What went right

gamasutra.com/blogs/ArminIbrisagic/20150220/236427/Goat_Simulator_Post_Mortem.php

Blogs

by Armin Ibrisagic on 02/20/15 06:19:00 pm Featured Post

The following blog post, unless otherwise noted, was written by a member of Gamasutra's community. The thoughts and opinions expressed are those of the writer and not Gamasutra or its parent company.

I've previously written about how Goat Simulator came to be our next game a couple of weeks before the game was released, and I remember finishing off the article with the following words:

"It could turn out great, it could also turn out terrible, but in either case, it'll be really, really interesting."

At the start of January 2014, some of our artists and programmers were developing a prototype for a new, big IP that we were going to work on. We didn't have work for everyone at the studio, so we decided to do an internal game jam until the first parts of the pre-production were done for our "real" game. A couple of weeks later, we put up a gameplay trailer on Youtube, received a million or so views in a week, and were bombarded into turning Goat Simulator into our next "real" game. It's by far been the strangest ride of our lives. It might be a stupid game about a goat, but it has changed our lives faster than we thought was possible. Before we released the game, we were unsure of how it would be received once it was out on Steam. Today it's safe to say that it's the most successful game we've ever made.

Perhaps the only thing that was more strange than the game itself was the way it was developed. We've discarded a large part of game development practices that are usually fundamental to how you make games. We've made a game in a way we never thought we would, and it actually worked.

1. Used our limited time where it mattered

Initially, our plans with Goat Simulator weren't very big - we were just playing around with a goat while waiting for work to open up on the "real" project. We didn't have that many artists working on Goat Simulator from the start, so we bought most of our assets from the internet. This still meant that we had to remake nearly all of them to improve performance, but it was still much, much faster than making every single asset ourselves. This allowed us to focus our development time where it actually mattered the most. Naturally, this might not be a way to go for every game - if you have a really unique artstyle (like we did when making Sanctum and Sanctum 2), buying all your assets from the internet is impossible. But we've learned that if you're making a game with a fairly normal artstyle, then there's no reason to make a car or a table yourself when there are dozens to choose from on the internet. It's better to put your artists on making unique content or polishing up the environment than to create things that already exist. Fun fact: the 3-D model for the actual goat in Goat Simulator cost $25 on TurboSquid. But it was on a 75% off sale, so we got a pretty good deal on it.

Another thing that allowed us to use our time more efficiently was that we decided to use Unreal Engine 3. It allowed us to start working on the actual game from day one, and cut out a lot of overhead time. In just a couple of days, we already had a goat running around that ragdolled when it collided with something. This did wonders for morale at the office, as you could see progress on the game very fast.

Even though the better and prettier Unreal Engine 4 was (almost) out when the development of Goat Simulator started, we still went with the older version since that was the one we knew best and worked fastest in. We've always tried not to be the first ones to adopt new technology, as that usually adds more overhead time and keeps us from focusing on the actual gameplay. Much like buying assets off of the internet, it allowed us to use our time on implementing cool features instead of learning a new game engine.

