BeWolf Studio

31 tips for the launch of your videogame

By Néstor García

Introduction

Hello! I am Néstor, Redd's Runaway developer, a procedural Run & Gun videogame inspired by the tale of Little Red Riding Hood, combining a frenetic gameplay style with an aesthetic reminiscent of the Halloween and Day of the Dead traditions, in a short experience aiming for a great replayability.

During the development of my first commercial title, I faced a variety of interesting problems, from the concept I wanted to develop or the ideal game engine, to improving the performance of the title and publishing the finished product on Steam and itch.io.

This guide comprises a list of 31 tips to help you during your development process, to make the launch of your project as easy as possible, with a perspective of the possible problems you may have to face in the way.

Through the development of any video game we will encounter a variety of difficulties, which, if handled properly, can be positive for the health of our title, in addition to freeing up a little of our time to dedicate to the most essential sections in order to create a captivating and solid experience.

Without further ado, let's get started: **I wish you a successful journey with your project and the best luck in the world with your next video game!**

Néstor García
BeWolf Studio’s founder

Day 1: Start with an idea, anything!

No matter how rare your idea may be, if you carry it out correctly, you could have a great experience in your hands. For Redd's Runaway, the main idea was to create a Run & Gun where you could find dozens of enemies and a giant final boss in each level, set up within a paranormal environment.

Even if you start with a big idea, you can scale down your proposal to design a plan that you can implement in a considerable time period; your first video game should not be the title of your dreams, so start developing your project with your current skills and abilities. **Eventually you will be able to repeat the process, with more experience, knowledge and ambitions that will be reflected in your next development.**

Day 2: Work with a great team

My artwork is very deficient and I still have a lot of trouble making good sprites or drawings. Considering this and after an art test inspired by Capcom's style (thanks to the help of my sprite teacher, Alex aka Li Kun), I decided that the approach was not ideal for this game and began to review other options. That's how I came up with a Metroidvania style, where I got in touch with Ariel Gonçalves, who has created some amazing sprite work (which you can see in the level bosses).

For Redd's design, I contacted Moone Blindgod, who developed the concept used as the basis for the game's protagonist. The most important point to me was to reflect a feeling of melancholy in her design, within a character with good agility to reflect she was able to convey to the player the feeling that she would know how to solve the problems on her way: a great determination that was born due to her personality and the rage she felt as the story unfolded. **By this time, Redd was already a character with strengths and weaknesses, which personally creates a bond with her that I can feel a lot of empathy for.**

Regarding the music, I had a big problem: with a limited budget, I decided to look for possibilities, and almost immediately discarded videogame stock music (since I found many generic themes in my search), and on one occasion I came across White Bat Audio's channel. **Several of its themes reminded me of my experience playing The Adventures of Batman & Robin on the Genesis many years ago, so I decided to take that route, which is complemented by the dark design of the settings and the vibrant**

design of the characters. Karl Casey is in charge of the main theme of the game, combining an Industrial Dark Synth and Synthwave style, very distinguished from the eighties and the films that came out at the time, especially the slasher genre.

I highly recommend following their work: **Redd's Runaway would not be what it is now if it were not due to the fantastic effort that Ariel, Moone and Karl put into the project.**

Day 3: Limits serve as a boost for creativity

Shortly before the idea of the game was defined, I was reading “Level Up”, a book on video game design written by Scott Rogers (totally recommended). Before Redd’s Runaway, I was working on “Reino”, a concept about a bounty hunter that would result in a very ambitious game that shares many ideas with Redd’s Runaway universe. **However, knowing that for my first commercial title, this idea could take a long time, I decided to take a step back and start with something smaller that I could see from beginning to end in a moderate time interval, in this case, little more than 7 months.**

On a certain page in the book, Scott comments that even a story like Little Red Riding Hood can be adapted to create a game. With that same idea in mind, I started brainstorming what I wanted the game to be; **although not everything was included, it gave me a direction that I could follow and that I could turn to in order to develop other ideas, especially needed for Redd’s world.**

Day 4: Put your project together with supporting documents

Before programming or creating any art for your title, I suggest you define your idea in one or several documents so it can help as a reference for current and future development; this is especially useful when you are working with a team.

