

A ideia é desenvolver um Jogo de cartas que ensina lógica de programação, Os usuários entrarão em contato com conceitos de programação expressados de forma visual e aprenderão a utilizar o pensamento computacional de forma intuitiva, sem contato direto com código e sintaxes de programação.

O jogador terá que cumprir certos desafios no jogo, utilizando-se das cartas disponíveis para formar a lógica de um algoritmo, recebendo orientação de um personagem controlado pela máquina. O jogador deverá posicionar as cartas disponíveis em sua mão de forma correta para que o algoritmo funcione, podendo então prosseguir para o desafio seguinte. Caso encontre dificuldade, o jogador poderá recorrer a dicas, as fases iniciais contam com tutoriais detalhados de cada conceito para que o jogador consiga resolver os problemas de forma independente nos desafios seguintes.

Visto que este protótipo é focado para indivíduos que nunca tiveram contato com programação, apenas serão abordados os seguintes conteúdos:

* Variáveis e tipos de dados
* Entrada, processamento e saída
* Operações aritméticas e relacionais
* Estrutura de decisão e de repetição

**Referencial Teórico**

Para esta pesquisa, no referencial teórico, serão estudados autores como Souza (2016) que apresentam aspectos históricos do pensamento humano e da lógica da sociedade ao longo dos tempos. Serão ainda estudados os conceitos de do desenvolvimento de jogos e gamificação, através das publicações de Lorenzoni(2016) e Espíndola (2016).

Livros de Lógica e introdução a programação: Use a cabeça. Programação. BARRY.

Livros de desenvolvimento de aplicativos: Google Android. LECHETA.

Currículo de computação da SBC

**Metodologia**

O projeto será desenvolvido acompanhando as seguintes Fases:

1. FASE DE PESQUISA DE CONTEÚDO
   1. Pesquisa bibliográfica: Pesquisa em livros e sites sobre os conceitos de programação que serão abordados no aplicativo, gerando a BASE DE CONHECIMENTO.
   2. Síntese do conteúdo: Adaptação dos principais aspectos de cada conceito para o formato mais adequado para o aplicativo.
   3. Definição da Estratégia do jogo: Quando se vence, quando se perde, o que é permitido ou não dentro do jogo.
   4. Definição da Jogabilidade: Como o jogador interage com o aplicativo, que tipo de movimentos ele poderá fazer, como é a Jogabilidade dentro das fases, como as cartas se comportam conforme a interação do usuário.
   5. Definição dos Desafios : Se os desafios serão divididos em categorias, por dificuldade, desafios que são bloqueados inicialmente,se haverá algum limite de tempo nos desafios, dicas para auxiliar o jogador no desafio, pontuação, conquistas que podem ser desbloqueadas.
   6. Adaptação dos Conceitos: Adaptar os conceitos da BASE DE CONHECIMENTO às mecânicas das cartas do jogo, o que cada carta faz, as combinações de cartas, a ordem que são colocadas no jogo.
2. FASE DE ANÁLISE
   1. Elicitação de requisitos: Definição dos requisitos de usuário e de sistema, definição dos requisitos funcionais e não funcionais.
   2. Documentação de requisitos: Documentação escrito contendo os requisitos do sistema.
   3. Documentação do Manual do Jogo: Documento escrito contendo as regras e funcionamento do jogo.
3. FASE DE PROJETO
   1. Elaboração dos diagramas usando UML: Elaboração do modelo de domínio, diagramas de classes, casos de uso e atividades.
   2. Recursos visuais e sonoros do jogo: Desenvolvimento das imagens, design das cartas, música de fundo e animações do jogo.
4. FASE DE DESENVOLVIMENTO
   1. Implementação: Desenvolver os módulos do aplicativo em linguagem JAVA, utilizando da plataforma Android Studio ou semelhante.
   2. Testes internos: Realização de testes e correção de bugs do aplicativo.
5. AVALIAÇÃO DO APP
   1. Disponibilização do protótipo para avaliação de aceitação através de uma versão de testes com os módulos parciais para os usuários.
   2. Análise da aceitação do protótipo: Coletar feedback dos usuários.
6. ENTREGA FINAL
   1. Entrega do aplicativo com todos os módulos planejados:

