

INDIE GAMES DEVELOPER

MAGAZINE

Issue #3



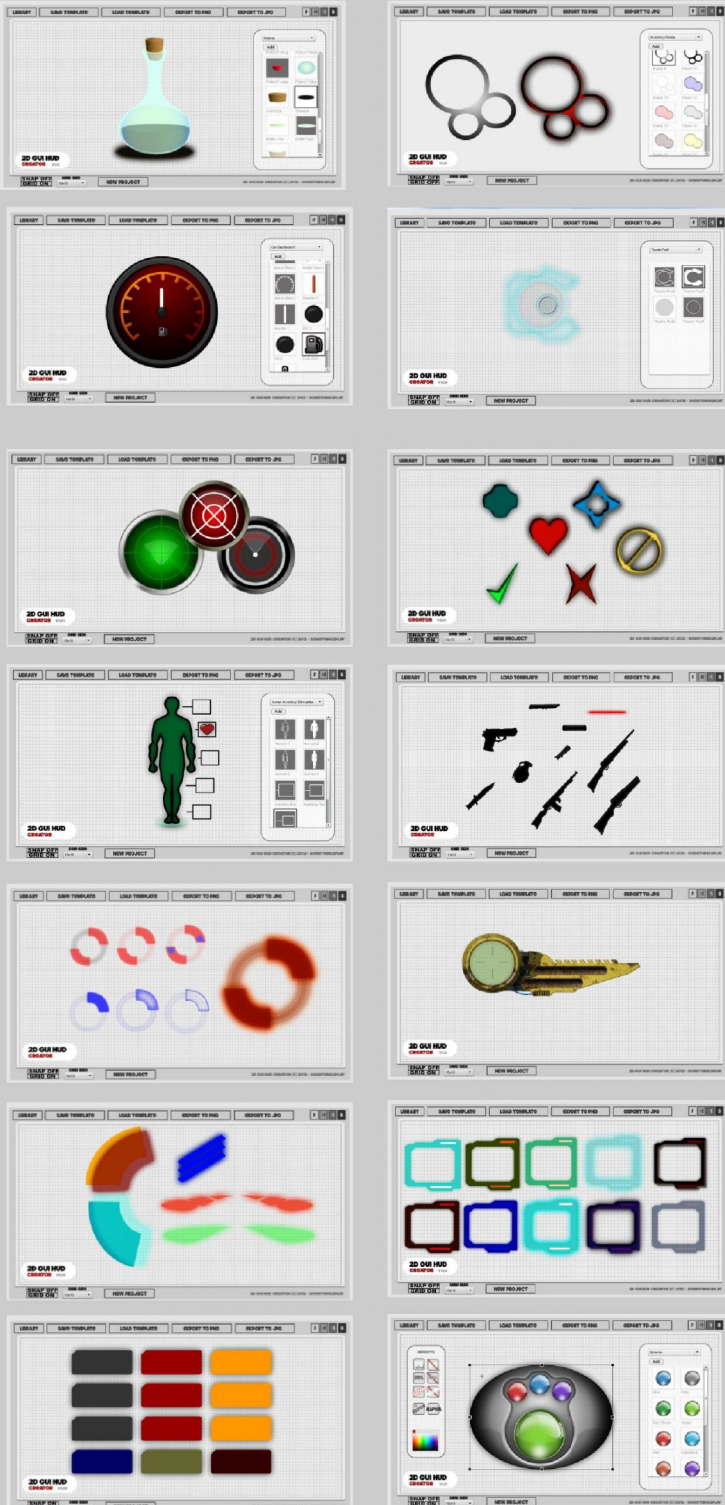
Construct 2

CONSTRUCT 2
CUBEMEN
FRAY
30 MINUTES OR LESS
TRANSCRIPTED
FRUIT ON RAILS
DERRICK THE DEATHFIN

GT OMEGA RACING
GUNSWORDS
DARK SCAVENGER
WOODEN SEN'SEY
JUMPING ELECTRON
ITLLE DEW

TUTORIALS
COMPETITIONS
IGD AWARDS
INDIE UPDATES
AND MORE!

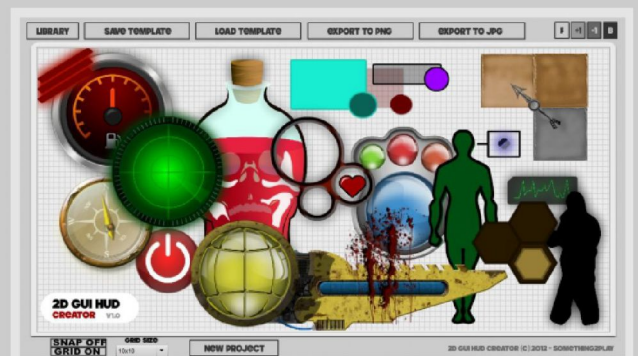
2D GUI HUD CREATOR



2D GUI HUD **Creator** is an easy to use artwork tool that comes complete with a growing resource library of GUI and HUD artwork that you can use in the games you develop. Royalty FREE!

2D GUI HUD **Creator** has been designed to be an easy to use artwork tool for game developers that have little or no artist skills, but would like to be creative and efficient in the creation of GUIs and HUDs for their games currently in development.

Included; is a growing Artwork Library of resources to help you rapidly create prototypes of game GUIs and HUDs saving you valuable time and money, allowing you to concentrate on other areas of game development, and without the initial cost.



WWW.2DGUIHUD.COM



CONTENTS

PART 1

2

2D GUI HUD CREATOR

sponsor

5

WHATS NEW

9



12



15



17



20

[TRANSCRIPTED]

33



36

30 Minutes...
or Less!



6

Construct2 - plus
3 Business Licences
to win!

23

Character creation
Tutorial Series by
Richard Cook: Part 2



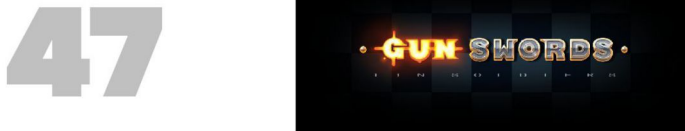
INDIE GAMES DEVELOPER



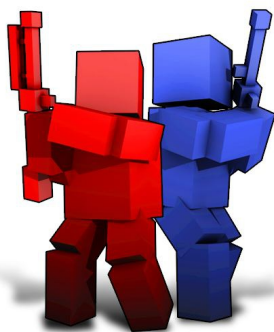
CONTENTS

PART 2

40 INDIE UPDATES

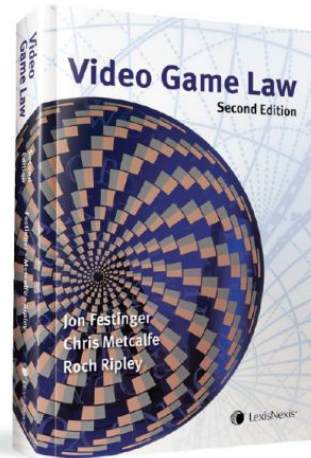


58 EVENTS DIARY



39

advert



VISIT OUR
EVENTS DIARY
ONLINE
WE ADD WORLDWIDE
EVENTS DAILY
FEATURED + LINKS

60

advert

GREEDY BANKERS
VS
THE WORLD

Available on the App Store

★★★★★ "App of the Week"
GameZbo, Sep 2011 Eup! Magazine, Oct 2011
"The [iPad] multiplayer is one of the best on iOS devices"
148 Apps, Nov 2011
"Simple to describe but complex to master...
Greedy Bankers delivers in short, healthy doses"
DIYGamer, Feb 2011

EUROGAMER INDIE GAMES
ARCADE FINALIST 2011

Created by indie developer Alastair Aitchison,
named as one of game development's rising stars
in Develop Magazine's 30 Under 30, Jan 2012

Now FREE on iPhone!

Follow the developer on Twitter: @alastairaitchison
Videos, info and reviews: www.greedy-bankers.com

INDIE GAMES DEVELOPER



WHAT'S NEW

Enquiries: info@indiegamesdeveloper.com
Press Releases: press@indiegamesdeveloper.com
Advertising: advertising@indiegamesdeveloper.com



BIGGER AND BETTER!

We've had time to reflect on the past two issues of our magazine, taken onboard all the feedback we've received, both positive and critical. The result: we have expanded our magazine to a whole new level!

Since issue 2 we made the decision to take our free magazine monthly and to set a concrete date for when you can expect to download it. This date will be the first of every month, even if it falls on a weekend!!

We really appreciate all the very encouraging feedback, and thank everyone who has taken the time to contact us, like us on Facebook and ReTweet / Follow us on Twitter and last but not least G+1 us :)

Enjoy our 3rd bumper packed issue!



WEBSITE: STAGE 2 LAUNCHED

Stage 2 of our website has now launched! We will be accepting registrations from the 10th of July 2012. Our website will cater for developers, gamers, and associated professionals.

www.indiegamesdeveloper.com

Keep updated on our progress and follow us on Twitter @igdmag



EVENTS DIARY UPGRADE

Our new Worldwide Events Diary is now being upgraded! We will be listing a number of events each month as standard entries, with the option to upgrade to a featured showcase.

If you are an event manager/organiser or representative be sure to send in your events, however big or small, to: events@indiegamesdeveloper.com and we'll add your entry.

www.indiegamesdeveloper.com/events-diary

All information contained within this magazine is for information purposes only. Whilst we work with our sources to ensure that the information and content provided at the time of release is correct and to the best of our knowledge. Indie Games Developer (IndieGamesDeveloper.com Ltd), its magazine, group of websites, contributors, editors, employees and consultants cannot accept any responsibility for errors or inaccuracies that may occur. Readers are advised to contact game developers, manufacturers of software, hardware, distributors, service providers and retailers directly for further information, pricing and technical support. Even though every care is taken, neither IndieGamesDeveloper.com Ltd nor its website(s), magazine(s), contributors, editors, employees and consultants shall be liable for any loss or damage. IndieGamesDeveloper.com Ltd authors and contributors reserve their rights in regards to copyright of their work. No part of this work covered by the copyright may be reproduced or copied in any way. Our magazine will always be in digital format. Whether viewable online or through download. If you prefer to print the magazine for reading, we encourage you to recycle the paper afterwards. Thank you!





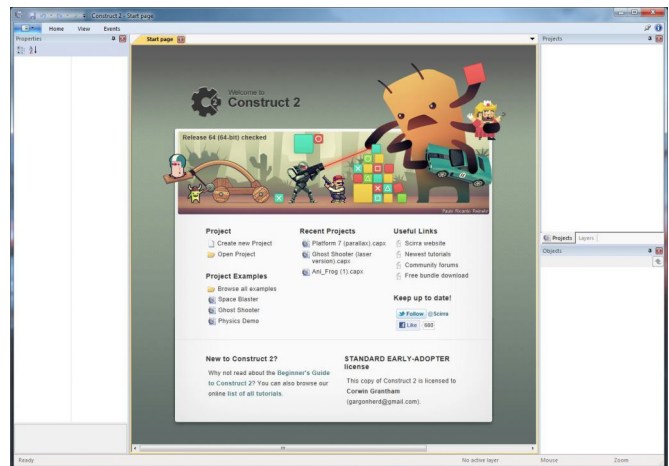
Ashley and Tom Gullen, of Scirra, are the brains and driving force behind the developer friendly Construct 2 engine. Tom took some time out of what sounds like an incredibly frantic schedule to share some of the latest developments. However, at the speed Scirra offers up new versions and extra features and capabilities there has probably been a new version while you have been reading this!!!

Scirra was started in 2007 by a team of students working in their spare time developing an open source project called Construct Classic, a non-programming game engine. Construct Classic has had over half a million downloads to date.

Late in 2010 Ashley decided to independently start work on Construct 2. One common complaint of Construct Classic was that it was relatively unstable so Construct 2 was re-engineered from the ground up to address these design issues. After a few months developing it, Ashley invited me (his brother!) to join in and establish Scirra as a tech startup. Scirra was established as a limited company in May 2011 and a couple of months later we met with a passionate Angel Investor who helped fund Construct 2's growth.

We have recently been getting a lot of positive feedback and surprise from people when they realise Scirra is a 2-man company! We've recently taken on our first long term contractor Paulo Ricardo Reinehr from Brazil - a very talented graphic designer and Construct 2 user himself so now there are 3 people working full time for Scirra.

We've been working hard to grow Construct 2. When I first joined the website was receiving around 20,000 unique visitors and 300,000 page views a month. Only around 8 months on from there we're now receiving 115,000 unique visitors a month and 1,200,000 page views.



Construct 2 is a Windows program. We've received a bit of criticism for this decision as Construct 2 exports to HTML5. The reason Ashley chose to develop Construct 2 as a Windows program is because that's the area he is most experienced in, from the development of Construct Classic. Also at the time he decided it would be too risky to launch Construct 2 as a web app. Remember we are talking late 2010 here and back then there were not many web apps in the wild that matched the complexity of Construct 2. We also couldn't find many examples of complex web apps that had the responsiveness we enjoy with native Windows apps. It made sense to make Construct 2 for Windows. We are well aware however that there is a high demand for Mac/Linux versions of Construct 2, it's something we are asked about on almost a daily basis!

www.scirra.com

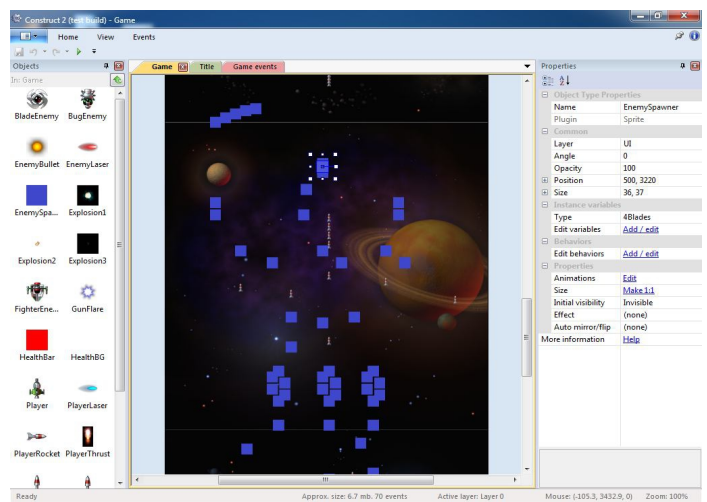
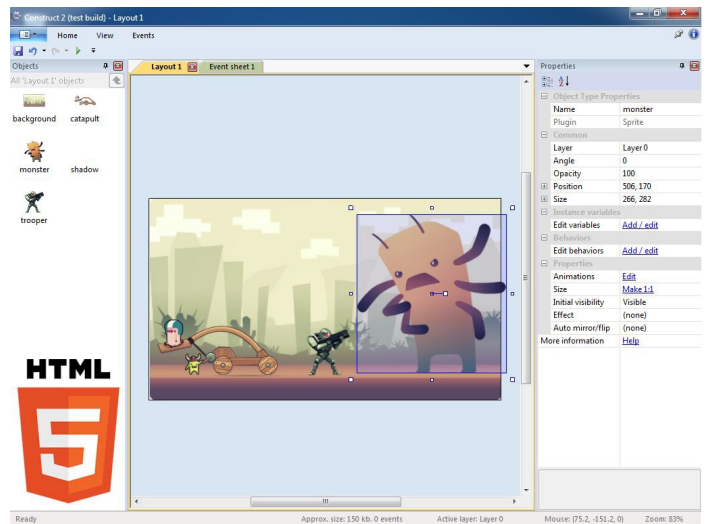


It was originally planned that Construct 2 would export to DirectX EXE games like Construct Classic does. However during development of Construct 2 we started reading a lot about HTML5. It seemed like an exciting technology. Not only was it exciting but we also believed in it. For that reason Construct 2 currently exports to HTML5 only. However HTML5 can be wrapped up in other ways, such as PhoneGap and appMobi for native phone apps, and we're experimenting with Awesomium which allows you to run HTML5 games in an EXE wrapper.

Deciding to export to HTML5 was also met with some resistance. I won't rehash the age old debate about HTML5 vs Flash but our position is that we believe HTML5 is certainly going to be the future of web gaming and dynamic content. We think our timing is pretty good in that regard.

Starting Scirra as a business for the first time has been a steep learning curve as we expected! We've faced a lot of challenges from every aspect you could imagine. Probably the most significant challenges have been on Ashley's end, dealing with endless reams of browser specific bugs and glitches. HTML5 audio is an area of HTML5 that we've researched and blogged about extensively. Audio in HTML5 is currently a spaghetti mess. However this is where Construct 2 really shows its advantages – leave all the fiddly bits of game development to us! We want Construct 2 to work all that out for game developers leaving them to do what they do best – make games!

We release 'early and often'. We've had over 40 releases of Construct 2 in 8 months, that's an average of 5 a month! Ashley who writes the engine itself has shown no signs of slowing down yet either – and lots more exciting features are planned! Unlike the legions of other game engines out there Construct 2 distinguishes itself as a 'no programming required' game engine. Games are built using 'events' which are sort of logical blocks. For example one event might read as "If RainDrop collides with Umbrella, destroy Raindrop". It's a more visual way of programming games.



However for those that wish to extend Construct 2's capabilities we have a fully documented Javascript SDK. This allows anyone armed with Notepad to write their own plugins and behaviours. Our community has come up with some great and very useful plugins which they share with everyone else!

Construct 2 is suitable for just about any type of 2D game: platform, fighter, isometric, birds-eye view etc!

The great thing about HTML5 is that it's a great technology for creating multiplayer games. Currently we have an AJAX object for basic multiplayer functions such as turn based games but we plan to fully support WebSockets in the near future.

Because Construct 2 uses a plugin based system to extend its capabilities it means it is capable of doing just about anything you can write in Javascript. We already have Facebook plugins that allow direct integration into Facebook apps.

We have also just released support for particle effects. These can create some very beautiful looking games.

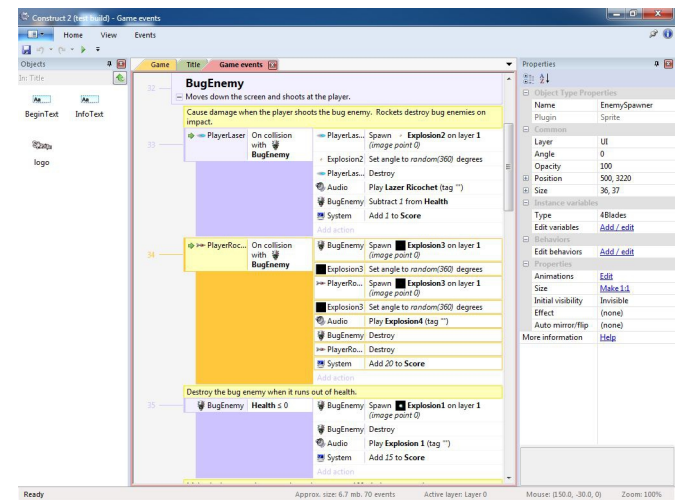
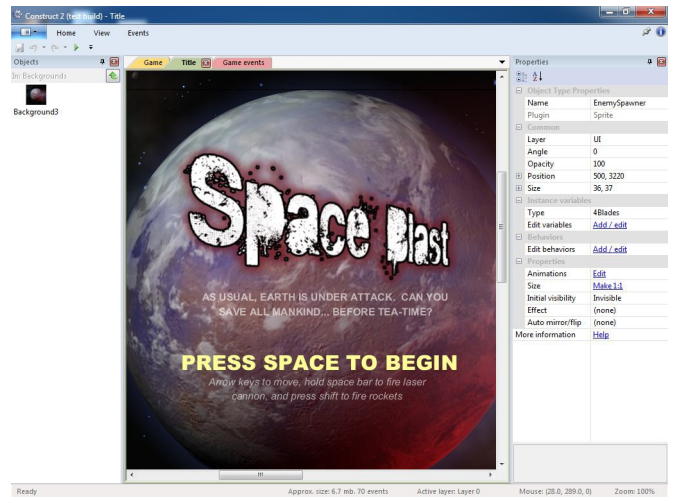
A summary of Construct 2 as an engine would be:

Platform: Windows XP, Vista, 7

Export: HTML5 (WebGL), Chrome Web Store, Facebook App, AppMobi, PhoneGap, Kongregate and soon also Windows 8 'Metro' apps

Features: Particle Effects, Box2D Physics, preview over Wifi, Javascript SDK, fully documented, offline support

Technical features: renders with WebGL where available (2-3x faster than Canvas 2D), uses the Web Audio API in Chrome and HTML5 audio recycling in other browsers for reliable sound effect playback



Example Games

Here is a list of my favourite games we currently have in our online HTML5 arcade:

Airscape

<http://www.scirra.com/arcade/games/addicting-rotary-games/848/airscape>

Magi

<http://www.scirra.com/arcade/action/495/magi>

Project Blaze Zero:

<http://www.scirra.com/arcade/addicting-shooter-games/349/project-blaze-zero>

Samira

<http://www.scirra.com/arcade/addicting-puzzle-games/144/samira>



ENTER OUR COMPETITION!

