

Java Orientado à Futuros Programadores (JavaOFP)

RESUMO

O presente trabalho constitui-se de um aplicativo para Android, tendo por objetivos ajudar os usuários estudantes de programação em Java a aprenderem de uma forma mais lúdica os conteúdos necessários para se tornar programador dessa linguagem. O aplicativo integra questões de múltipla escolha, links para sites recomendados que ensinam os assuntos abordados e um *ranking* ligado às pontuações para incentivar cada vez mais as pessoas a atingirem seu objetivo. Atualmente o app foca em programação estruturada (PE), podendo futuramente também agregar Programação Orientada à Objetos (POO). Decidiu-se a partir dos dados aqui apresentados que o nome deste *software* será Java Orientado a Futuros Programadores (JavaOFP). Como resultados espera-se uma diminuição no índice de reprovação dos alunos do curso de informática tanto em PE quanto em algoritmos, além de disponibilizar outro meio de aprender a matéria de uma forma gamificada.

Palavras-chave: Programação, Java, Aplicativo.

ABSTRACT

This work is an Android application designed to help Java programming students learn more about the content required to become a Java programmer. The app integrates multiple choice questions, links to recommended websites that teach the topics covered, and a ranking linked to scores to increasingly encourage people to reach their goal. Currently, the application focuses on structured programming (PE) and may also add object-oriented programming (OOP) in the future. It was decided from the data presented here that the name of this software will be Future Developer Oriented Java (JavaOFP). As a result it is expected a decrease in the failure rate of students of the computer course in both PE and algorithms, as well as providing another way to learn the subject in a fun way.

Keywords: programming, Java, Application.

1. Introdução

No curso Técnico Integrado de Nível Médio em Informática, do IFRN *Campus* Santa Cruz, os alunos têm o primeiro contato com disciplinas relacionadas a Algoritmos e Programação de computadores nos dois primeiros anos letivos do curso. Sendo, a disciplina de Fundamentos de Lógica e Algoritmos (FLA), no primeiro ano. E, no segundo ano, a disciplina de Programação Estruturada e Orientada a Objetos (PEOO). Ambas são muito importantes para o currículo dos alunos e são base para continuidade do aprendizado de novos conteúdos e linguagens de programação que são estudadas ao longo dos 4 anos nas demais disciplinas do Curso Técnico em Informática.

Por serem disciplinas de natureza técnica - bem diferentes daquelas já vistas pelos alunos em etapas anteriores do ensino - e também por apresentarem conteúdos do Ensino Superior, tais disciplinas apresentam características desafiadoras para compreensão de boa parte dos alunos e são as que apresentam maior taxa de reprovação. Tal percepção pode ser corroborada levando em consideração a experiência do próprio grupo nas disciplinas de FLA e PEOO. E, principalmente, com base em dados coletados do SUAP (SUAP, 2019), referentes às taxas de reprovação desses estudantes, nos anos de 2016 a 2018, nas duas matérias citadas anteriormente. Na disciplina de FLA, as taxas mínima e máxima de reprovação foram de aproximadamente 30% e 50%, respectivamente. Por sua vez, em PEOO as taxas mínima e máxima de reprovação foram de aproximadamente 20% e 50%.

Partindo disso, o principal objetivo neste trabalho é desenvolver um aplicativo denominado JavaOFP, para plataforma Android, que auxilie na aprendizagem e possibilite a redução da quantidade de alunos reprovados nas disciplinas de FLA e PEOO.

Para atingir o objetivo geral proposto neste trabalho, foram definidos os seguintes objetivos específicos:

- Desenvolver o app de forma gamificada. Para isso, foi criado um *ranking*, usando do tempo e da quantidade de tentativas utilizadas ao responder questões para atribuir pontos ao usuário, premiando-o com avatares e medalhas;
- A criação e implementação de um Banco de Dados, ao qual foi escolhido se trabalhar com *Structured Query Language* (SQL) no Mysql Workbench;
- Desenvolver um *Web Service* hospedado no Apache TomCat;
- Para as funcionalidades em geral que requerem programação, a utilização das ferramentas Android Studio e Eclipse;
- Criação das imagens com o programa Inkscape.

