VISCA'S EARTH CONQUEST PATCH NOTES



"Prepare for my epic malware strats, lower life-form"

-The Alien that causes all the glitches in my game (not my fault, basically)

VERSION 1.1.2

-Fix of fix of fatal error caused by loading in credits controller and date formatting malfunctioning

Okay so it turns out GameMaker does NOT like when you try to convert an empty string to a real number, big shocker. Unfortunately

this glitch is a bit hard to test for, in all likelihood in the future from here on out I'm just gonna hand-jam the credits myself rather than relying on the CSV that patreon generates lol. Thank you all for being so patient with me, hopefully this one should prevent this issue from happening anymore.

-Infinite blocking glitch fix... 2!!

Crabs are still dead set on putting Visca in an infinite time loop, I've made it so they reset completely after being Blocked so hopefully they won't get stuck making hitboxes over and over this time. If you get this issue however, pausing and unpausing works as a temporary fix.

-Defense scaling tweaked

Damage felt a little bit unforgiving at bigger sizes, especially considering how much bigger your hurtbox becomes, I've tweaked the scaling ever so slightly to give you a little more defense so being hit isn't as punishing.

VERSION 1.1

-Potentially fixed fatal error caused by loading in credits controller and date formatting malfunctioning

The credits controller would load "strings" (which is programming speak for text, versus something like a number) from the spreadsheet file that contains the date of a patron's sign-up, however there is a possibility of these date formats not being properly translated when being unencrypted, thus resulting in a weird string/integer mix, which would cause a crash. I added a failsafe measure that should prevent this from causing a crash, however it may cause the credits to not be in proper order (I'll look into fixing that later, for now getting

rid of fatal errors is the main focus).
If you get another error showing a crash message mentioning oCreditsController, that means the failsafe didn't work, please make sure to just throw that in the feedback form if you have the time.

-Fixed Mecha spawning on the wrong side of the screen and subsequently getting stuck

The way the game controls the Mecha following you between rooms is that it'll check which direction you entered the room from, so if you entered from the right side of the screen, the Mecha would also spawn in on the right side of the screen. The initial spawn in (when he says "Stop right there alien scum") also uses this same code, however it's meant to ignore which direction you entered from and always enter on the left (so that your first impression of him wouldn't be seeing the EDF text flipped backwards lol). I fucked up making sure that didn't happen basically. Whoops!!

-Fixed resolution scaling problem with ultra wide monitors (Which would subsequently result in a fatal error when attempting to adjust it)

Looking back at my window scaling code I'm kind of shocked at what a dumb oversight I made. Essentially, on your initial boot-up, the game will default to windowed mode, and scale the window up until its width fits within your monitor.
Only the width. So very wide monitors will cause the game to scale up so much that the top and bottom of the screen can't be seen, and it'll scale it far beyond any resolution settings I had in place for the scaling, which would result in essentially the game trying to set the window scale variable to a position that is "out of bounds". I've since made sure to check both width AND height match the monitor, so neither should go beyond the limitations of the screen now.
IMPORTANT: If the resolution is still overly big when you boot up

the game, you may need to delete your settings file. Open your file browser and type %localappdata% into the address bar, look for the folder named “ViscasEarthConquest” and delete “settings.sav”

-Fixed player getting stuck in sand castle room after destroying crabs.

I’ll be honest with you. I don't know how I didn’t even *consider* that the player might try destroying the crabs in the sandcastle, resulting in scaling Visca up. I guess I just thought they were so cute I never even tried during playtesting (and neither did my playtesting friends lmao). Anyway doing that won’t get you stuck in the sandcastle now.

-Fixed “infinite blocking glitch”

When blocking 2 crabs at the same time, the player would get stuck in an infinite loop of blocking continuously spawning hitboxes, this was because when you block it causes “hitstop”, meaning that the game pauses for a short while to emphasize what just happened. This also pauses the animation of everything in the room, meaning that the crab (or any meleeing enemy actually) would repeatedly be on the same frame, making the hitbox over and over. When two are in the same place, these two hitboxes would constantly ''trade places'' each frame, resulting in an infinite blocking loop. Just adding a simple check has fixed this issue.

-Fixed multiple memory leaks which I’m *guessing* has been the cause of “silent crashes” for people

If you don’t care about the technical details, I’ve fixed a severe performance problem which should result in a smoother experience and hopefully will do away with sudden abrupt crashes that leave no error message. If they keep happening please let me know in the feedback form.

So games made in GameMaker have an automatic process called “Garbage Collection”, where it will remove unneeded items from memory (such as a function that is attached to an object that was destroyed during gameplay). Most data is automatically garbage collected for you, however something called Data Structures are not. These will persist throughout the entire game until the programmer explicitly puts in a command to delete it. Having a DS sitting in memory on its own isn’t an issue, unless you’re continuously creating DS’s and not removing any of them. That’s when you get a “memory leak”. Which just means your game is constantly requiring more and more RAM the longer it runs.
The “bones” which 2d rigs use can be manipulated at runtime by grabbing their information, storing it into a DS, and referencing that information to make relative changes to that data. This is how NPCs will turn their head to look at you, or point a gun at you, or Visca’s boobs will slowly scale up as she grows in size.
I was pretty diligent about destroying the DS’s after using them, but not diligent enough, even one tiny mistake will slowly rot away the game’s performance because this DS is being created EVERY FRAME. 60 times a second, which is a LOT especially over an hour of gameplay. Wanna know how I fixed the issue?

```
//Update scaling
for (var i = 0; i < 5; i++){
```

```
//Update scaling
for (var i = 0; i < 6; i++){
```

I turned a 5 into a 6. Memory leak gone!
I’m an idiot lmao.

KNOWN ISSUES

These are issues that I am aware of that are not top-priority fixes so they will be rolled up into a future patch.