

Programação Web para Jogos



Cruzeiro do Sul Virtual
Educação a distância

Material Teórico



Introdução ao Desenvolvimento de Jogos para Web

Responsável pelo Conteúdo:

Prof. Me. Alcides Teixeira Barboza Junior

Revisão Textual:

Prof.^a Esp. Kelciane da Rocha Campos

UNIDADE

Introdução ao Desenvolvimento de Jogos para Web



- [Introdução;](#)
- [O Surgimento da Internet;](#)
- [A Criação da Web;](#)
- [Arquitetura Cliente-Servidor;](#)
- [Tecnologias para o Desenvolvimento de Jogos Baseados em Web.](#)



OBJETIVO DE APRENDIZADO

- Apresentar uma breve história do surgimento da Internet;
- Apresentar como a Web foi criada e sua relação com a Internet;
- Discutir sobre as tecnologias de desenvolvimento web e sua utilização junto à arquitetura cliente-servidor;
- Apresentar algumas ferramentas para desenvolvimento de jogos para web.



Orientações de estudo

Para que o conteúdo desta Disciplina seja bem aproveitado e haja maior aplicabilidade na sua formação acadêmica e atuação profissional, siga algumas recomendações básicas:

Determine um horário fixo para estudar.

Mantenha o foco! Evite se distrair com as redes sociais.

Procure manter contato com seus colegas e tutores para trocar ideias! Isso amplia a aprendizagem.

Seja original! Nunca plágie trabalhos.

Aproveite as indicações de Material Complementar.

Conserve seu material e local de estudos sempre organizados.

Não se esqueça de se alimentar e de se manter hidratado.

Assim:

- ✓ Organize seus estudos de maneira que passem a fazer parte da sua rotina. Por exemplo, você poderá determinar um dia e horário fixos como seu “momento do estudo”;
- ✓ Procure se alimentar e se hidratar quando for estudar; lembre-se de que uma alimentação saudável pode proporcionar melhor aproveitamento do estudo;
- ✓ No material de cada Unidade, há leituras indicadas e, entre elas, artigos científicos, livros, vídeos e sites para aprofundar os conhecimentos adquiridos ao longo da Unidade. Além disso, você também encontrará sugestões de conteúdo extra no item **Material Complementar**, que ampliarão sua interpretação e auxiliarão no pleno entendimento dos temas abordados;
- ✓ Após o contato com o conteúdo proposto, participe dos debates mediados em fóruns de discussão, pois irão auxiliar a verificar o quanto você absorveu de conhecimento, além de propiciar o contato com seus colegas e tutores, o que se apresenta como rico espaço de troca de ideias e de aprendizagem.

Contextualização

O mundo dos jogos digitais é fascinante, você não acha? Você deve adorar jogar e passar horas na frente do jogo, certo? Tenho certeza de que, em algum momento, você se perguntou: como se programa um jogo?

Podemos desenvolver jogos para dispositivos móveis, consoles como Xbox, Playstation etc. Também podemos desenvolver jogos para PC e principalmente os jogos para Web, que rodam nos navegadores.

Como você irá perceber, existem diferentes formas para criar um jogo. Pensando, por exemplo, nos jogos para Web, existem ferramentas como o GDevelop, no qual você utiliza lógica para desenvolver o jogo e não manipula nenhuma linguagem de programação direta. Já em outros casos, podemos utilizar a Unity, por exemplo, e programar utilizando a linguagem C# para criar o jogo e publicar o mesmo para Web.

Nesta unidade, iremos conhecer os princípios que norteiam a chamada Web e, principalmente, quais são as tecnologias que podemos utilizar, focadas em programação, para desenvolvemos os nossos jogos, que irão ser disponibilizados e executados tendo como meio a Web.

Vamos começar!

Introdução

Há pouco tempo houve um grande avanço dos dispositivos móveis e das diversas plataformas oferecidas para se jogar. Com esse novo mercado, foram criadas novas ferramentas e recursos para auxiliar na produtividade e desenvolvimento de jogos, fazendo com que os desenvolvedores gastassem menos tempo com sistemas de física, animações e efeitos criados do zero, por exemplo.