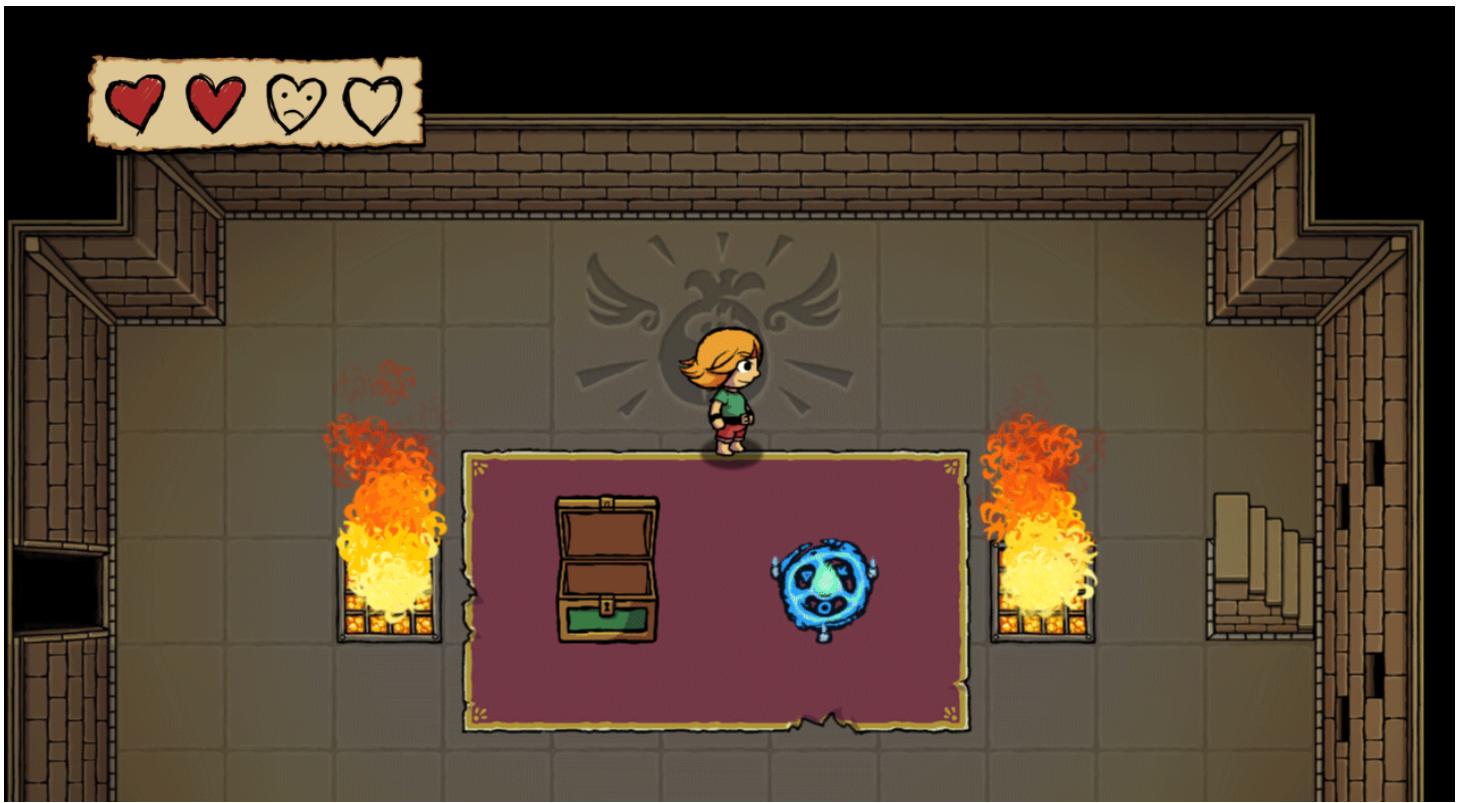
Indie Games Developer Magazine has teamed up with Scirra who are providing us with Three **Business Edition** Construct 2 licences. All you have to do is follow us on Twitter @igdmag and look out for our Tweets promoting this competition and RT it. We will then pick one winner each week randomly. For terms and conditions, visit our website: <http://www.indiegamedeveloper.com/scirra-competition>

INDIE GAMES DEVELOPER

LITTLE DEW



 Ludosity



“...the classic Zelda-formula boiled down to its very essentials”
That’s how Joel Nyström of Ludosity describes their soon to be completed game Ittle Dew.
He talks us through the journey the company has made to reach this point.

Ludosity is a small Swedish game studio that started out in 2008 as a flat organization made out of students - fresh out of university. There were quite a few of us, more than 10 people in fact. Over the first two years however we reduced the team size down to the very core team of talents before we started growing again. It was at that time it all clicked together and we felt we had the perfect team. It was a painful journey, especially firing friends, but we came out much stronger for it. Today we’re back up at 9 employees, not including freelancers and interns.

We’ve had a lot of help from the business incubator we reside in. Business coaching, office space, hardware, software were all provided for us the first year. Today we have “graduated” from the incubation to the science park proper. I can tell without a doubt that we would have not existed as the thriving company we are today without this incubation program. Though we might have been solo developers each on our own instead!

We almost always do many projects in parallel, it’s a great way to maintain productivity and keep energy-levels high. We also have a prototyping-effort called Ludosity Play where we publicly display all prototypes on our website.



All concepts are pitched internally by all team members and then voted on. The winning concept gets 1-2 weeks of development time and is then released to the public so we can measure how it works “in the wild”. The big idea is of course to find our next big thing to work on, but it’s also worth it for the energy it brings to the team alone.

Ittle Dew is both the title and the name for our heroine. Ittle is a bratty girl with absolutely no thought of anyone but herself. She’s out adventuring the vast seas one day when she crashes on an island. Immediately she starts breaking and entering every castle, hut and cave she sees to find any items that will help her fix her boat and get back home. The natives would probably have helped her if she asked, but she decides to kill them and steal their treasures instead. It’s very 90’s RPG-ish in that sense.



The gameplay could be said to be the classic Zelda-formula boiled down to its very essentials. We have made away with item swapping in menus and focus on just 4 items that are always-ready: The stick, the portal wand, the ice wand, and bombs. These items can be used together in every combination, and we use that to its fullest extent in the puzzles. The level design is really tight and built with extreme “replayability” in mind. You can actually finish the game with only two items, any two you like. You can also find a great many shortcuts and secrets made for speed-running the game. The level design is very dense, and there are a lot of layers of interactivity happening at once – though they may not be immediately apparent.

Our main focus with the game is the very core gameplay and the level design, with the latter taking up more than half of our development time.

We are pretty late-on in development now, with basically just bosses and hardware-specific features left to implement. As for what hardware features that may be, it’s something that is currently under wraps, but we hope to be able to make an announcement of platform and release date later this summer!



www.ludosity.com

INDIE GAMES DEVELOPER

We've heard from Alex Gold the CEO of Psydra Games LLC. He was also the Project Lead, Designer and Writer of Dark Scavenger.

How did you get started?

Psydra Games came about at a critical time in our careers. Jim Otermat (our lead coder) was in between jobs and I was employed, yet looking for new opportunities. Both of us sought a way to expand ourselves both professionally and as developers.

We had been discussing working on a project together for a while and it was finally the right time for us to spring into action.

How quickly did you expand?

While our operation initially started out with just the two of us, things took a dramatic turn when we brought on Sean Kearney, an excellent artist and colleague from my former job. Though we initially considered populating the game with my awful stick figures, the day our first piece of art came through, we knew we could make something much more ambitious. We quickly reached out to many friends of the artist persuasion to see if they wanted to get involved. Fortunately, the sheer craziness of our project offered plenty of opportunities for many of them to flex some creative muscles that they hadn't been able to use for a while. Soon enough, we had a plethora of talent, each handling different areas that appealed to their individual skill sets.

I was also in a band at the time. While preparing for our first (and only) show, our guitarist Joe Kelly caught wind of the project and offered his assistance on the audio front. Before any of us knew what was happening, production of our first real game was underway.

What challenges did you face when you first started?

The trickiest thing for me was teaching players to play the game. We had many complicated systems at work, many of which were unique to our title. It took months of play-testing and several changes to the user interface before we got it to an acceptable stage.

As the project went on and our art resources grew thinner, we ultimately abandoned the concept of animation and opted for a large quantity of beautifully crafted still-sprites. Through this, we were able to gain a large variety of enemies and NPC encounters, pinning most of the work on Design and Code to implement.

Finally, as there are over a hundred items in the game, continuing to create new and unique Equipment proved to be incredibly taxing on the brain. As I heedlessly nagged

DARK SCAVENGER

www.darkscavenger.com





Jim for more and more design variables to keep the Equipment fresh, it got to the point where he was threatening to strangle me. And then he did!

What platforms did you focus on first and why?

We focused primarily on the PC platform for the development of Dark Scavenger. Developing for PC gave us a lot of freedom to focus on creating the game that we really wanted to make and gave us the strongest chance of allowing a bizarre title like our own to find an audience that could appreciate it.

What would you do differently if you could go back and do it again?

If I could do anything differently, I would have found a different way to fund the game. We crafted the title offering percentages for all the people involved with the promise that if the game sold well, everyone would profit from it post-release. The harsh lesson we learned is that people aren't reliable and this proves even truer when you're not paying them upfront.

Although this title was a work of passion for us, money is unfortunately a more effective motivator and this led to several people dropping out at various points of the project. Additionally, several of us (myself included) found ourselves gainfully employed once again in the industry partway through the project which essentially turned us into double-developers, stacking one full-time job on top of another.



There were a lot of sleepless nights and many intense Skype debates. The amount of effort involved to finish the project is unrivaled to anything I have done before but despite all the arguments and explosions, I don't think a single one of us has any regrets with how it turned out.

Should we ever attempt such a thing again though, we will investigate crowd-funding options or investors to ensure that we do not reproduce a similar experience.

Tell us a little bit about the story behind the game?

Dark Scavenger is actually based off an old pen-and-paper RPG that I created a long time ago which gave us a lot of established universe and lore to draw from. What appears in the final title is only a small snippet of that elaborate world.

Our lead BizDev dude (Kyle Perry) was actually one of the original players of that pen-and-paper RPG and was able to assist with play-testing and balancing outside of his business role.



As nerdy as it is, many of us still play that game today. However, in the middle of development, one of the original players (James Healy) tragically passed away from Leukaemia. We have dedicated the game to him.

What is the objective of your game?

Outside of delivering a strategic battle system, our goal was to immerse the player within a universe that they had never seen before and challenge ourselves to create a new and unusual experience for anyone who embraced it.

We don't expect every player to see all of the content in a single play through. There are secret items, encounters and an entire uber-boss fight that we suspect a very small percentage of players will be able to find. If players enjoy the game, we hope that they will discuss it with each other and uncover everything we hid within it.

Most importantly, though, we want the player to laugh. A lot!

Tell us about your main character in the game?

The main character is simply titled "The Dark Scavenger." We never describe their specific features; we only imply that the character is human and leave the rest to the player's imagination.

Unlike many modern games of its type, there is no "Good or Evil" system. The player can make the choices that best benefit themselves and their play style which we believe best lends itself to true role-playing.

Is your game single/multi-player, or both?

Our game is single player. It helps to have another person read all the dialogue over your shoulder in a funny voice though!

What features will your next game release have?

Although there is plenty of room in the Dark Scavenger universe to expand, any future titles from us will depend entirely on the success of this one. If we do end up making another game though, it likely will be an action title... that's all I can say about that for now.

Is there an age restriction?

Well, not officially. Most of it is comic mischief but there's some pretty dark stuff in there, especially towards the end. And some of the event and battle descriptions are pretty brutal.



Is your game free, a one off cost, monthly payments or micro transactions?

Our game is \$4.99. Our soundtrack will be \$2.99. There is no on-disc DLC. We're cool like that.

When did your game launch?

Our game released on April 30th, 2012.

What are the minimum software/hardware requirements to play your game?

Our system requirements are as followed:

OS: Windows XP/Vista/7, OS X 10.3
Processor: 2.0 GHz
RAM: 2GB (XP/OS X), 3GB (Vista/7)
Native Resolution: 800x600

Does your game require a download patch after the initial installation?

Nope!



Jumping Electron

POWERED BY FINMOUSE

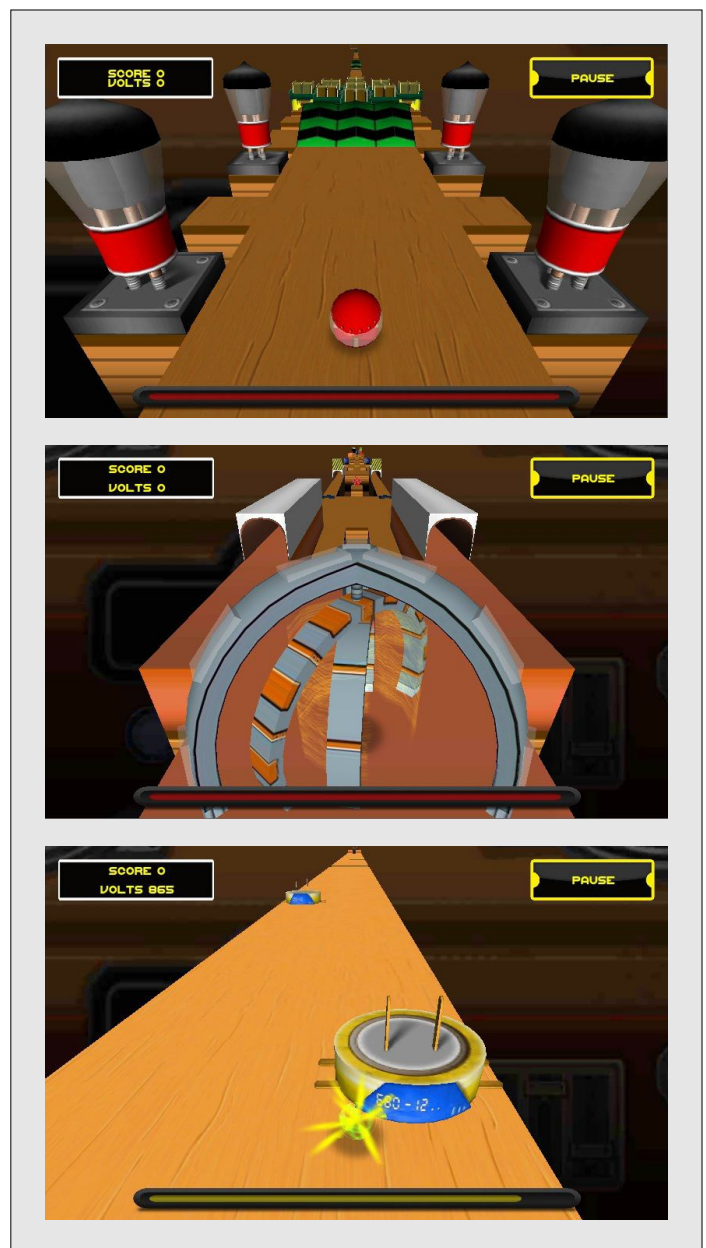


Romania is a swiftly growing force to be reckoned with in game design. Enikő László of Finmouse tells us a little about how they started out.

We started bonding with mobile apps back in 2010 as it was already a hot space. We did some client consulting and also built a productivity app for Android, Blackberry and Symbian. During the World Mobile Conference at Barcelona last year we were invited to present some game ideas for a specific platform with the possibility of a future partnership. We came up with a few ideas, none of them were selected, however the gaming seeds were planted and they grew.

After starting on the wrong foot, trying to build a complex 3D game with puppets and helicopters and realistic graphics. When we realized there would be no way to launch the Game in this century – we were in way over our heads ... However we used all that experience – especially with Unity 3D and Gaming in general and we thought about creating a casual addictive 3D Game – something between the “2D cute animation games” and “3D AAA Games with realistic graphics”... And since most of us have engineering backgrounds we ended up developing a game with electrons running and jumping over electronic components.

We then focused on Android as the customer feedback loop is much faster – if you have an issue you can fix it right away. We chose to work with Unity 3D that we think it is an obvious (and best) choice for mobile games as it is a truly multi-platform engine. Also they really support the indie community. We are releasing the game on iOS as well.



There were around 5 people working on Jumping Electron (concept artist, 3D modeller and animator, developer, game designer, marketing and PR person). We started with 3 people in 2012 and now we are around 10 people focusing on application and game development. We hope to grow and continue creating more games in the future.

Jumping Electron is a techie arcade game for Android and soon iOS with a classic platformer look and feel, and a gameplay fuelled with vintage adrenalin. First game of the series, Electronic Age is a journey back to the 20th century. Transformers, diodes, lamps and radiators become obstacles on a highway inside a Radio, a Jukebox, a Phone Switchboard and a TV. Gamers will have to skillfully guide the Jumping Electron on the highway inside electronics, avoid crashing with obstacles, tunnels and gaps, gather Electrons, Protons, Neutrons to maintain energy level. Gameplay is controlled with accelerometer and tap screen functionality. It's easy and casual.

Jumping Electron is a single player game, fully integrated with OpenFeint and Gamecenter leaderboards that allow score contests. Jumping Electron will have a total of 4 chapters: Radio, Jukebox, Phone Switchboard and TV with 80 levels. The series will probably continue with Transistor and Optical Ages. Jumping Electron has no age restriction. It could even be considered an educational game, motivating kids to become interested in science, physics, electronics.



www.jumpingelectron.com

We launched Jumping Electron for Android on June 13th and we are planning to release it to iOS in early July. For now we are focusing on building up a high quality, fun game, getting the word out to users. We might start thinking about ways to finance further development and so may have to look at different ways to raise funds. Jumping Electron is Free to Download from Google Play. Jumping Electron game story is focused on an Electron who saves the situation in each level, powering up the electronics.

Techie Stuff:

Jumping Electron will be available for both Android and iOS. We have already released two versions for Android, the normal one and the HD Version for Tegra 2 Phones. The Game will run on any 3D enabled Phone – for example I play it on an HTC Desire. It is around 20 Mb however most of it will be stored on your SD Card. It is a medium sized game that does not require a patch – you just download, install and play. 20Mb is not small but it is manageable even on any 3G network or WIFI nowadays. We went with the best graphics possible that can still run at a good frame rate even on lower end devices.

One of the technical issues we encountered was related to the music loop that needed to have a smooth transition when replaying levels. Mp3 format was not up to the task, OGG was fine but we had some issues in Unity so we imported the WAV file directly and we compressed it inside Unity and now it works very well.