Por fim, foi preciso que o grupo aprofundasse seus conhecimentos em todas as ferramentas citadas anteriormente, levando em consideração de que nem todas eram de seu conhecimento.

2. Metodologia

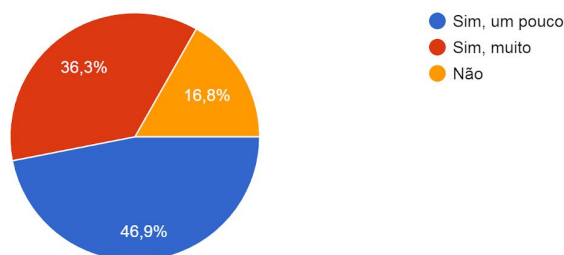
Neste trabalho tem-se como objetivo metodológico o desenvolvimento de um aplicativo que tem como principal função auxiliar iniciantes na programação da linguagem Java por meio de links que os direcionam para sites e vídeos confiáveis para que assim eles estejam aptos para responder questões com alternativas de múltiplas escolhas. Para isso, foi necessário saber quais são as principais dificuldades dos estudantes. Desse modo, foi feita uma pesquisa por meio do *Google Forms*, em que 113 alunos do Curso Técnico Integrado em Informática do IFRN *campus* Santa Cruz, do 2º ao 4º ano, respondessem a algumas questões elaboradas pelos próprios desenvolvedores. Exemplos de questões utilizadas nesta pesquisa se encontram nos gráficos

da Figura: 1 (a), onde é possível observar o percentual de dificuldade dos alunos. Se forem somados o total de alunos que sentem muita dificuldade com aqueles que sentem pouca dificuldade, será obtido um resultado de 83,2%, ou seja um valor muito alto. Já na Figura: 1 (b), é possível saber quais são os principais conteúdos das disciplinas de FLA e PEOO que os estudantes sentem mais dificuldades.

Figura: 1 (a) Pesquisa

Você já teve dificuldades na matéria de programação estruturada?

113 respostas

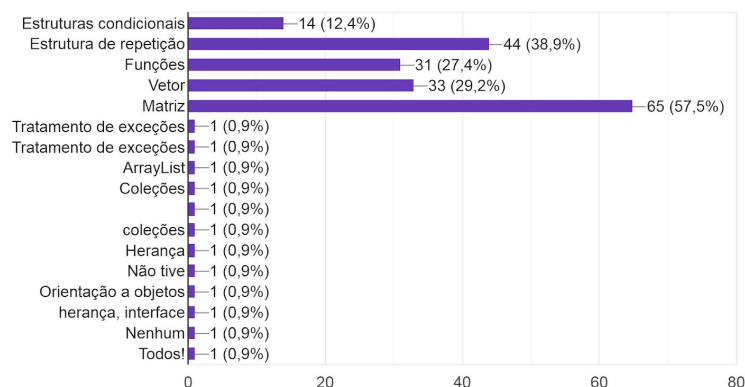


Fonte: Própria

Figura: 1(b) Pesquisa

Qual o assunto que você teve/tem mais dificuldade para aprender em Java?

