

TABLE OF CONTENTS



TEAM GUIDE

VERSION 0.2.0 - 12/06/2025

TABLE OF CONTENTS

As a shortcut, you can click on the chapter you wish to visit below.

CHAPTER 1 - ONBOARDING

CHAPTER 2 - ABOUT US

CHAPTER 3 - EXPECTATIONS

CHAPTER 4 - ROLES

CHAPTER 5 - TOOLS

CHAPTER 6 - COMMUNICATION

CHAPTER 7 - PROJECT FRAMEWORK

CHAPTER 8 - TRAINING

CHAPTER 9 - DOCUMENTATION

CHAPTER 10 - GUIDELINES

CHAPTER 11 - SUPPORT

CHAPTER 12 - DICTIONARY OF TERMS

CHAPTER 13 - RESOURCE LIBRARY

(illustration here)

WELCOME TO OUR GUIDE!

Welcome to our rewritten and improved Team Guide! In this Guide, you should be able to find everything you need to know!

This includes links to any other essential, more in-depth resources and templates; isn't that neat? Everything you need is in one place. Think of it as a library or directory where everything is up to date, as changes are made within the Studio.

Now, what are the improvements? Well, simply put, I didn't overcomplicate anything in this document itself this time (at least i'd like to think so); instead, I simplified it. For more in-depth information, documents related to the topic can be found in the relevant chapters and the resource library section.

You navigate this Guide using the tabs & outlines overview; most extensive documents going forward will use this format for chapters.

Anyways, thanks for sticking around

- Dunno

(illustration here)

NOTES ON PROGRESS

This guide is still a work in progress. Some resources, links, and visuals are missing or incomplete; however, they'll be added, updated, and improved over time.

Thanks for your patience. If something's missing and you're unsure how to proceed, feel free to ask for help. Once our first project wraps up, we'll use the one-month break period to polish and finalize as much as possible.

I will continue to update and refine this guide as we grow.

DISCLOSURE

Transparency is one of our most important values as a Studio and Team; hence, I must disclose that this document, as well as similar ones, was written with the assistance of generative AI.

While I have revised, rewritten, and reiterated versions of the exact text so many times that you can barely trace back my initial prompt, I still can't take credit for what a machine helped write for me, and neither should you if you were in my position.

I do not encourage nor accept the use of Generative AI without explicit permission in CREATIVE Projects, and never within the Production phase in final deliverables. Nor will it be acceptable to use Generative AI to generate sound or visuals, EVER. And I highly doubt I'll have to make an exception, even for reference material. So, why did I use it for this document?

I permit the use of AI in certain instances, such as these, because what AI does best is repetitive and data-intensive tasks. What humans do and understand best is creativity and social-emotional intelligence.

While this is a boiled-down version, the idea is clear: using AI for something like creativity is ultimately meaningless, soulless, and brings no value. Art is what humans crave and have treasured for centuries: the ability to express ourselves. Letting a machine do that for us is nothing near as fulfilling as facing the hardships, making discoveries and doing so ourselves.

But hey, I'm proud to say I wrote this, including the "WELCOME TO OUR GUIDE" section, without a single trace of AI except grammar correction tools.

To learn what we deem as acceptable and unacceptable use of generative AI, head to our guidelines:

[GENERATIVE AI GUIDELINES →](#)

ONBOARDING

ONBOARDING

ONBOARDING

Welcome to the Team, newcomer! Guess what ... This chapter is all for you! Or, well, any Team Member who needs to reboot their system.

Anyway, this chapter is all about the onboarding process and how you'll integrate into our Team. Essentially, we would like to introduce ourselves, outline a timeline, and provide you with an overview of the opportunities available.

And if you don't enjoy reading, you could pair it with our introduction video, which essentially covers this entire chapter and more:

VIDEO: W.I.P

MISSION & MINDSET

As a part of HELLINGTON STUDIO, we must establish that you align with what we seek to accomplish. [OUR MISSION](#). Like many Indie studios, we want to push unfiltered and original storytelling. However, not just for entertainment, but to evoke something in people, provoke thought, and ultimately leave an impact, leading to a more critical-thinking audience.

For every project we do, this should be a priority. And if you want to stick around, we need to know that we're on the same page about achieving this goal.

ONBOARDING TIMELINE

To get started, you'll need to complete the onboarding, integration, and training processes. Familiarize yourself with each process below.

1. ONBOARDING

Essentially, we gotta get you up to speed with things. We'll help introduce you to our Studio by going through all the essential tools and systems you'll be using and explaining generally how we operate. This will be accomplished through one-on-one meetings with a *MANAGER* or *ADVISOR*, where we will walk you through each step.

2. INTEGRATING

Through the *TEAM LEADERS* and *MEMBERS*, you'll be introduced to any ongoing projects. We'll try to involve you in Projects just getting started, so you can get an idea of how a production works from start to finish. However, if there are no projects where you can be assigned immediately, we'll likely use it as an opportunity to provide you with training sessions.

3. TRIAL PHASE

Depending on your Skillset, you'll go through a set of tailored training sessions, which can include tool tutorials, feedback cycles, small missions, and one bigger thing: a Trial Project.

The Trial Project is an opportunity to gain hands-on experience. You'll work on something small but complete that fits your Skillset Roles. This helps us get to know your process and style, and it helps you figure out if this way of working actually feels right for you.

We'll assign an *ADVISOR* to guide you and give feedback along the way. They're not here to micromanage; they're here to support and help you grow.

By the end, you get to decide whether our Studio is for you or not. If not, no pressure - you can always change your mind, come back, or take some time to reflect before officially joining. That's what this trial phase is for.

4. OPPORTUNITIES

Everyone starts as a *TEAM MEMBER*; however, by contributing to Projects, Missions, and the Community, you build trust and belonging within our Studio, proving yourself a candidate for potential higher positions, such as *MANAGER*. For instance, a *CREATIVE MANAGER* who is exceptionally skilled within the *VISUAL* department is entrusted with primary influence in key decisions regarding visuals.

Essentially, the more we trust you, the more we entrust you with key decisions within the Studio and Projects, which are reflected in a promotion.

5. FEEDBACK & SUPPORT

Feel free to ask for help whenever you need it. Higher ranks should always be available to support you and answer most of your questions.

GETTING STARTED

Now that you have a general idea of the onboarding timeline and process, let's get started with helping you settle in.

NOTE: If you don't want to go through this process on your own, you can ask a Higher-ranking *MEMBER* for help completing this process. As mentioned in the Timeline, a Manager or Advisor will be helping you out with this anyway; however, if you want a head start, keep following this Guide.

TOOL SETUP

The most important tools we'll be using are Discord for communication and Google Drive for file management. When it comes to Discord, all you need to know is how to navigate the app. However, for Google Drive, we have a template set up that we advise you follow, along with guidelines for specific formatting and naming conventions for files.

DISCORD

Follow this Guide to learn how to navigate our Discord server:

[DISCORD TEAM SERVER GUIDE HERE →](#)

GOOGLE DRIVE

Follow this Guide to set up your HELLINGTON STUDIO Gmail and Google Drive workspace:

[GOOGLE DRIVE SETUP HERE →](#)

OTHER TOOLS

Any other tools we all use in common will be introduced when relevant. Tools that we approve for use can be found in the [TOOLS](#) chapter. Here, you can also find tools categorized by relevant skill sets and roles.

CHAPTERS TO READ

While starting out, you won't need to know everything right away. But the recommended chapters to begin with are:

- **ABOUT US** - Discover more about our Studio, what we do, and what we value. This will further help you understand the mindset you should have while approaching projects here.
- **EXPECTATIONS** - Learn what we expect from you and what you can expect from us in return.
- **COMMUNICATION** - Gain a better understanding of how we communicate and what tools and conventions we use to make our workflow work together
- **PROJECT FRAMEWORK** - Gain a detailed overview of how projects work from start to finish and learn about the types of projects we typically take on so you can pitch some of your own.
- **DOCUMENTATION** - Become familiar with how to document your work within the Studio and why it is crucial to do so.

ONBOARDING CHECKLIST

We have developed a simple checklist for you to follow to ensure you've completed all the necessary steps you need for onboarding:

ONBOARDING CHECKLIST HERE →

(illustration here)

ABOUT US

ABOUT US

In this chapter, you'll learn about our Studio, our story, our goals and values. It's important you understand and share our mindset to help us achieve them.

OUR STORY

We had a simple goal: to tell stories, no matter what medium. But looking at what the industry has come to, we knew we had to do more than just entertain.

Today's media is designed to be passive, simply quick dopamine hits to keep people numb and obedient (complacent if you will). Content is consumed mindlessly, distracting you instead of influencing you to think for yourself.

So we want change. We seek stories that alter perceptions, leave lasting impressions, and ultimately inspire people to take action. We aim to initiate a movement to revive critical thinking.

OUR MISSION

We want to provoke critical thinking through storytelling. While our work comes across as silly and straightforward on the surface, those who dig will find it challenging to hit rock bottom.

OUR VISION

We're here to take back control, returning power to the people by reshaping audiences into critical thinkers and setting a new standard for storytelling.

OUR POSITION

We want change in an industry where corporations drain the same franchises, strip stories of their meaning, and reduce creativity to a formula for profit.