Another one was related to grouping textures in fewer files to improve performance. There are lots of issues that come up when you start developing your first game, however we are very happy with the initial result. We got some really good feedback from gaming media, as well as users and our home-based community. We are working hard on the next chapter: Phone Switchboard and we are thinking about the next Jumping Electron Age.



3QC Games presents



Player name **Jacco van Weert**

Wave **Random**

Mode **Normal**

 Credits

 Website

Play

Greenfields

 Help

 Options

 Quit

v1.1

And now (as the old saying goes) for something a little bit different ...Fruit on Rails from Dutch studio 3QC has the player directing a series of fruit laden trains around an increasingly busy railway system, using logic gates (signals) to keep the trains and their cargo safe. Jacco van Weert tells us more about the game and the studio.

Tell us a little about your game studio.

How did you get started?

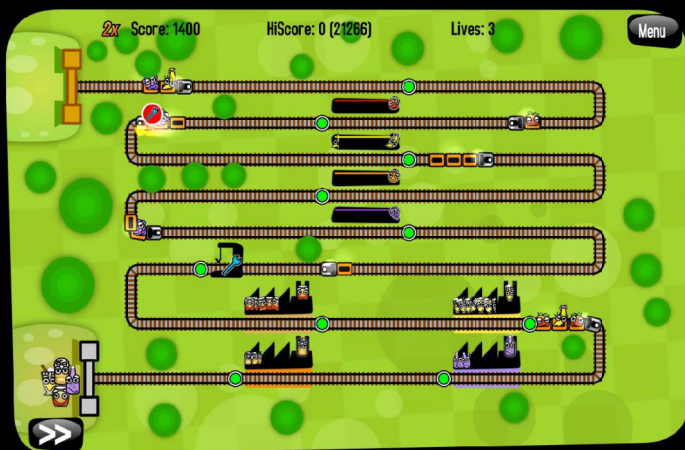
My programming urge started in the early 80s. My greatest achievement at that time was a 4K hand-coded Frogger clone in Z-80 Assembler. My serious attempts, however, started only 4 years ago.

How quickly did you expand?

I am still on my own. I am also doing the art on my own. The music however was composed by Pedro Alonso Pablos.

What challenges did you face when you first started?

In my daytime job I am a professional software engineer, so the engineering part of programming a game is pretty well known terrain. But what has been much more difficult has been determining what the definition of a "fun game" is. My first try was a turn-based space strategy game. It went pretty well until I realized that the game play was too complex. The next game



Options

2560x1440 32bits **Full screen**

Screen size

1:1 Pixel

Half Size

Sound

Effects On

Effects Off

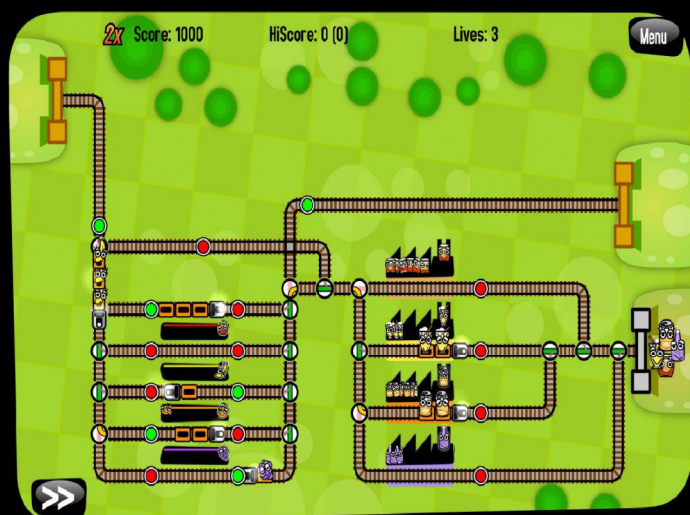
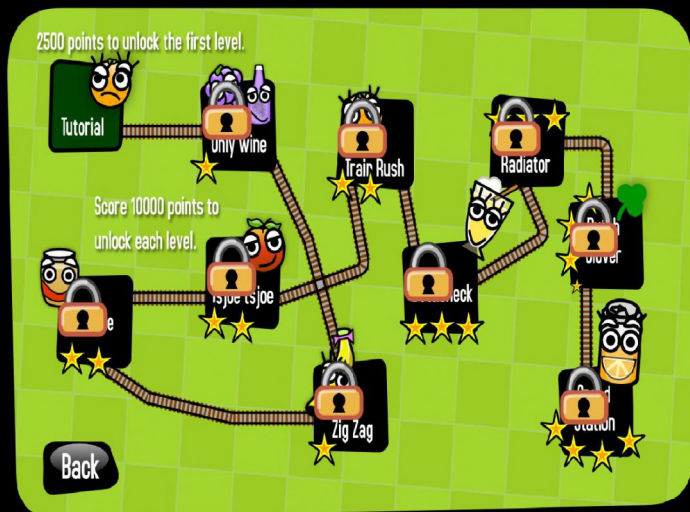
Music On

Music Off

Cancel

Save

Ok



was a casual "marble" physics game. This game was much better only the point was it was in need a lot of levels and I concluded that the levels weren't challenging enough. So my next game is an operational war game and this game is still under development, while the development was under-way I got the idea for a casual game, and this was Fruit on Rails.

What platforms did you focus on first and why?

All my games are programmed in Java, so by principle they are multiplatform, but my development platforms are Mac and PC. I created a framework in which I can choose which backend I will use, Java2D for maximum compatibility, Slick2D/LWJGL for performance (currently used in Fruit on Rails) and libgdx for the Android platform.

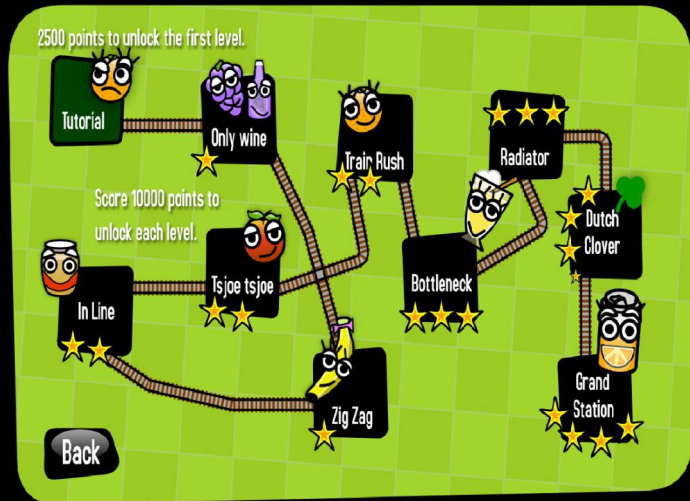
What would you do differently if you could go back and do it again?

At the start I used third party libraries directly which can be a problem when a library isn't supported any more. Currently it really pays off to have a home-grown middleware library which allows abstraction from the third party library.

Tell us a little bit about the story behind the game?

My goal was to create a mix of "Flight Control" and "Harbor Control", the time management games and "Airport Mania" for trains. This game should also appeal to my own children, who end up having to play in my desperate measures to get beta testers! The game will have an online leader-board. As it would be my first game release I hoped to learn every aspect of game design. I also did the graphics design by myself but couldn't really decide what sort perspective look I would give the game. In that respect I got some great advice from Chris Hildenbrand, a man I greatly admire. He put me on the right graphics track.





www.3qc.nl

What is the objective of your game?

There is no story behind the game, it's simply getting the highest score.

Is your game free, a one off cost, monthly payments or micro transactions?

The game is on sale for \$3.99

When did your game launch?

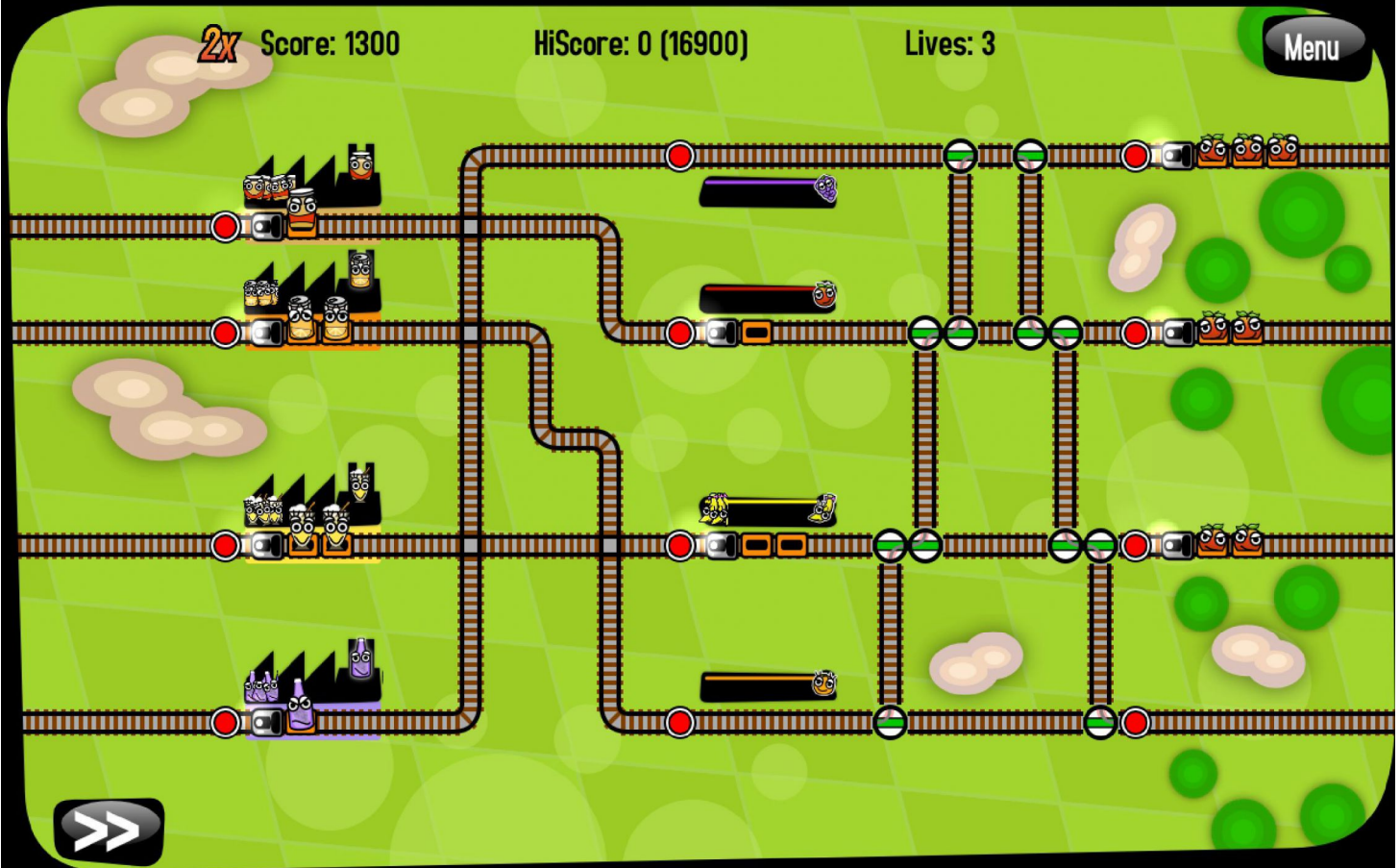
Mid April.

What platforms does your game support?

Because it's written in Java it is multi platform by design so Mac and Windows, the Linux is on hold. The Linux version has more issues, especially with the audio and graphics. Every Linux system is different so when Fruit on Rails is released for Linux the buyers should first try the demo version before buying.

Does your game require a download patch after the initial installation?

That's a nice feature of the game it will automatically update itself... except the Desura version, as they don't allow that.



Alkemi started as a two man team working on very small projects to test some ideas as Flash games mainly. Since we both came from a web agency, we also worked on contract jobs in the marketing field to pay the rent and finance our internal productions. Web sites, Facebook apps, Flash banners, all that kind of stuff...



TRANSCRIPTED

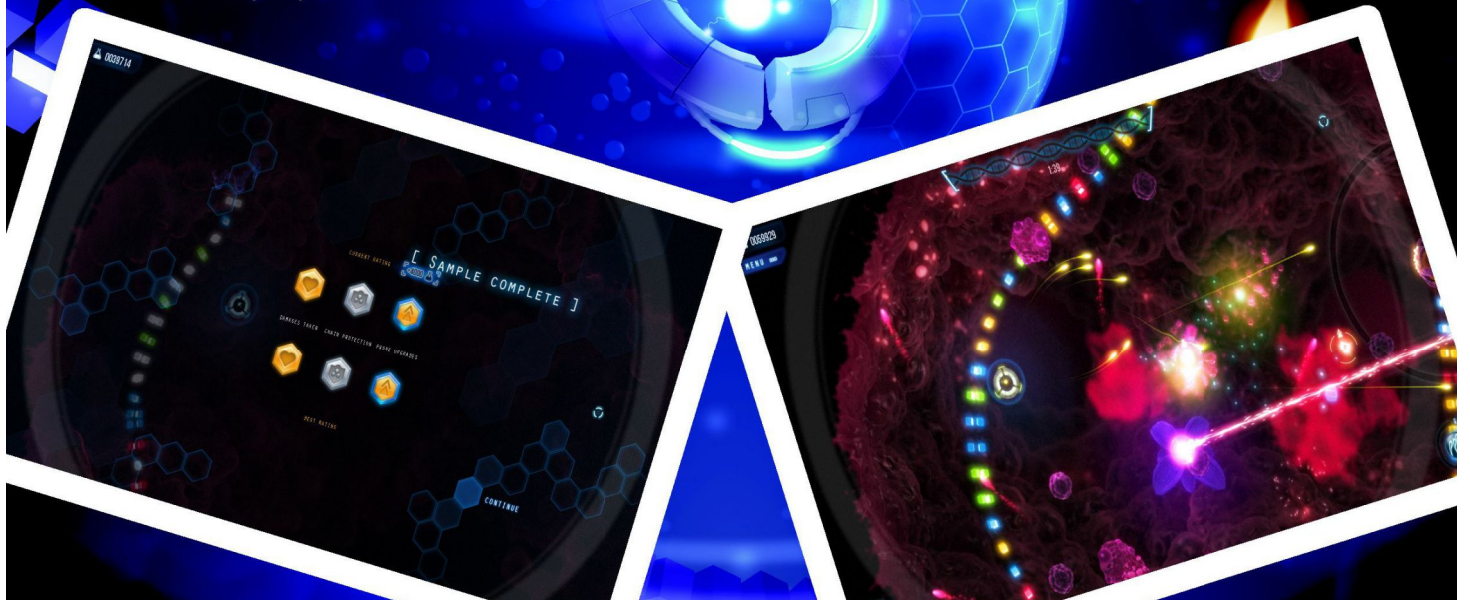
The original team is composed of Mickael Askren, a Jack-of-all-trades in development and me, Alain Puget, with a profile of technical artist. After 2 years, we hired a third friend and ex co-worker (Christophe Zerr) with skills in 2D, 3D and video production.

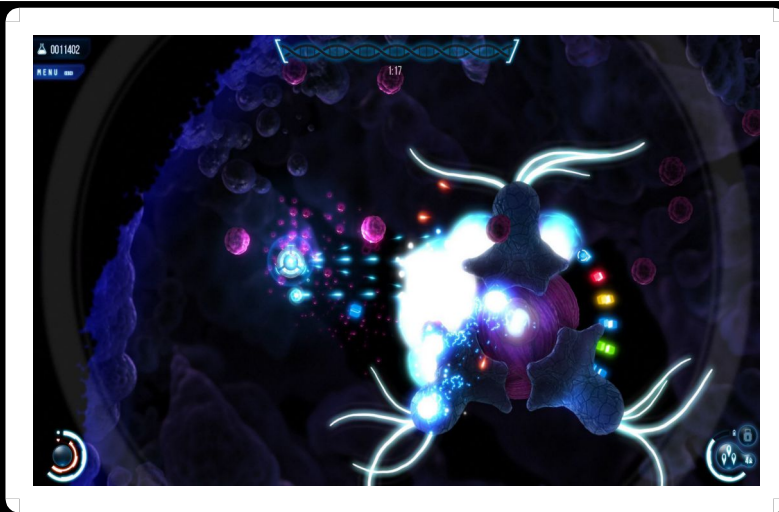
Obviously the biggest challenge when you start a company is finding money. We knew from the start we would have to split our time between working on our games and working for customers. We had quite a lot of luck to find a few agencies in Paris, with plenty of work for us and we've been working for the past 3 years mainly with the same cool people.

It took us more than a year and a half to store enough funds to actually start the production of our first game. During that time, we've tested some concepts improved our skills and developed tools mainly with Unity3D

We first focused on PC & Mac because that's the easiest platforms to work on with Unity3D. You can have a lot of fun with visual stuff and not have to bother too much with hardware limitations. We have future plans for mobiles and tablets but that will come later.

When there were only 2 of us, we mainly worked at home. Well, in Mickael's living



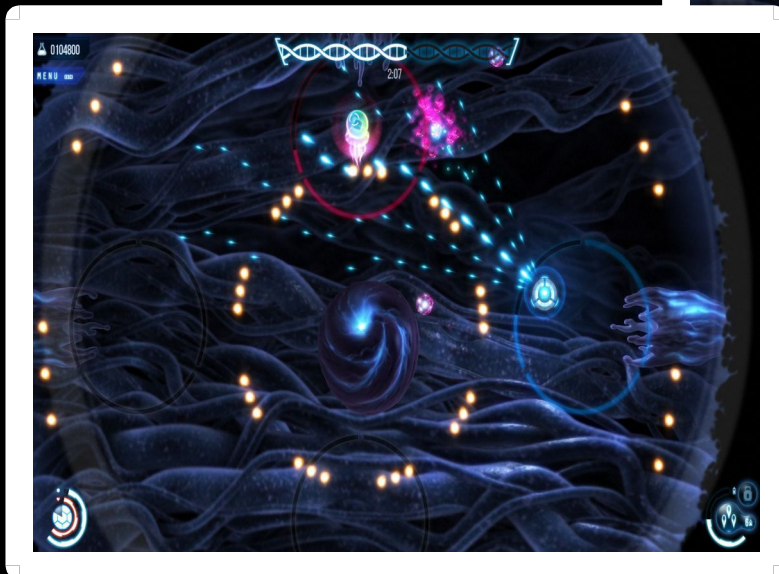
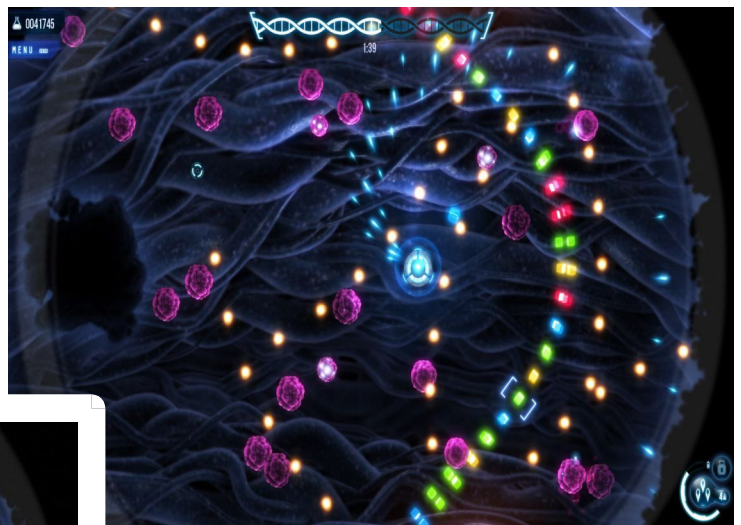


which turned out as a very tactical feature and a key element of this original mash-up. Since it was also a shooter, we had to add upgradable weapons and boss fights, etc. In a month, we had a playable flash prototype which we tested online on a number of game portals.