Iremos abordar nesta unidade alguns conceitos básicos para o desenvolvimento e programação de jogos, apresentar as várias ferramentas e tecnologias usadas no mercado para a criação de jogos para a Web.

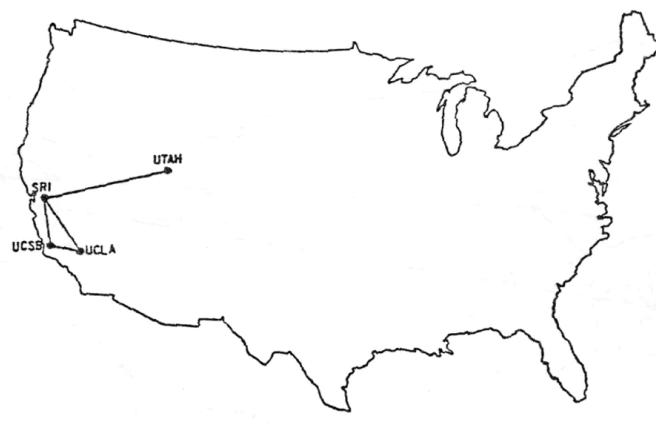
Antes de apresentarmos as tecnologias ou ferramentas para desenvolvimento de jogos para Web, precisamos entender um pouco sobre o surgimento da Internet, e principalmente da Web, e como podemos trabalhar nesse meio.

O Surgimento da Internet

A Internet teve início devido à necessidade de comunicação humana. A história começa em 1969 com a Darpa - Agência Nacional de Defesa - criada pelo exército dos Estados Unidos para a criação de tecnologias durante a Guerra Fria.

A missão da Darpa era interligar quatro computadores diferentes e de lugares distintos. Foi criada, então, a rede chamada ARPANET, em 1969, a primeira forma de comunicação antes da Internet que funcionava através de um sistema conhecido como chaveamento de pacotes, que é um sistema de transmissão de dados em rede de computadores no qual as informações são divididas em pequenos pacotes.

Os quatro computadores interligados na ARPANET pertenciam a universidades e centros de pesquisa, Figura 1.



The ARPANET in December 1969

Figura 1 – Computadores interligados, ARPANET 1969

Fonte: classes.design.ucla.edu

Com o tempo, a ARPANET evoluiu e foi ganhando mais computadores por meio de outras formas de conexão, como satélites. Em questão de 4 anos, o que eram quatro computadores passou para mais de vinte computadores espalhados pelos Estados Unidos, conforme pode ser notado na Figura 2.

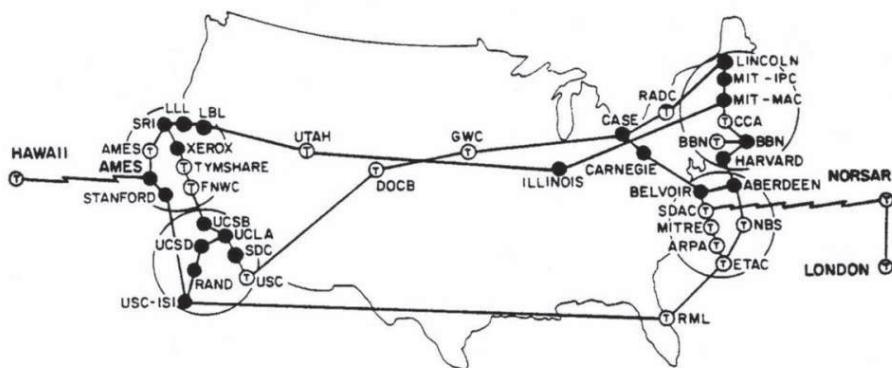


Figura 2 – ARPANET 1973

Fontes: bytes.usc.edu

Com o seu crescimento, os envolvidos na criação da ARPANET perceberam que seriam necessárias algumas mudanças. No seu primórdio, como já comentamos, a ARPANET era controlada por militares, de forma centralizada, porém com sua expansão perceberam que isso não seria muito usual e decidiram descentralizá-la e, assim, organizações poderiam criar suas redes e se conectar às demais já existentes.