Creating these documents can help you prepare small advertisement sketches for your title. **The One-Sheet, as its name suggests, is a document where you will capture the platforms, target audience, game summary, scheme and USP's** (Unique Selling Points, the most important features of your game that will help you direct the presentation of your title with your audience); also, similar products, competitors, and the main theme.

On the other hand, the Game Design Document (there is a version that also consists of 10 pages) establishes both the estimated release date, the key points of your development, and several of the previous themes in greater depth (game scheme , characters, controls, gameplay, game experience, mechanics, enemies, levels and advertising strategy).

The purpose of these documents is to help guide development with the least possible risk from the start. Good design documents can show you threats almost immediately, so you can start to consider possible solutions or even discard certain ideas to bring your game to fruition.

Nowadays, software development (including videogames) is looking to be more agile, and that's why sometimes, several teams choose two alternatives, more flexible and compact, to develop the vision of their game: the **Back of the Box Vision Statement**, and the **High Level Design Document**. The first option focuses more on a marketing-oriented approach, with bullet points describing the most important features of your video game. The latter option is a paragraph citing these characteristics in the same way, with an approach focused towards the development team, and therefore, more detailed.

In my case, reviewing the documents against the final project, I could find certain changes, but the vast majority of them served as the foundation for Redd's Runaway, **especially the key development points: Atmosphere, Control, Replayability and Action.**

Day 5: Choose a development engine

If you are working on your first video game, this section should not take you long, **considering that the scope of the project is well established and you do not plan to make an open world multiplayer 3D game or another idea as ambitious as this one.**

For Redd's Runaway I decided to use Godot after a good friend, Aldo Pedro, told me about the engine some years ago. **After doing my first exercises and creating a prototype of a platform game, I realized that I had found the perfect tool for my needs and requirements, which continues to grow and will be a fundamental pillar for my future projects.**

On the other hand, if you don't know how to program, there are certain engines that allow you to create a video game without even writing a line of code, so there are many options nowadays, a lot of them for free. You can start by taking a look at Godot, Unity, Unreal Engine, Game Maker Studio, or Buildbox.

Day 6: Create your prototype, a Minimum Viable Game

Once you have chosen the engine for your game's development, and before focusing on creating each and every mechanic, you should create a Minimum Viable Product, or in this case, a Minimum Viable Game. It should contain the most essential mechanics for your project, so its scope will be tight enough to see it complete in a couple of months. Here you can find a video prepared by **Extra Credits** (one of the best channels with game dev resources, its content is very educational and entertaining), which perfectly explains the idea behind the MVG.

In Redd's Runaway, the initial prototype took about a week, which consisted of moving back and forth, jumping, and creating random platforms so the player would face a challenge navigating between 3 different levels. **After that, you have to look out for feedback and iterate on the prototype.**

During the development of Gravity Spark, I decided to take several of the concepts from Extra Credits, and before I even made a sprite, I used blocks and other geometric figures to create the game's logic. **This made the development process more focused, so I worked with the base logic and mechanics at the beginning, and then at the end, I**

could adjust the assets without the hurry to finish, as the game was already done, mechanic-wise. Once complete, adjustments and fixes will always be necessary, but spending separate time between art and logic can provide a better approach to our game.

Day 7: Use gamification to get organized

Gamification "is a learning technique that transfers the mechanics of games to the educational-professional field in order to achieve better results, **whether to better absorb some knowledge, improve some skill, or reward specific actions, among many other objectives".**

For Redd's Runaway I spent many days trying to find the right way to organize my development schedule, and I remembered a video made by RagnarRox. After watching it again, I began to prepare my work schedule and check boxes in a notebook, working in order to fill at least 10 blocks daily; by the time I was in the 10th box, if I had more pending tasks, the remaining 8 would be done without problem; I did not always fill 18 boxes**, but this technique was especially useful on complicated days, since each box represented only a small portion of time, such as:**

- Add 3 boss sounds
- Apply a color palette
- First preview of the interface style

25 minutes of work, 5 minutes of rest, tasks better defined, and above all, more confidence by having clear objectives and problems to solve day by day.