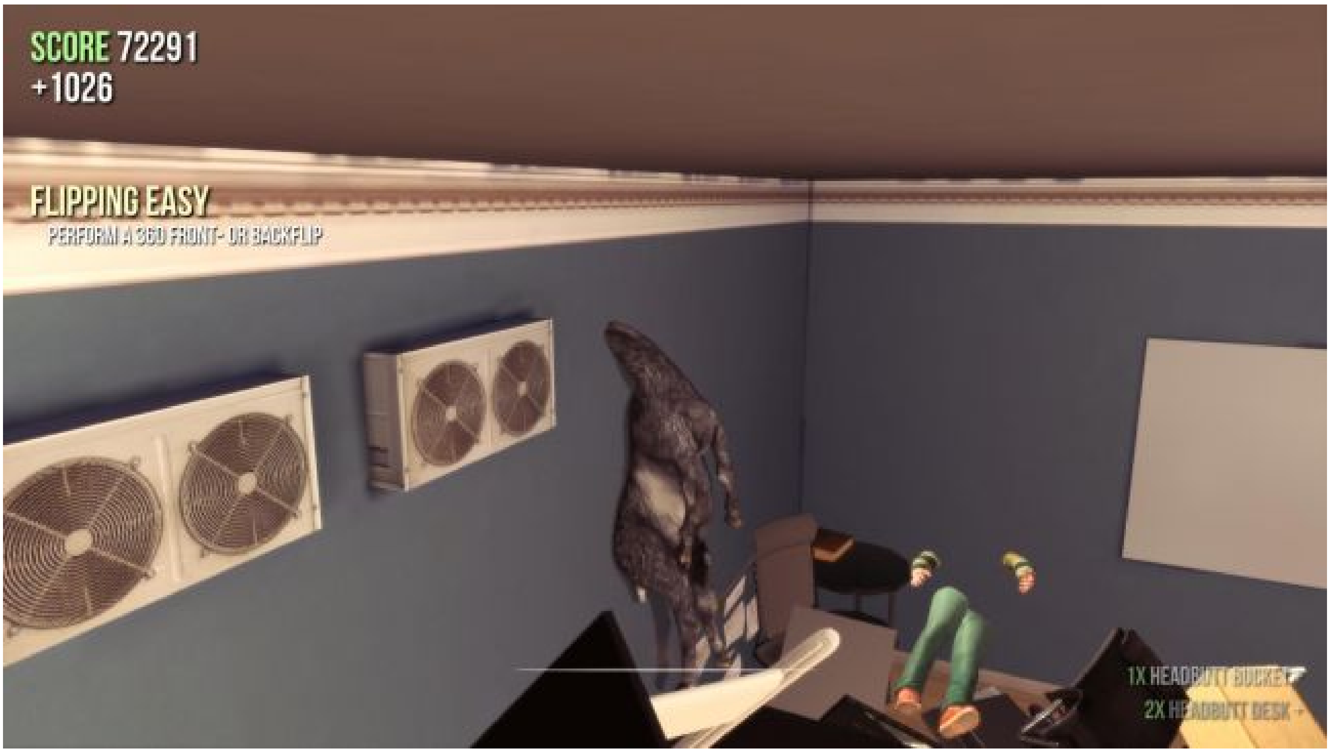
2. No planning or long-term think

Since we never intended for the game to be released initially, we didn't start out with a proper project plan and we didn't have predetermined roles of who was going to do what. This resulted in everyone feeling like they really had the freedom to create whatever they wanted, did wonders for morale at the studio, and simply resulted in way funnier content than if we had a top-down approach from start. One of the most important parts of our employees not having a pre-set schedule was that everyone could find some time if they suddenly had an idea for something, or if one of their co-workers had an idea. For example, one of our artists figured one afternoon that it would be funny if you had a jetpack, made a 3D model of it, and implemented it into the game before the day was over. Now the jetpack is one of our most popular features.

Another example happened a couple of weeks before release when I was in an interview with a games journalist. He asked me why you played a goat in the game and why not a giraffe or a bear. I jokingly told him that all animals are goats - giraffes are just really tall goats and bears are just really big and fluffy goats. It was kind of funny, we laughed a bit, but I didn't think much of it then. Once the interview was over and I was going back to my desk, I figured HEY WAIT A SECOND, and I went to our animator and asked if he had time to animate a giraffe. He said "yeah, sure, I have some time," so we bought a 3D model of a giraffe off of the internet, animated it, and at the end of the week it was in the game. It was called Tall Goat everywhere in the game, even in the code. As soon as it was inside the game and you could run around as a giraffe, kick things and get hit by cars, we just knew that the Tall Goat was here to stay. Now, the fact that you can unlock new types of goats is one of the most important features we have, and it’s a feature we’d never have if we had a fully-loaded production schedule and no time to do sudden strange stupid ideas.