The biological theme of the game is a consequence of my scientific background. I have a PhD in organic chemistry and became a 2D / 3D artist only in my mid-twenties. The game is in no way scientifically accurate. It is pure science fiction but I enjoy adding references to my old field of study.

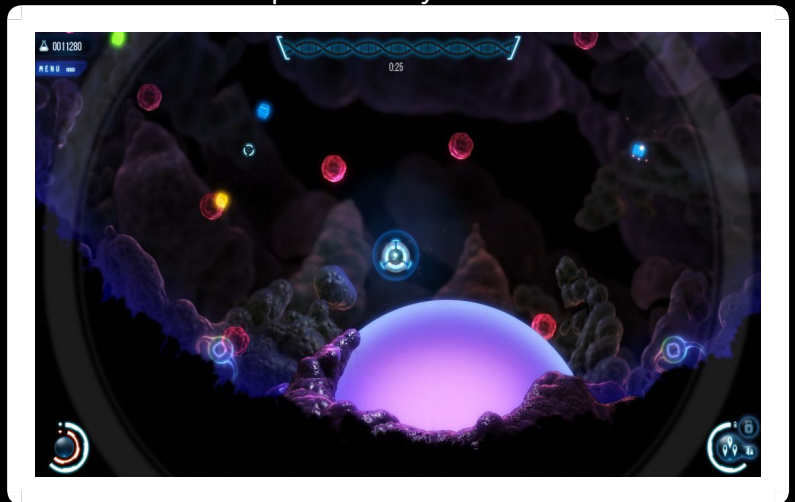
room in fact. With our first employee, we rented a small office with room for 3 or 4 people max.

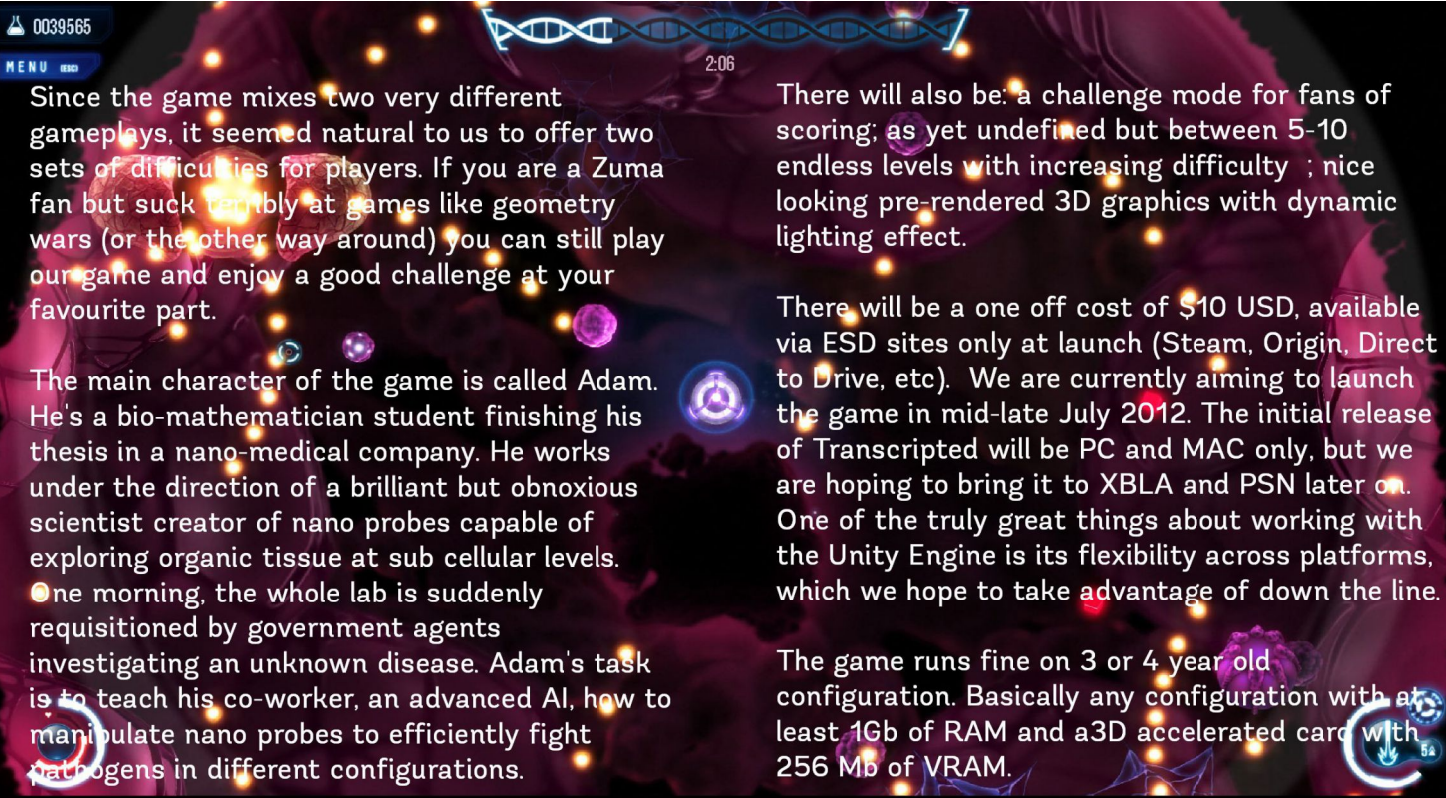
We are cautious, very cautious people. Alkemi is actually my second try at creating a game studio. Ten years ago, as I was finishing my studies, I tried this experience with friends. It didn't turn out that well but we were more hobbyists than professionals at the time. I guess you could say Alkemi is already my way of going backward and do things differently.



Transcribed is basically a mash-up of two genres : a match-3, reminiscent of games like Zuma and a twin stick shooter. The game is designed to blend as much as possible the two very different gameplays and make them interdependent. You need coloured cubes to break the pseudo DNA chain which is the actual goal of the game, but to get cubes you need to destroy hostiles. You want to launch cubes as fast as possible to rapidly break the chain but keeping them makes you invulnerable so you are forced to permanently make tactical choices.

Transcribed was born from a random thought while I was working on a contract job. Three years ago, someone hired me to create a generic clone of Zuma for a game portal. I'm a huge fan of this game which is by far my favourite casual game. Once the job was done, I started to mess around with the original concept and add unrelated features. What if the ball launcher could move? What if you had to avoid monsters while breaking the colored balls? What if the game was also a multi directional shooter? From there, a lot of the main features of the game came naturally. Since you needed time to aim in the match-3 part of the game, we added invulnerability





0039565
MENU

2:06

Since the game mixes two very different gameplays, it seemed natural to us to offer two sets of difficulties for players. If you are a Zuma fan but suck terribly at games like geometry wars (or the other way around) you can still play our game and enjoy a good challenge at your favourite part.

The main character of the game is called Adam. He's a bio-mathematician student finishing his thesis in a nano-medical company. He works under the direction of a brilliant but obnoxious scientist creator of nano probes capable of exploring organic tissue at sub cellular levels. One morning, the whole lab is suddenly requisitioned by government agents investigating an unknown disease. Adam's task is to teach his co-worker, an advanced AI, how to manipulate nano probes to efficiently fight pathogens in different configurations.

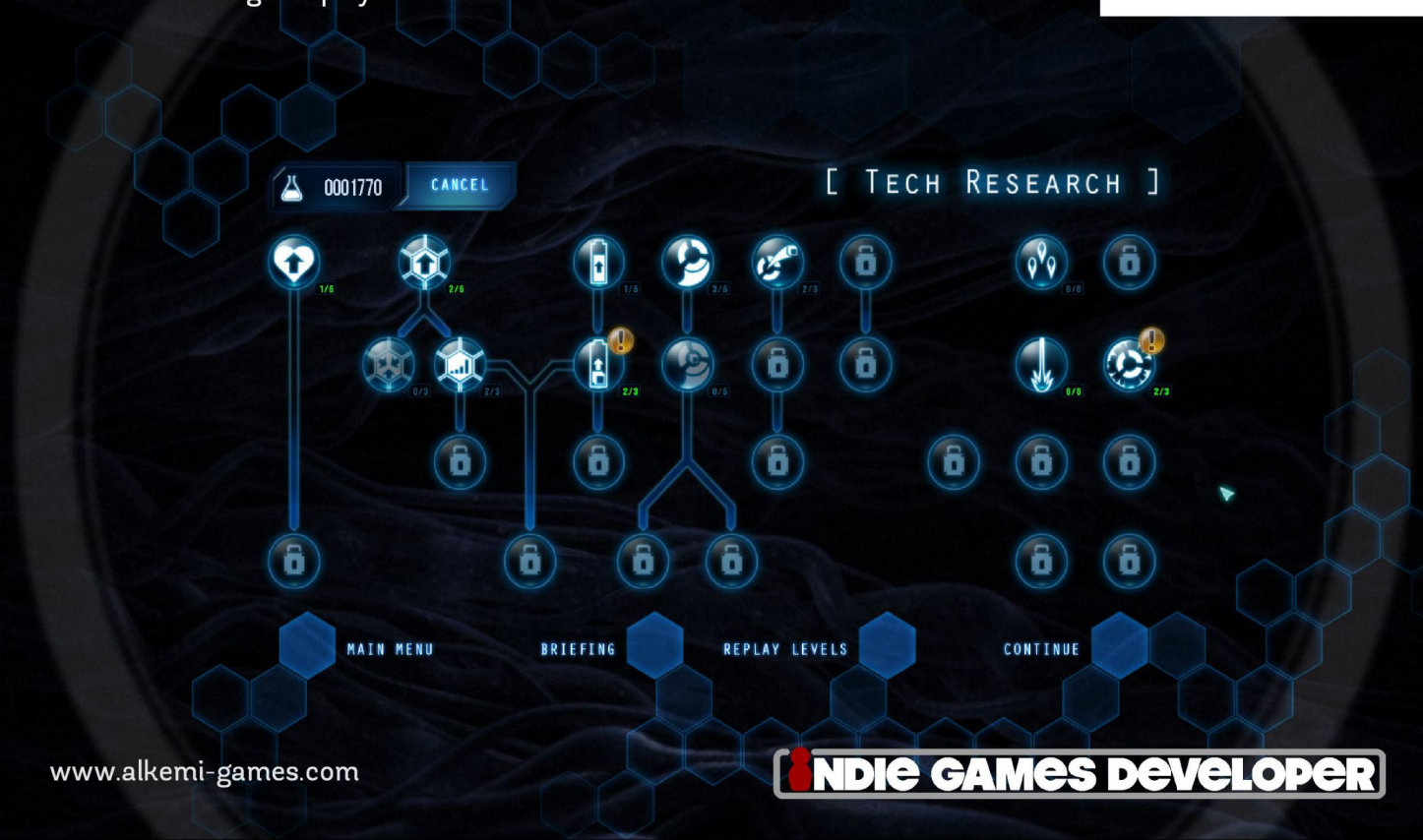
There will also be a challenge mode for fans of scoring; as yet undefined but between 5-10 endless levels with increasing difficulty ; nice looking pre-rendered 3D graphics with dynamic lighting effect.

There will be a one off cost of \$10 USD, available via ESD sites only at launch (Steam, Origin, Direct to Drive, etc). We are currently aiming to launch the game in mid-late July 2012. The initial release of Transcribed will be PC and MAC only, but we are hoping to bring it to XBLA and PSN later on. One of the truly great things about working with the Unity Engine is its flexibility across platforms, which we hope to take advantage of down the line.

The game runs fine on 3 or 4 year old configuration. Basically any configuration with at least 1Gb of RAM and a3D accelerated card with 256 Mb of VRAM.

The game is single player only and has an original mix of gameplay : match-3 meets twin-stick shooter playable on both Mouse & Keyboard or XBOX pad control setups. The gameplay will be spread across a couple of alternate modes including: a scenario mode with 25 levels among which there are 5 boss fights; a skill tree with more than 20 skills to unlock and upgrades along the game; 5 different and upgradable weapons; two sets of difficulty (puzzle / shooter) to adapt to the widest range of players.

We hope to offer additional DLC and add-ons as the game matures, which will come in the form of patches.



0001770 CANCEL

[TECH RESEARCH]



MAIN MENU

BRIEFING

REPLAY LEVELS

CONTINUE

www.alkemi-games.com

INDIE GAMES DEVELOPER



INTRODUCTION

Hello once again everyone! Richard here, back once more for the second part of my MMX-1985 tutorial, and we're going to finish this guy up right by getting into the last 3 major steps of the characters construction, and taking him to the finish line with a powerful render. So, after a quick re-introduction to where we are at so far, we'll get right into the thick of it!

If you have not checked out the first part of this tutorial in the last issue of the magazine, please do yourself a favour and go take a look. It's always free to read the magazine, so just do a few clicks on the website and you are there. It will also help to bring you up to speed with where we are now.

To put it in a nutshell, in the last issue, we left off with sculpting our mech. Now we're going to start from where we left off, jump into texturing, retopology, then rendering and presentation. So without stalling further, let's get into it right now!



A freelance 3D artist living in Orlando, Florida in the United States. Richard briefly attended Full Sail University in 2003 and has been self taught in his craft for several years, teaching himself many of the industry standard programs and continuing to grow his skill set through rigorous practice, development, and participation. He has contributed to many freelance and volunteer projects and continues to actively look for work in the games industry. He is available for freelance and in house positions anywhere in the World.

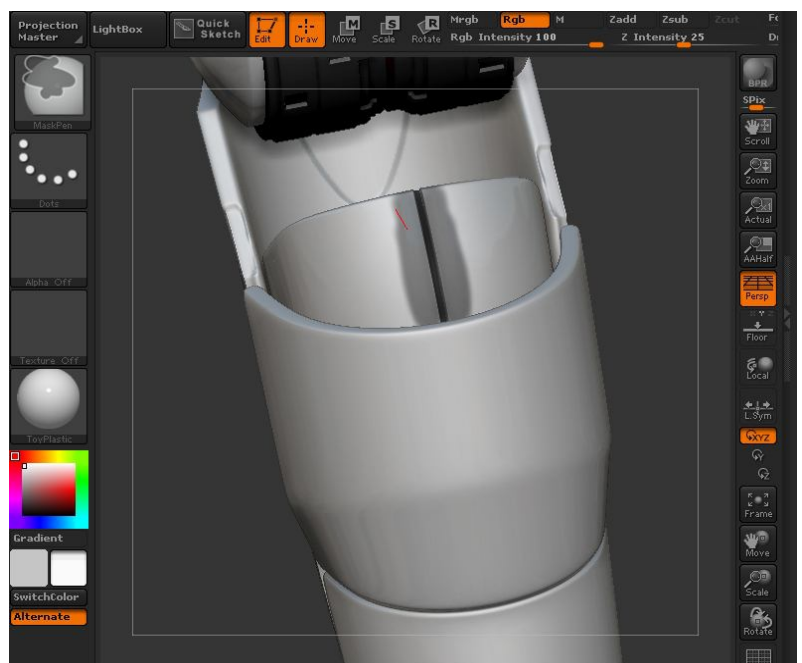


TEXTURING

With texturing, we have run into a rare but fortunate situation with this character. Typically, I will save texturing for until after retopology, doing that step in a program like Photoshop or Mudbox. I seldom use Zbrush for texturing anymore, however, for this character, using Zbrush for texturing will not give us many drawbacks, due to heavy use of flat colors and clean definitive detail on our sculpt. And since we already have our subtools set up and listed, ready to paint, we can cut out the process of exporting high res objects for painting elsewhere, as this can become pretty heavily intensive on memory. So we will avoid that by painting directly in Zbrush. I still advise you research the texturing paradigm between Zbrush and other programs like Mudbox though, as it can prove quite a difference in most models textures in the end.

With that out of the way, let me talk a little bit more about reference. Referencing during conceptualization can provide you with a great many ideas. Reference any remotely relatable piece of art that can give you a better understanding of where you want to take your work. In this case, obviously I referenced a lot of mech concepts. Though, while looking for important things like shapes and form, I was also building up a colour palette in my mind. I had found one mech from a popular brand that really drove home the colour scheme I wanted. I thought the best approach would be to keep it simple, as is its design, and thus chose a nice monotone 3 colour layout to work with.

One good thing to keep in mind is to keep your palette limited. This may sound counterproductive, but by "limited", Choosing 3-4 primary colours to represent is a good practice.



Masking around your sculpted details can assure that the colours stay sharp and in place.

I simply mean to keep it manageable. Choosing 3-4 primary colours to represent is a good practice. This doesn't mean you cannot use other colours, but rather that the primary colours you are using be the most prominent, and what defines the overall texture design. I've seen far too many great models and sculpts not live up to their potential by being textured poorly. This is usually the result of a few common mistakes, with those being the following...



TEXTURING ...

Simplicity

I realise I've mentioned the word "simple" so many times in this tutorial. Well, it's for a reason. I can't stress enough how simplicity can gain you a great result in both the final look of the piece and the ease of production. If you avoid over-detailing, you will award yourself the luxury of not needing to go back and fix these things upon realizing they are not what you want. It's much easier to add than it is to subtract. This goes the same for texturing. Do not mix too many prime colours together! Instead focus on shades of the primary colours you are using. This allows you to break things up without destroying the flow of the piece.

Over-Detailing

To go more in depth with what I just mentioned about over-detailing ... pay attention to the big shapes in your sculpt or model. We don't have much of a problem with this guy, but it's valuable to find out what amount of colour is needed to make those shapes as prominent as possible. Do not spend an hour detailing a small part of the model that will be seldom seen. This sounds like common sense, but I've seen it done a great deal before. Remember, your model will not always be seen at close range, and in almost every instance, detail that looks great at the closest view, will look like completely noisy crap at a normal full body view. There is a fine line you must walk to balance out the detail, and make sure it looks great from all ranges of view. Focus on the large details before the small ones, they take precedence!



Small and simple placements of standout colours can break up the texture nicely and still remain prominent.

Contrast

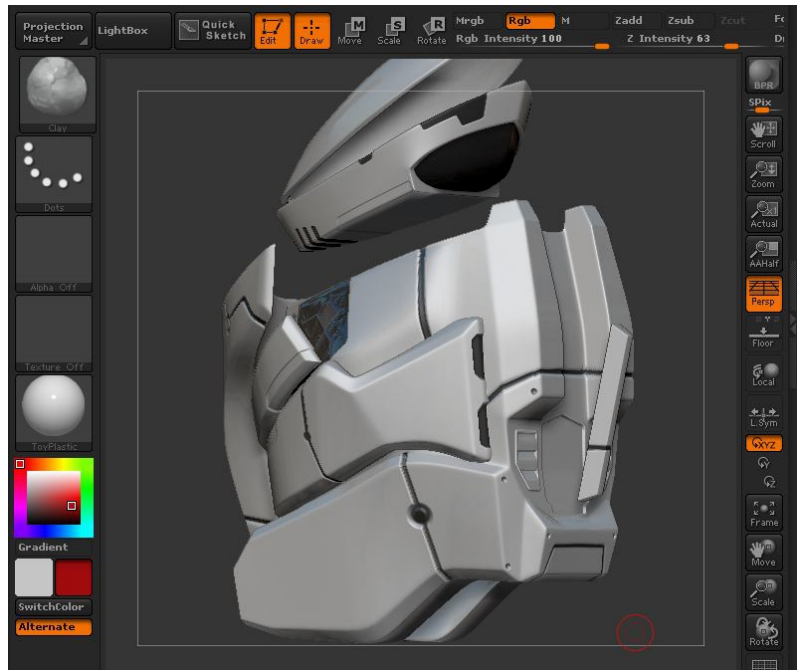
This is a big one that is very often overlooked. The short answer to this is to study colour theory. There is a lot of information that encompasses colour theory however, so if anything, try and understand a few rules of thumb. Firstly, know your colours of course! Primary, Secondary, Tertiary. Acquire a colour wheel. This helps greatly when choosing a palette also. Secondly, darks behind brights work best! Ok, so "Color Context" is the actual term, but it's pretty self explanatory. Brights will always appear dominant when they are above darker colors. Make sure that if you are blending, that the brighter color has a level of "fade" into it's counterpart. This will help you avoid mixing a bright with another semi-bright color, as that approach can get really ugly sometimes. Lastly, practice equality! A well balanced palette is one of your most important tools. Have an equal number of darks to lights, and make sure they work well with each other!