Like many indie creators, we believe in the power of originality and unfiltered storytelling despite the risks. We push boundaries not for shock value but to restore meaning to an industry that too often sacrifices soul for sales.

OUR VALUES

AWARENESS

We don't just consume, we're attentive. We value thinking for yourself, questioning what you're told, and recognizing what hides beneath the surface. The first step to change is being aware.

POTENTIAL

We see potential in everyone and everything. No idea is a waste, and even flaws have a purpose. We believe greatness often grows from the unexpected or the overlooked.

TRANSFORMATION

Learning and change are an endless pursuit. We're here to evolve creatively, personally, and collectively. Nothing stays static, whether it's an idea, a person, or a story.

WHY THIS MATTERS

This isn't just about talent. It's about mindset. It's important that you understand and share this mindset if you want to work toward the goals we've set for this Studio.

Remember, this is not about profit.
It's not just about entertainment.
It's about change.

(illustration here)

EXPECTATIONS

EXPECTATIONS

This chapter is about what we expect from you and what you can expect from us.

MINDSET

The mindset you should have when approaching your work and growth at HELLINGTON.

CONTINUOUS LEARNING

Learning is an endless pursuit, and change is constant. As technology and the industry evolve, so should we. Refine your skills whenever the opportunity arises. Keep an eye on emerging trends and stay committed to your personal growth. Participate in training sessions when you can; no one is ever done learning.

COMMITMENT

Don't just show up, show you care. Stay committed to our Mission and give your best to every task, big or small. Create work you can be proud of. If you join a project, see it through to the end. Challenges will come, push through them, and grow from them.

CREATIVE FREEDOM

You are free to experiment and try out new things. Don't treat this guide as law; if you discover a better way, propose it and set a new standard. Have a mind of your own, don't be a blind follower. We reward those who take risks, challenge assumptions, and drive change. Just make sure you know the rules to break them meaningfully.

(illustration here)

BEHAVIOR & ATTITUDE

How we expect you to behave, communicate, and carry yourself within the Team.

RESPECT

Respect your fellow Team Members and recognize that their ideas and processes may differ from yours. Although an idea may seem stupid to you, you need to consider that you may not be the right audience for it. Approach everything with an open mind and offer constructive, balanced criticism. Be honest, but also fair.

ACCOUNTABILITY

Own your actions. Own your mistakes. We don't expect perfection, but we do expect honesty. If you make a mistake, don't hide it; address it, learn from it, and move forward. That's how we grow.

Essentially deliver what you commit to, or communicate if you can't. If you take on a task, follow through on it. Life happens; if you can't complete something, just say so as soon as possible, especially considering others may be dependent on you finishing first. We'd rather adjust than be left guessing.

FEEDBACK

Give and receive feedback frequently, and make sure you learn from it and apply the lessons. It's essential to be fair and constructive in your criticism, meet the person where they are, and focus on what will help them progress rather than discourage them.

When giving feedback:

- Be clear, specific, and constructive, not vague.
- Focus on helping them improve, not tearing them down with excessive criticism.
- Always praise what's working, too. People need to know what to continue doing, not just what to fix.

When receiving feedback:

- Listen openly, even if it's hurtful.
- Ask questions if you don't fully understand it.
- Apply what's useful, and don't let your ego block your growth.

If you can't give and receive feedback well, you won't improve where you need it most. That's why it's crucial to approach both sides of the process with full attention and an open, mindful attitude.

TECHNICAL

Comfort with modern technology and software isn't just a preference; it's a necessity. You're expected to adapt and navigate the digital tools we rely on. You should have the digital literacy to:

- **ORGANIZE AND NAVIGATE FOLDER SYSTEMS** - Know where to find what, and keep things tidy
- **ADAPT QUICKLY TO NEW TOOLS AND USER INTERFACES** - Be comfortable learning as we evolve our workflow
- **USE COLLABORATIVE PLATFORMS EFFECTIVELY** - With the Team, Discord, Google Drive, and others
- **HANDLE BASIC FILE MANAGEMENT** - Export, compress, and share files in the correct formats
- **RESPECT VERSION CONTROL AND BACKUPS** - Duplicate and label versions correctly, avoid overwriting others' work
- **COMMUNICATE CLEARLY IN DIGITAL SPACES** - Use threads, replies, tags, and comments where appropriate
- **APPLY BASIC TROUBLESHOOTING AND INDEPENDENCE** - Look up small fixes before asking, be self-driven
- **MAINTAIN BASIC SECURITY AND PRIVACY** - Strong passwords, keep Studio content confidential unless cleared to share

We're not asking you to be a tech expert, but if you struggle with basic digital workflows, this environment may be challenging for you to work in. Adapt to the best of your ability; the tools will evolve, and so should you.

(illustration here)

LOGGING

Log your work consistently. Document your progress (see [DOCUMENTATION](#) chapter). This isn't about micromanaging everything; it's about tracking progress, recognizing everyone's contributions, and keeping the Team aligned.

(illustration here)

AVAILABILITY

If life happens, communicate. If your availability changes, please let us know. We'd rather adapt or find alternatives than not hear from you at all.

Taking a break is okay, but ghosting is not. Need time off? No problem. Just communicate it. Ghosting without explanation, leaving us hanging, will only break our trust and make it harder for us to rely on you in the future.

Passion and contribution matter more than the hours spent. We don't care how many hours you log; we care about your impact and the presence you bring. Quality beats quantity.

CONFIDENTIALITY

Please be respectful and refrain from sharing any information regarding the Studio outside of the Studio without explicit permission. Simple as that.

As a Team Member, you must keep all projects, discussions, and your involvement private until we decide it's ready to be shown to the world.

This also applies to sensitive situations that may arise internally. If something happens, it is almost always better to keep it within the Team than to let it spread or risk being misinterpreted by bad actors.

When in doubt: ask first. It's best if everyone is on the same terms and aware of what's being shared.

(illustration here)

WHAT TO EXPECT FROM US

What you can count on from the Studio and how we support your creative journey.

A TRANSPARENT ENVIRONMENT - This should go without saying, but we value and prioritize transparency and authenticity. No PR stunts or unnecessary twisting of the truth, in the end, it only degrades trust.

A SPACE TO TAKE CREATIVE RISKS - You are encouraged to experiment, push your ideas, and challenge conventions, rather than playing it safe.

FEEDBACK THAT HELPS YOU IMPROVE - We provide honest and respectful feedback. The goal is to help you grow, not to tear you down.

SUPPORT FROM MANAGERS AND TEAM MEMBERS - You won't be left on your own. You'll have access to guidance, resources, and people who want to see you succeed.

OPPORTUNITIES TO LEARN AND IMPROVE - Through projects, feedback, and resources, you'll have chances to develop your skills and expand your thinking.

RESPECT FOR YOUR TIME AND WELL-BEING - We value your mental and physical health. If you need a break, take it, but please communicate this in advance.

A HIERARCHY VALUING EVERYONE THE SAME - While we have a system for leadership and management and trust them with key decisions, we still value everyone's input the same. Everyone is still valued equally, regardless of administrative status.

A CULTURE THAT VALUES THINKING PEOPLE, NOT BLIND FOLLOWERS - We want critical thinkers and curious minds. You won't be asked to follow rules without challenging them, you'll be expected to contribute with purpose and a mind of your own.

ROLES

ROLES

This chapter outlines how our role system and hierarchy function, as well as the practical implications of each role.

TEAM ROLES

While we maintain a hierarchy for the sake of structure and clarity, everyone's voice is valued equally. A **MANAGER** may hold decision-making power within their department, but that doesn't make their ideas more important than a **MEMBER's**.

The hierarchy simply reflects the level of responsibility and influence each role holds in guiding and managing the Studio's direction, not a measure of worth or creative importance.

(illustration here)

LEAD

Directs the overall Studio or a significant project. Responsible for the final vision, alignment across teams, and key creative and strategic decisions. The LEAD ensures that everything HELLINGTON produces aligns with its values, goals, and purpose.

RESPONSIBILITIES

- Define and communicate the project or Studio vision
- Make final decisions on creative and directional matters
- Approve major ideas, concepts, and deliverables
- Support MANAGERS and help resolve interdepartmental conflicts
- Represent the project or Studio in external communication

EXPECTATIONS

- Good leadership and decision-making skills
- Big-picture thinking with attention to cross-team alignment
- Clear communicator and confident presenter
- Ability to stay calm under pressure and guide teams through ambiguity
- Upholds and protects the core values of HELLINGTON STUDIO

WORKS CLOSELY WITH

ALL MANAGERS, PROJECT MANAGER, IDEATOR, ADVISOR

OUTPUT EXAMPLES

Vision documents, final creative decisions, Studio goals, project approvals

MANAGER

Oversees a specific department or function (e.g., VISUALS, WRITING, SOUND, TECHNICAL, FINANCES). Responsible for guiding contributors in their field, maintaining quality, and ensuring the department delivers on time and in alignment with the overall vision.