**CRONOGRAMA**

Este cronograma foi desenvolvido a partir dos princípios da metodologia ágil. SOMMERVILLE(2011), utilizando o processo de desenvolvimento incremental, onde o sistema é implementando em versões, onde novos módulos são incluídos a cada nova versão.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Fase/ mês | 1 | 2 | 3 | 4 |
| Pesquisa | X |  |  |  |
| Análise | X | X |  |  |
| Projeto |  | X | X |  |
| Desenvolvimento |  |  | X | X |
| Avaliação do APP |  |  |  | X |
| Entrega final |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Fase/ mês | 5 | 6 | 7 | 8 |
| Pesquisa | X |  |  |  |
| Análise | X | X |  |  |
| Projeto |  | X | X |  |
| Desenvolvimento |  |  | X | X |
| Avaliação do APP |  |  |  | X |
| Entrega final |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Fase/ mês | 9 | 10 | 11 | 12 |
| Pesquisa | X |  |  |  |
| Análise | X | X |  |  |
| Projeto |  | X | X |  |
| Desenvolvimento |  |  | X | X |
| Avaliação do APP |  |  |  |  |
| Entrega final |  |  |  | X |

**Referências Bibliográficas**

BARKER-PLUMMER , DAVE. (2013). Ensino de lógica não precisa estar restrito à universidade, diz professor. Disponível em <<http://g1.globo.com/educacao/noticia/2014/12/ensino-de-logica-nao-precisa-estar-restrito-universidade-diz-professor.html>>Acesso em 18 de mai. de 2017.

BLIKSTEIN ,PAULO. (2008). O pensamento computacional e a reinvenção do computador na educação. Disponível em <<http://www.blikstein.com/paulo/documents/online/ol_pensamento_computacional.html>>Acesso em 18 de mai. de 2017.

CELES, VALDEMAR. (2016). A importância de aprender programação. Disponível em <<https://webinsider.com.br/2016/05/17/importancia-de-aprender-programacao-de-computadores/>>.Acesso em 18 de mai. de 2017.

Code.org.(2016). CODE. Disponível em <https://code.org/>. Acesso em 18 de mai. de 2017.

CsedWeek.org (2016). Computer Science Education Week. DIsponível em <<https://csedweek.org/>>. Acesso em 18 de mai. de 2017.

PIVA , NAIADY. (2016). Inclusão da programação nos currículos escolares avança no exterior. Disponível em <<http://www.gazetadopovo.com.br/educacao/inclusao-da-programacao-nos-curriculos-escolares-avanca-no-exterior-9dlmbpvkpztvrp4vp1647l8fk>> Acesso em 18 de mai. de 2017.

RESNICK , MITCHEL. (2013). Learn to Code Code to Learn. Disponível em <<http://web.media.mit.edu/~mres/papers/L2CC2L-handout.pdf>>Acesso em 18 de mai. de 2017.

SEMESP. (2012). Índice de evasão de alunos é maior na área de tecnologia da informação. Disponível em <<http://www.semesp.org.br/site/indice-de-evasao-de-alunos-e-maior-na-area-de-tecnologia-da-informacao/>>Acesso em 18 de mai. de 2017.

SOMMERVILLE, IAN. Engenharia de Software / Ian Sommerville. Tradução Ivan Bosnic e Kalinka G. de O. Gonçalves. Revisão técnica Kechi Hirama. — 9. ed. — São Paulo : Pearson Prentice Hall, 2011.

SOUSA, FERNANDO F. S. Computadores e Sociedade – da filosofia às linguagens de programação. Curitiba: Editora Intersaberes, 2016.

LORENZONI , MARCELA. (2016). Gamificação: o que é e como pode transformar a aprendizagem. Disponível em <[http://info.geekie.com.br/gamificacao/](http://info.geekie.com.br/gamificacao/%20)>Acesso em 18 de mai. de 2017.

ESPÍNDOLA , RAFAELA. (2016). O que é a gamificação e como ela funciona? Disponível em < [http://www.edools.com/o-que-e-gamificacao/](http://www.edools.com/o-que-e-gamificacao/%20) >Acesso em 18 de mai. de 2017.

BARRY,PAUL. Use a cabeça. Programação. 1ª Edição. Editora ALTA BOOKS

LECHETA , RICARDO R. Google Android. Aprenda a criar aplicações para dispositivos móveis com o Android SDK. 5ª Edição. Editora Novatec.