TEXTURING ...

You can see that in the mech, I used a simple black and white palette, with some red to break it up. My darks are kept in the background of the underlying parts, and in tight contrast to the brights in the foreground of the armor plates. Of course, there are a gajillion combinations of colours out there, so just use what is best for your work while following these principles, and you will have some great results.

Next, we move on to everybody's favourite part (sarcasm), retopology!



RETOPOLOGY

So now we're moving on to retopology, topology, whatever you want to call it. But first, let me explain for artists new to the game, what exactly topology and re-topology are.

Keep bright colours in the foreground and darks in the back. This helps the viewer to keep focus on the important parts and discern them from the rest.

In a game environment limited by our modern technology, optimization is king. This used to not be so much of a problem, as in the early days of 3D gaming, the optimization came with the art process itself. Most of these early game models were made up of no more than 100-200 triangles, or even less in a lot of instances. This basically meant that optimization wasn't needed, because there was next to nothing to optimize. Nowadays, we have game engines that can support hundreds of thousands of polygons, we're sculpting our models into the millions of polygons, and it's just getting bigger and bigger with each, new iteration. That is why some time ago; we introduced **retopology** into our development pipeline.

With retopology, we are given the ability to put polygons in the places that matter, and keep them out of the places that don't matter as much, thus giving us a nice clean mesh prime for use in a real time game engine. In the end, our newly retopologized mesh will benefit important things like the games frame-rate, it's GPU handling, and much more.

You have many software choices when it comes to this step. There is no "right" choice, and they all have strengths and weaknesses. I personally use Topogun, as it has given me the best results consistently.

When beginning retopology, keep in mind your end polygon count. You may remember in the first part of the tutorial, I stated that we will be shooting for 18,000 triangles. In reality, even for current ..



RETOPOLOGY ...

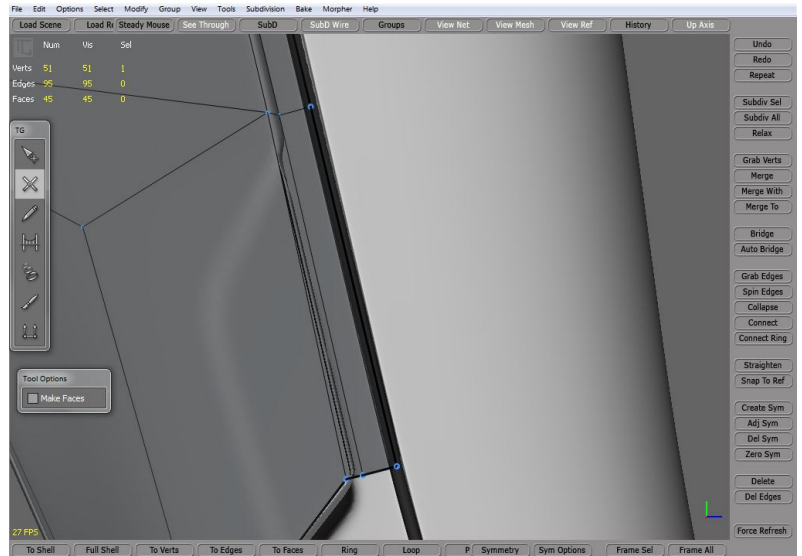
generation hardware, this is steep. So what I forgot to mention is that this is only 1 level of detail, or 1 L.O.D. In the industry, we use many L.O.D.'s to make sure our work is of quality (both visually and technically) at every angle and distance. This same principle can be applied to how many of these guys we have on screen at once. At 100, if they were all 18,000 triangles, our game would crash hard. That is something we want to avoid, and at the same time, we want it to look good when one is up real close to the camera, or when 100 are in the background, all while sharing the grand real estate that is a game engines maximum triangle count.

Again, as with texturing, try to focus first on the parts of the model that will be seen the most. The chest plate and helmet, the thigh plates, the shoulders and forearms, these are the biggest parts of the mech, and so in turn, they get the most of the polygon real estate. Once more, as I stated in the modeling section last month, only put polygons where they are needed. This may have been able to be bi-passed before, but now that we are working on the in game model, it is more important that ever. Do not give a straight piece multiple divisions. It's wasteful. Try and only use polygons where they are supported. Since this guy is also organic in some nature, we need a lot of polygons to support the curviness of it. Don't be afraid to compensate for what is needed, you can always optimize more later when we do the other L.O.D's.

Some artists like to work in triangles, some in quads. In short, a model will always be converted back to triangles when in an engine. This is how a game engine calculates our models, in triangles. However, building up your model in quads is going to benefit you by giving you better control over manipulating it, a cleaner UV map, and the Animator for your studio will thank you endlessly, as quad meshes are much easier to deform. If you are building a static mesh(one that will not deform at all), you have the freedom to use what you like, but mostly quads is a good practice for anyone in my opinion.

Retopology has very simple principles that are easy to follow. It is a tedious process to some, but is also a very integral part of the game pipeline in modern times, and any developer will put great value on anyone who can do it properly.

Protip: You can easily hide the seams of your model in the dark parts of the textures. If light hits your model the wrong way or the model is too far from the camera, seams have a tendency to show. Hiding them behind darker colours and on corners and inside cuts on your model make them less noticeable than on bright colours or other surfaces.



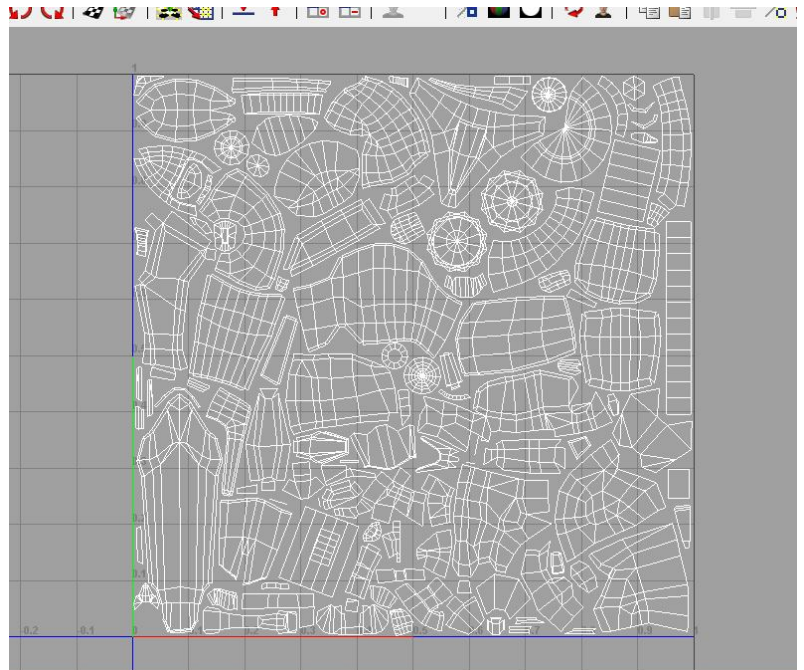
Placing seam borders on cuts and dark colours in your mesh can give the illusion of them being hidden.



UV LAYOUT

UV Mapping is also another process that many consider tedious, though while I am "UV'ing", I often think of it as trying to put together a really complex puzzle. The pieces need to fit, and in the latter example you cannot just hammer them into place. Fortunately, in the digital realm, we are awarded the luxury of resizing and manipulating our UV maps in order to make them fit together well.

Once again for the newcomers, a short introduction to UV Mapping. Imagine roadkill. Once you're over the disgust, try to imagine how it is layed out. It's flat and spread about. Now imagine (please don't do this in reality!!!!) cutting a line from the top of your forehead all the way straight back to the base of your neck, and then digging both fingers in that cut and peeling your skin off forward. Now take your skin and lay it out flat. That is essentially UV Mapping. The skin is the "texture" that "wraps" around the object. Another way is to think of it like a map of colour. Each point of colour, or a "pixel" on this flat surface corresponds to a point of location on the mesh that you wrap it around. It's really a fairly simple principle to understand, but it can easily be manipulated wrongly, thus causing errors in the display of your texture. I will explain a bit more about some common mistakes.



Keeping your UV islands evenly spaced apart while tightly packed together will give you a lot of space for your texture while remaining optimized.

One of the most common mistakes made is in relation to the texture space available, as well as the "pixel aspect ratio" of the texture being applied. In a nutshell, the "pixel aspect ratio" corresponds to how much space is given to any part of the texture. It would make sense to give a part of the model that stands out a sizeable amount of space on the UV map, as opposed to say, a gear that is lying underneath the armor plates on this guy. Though, this mistake is still commonly made, and usually causes the parts of the model with lesser texture space given to appear very pixelated and grainy or blurry. Not what is wanted in most cases. Now, even though we are handed the benefit of flat colours for the mech, which will not cause as much of a problem with pixel aspect ratio, we are not exempt from the practice. The details will definitely suffer if we are not careful with this.

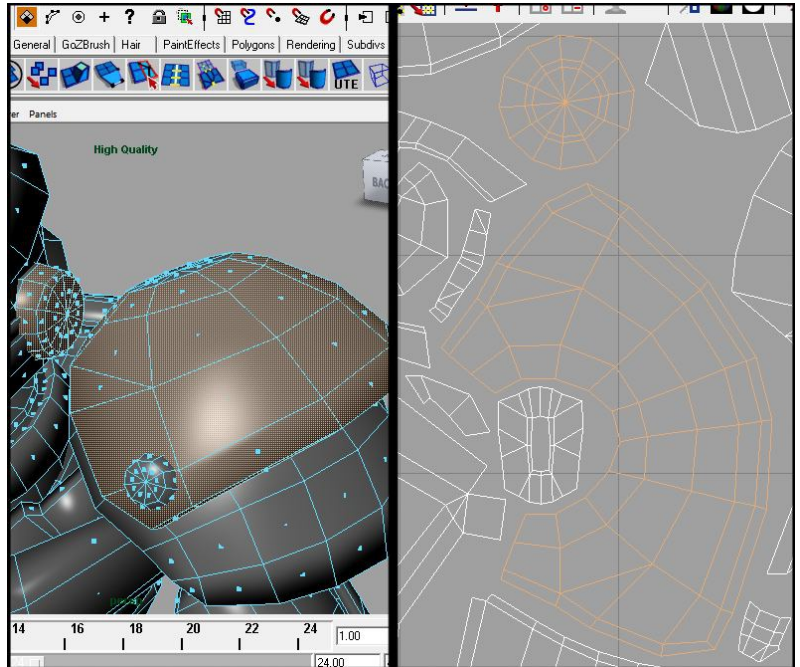


UV LAYOUT ...

Also remember to keep a close eye on your UV "islands". The way you position them in relation with each other is of importance, so not to cause unwanted "overlapping UV's"(although this is wanted in some cases, it must be carefully articulated). Also, pack your UV's tightly to use up as much texture space as possible, but keep them far enough apart to allow for "padding", which I will explain more when we talk about baking the textures down.

Another thing, as with my retopology protip, is to keep seams as hidden as possible. Use parts of your model to cleverly hide these seams, so that they are not easily visible.

You can use many programs to UV, including using the UV functions of your proprietary software (Maya, Max, Blender, Etc.) - I personally use Roadkill UV. It is a very old program, but very simple, and it still gets the job done for me. Again, no right answer on programs used.



Giving the proper UV space to each object will help to avoid too much detail on a seldom seen part of the mesh, and give the most detail to the more prominent regions.

TEXTURE BAKING

This process can usually be a short setup time, but a long waiting time. So I will go over a few important steps for setup, as most of this process after, will be you getting up to get a sandwich or sitting and patiently waiting.

Again, many solutions to choose from, including proprietary - I personally use my same retopology program, Topogun, as it has some nice baking features and doesn't require me to jump to another program, allowing me to load up my work, set a few things, and click "generate".

The short story of it all is that you now have all your ingredients. You have your high polygon and textured mesh, and you have your low polygon and UV'd mesh. Now we need to take all of the high mesh's details and "bake" them down to our low mesh's optimized form. Without getting technical, this is what gives us the illusion that our high end looking mesh is actually a low end one.

We have several maps we can choose to bake down, but I will focus on the three most common ones.



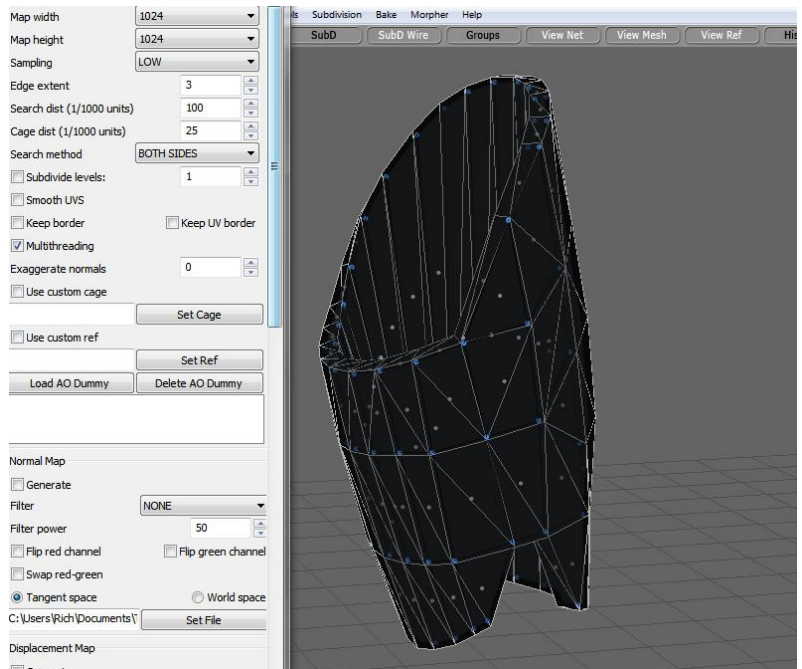
TEXTURE BAKING ...

Diffuse

Pretty straight forward. This is what displays the colour on our model. It is also the quickest and easiest one to bake down, due to its simplicity. Without getting technical, there is not much more to it than that.

Normal

The normal map has become much more common since the inception of the last generation of consoles, and now that they are easily handled by current gen hardware. Before, the illusion of depth needed to be cleverly faked, through well placed shadowing on the diffuse map. Now, we have the ability to generate a normal map, which affects how the light bounces off the model. This gives the illusion of depth in details where there is no actual geometry present to support it. Almost essential in new age development pipelines, this is a must know skill in today's industry.



Proper cage setup is integral to getting a proper bake. Make sure your cage fits firmly in relation to your model.

Ambient Occlusion

Although not new, ambient occlusion or "AO" is used more often in the current gen than it was before. At its baseline, it is really just "faked shadows". These maps help emphasize your normal and diffuse map, especially down to the details of them. Typically, only AAA games make heavy use of them, as highly detailed models will benefit the most. These also have lengthy baking times, due to the number of calculations it takes to project light on the model and translate that to graphical data in the form of a texture map.

Outside of understanding your maps, and again, there are several - research and experiment with what settings work for you. There are a great number of settings that will cause different results. Research your chosen programs baking capabilities and understand how they work.

Protip: Keep an eye on your "padding". Padding is basically the lining that lies on the outside of your UV islands in the texture. This will start to show if the model is viewed from a distance or certain angles. On larger maps, like 2048x2048, keep your padding relatively high.



RENDERING AND PRESENTATION

Here's the fun part. We get to present this machine of war in all its glory. But first, let's finish what we were doing, taking our baked out maps and assembling them in Photoshop.

In Photoshop, blending modes are our friend. When it comes to piecing together your diffuse, AO, and Specular maps, if you make sure they have a black background, then the "Screen" blending mode will be the one you want. This allows us to avoid needing to cut out our UV islands out, which can become a time waster we don't need.

Assembling normal maps are a little more complicated. Because of their bluish background color, using straight blend modes will not yield the best results. The Overlay option is the closest you will get, although alone, the normal map will not be well emphasized. A good fix for this can be done in just a few steps -

In Photoshop, make sure your foreground and background colors are set to white (RGB- 255,255,255) and a neutral grey (RGB- 128,128,128). Making sure that grey is set as the foreground colour, set the top most layer to an "overlay" blending mode. Before anything else, **duplicate** the layer and set the duplicated layer to a "multiply" blending mode. In the channels menu, select the **blue** channel of the layer set to overlay. With only the **blue** channel selected, **fill** this channel with the neutral grey. With the multiplied layer above it, this will look strange and off. That's ok, because now, we can select the multiplied layer, jump into its channel menu, and we will now be selecting the **green** channel. After it is selected, we will fill this with our pure white colour. Follow these steps carefully, as it's easy to make a mistake during this, which is why I highlighted the keywords.



Using clever light placement and camera angles can give you some pretty cool effects in the final render.

This should fix your normal maps. You will need to do this for each new map you bring in to assemble with the existing ones, so a good practice is to group the two layers together and keep them organized. If you want to avoid needing to do this whole process for each new layer, you can use Photoshop's "Actions" ability to record your clicks, so that you only have to click once for each layer.



RENDERING AND PRESENTATION ...

Once you are finished constructing your maps, it's time to export them out and apply them to our baby! Once applied, we can take our mech into a nice rendering program for final presentation. Rendering solutions are plentiful. You have the functions of your proprietary software in Maya, Max, or any others, to well known in game engines like Unreal Engine via the Unreal Development Kit. It will help to research more into shader setup and such, as engines like Unreal have a very in depth shader/material function, and can be quite daunting if you don't know how to work it. For us, this is a game resolution model, so of course we want to render him inside a game engine!

EPILOGUE

When your work is rendered out, that's it! You are finished and ready to show it off. Firstly, I want to thank you for checking out my first tutorial, and I hope you enjoyed it as much as I did.

I would also like to say thank you to the director of this magazine, for allowing me an opportunity to share my work, as well as my methods, and to have an outlet for my creativity. Hopefully this is not the last you've seen of me, and perhaps I will be back in future issues to show you some more awesome things to create.

In the meantime, please check out my team's first game project, entitled "30 Minutes... Or Less" over at - www.adaptivelite.com - as well as in this issue of IGDM. The game will be finished very soon, and we would be grateful for your support!

Also, please check out my portfolio at www.richcg.com - I am always looking for more work!

With that, I bid you all a farewell. Thanks for stopping in and see you all soon!

