

# HBGAMES.org

The E-zine!

HBGAMES.org's Official E-zine!

Issue #26 November 2020



## PERCEIVING SHAPES



**GAME OF THE YEAR EDITION**

**THIS AND SEVERAL OTHER YEARS**



Welcome to HBGames.org – the eZine!

Our last issue was some six years ago, please forgive us. In this issue, Byron discusses his Observer system in what I'm going to call Nerdy Programming Talk, Remilia discusses characters, and Ellie talks about crafting systems and their effect on a game's economy, as well as looking into whether merchandise is a viable means of funding a game making project.

You'll notice the icon above literally all over this issue. This is intentional, it's our new ident, and the new logo that you'll see in our discord and on the website. I want this little guy to become synonymous with HBGames and all that we do here, and wearing this badge should be a thing of honour. Or something like that.

# CONTENTS

## 2 Who are You?

A brief history of HBGames for those wondering just who the hell we are.

## 3 New Look

HBGames has had a new look - after a decade!

## 4 Game of the Year

Throughout this issue you'll find out the winners of years 2015 and onwards of HBG Game of the Year.

## 5 Crafting an Economy

Ellie looks at crafting systems and the effect they can have on a game.

## 8 Discord

We introduce Discord, which is the new hub for HBGames

## 11 The Merch Model

Ellie looks at merchandise and whether custom merch is a viable model to fund an indie game.

## 14 Observer System

Byron introduces his observer system in an in-depth programming article.

## 18 Remi's

Remilia takes an in-depth look at character design.

## 20 GotY 2020

Finally announcing the winner of HBGames.org's Game of the Year 2020!

## 22,16 Featured Artwork

A selection of art from around the forums.



# Who are You?

In the year 2004, when RPG Maker was in its infancy, the only way to obtain RPG Maker in English was through illegal translations, however these were very easy to obtain. Widely distributed in the English speaking world, there became a subculture surrounding homemade RPG games, at websites such as Gaming Ground Zero, Crankeye or Gaming World.

In 2004 the fourth instalment in the franchise was released – RPG Maker XP. It was swiftly fan-translated, again illegally, and again widely distributed amongst the English speaking world. One of the first fansites to crop up especially for this new innovative engine was RMX.net.

RMX.net grew to be the single most active RPG Maker website there has ever been, amassing some 324,000 posts in the little-over-a-year that it was alive. Eventually the creator of the website (Lord Gradient) disappeared, and around the same time, RPG Maker XP was officially released in English, largely owing to the popularity of RMX.net and fans who wanted a legitimate way of buying into the franchise.

To replace the dying RMX.net, which was facing hosting issues eventually leading to the complete loss of all content and its domain name, RMX.org was created as a replacement. Led by Ccoa and Erk, this became a friendly, massive community that became the backbone of the English RPG Maker world. The best scripters in the world found a home here, from SephirothSpawn to the creators of Netplay, the first MMORPG scripts for RPG Maker.

In this time RMX.org gained a reputation for pushing the boundaries of RPG Maker with innovative scripts and external tools. Pixel artists such as Despain, Venetia and Perihelion rushed to the site and found a comfortable home, eventually

becoming the site's administrators in Erk and Ccoa's wake. A massively participated pixel art contest cemented RMX.org as the place for high quality pixel art and a do-it-yourself attitude to game making.

Eventually Despain grew tired of the site and turned into a troll, leading to a lot of members leaving, until the other admins stepped in and overthrew his lead, taking the reigns. Venetia led this new era, pushing towards a site that was inclusive of other game making engines. Eventually the name HBGames – Home Brewed Games – was settled on and a new era began.

Unfortunately Despain had ultimate control over the domain and took it for his own purposes, eventually dropping it altogether, and it is now lost to the ether of the Internet never to return. In the confusion many thought RMX.org had died, but it lives on in HBGames.org, and has done now for over a decade.

Ellie took the reigns in 2010 and alongside Xilef manages and maintains what is now a smaller, less active community. Our history of being progressive both in our extending RPG Maker and in the high quality of our pixel art continues, however, and those who have called our little website home have gone on to work for such high profile games companies as Ubisoft and Mojang.

Where the future lies depends very much on you guys, but with the advent of Discord and Steam, there are a whole new generation of RPG Makers ripe and ready to be converted to the HBGames way of life!



# New Look



Last time HBGAMES.org had a redesign was almost exactly ten years ago, in February 2010. Back then times were very different. Gradients were king, and web 2.0 was not only very new, but nobody really knew how to implement it. Sure we toyed with a Twitter and a Facebook feed but it wasn't until Discord came along that we really embraced the New World.

I realised the other day how dated the site is, and how much is physically broken, due to poor coding from myself. It was time for a change, however light, and so as a start we have had a bit of a redesign. It's not actually that big of an update but it has completely changed the look of the place.

For one we have a new logo – sorry Venetia, to get rid of the one we have had for the past decade, it served us well – one which fits pride of place as a Discord channel badge, as a phone app icon, and as a favico.

I wanted a logo that when seen screams “that's HBGAMES!”. Something you can see from the corner of your eye and know that's us.

Further, we've revived our decades old slogan which dates back to the days of RMX.net: Envision, Create, Share. It really speaks as for what our community is all about.