113 respostas



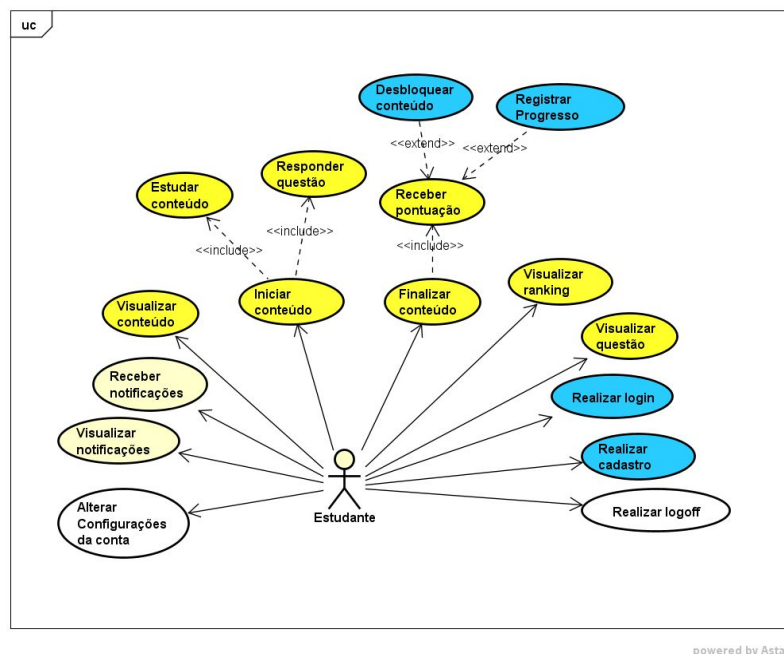
Fonte: Própria

Depois de ter sido constatada a real dificuldade dos estudantes e os níveis de reprovação nas disciplinas de Algoritmos e Programação Estruturada e Orientada a Objeto, foi realizada uma pesquisa na *Google Play Store* por soluções similares ao projeto pretendido. Foram encontradas as seguintes soluções: *Programing hub*, que disponibiliza para o usuário aprender várias linguagens de programação, porém só é possível encontrar esse app em inglês. E, *Pseudocode Visualg Algoritmos* que é um editor de texto para a codificação pseudocódigo que acaba gerando muitos bugs como por exemplo não reconhecendo alguns caracteres.

JavaOFP, no entanto, é um aplicativo totalmente em português e gratuito onde o usuário vai poder acessar em qualquer lugar que tenha internet podendo aprender conteúdos relacionados a PE, de uma forma gamificada, ou seja, o usuário vai aprender por meio de um aplicativo que possui a mecânica de um jogo. Dessa forma quanto mais rápido ele responder as questões e com o menor número de tentativas, o usuário poderá chegar no topo do ranking. Depois de ter sido descoberto os maiores problemas, foi o momento de colocar esse conhecimento em prática e desenvolver o app. Para que o projeto ficasse bem organizado e as prioridades fossem feitas primeiro, foi necessário usar o SCRUM (SUTHERLAND, 2014) juntamente com o Trello, pois ao usar essa metodologia ágil para gerenciamento de projetos o trabalho seria feito de uma forma mais produtiva e em menos tempo.

Outro passo importante para o progresso do projeto foi fazer o Diagrama de Casos de Uso (SOMMERVILLE, 2011) com o auxílio do *software* Astha. O diagrama se encontra na Figura 2, onde o ator é representado pelo estudante e as elipses representam as funcionalidades da aplicação. Sendo que as que estão em cores amarelas já foram feitas, as de cores azuis estão em processo de desenvolvimento, as de cor amarelo claro podem ser para melhorias futuras e as que estão em branco são as de menor prioridade.

Figura: 2 Diagrama de Casos de Uso

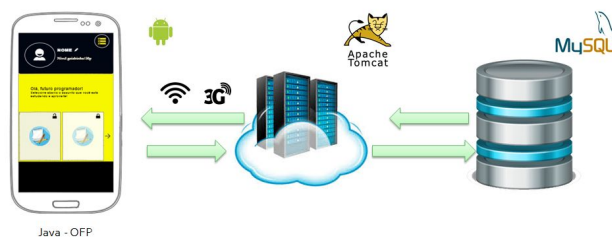


Fonte: Própria

Em seguida foram feitos os modelos conceitual e lógico (HEUSER, 2009) por meio do TerraER e MySQL Workbench que se integram ao aplicativo, para que dessa forma facilitasse a criação do modelo físico que corresponde ao Banco de Dados (HEUSER, 2009), juntamente com a sua implementação. Também foram utilizadas as bibliotecas do Android Retrofit2 e Jackson que tem a função de fazer a conversão dos dados para o formato JSON (*Javascript Object Notation*), outra biblioteca muito importante é mysql-connector, que permite estabelecer uma conexão com o Banco de Dados e realizar consultas SQL via programação.