Veja que com essa descentralização e com o surgimento de diferentes organizações, tinham que garantir que todas as redes e as possíveis novas redes se comunicassem, então os cientistas da época (Robert Kahn e Vint Cerf) foram acionados para que criassem um padrão para a comunicação. Surge, então, o chamado TCP/IP, a ARPANET passou a utilizar esse padrão a partir de 1983. Basicamente, o padrão criado especifica o formato dos pacotes de dados que são transmitidos pela Internet.

A partir do momento em que organizações entraram no jogo, novas redes foram criadas e conectadas, o que começou a dar corpo para a Internet que conhecemos hoje em dia. Contudo, foi somente após 1988 que se criou um movimento para tornar a Internet comercial, pois até então ela era destinada somente aos centros de pesquisa e organizações.

Foi em 1993 que a Internet se tornou de fato comercial, abrindo a possibilidade para diferentes países criarem suas redes e se conectarem na grande rede.



Definição de Internet: rede mundial de computadores, na qual redes de todo o planeta são interligadas, que permite a troca de dados e mensagens através de um protocolo comum.



Veja as conexões submarinas que interligam continentes em: <https://goo.gl/MnNwQf>.
O funcionamento da Internet pode ser visualizado no infográfico disponível em:
<https://goo.gl/mDE2yu>.

A Criação da Web

Como você percebeu, a Internet é uma rede composta por diversas redes e fornece um meio para comunicação entre todos os computadores que estão ligados a ela. Porém, precisamos entender que a Internet disponibiliza diferentes serviços nessa comunicação, como, por exemplo, transferência de arquivos usando o FTP (*File Transfer Protocol*), envio de e-mails usando o protocolo SMTP (*Simple Mail Transfer Protocol*), entre outros.

Assim, em meados de 1989, o cientista Tim Berners-Lee, em seu trabalho de doutorado, deu início ao projeto que denominou de *World Wide Web* (WWW) no laboratório CERN na Europa. O projeto WWW basicamente consistia em disponibilizar por meio da Internet documentos no formato de hipertextos (documentos compostos por textos, imagens, recursos multimídia e principalmente links, ligações entre diferentes documentos). Para se ter acesso a esses documentos, os usuários precisavam de um programa chamado de navegador.

O pesquisador Tim Berners-Lee desenvolve, então, em 1990, o primeiro servidor Web e seu navegador, intitulado de *World Wide Web*. Para que pudesse ter um produto de prova de conceito do seu projeto de doutorado, o pesquisador desenvolveu também a primeira linguagem para criação de documentos de hipertexto, que denominou de HTML 1.0 (*HyperText Markup Language*).

É importante ressaltar que nesse projeto, o pesquisador fez uso de um protocolo chamado de HTTP (*Hypertext Transfer Protocol*), protocolo usado para transferência dos documentos em formato de hipertextos.

Você deve ter notado que está tudo interligado, ou seja, temos nosso meio, que é a Internet, que fornece diversos serviços - neste caso, estamos interessados no serviço web, que seria a disponibilização de documentos, páginas, em formato de hipertexto, que fazem uso do protocolo HTTP e, de forma geral, da linguagem HTML. Logo, nossos jogos serão desenvolvidos para a Web utilizando tecnologias próprias para esse meio.



Veja a evolução e as versões dos navegadores que foram desenvolvidos desde o surgimento do primeiro navegador *World Wide Web* em 1991. Acesse: <https://goo.gl/JHhSqb>.



A primeira página HTML criada pode ser visualizada no link disponibilizado pelo laboratório CERN em: <https://goo.gl/BVQ9kv>.

World Wide Web Timeline - veja a linha do tempo da evolução da WWW em: <https://goo.gl/Q5LzRu>.

Linha do tempo da história da Internet disponível em: <https://goo.gl/ddrG2J>.