Portfolio: www.richcg.com

IGD Profile: www.indiegamedeveloper.com/profile/richardcook



**LOOK OUT FOR THE VIDEO TUTORIALS ON OUR WEBSITE
WWW.INDIEGAMESDEVELOPER.COM/TUTORIALS**

Seon Rozenblum of Australian game studio 3 Sprockets, explains why down sizing his business operations has worked out so well for him.

www.3sprockets.com

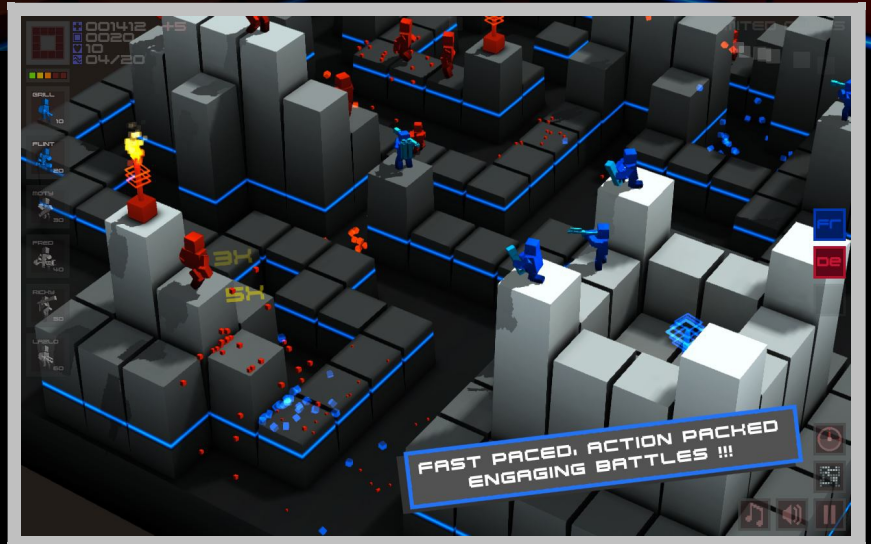
CUBEMEN

How did you get started?

I owned and ran an Animation & VFX studio called Sector3. We had dabbled in making games for clients and tried making some of our own ideas using Macromedia Director, they weren't very good due to software 3d rendering and poor hardware capabilities of the day. It wasn't until late 2006 when I stumbled upon the Unity game engine and instantly fell in love with it and realised that my dream of making my own games could finally be achieved.

Within the next 16 months I had shut down the Animation & Effects side of the business, let go of all of my staff, closed my office, fired all of my clients and moved into my home garage to become a one man Indie.

In 2009 I sold Sector3 to a console game developer, Trickstar Games. After a 12 month handover and some contract work here and there, I decided I really wanted to be a One Man Indie Games developer, so I started up 3 Sprockets and haven't looked back!



How quickly did you expand?

I am not planning any real expansion at the moment. I love the freedom of working for my self, by myself, though I do plan on outsourcing more of the art creation jobs on future projects to allow me to concentrate more on design and code.

What challenges did you face when you first started?

I learnt a lot running Sector3: about business, marketing, client management and staff management. When I first started making my own titles, I knew what I wanted to do, but had very little knowledge about the industry as a whole. I didn't know what was considered achievable or outrageous, so I just did what I wanted to do and somehow succeeded.

Once I turned Sector3 into an Indie Studio, I learnt a lot about the marketing and support requirements of creating and shipping your own titles, so going into 3 Sprockets, I had a lot of experience under my belt. Some game failures and quite a few successes.

What platforms did you focus on first and why?

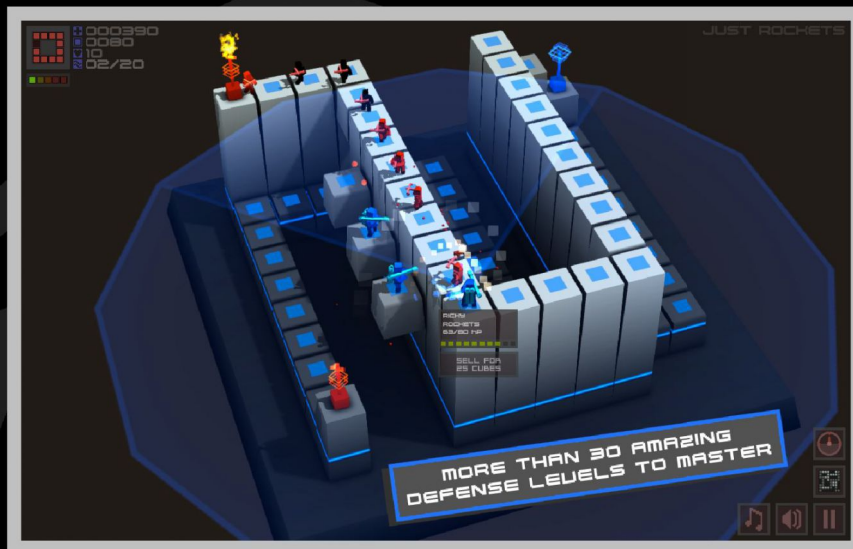
I strongly believe that mobile and desktop platforms are both as strong as each other still, from consumer demand and financial reward perspectives, and to be honest, though I am an iDevice fanboy, I still prefer to play my games on my desktop. So with that in mind, all of my current titles are designed to run on PC, Mac and Mobile (iPad and iPhone if appropriate).

How many staff and offices did you have when you first started?

With 3 Sprockets, it's just me, in my awesome office. We had a 3 car length garage at home, so I spared no expense and renovated half of the garage, turning it into my dream office. I have couches, desks, AV setups, Consoles, Alarm, Server room... the works.

How many staff and offices do you have now?

Still just me and my office :-)



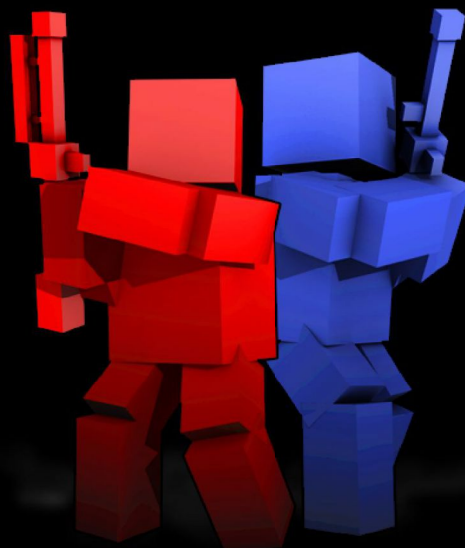
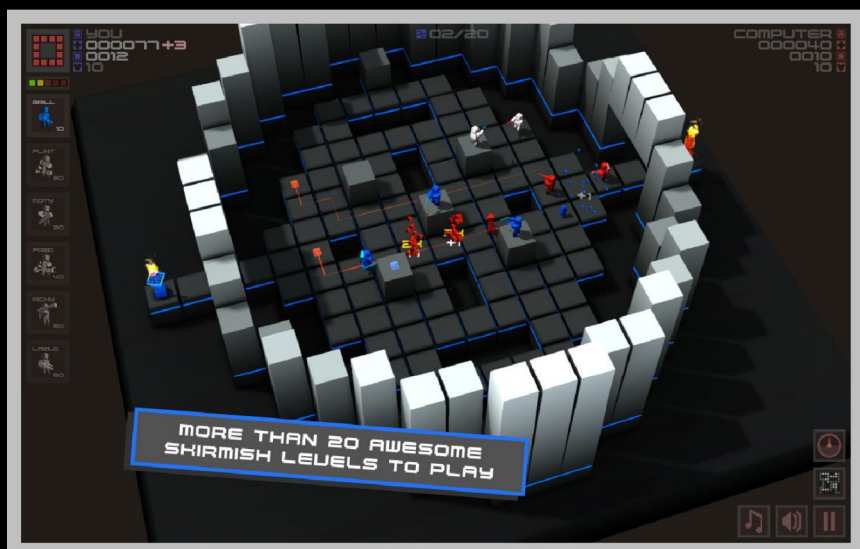
What would you do differently if you could go back and do it again?

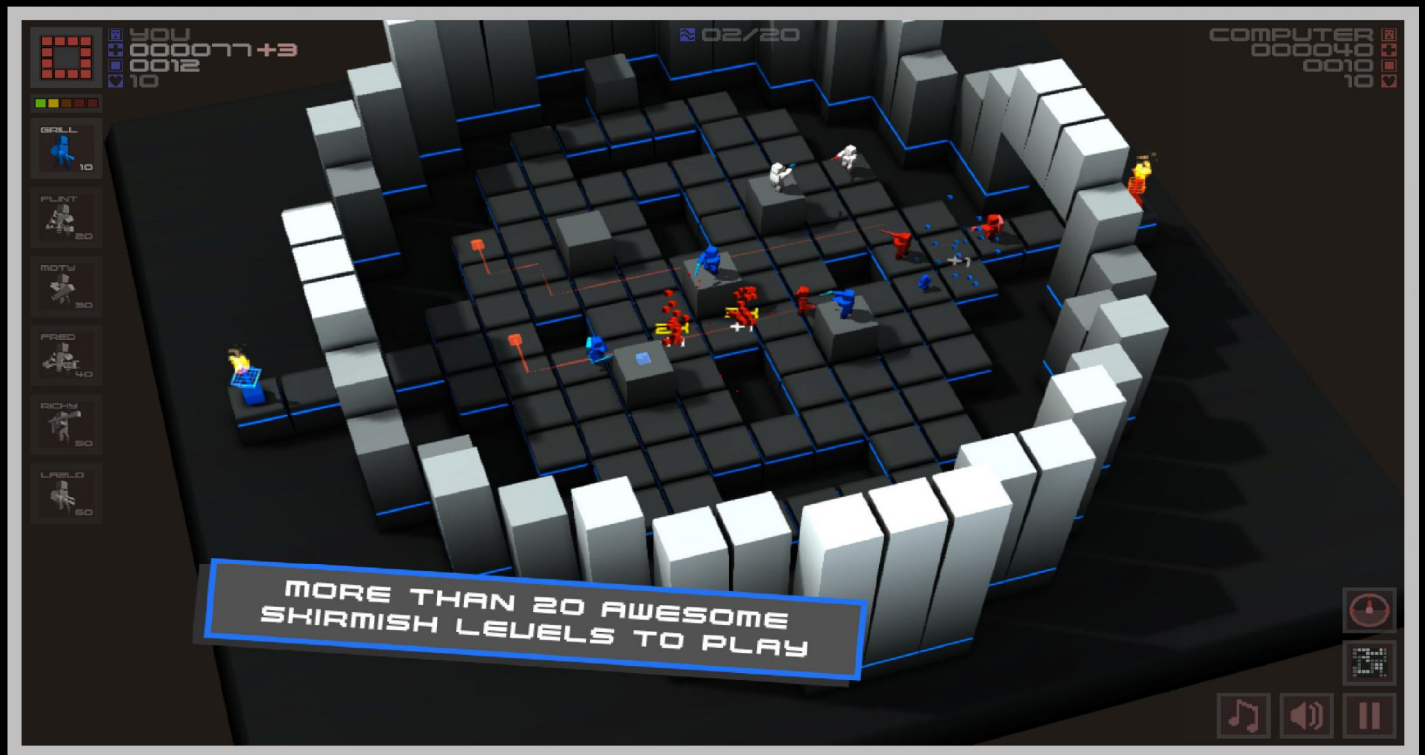
Well, as I stated, this is my second time around, and I really feel like I am doing it properly this time round. Some of the key things I am doing differently this time are:

- Not worrying about keeping things secret. I talked about and showcased early development images and ideas for Cubemen well before I even had a planned release timeframe.

- Going DRM (Digital Rights Management) free! I used to spend more time worrying about copy protection than good game design. Now, none of my games have DRM and after the initial shock of releasing DRM free, it's now the only way I would go.

- Plenty of pre-launch marketing to both industry and consumers. Honest answers to hard questions and the decision to put consumers first.





Tell us a little bit about the story behind the game?

Cubemen was born during a discussion between myself and Andrei Nadin, a friend also in the games industry, during the Melbourne Freeplay 2011 Conference, when we both should have been listening to the sessions, but instead were brainstorming game ideas.

The next thing we knew, we were both sitting in the State Library, working on a proto-type ... me coding a level editor and Andrei working on a Cubeman model. That's how it started, though what Cubemen is today is a far cry from our initial thoughts.

What is the objective of your game?

Cubemen is primarily a new take on the TD genre. But the key thing that we strived to do was to make it simple to play and accessible for casual and non TD enthusiasts. Our aim was to make it about strategy and simplicity, removing things like towers and upgrades and replacing them with moveable units. We ended up with a bit of a hybrid between TD and RTS.

Is your game single/multi-player, or both?

Cubemen is both single player and cross-platform multiplayer.

There are now 3 modes of play (as of the next Cubemen v1.1 update due out mid 2012)

Defence: Is a standard single player TD game with a choice from 6 game modes, 35+ levels and varying range of difficulties from single enemy spawn points to up to four enemy spawn points.

Skirmish: Is a new 2 way TD gaming experience played One vs One either against AI or another online networked player.

Mayhem: Is the new up to 6 way online gameplay experience that offers either individual play or team (co-op) play with either you versus all AI, mixed AI and Humans, or just Humans. It's AWESOME!

When did your game launch?

Cubemen launched on Steam (PC & Mac) on March 16th, 2012 and then on the iOS App Store (iPad 2+ only) and Mac App Store on the 29th March, 2012. Cubemen v1.1 is set to launch in June 2012.

So what is the future like for Cubemen?

There have already been six updates between v1.0 and v1.1 of Cubemen and I have a lot more planned for the future. More levels, More content, all as free update on all platforms.

Thanks for your time!

INDIE GAMES DEVELOPER

30 Minutes... or Less!



The Rise of AdaptivElite

AdaptivElite started as a group of hobbyists working on a relatively over-sized project called Warspace back in October of 2010. For now, all you need to know is that this custom engine called the ExEC engine is currently in a very early stage of development (playable) that will power this awesome game. This article is not about that game, though.

I, myself, am an Audio guy, but wanted to delve into Game design and Game writing. I've been through about three different core teams to start working on this idea. I was thinking in terms of Space and lots of it. Anyway, after scouring the GameDev.net boards over the course of 3 months, I came to what I call the 3rd round of skilled and enthusiastic prospects that wanted to make a game. And so the birth of Warspace started taking shape. It was about six months in, and I had just returned from my second trip to GDC. That's when I decided that if we wanted to break into the industry, we would need to do something smaller, projects that we could rapidly develop. We could use these small projects to help bootstrap our company as a player in the game

industry. I had been toying around with other game ideas at the time, one a beer-drinking game and the other a delivery driver game, so we chose between the two ideas, and started working on what would become 30 Minutes... Or Less! We eventually want to get back to Warspace, but until then, we thought we could essentially create this top-down delivery driver game in a relatively short period of time, like 3-4 months (challenge accepted).

By the time 30MOL started in mid March of 2011, I also had been playing around with company names, assuming that we were going to finish this project in a relatively short period of time and be on to the next. After some early iterations and a couple of models later, AdaptivElite, LLC was adopted as our company name. We had a team of roughly twelve members, with a good mix of programmers and artists in March of 2011. Most of the people were on board from Warspace, but the project was so daunting that a lot couldn't commit the time they wanted to it. Our team fluctuated from that initial point, eventually turning out the reasonable seven members we have now.



So June rolled around pretty quickly, and of course, my programmers didn't quite hit that deadline. We were slowly realizing how much more went into even a game as simple as this (You know, a game that could essentially run on smartphones and tablets and PC's, oh my!). It was easy, touch the screen, and the car drives towards where you are touching. move your finger around, and the car follows. And all you needed to do was take deliveries from the delivery shop to the drop points. And there might be jumps, and crazy obstacles, and a few secrets, and possibly a bunch of different levels, etc. My point is, the simple game idea I had started fleshing into something A LOT bigger than anticipated. Basically, a project that could not be done in the 3-4 months my programmers had said it would take. Kind of wish we had taken on the beer-drinking game after about 8 months, ha.

So here we are, a year and 3 months later, with what is actually looking to be quite a polished, and delicious looking game. Here's to many more! With much shorter turn-around times!

30 Minutes ... Or Less!

So, if you haven't gathered from the earlier paragraphs, 30 Minutes... Or Less! (here on in referred to as 30MOL) is a delivery driver game in which your goal is to get from the shop to the drop in 30 minutes or less. That's 30 minutes of game time, mind you. The delivery shop is an arbitrary delivery shop, and the delivery items are arbitrary delivery items, in that we kept it vague enough that you can imagine the delivery boy is delivering whatever he wants, be it pizza, sandwiches, sushi, or whatever. We kept it open like that so that the game can easily be expanded on in the future.

The premise of the game is that you're a delivery boy trying to make it in a wild, crazy world, where pedestrians and obstacles are no match for your skills behind the wheel. Drive through urban, suburban, and rural areas and possibly everywhere in-between, from the moon to underwater in possible future updates (but no promises!) as you speed your way to ridiculous tips to help you buy phat new rides.



Eventually, show your boss you're king of the road, and conquer each stage with exceptional driving skills to get promoted and move on to bigger and badder stores. Just when you thought you earned enough money to move out of your parent's basement, your final delivery might be more than you expect.

30MOL features a single player campaign with 3 different levels and 10-15 stages per level. More levels and stages will be added after the initial release. A multiplayer version is also in the works, but for now, players can compete with others across the world on the highscore boards, with fastest delivery times and most tip money earned. There are two other modes which we plan on having as unlockable. "Busy Lunch" mode is where players are challenged with randomized delivery zones, power-ups and secrets in a free driving mode. They must take as many deliveries as possible in the 11:00am-2:00pm timeframe, and are challenged with random obstructions placed on the maps. And then there's the ever-exciting "I'm Getting Fired Anyway!" mode, where your goal is to rack up as large a tab as possible by hitting cars, pedestrians, mailboxes, and the like in this free-for-all destruction derby.

The player starts with a his trusty Volksbug, and will be able to purchase and/or unlock new vehicles in game, including a sports car, a scooter, a skateboard, and even crazier forms of transportation! (Can you say segway?) There are various power-ups scattered throughout the level, including money multipliers, gas cans, NoS tanks, Immunity Insurance, and lots of extra tip money to pick up! There will be achievements that can be earned for exceptional game play.

Finally, I thought to myself, "How can I take this game one step further?" As I was playing

our early alpha build of our demo/tutorial level, I had the crazy notion of offering rewards to the player. And by rewards, I mean real-world rewards. Using location-based services on smartphones and tablets, we thought we could really up the experience of 30MOL by offering tangible rewards for exceptional gameplay: By partnering with several name-brand companies (who will remain unnamed for the moment), we hope to offer coupons and possible freebies to those that can show us they have their game-playin' skills polished. Why not reward the player with \$3 dollars off their next pizza purchase or sandwich delivery?

We plan on initially releasing 30MOL as a free to download app with in-game purchases fixed at \$0.99, but most, if not all, in-game purchases unlockable by playing long enough. I mean, you eventually have to get enough tip money in-game to be able to purchase that sports car, right? Or perhaps you want to chalk up a small chunk of change to unlock it now? The release date is TBA at the moment, but our fully functional beta will be ready by mid June. If you'd like to donate to the project, you can visit our website and click the big blue button you find on almost every page! There's a break down of the goodies we will give to you depending on what you donate, so the more you donate, the merrier the reward! Of course you don't have to donate. That's just a gimmick to give us money. LOL, but seriously, we are in the process of seeking funding. As a company, we have won some small funds to help us get started, so we are using those to help us start a small marketing campaign to raise some awareness and interest in AdaptivElite and 30MOL. Next step, Venture Capital!

So what's under the hood?

30 Minutes... Or Less! will be released for Android and iOS devices, as well as a downloadable PC version. The minimum requirements are still being worked on, but we plan on being able to run comfortably on iPhone 4/4S, iPod Touch 3rd Gen and above, and iPad. Android users will need hardware running Gingerbread 2.3 and above.

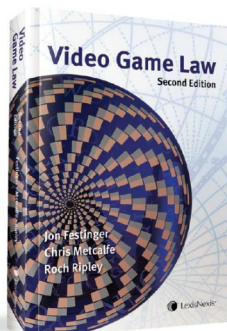
www.adaptivelite.com

Butterworths®

Video Game Law, 2nd Edition

Jon Festinger, Q.C., Chris Metcalfe & Roch Ripley

New Edition!