You'll notice much continuity. The design of the forums themselves is unchanged and is unlikely to ever change. They are still those designed by Shadow way back in 2008, themselves based on the original theme for RMX.org in 2006, harking back to the original design of RMX.net by Lord Gradient.





# GAME OF THE YEAR

In my continuing efforts to update the HBGames website and bring it into the 21st century, I found that we haven't actually announced a Game of the Year since 2014. We've had many great games in that time, some of which have gone largely unnoticed – they include, in my opinion, some of the best RPG maker games ever made. So, this being a democracy, I have taken it upon myself to choose a game from each year, up to 2020 (any games released now will go into next year's pool).

GotY games win pride of place on the HBGames homepage, and also get a neat badge which they are allowed to show anywhere and everywhere they wish to (though whipping it out on the tube is frowned upon in most societies).

Next year we can have a proper vote, if we're active enough. Previous years' winners are all shown on the HBGames homepage, though you'll have a hard time finding games such as Master of the Wind nowadays. These go all the way back to the first year of RMX.net all that time ago.

If anybody wishes to do a thorough review of any of these games we can feature them in future eZines. For now I have focused on this year's winning title, which we will come to shortly. For now here are the last five years' winners.

Special mention to one particular game maker, who is having an exciting year this year. We'll come to that shortly!

We'll be feeding out winners throughout this issue.

Born Under  
the Rain 2015



# Crafting an Economy

Modern MMORPGs are built on complex crafting systems. Popularised by games such as RuneScape they often provide the backbone to a game's economy, through use of a mixture of gathering skills and processing skills. Often these skills have arrived out of necessity rather than design however, due to the evolving nature of an MMORPG's economy, which leads us to a complicated design when applying such ideas to either a typical offline RPG or to a "quiet" MMORPG, which is what most of our games will be in the end.

Crafting in RuneScape began as a simple, logical way of turning items into other items. Copper and tin makes bronze, which is then formed into weapons. This provides a multi-level economy already: some players spend their time fighting monsters, which gives a value to weaponry. This makes making those weapons a viable career, which gives value to the raw materials that create those weapons, providing us with jobs for new players: early mining.



The issue comes when any of this is imbalanced. This could be by devaluing the cost of those weapons through quest or monster drops. For example, Rune weaponry became far more common as a drop from Slayer than from actual manufacturing, massively devaluing the end items, which ruined the economy for those raw materials in the first place.



RuneScape's method of fixing this imbalance has been to keep adding further content using those raw materials. The first was Alchemy, a way of turning items into a fixed amount of gold. This essentially gave these items a base limit where they would not drop below, because alchemists are guaranteed a certain value for their items. This also made the RuneCrafting skill more worthwhile as those runes made were used in alchemy - nature runes became a valuable commodity.

Every new update to the game has an effect on this economy. The issue comes when applying any of this to an offline game.

In many instances there is simply no need for the huge levels of skills that a game such as RuneScape has, because there is no damaged economy to try and fix. A difference to this is where levels progress through a game, which gives certain items and weapons a value in itself. For example, if the player really wants a tier 90 weapon, that weapon now has more value than a tier 80 item.

If we don't go down the routes of huge multi-level skills creating our crafting systems, is there a alternative?

In my mind there are two types of crafting: scripted and guesswork. Scripted crafting requires a list of ingredients that create certain outcomes. Copper and tin make bronze and always make bronze.



Guesswork is more complicated, and takes me to games such as Little Alchemist. In these games most items are combinable in one form or another: wind plus energy makes tornado, fire plus water makes steam, etc. The important part of this system is that almost every combination should work in some way or at least have some kind of outcome, be it a cute message or a warning. The player is in this instance rewarded by their guesswork and intuition - instead of being told what



creates bronze they could have a stab at guessing at what mixing copper and tin would do.

The guesswork crafting system almost requires that the content of levels is turned on its head. Perhaps instead of being a requirement, levels are instead a flashy show of what you have done so far. This could still apply to an MMORPG, especially a relatively "quiet" one.

What impact does this have on the economy? Quite a profound one, as with so many outcomes, the raw materials have different or negligible worth. It all boils down to what is important in your game.

The main thing is that players need to have fun. Grinding is only fun if it has a positive outcome. Few players in an offline RPG or quiet MMORPG are going to be interested in actually building the economy. This does mean that quiet MMORPGs need a replacement for trading - usually in the form of an Exchange or Auction House system.

In all it is up to you what is important in your game. But consider the impact on the economy, or the lack thereof, in how your crafting system is implemented.



**GAME OF THE YEAR**

# **Karma Flow**

**2016**





# DISCORD

As part of our new look and refocus, you'll notice a lot of talk about our Discord server.

What is Discord, and why are we using it?

In the past we have used IRC as a means of having live chat between members of the site. This was always a niche and was only used by a small portion of the wider RPG Maker community, but nonetheless we had fun and it was a great way of keeping in touch in ways a forum sometimes cannot provide – the closest the forum comes to this is the Post What's On Your Mind thread (and similar dump threads around the place).

Discord is essentially the modern era's version of IRC. The main differences are moot for your average user – basically Discord is centrally hosted, making it more of a social network, and with more interplay between communities.

