

Game Engine Löve : um levantamento do Ambiente Integrado de Desenvolvimento existentes

Guilherme Azevedo Santos¹, Felipe Batista Giovanini¹, Leandro Mazute¹, Hélio Toshio Kamakawa¹, Wilian Nalepa Oizumi¹

¹Campus Paranavaí - Instituto Federal do Paraná (IFPR)
Paranavaí- PR – Brasil

{azevedo.guilherme,felipebatistagiovanini,leo.mazute}@gmail.com,
{helio.kamakawa,wilian.oizumi}@ifpr.edu.br

1.Introdução

O desenvolvimento de jogos requer um conjunto de recursos integrados que fazem o jogo funcionar. A criação desse complexo sistema não é uma tarefa simples e não pode ser realizado pelas linguagens de programação tradicionais, sendo necessária o uso ou o desenvolvimento de ferramentas específicas para que seja possível iniciar a produção de jogos [GUANA, STROULIA e VINA 2015].

O *Game Engine* (GE) é um *software* que provê serviços comumente necessários para a criação de jogos [CLAYPOOL 2013]. Devido à sua natureza proprietária, os princípios de engenharia de *software* em relação a elaboração de ferramentas para a criação de jogos não são devidamente documentados [LEWIS e WHITEHEAD, 2011].

Uma das GE's de desenvolvimento de jogos é o Löve, com o código-fonte disponibilizado gratuitamente através da plataforma BitBucket, e que ao analisar o seu código-fonte, é visto que a GE foi desenvolvida em C++, e utiliza da linguagem de programação Lua em seus projetos. Coscarelli et al. (2010) afirmam que o Löve é uma das melhores GE's em relação à facilidade de uso, velocidade de execução e portabilidade. Ainda segundo os autores, o GE Löve possui uma comunidade ativa e fornece tutoriais e a documentação necessária para o desenvolvimento de jogos 2 dimensões (2D).

O GE Löve, juntamente com a linguagem Lua são de fácil aprendizado, intuitivos, e permitem que o foco da implementação seja a lógica do jogo, e não detalhes de configuração do *hardware* [COSCARELLI et. al. 2010].

De acordo com o site oficial, Löve possui a licença ZLIB, ou seja, não há custo, pode ser usado para fins comerciais e não há limitações de recursos [LOVE COMMUNITY 2019]. Atualmente a plataforma Löve suporta diversos sistemas operacionais, como Windows, Linux, Mac OS, Android e IOS [LOVE COMMUNITY 2019].

Apesar da popularidade e diversos recursos que facilitam o desenvolvimento de jogos em 2D, a plataforma Löve não possui a ferramenta oficial de apoio ao desenvolvimento de jogos, o *Integrated Development Environment* (IDE).

Gasparic et. al. (2017) afirmam que os recursos de um IDE podem evitar processos repetitivos e permitir que o desenvolvedor tenha foco nos aspectos criativos do projeto, e assim, melhorar a eficiência e a qualidade da criação do jogo.

Desta forma, o objetivo deste estudo é realizar um estudo das IDE's mais populares para a plataforma Löve indicadas no fórum oficial da plataforma: Game-Kitchen, ZeroBrane Studio e REPL.IT. Serão apresentados as principais funcionalidades, bem como as suas limitações.

2.Game-kitchen

Game-Kitchen é um IDE específico para o GE Löve que possui recursos de depuração do projeto, auto-complete e a análise dos erros de sintaxe [GAME-KITCHEN 2019]. Sua última versão conta com o GE Löve na versão 0.10.0, disponibilizada gratuitamente para a plataforma Windows. A IDE possui a licença GNU, que permite a distribuição gratuita do software [YH Lin et al. 2006].

Segundo CLANCY(2019), Game-Kitchen possui os recursos de depuração de código, tecnologia de complemento de código (intellisense), opção de ir para determinada definição do código, explorador vertical do projeto, e validador de código.

A IDE contém uma área de trabalho do Game-Kitchen, que na configuração padrão a área de trabalho é subdividida em 3 consoles, no qual, o editor de código é apresenta a esquerda da área de trabalho.

3.Zerobrane

A IDE ZeroBrane Studio foi desenvolvida para trabalhar com a linguagem de programação Lua, fornecendo recursos como a syntax highlighting, auto-complemento e suporte para depuração de código Lua [KULCHENKO 2018].

ZeroBrane Studio possui a licença MIT, que permite o uso, alteração, cópia, publicação e distribuição do *software*, permitindo ser utilizado nas plataformas Windows, Mac OS e Linux [KULCHENKO 2018]. Segundo o site oficial da IDE, a ZeroBrane Studio tem os recursos de, complemento de código, depuração para determinados motores para Lua, incluindo a LÖVE e analisador de código.

A área de trabalho da ZeroBrane é subdividida em 3 partes, o editor de código, o gerenciador de arquivos e o depurador.