3. Try really hard to not try too too hard

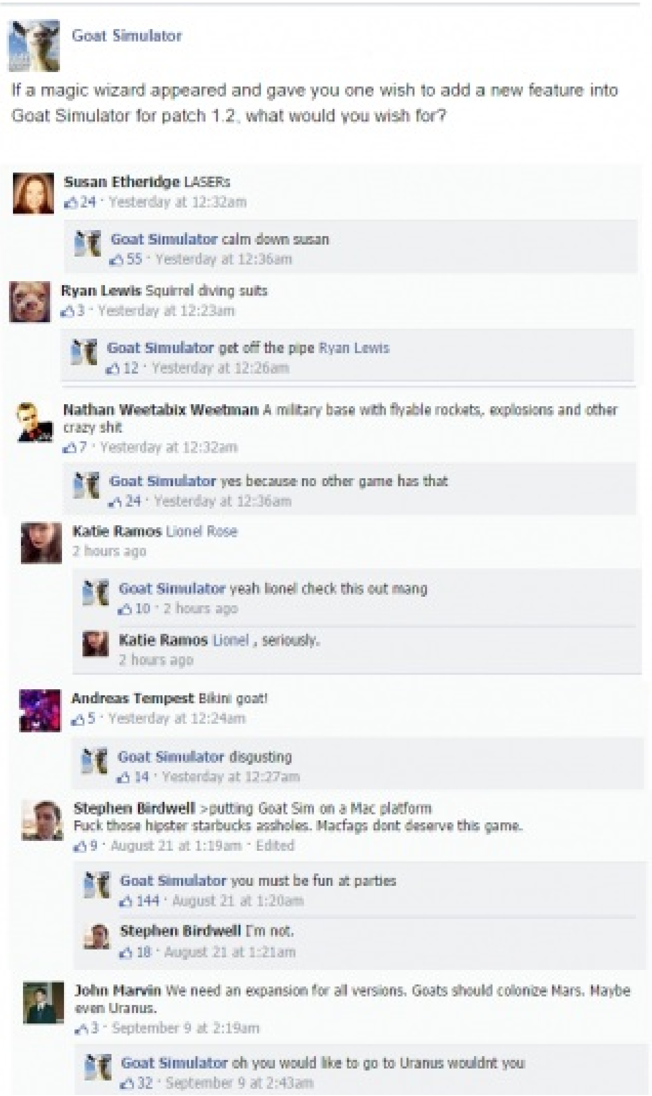
This has by far been our most important design mantra. Once Goat Simulator went viral on Youtube, we sat down and had a talk about what our collective vision for the game was. Since we had kind of designed the game on a whim, and the whole purpose of the project was to have something to do while pre-production on another game ended, we were largely lacking focus. After the trailer had received over a million views, we sat down and had a very long design discussion about the future of Goat Simulator. Some of our fans asked us on Twitter to release the game immediately, while others asked for a full-on Grand Theft Auto game where the protagonist is a goat. Some people were just tweeting goat sounds at us. We didn’t have much money left in our company, but with the huge buzz around our first alpha trailer, we were pretty sure we could have received funding from a publisher if we wanted to extend the development time of the project. However, we felt that a AAA goat game with cutscenes, different NPCs, missions, and a $60 price tag wouldn’t really have fit into the spirit of what should be Goat Simulator. We think the trailer went viral because the game was small, buggy and a bit stupid, so if we would have taken the game super seriously and tried to expand on it in all directions, the whole thing would have felt too “try-hard” and lost its soul and humor. So we decided to make a game that was as similar as possible to what our first trailer had shown, to keep all the bugs that made the game so much fun, and to not try too hard or take ourselves too seriously while doing so. We think that’s the main reason why Goat Simulator became the labor of love that it is today.



Naturally, this might not be something that applies to every single game, but we think that having a level-headed approach is really important once something huge and unexpected like Goat Simulator happens in your life. Getting too excited and selling out for promises of **MAXIMUM PROFIT** short-term is something that hurts the charm and quality of a game in the end.