For us the main difference is how media is displayed in a Discord server. There is a heavy focus on emojis (yes, we've finally succumbed to the terminology) and images that are posted can be seen as part of the chat. This makes quick critique of sketches and loose screenshots easy.





What we have done however is to tie the Discord into the forums. At the moment this is by means of a bot called MonitoRSS (you'll notice him posting about the place). MonitoRSS trawls the forums for specific types of content and dumps them in the Discord chat. Screenshots appear in the #showcase channel, artwork in the #artworkandcreativity channel, and then a general feed of new posts appears (for now – while we're this quiet – but this may have to be limited in future if things pick up!).

There are many other channels on the Discord, including a General channel and a Sergeant channel (don't ask). You'll also find channels for development of this eZine, for self promotion, showcasing works, and discussing programming and music making, amongst other things. There are also voice chat channels if you are that way inclined.

Discord has quickly become the industry standard for this type of community and it is where the majority of RPG Maker activity outside of RPG Maker Web and Steam occurs. Steam is used for hosting and distributing games, RM Web is used for support in an official capacity, and Discord is used for chat elsewhere.

Of course, our forums still play a role in the community, but certainly not as large a one as in previous ages. Gone are the days of RMXP.net where we were literally the centre of the RPG Maker world.

For the moment the only interplay back to the forums is that we have in the Rec Room a Discord Quotes thread, a spiritual continuation of our old IRC Quotes thread, for anything humorous that crops up from time to time. In future there may be more we can do. In particular, in my game making endeavours I have discovered ways of actually using game accounts that are Discord accounts – you can log into my game using Discord. Where this leads us is questionable at the moment, but who knows, in the long term perhaps a thorough detailed API plugin can be created, safely and securely, allowing leaderboards and high scores for games.

If you want to quickly spam a link to your latest game, or your favourite game, you can use the Mindless Self Promotion channel. This is also the place to link to your personal discords should you so wish.

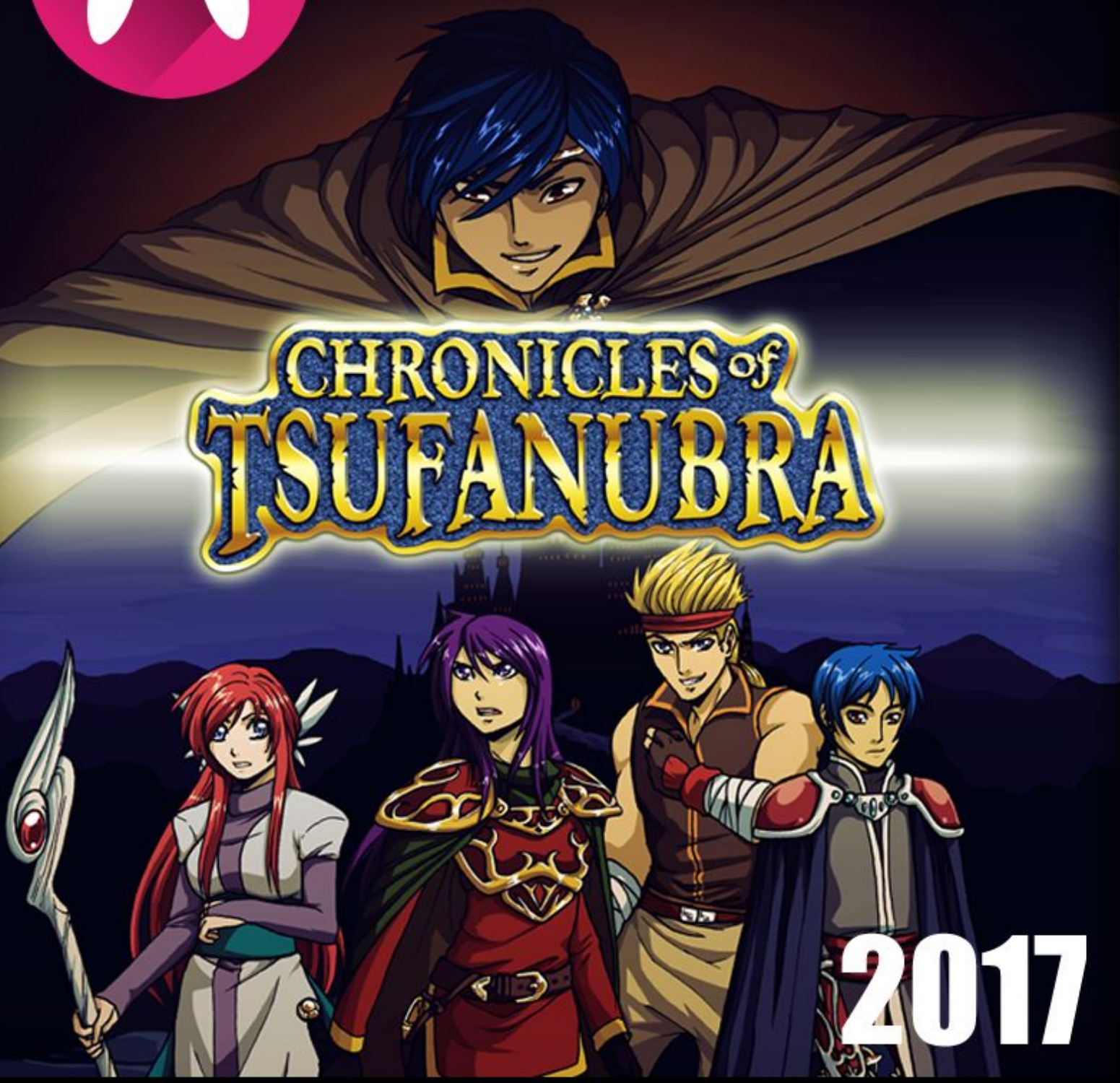
At present the only rule on the Discord is: Don't Be A Dick. If this is ambiguous, you are being a dick. Please don't be the one to make us have to introduce more rules.