Day 8: Develop your own style

After successes like Fall Guys or Among Us, it's common to see a wave of titles trying to replicate the virality of those games by taking what they believe are the key points of their development and leaving aside their most important details. I think there are great lessons to be learned from these videogames (such as the simplified and attractive visuals they have), and among those points, the fact that they developed their own style that is now part of their identity.

Personally, I was afraid on several occasions about public reception with Redd's Runaway, but I know that I also needed to take risks with a title that seeks to innovate in one aspect or another, perhaps with the greatest risk in trying to combine two beloved traditions, like Day of the Dead and Halloween, or molding this into a procedural action game, moving away a bit from a more horror-oriented proposal; **this risk is understandable and it only remains to create these ideas in the best possible way to reduce it as much as possible, for the benefit of the development and the players.**

Day 9: Participate in Game Jams to test your skills

There are dozens of Game Jams that are organized on a monthly basis, and their scope varies from one to another; in 2020, I had the chance to enter to GMTK Game Jam, where I developed a little game called Gravity Spark. **As I was in the middle of the development of Redd's Runaway, the planning of this Jam was perfect for me: it would only take 48 hours and in the end, I might get some feedback to consider if I would like to continue expanding this game in the future.** I was supposed to enter this jam with some close friends, but in the end only I was able to enter, so if you can, take this as an opportunity to participate with some of your friends, work on your own, or cooperate with new people to create a small game.

Not only that: I also applied some of the knowledge that I learned in recent months from various developers to make a more efficient project, and I uploaded a game to the itch.io and Play Store platforms for the first time. This was especially helpful for Redd's Runaway launch, as I already had an idea by then of the requirements that itch.io or Steam would ask for in order to accept the game and get as much exposure as possible.

Here are some interesting links so you can have more information before participating in a game jam; **I hope you can find something useful for you and your development!**

- Game Jam - Extra Credits - https://youtu.be/2xfxx27HbM4
- 13 Things to Know For Your First Game Jam - https://youtu.be/2KOEBWFC-1M
- The Top 3 Important Things To Polish - https://youtu.be/zBDeAILHV5k
- Making Your First Game: Minimum Viable Product - https://youtu.be/UvCri1tqIxQ
- How to Quickly Get Ideas in Game Jams - https://youtu.be/xe2X0WJgI-4
- 51 Game Design Tips! (In 8 Minutes) - https://youtu.be/5ijuH_oMu-U
- Lospec - Pixel Art, Tutorials & Palettes - https://lospec.com/
- MiniBoss' Pixel art tutorials - https://blog.studiominiboss.com/pixelart
- Videogame UI: The design process explained - https://medium.com/@a_kill_/video-game-ui-the-design-process-explained-14c23fa70d37
- TeknoAXE's Royalty Free Music - https://youtu.be/r5L-CY9TcRc

Day 10: Analyze your game mechanics

During development, it's likely that a couple of mechanics will not be in synergy with the rest of the project; in this case, it is recommended to analyze if discarding these snippets is the best solution, or if they require more work to create a much more cohesive proposal.

The development of Redd's Runaway initially included items such as explosive boxes that, once destroyed, would wipe out all enemies around, as well as damage Redd. **As development progressed, I realized that this mechanic would bring more problems than what it tried to solve for the player: this would require more precision to not destroy boxes nearby, due to the number of enemies, so they would end up being a hazard instead of a help,, in addition to require more effort to add new statuses to Redd and enemies.**

On the other hand, a good friend, Humberto, suggested that an evasion mechanic could help the player. **With the corresponding feedback and iteration, I decided to add the mechanic and realized that this gives the player more possibilities to make immediate decisions, be it attack, jump or slide to face the problem at hand.**

This case cites a small mechanic that makes the gameplay more dynamic; try to avoid major changes in mechanics as your development progresses, where it's best to focus on the work already done to make it solid and assuring the game is entertaining. **However, if you find this type of change necessary to create a more engaging player experience, take a closer look at the work required and plan to implement the new changes in the future, as soon as possible.**

Day 11: Fail faster

There has always been a great stigma around mistakes and failure. Perhaps in your personal life, education, or in your professional career, you have experienced that feeling, where there is no place to make any kind of mistake. This Extra Credits video came at a time when I needed it, as I had doubts about some of the mechanics I was planning, if I would be able to finish everything on time, and the story of the game.