4. Fan interaction over social media

Being the person that does most of our PR work, I always try to have a contact with our fans that's very down-to-earth. I honestly communicate with our fans the same way I would communicate with my real-life friends, making dumb jokes and not always taking everything very seriously. I feel that it fits well with the spirit of Goat Simulator. Instead of using extremely vague pre-written responses which are often seen from bigger corporations, I just use the same honest and "don't try too hard"-approach with our community as we do when it comes to development. I think having this relaxed approach helps us connect much better with our fans. When you're talking with someone from the Goat Simulator team, you shouldn't feel like you're talking to some corporate PR person in a suit, you should feel like you're talking to a normal 20-something that's sitting in front of their desk, rubbing their tired eyes and sipping a cup of morning coffee, because that's exactly how it is on our end. This approach has given us extremely positive feedback from our fanbase, and our Facebook likes have steadily been rising. Today we have about 210,000 likes.



Goat Simulator

If a magic wizard appeared and gave you one wish to add a new feature into Goat Simulator for patch 1.2, what would you wish for?

Susan Etheridge LASERs
24 · Yesterday at 12:32am

Goat Simulator calm down susan
55 · Yesterday at 12:36am

Ryan Lewis Squirrel diving suits
3 · Yesterday at 12:23am

Goat Simulator get off the pipe Ryan Lewis
12 · Yesterday at 12:26am

Nathan Weetabix Weetman A military base with flyable rockets, explosions and other crazy shit
7 · Yesterday at 12:32am

Goat Simulator yes because no other game has that
24 · Yesterday at 12:36am

Katie Ramos Lionel Rose
2 hours ago

Goat Simulator yeah lionel check this out mang
10 · 2 hours ago

Katie Ramos Lionel , seriously.
2 hours ago

Andreas Tempest Bikini goat!
5 · Yesterday at 12:24am

Goat Simulator disgusting
14 · Yesterday at 12:27am

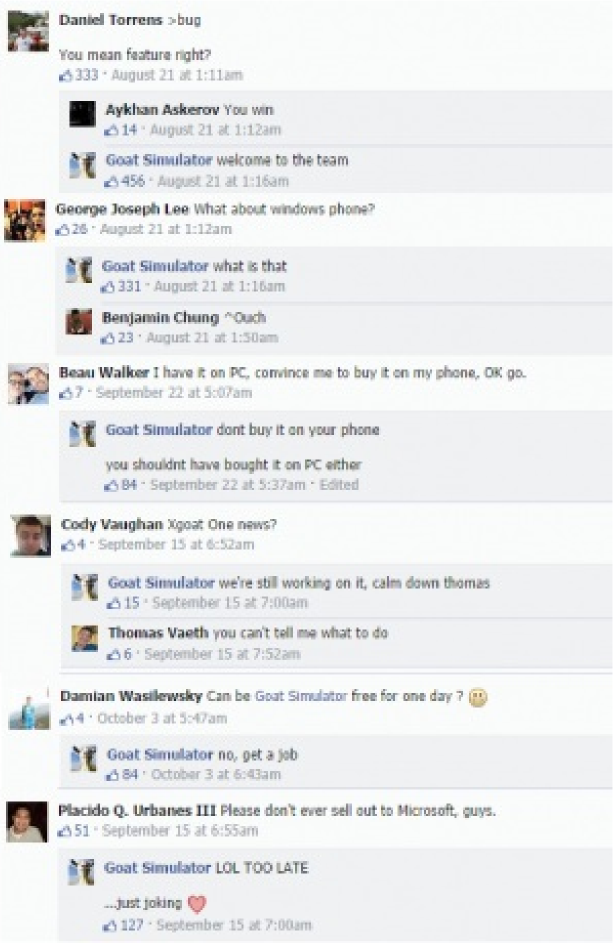
Stephen Birdwell >putting Goat Sim on a Mac platform
Fuck those hipster starbucks assholes. Macfags dont deserve this game.
9 · August 21 at 1:19am · Edited

Goat Simulator you must be fun at parties
144 · August 21 at 1:20am

Stephen Birdwell I'm not.
18 · August 21 at 1:21am

John Marvin We need an expansion for all versions. Goats should colonize Mars. Maybe even Uranus.
3 · September 9 at 2:19am

Goat Simulator oh you would like to go to Uranus wouldnt you
32 · September 9 at 2:43am