CDN \$80 + tax
344 Pages
Softcover | April 2012
ISBN: 9780433460756

Foreword by:

Don A. Matrick

President, Interactive Entertainment Business,
Microsoft Corporation

Video games, the precocious offspring of the Information Age, continue to be the source of some of the most fascinating and challenging questions in law. In the second edition of *Video Game Law*, the authors reveal how this burgeoning industry is creating more than games – it is posing fresh challenges for legal systems throughout the world. This work provides a comprehensive, clear and well documented overview of this rapidly expanding legal territory.

Video Game Law, 2nd Edition addresses the overlapping and emerging issues relating to IP, freedom of speech, employment, defamation, privacy, licensing and torts as they arise within the context of the video games industry, offering unique legal analysis and guidance unavailable elsewhere.



Order Today! Take advantage of the **30-Day Risk-Free[†]** Examination.

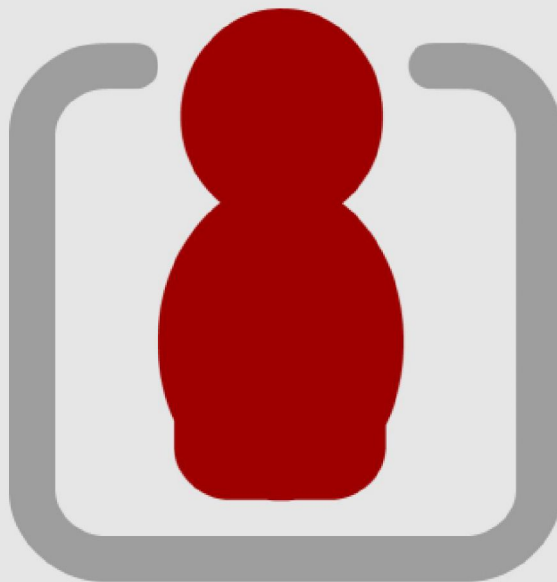
Visit www.lexisnexis.ca/bookstore or call 1-800-668-6481

[†] Pre-payment required for first-time purchasers.

Price and other details are subject to change without notice. We pay shipping and handling if payment accompanies order.

LexisNexis and the Knowledge Burst logo are registered trademarks of Reed Elsevier Properties Inc., used under licence. Butterworths is a registered trademark of Reed Elsevier (U.K.) Limited and its affiliated companies. Other products or services may be trademarks or registered trademarks of their respective companies. © 2012 LexisNexis Canada Inc. All rights reserved.

INDIE GAMES DEVELOPER AWARDS



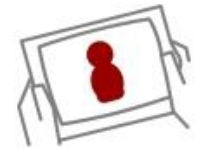
Vote daily and submit indie games from the 10th of July 2012.
IGD Awards and its sponsors support indie game developers worldwide.

www.indiegamesdeveloper.com/awards



INDIE UPDATES

We've now introduced the indie updates section of our magazine, where we will cover updates from indie developers and studios who were shown in previous issues of our magazine, as well as a select few submissions from our website.



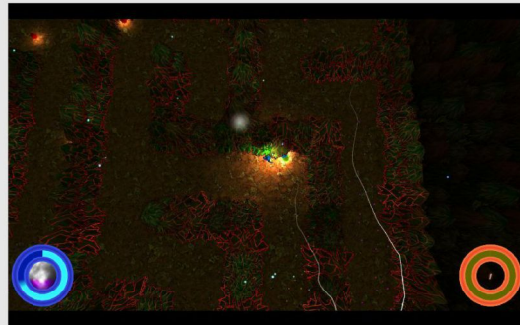
- indie
- games
- developer



contact us for further information on how we can help promote your game, tech or tool.

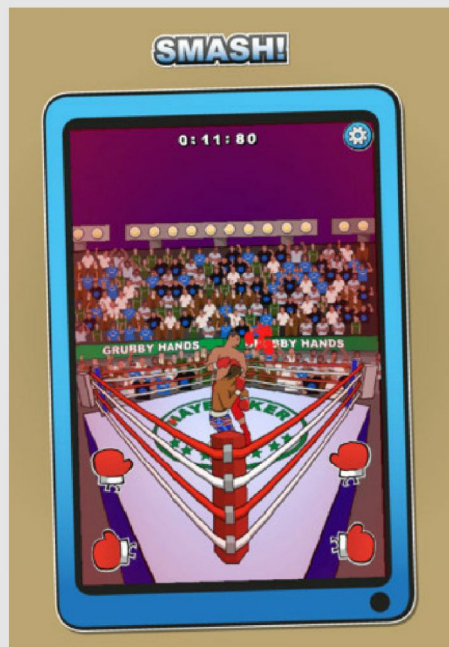


Wisp – The Redeeming (Game last shown: issue 2)
Noticeable cosmetic changes have been made to the GUI and forests since we covered Wisp in our last issue, which I'm sure you'll agree, it brings out the detail nicely. We look forward to more updates from Jungle Troll and their game Wisp over the coming months ahead. www.jungle-troll.com



David Hayes KO
(Studio last shown: issue 2)
Since the release of Boy Loves Girl; Grubby Hands Studio have released another game called David Hayes KO which is still packing a punch out there.

www.grubbyhands.com



Developers – Don't forget to send us your game updates!





FRAY

SIMULTANEOUS . SQUAD . TACTICS

www.braincandy.fr

Two year old game development studio Brain Candy talk to us about their brand new, only just released, title Fray.



Tell us a little about your game studio.

Brain Candy was founded in 2010 and we started working on Fray with a team of 5 people. As production ramped up, so did the staff and we added some 3D artists, a second game designer, a second programmer and an animator to the team. Today, Brain Candy is a 10 man team as we finish up Fray.

Working on a new IP, with a new engine (Unity), with a new team, starting from scratch can be a daunting task. The set up phase for Fray was a lot harder than expected. Had we chosen to create a platformer or an FPS game, design and preproduction phases would have been much more straight forward than when creating something a lot more original and elusive as a simultaneous turn base online game. We had to struggle with many different aspects of development, had to code special tools, develop our own set of shaders, discover technical limitations of the engine etc. It is a lot of trial and error that costs a lot of time, but is necessary to acquire the knowledge to create a game. Some decisions taken at the beginning of the project would never have been made today, but this knowledge will help us avoid pitfalls for our next game and we ultimately managed to create a product we are proud of that is truly different from anything on the market.



BRAIN CANDY



Tell us about your game.

We started working on Fray straight after creating the company with one goal in mind: update turn based strategy gaming to modern multiplayer standards, stemmed from MMOs and FPS games. Growing up, we have played large amounts of turn based strategy and multiplayer FPS games, and we wanted to combine the feeling of both styles, by speeding up the game play all the while retaining the strategy elements of the genre. We therefore started working on a simultaneous turn system where all players acted at the same time to reduce downtime to a maximum, and also allow a higher number of players in a single game session. To accompany the gameplay, we have created an expansive cyberpunk universe, with a rich background, in which players pledge their allegiance to a corporation and fight with a squad of 4 characters, armed to the teeth with futuristic weapons and gadgets against other players. After two years of development, Fray was finally ready to launch on June 19th for the PC.





Technical Features.

Fray is compatible with PC Windows systems, and the minimum specs are a Dual Core, 3GB Ram and a Nvidia 8800GT or higher. Being a multiplayer online game, Fray requires an active internet connection.



VISIT OUR
EVENTS DIARY
ONLINE
WE ADD WORLDWIDE
EVENTS DAILY
FEATURED + LINKS

www.indiegamedeveloper.com/events-diary

INDIE GAMES DEVELOPER



TECH SHOWCASE



We couldn't wait to introduce our very first Tech Showcase after looking into simulation games and racing cockpits. During our research we found **GT Omega Racing**, a UK based company that started out by designing an affordable racing cockpit for sim racers back in 2009. They have multiple products, each with numerous configurations and optional accessories. In this showcase we're going to take a look at the GT Omega Racing Simulator Supreme, a high end cockpit and a GT Omega Steering Wheel Stand at the lower end of the pricing range.

The GT Omega Racing Simulator Supreme cockpit comes with many accessories to create the Ultimate sim racing experience with the added option of single or multiple monitors, and the best thing about this configuration is that they include both stands in the package. The rigid racing cockpit frame mountings house the wheel and pedals, along with an authentic racing seat, a console tray, keyboard and mouse tray and gear shifter mount with added speaker mounts to the rear.

For extra durability, the triple monitor stand is not mounted onto the frame. So you won't end up flipping those monitors when you get carried away with your driving experience. However, you can be safe in the knowledge that any force feedback from the wheel or movement in the seat will not affect the stability when mounting a single monitor to the cockpit.

The GT Omega Racing Supreme has proven to be a popular cockpit and is backed with excellent customer support and rave reviews.

GT Omega Racing Simulator Supreme

Price at press: £299.96 / \$565.09 + vat @ 20%.
Delivery and duty taxes may also apply.

Website: www.gtomegaracing.com



Cockpit does not include steering wheel, pedals, monitors, speaks or other accessories unless specified. For full specifications visit the GT Omega Racing website.





TECH SHOWCASE



For those of us whose space is more limited and for whom a full rig just isn't practical then the GT Omega Steering Wheel Stand offers us the next best thing. Not only does it arrive in a deceptively compact box, once assembled it is foldable for easy storage! The wheel stand is fully adjustable, meaning you can fine-tune both the height (between 56cm and 77cm) and the angle to suit the player, and therefore it is practical for players of all ages. In addition to this you can also adjust the angle of the actual wheel plate! The strength and durability of the frame have meant that the designers have been able to do without a middle support bar giving clear pedal access, making for quicker, cleaner smoother gear changes.

The stand is pre-drilled for: the Logitech G25/27 wheel, pedal and shifter; the Fanatec heel and Clubsport pedals; the Thrustmaster T500RS wheel and pedals and the Thrustmaster F430 wheel and pedals. The gear shifter mount comes pre-drilled for the TH8RS and Logitech G25/27 gear shifters. It is mountable on both the left and right hand sides of the stand. It is compatible with the CSR pedals and is suitable for wheels that can be clamped to tables.

This product has proved very popular and has received very positive reviews.

For full specifications visit the GT Omega Racing website.

GT Omega Racing Steering Wheel Stand

Price at press: £81.63 / \$127.92 + vat @ 20%.
Delivery and duty taxes may also apply.

Website: www.gtomegaracing.com



Stand does not include steering wheel, pedals or other accessories unless specified. For full specifications visit the GT Omega Racing website.

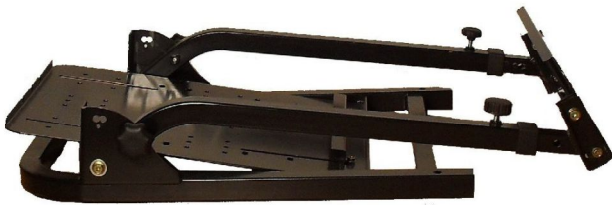




COMPETITION



GT Omega Racing have provided us with one of their Steering Wheel Stands to offer one lucky reader a prize in our competition. To enter, visit the following page on our website:
www.indiegamedeveloper.com/gtor-competition



**Competition Ends:
31st of July 2012**

Stand does not include steering wheel, pedals or other accessories unless specified. For full specifications visit the GT Omega Racing website.

Competition terms and conditions apply, see website link for details.



Starting an Indie studio is rarely easy. Dmitriy and Aleksey from One Button Studio in Russia explain the ups and downs they have experienced.

GUN SWORDS

T I N S O L D I E R S

Who are you?

Dmitriy Mikhalev – CEO and Co-founder of One Button Studio

Aleksey Baryzhikov – Game designer and Co-founder of One Button Studio.

How did you get started?

AB: Before joining my friends Dmitriy and Sergey, as well as co-founding One Button Studio, I worked for one of the major Russian companies “1C”, in one of its game development studios.

Working in a big corporation is often a very restraining experience, and that was true in my case. Imagine me, the guy who designed board games since childhood, decorated them and acted as game master, just sitting and doing 3D models for someone else’s project. It felt like not enough air for me. That’s why when the big project was finished, I was seriously thinking about moving to indie development.

Thanks to good odds I was not alone in my quest, so in August 2010 Sergey Popov, Dmitriy Mikhalev and I officially founded One Button Studio.

DM: I was a Counter-Strike pro-gamer when I was a student. I took a break to work in the financial sector for several years and finally got back into the gaming industry to make a game for the fans of E-sport. The game is called “Gunswords”, which is influenced by such games as “X-COM: Terror from the Deep”, “Incubation”, and others.



How quickly did you expand?

AB: During the last year the number of core developers has been stable, but we always expand the list of skilled free-lancers who can add a special touch to the visuals and game features.

DM: Back in August 2010 our initial staff consisted of two artists and one businessman, so we had to find a programmer asap. That was harder than we thought. All programmers that we knew were sitting in warm places in major studios and didn't have a burning desire to surrender that for a startup indie project. We started posting vacancies on all job websites and interviewed many candidates. After a month we ended up with a part-time flash programmer, who inspired us by quickly, just in one evening, generating simple prototype of a combat system for GunSwords, which was originally called "Guns'n'Swords" back then. After choosing Unity as the game engine, we gave the programmer preliminary text that was the basis for our design document and thus we completely sunk into creating art for the game (the thing that we were best at back then). After three weeks we met our programmer to see the first version of our product. We were expecting it to be with characters moving across the battlefield, basic interface functioning and so on. Unfortunately he had not invested enough of his time into the project. Furthermore, he stated it was not possible for him to make such a game working only part-time and therefore left the company. What could we do? We started posting vacancies and doing interviews again.

Eventually we found several new candidates. One of them having solid experience in C++, very enthusiastic, self-motivating and highly paid. His biggest flaw though, which was not apparent to us at the time, was that he didn't have any gaming industry experience and was not a fan of game play in the slightest of sense. Long story short, we worked with him for half a year, generated a lot of documentation, thus I explained to him every little aspect of the game we were creating, organized version control, task tracking system and so on. Months passed, and 2010 was coming to an end, the game art was almost completed, but no working version of the game or even a basic prototype had appeared. That's when we realized we had to expand even more.

Vacancies, interviews... finally we hired 3 programmers from a Russian province for working remotely. One of them had experience (worked as a web-developer, and more importantly – programmed his own MUD), the others had just graduated from different universities. The experienced programmer is Aleksey Senilov, he's from the city of Kirov. He's disabled: he is mute and unable to use a mouse, but has a very bright mind and programming talent. I must give him credit, we would've failed at that stage of development for sure without him. The other two programmers were Aleksey Pustokhailov (works with us till this day) and Maksim Valeev, both from Astrakhan.



The next several months were exceptionally productive compared to past experiences within the development of the game. We got the combat system working with animation, special effects, squad generator, and first iteration of lobby interface, so thus the game was starting to look like a whole at last. However, we then had some major staffing issues, which led to a grim atmosphere inside One Button Studio, and in turn led to a rapid decline in productivity and brought the project to the brink of closure. It took half a year to recover. In October 2011 we rebounded, just us three musketeers like in the beginning, as well as adding again Aleksey Pustokhaliov. By the end of 2011 we hired two good programmers (Nikolaiy Timofeev and Pavel Ryzhikov), reviewed our project and started working at maximum capacity like never before. Now we have a solid and productive team.

What would you do differently if you could go back and do it again?

AB: Actually there are so many things that I would do differently. I'd say in hindsight, the most important thing is having a clear cut project implementation strategy and a basic overall final plan that is specifically drawn out. Obviously when dealing with a start-up things are constantly changing within the company. Overall, though I would say that I wouldn't start production until there is a clear design document, not just game concept, and with major features assigned to specific production stages as well, with adequate estimation of human capital and time resources.

Then I would hire "world class" gaming programmers and have them sign contracts to start and finish the program. Consistency is key, not "cowboy-type" programmers. Actually finding people that suit your company's culture is a subject worth a whole different interview.

Taking into account the complexity of "GunSwords" and it's sandbox character, I would concentrate on making a working version asap. For example it could look like a basic combat system with schematic visuals and a functional interface, thus keeping every aspect flexible until major game mechanics are settled and worked out. Concentrating on programming support and overall management of the company rather than art production is crucial during this period. After that we would be able to pack the core game into beautiful visuals, continually adding secondary features and then start a marketing campaign, and begin looking for publishers with a working version of the game ready to be presented.

DM: If we could start from the beginning we would split production of the project into three parts. First is project conception and detailed feature planning for 20% of time till release, production itself for 60% and usability/bug testing for the last 20%. Spending more time searching for skilled and feature-oriented engineers, would have saved us huge amounts of time and resources.



Tell us a little bit about the story behind the game?

AB: I'd better say we have a setting, not a story. Since it's a multiplayer game the stories in it emerge from twists and turns of a combat. You can see players tell stories to each other about saving the last fighter in a battle during a very close situation when odds of winning seem almost non-existent. Because these stories are real, they are often more valuable than a carefully written single-player campaign plot. So basically the setting of GunSwords is: bands of fighters from different times and places, gathered on mystical arenas, to prove their valour. The latter is up to players.

DM: We are all fans of E-sport games, especially Quake3, CS, Starcraft and Dota. That's why we designed GunSwords as a highly competitive game in an unfairly forgotten genre of turn-based fights. We have considered all good ideas from the best turn-based games. But we had to redesign everything to fit the goal of making short (5-15 mins) and never-the-same battles. And we did it! We've played against each other for more than a year and want to play GunSwords more and more, because there are still a lot of fighting tactics to try out.



What is the objective of your game?

AB: In combat – killing all the fighters of your opponent. Speaking about the game in general, the ultimate objective is becoming a champion in your favourite weight class and keeping the title for as long as you can.

DM: Let's put it simply: the objective is to win fights and build up a squad that will fit your tactical style. Climb up to higher weights and meet more skillful opponents.



Is your game single / multi-player, or both?

AB: The main game mode is PvP multiplayer, but there's a PvE content also in form of survival battles against waves of AI-controlled golems attacking your squad and a boss in the end. We intentionally kept away from a single-player campaign, because we want our game to be the best turn-based competitive arena out there.

Is your game free, a one off cost, monthly payments or micro transactions?

DM: GunSwords is a try-before-you-buy with a 10-days free trial.

Is there an age restriction?

DM: Of course rating should be done by appropriate organizations. But GunSwords is supposed to be for everyone. We don't have blood/gore, obscene language, sexual references, or other socially-inappropriate items. The only exception may be some cartoon violence. But our fighters are obviously stylized, they are tin soldiers, so we think we will be ok.

What features will your next game release have?

AB: Features that we plan to implement in the near future are:

General improvements

- Barracks – advanced squad customization interface
- Powerful special attacks for each weapon type

PvP content

- 2v2 matches
- Clans – it will allow players to join clans and interact more effectively, for example tutoring less skillful members and contributing to combined clan rating

PvE content

- Missions – survival arenas with AI-controlled golems attacking your squad
- Map editor – best custom maps will be added to the overall list of maps on which ranked matches are played
- Player's fortress, which is a PvE map built by a player that can be assaulted by any other player (we'll say more about this feature in time)

Concerning release date I can say that we are working hard to make it as soon as possible, but it depends a lot on us having funds to finish the current development stage and provide a high level of quality. So no particular date is set for now.

Are you looking for funding / investment to further your development?

DM: Yes we are. On May 16, 2012 we started a campaign on Kickstarter.

What platforms does your game support?

DM: At the moment we support PC and MAC. In the near future we plan to add a browser version to play on our website. Later we plan to port it on IOS and Android devices. There is still nothing on appstore like GunSwords. Hope it won't be too late when we come up with GunSwords on Ipad.

Does your game require any web plug-in's to play it?

DM: It will require Unity Web Player plug-in to play GunSwords browser version.

Does your game require any additional hardware?