Arquitetura Cliente-Servidor

Juntamente com a criação do HTTP e do HTML, Tim Berners-Lee criou uma rede virtual de hipertexto chamada *World Wide Web* (WWW), que engloba todos os servidores de conteúdo multimídia que utilizam o protocolo HTTP.

Para uma página web funcionar, há a necessidade de utilizar um navegador (browser), como, por exemplo, o Google Chrome, Firefox, Internet Explorer, Safari, entre outros. A comunicação é feita pelo protocolo TCP/IP, onde é usado um protocolo de alto nível, o qual já foi mencionado, o HTTP.

A Web funciona conforme o modelo cliente-servidor (Figura 3).

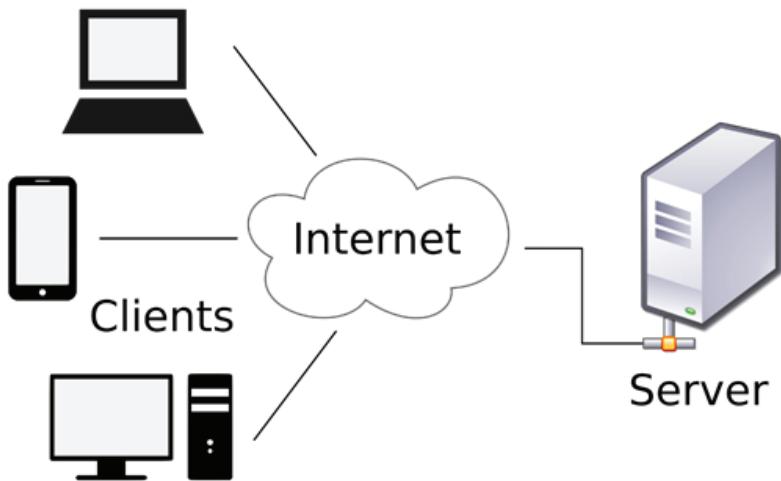


Figura 3 – Arquitetura cliente-servidor básica

Fonte: Wikimedia Commons

Nessa arquitetura, existem uma ou mais máquinas que atuam como servidores, disponibilizando recursos para as demais máquinas, as quais atuam como clientes. Em geral, os servidores são equipamentos com maior poder de processamento e armazenamento e os clientes, geralmente, são microcomputadores ligados em rede.

Basicamente, funciona da seguinte forma: o cliente tem a função de fazer uma solicitação de, por exemplo, um arquivo HTML ao servidor, deve ser capaz de receber este arquivo e interpretá-lo através do navegador. Junto com a solicitação, são enviadas informações que compõem o arquivo de hipertexto. Dentre as informações, já vimos que existe a linguagem HTML, além dela também são enviadas para

o cliente as folhas de estilo (arquivos CSS) e interatividade do documento (arquivos JavaScript), que serão comentados ainda nesta aula, entre outros recursos. Desta forma, então, a função do servidor neste cenário é fornecer os dados e arquivos ao cliente.

Perceba que nosso foco será criar os jogos para serem disponibilizados por servidores e processados pelas máquinas clientes através do seu navegador. No entanto, é importante ressaltar que o servidor também pode fazer o processamento utilizando para isso tecnologias ou linguagens específicas, como PHP, JSP, inclusive com acesso aos chamados banco de dados, como, por exemplo, o MySQL. Note que podemos fazer uso de todas essas tecnologias para desenvolver nossos jogos, mas precisamos partir do início, então novamente estaremos interessados aqui no lado cliente dessa arquitetura cliente-servidor.

Na próxima seção, veremos quais as tecnologias que irão nos dar suporte para que possamos desenvolver nossos jogos para a Web utilizando programação.

Tecnologias para o Desenvolvimento de Jogos Baseados em Web

O desenvolvimento de jogos para Web pode fazer uso de diversos tipos de ferramentas ou softwares. Estamos interessados aqui no desenvolvimento por meio da programação e que esta utilize as tecnologias web para disponibilização dos nossos jogos.

