

O uso de uma *engine* no desenvolvimento de um aplicativo

Ronaldo de Castro Del-Fiaco^{*1}(PQ), NICOLAS ROMÁRIO MOURA DE OLIVEIRA²(IC), ANDRÉ VICTOR SILVA FREIRE³(IC), BRUNO FERREIRA M PASSOS⁴(IC), JONAS GOMES DE SOUZA⁵(IC)

1.Universidade Estadual de Goiás – Câmpus Anápolis de Ciências Exatas e Tecnológicas, Henrique Santillo/Curso Sistemas de Informação – CCET.
ronaldo.delfiaco@ueg.br

2.Universidade Estadual de Goiás – Câmpus Anápolis de Ciências Exatas e Tecnológicas, Henrique Santillo/Curso Sistemas de Informação – CCET.

3.Universidade Estadual de Goiás – Câmpus Anápolis de Ciências Exatas e Tecnológicas, Henrique Santillo/Curso Sistemas de Informação – CCET.

4.Universidade Estadual de Goiás – Câmpus Anápolis de Ciências Exatas e Tecnológicas, Henrique Santillo/Curso Sistemas de Informação – CCET.

5.Universidade Estadual de Goiás – Câmpus Anápolis de Ciências Exatas e Tecnológicas, Henrique Santillo/Curso Sistemas de Informação – CCET.

Resumo: Desenvolver software para smartphones é atender um mercado emergente, a quantidade deste equipamento no Brasil até o final de 2017 será a mesma de habitantes, principalmente softwares de entretenimento. O processo de desenvolvimento de um software de entretenimento é diferente dos aplicativos comerciais ou administrativos, sendo assim, é mais produtivo usar uma IDE que foi projetada para esta finalidade as ENGINES. No projeto SOFTWARE EDUCATIVO E EDUCAÇÃO AMBIENTAL: UM ESTUDO APLICADO, os desenvolvedores utilizam a Engine Unity 3D para desenvolver os aplicativos do projeto, nesta mesma plataforma foi desenvolvido um aplicativo de utilidade pública para notificar o usuário a dar vistoria em objetos e locais previamente devidos, como possíveis criadouros do mosquito Aedes Aegypti em suas residências, disponíveis nas App Store e na Play Store. A vantagem de utilizar a Unity 3D é que ela gera para as duas plataformas, enquanto que outras IDEs desenvolvem para iOS ou para o Android.

Palavras-chave: Unity 3D. Aedes Aegypti. Combate. Notificação. Engenharia de software.

Introdução

Desenvolver software para *smartphones* é atender um mercado emergente, a quantidade deste equipamento no Brasil até o final de 2017 será a mesma de habitantes, ou seja, um *smartphone* para cada cidadão brasileiro, segundo dados da

28ª Pesquisa Anual de Administração e Uso de Tecnologia da Informação nas Empresas, realizada pela Fundação Getúlio Vargas de São Paulo.

O software para *smartphone* é um aplicativo que pode ser obtido diretamente no próprio equipamento, desde que ele esteja conectado na Internet. Nas lojas virtuais como a App Store – loja virtual da Apple, Play Store – loja virtual do Google para os equipamentos que utilizam o sistema operacional Android, Windows Phone Store – loja virtual da Microsoft para os dispositivos com Windows Phone. Nestas lojas é possível encontrar todo tipo e gênero de aplicação.

Os smartphones são fornecidos por vários fabricantes de hardware com arquitetura e sistemas operacionais móveis diferentes, exigindo aplicativos executáveis específicos para cada sistema operacional. Isto significa que, um aplicativo desenvolvido para Iphone da Apple não funcionará nos sistemas operacionais da Black Berry ou nos dispositivos com Android e nem com Windows Phone, ou seja, para cada sistema operacional, deve haver um novo executável.

Para minimizar este cenário e também para obter mais aplicativos para suas plataformas, os fabricantes disponibilizam kits de desenvolvimento de software, com treinamento e suporte. Algumas empresas entendendo a complexidade e o retrabalho, em criar um executável para cada plataforma, confeccionaram sistemas para desenvolver aplicativos (IDE) independentes do sistema operacional do *Smartphone*, facilitando o trabalho dos programadores (PRESSMAN, 2011).

O processo de desenvolvimento de um software de entretenimento é diferente dos aplicativos comerciais ou administrativos, sendo assim, é mais produtivo usar uma IDE que foi projetada para esta finalidade. Estas IDEs são denominadas *GAME ENGINE* ou simplesmente *ENGINE*, sua tradução é motor de jogo (SANTOS, 2012).

Tipos diferentes de sistemas necessitam de diferentes processos de desenvolvimento. Como qualquer processo de desenvolvimento de software as atividades básicas são: levantamento, projeto, implementação, teste e evolução. Com algumas particularidades como: elaboração de uma enredo, cenários, design, fases do jogo entre outras (SOMMERVILLE, 2011).

Os autores participam do projeto SOFTWARE EDUCATIVO E EDUCAÇÃO AMBIENTAL: UM ESTUDO APLICADO, estabelecido pelo convênio da UEG com AMBEV (American Beverage Corporation). Em uma reunião com Sr. Matheus gerente da AMBEV que acompanha o projeto, foi apresentada uma campanha de

combate ao mosquito *Aedes Aegypti* promovida pela AMBEV. No final da reunião em comum acordo resolve-se que, o projeto irá desenvolver um aplicativo para participar desta campanha.