RESPONSIBILITIES

- Coordinate and support MEMBERS in a specific category
- Review and give feedback on departmental work
- Align tasks with broader project timelines and goals
- Act as the communication bridge between *LEAD* and their Team
- Flag roadblocks and manage task flow

EXPECTATIONS

- Deep understanding of their field or discipline
- Reliable, communicative, and well-organized
- Able to give actionable feedback respectfully
- Keeps contributors motivated and on track
- Helps maintain clarity and consistency within their department

WORKS CLOSELY WITH

LEAD, *MEMBERS* in their department, *PROJECT MANAGER*, other *MANAGERS*

OUTPUT EXAMPLES

Team updates, department task lists, feedback notes, short-term roadmaps

MEMBER

The core of the Studio, a creative or technical contributor actively working on projects. MEMBERS bring their individual skillsets to the table, collaborate with others, and help turn vision into reality.

RESPONSIBILITIES

- Execute tasks and contribute based on their skillset role(s)
- Participate in giving feedback and Team meetings
- Communicate progress and ask for support when needed
- Align work with Studio tone, values, and project goals
- Share ideas, suggestions, and creative solutions

EXPECTATIONS

- Delivers work in a timely manner
- Open to feedback and collaboration
- Actively communicates availability and progress
- Committed to both the creative process and the Team

WORKS CLOSELY WITH

Their MANAGER, *PROJECT MANAGER*, other MEMBERS on the same project

OUTPUT EXAMPLES

Depends on their skillset: scripts, art, code, sound, assets, documents, etc.

ADVISOR

An experienced Team member with a deep understanding of HELLINGTON STUDIO's systems, values, and workflows. ADVISORS act as internal guides, offering mentorship, helping onboard new members, and supporting other Team members with insight, direction, and process knowledge. While they do not manage others, they are often the go-to for advice and clarity.

RESPONSIBILITIES

- Support new members during onboarding and integration
- Offer guidance to Team members who need help navigating tools, roles, or workflows
- Share knowledge about Studio processes, expectations, and culture
- Participate in key feedback when insight is needed
- Act as a trusted source for *LEAD* and *MANAGERS*

EXPECTATIONS

- High level of familiarity with HELLINGTON STUDIO's systems, structure, and vision
- Good communicator: helpful, supportive, and patient
- Gives constructive feedback without micromanaging
- Proactively offers help when they see someone struggling or confused

WORKS CLOSELY WITH

LEAD, MANAGERS, NEW MEMBERS, IDEATOR

OUTPUT EXAMPLES

- Onboarding walkthroughs or welcome sessions
- Internal guide documents or clarifications
- Peer mentorship conversations or DMs
- Process feedback and best practice suggestions
- Flagging gaps in documentation or communication
- Occasional strategic insight shared with *LEAD* and *MANAGERS*

SKILLSET ROLES

In addition to your Team Role, every member holds one or more Skillset Roles based on their creative or technical strengths. These roles define what you contribute to the Studio, whether it's writing, art, sound, systems, or ideas.

Unlike Team Roles, Skillset Roles are not hierarchical. They exist to show what you do, not how influential you are in leading the Studio. You can hold multiple Skillset Roles, and they can evolve as you grow. Each Skillset Role belongs to a specific category for organizational clarity.

(illustration here)

WRITING

Focused on story, structure, dialogue, and written content of all forms.

NARRATIVE DESIGNER

Designs how the story is experienced through gameplay. A NARRATIVE DESIGNER doesn't just write; they engineer the structure of storytelling across systems, choices, pacing, and interactivity. They ensure that every narrative element serves the player's experience, making the story playable, dynamic, and meaningful.

RESPONSIBILITIES

- Design branching paths, player choices, and interactive story systems
- Translate narrative arcs into flowcharts, logic maps, and scene structures
- Collaborate with *GAME DESIGNERS* to embed story into mechanics and gameplay
- Work with *STORYWRITERS* to adapt story content into interactive formats
- Define tone, player emotion, and pacing at key narrative beats
- Maintain narrative bibles and track story consistency across content

EXPECTATIONS

- Deep understanding of narrative structure for games and non-linear media
- Able to think systematically about a story (not just what's told, but how and when it's experienced)
- Familiar with flowchart tools or scripting formats (e.g. Twine, Ink, Yarn)
- Excellent communicator who bridges creative writing and design logic
- Open to iteration and balancing storytelling with gameplay constraints

WORKS CLOSELY WITH

GAME DESIGNER, STORYWRITER, SCREENWRITER, 2D ARTIST, PROJECT MANAGER, IDEATOR

OUTPUT EXAMPLES

Branching dialogue trees, narrative flow diagrams, story logic maps, interactive scripts, narrative system design docs

STORYWRITER

Specializes in crafting compelling stories, characters, and worlds across various media. The STORYWRITER focuses on the core narrative, developing ideas, themes, and plotlines that serve as the creative backbone for games, films, or other formats. Unlike a *NARRATIVE DESIGNER*, the STORYWRITER is not concerned with systems or structure; they focus on what the story is, not how it plays.

RESPONSIBILITIES

- Create original stories, character arcs, and narrative ideas
- Write story treatments, summaries, and chapter outlines
- Develop compelling dialogue, lore, and emotional stakes
- Work closely with *NARRATIVE DESIGNERS* to adapt the story for interactive use
- Contribute to early ideation and concept development

EXPECTATIONS

- Good creative writing ability and storytelling instinct
- Able to write emotionally resonant narratives across genres
- Comfortable working in prose or outline format
- Willing to revise and collaborate with feedback from other roles
- Understands character development, theme, and structure, even in a linear format

WORKS CLOSELY WITH

NARRATIVE DESIGNER, *EDITOR*, *IDEATOR*, *SCREENWRITER*

OUTPUT EXAMPLES

Story outlines, narrative treatments, character bios, short stories, narrative pitch documents

SCREENWRITER

Writes dialogue and scenes in screenplay format for cutscenes, trailers, voiceovers, or cinematic content. The SCREENWRITER focuses on how the story sounds and flows moment-to-moment, bringing characters, pacing, and tone to life through structured, performance-ready writing.

RESPONSIBILITIES

- Write cinematic scripts, dialogue scenes, and VO sequences
- Translate story beats into screen-ready sequences
- Collaborate with *CINEMATOGRAPHERS*, *VOICE ACTORS*, and *EDITORS* to align text with visuals and sound
- Revise scripts based on tone, pacing, or visual timing needs
- Ensure character voices are consistent and emotionally grounded

EXPECTATIONS

- Proficient in screenplay structure and formatting
- Good understanding of subtext, pacing, and spoken rhythm
- Able to write natural, emotionally resonant dialogue
- Open to feedback from directors, editors, and voice talent
- Understands how scenes translate visually and audibly

WORKS CLOSELY WITH

NARRATIVE DESIGNER, *STORYWRITER*, *CINEMATOGRAPHER*, *VIDEO EDITOR*, *VOICE ACTOR*, *COMPOSER*

OUTPUT EXAMPLES

Screenplay documents, dialogue scripts, trailer scripts, VO sheets, timing-locked cutscene scripts

COPYWRITER

Crafts clear, concise, and compelling written content that communicates ideas across interfaces, campaigns, and platforms. The COPYWRITER's job is to deliver the message, whether it's a one-line tagline, a dev log, or a tooltip, in a way that's sharp, engaging, and aligned with the Studio's voice.

RESPONSIBILITIES

- Write UI copy, in-game descriptions, prompts, and tooltips
- Create marketing content: taglines, announcements, campaign blurbs
- Draft blog posts, dev logs, and other article-style content
- Collaborate with *MARKETERS* and *GRAPHIC DESIGNERS* to align words with visuals
- Maintain tone and voice consistency across all communication

EXPECTATIONS

- Able to switch between short-form and long-form writing effectively
- Writes with precision, clarity, and personality
- Adapts voice based on audience, platform, and project tone
- Familiar with writing for creative projects, games, or digital products
- Works well in fast-paced feedback cycles

WORKS CLOSELY WITH

MARKETER, GRAPHIC DESIGNER, UI DEVELOPER, PROJECT MANAGER

OUTPUT EXAMPLES

UI labels, tooltips, tagline variations, dev logs, social media posts, website text, Discord announcements, newsletter blurbs

EDITOR

Ensures that all written content is clear, polished, and consistent with the Studio's voice. The EDITOR focuses on refining language, maintaining tone, correcting grammar, and elevating the overall quality of written materials. Their role is to catch what others miss and make good writing great.

RESPONSIBILITIES

- Proofread and edit all types of written content (scripts, UI, articles, lore docs, marketing)
- Provide constructive feedback to *WRITERS* and *COPYWRITERS*
- Ensure consistency in grammar, punctuation, formatting, and tone
- Help maintain internal language/style guides where necessary
- Identify continuity issues or unclear passages

EXPECTATIONS

- Excellent grasp of grammar, syntax, and flow
- Skilled at editing without losing the writer's voice
- Clear communicator, both when editing and giving feedback
- Detail-oriented, patient, and reliable
- Comfortable editing across formats: narrative, technical, marketing, and UI

WORKS CLOSELY WITH

NARRATIVE DESIGNER, STORYWRITER, COPYWRITER, PROJECT MANAGER, MARKETER

OUTPUT EXAMPLES

Polished scripts, revised UI text, proofread blog articles, inline feedback comments, final QA passes on copy

VISUALS

Focused on artistic, illustrative, and design-based visuals for assets and storytelling.

2D ARTIST

Creates illustrations, icons, and visual assets in 2D format. 2D ARTISTS define the look and feel of characters, environments, and props through drawings and painted artwork, used in concept, in-game visuals, or promotional content.