Antes do aprimoramento da linguagem HTML para a versão 5, a maioria dos jogos para a Web era desenvolvida usando-se o software Adobe Flash, uma tecnologia que atualmente não é mais suportada pelos navegadores. Com a chegada do HTML5 (2008) e de APIs como o Canvas e o Local Storage, conceitos que serão estudados posteriormente, desenvolver jogos utilizando as linguagens HTML, CSS e JavaScript tornou-se uma opção para quem deseja desenvolver jogos para a Web.

Como dito anteriormente, o HTML é uma linguagem padrão para a construção de documentos com hipertexto desenvolvido por Tim Bernes-Lee. Essa linguagem está em sua 5^a versão e é utilizada para marcação e para estruturação e apresentação de conteúdo na *World Wide Web*. Oficialmente, o início da sua especificação foi em 2008, desde então essa nova versão do HTML (Figura 4) trouxe novas funcionalidades, como semântica, acessibilidade e novos recursos atrelados à linguagem JavaScript.

A linguagem HTML é a nossa base, a partir dela estruturamos nossos documentos de hipertexto, ou páginas HTML, porém ela não trabalha sozinha, temos que conhecer ainda o CSS (*Cascading Style Sheets*) e o JavaScript, para então termos as tecnologias básicas para o desenvolvimento dos jogos.



Figura 4 – Logo oficial do HTML

O CSS (em português, folhas de estilo em casca-ta) é uma linguagem separada do HTML que está na sua versão 3, é utilizada para formatação visual e estilização de uma página feita em HTML. Dentre os recursos, citamos alguns, como tipo de fonte, cores de fontes ou fundo, espaçamentos, entre outros. Diferentemente do HTML, que é utilizado para estruturar as páginas, definindo parágrafos, listas, tabelas, cabeçalhos etc., o CSS é utilizado para formatar o visual desta página, logo exibido na Figura 5.

Já o JavaScript é uma linguagem de *script* que torna a página HTML dinâmica. Mas o que é tornar uma página dinâmica? Se usado somente o HTML e o CSS, as nossas páginas web seriam totalmente estáticas, sem nenhuma interatividade. Adicionando a linguagem JavaScript a elas, a página ou nosso jogo terão interatividade e, o mais importante, funcionalidade.

Esses conceitos serão trabalhados nas próximas unidades desta disciplina. No momento, você precisa só saber distinguir as três, ou seja, qual a função de cada uma delas quando criarmos nossos jogos.

Existem também ferramentas ou softwares no mercado que nos auxiliam no desenvolvimento de jogos para a Web, inclusive alguns desses softwares trabalham diretamente com essas três linguagens citadas. Nas próximas seções veremos algumas opções.

Godot Engine

Godot é uma Game Engine (ferramenta de desenvolvimento de jogos) feita para desenvolver jogos em diversas plataformas, como computadores, dispositivos móveis e Web. É uma ferramenta gratuita e open source (código aberto).

Godot é um programa bem leve, que pode ser executado em Windows, Mac ou Linux e possui um mecanismo de design de renderização para modos 2D ou 3D. Na programação, o Godot utiliza uma linguagem própria, a GDScript (Figura 6), que é muito semelhante às linguagens Python, JavaScript e PHP. A nova versão desta ferramenta permite utilizarmos também a linguagem C#.



Figura 5 – Logo oficial do CSS

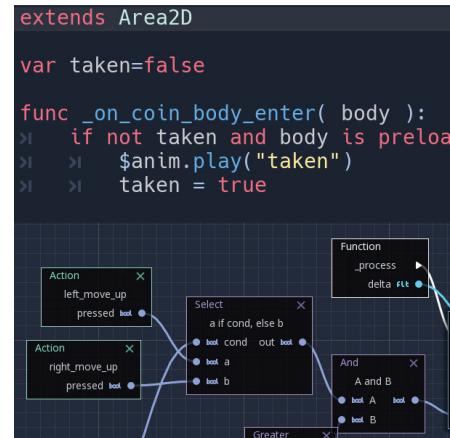


Figura 6 – Linguagem GDScript, GODOT
Fonte: godotengine.org



Conheça mais sobre o GODOT em: <https://goo.gl/QL27P7>.