While we don't presently have any "partner servers" we are open to suggestions.

# GAME OF THE YEAR



## CHRONICLES of TSUFANUBRA



2017



# The Merch Model

Gone are the days of the complexities of Cafepress, and in are the new days of Redbubble. How lucrative are these methods of creating personalised merchandise for your game?

Right off the bat the idea of creating t-shirts, notebooks and other memorabilia seems like a great way of funding a game, especially one that you want to remain free but still need funds to finance, be it for assets and resources or for promotional costs.

Cafepress provided us with the first concept of this way back in the early 2000s. The high cost and the complexity kept people away however.



## REDBUBBLE

I have had some success creating items on Redbubble. My sister loves it, for the high quality personalised merchandise she can create for herself. As a marketplace however things get a little difficult. For every t-shirt I sell (for a private company) Redbubble charges an extortionate manufacturing fee, leaving me with about 10% on any garment made. I would have to take taxes off this were I commercially successful enough. Though the cost of items initially seems low in comparison to Cafepress of old, in order to actually make money through the system your goods must be marked up to near enough their prices. In truth the company making profit is Redbubble.



It seems that Redbubble is a lot like Lulu. Lulu is a website where you can self-publish, making your own book and getting it out there in the world. Amazon have a similar system. But the only people that will really buy these books are yourself and your inner circle. Far too often self-publishing means creating a product that only you would want to buy in the first place, but this is their business model. They know that if you personally design a t-shirt one person is going to buy it: you.

The other way they make their money is in the more shady side of things, and is the reason you are unlikely to sell anything through their actual marketplace. I am absolutely not suggesting that this is intended, of course, but you can see the thousands making money by creating merchandise for products and franchises they do not own. In particular, football clubs.

I was creating products and gifts for my local city, which were selling well, until I had a nasty e-mail in my inbox. My local football club had considered any item featuring the name of my city to be their IP, and removed it all from sale, giving me a stark warning. However, search for some premiership football clubs on the site and you will see why they feel they have to do this. Redbubble is absolutely swamped with fake produce.

The companies doing this do not even have to charge that much as they know they will make the volumes they need to profit: that leaves your little old game with very little room for profit, as you cannot match their prices. Anyone looking at t-shirts for your tiny RPG will see products for Final Fantasy or Zelda for a lower price buying those instead, rendering you penniless.

# FINAL FANTASY VII

ファイナルファンタジー VII

In short, these do-it-yourself-ish marketplaces do not work for what I require them to and they are not an answer to replace donations and actual physical payment for our games.



I guess there is an answer however, and a way that we can use Redbubble to our advantage, and that is one of marketing and promotion. Create a t-shirt for your game as something you would want to wear, and feature your website address, your twitter handle or your hashtag prominently, and use this as a way of drawing people to your franchise. You might not make any money in the process but at least you can look good along the way.





**GAME OF THE YEAR**



*Perseverance*

FULL CLEARANCE



**2018**



# Observer System

by Byron

## 1. Intro to Observers

Observers are incredibly well documented programming patterns. Seriously, if you want a better explanation just do a quick google. With that being said, what I'm writing about is specifically the implementation I use in almost all of my projects. Firstly I'll break down the theory of the observer system. Secondly I'll move onto implementation and code samples within unity using c#. Lastly I'll discuss some considerations with the system including some of its flaws.

Without further ado, let's discuss some theory:

## 2. The Theory

The basic principle of an observer is quite simple. Within the object you wish to observe (I call this the subject), store a collection of objects (I call this the observer). When the subject makes a change, notify the observers (I call these events, this is technically incorrect terminology though). Observers only care about events that they subscribe to.



In my implementation I use a couple of techniques to make things easier. This first is using C#'s handy Interfaces. I typically have Observers as interfaces. This serves a dual

purpose. It means that objects can be both an observer and a subject which helps things run smoother. Along side this, each observer deals with its events uniquely. Having it as an interface forces the programmer to implement the interfaces functions.

The second technique I use pattern matching. Quite simply I store my events as classes, all deriving from an "EventBase" class.

This also serves a dual purpose. It means you can check which events an observer subscribes to via casting the passed in event. It also means that events can store data, which is useful for various reasons. I think it's time to see an implementation.

## 3. Implementations

In this implementation I'll show the observer pattern in quite a simple points system. Here's how it will work: The "PointManager" will observe a player. When a player collides with a coin (or whatever we have representing points) the player sends a notification with a "CurrencyEvent" to its manager. A quick note before we begin coding. I typically use Lists as my collection of choice with this system. Here I used an array due to better searching.