RESPONSIBILITIES

- Design characters, items, scenes, and UI icons
- Create concept art to guide other visual roles
- Produce final assets for in-game or promotional use
- Adapt work to fit a defined visual style
- Support storytelling through expressive visual design

EXPECTATIONS

- Good fundamentals in drawing, composition, and color
- Able to work in a consistent visual style
- Familiarity with software such as Photoshop, Clip Studio Paint, or Krita
- Open to feedback and able to iterate quickly
- Understands how to create visual clarity in storytelling

WORKS CLOSELY WITH

STORYBOARD ARTIST, *GRAPHIC DESIGNERS*, 3D ARTIST, IDEATOR, *WRITERS*

OUTPUT EXAMPLES

Character sheets, environment illustrations, visual assets, UI icons, item artwork

3D ARTIST

Transforms concepts into three-dimensional assets that serve gameplay, cinematics, or promotional visuals. The 3D ARTIST is responsible for modeling, texturing, and preparing assets that feel alive and cohesive in a game or media environment, bringing depth, form, and fidelity to the visual world.

RESPONSIBILITIES

- Create models for characters, environments, props, and modular assets
- Texture and surface 3D assets with visual consistency and style accuracy
- Translate 2D concepts into accurate, polished 3D forms
- Optimize models for performance, animation, or rendering pipelines
- Collaborate with *TECHNICAL ARTISTS* and *ANIMATORS* to ensure compatibility and functionality

EXPECTATIONS

- Proficient in 3D software (e.g. Blender, Maya, ZBrush, Substance Painter)
- Good understanding of topology, UV mapping, materials, and baking workflows
- Able to match concept art and maintain style consistency
- Open to feedback and able to iterate based on both technical and artistic direction
- Familiar with exporting assets into game engines or animation pipelines

WORKS CLOSELY WITH

2D ARTIST, TECHNICAL ARTIST, ANIMATOR, VIDEO EDITOR, GAME DESIGNER

OUTPUT EXAMPLES

Character models, textured props, environment kits, asset packs, render-ready scenes, rigging-ready meshes

GRAPHIC DESIGNER

Creates visual materials for branding, marketing, UI, and in-studio communication. The GRAPHIC DESIGNER ensures clarity, consistency, and aesthetic quality across all visual communication, from vector graphics to complete layouts.

RESPONSIBILITIES

- Design logos, icons, social graphics, posters, and pitch visuals
- Develop clean and readable UI layouts or mockups
- Create branded materials (PDFs, decks, social post templates)
- Design vector graphics and scalable visual elements
- Collaborate with *COPYWRITERS* and *MARKETERS* to unify visual messaging
- Maintain and evolve the Studio's visual identity system

EXPECTATIONS

- Good understanding of layout, typography, and color theory
- Proficient in tools such as Adobe Illustrator, Figma, or Affinity Designer
- Able to create scalable, clear vector assets and UI icons
- Understands branding principles and user-centered design
- Open to creative feedback and collaborative iteration

WORKS CLOSELY WITH

MARKETER, COPYWRITER, UI DEVELOPER, PROJECT MANAGER

OUTPUT EXAMPLES

Vector icons, logos, social templates, UI mockups, pitch decks, brand assets, web graphics

CINEMATOGRAPHER

Responsible for visual storytelling through framing, composition, and movement. The CINEMATOGRAPHER plans and captures both still and motion visuals, often working inside digital environments or using photography and video tools.

RESPONSIBILITIES

- Frame scenes, set camera angles, and define lighting setups
- Direct visual tone in cutscenes or cinematic shots
- Capture photo or video content for references or promotional use
- Work with *STORYBOARD ARTISTS* to pre-visualize sequences
- Support *VIDEO EDITORS* and *ANIMATORS* with visual guidance

EXPECTATIONS

- Good eye for composition, color, and lighting
- Experience with either virtual cameras (e.g. Blender, Unreal) or real cameras
- Understands the emotional and narrative impact of visuals
- Able to shoot or simulate reference imagery
- Thinks in motion and shot progression

WORKS CLOSELY WITH

STORYBOARD ARTIST, VIDEO EDITOR, 3D ARTIST, MARKETER, SCREENWRITER

OUTPUT EXAMPLES

Cinematic reference shots, still photography, camera layouts, trailer framing plans

STORYBOARD ARTIST

Translates scripts into sequential visual plans. STORYBOARD ARTISTS break down narrative scenes into readable panels that guide animation, cinematics, or interactive sequences.

RESPONSIBILITIES

- Sketch sequential visual plans for cutscenes or trailers
- Translate narrative beats into clearly framed compositions
- Work with *SCREENWRITERS* and *CINEMATOGRAPHERS* to establish visual flow
- Help communicate timing, emotion, and pacing through thumbnails
- Iterate based on feedback and production needs

EXPECTATIONS

- Able to sketch quickly and clearly
- Understands visual storytelling, camera angles, and pacing
- Good grasp of cinematic language and emotional cues
- Works from scripts and contributes ideas visually
- Familiar with tools like Storyboarder, Photoshop, or traditional sketching

WORKS CLOSELY WITH

SCREENWRITER, VIDEO EDITOR, CINEMATOGRAPHER, NARRATIVE DESIGNER

OUTPUT EXAMPLES

Storyboard panels, animatics, shot sequences, visual breakdowns

TECHNICAL

Focused on game systems, logic, mechanics, and interactive structures.

GAME DESIGNER

Designs how the game is played, from mechanics to systems to the player's moment-to-moment experience. The GAME DESIGNER balances structure, challenge, and creativity to ensure gameplay feels intentional, engaging, and meaningful.

RESPONSIBILITIES

- Design and document core mechanics, features, and player progression
- Build systems for inventory, dialogue, exploration, combat, or resource management
- Collaborate with *PROGRAMMERS* and *NARRATIVE DESIGNERS* to unify systems and story
- Playtest mechanics and adjust for clarity, pacing, and fun
- Create design bibles, wireframes, or mockups for gameplay systems

EXPECTATIONS

- Good understanding of player psychology and interactive feedback
- Able to balance depth with simplicity in mechanics
- Familiar with tools like Miro, Notion, spreadsheets, or prototyping platforms
- Analytical and iterative mindset, constantly refining
- Communicates ideas clearly with both creative and technical teams

WORKS CLOSELY WITH

PROGRAMMER, SYSTEMS DESIGNER, NARRATIVE DESIGNER, PROJECT MANAGER, IDEATOR

OUTPUT EXAMPLES

Mechanic flowcharts, gameplay mockups, progression systems, design docs, stat balancing charts

PROGRAMMER

Builds the technical foundation of the project, translating ideas into functional systems. The PROGRAMMER brings interactivity to life by writing, testing, and maintaining code across the game or platform.

RESPONSIBILITIES

- Write and optimize scripts or code for gameplay systems
- Implement mechanics, menus, dialogue systems, or logic triggers
- Debug and maintain stability across builds
- Collaborate with DESIGNERS and ARTISTS to bring ideas to life
- Document technical workflows and tool usage

EXPECTATIONS

- Proficient in relevant programming languages (e.g. C#, Python, Lua)
- Familiar with the Studio's game engine(s) or framework
- Able to debug and troubleshoot efficiently
- Understands performance optimization and clean code practices
- Communicates technical possibilities and constraints clearly

WORKS CLOSELY WITH

GAME DESIGNER, TECHNICAL ARTIST, SYSTEMS DESIGNER, UI DEVELOPER, PROJECT MANAGER

OUTPUT EXAMPLES

Code scripts, working prototypes, gameplay functions, in-game systems, bugfix reports, build releases

SYSTEMS DESIGNER

Architects the deep, underlying structures that govern how a game works at scale. The **SYSTEMS DESIGNER** builds mechanics that interconnect, from progression and economy to AI behavior and simulation logic, ensuring the game feels responsive, balanced, and alive. They don't just design a mechanic; they design how all mechanics work together.

RESPONSIBILITIES

- Design complex game systems such as leveling, progression, crafting, economy, or AI
- Define how systems interact, evolve, and respond to player behavior
- Collaborate with *PROGRAMMERS* to bring systems into functionality
- Build balancing models, simulations, and system logic flowcharts
- Playtest and adjust systems for fairness, feedback, and clarity

EXPECTATIONS

- Proficient understanding of system design principles and player motivation
- Able to think modularly, building mechanics that scale and adapt
- Proficient with spreadsheets, flow diagrams, and balancing tools
- Comfortable working with both creative teams and technical staff
- Analytical mindset with a love for systems thinking and iteration

WORKS CLOSELY WITH

PROGRAMMER, GAME DESIGNER, PROJECT MANAGER, TECHNICAL ARTIST, IDEATOR

OUTPUT EXAMPLES

Progression trees, economy spreadsheets, system flowcharts, balancing docs, mechanic breakdowns, simulation prototypes

UI DEVELOPER

Designs and builds user interfaces that are both visually sharp and functionally smooth. The UI DEVELOPER handles the whole process of creating interfaces, from layout design and prototyping to front-end development, ensuring HELLINGTON's digital experiences feel good to use and look good doing it.