Daniel Torrens >bug

You mean feature right?
333 · August 21 at 1:11am

Aykhan Askerov You win
14 · August 21 at 1:12am

Goat Simulator welcome to the team
456 · August 21 at 1:16am

George Joseph Lee What about windows phone?
26 · August 21 at 1:12am

Goat Simulator what is that
331 · August 21 at 1:16am

Benjamin Chung ^Ouch
23 · August 21 at 1:50am

Beau Walker I have it on PC, convince me to buy it on my phone, OK go.
7 · September 22 at 5:07am

Goat Simulator dont buy it on your phone

you shouldnt have bought it on PC either
84 · September 22 at 5:37am · Edited

Cody Vaughan Xgoat One news?
4 · September 15 at 6:52am

Goat Simulator we're still working on it, calm down thomas
15 · September 15 at 7:00am

Thomas Vaeth you can't tell me what to do
6 · September 15 at 7:52am

Damian Wasilewsky Can be Goat Simulator free for one day ? 😕
4 · October 3 at 5:47am

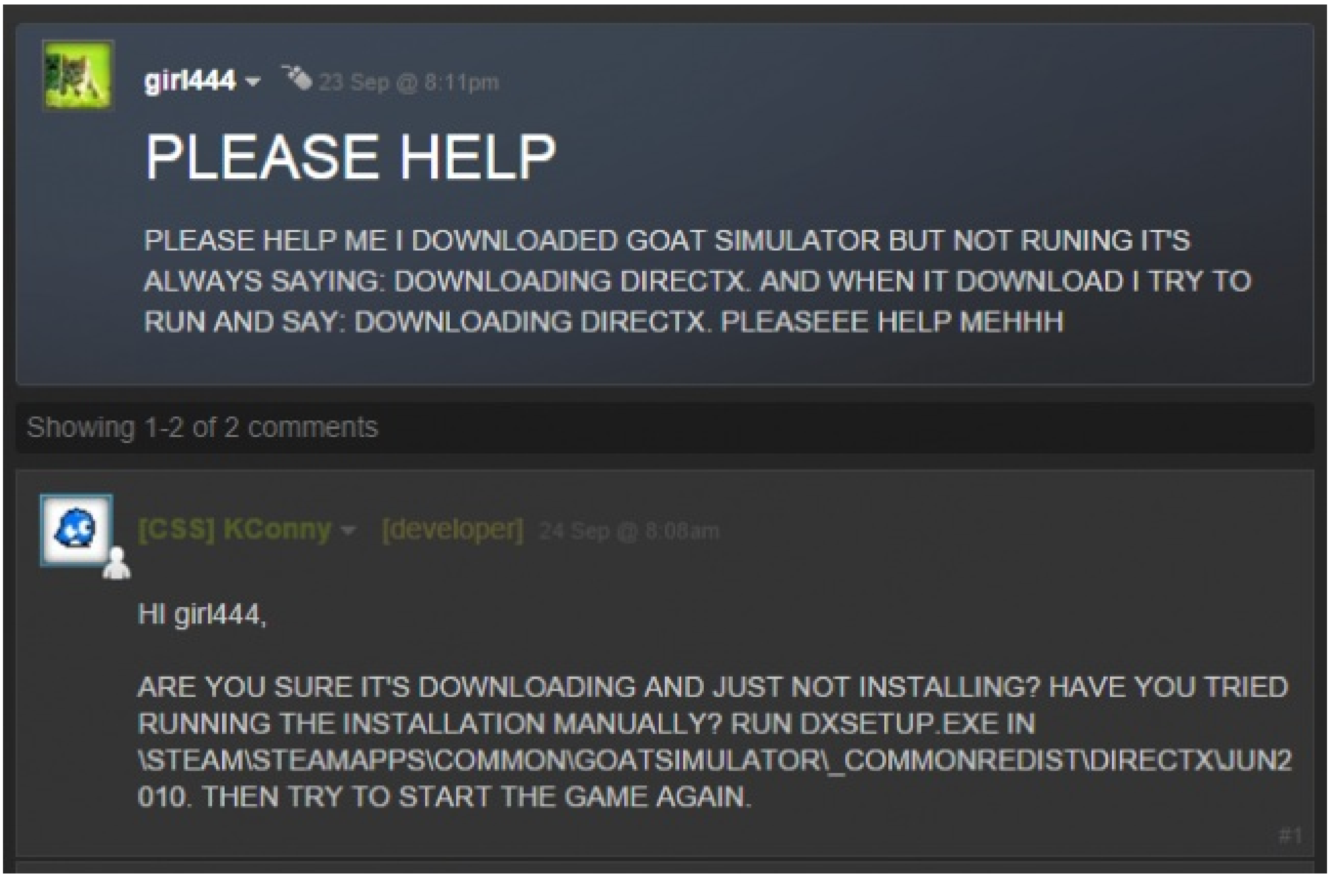
Goat Simulator no, get a job
84 · October 3 at 6:43am