Unity

A Unity (Figura 7) é uma ferramenta multiplataforma, para jogos que serão publicados em dispositivos móveis, computadores (desktop), aparelhos que suportam realidade virtual e realidade aumentada, consoles, web, smart-TVs, que permite criar jogos em 2D ou 3D. Essa ferramenta tem uma versão gratuita, que por sinal permite utilizar muitos recursos sem a necessidade de investir nas versões pagas. É possível baixar no Windows, Mac e Linux.



Figura 7 – Unity Engine

A Unity permite a especificação de configurações de compactação e de resolução de textura para cada plataforma em que o jogo está sendo desenvolvido. Oferece suporte para mapeamento de colisão, de reflexão, parallax, sombras dinâmicas utilizando mapas de sombra, entre muitos outros recursos.

A linguagem utilizada é o C# para a programação. E para jogos desenvolvidos para a Web, ela é compatível com os principais navegadores (Internet Explorer, Firefox, Safari, Opera, Google Chrome) através do uso do plugin Unity Web Player.



Unity: <https://goo.gl/BpmYCC>.

Veja o vídeo: <https://youtu.be/DDsRfbfnCA>, para entender por que essa é uma das ferramentas preferidas dos desenvolvedores de jogos de todo mundo.

Unreal

Assim como a Unity, a Unreal Engine (Figura 8) também permite criar jogos para diversas plataformas, desde smartphones até consoles e aparelhos que exploram a realidade virtual. Os códigos da programação são todos feitos em C++.



Figura 8 – Engine Unreal

A engine possui sistema de renderização, sistema de física, detecção de colisões, inteligência artificial, entre outros recursos.



Unreal: <https://goo.gl/BygYWH>.

Veja o vídeo: https://youtu.be/5R_dniXqMvU, para entender por que essa é uma das ferramentas mais utilizadas para o desenvolvimento de jogos em grandes e pequenos estúdios.

A
Z

Game Engine: nomenclatura atribuída às ferramentas de desenvolvimento de jogos que fornecem diferentes pacotes de recursos ou funcionalidades visando facilitar o desenvolvimento dos jogos. As engines atuais demoraram muito tempo para chegar no ponto em que estão, foram anos de evolução, o que se traduziu em ferramentas repletas de recursos, que vão desde a parte visual até a utilização de inteligência artificial de forma simplificada.

Frameworks para Desenvolvimento de Jogos Web

Podemos tanto trabalhar com os softwares citados anteriormente para desenvolvemos um jogo, como podemos também utilizar frameworks como suporte, que irão auxiliar no desenvolvimento economizando muito tempo, pois fornecem códigos prontos, testados e feitos para atender às necessidades comuns para a criação de um jogo web.

Dessa forma, reunimos nas próximas seções alguns frameworks de JavaScript para desenvolvimento de jogos customizados e que fornecem uma estrutura para construir jogos baseados em navegador usando a tecnologia padrão da Web.

Phaser

O Phaser (Figura 9) é um framework open source com recursos de tratamento de colisões, simulação de física, entre outros. Permite criar jogos para desktop (navegador web) ou dispositivos móveis com uso de HTML5.

É possível criarmos jogos usando WebGL ou Canvas no HTML. No site do Phaser, junto com a documentação, é possível também acompanhar diversos exemplos de jogos desenvolvidos com o framework.

Nas próximas unidades, utilizaremos este framework para criarmos um jogo. O Phaser tem recebido diversas atualizações recentes, o que nos mostra que esta ferramenta tem ganhado contínuas melhorias para desenvolvermos um jogo para Web. É uma boa opção para iniciarmos o entendimento referente à programação de jogos para Web usando JavaScript.



Figura 9 – Phaser Framework



WebGL é uma API (*Application Programming Interface*) de JavaScript disponibilizada por meio do elemento Canvas do HTML5. Permite criar jogos 2D e 3D de alta performance. Para conhecer mais, acesse: <https://goo.gl/yXRnps> e <https://goo.gl/BZFfoo>.