What happened? I did my best and aside from the discarded mechanics, I checked my schedule to be sure the game would launch in the planned release window (October 2020), and also wrote the first draft of the game's story. **Although there were great days where everything flowed in such a way that it helped me get to the goal quickly, others were a bit more difficult than expected;** even with the story, I wrote two more drafts, going from a basic plot to a more complex story, with characters imbued with flaws and virtues that evolve as the game progresses.

Fail faster: test what you've created, get feedback and try one more time; you will eventually improve your skills and create more polished work!

Day 12: Seek help for testing

As indie developers, it is likely that on many occasions we will be the only ones to test and play our project until its launch, thinking that this is more than enough. **However, it’s always recommended to ask other people for feedback, so that they can give us a different point of view that can help us improve our project.**

Whether it's with our family or close friends, or a testing-focused group, be sure to share your progress with others; If you grow as a developer, you will likely be able to post your progress with a small community that will help you create an entertaining game that hopefully, will be liked by as many people as possible.

Day 13: Come and collaborate with the community

In case you have questions or are looking for feedback, turn to the indie community for help. There are many groups, especially on social media, focused on both indie games and specific development engines, such as Unity, M.U.G.E.N. and Godot.

Maybe a person has been through the same problem that you have now, and most of the time they will be happy to help you (I think the only time when this could be difficult is near a release date). Want feedback on your sprites, a game mechanic, or a new trailer? **The community is here to help. Try to help back as much as you can; you never know when your knowledge or experience can help a colleague to finish their next project.**

Day 14: Set the limits of your project and overcome perfectionism

Setting limits for your game can be a good long-term choice. There have been many titles, both indies and AAA, which have failed due to lack of limits. We must remember that our skills will improve and that we can add new ideas in a new title or sequel; **it is not necessary to add every idea we have in our current project.** Perfectionism, wealth, and endless continuous improvement were the deadly enemies of Duke Nukem Forever and Daikatana, to name a few games.

Plan a release date and a specific set of content for the player; develop your game and respect your schedule as much as possible. Delays are common (Redd's Runaway suffered a few of them), but it should be your priority to keep them under control as much as possible. Take this into account if you plan to release a game around a certain holiday, such as Easter, Christmas, or Halloween and the Day of the Dead. **Small ideas may appear from time to time; do a cost-benefit analysis and decide if it's worth adding a feature to your current project, if you can wait for your next title, or if you can add it in a possible post-launch game update.**

Day 15: Consider the timing to post your game

Most digital stores today require a couple of days to review your videogame and approve it. In the case of Steam, you must wait at least 1 month since the purchase of your first app credit (the currency used to publish a game), along with 2 weeks of presence in the store and the review of your game (about 1 week in case no major problems arise).

Consider these lapses so you can launch your game on time! This is especially important in games related to a specific date, such as a celebration, so be careful not to let the dates slip and miss your initial release window, and with it, potential sales.

Day 16: Give feedback to the player

When playing games, it’s a good idea to have indications of what is happening; whether it is using a user interface or a natural interface, it is good practice to give feedback to the player about their current state in-game. Examples vary, from the red borders in Silent Hill once the player is low on health, to the announcer voices and messages in Marvel Vs. Capcom, and the Ult indicator in Overwatch. Redd's Runaway implements a user interface inspired by the Run & Gun and Beat 'Em Up genres, so players can start a game with immediate information on energy, lives and weapons, for example.

You don't need to think about big systems to help the user: start with a readable user interface, a suitable set of sounds, and animations that clearly indicate when the player has done something right or wrong. This will also improve the user's gaming experience, so take your time to implement and refine the best feedback systems you can.