RESPONSIBILITIES

- Design clean, intuitive, and accessible user interfaces
- Translate wireframes and mockups into responsive front-end code
- Develop interactive features (navigation, transitions, hover states, etc.)
- Collaborate with *GRAPHIC DESIGNERS* and *WRITERS* to refine layout and user flow
- Test and optimize interfaces across browsers and devices

EXPECTATIONS

- Preferably proficient in HTML, CSS, and JavaScript (React, Vue, or GSAP is a plus)
- Strong eye for visual hierarchy, layout, and interaction patterns
- Comfortable designing in Figma or Adobe XD
- Understands UX principles, accessibility basics, and responsive design
- Can work independently from concept to polished implementation

WORKS CLOSELY WITH

GRAPHIC DESIGNER, MARKETER, COPYWRITER, PROJECT MANAGER

OUTPUT EXAMPLES

Landing pages, UI mockups, front-end components, interactive embeds, mobile layouts, responsive site sections

TECHNICAL VISUALS

Here is where it gets awkward: these roles combine visual and technical skill sets. We didn't exactly know how else to categorize them ...

2D ANIMATOR

Brings visuals to life through frame-by-frame or rigged animation. The 2D ANIMATOR adds movement, emotion, and rhythm to illustrations, UI, and cinematic elements, blending storytelling with technical skill in motion.

RESPONSIBILITIES

- Animate characters, UI, and scene elements in 2D
- Create smooth loops, expressive actions, or stylized motion sequences
- Work from STORYBOARDS or KEYFRAMES to animate scenes
- Prepare assets for implementation in-game or in videos
- Optimize animation for performance, clarity, and mood

EXPECTATIONS

- Proficient in 2D animation tools (e.g. Toon Boom, After Effects, Adobe Animate)
- Strong sense of timing, motion, and visual storytelling
- Understands animation principles (squash/stretch, easing, anticipation, etc.)
- Can animate within existing art styles
- Works well with *VIDEO EDITOR*, *DESIGNERS*, and *PROGRAMMERS*

WORKS CLOSELY WITH

2D ARTIST, STORYBOARD ARTIST, VIDEO EDITOR, GAME DESIGNER, TECHNICAL ARTIST

OUTPUT EXAMPLES

Animated UI elements, motion loops, cutscene animation, gameplay spritesheets, layered animation sequences

3D ANIMATOR

Animates 3D characters, props, and cameras for in-game and cinematic content. The 3D ANIMATOR focuses on performance, movement, and expression within a three-dimensional space.

RESPONSIBILITIES

- Rig and animate 3D models (characters, objects, cameras)
- Develop action sequences, idle loops, or dialogue performances
- Match animation to scene pacing, sound, and intent
- Collaborate with *CINEMATOGRAPHERS*, *VIDEO EDITORS*, and *VOICE ACTORS*
- Export and implement animations into engine or cinematic pipeline

EXPECTATIONS

- Proficient in tools like Blender, Maya, or similar 3D animation software
- Strong understanding of body mechanics, weight, and timing
- Able to animate both gameplay and narrative-focused actions
- Familiar with rigging, skinning, and animation export workflows
- Communicates closely with *TECHNICAL ARTISTS* for smooth integration

WORKS CLOSELY WITH

3D ARTIST, TECHNICAL ARTIST, VIDEO EDITOR, GAME DESIGNER, CINEMATOGRAPHER

OUTPUT EXAMPLES

Character animations, facial performances, prop/camera animation, dialogue syncs, cutscene motion files

VIDEO EDITOR

Cuts, assembles, and polishes video content for trailers, dev logs, in-game cinematics, and promotional media. The VIDEO EDITOR shapes raw assets, visuals, sound, voice, and motion into cohesive, emotionally timed sequences that match the Studio's tone and intent.

RESPONSIBILITIES

- Edit footage into structured, impactful video content
- Sync visuals with sound effects, music, and voiceover
- Apply transitions, cuts, pacing, and color grading
- Collaborate with *SCREENWRITERS* and *COMPOSERS* to maintain narrative rhythm
- Optimize exports for different platforms (web, social, game engine)

EXPECTATIONS

- Proficient in editing software (e.g. Adobe Premiere, DaVinci Resolve, Final Cut Pro)
- Strong storytelling sense through timing and rhythm
- Understands tone, pacing, and emotional flow in visual sequences
- Familiar with audio editing and basic VFX integration
- Able to work fast and iterate based on feedback

WORKS CLOSELY WITH

SCREENWRITER, CINEMATOGRAPHER, SOUND DESIGNER, COMPOSER, MARKETER

OUTPUT EXAMPLES

Trailers, dev log videos, social media clips, cutscene edits, teaser reels, internal sizzle edits

SPECIAL EFFECTS ARTIST

Creates visual effects that enhance atmosphere, storytelling, and impact, from particle systems to motion graphics and stylized in-game effects. The SPECIAL EFFECTS ARTIST combines technical tools with visual flair to amplify emotion, energy, and immersion.

RESPONSIBILITIES

- Design and implement visual effects for gameplay, cutscenes, or promotional media
- Create effects such as smoke, fire, light bursts, UI flares, or glitch effects
- Animate textures, particle systems, shaders, or overlays
- Work with *VIDEO EDITORS* and *ANIMATORS* to time effects to music or motion
- Optimize effects for clarity and performance

EXPECTATIONS

- Skilled in VFX tools such as After Effects, Blender, Unity/Unreal VFX editors, or shader graphs
- Understands timing, composition, and how effects support storytelling
- Capable of both technical implementation and creative direction
- Familiar with layering effects into video, UI, or in-game sequences
- Able to work independently or as a finishing touch collaborator

WORKS CLOSELY WITH

VIDEO EDITOR, 2D ANIMATOR & 3D ANIMATOR, TECHNICAL ARTIST, COMPOSER, GAME DESIGNER

OUTPUT EXAMPLES

Cutscene effects, VFX overlays, stylized transitions, gameplay particle systems, motion graphics sequences

TECHNICAL ARTIST

Acts as the bridge between *ARTISTS* and *PROGRAMMERS*, ensuring that visual assets are both beautiful and functional. The *TECHNICAL ARTIST* handles asset integration, optimization, tool development, and workflow automation, making sure the creative vision translates smoothly into production-ready results.

RESPONSIBILITIES

- Optimize 2D/3D assets for performance and compatibility
- Assist with shader development, lighting setups, and material systems
- Develop or streamline pipelines and tools for artists
- Troubleshoot visual bugs or pipeline issues between departments
- Support integration of assets into engines or animation software

EXPECTATIONS

- Familiar with game engines (e.g. Unity, Unreal) and 3D asset pipelines
- Understanding of shaders, lighting, rigging, or VFX pipelines
- Able to write scripts/tools (e.g. Python, MEL, custom exporters) to support workflows
- Communicates well across technical and visual teams
- Balances creative standards with technical constraints

WORKS CLOSELY WITH

3D ARTIST, PROGRAMMER, ANIMATOR, VIDEO EDITOR, GAME DESIGNER

OUTPUT EXAMPLES

Shader setups, optimized textures, tool scripts, asset integration checklists, visual debug passes, rigging pipelines

SOUND

Focused on the audio identity of a project, including music, sound design, and voice.

COMPOSER

Creates original music using a DAW to support the emotional and narrative feel of a project. The COMPOSER translates story beats, themes, and gameplay moments into sound, building tracks that help shape mood, pacing, and atmosphere.

RESPONSIBILITIES

- Compose original soundtracks, stingers, and loops
- Build songs or themes that fit scenes, gameplay, or cutscenes
- Align music with pacing, tone, and emotion
- Collaborate with *SOUND DESIGNERS* and *VIDEO EDITORS* to sync music with visuals
- Deliver stems and mixes for layering, editing, or implementation

EXPECTATIONS

- Skilled in music composition, arrangement, and digital production
- Proficient with DAWs (e.g. FL Studio, Logic Pro, Reaper)
- Able to match genre, tone, and tempo to project needs
- Basic knowledge of how music is used in games, trailers, or scenes

WORKS CLOSELY WITH

VIDEO EDITOR, SOUND DESIGNER, SCREENWRITER, PROJECT MANAGER

OUTPUT EXAMPLES

Original soundtrack cues, mood loops, trailer music, ambient tracks, cue sheet breakdowns

SOUND DESIGNER

Designs the non-musical sound world of the project, from footsteps and UI clicks to explosions, eerie atmospheres, and glitch effects. The SOUND DESIGNER adds texture, realism, and emotional depth through custom-created or manipulated audio.

RESPONSIBILITIES

- Create and implement sound effects for environments, characters, UI, and mechanics
- Design layered, reactive soundscapes for specific scenes or actions
- Record or source original sounds (Foley, SFX libraries, etc.)
- Collaborate with *COMPOSERS* and *DEVELOPERS* to ensure timing and context
- Prepare audio for implementation in engine or post-production

EXPECTATIONS

- Proficient in audio editing and sound layering
- Familiar with DAWs (e.g. Reaper, Pro Tools, Audition) and middleware (FMOD, Wwise)
- Understands spatial audio, timing, and scene dynamics
- Has an ear for subtlety and creative problem-solving
- Organized with naming, versioning, and layering

WORKS CLOSELY WITH

COMPOSER, PROGRAMMER, VIDEO EDITOR, GAME DESIGNER, TECHNICAL ARTIST

OUTPUT EXAMPLES

Footstep variations, UI click sets, ambient layers, cinematic sound beds, SFX packs

AUDIO ENGINEER

Mixes, masters, and refines all audio content to professional standards. The AUDIO ENGINEER ensures that everything, from music to SFX to voice, is clear, balanced, and immersive across different playback formats.