4.Repl.It

O REPL.IT fornece recursos de um IDE no modelo cliente-servidor, ou seja, o GE fica hospedado no servidor e suas interfaces foram desenvolvidas com HTML, CSS e JavaScript, permitindo que qualquer pessoa usufrua dos seus recursos por meio de um navegador [NEOREASON 2019].

Pelo fato dos recursos estarem hospedados em um servidor, a instalação não é necessária, sendo acessível em dispositivos centrados na Web, como computadores, notebooks, celulares e tablets [BERSSANETTE et al. 2018].

A IDE REPL.IT é executada por meio do navegador de internet. Na direta do navegador, a IDE apresenta dois consoles na vertical, a superior apresenta o resultado da execução do código, e a inferior é a saída do compilador.

No desenvolvimento de jogos, grande parte do processo envolve a definição do *design* de cenários, a falta de um editor apropriado restringe o desenvolvimento do mesmo em linhas de programação. Sem o recurso de pré-visualização de cenários, é necessário a compilação e o teste do código fonte para a visualização do que foi produzido. Este processo, além de consumir muito tempo na criação de cenários, exige competências de programação e *design*.

Atualmente existem algumas IDE's gratuitas, porém, nenhuma contém o editor de cenário para pré-visualização, impossibilitando que o cenário seja editado na forma renderizada. Outras soluções integram o GE Löve em IDE genéricos, entretanto, ou são pagas que elevam o custo de desenvolvimento, ou possuem restrições que impedem a utilização de recursos específicos e/ou restringem o uso comercial.

5.Considerações Finais

Apesar da plataforma Löve não possuir um IDE oficial, comunidades, empresas e usuários desenvolvem e disponibilizam IDE's específicas ou genéricas para a plataforma. Conforme indicadas no fórum oficial da plataforma, as mais utilizadas são: Game-Kitchen, ZeroBrane Studio e REPL.IT.

As IDE's apresentando provêm recursos que auxiliam no desenvolvimento de jogos com foco nos aspectos criativos do projeto, evitando processos repetitivos, e assim, melhorando a eficiência e qualidade de criação de jogos. Porém, nenhuma das IDE's apresentadas fornecem o editor de cenário para pré-visualização, impossibilitando que o cenário seja editado na forma renderizada, recurso fundamental no processo de desenvolvimento, visto que, sem este recurso, a cada alteração, é necessário a compilação do projeto para a pré-visualização.

Referências

- Atlassian.(2019) “Software Development And Colaboration Tools”, Bitbucket.
Disponível em:<<https://bitbucket.org/>>. Acessado em: 20 de julho de 2019.
- Berssanette, J. H., Francisco, A. C. and Baran, L. R. (2018) “Espaços Ampliados Apoiados Por Tecnologias Digitais Para o Ensino de Programação de Computadores”, Pleiade vol.12, nº25.
- Coscarelli, C. V., Cafiero, D. and Nogueira, D. (2012) “Papa-Letras: um jogo de auxílio à alfabetização infantil”, Educação & Tecnologia, [S.l.], v. 16, n. 3, out. .

- Claypool, M. (2013) “Strengthening Programming Skills By Building A Game Engine From Scratch”, Computer Science Education, Vol. 23, Nº 2, Page 1-18.
- Gasparic, M., James, A., Ricci, F. and Murphy, G. C. (2004) “Desenvolvimento De Software”, Revista Eletrônica De Sistemas De Informação Vol. 3, Nº1.
- Game-kitchen (2019) “An Ide For The Popular Game Development Framework Löve”, Bitbucket. Disponível Em: <<https://bitbucket.org/kevinclancy/game-kitchen/wiki/home>> Acessado Em 20 De Julho De 2019.
- Guana, V., Stroulia, E. and Nguyen, V. (2015) “Building A Game Engine: A Tale Of Modern Model-riven Engineering”, 4th International Workshop On Games And Software Engineering.
- Lewis, C, Whitehead, J. (2011) “The Whats And The Whys Of Games And Software Engineering”, 1st International Workshop On Games And Software Engineering, New York, Nt, Usa, Acm, Pp. 1–4.
- Lin, Y. H., Ko, T. M., Chuang, T. R. and Lin. K. J. (2006) “Open Source Licenses And The Creative Commons Framework: License Selection And Comparison”, Journal Of Information Science And Engineering Vol.22, Nº.1-17.
- Love Community (2019) “Framework Love”, Love. Disponível Em <<https://love2d.org>>. Acessado Em 20 De Junho De 2019.
- Kulchenko, P. (2019) “ZeroBrane Studio Lightweight: Ide For Your Lua Needs”, ZeroBrane Disponível Em: <<https://studio.zerobrane.com/>>. Acessado Em 20 De Julho De 2019.
- Neoreason (2019) “The World’s Leading Online Coding Platform”, Repl.it. Disponível Em: <<https://repl.it/>>. Acessado Em 20 De Julho De 2019.