Day 17: Translate your game

To reach a wider audience, you should translate your game into other languages; depending on the genre of your title, this can make a big difference. Translating a fighting game may not be as essential as translating a visual novel. **It is a good idea to consider the variety of translations before starting your development, with a system designed to change texts and toggle images based on the language chosen by the player.**

Redd's Runaway uses a system focused on two languages: English and Spanish, and is designed to change the user interface and the texts of the stories based on the option chosen. **Allot a couple of days so that you can develop and implement a robust language system tailored to your needs.**

Day 18: Take care of your health

Remember to take care of yourself while you develop your project; I do not recommend you to work every day, day and night, to materialize your videogame, not even in game jams. **It's more important to create a good strategy that allows you not only to create a title with which you feel satisfied, but also that helps your development both personally and emotionally by avoiding all the non-required problems that you will encounter along the way.** If you maintain an exhaustive routine of more than 12 hours of work a day, 7 days a week, until the launch of your game that can last months or even years, you will most likely end up burnt-out and fed up with videogame development, for what you could give up to follow this line of work in the future.

Eating properly, getting some fresh air, doing at least 30 minutes of exercise, clearing up your mind and doing other types of unrelated activities is more than welcome; **don't forget to rest and take care of yourself!**

Day 19: The importance of a game trailer

Creating a trailer for your game is a complicated task. Even when it is not a requirement in all digital game stores, sometimes it turns out to be one of the most attractive elements to capture the attention of your audience in just a couple of minutes and convey the idea of your game in the simplest and most direct way possible.

Thanks to GMTK and Derek Lieu, during 2020 I prepared a trailer for Gravity Spark, implementing several of their teachings, in just one minute. **It is a fact that there will be more people watching the trailer of our game than playing it, so try to catch their attention from the first seconds and of course, focus on your game system (unless your title is focused on narrative), at least for your first game.** With enough experience, you will develop a style and be able to improve your technique over time, or on the other hand, work on the creation of new videos with professionals in the area.

Day 20: Sound effects are essential in the experience

A game session can be totally different depending on your sound section, so find and prepare the best sound effects for your project. This section can add a strong impact to certain scenes, or on the contrary, remove all impact from your gameplay. Not only that, but it can also be used to provide feedback to the player: you can assign a sound for life recovery, for example, and the user will eventually be able to associate such an effect with that action within your game.

The best horror video games are great examples of this point, as they are capable of generating endless emotions in the player with just a dark environment, using a variety of excellent sound effects.

Day 21: Create conflict between your characters

Redd's Runaway features a variety of main characters with conflicting values. Inspired by the work of John Truby and the Just Write YouTube channel, especially the 4 Corners Opposition, I tried to shape a story where Redd would be under constant pressure due to different problems with each person in the story, including herself. I wanted all the characters to have a personal and intimate point of view on revenge. **Redd is, at heart, a protector, but throughout the game, her values will be in constant struggle, as she is still in a state of mourning due to the events leading up to the start of the game.**

By developing characters with different values, we can create a story full of problems, not only external, but internal. I know this can be very ambitious, especially for a Run & Gun game where the story is generally unimportant or too basic, but I also wanted to express my love for writing here, by developing a story that I hope will captivate the player.

Day 22: A customizable experience is vital

Provide as many options as possible for your player to customize their gaming experience. **From graphics to controls to audio, these settings can help create a much more enjoyable run with your videogame.** Some ideas are control mapping, volume change, or removing any filters from the screen.

As you deepen your development techniques, with new updates or even new games, you will be able to develop a library that will incorporate many different customization options to welcome new and established players, allowing each one to have a personalized gaming experience to their liking.

Day 23: A contained situation may be your ideal choice

Nowadays, it is quite common to watch movies or games in which the world or the entire universe is at risk. **While this can be an interesting idea, you can always turn things around and limit the event to a very specific region, city, family, or person.**

Contained situations can open doors for the players to question their ideas, especially if there are some unexpected twists throughout their journey. Creating an end of the world within a story must be very difficult to perform and even more to make it credible to the public, while creating a single accident, even if it is not a large-scale event to a global scale, can be much more believable, and it will make us think that there are many immediate consequences at stake for the characters present.

Day 24: Create a website for your projects

The creation of a website can be of great help for your projects; nowadays, you can even create a free portal in just a couple of hours, which is especially useful to share more information with your players and the press. Wordpress or development hosting services like Github allow you to create a free public website quickly and easily.

For BeWolf Studio, the simplicity of the site was important, which is why you can find key information on each page written in just a couple of lines, along with high-quality images of the developments created and a press kit of the title, courtesy of https://dopresskit.com/

Day 25: A repository is important!