RESPONSIBILITIES

- Mix dialogue, music, and effects for balance and clarity
- Master audio for use in trailers, in-game, or online content
- Remove noise, normalize levels, and apply EQ/compression
- Prepare final exports for different platforms and file types
- Advise Team on audio quality, loudness, and consistency

EXPECTATIONS

- Skilled in mixing/mastering within a DAW (e.g. Pro Tools, Logic, Reaper)
- Understands dynamic range, stereo imaging, and audio balance
- Pays obsessive attention to sound clarity and consistency
- Comfortable giving technical feedback to *COMPOSERS* or *VOICE ACTORS*
- Familiar with loudness standards for games, trailers, and web

WORKS CLOSELY WITH

COMPOSER, SOUND DESIGNER, VIDEO EDITOR, VOICE ACTOR

OUTPUT EXAMPLES

Final trailer mixes, balanced cutscene audio, cleaned voice files, audio mastering reports

VOICE ACTOR

Performs characters or narration for scripted content. The VOICE ACTOR brings emotion, presence, and personality to characters and storytelling through voice alone.

RESPONSIBILITIES

- Record character dialogue or narrator voice lines
- Match tone, emotion, and pacing to scene needs
- Perform directed takes based on feedback
- Deliver clean, high-quality audio files with consistent levels
- Collaborate on pronunciation, timing, and tone with *WRITERS* or *VIDEO EDITORS*

EXPECTATIONS

- Strong vocal performance range and emotional flexibility
- Access to a clear recording setup (or willing to travel for sessions)
- Follows scripts, feedback, and performance direction
- Understands pacing, breathing, and delivery for scenes
- Able to re-record or provide alt takes when needed

WORKS CLOSELY WITH

SCREENWRITER, COMPOSER, VIDEO EDITOR, AUDIO ENGINEER

OUTPUT EXAMPLES

Character voice lines, narration tracks, alternate takes, pronunciation guides, VO session folders

OTHER

Studio-wide or flexible roles that support ideation, research, and outreach.

IDEATOR

The creative fuel of the Studio. The IDEATOR generates concepts, twists, solutions, and story sparks across all disciplines. They don't just come up with ideas; they provoke direction, push innovation, and help refine creative decisions with fresh thinking.

RESPONSIBILITIES

- Contribute raw ideas during brainstorming sessions, pitch sessions, or planning phases
- Support other roles by suggesting narrative, visual, or design concepts
- Challenge existing plans with new perspectives or more imaginative options
- Assist in naming, phrasing, or tone-setting across creative output
- Help define the “why” and “what if” of any project

EXPECTATIONS

- Highly imaginative, with a wide range of references and intuition
- Willing to collaborate, build on others' ideas, and shift when needed
- Not afraid to throw wild ideas on the table (even if some fail)
- Capable of articulating why an idea works (or doesn't)
- Understands HELLINGTON's voice, tone, and Mission

WORKS CLOSELY WITH

EVERYONE, especially *WRITERS, GAME DESIGNER, PROJECT MANAGER, MARKETER*

OUTPUT EXAMPLES

Idea lists, concept prompts, story hooks, alternate mechanics, brainstorm boards, creative direction notes

RESEARCHER

Digs deep to bring authenticity, context, and creative grounding to a project. The RESEARCHER provides references, examples, and real-world insight that support ideation and development, making work richer, smarter, and more informed.

RESPONSIBILITIES

- Collect references for worldbuilding, visuals, tone, and themes
- Gather data or inspiration based on creative briefs or prompts
- Summarize findings in a clear, organized format for Team use
- Fact-check and cross-reference lore, names, designs, or systems
- Suggest content backed by historical, stylistic, or cultural depth

EXPECTATIONS

- Loves to dig, browse, and follow rabbit holes
- Skilled at sourcing relevant, accurate, and diverse material
- Understands how to filter info for creative usefulness
- Organized in sharing findings and citations
- Adds depth without overwhelming direction

WORKS CLOSELY WITH

IDEATOR, STORYWRITER, NARRATIVE DESIGNER, MARKETER

OUTPUT EXAMPLES

Reference docs, image boards, annotated research decks, name lists, mechanics studies, inspiration libraries

MARKETER

Shapes how the Studio presents itself to the world, from project announcements to long-term audience engagement. The MARKETER is responsible for how we're seen, heard, and remembered beyond development.

RESPONSIBILITIES

- Plan and execute marketing strategies for projects, updates, or launches
- Write or coordinate social media posts, announcements, or visual assets
- Analyze audience response and adjust communication accordingly
- Work with DESIGNERS and COPYWRITERS to align visuals and messaging
- Help build community interest through previews, newsletters, or web content

EXPECTATIONS

- Strategic thinker with creative instincts
- Comfortable with planning content across platforms
- Understands branding, audience targeting, and platform tone
- Can collaborate across departments and work with deadlines
- Has a pulse on trends, audience behavior, and digital presence

WORKS CLOSELY WITH

COPYWRITER, GRAPHIC DESIGNER, UI DEVELOPER, PROJECT MANAGER, IDEATOR

OUTPUT EXAMPLES

Campaign plans, social media calendars, Discord announcements, blog post briefs, launch strategies, engagement reports

MANAGEMENT ROLES

HELLINGTON STUDIO thrives on creative chaos, but someone's gotta manage it. That's where Managers come in. Management roles exist to support, coordinate, and guide, not to dominate. While every member contributes creatively, Managers take on the responsibility of directing their department, helping others stay aligned, supported, and on track.

Managers are expected to:

- Maintain momentum
- Uphold quality
- Ensure clear communication
- Create structure without killing the passion

Some Managers lead creative departments (like *VISUALS* or *SOUND*), while others manage Studio functions (like *MARKETING* or *PROJECT MANAGEMENT*). Regardless of focus, every Manager plays a key role in ensuring the Studio runs smoothly, so creatives can continue to create without too many complications.

This section defines each Manager role, outlines their responsibilities, and explains how they support the Team.

(illustration here)

CREATIVE MANAGERS

Creative Managers oversee the departments responsible for producing the Studio's core content. Their role is to provide structure, feedback, and direction, ensuring that creative work is consistent, high-quality, and aligned with the project's vision.

They support contributors, help manage deadlines, and act as a bridge between Team members and Studio leadership.

WRITING MANAGER

Leads the *WRITING* department, overseeing story, dialogue, structure, copy, and tone. The *WRITING MANAGER* helps keep narrative vision cohesive while managing the workload of multiple *WRITERS* and *EDITORS*.

RESPONSIBILITIES

- Assign and track writing tasks across the Team
- Review written work and ensure tone and quality consistency
- Align story elements across different mediums (scripts, UI, promo)
- Support *WRITERS* during revisions or collaborative scenes
- Facilitate writing brainstorming or feedback

EXPECTATIONS

- Strong editing and narrative oversight ability
- Excellent communicator with a feel for voice and flow
- Able to structure deadlines around different writing types
- Encourages collaboration while maintaining vision clarity
- Keeps written work aligned with gameplay and Studio tone

WORKS CLOSELY WITH

NARRATIVE DESIGNER, STORYWRITER, SCREENWRITER, COPYWRITER, EDITOR, PROJECT MANAGER, LEAD

OUTPUT EXAMPLES

Story structure maps, Team feedback docs, task assignments, tone guidelines

VISUAL MANAGER

Oversees the *VISUALS* department, supporting all artists and designers working with 2D, 3D, storyboards, cinematography, and visual branding. The **VISUAL MANAGER** ensures consistency across styles, deadlines, and communication, translating creative goals into actionable plans for the Team.

RESPONSIBILITIES

- Coordinate tasks and timelines for *2D ARTISTS*, *3D ARTISTS*, and *DESIGNERS*
- Maintain style consistency and visual standards across projects
- Review visual work and give constructive feedback
- Communicate with the *LEAD* and *PROJECT MANAGER* about Team progress
- Support contributors with resources, references, and clarity

EXPECTATIONS

- Deep understanding of visual workflows (illustration, modeling, layout, etc.)
- Strong visual taste and ability to give actionable critique
- Able to manage timelines without stifling creativity
- Keeps the Team creatively motivated and aligned
- Communicates direction, references, and deadlines

WORKS CLOSELY WITH

2D ARTIST & 3D ARTIST, GRAPHIC DESIGNER, CINEMATOGRAPHER, PROJECT MANAGER, LEAD

OUTPUT EXAMPLES

Visual asset tracking boards, style guide updates, feedback notes, visual review checklists

TECHNICAL MANAGER

Oversees the *TECHNICAL* Team, including *PROGRAMMERS*, *GAME DESIGNERS*, *SYSTEMS DESIGNERS*, and *TECHNICAL ARTISTS*. The *TECHNICAL MANAGER* ensures system stability, workflow clarity, and alignment between creative and code.