I figured searching is more likely than adding/removing at runtime.

Here's the code:

```

public interface ObserverBase
{
    // Take a notification
    void OnNotify(EventBase eb);
}

public class SubjectBase : MonoBehaviour
{
    protected ObserverBase[] observers;

    protected Notify(EventBase eb)
    {
        // Send an event to all observers
        // (this is where the magic happens)
        for(int i = 0; i < observers.Length; i++)
            observers[i].OnNotify(eb);
    }

    protected void AddObserver(ObserverBase subject)
    {
        // Avoid annoying initialisation errors.
        if(observers[0] == null)
        {
            observers[0] = subject;
            return;
        }
        // Increase the size of the array and
        // assign the latest value to the subject
        ResizeArray(observers.Length+1);
        observers[observers.Length-1] = subject;
    }

    protected void RemoveObserver(ObserverBase subject)
    {
        int index = -1;
        // Find the object you wish to remove
        for(int i = 0; i < observers.Length; i++)
        {
            if(observers[i] == subject)
            {
                // Set it to null and store the index for swapping
                observers[i] = null;
                index = i;
            }
        }
        // Check if an object is found
        if(index > -1)
        {
            // Swap if the object wasn't the last in the array
            if(index < observers.Length)
            {
                ObserverBase temp = observers[observers.Length];
                observers[index] = temp;
            }
            // Resize the array
            ResizeArray(observers.Length-1);
        }
    }

    protected void ResizeArray(int newLength)
    {
        // Temporarily store the array
        ObserverBase[] temp = observers;
        // Make the array bigger
        observers = new ObserverBase[newLength];
        // Read in the previous values
        for(int i = 0; i < temp.Length; i++)
        {
            observers[i] = temp[i];
        }
    }
}

public abstract class EventBase {}

```

### 3.2. Implemented Classes

```

public class CurrencyEvent : EventBase
{
    public CurrencyEvent(int newValue)
    {
        // Construct this event to store relevant data
        currencyValue = newValue;
    }
    public int currencyValue;
}

public class Coin : MonoBehaviour
{
    // Collectable value
    public int coinValue;
}

public class Player : SubjectBase
{
    public void Start()
    {
        // Initialise the array and find any observers
        observers = new ObserverBase[1];
        AddObserver(FindObjectOfType<CurrencyManager>());
    }
    public void OnCollisionEnter(Collider other)
    {
        // Make sure you're colliding with a coin and send a currency event
        Coin collidedObject = other.gameObject.GetComponent<Coin>();
        if(collidedObject != null)
        {
            CurrencyEvent ce = new CurrencyEvent(collidedObject.coinValue);
            Notify(ce);
            collidedObject.gameObject.SetActive(false);
        }
    }
}

public class CurrencyManager : MonoBehaviour, ObserverBase
{
    // Initialise currency value
    int totalCurrency = 0;
    https://www.overleaf.com/project/5fa976a30b3f882bc925570e
    void OnNotify(EventBase eb)
    {
        // If it receives a currency event, add it to the total currency
        switch(eb)
        {
            case CurrencyEvent ce:
                totalCurrency += ce.currencyValue;
                break;
        }
    }
}

```

### 4. Considerations

There's a few things I'm aware aren't amazing with this system. For instance, a large number of observers on one object could receive a notification despite only one object being subscribed. This would mean a search through an array each time you wish to notify.



Another consideration with this system is in the actual collection.

I would say this just comes down to personal taste as well as being situational. If you intend on adding and removing observers often, use a list. If you're intending on searching often use an array or a hashset.

Obtaining objects using unities "FindObjectOfType" or "FindObjectsOfType" are heavily inefficient. I highly recommend you only do this on the start of the scene or create your own tag system.

Storing the events as classes could also be problematic. There's no need to store things as a class if it's just holding data. Yet the obvious advantage come with pattern matching. Enabling you to simply switch incoming event. I'm sure there's plenty more to consider with this system, and a billion more improvements I could make, however I've found it quite useful in decoupling my code.