Objetivo do aplicativo é notificar semanalmente o usuário a fazer vistoria nos objetos e locais que acumulam água em sua residência, selecionados previamente, eliminando assim os criadouros do *Aedes Aegypti*.

Como os desenvolvedores que integram o projeto já estavam com experiência no uso de uma *ENGINE*, resolveram em usar esta IDE para a implementação do aplicativo, mesmo sendo uma ferramenta específica para o desenvolvimento de jogos.

Objetivo deste artigo é relatar a experiência de uma equipe de desenvolvedores de software, no uso de uma ferramenta de desenvolvimento de jogos - *ENGINE* – na implementação de um aplicativo de utilidade pública, diferente de um software de entretenimento.

Material e Métodos

O processo de desenvolvimento utilizado foi híbrido, porque o escopo era bem definido e não tinha risco de sofrer alterações durante o desenvolvimento, como foi a primeira vez que a equipe desenvolveu um aplicativo diferente de um jogo digital na Unity 3D era necessário maior interação entre eles para se ajudarem.

A *ENGINE* utilizada no desenvolvimento do aplicativo foi a Unity 3D, as figuras 3D foram modeladas no Blender e os desenhos foram elaborados no Photoshop.

Resultados e Discussão

O aplicativo foi desenvolvido dentro do prazo estabelecido e disponibilizado na App Store e na Play Store, para disponibilizar lá nas duas lojas virtuais, foi necessário criar dois executáveis, um para o sistema operacional Android e outro para o sistema operacional iOS. O fato de ter usado a Unity 3D facilitou a geração dos dois executáveis.

Os conceitos de programação orientada a objetos foram utilizados na implementação, bem como as boas práticas de programação. A linguagem de

programação utilizada foi a C#, dentro do ambiente da Unity 3D, onde possui as classes disponíveis para manipular os objetos das cenas.

Foi necessário adquirir um *plugin* para não comprometer a entrega do projeto, a funcionalidade deste é gerar a notificação. Mesmo assim foi necessário a implementação de uma classe para obter do usuário o dia da semana e horário da notificação e outra para configurar o *plugin* com estes dados. A responsabilidade dele é somente gerar notificação no celular no horário definido, conforme figura 1 aparece um balde despejando água com a frase “Hoje é dia de combater o Aedes!”.

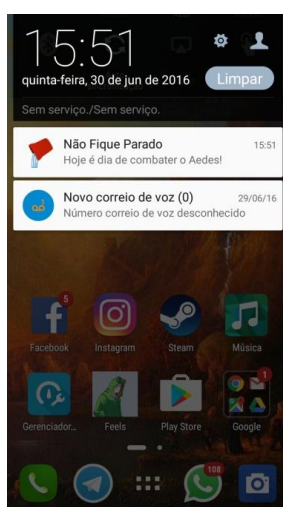


Figura 1 Notificação do Aplicativo

Para publicar um aplicativo na App Store são realizados testes automatizados para verificar se o aplicativo está obedecendo as diretrizes da empresa, no caso o aplicativo “Não fique parado” foi enquadrado nestas diretrizes. Na Play Store não são realizados testes para publicar um aplicativo, basta ter uma conta de desenvolvedor e pagar uma taxa de adesão, enquanto na App Store há uma anuidade ficando dispendioso em manter um sistema lá.

Os números de downloads nas lojas virtuais do aplicativo “Não fique parado”, na App Store é de 247 e na Play Store é de 32, considerando que não foi feito nenhum trabalho de divulgação são números significantes.

Considerações Finais

Se tivesse sido utilizado o Android Studio, por exemplo, não teria sido necessário comprar o plugin de notificação e o desempenho do aplicativo seria melhor, já que os recursos seriam nativos do sistema (android). Porém o tempo estava curto, e na época a equipe não tinha experiência na plataforma Android Studio. Outro fator é que se fosse utilizado Android Studio seria necessário utilizar outra IDE pra desenvolver para iOS. A vantagem do Unity 3D é que ele gera para as duas plataformas.

Agradecimentos

Agradecemos a Deus pelo dom da vida.

Agradecemos a direção do CCET por ter confiado a nós a coordenação e o desenvolvimento do projeto SOFTWARE EDUCATIVO E EDUCAÇÃO AMBIENTAL: UM ESTUDO APLICADO. E também a todos os alunos do curso de Sistemas de Informação que colaboram como bolsistas e voluntários, ou simplesmente como simpatizantes ao projeto.

Agradecemos a AMBEV na pessoa do Sr. Matheus por acreditar que projetos como este são possíveis de sensibilizar as pessoas a mudar seus comportamentos em relação ao meio ambiente, na sua conservação para termos uma qualidade de vida melhor no nosso planeta.

Referências

SANTOS, Rafael A.; GÓES, Vinicius A.; ALMEIDA, Luis F. **Metodologia OriGame: um processo de desenvolvimento de jogos**. Artigo apresentado no XI SBGames. Brasília, 2012.

SOMMERVILLE, Ian. **Engenharia de Software**. 9. ed. São Paulo: Addison Wesley, 2011.

PRESSMAN, Roger. **Engenharia de Software**. 7. ed. São Paulo: McGraw-Hill, 2011.

NERY, Marcelo. **Fundamentos de Jogos Digitais**. Belo Horizonte: PUC MINAS, 2014.