RESPONSIBILITIES

- Manage technical task schedules and system documentation
- Facilitate communication between dev and non-dev teams
- Review gameplay features, mechanics, and systems
- Identify technical bottlenecks or missing resources
- Ensure assets and codebases are clean, optimized, and accessible

EXPECTATIONS

- Deep familiarity with technical pipelines and problem-solving
- Able to translate technical needs into studio-wide priorities
- Strong understanding of version control, builds, and QA
- Clear communicator between technical and creative sides
- Works proactively to keep systems stable and evolving

WORKS CLOSELY WITH

PROGRAMMER, *GAME DESIGNER*, *TECHNICAL ARTIST*, *PROJECT MANAGER*, *LEAD*

OUTPUT EXAMPLES

Sprint schedules, bug report lists, system checklists, optimization guidelines, dev documentation

SOUND MANAGER

Leads the *SOUND* department and ensures that all audio, from music to sound effects to voiceovers, is cohesive, high-quality, and delivered on time. The **SOUND MANAGER** coordinates audio contributors, provides direction, and ensures audio supports the overall emotional and narrative goals of the project.

RESPONSIBILITIES

- Assign and oversee tasks for *COMPOSERS*, *SOUND DESIGNERS*, *VOICE ACTORS*, and *AUDIO ENGINEERS*
- Review and give feedback on sound and voice assets
- Coordinate recording schedules and implementation needs
- Ensure audio aligns with gameplay, tone, and cinematic timing
- Manage version control and file organization for audio assets

EXPECTATIONS

- Deep understanding of audio production for games and visual media
- Clear communication with both creative and technical collaborators
- Ability to give feedback on sound in terms of emotion, clarity, and timing
- Keeps the audio pipeline running efficiently and cleanly
- Ensures assets are named, exported, and formatted correctly

WORKS CLOSELY WITH

COMPOSER, *SOUND DESIGNER*, *AUDIO ENGINEER*, *VOICE ACTOR*, *VIDEO EDITOR*, *PROJECT MANAGER*

OUTPUT EXAMPLES

Audio implementation lists, feedback summaries, mixing notes, recording schedules, naming conventions

IDEATION MANAGER

Facilitates structured creativity across all departments. The IDEATION MANAGER leads brainstorming, idea curation, and early concept development. They help transform raw ideas into usable formats and guide the Team toward bold yet achievable creative decisions.

RESPONSIBILITIES

- Lead ideation sessions and creative sprints across departments
- Curate and refine raw ideas from *IDEATORS* or contributors
- Help teams develop their concepts into clear, actionable formats
- Organize and archive ideas for later use or reference
- Foster a creative culture where all members feel free to contribute

EXPECTATIONS

- Big-picture thinker with strong taste and communication skills
- Good at identifying high-potential ideas early
- Able to guide creators without dominating their voice
- Keeps track of creative momentum and ensures follow-through
- Comfortable switching between abstract thinking and practical formatting

WORKS CLOSELY WITH

IDEATOR, PROJECT MANAGER, WRITING MANAGER, VISUAL MANAGER, LEAD

OUTPUT EXAMPLES

Idea tracking boards, brainstorm summaries, pitch outlines, concept briefs, Studio creative archives

STUDIO FUNCTION MANAGERS

Studio Function Managers focus on the operational side of the Studio. They handle planning, communication, organization, outreach, and long-term sustainability.

Their job is to make sure that projects stay on schedule, documentation is accessible, the public-facing voice is consistent, and the Studio runs efficiently from the inside out.

PROJECT MANAGER

Keeps the entire Studio running smoothly. The PROJECT MANAGER ensures that tasks are clear, timelines are realistic, and communication between departments is steady and structured.

RESPONSIBILITIES

- Track all ongoing tasks, deadlines, and dependencies
- Create and maintain project timelines, calendars, and boards
- Organize Team meetings, sprint cycles, and checkpoints
- Act as the bridge between *LEAD*, *MANAGERS* and *CONTRIBUTORS*
- Monitor workload distribution and flag potential bottlenecks

EXPECTATIONS

- Highly organized and detail-oriented
- Skilled with task management tools (Notion, Trello, Asana, etc.)
- Clear, respectful communicator with all departments
- Able to anticipate blockers and suggest workflow improvements
- Keeps documentation clean, visible, and updated

WORKS CLOSELY WITH

LEAD, *ALL MANAGERS*, *CONTRIBUTORS* across departments

OUTPUT EXAMPLES

Production calendars, sprint boards, status reports, meeting notes, pipeline charts

MARKETING MANAGER

Oversees the planning, execution, and coordination of the Studio's marketing efforts. The MARKETING MANAGER aligns internal creative work with external visibility, helping the Studio reach the right audience at the right time, in the right tone.

RESPONSIBILITIES

- Plan marketing campaigns for project milestones, releases, or events
- Manage social content pipelines, schedules, and strategies
- Collaborate with *COPYWRITERS* and *GRAPHIC DESIGNERS* to execute campaigns
- Track analytics and engagement for continuous improvement
- Ensure all public-facing content aligns with the HELLINGTON brand

EXPECTATIONS

- Strategic mindset with a strong grasp of visual and written communication
- Familiar with social platforms, content trends, and digital storytelling
- Able to coordinate between multiple contributors on a schedule
- Comfortable analyzing data and making audience-focused decisions
- Confident in maintaining the studio's voice and presence

WORKS CLOSELY WITH

COPYWRITER, GRAPHIC DESIGNER, UI DEVELOPER, IDEATOR, PROJECT MANAGER

OUTPUT EXAMPLES

Marketing campaign calendars, asset briefs, platform plans, launch checklists, analytics reports

DOCUMENTATION MANAGER

Maintains the knowledge base of the Studio. The DOCUMENTATION MANAGER ensures that all relevant guidelines, systems, workflows, and Studio lore are organized, up-to-date, and easily accessible for the Team.

RESPONSIBILITIES

- Organize and maintain internal documentation (Notion, Google Drive, etc.)
- Ensure all departments have access to updated guides and references
- Archive completed projects, documents, and production artifacts
- Collaborate with *MANAGERS* to document workflows and best practices
- Create templates and standards for easy internal use

EXPECTATIONS

- Highly organized and documentation-focused
- Able to translate abstract workflows into clear, readable formats
- Familiar with collaborative documentation tools
- Committed to clarity, accessibility, and long-term usability
- Acts as a quiet force keeping the Studio from falling into chaos

WORKS CLOSELY WITH

PROJECT MANAGER, ALL MANAGERS, NEW MEMBERS, ADVISOR

OUTPUT EXAMPLES

Role guides, process docs, onboarding kits, Team templates, project archives

COMMUNITY MANAGER

Manages the Studio's interaction with its audience. The COMMUNITY MANAGER is the bridge between internal creative work and public reception, maintaining a healthy, engaged, and informed community around HELLINGTON STUDIO's projects, platforms, and personality.

RESPONSIBILITIES

- Moderate and manage public platforms (e.g. Discord, social media, forums)
- Share project updates, news, and events in accessible, friendly formats
- Respond to feedback, questions, and interactions with authenticity
- Organize public-facing events (Q&As, launch streams, community polls)
- Provide internal teams with insight on community sentiment and behavior

EXPECTATIONS

- Approachable, calm, and professional communication style
- Deep understanding of online culture and platform etiquette
- Can maintain Studio tone across written and live interactions
- Organized in scheduling community updates and announcements
- Responsive and empathetic, without overpromising or overexposing the Team

WORKS CLOSELY WITH

MARKETER, COPYWRITER, PROJECT MANAGER, LEAD

OUTPUT EXAMPLES

Community updates, announcement posts, moderation logs, community feedback reports, event plans

FINANCE MANAGER

Oversees the Studio's budgeting, expenses, and financial planning. The FINANCE MANAGER ensures that creative vision remains sustainable, helping balance ambition with resources and guiding the Team through cost-aware production decisions.

RESPONSIBILITIES

- Track project expenses, tools/subscriptions, and funding allocations
- Maintain up-to-date budgets and financial logs
- Advise on cost-efficient production decisions and timelines
- Collaborate on pricing strategy for paid releases (if applicable)
- Prepare financial reports or breakdowns for transparency

EXPECTATIONS

- Organized and detail-oriented with financial tracking
- Familiar with spreadsheets, invoicing, and digital tools
- Understands how to manage budgets for creative production
- Clear communicator, especially when flagging risks or limits
- Helps the Team think long-term while staying financially grounded

WORKS CLOSELY WITH

LEAD, PROJECT MANAGER, MARKETING MANAGER, TOOL-DEPENDENT ROLES

OUTPUT EXAMPLES

Budget breakdowns, subscription audits, financial updates, cost-saving proposals, grant tracking sheets

TOOLS

TOOLS

This chapter covers the tools we use or officially approve for Studio projects. If you have a tool you prefer using, simply ask. If it meets our needs and maintains high quality, we'll likely approve it. We might even add it to our list once we've properly looked into what it offers.

We aim to be flexible, but compatibility and quality still matter. The tools we use reflect the kind of work we want to create and the standards we aim to maintain.

And yes, we know that any tool can be great if the person using it knows what they're doing. This list isn't about limiting you. It's about staying organized, collaborative, and consistent across the Team. Our primary concern is the limitations and compatibility of the tools we use.

You can click on the headings of the Tools to be redirected to their site.

(illustration here)

COMMUNICATION

This section covers the tools we use to talk, meet, and stay connected. Whether it's a quick message, a weekly check-in, or a big project update, these are the platforms we rely on to keep the conversation moving.