*Smoke Signal by CoyoteCraft*

# ARTWORK

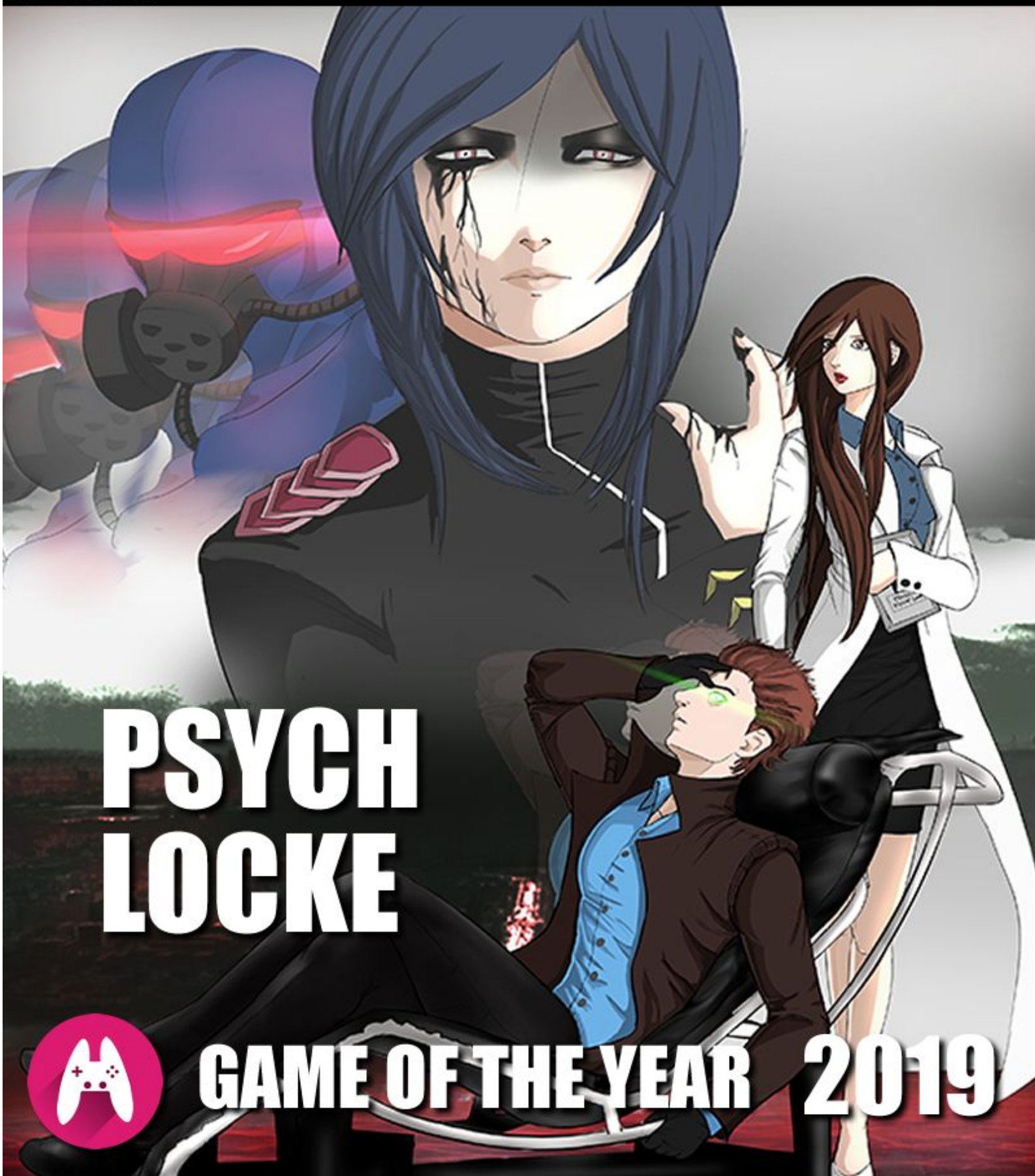


*Fantasia has been Forged by Remilia*

## ADVANCED RPG Maker



You'll find more coding talk and programming examples in our old sister mag, Advanced RPG Maker Magazine, which can be found at [aRPGMaker.com](http://aRPGMaker.com) (it's just our RPG Maker subsection).



# PSYCH LOCKE

GAME OF THE YEAR 2019





# Remi's Recipes for Relatable Characters!

## by Remilia

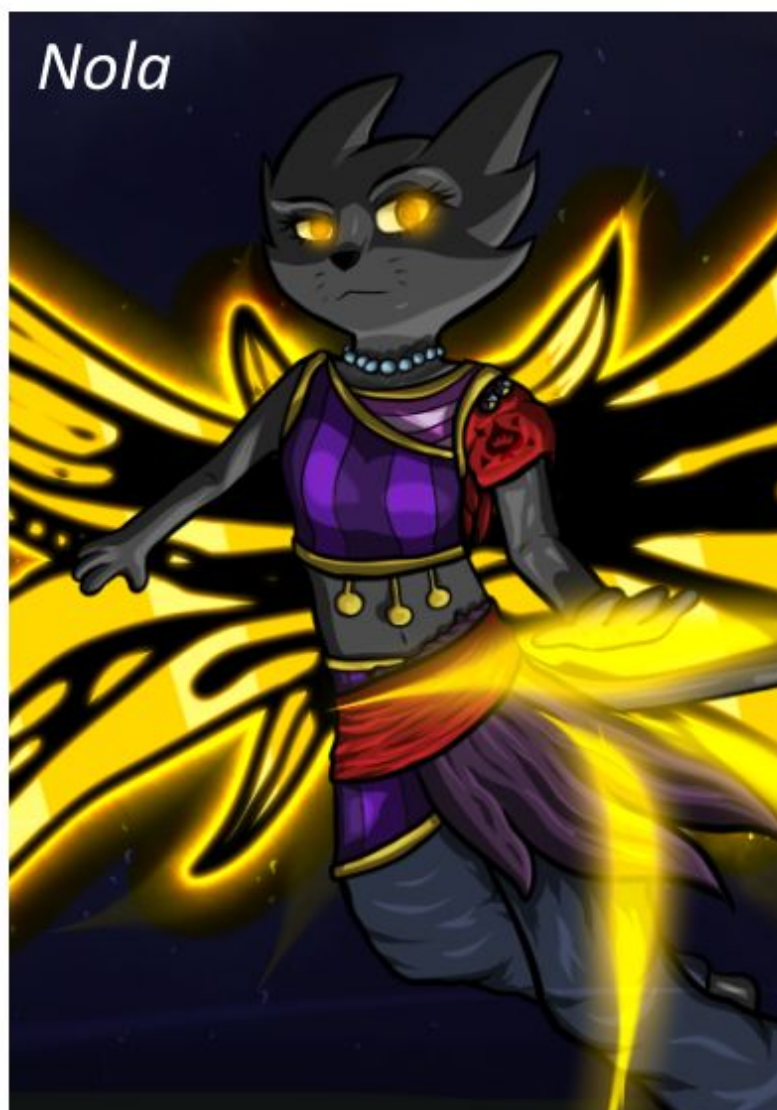
Making lovable memorable characters can be quite a struggle and is a task of much consideration, even down to cosmetics! But what all characters need if they are going to be on screen a lot is relatability, to make them feel human like us! Even if they are a superpowered goddess, a shred of humanity must exist for her to seem real.