Phaser Framework - acesse o link: <https://goo.gl/xVX8uM>, para conhecer os recursos e a documentação deste framework.

Cocos2d-x

Outro framework interessante é o Cocos2d-x (Figura 10). É um framework desenvolvido em C++ que dá suporte à criação de jogos para diferentes plataformas, como desktop, dispositivos móveis, consoles e Web. É possível desenvolver em C++, JavaScript ou Lua neste framework.

No site do framework, também temos disponível o Cocos Creator (Figura 11), uma ferramenta que inclui uma game engine baseada no Cocos2d-x. A ferramenta fornece todo o suporte para o desenvolvimento de um jogo, desde a criação de cenas até a publicação do jogo final. Na ferramenta criamos os nossos scripts com a linguagem JavaScript, inclusive podemos escrever extensões em JavaScript de forma a aprimorar funcionalidades da ferramenta.



Figura 10 – Logo do Cocos2d-x
 Fonte: Divulgação

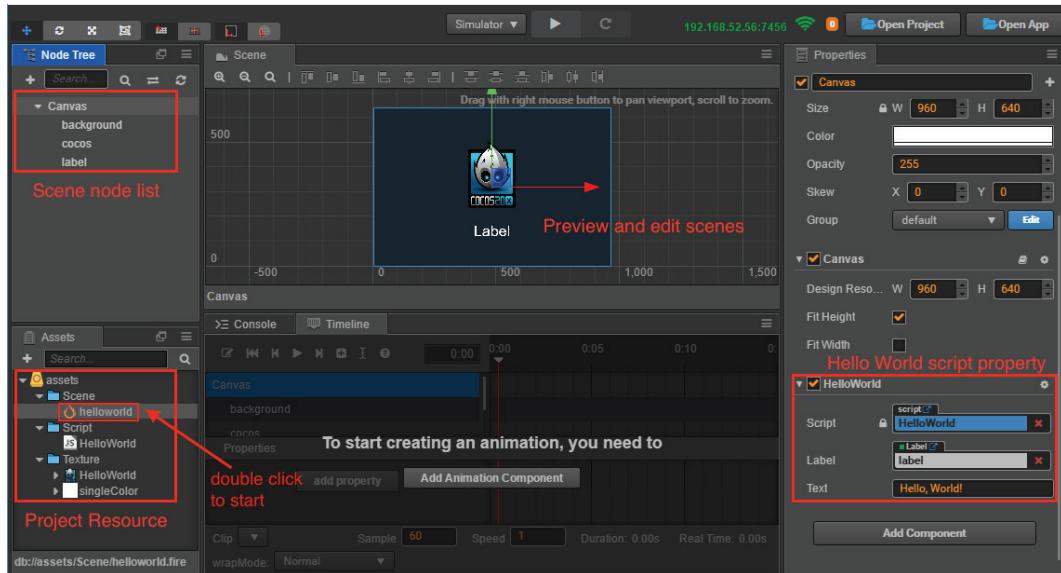


Figura 11 – Cocos Creator

Fonte: Cocos 2d-x



Cocos2d-x - para conhecer mais sobre esse framework, acesse: <https://goo.gl/F4kYuZ>.

Documentação do Cocos Creator disponível em: <https://goo.gl/sxgdZ1>.

Para conhecer o Cocos Creator, veja o vídeo em: <https://youtu.be/UTy7IkGxMU>.

MelonJS

Um framework bem interessante e ao mesmo tempo simples em questão de aprendizagem é o MelonJS. Este framework possui diversos recursos para desenvolvermos jogos 2D como física, que usa pouco recurso de processamento, WebGL, WebAudio, entre outros.

Um ponto favorável deste framework é que ele permite integração e suporte com outras ferramentas, como PhysicEditor, Cordova/PhoneGap, etc. Outro fato interessante é ter integração com o formato Tiled Map, o que torna fácil a criação dos designs dos níveis para os jogos 2D.

É possível, por exemplo, criar um tiled map do jogo no Tiled Editor (Figura 12) e posteriormente utilizar esse mapa juntamente com o MelonJS.