Placido Q. Urbanes III Please don't ever sell out to Microsoft, guys.
51 · September 15 at 6:55am

Goat Simulator LOL TOO LATE

...just joking ❤
127 · September 15 at 7:00am

We use the same approach on other social media as well. Replying to people in a way that is funny and unexpected makes interaction just way more fun. Here’s an example of a bug report one of our programmers received in caps lock, to which he replied to in caps lock as well:



girl444 23 Sep @ 8:11pm

PLEASE HELP

PLEASE HELP ME I DOWNLOADED GOAT SIMULATOR BUT NOT RUNING IT'S ALWAYS SAYING: DOWNLOADING DIRECTX. AND WHEN IT DOWNLOAD I TRY TO RUN AND SAY: DOWNLOADING DIRECTX. PLEASEEE HELP MEHHH

Showing 1-2 of 2 comments

[CSS] KConny [developer] 24 Sep @ 8:08am

HI girl444,

ARE YOU SURE IT'S DOWNLOADING AND JUST NOT INSTALLING? HAVE YOU TRIED RUNNING THE INSTALLATION MANUALLY? RUN DXSETUP.EXE IN \STEAM\STEAMAPPS\COMMON\GOATSIMULATOR_COMMONREDIST\DIRECTX\JUN2010. THEN TRY TO START THE GAME AGAIN.

#1

Since our game is weird, satirical and humorous, we think that it makes sense for us to handle social media in the same in-character matter.

What went wrong

1. No planning or long-term think

Yeah, there were definitely setbacks to having no planning or long-term thinking. One of them was that we had to refactor a lot of programming once we decided that the game was to be launched on Steam. Earlier on, we kind of implemented things in fast-but-bad ways, so now when we implement new things today, we still have to keep in mind all the ugly coding we did earlier in the first couple of weeks of development. When you're making a joke prototype game where the main character is a goat, you don't exactly focus on thinking proactively. In hindsight, it's resulted in a lot of extra work. An example of this is the fact that we never planned on having multiple maps at the start of development, but when we chose to release the game on Steam and later, make a new map for our first 1.1 patch, we had to do a lot of copy-pasting of classes from one level to the other.

Another example is the fact that we didn't plan for Mac and Linux support on day one. When the development of Goat Simulator started, no one thought of how it's going to run on other operating systems since we didn't expect the game to become this huge. However, once the game was set for release, we had to scramble to finish a Mac and Linux version too. This took a lot of time and effort, and ended up being released several months after the PC version, which I think lost us a big chunk of sales. If you're making a game that you're planning on selling sooner or later, then I would strongly recommend having a Mac and Linux port on day one, if your budget affords it.

2. We should have focused on optimization since day one

Our biggest mistake with the optimization was that we didn't work with it from the very start. We never planned on selling the game on a mass market, so we thought that as long as it ran on our computers, we were fine. However, it became clear to us pretty fast that this was one of the game's biggest issues. A common misconception is that "if the graphics aren't super pretty, it's probably not a very demanding game". It's possible that a lot of people bought Goat Simulator and expected it to run on their computers since it doesn't look like a AAA-game. However, one of the main problem when it comes to performance is the fact that there are so many physics objects in Goat Simulator, which is very taxing on even the strongest computers. So just the core-gameplay of the game (crashing into physics objects) is very heavy on performance, especially since the physics need to be turned on all the time. If you accidentally knock a basketball across the map and it hits an NPC in the head and he falls down and knocks a table over, which has 10 other props on it that also need to simulate physics, it easily becomes too much for your computer. We get complaints sometimes that the game lags for no reason, and it's usually when people are unaware that they start a physics-chain reaction on a part of the map that they can't even see themselves.