And you may wonder... "well, if they are a goddess, and therefore brush the line of mythos and incorporeality... would it not be better they felt UNREAL?" and yes, that is- if you just want her to sound like a stoic 1 dimensional piece of cardboard.

The point is if you want your players to CARE about a character, then you need to make them have a value that is human. For this i personally just pick one trait from myself and build onto that from there. Also consider where your characters grew up, what they may have seen that would define them as a person, what tastes they may have acquired- all of these things are helpful, then even small details are likely to come naturally, for instance, typing/speech quirks, Kloe, for instance, sometimes refers to herself in the third person, statements such as "Kloe knows, as Kloe saw!" can give characters a bit of seperation from the mold of english and its formulaic same-sounding tendencies. Think of characters like Yoda, his sppedch gimmick was simple but it was consistent and gave him a sense of memorability. Know this, do you?

Character design is a whole other ballpark and each person has different ideas on it. Personally I use colors to exemplify certain behaviours, red is passion, orange is vigilance, yellow is justice, green is courtesy or kindness, pale blue is patience, blue is independance, purple is pride, and pink is bravery. The darker the color, the more ngative the trait, the lighter the color, the more positive. Black is pure negativity, white is pure positivity.

This isn't hard and fast, even i don't always stick to it, but there are examples i found long after I'd wrote out this system a few years back and it basically crept into what i had defined. Pep for instance, is INCREDIBLY passionate about everything he does, his green coat symbolizes his ability to be kind and show humility when he'd rather keep friends than make enemies, Nola, meanwhile, is largely black and yellow, vengeance-guided justice- which is her whole shtick in Perseverance, even her friends were a bit scared of her. She does have a sweeter side to her, though, emphasized by the amount of different colors on her outfit.





Pep



^ Kloe - credit Xiie

v Sven - credit CoyoteCraft



Sven is a paler red, and he has a very large green coat, and Sven is a very kind a courtueos character, so that checks out too. meanwhile, Esperia, who has mostly dark colors, is both incredibly independant and patient, waiting decades to bring about a plan only to eventually realize it wasnt worth it and finding another way, which took over a century to achieve. She doesn't always work alone, sometimes she has her blood maidens along, but she does through most of PFC and she is remarkably good at it.

But again, that's character design and you may do it differently. I'm here to tell you of how to make a relatable character!

Characters will always inevitably be a reflection of the self in some way, or should be. I choose a trait i am proud of and load that into a character and work out details, how long can this take/ well, ages! A good character isn't born in a single night, it can take weeks to really flesh them out, but once you have, you'll know! if not, here's a good test, if you can think out a conversation with them and start giggling or being moved by their antics, then, yes- you're at the right place!