Conheça mais sobre o MelonJS no site: <https://goo.gl/iepWYD>.

No link: <https://goo.gl/uhpYjd>, você encontrará diversos exemplos de jogos feitos com o recurso de tiled map.

Documentação do MelonJS disponível em: <https://goo.gl/Ef6R9c>.

Tutorial de um jogo de plataforma feito com MelonJS e Tiled Editor: <https://goo.gl/3Q5t2P>.

Tutorial para criar o jogo Space Invader com o MelonJS: <https://goo.gl/9XgXxK>.

Para conhecer o Tiled Editor, acesse: <https://goo.gl/3jxEaB>.

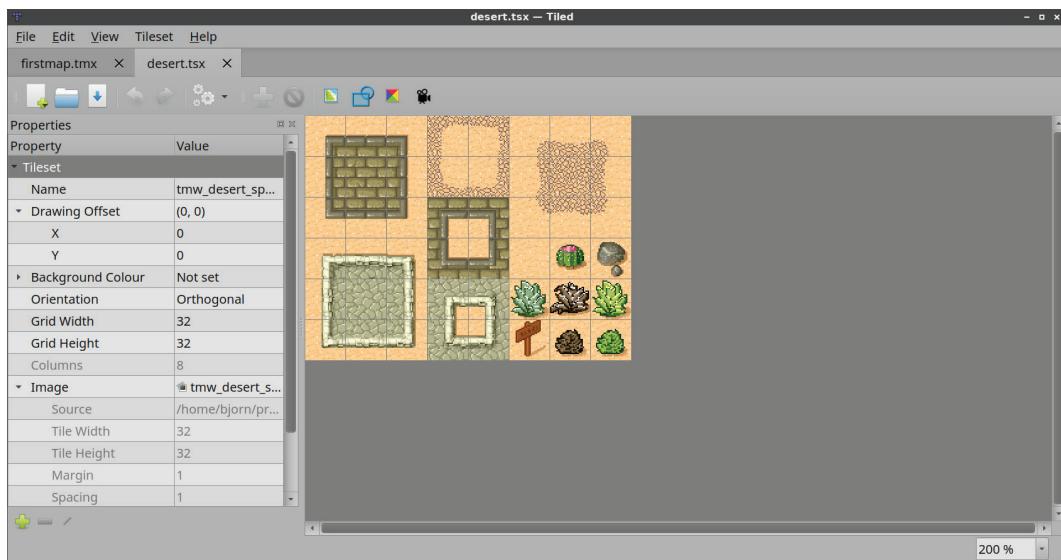


Figura 12 – Tiled Editor

Fonte: doc.mapeditor.org

Agora que já entendemos os conceitos básicos no desenvolvimento de jogos para Web e suas principais tecnologias, vamos mudar de nível. Nossa próximo nível será utilizar as tecnologias HTML, CSS e JavaScript na sua forma mais simples para criar nosso primeiro jogo.

Material Complementar

Indicações para saber mais sobre os assuntos abordados nesta Unidade:



Sites

W3SCHOOLS HTML: the language for building pages

<https://goo.gl/MfDkRE>

Sololearn – Everyone can code

<https://goo.gl/SFYSLF>

Codecademy

<https://goo.gl/tcgk9v>

Phaser Framework

<https://goo.gl/8vuCJq>

Referências

BUCHARD, E. *The web game developer's cookbook*: using JavaScript and HTML5 to develop games. Addison-Wesley, 2013.

RAMTAL, Dev; DOBRE, Adrian. *Physics for JavaScript games, animation, and simulations*: with HTML5 canvas. New York: Apress, 2014.

SILVA, Mauricio Samy. **Html 5**: a linguagem de marcação que revolucionou a Web. São Paulo: Novatec, 2011.

Sites Visitados

WORLD WIDE WEB CONSORTIUM W3C. Disponível em: <<https://www.w3.org/html/>>. Acesso em: 09 jun. 2018.



Cruzeiro do Sul
Educacional