DM: No additional hardware is required.



What are the minimum software/hardware requirements to play your game?

AB: 2 GHz CPU, 1 Gb RAM and Video Card capable of anything more than browsing websites and playing video will be enough. And connection to the Internet, because the game is online only.

What are the recommended software/hardware requirements to play your game?

DM: 2 GHz CPU, 2 Gb RAM and Video Card compared by performance to GeForce 300 Series. Internet connection.

Does your game require a download patch after the initial installation?

DM: We have automatic patching/updating system. 1-2 times a week we make updates to the public version of GunSwords.

Thanks guys for an in-depth interview. IGDM



www.derrickthedeathfin.com

Gordon Midwood, of Different Tuna, talks about the hugely exciting, wonderfully original and soon to be released Derrick the Deathfin. Gordon himself describes the development of Derrick as “long-winded” but it seems to us to be a real labour of love.

Tell us a little about your game studio.

Different Tuna was started as collaboration between Different Cloth and Ronzo with the intention of creating handcrafted, organic videogames that are also a little bit silly. We are really just 2 people hiding behind the guise of being companies. Different Tuna’s first project – and it’s most successful to date! – is Derrick the Deathfin, which is coming to PSN this very summer.

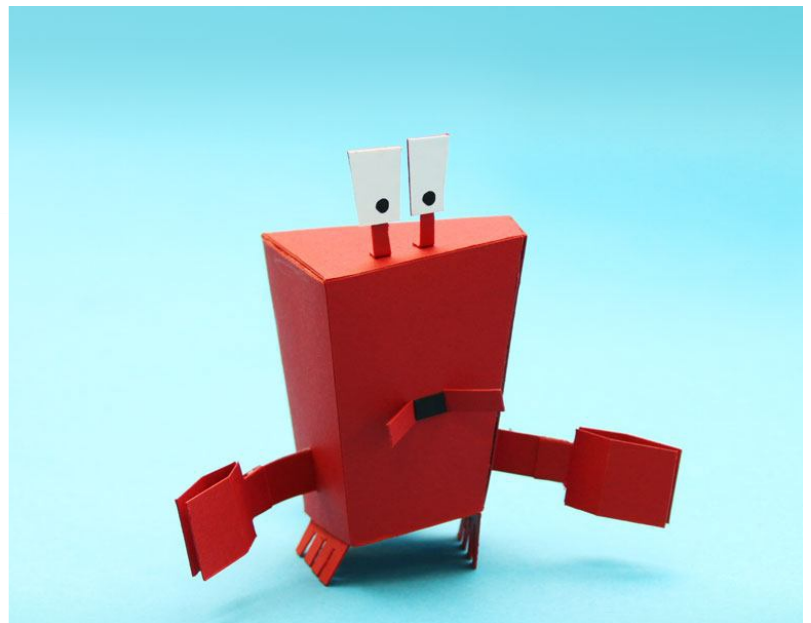
Derrick has been generously funded and supported by Channel 4 and Screen Yorkshire, and we have worked with a host of collaborators along the way, including

Tuna, who helped us out with the initial tech side, Pitstop, who made all the excellent audio in the game and ten24, who are nothing short of 3d geniuses.

Different Tuna is still only comprised of the two of us though, and we have no offices or staff or credibility to speak of! I work from home, in the loft of dreams to be precise, and Ron is based out of his art studio near Brick Lane.

We have no real plans for the business to expand or things of that nature, we just hope that bringing Derrick to PlayStation will allow us the opportunity to make the next game!





Tell us about your game.

Derrick the Deathfin is the world's very first underwater paper craft video game. It is a fast-paced, irreverent and beautiful adventure which tells the tale of Derrick and his worldwide rampage of destruction after the "soupification" of his parents gave him a few anger management issues.

Derrick travels through 32 levels, 11 environments and 4 different continents on his epic quest to avenge the death of his folks and make himself feel a little bit better. On his way he must battle against his overactive metabolism, constantly eating and moving to stay alive whilst collecting diamonds and jumping through flaming tyres in the sky. He also encounters against-the-clock challenges and destructive puzzles along the way.

Derrick's gameplay can be described as a combination of Ecco the Dolphin and Sonic the Hedgehog, with a little bit of Joe Danger thrown in for good measure. It is intended as a fast-paced, explosive and accessible game with hidden depths for advanced players.



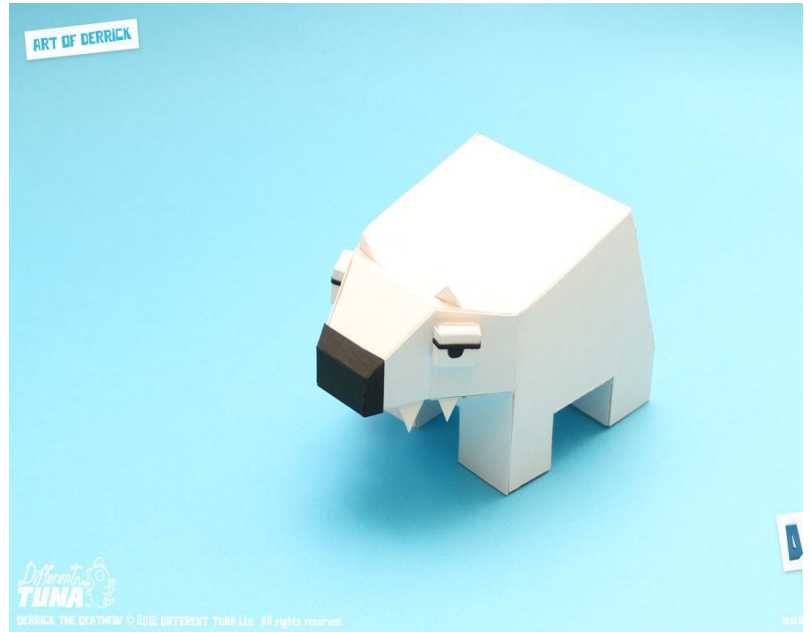


Technical Features.

The only technical features that Derrick possesses are his intricately cut triangular teeth. Derrick the Deathfin is a single player game coming exclusively to PSN this summer, so the only requirements for entry are a PlayStation 3 and some good taste.

Derrick's art style is based on the character designs and all-round irreverent genius of Ronzo and is set in a world entirely comprised of paper and card. The process of making characters and scenery for the game is quite unique; typically Ronzo sketches objects, creates them out of paper, photographs them from all sides and then sends them on to ten24 to be modelled and animated in 3d, which are then imported into the game. This process is quite long-winded, but we passionately wanted to produce a world full of colour and life and idiocy unlike anything we had seen in games before.

Derrick will be shredding his way onto PSN later this summer.



WOODEN SENSEY

Upper Byte is a French indie studio, founded by Camille Masson and Cedric Cadiergues, in 2011, having worked together for a few years in other game studios before starting their own. More than just colleagues, they consider themselves to be friends and this is really important to them.

Tell us a bit about your studio.

Our main objective is to produce games we would like to play too, without being restricted by licenses or editor schedules. Due to the small size of our studio (the game was developed by just the two of us), the most important factor in successful working, is the way team members complement one another. You have to be reliable within your field of expertise and able to discuss things that are outside of your speciality. At Upper Byte, for example: Camille is in charge of the programming part while Cedric handles the graphic side, but we're both working on global game design and gameplay aspects.

Just like many developers we're very fond of old school games, so we decided to pay homage to the platform genre with our first production: Wooden Sen'SeY.

Tell us about Wooden Sen'SeY

Wooden Sen'SeY is our first project, it has required more than a year of development and is almost finished. The game will be released this summer on dematerialized platforms for about 10 dollars. It's a 2.5D single player, platformer for PC and Mac. It introduces you to Goro, a village chief, in a colourful Japanese-style universe threatened by an evil thief and his army of minions.





Indeed, Goro was living peacefully, drinking his favourite beverage called SeY, when suddenly, a flying machine controlled by the unfriendly Yokai, your arch enemy, steals all the SeY bottles ! Furious, Goro goes back to his house, takes his axes and gets ready to fight the whole world in order to recover the precious cocktail. One important thing you will learn about Goro's gameplay is that you will have to wisely use his axes because he can use them not only as minion-slicer, but also as grapnel or propellant.



The player will be going through 9 unique and epic levels, using a submarine underwater, fighting on flying ships over clouds, slicing monsters in underground caves, and so on. We wanted our game to be as exotic and fun as Goro's journey. His quest will bring you into original environments embellished with beautiful and punchy music themes composed by our friend Maxime Ginolin, a.k.a. MagiCJack.



Another important point about the game is the difficulty level. Do you remember playing old games, having to learn by heart the beginning of a level, just to reach the part you failed at miserably the last time? Do you remember how proud you were when you actually did it? We tried to induce the same feelings with Wooden Sen'SeY. The player can really see himself handling a difficult stage "like-a-boss" after a few retries, and getting proud of his course. However, there is the risk that he will do it too hastily, taking inconsiderate risks which, sometimes, lead to the death and game over screens. The key to success is to stay concentrated, be meticulous, precise and not too greedy. It is also important to not get impatient when retrying a stage 2 or 3 times!



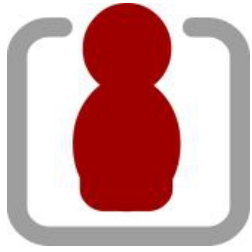
The game will be available for PC & Mac for around 10 dollars and there is no age restriction, no violence or shocking content, not even a blood drop! We've been testing the game on several types of hardware. The slowest was a dual core 2,2Ghz, with 2G RAM and a GeForce 320M. The game ran smoothly with good quality (real-time shadows, bloom and nice textures). It can run on much slower hardware with real-time shadows and lighting disabled. We recommend a better graphics card to enjoy the game with HD textures and smooth shadows!

For input devices, we let people play as they want. Everything can be changed in the options screen. If a gamepad is plugged in, it is selected as a default device; otherwise, the character is controlled with the keyboard + mouse combo. Overall, playing with a gamepad is easier, but the keyboard and mouse combo, while being a bit harder at first, is more precise.



www.upperbyte.com





EVENTS DIARY



July 2012

6th - 7th July - Rezzed - The Brighton Centre, Kings Road, Brighton, East Sussex BN1 2GR, **UK**
10th - 11th July - GamesBeat 2012 - Palace Hotel, 2 New Montgomery Street, San Francisco, CA 94105, **USA**
10th - 12th July - Develop 2012 - Hilton Brighton Metropole, Kings Road, Brighton, East Sussex, BN1 2FU, **UK**
10th - 12th July - World Gaming Executive Summit 2012 - W Hotel, Barcelona, **Spain**
12th - 15th July - Comic-Con San Diego - San Diego Convention Center, San Diego, California, **USA**
14th July - Chicago Experimental Game Development Group Video Game Summit 2012 - Heron Point Convention Center, 665 W. North Avenue, Lombard, Illinois, **USA**
24th - 26th July - Casual Connect Seattle - Benaroya Hall, 200 University St., Seattle, WA 98101Seattle, **USA**
26th - 29th July - ChinaJoy - New International Expo Centre, Shanghai, **China**
27th - 29th July - AVCON 2012 - Adelaide Convention Centre, North Terrace, Adelaide, **Australia**
28th - 29th July - California Extreme 2012 - Hyatt Regency Santa Clara, 5101 Great America Parkway, Santa Clara, California 95054, **USA**

August 2012

5th - 9th August - Siggraph 2012 - Los Angeles Convention Center, Los Angeles, California, **USA**
9th - 11th August - Edinburgh Interactive - The Radisson Blu Hotel, 80 High St, The Royal Mile, EH1 1TH Edinburgh, **UK**
13th - 15th August - GDC Europe - Congress-Centrum Ost Koelnmesse/ KölnKongress, Cologne Congress Center East Deutz-Mülheimer Straße 51, 50679 Köln **Germany**
15th - 19th August - GamesCom - Koelnmesse GmbH Messeplatz 1, 50679 Cologne, **Germany**
31st Aug - 3rd Sept - Gateway 2012 - Sheraton Gateway Los Angeles Hotel, Los Angeles, California, **USA**

September 2012

5th - 6th September - Games Conference New York - Museum of Jewish Heritage, New York, **USA**
11th - 12th September - Cloud Gaming USA - Convention Plaza Hotel, San Jose, California, **USA**
19th - 20th September - 3D Entertainment Summit - Hollywood and Highland Center, Hollywood, California, **USA**
20th - 23rd September - Tokyo Game Show - Makuhari Messe, Tokyo, **Japan**
26th - 27th September - LOGIN Conference - Bentley Reserve Hotel, San Francisco, **USA**
27th - 30th September - Eurogamer Expo - Earls Court One, Earls Court, Warwick Road, London SW5 9TA, **UK**

October 2012

9th - 11th October - GDC Online - Austin Convention Center, 500 East Cesar Chavez Street, Austin, TX 78701, **USA**
18th - 19th October - AppConference - Santa Clara Convention Center, 5001 Great America Pkwy, Santa Clara, CA 95054, **USA**
19th - 21st October - Casual Connect Kyiv - Rus Hotel, Kyiv, **Ukraine**
18th - 21st October - SOE Live - Bally's Hotel and Casino, Las Vegas, **USA**
24th - 25th October - GameSoundCon2012 - Pyramind Studios, San Francisco, CA, **USA**
31st October - 4th November - Paris Games Week - Hall 1, Paris Expo, Porte De Versailles, Paris, **France**

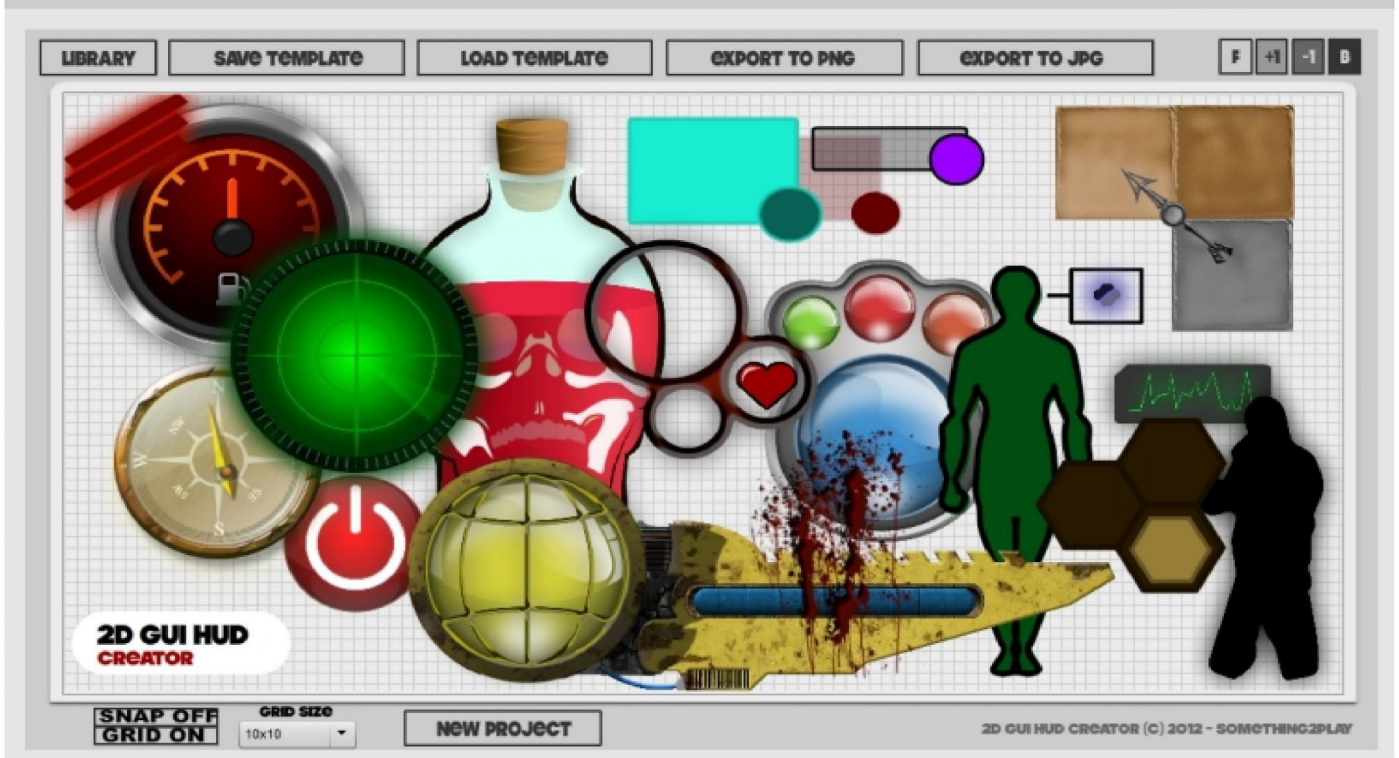
November 2012

8th - 11th November - G-Star★20112 - Bexco Exhibition, Busan, Republic of Korea
17th - 19th November - GDC China - Western Hall No.2, Shanghai Exhibition Center, Gate 7, Shanghai, **China**
27th - 28th November - Dubai World Game Expo - Dubai International Convention and Exhibition Centre, Dubai, **UAE**
28th - 30th November - Game Connection Europe - Venue to be confirmed - Paris, **France**



TOOL SHOWCASE

2D GUI HUD CREATOR



2D GUI HUD Creator is fast becoming a popular GUI and HUD prefab creation tool for indie game developers with little or zero experience in art. Saving valuable time and money that a new start-up may not have available to spend on art. At least not until they can create a prototype and secure crowd funding or investment to help complete their game with some paid help.

With just a few clicks of your mouse, you can easily put together many combinations of prefabs with the added option of applying colour filters, alpha transparency and effects, such as; glow, bevel and drop shadow.

Artwork updates are free after your initial purchase, however there is currently no automatic update functionality, so you will need to uninstall the existing installation before you can benefit from the ongoing updates.

Incidentally, the artwork shown in the above

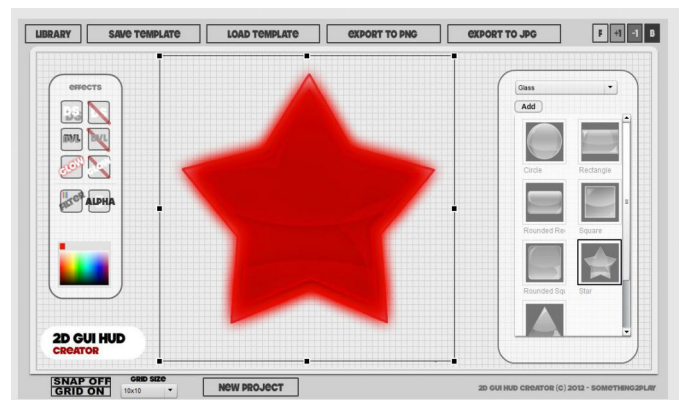
screen shot does not show all of the artwork available and does not really do this tool justice.

Artwork can be exported in PNG and JPEG formats, with the option to increase or decrease the scale on export.

Price: \$29.99 (£19.20)

Website: www.2dguihud.com

Demo available with restricted save functionality.



INDIE GAMES DEVELOPER



GameZebo, Sep 2011

"App of the Week"

Tap! Magazine, Oct 2011

"The [iPad] multiplayer is one of the best on iOS devices"

148 Apps, Nov 2011

"Simple to describe but complex to master...
Greedy Bankers delivers in short, healthy doses"

DIYGamer, Feb 2011

**EUROGAMER INDIE GAMES
ARCADE FINALIST 2011**

Created by indie developer Alistair Aitcheson,
praised as one of game development's rising stars
in Develop Magazine's *30 Under 30*, Jan 2012

Now FREE on iPhone!

Follow the developer on Twitter: @agAitcheson
Videos, info and reviews: www.greedy-bankers.com

Available on the
App Store