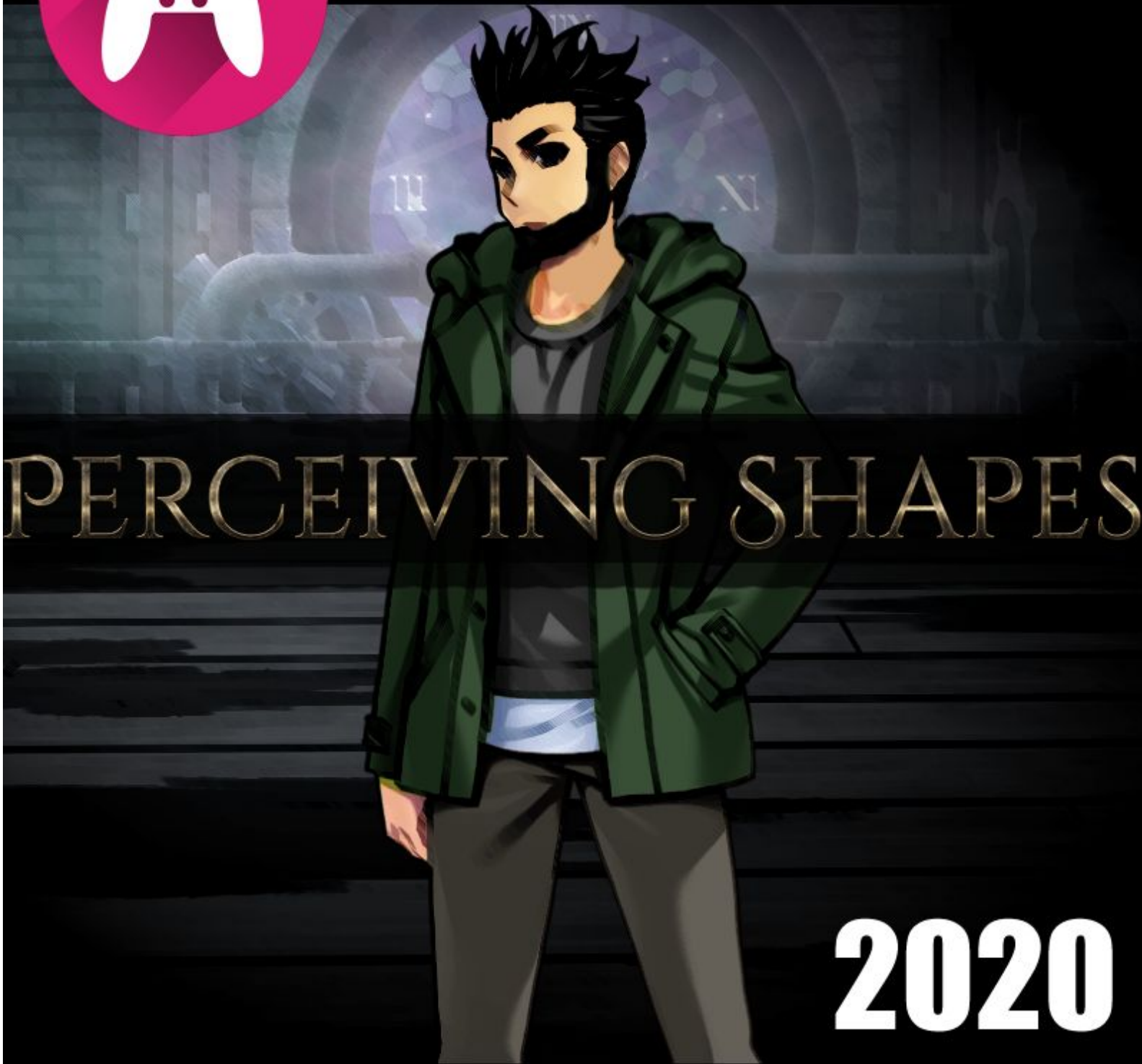
Now you may not do the trait thing i do, maybe something like drawing from a single past experience or a single object of interest for you and turning that into a character will be your

shtick? This is what i had to do when traits began becoming too blurred, for instance, The Queen of Shadows isn't based of a trait, i based her off of patchouli from touhou, but made her a bit more playful and made her design radically different. She's grown more into her own sort of being now, a character I enjoy writing so much I wrote an 88 page book starring her as the main character.

Once you know your characters, then it's time to have em introduce each other, see how their personalities conflict, what has happened in the past and how that would help them predefine people. Think about it, because at the end of the day you're not making a character, you're making a person, a living breathing flawed being who people will be expected to interact with in favor of people in the real world. If your characters don't feel approachable, your game may suffer from the same issue.



# GAME OF THE YEAR



# PERCEIVING SHAPES

2020



# GAME OF THE YEAR



Oh? A patient? At this hour?

This year's winner comes from Starmage, the author of two games released this year – and a whopping eleven in total. A voice acted horror, *Perceiving Shapes* was an entry to Itch Io's Hawktober Game Jam, a Cosmic Horror competition, and featured fourth place on people's choice.

Starmage is an indie developer from the Philippines whose influence comes from the *Final Fantasy* and *Grandia* series'. Beginning with *RPG Maker 2003*, with titles including *BlueSkies*, he has a particular style that flows throughout his works.

We look forward to seeing what 2021 will bring!





# FEATURED ARTIST: REMILIA



# AND FINALLY

Hopefully you've enjoyed this issue, which was put together quickly in a day the same as all the others. Whether you're new to HBGames or an old face, thank you for reading.

This year has been a hell of a year but it has led to an amazing amount of time for game development. Let's see what fruits that labour brings in the coming year – we're expecting a lot of new releases.

This is the issue where random columns and bits of graphic aren't thrown in to save space because people actually submitted articles. Wooh!



Back issues of varying quality are as always found on HBGames.org. Our sister mag, which is more indepth and professional, can be found at aRPGMaker.com, although that hasn't been updated in a long while.

A new RPG Maker has been released since we last spoke. Unfortunately RPG Maker MV was so good that we don't even know what the new one is called. I'm sure it's great.

## Did you know?

**HBGames is entirely non-profit!**

Not through want of trying, we just don't get any. Joking aside, we're a completely not-for-profit website and any money from donations is invested back into you guys (although mainly it goes towards hosting, for which Ellie pays entirely).

We've just grabbed images from around the forums and also from Google searches. If you aren't happy with your image being used please let us know and we'll remove them. We know, it's all a bit sloppy.

And by we I mean Ellie. The royal we.