This should be one of the priority actions to take just to start with the development of your videogame: create a repository for your project! I remember many cases with my friends and colleagues in the MUGEN forums, where suddenly, they had lost all the progress made during months and even years of work, and they had no backup to continue, which made them lose the motivation to redo their projects again. For Redd's Runaway, it took me longer than I would've liked to, but with my entry in new game jams, this step was ready even before the projects started. Whether its code, resources, or documents, a repository allows you to store a variety of information to keep your work up-to-date in the cloud.

The advantage of repositories is that they also open the doors to collaborative work, with different people working on different parts of the project, or if they are working on the same file, it creates a merge between the changes that each one has uploaded. Personally, I recommend working on separate files, downloading changes at the beginning of your day, and uploading your new additions at the end of your day, but in the end, these types of work routines change eventually with each person's experience.

Services like Github or Gitlab can be helpful for this task, and there are many guides available to help you step by step. **If you have made a lot of progress with your game but you don't have a repository created yet, take at least a day to prepare it before continuing, and have a little more certainty regarding the security of your project.**

Day 26: Check your game performance

Be sure to review your title performance when starting your game, and periodically throughout the development cycle. If you see any kind of fluctuation in your performance data, you may be able to easily solve it up front, which will also provide a smoother experience for your testing team. Don't take long periods of time to do tests, as your project can accumulate too much information, systems or functions, which will make it more difficult to trace the origin of any type of performance anomaly.

Day 27: Prepare for launch day

As the launch of your game approaches, consider creating and updating all the content necessary to announce it to the public, such as:

- Website update
- Presskit
- Latest game build on digital platforms
- Trailer
- Playlists
- Etc.

This will help you to free a bit of your workload during that day, taking time exclusively to share posts about the game and answer as many questions as possible, in addition to fixing any critical bugs that may come up and ruin the gaming experience.

Day 28: Congratulations on your first game!

Good job! Even in a month, you can create a short but fun gaming experience that doesn't necessarily have to be sold. **Don't be afraid and experiment: you can adapt your marketing strategy around this, upload your game to some stores like itch.io, and gradually get used to the experience of finishing and launching a game.**

In general, remember that you can only get better and that not everything will be perfect on your first try. **As long as you keep in mind that this is an iterative process that will help you improve your skills and therefore your games, you can follow this workflow in the future and release better games, with a more detailed design or a longer playtime, focused on captivating your players and meeting your expectations.**

Day 29: Adapt to the circumstances

Redd's Runaway went through a lot of ideas and iterations, discarding some elements and adding others that were not considered at the beginning. Still, in the end, I am grateful for the experience of designing a game and releasing a title that I hope will be entertaining for its players; **to make it possible, you have to adapt to many circumstances, some within your control and others totally unrelated, such as time, budget or ideas.**

I think that game design and development are all about problem solving, so when you are faced with a great difficulty that you can't seem to find a solution for, remember that it is part of the process and don't be discouraged; break the problem down into smaller parts and find a solution for those parts. Ask for help, take some time, take a different approach - take a look at your options and keep moving forward step by step.

Day 30: Make a devlog of your experience

Document your journey through your game's design to learn from your own experience and share it with the community. We can learn from your situation, problems and successes in the development of the game you have worked on so far for so long, and it can also help us to continue improving our projects, creating a complete feedback network.

Not only that, but it will help you especially when it comes to reviewing your work objectively, to see which aspects need to be polished and which ones were key to the development and success of your game.

Day 31: Iterate about what you've learned

With the release of your first game, you will have a wealth of new knowledge, accumulated from all your development time. It is a good idea to consider if you would like to make another game, and if the answer is yes, try to put everything you have learned into practice. **You now have a better perspective on the requirements of a game, as well as more experience related to work schedule, best development practices, mistakes to avoid, and interesting ideas you want to improve, to name a few examples.**

This was a great experience, since I had not created a daily devlog around a commercial project, now it has become a resource guide that I hope will be useful for you. Thanks for checking out this development guide, and thank you very much for all your support.

Developing Redd's Runaway has been an amazing experience that I am truly grateful for! Thank you all!