O projeto possui como solução tecnológica, ilustrada na Figura 3, a implementação do aplicativo JavaOFP, que está disponível para a plataforma android, conectado via wifi, 3G, ou qualquer meio de troca de dados ao *Web Service* (implementando o padrão REST - *Representational State Transfer*) hospedado no apache TomCat, que se comunica com o Banco de Dados e que tem função de transferir e consumir dados no formato JSON. O app realizará requisições ao *Web Service*, como por exemplo dos conteúdos existentes, sendo ele encarregado de interagir com o Banco de Dados e buscar ou levar informações de um lado para outro.

Figura 3 - Solução tecnológica



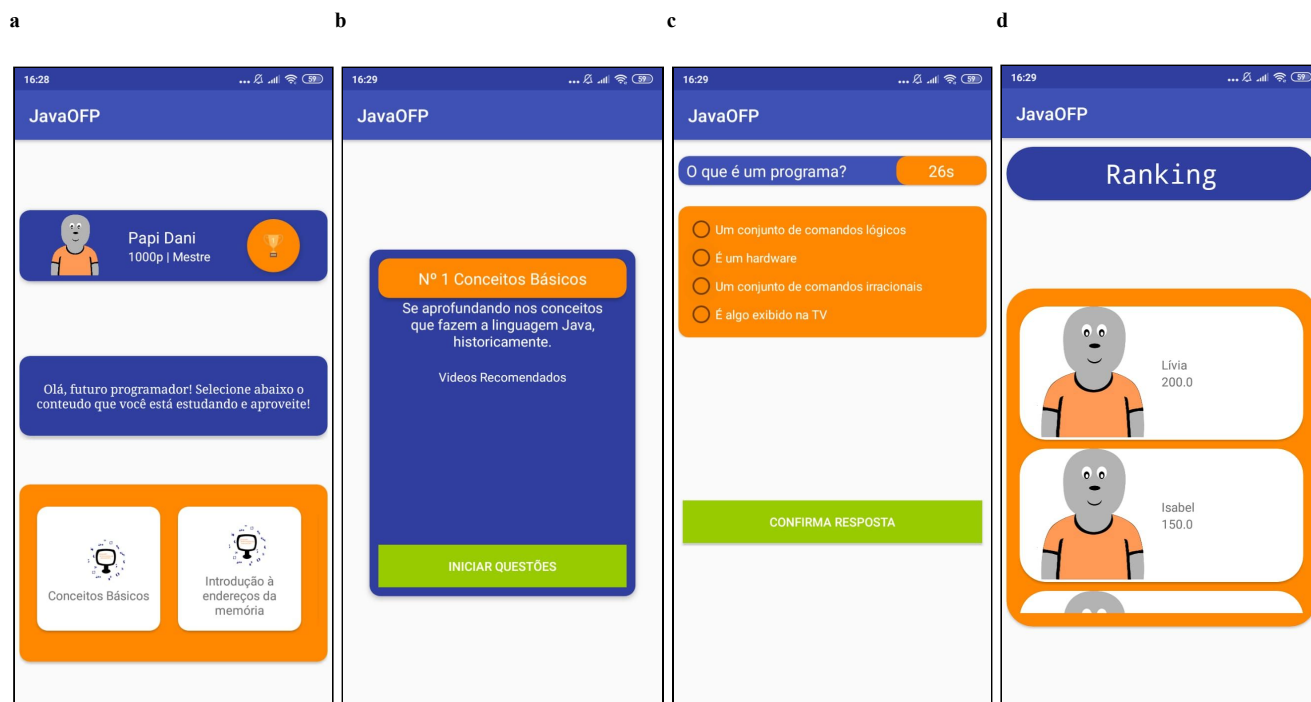
Fonte: Própria

3. Resultados e Discussões

A metodologia adotada e aplicada para a criação e organização do projeto foi a SCRUM que tem finalidade acelerar a produção de uma forma bem simples, dividir o trabalho em pequenas partes denominados *Sprints* que ao serem finalizadas permitiram alcançar um objetivo maior.