Because of this, optimization is something we should have worried about since day one. Once we knew we would release the game on Steam, we did optimize it as much as we could, and we've kept optimizing the game ever since we launched. The game is way more optimized and smooth today than compared to launch day but sadly, first impressions persist.

3. We should have started working on the mobile version earlier

Once the requests started flooding in to make Goat Simulator into a real game on Steam and we started scrambling to finish the game until April 1st (because no matter how you think about it, April 1st is a hilarious release date and we just had to release it then). During these stressful months, the last thing on our minds was a mobile port. We had no idea if the game would sell well on Steam, as we were afraid that people would watch the game on Youtube and joke about it, but never buy it for real. We basically thought that we would release it on Steam first, and then if that goes well, we'll release it on mobile. However, only a couple of weeks after the first video of Goat Simulator went viral, there were already clones on the App Store and Google Play that had millions of downloads. Some people actually thought they had bought the real Goat Simulator, and were emailing our Coffee Stain support mail once their app crashed. We realized that it didn't matter if we wanted to make a mobile version of Goat Simulator or not, there were already mobile versions of the game out already, whether we liked it or not.

We eventually released the mobile version of Goat Simulator about 6 months after the PC version came out. However, in hindsight, we should have gone all-in and made a mobile version earlier rather than playing it safe and seeing whether the PC version was successful or not. We could have released the mobile version a lot earlier, which would be better for both our company, and the customers.

Mobile devices are becoming more and more powerful, and it's possible that we are soon reaching a point where performance isn't the main issue anymore when it comes to mobile ports, but controls and UI. Mobile ports are becoming more and more possible for developers, and for some games, porting the game to mobile might not be a crazier thought than porting it to consoles.

4. We should have focused more on Steam Workshop, and promoted it better.

Workshop support was one of the biggest features we had planned for the release of Goat Simulator. No matter how hard you work, people are always going to be able to play through content faster than you'll be able to create it, so it just made sense to let players create and share content with each other. It's worked okay so far, we haven't stumbled upon any major problems, but we've learned that you can't just put Steam Workshop in a game, ship it, and then forget about it and expect players to run everything by themselves.

For example, even though it says on a game's store page that it has Steam workshop, not all players will see that. Some might see it when they buy the game, but forget about it later. Some might see it, download some mods, then just play them once. There are tons of possible outcomes for players when it comes to how much they will engage in a mod-community. What we could have done better was to promote Steam workshop more without our own game, help funnel more players into actually trying out the mods that are available, and being better at encouraging content creators. It's become apparent to us that implementing Steam Workshop is just maybe one third of the work, the other two thirds should be continuous community management of the players making the mods, and updating the modding tools and making them easier to use for everybody. If you're looking to add Steam Workshop into your game, I'd definitely recommend to not just make time for the integration of Steam Workshop into your game, but also for all the programming maintenance and community management around it. Without it, you'll never reach up to Steam Workshop's full potential.

Today we feel like we have great content in our Steam Workshop, (some of it has even earned us media attention) but the submission of content could have went faster and easier on the content-creators' side if we had handled the community-side and maintenance better.

Conclusion

We've learned many very valuable lessons from developing Goat Simulator. The open and free form of production where everyone did whatever they wanted was extremely different from what other studios are doing, but that's also why it gave us an extremely different product from all the others. It might not be perfect for all game projects, but it worked for us. Releasing a game in such a short amount of time is very hard and tricky, but on the other hand, less time to develop a game means less time to mess things up.

Data box:

Developer: Coffee Stain Studios
Publisher: Coffee Stain Studios
Release date: April 1st, 2014

Platforms: PC, Mac, Linux, iOS and Android

Number of developers: ~10

Length of Development 10 weeks

Development tools: Unreal Engine 3, Notepad & MS paint

Budget: 10 pounds of organic, home grown hay

Number of legs: four

Number of horns: two

Tail: Yes

Sound: Bää