A partir de reuniões realizadas semanalmente e registradas no diário de bordo foram desenvolvidos o Banco de Dados, *Web Service* e a aplicação android desenvolvida através do Android Studio que já disponibiliza a tela principal representada na Figura 4 (a) com as funções de mostrar e inicializar os conteúdos existentes além de exibir as informações do estudante como *nickname* e sua pontuação. A de conteúdo na Figura 4 (b) detalha os dados mais importantes a partir da seleção de um dos conteúdos; Já a de *ranking*, na Figura 4 (d), lista todos os estudantes cadastrados no app JavaOFP ordenados pela pontuação e a tela de fase na Figura 4 (c), que possui várias ludificação que seria características de jogos implementadas como o tempo e os pontos que são calculados a partir de uma fórmula matemática que leva em consideração as tentativas e também a velocidade das respostas.

Figura: 4 - (a) Tela principal; (b) Tela de conteúdos; (c) Tela de fase; (d) Tela de ranking.



Fonte: Própria

4. Considerações Finais

Neste trabalho foi apresentado o aplicativo JavaOFP, que visa servir de apoio aos estudos de Java de uma forma gamificada e assim, ajudar a reduzir o índice de reprovações nas matérias técnicas de Programação do curso de Informática. De acordo com os fatos apresentados no decorrer do projeto, podemos concluir que os objetivos propostos foram alcançados, embora os desenvolvedores tivessem encontrado obstáculos durante o percurso, como: limitações de meios adequados para o progresso do aplicativo, conciliação do projeto integrador com outras disciplinas acadêmicas e dificuldades na utilização da ferramenta Android Studio. Mas é importante levar em consideração que melhorias podem ser feitas, como por exemplo: adicionar POO para complementar a PE, ampliar as linguagens de programação para que os usuários tenham mais possibilidades de aumentar seus conhecimentos na área da tecnologia, diversificar os tipos de questões e fazer um módulo professor no qual permita o acompanhamento em tempo real do desenvolvimento do aluno possibilitando que ele detecte as dificuldades no decorrer da disciplina.

Referências

BAPTISTA, Luciana Ferreira. **Linguagem SQL: guia prático de aprendizagem**. 1.ed. São Paulo: Érica, 2012. ISBN 978-85-365-0373-8.

- BRITO, Robison Cris. **ANDROID com Android Studio**: passo a passo. 1. ed. Brasil: Ciência Moderna, 2017. 336 p. v. Não especificado. ISBN 8539907631.
- DETERDING, S., DIXON, D., KHALED, R., NACKE, L. **From game design elements to gamefulness: defining "gamification"**. In: Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments (MindTrek '11). ACM, New York, NY, USA, 9-15., 2011. DOI: <https://doi.org/10.1145/2181037.2181040>
- DATA, Refsnes. Tutorial SQL. [S. l.], 1999. Disponível em: <https://www.w3schools.com/sql/default.asp>. Acesso em: 4 out. 2019.
- GUEDES, Gilleanes T. A. **UML 2: Uma Abordagem Prática**. 2º ed, São Paulo: Novatec Editora, 2011
- HEUSER, Carlos Alberto. **Projeto de Banco de Dados** 6. ed. Porto Alegre: Bookman, 2009. 282 p. il. (Livros didáticos informática ; 4).
- JANDL JÚNIOR, Peter. **Java: guia do programador: atualizado para java 8**. 3. ed. São Paulo: Novatec, 2015. 704 p. il.
- PODEROSO DE OLIVEIRA, Celso Henrique. **SQL: Curso Prático**. 1. ed. Brasil: Novatec, 2002. 272 p. v. Não especificado. ISBN 8575220241.
- SEABORN, K., FELLS, D. I. **Gamification in theory and action: A survey**. International Journal of Human-Computer Studies, Volume 74, February 2015, Pages 14-31, ISSN 1071-5819, <http://doi.org/10.1016/j.ijhcs.2014.09.006>.
- Sistema Unificado de Administração Pública. **Relatório de reprovações**. Disponível em: <<https://suap.ifrn.edu.br/accounts/login/?next=/>>>. Último acesso em: 09/10/2019
- SOMMERVILLE, Ian. **Engenharia de software** 9. ed. São Paulo: Pearson Prentice Hall, 2011. 529 p. il.
- SUTHERLAND, Jeff; INC, Scrum. **Scrum: a arte de fazer o dobro do trabalho na metade do tempo**. São Paulo: [s. n.], 2014.