(illustration here)

DISCORD

Discord is our main communication hub. It's where we chat, call, share files, and keep everyone up to date on everything from project progress and training to random ideas. We also use Discord for all our Team meetings, whether it's a simple check-in or a full production call. It's simple, flexible, and keeps the Studio connected.

COST

FREE

- Optional Nitro provides higher upload limits and profile perks, but it's not necessary for work

USE

Messaging, voice meetings, sharing feedback, file drops, project updates, casual conversations and overall most communication.

ROLES

EVERYONE

GMAIL

Gmail is what we use when communication needs to be more formal or external. It's good for reaching out to collaborators, handling business-related messages, or submitting things that require a bit more professionalism. It also comes in handy for file sharing, especially when Discord's limits aren't enough. Many file delivery services utilize Gmail to send or verify files, making it useful for that purpose as well.

COST

FREE

USE

External communication, formal messages, and sharing larger files.

ROLES

Used occasionally by most members.

Most relevant for:

- **LEAD** - Formal communication and follow-ups
- **PROJECT MANAGER** - Sending files and keeping things organized
- **DOCUMENTATION MANAGER & MARKETING MANAGER** - Delivering drafts or final assets
- **MEMBER** - Sharing files when needed

GOOGLE CALENDAR

Google Calendar is what we use to keep track of meetings, deadlines, and other important dates. It helps make sure we're all on the same page about when things are happening, especially for Team check-ins, production calls, and major milestones. You'll mostly use it to know when you're expected to show up for something or when something's due.

COST

FREE

USE

Scheduling meetings, setting deadlines, and viewing Studio-wide timelines.

ROLES

Used by everyone, mostly to stay informed.

Managed more actively by:

- **PROJECT MANAGER** - Scheduling meetings and setting timelines
- **LEAD & MANAGERS** - Planning check-ins or milestone reviews
- **MEMBER** - Checking in on meeting times and key deadlines

ORGANIZING & MANAGEMENT

These tools help us keep track of everything from files and tasks to schedules and documents. They're the backbone of our workflow, ensuring that nothing gets lost, forgotten, or buried amidst the chaos.

(illustration here)

GOOGLE DRIVE

Google Drive is where we store all our files. Every file, doc, asset, and template lives here in a structured folder system. It's our shared digital archive; if it's part of a project, it should be in Drive. We use it to stay organized, avoid losing things, and ensure everyone is working from the same version of documents.

COST

FREE

- Additional storage is available with a paid plan if ever needed

USE

Storing and sharing files, organizing project folders, keeping resources accessible to the Team.

ROLES

EVERYONE

GOOGLE DOCS

Google Docs is where most of our writing and documentation happens. We use it for scripts, brainstorming, feedback, planning, and especially for logging progress. Every Team Member is expected to log their work regularly, and Docs is where most of that work is stored. Since everything saves automatically and multiple people can work in the same file, it's easy to stay in sync.

COST

FREE

USE

Writing, logging progress, planning, reviewing, and giving feedback.

ROLES

EVERYONE

GOOGLE SLIDES

Google Slides is what we use for presentations. It's great for pitches, project overviews, roadmaps, and any other content that requires a clear visual breakdown. We also use it often during meetings to help guide the conversation, especially when presenting updates, timelines, or ideas to the Team.

COST

FREE

USE

Meeting visuals, pitches, project overviews, and planning decks.

ROLES

Used by a lot of roles depending on the context.

Most often used by:

- **LEAD & IDEATOR** - Presenting concepts and direction
- **PROJECT MANAGER** - Sharing timelines and production plans
- **DOCUMENTATION MANAGER** - Formatting walkthroughs and Studio info
- **MARKETING MANAGER** - Building announcements or campaign decks

GOOGLE SHEETS

Google Sheets is where we track stuff. It's used for things like timelines, budgets, checklists, or anything that benefits from a table or numbers. It's also great for organizing tasks or building simple systems that require regular updates. If it needs structure, Sheets is usually the answer.

COST

FREE

USE

Task tracking, timelines, checklists, budgets, data organization.

ROLES

Used by multiple roles, depending on the project.

Especially useful for:

- **PROJECT MANAGER** - Tracking production schedules and progress
- **FINANCE MANAGER** - Organizing budgets or cost breakdowns
- **SYSTEMS DESIGNER** - Planning game systems or balancing mechanics
- **DOCUMENTATION MANAGER** - Maintaining checklists or resource trackers
- **MANAGER** - Organizing department-specific info

NOTION

Notion is a tool we use for organizing things in a more visual or flexible way. It's excellent for building dashboards, task boards, timelines, and internal wikis. We use it to stay on top of projects, brainstorming sessions, or department-specific information. It's not required for everyone, but if it helps your workflow and the rest of your Team agrees, go for it.

COST

FREE

- Upgrade available for personal accounts, but the free version is more than enough for Studio use

USE

Dashboards, task boards, internal planning, project pages, idea tracking.

ROLES

Optional tool, used more by certain roles.

Most commonly used by:

- **PROJECT MANAGER** - Organizing timelines, sprint boards, or Team check-ins
- **DOCUMENTATION MANAGER** - Building internal guides or hubs
- **IDEATION MANAGER** - Tracking brainstorming sessions or idea libraries
- **MARKETING MANAGER** - Outlining campaigns or schedules

REFERENCES & INSPIRATION

These tools are where ideas begin. Whether you're collecting visual moodboards, saving useful links, or browsing creative prompts, they help spark new directions and keep your brain fueled. Great for getting unstuck, finding a vibe, or just exploring what's possible.

(illustration here)

PINTEREST

Pinterest is where we go for visual inspiration. It's a great place to gather references, whether it's for art styles, environments, UI, fashion, vibes, or mood. It's not a planning tool; it's more of a research and discovery tool. We often use it early in the creative process to spark ideas or define a visual tone before jumping into actual production.

COST

FREE

USE

Style research, reference gathering, mood inspiration, early visual direction.

ROLES

Mostly used during concept and pre-production phases.

Especially useful for:

- **2D ARTIST & 3D ARTIST** - Collecting references for style and form
- **GRAPHIC DESIGNER** - Building branding or UI inspiration boards
- **IDEATOR & VISUAL MANAGER** - Exploring mood or creative direction
- **MARKETING MANAGER** - Finding visual ideas for campaigns or posts

FRAME SET

Frame Set is a visual reference library built for treatments, storyboards, and shot lists. It lets you search through thousands of curated frames from music videos, commercials, and films, filtered by vibe, color, composition, and more. Whether you're building a lookbook or pitching a visual concept, it's a solid tool for finding the right frame to communicate tone, style, or mood.

COST

FREE tier available

- Paid plan unlocks full access, saved sets, and advanced filters

USE

Moodboards, shot lists, visual treatments, cinematic reference.

ROLES

Mostly used by the visual Team.

Especially useful for:

- **CINEMATOGRAPHER** - Studying lighting, shot composition, and camera movement
- **GRAPHIC DESIGNER** - Gathering visual references for treatments or layouts
- **MARKETER** - Pulling cinematic references for campaign visuals
- **VISUAL MANAGER** - Curating lookbooks and reviewing visual tone
- **IDEATOR** - Exploring visual style options to support pitches or concepts

WEB INTERACTIONS GALLERY

Web Interactions Gallery is a curated archive of standout web animations, transitions, and interactive experiences. It's built for creatives looking to push web design beyond the static and into the expressive. You'll find examples of scroll effects, hover states, cursor dynamics, and more, perfect for prototyping ideas or showing devs how a site should feel.

COST

FREE

USE

Website motion reference, animation inspiration, UI interaction examples.

ROLES

Mostly used by the visual and technical Teams.

Especially useful for:

- **GRAPHIC DESIGNER** - Collecting layout or motion ideas for web-based visuals
- **UI DEVELOPER** - Referencing animation behavior and interactive effects
- **VISUAL MANAGER** - Reviewing motion references during concept development
- **IDEATOR** - Finding stylistic cues to pitch a specific vibe or direction

BRAINSTORMING & IDEATION

When we're exploring ideas or building something from scratch, these tools help us map out our thoughts and gather inspiration in a structured way. Perfect for early-stage thinking and creative chaos.

(illustration here)

✉ **MILANOTE**

Milanote is a visual planning tool. It's essentially a digital pinboard where you can drop in images, text, links, and ideas to organize things in a visually appealing way. We use it for early-stage planning, such as moodboards, visual references, or organizing creative thoughts before they're ready for a document or timeline.

COST

FREE plan available

- Paid version unlocks more storage and boards, but most Teams can work fine with the free version

USE

Moodboards, reference collections, visual planning, early concept organization.

ROLES

Used mostly in the early stages of creative work.

Especially useful for:

- **IDEATOR** - Rough concepts, storyboards, visual dumps
- **VISUALS TEAM** - Collecting style references or exploring visual direction
- **MARKETING MANAGER** - Laying out campaign visuals or inspiration
- **LEAD** - Communicating the look and feel of a project

OBSIDIAN

Obsidian is a note-taking tool, but it's built for deeper thinking. It lets you connect notes together, map out ideas, and build a sort of personal web of information. We use it mostly for worldbuilding, lore tracking, and anything that gets complicated or layered. It's especially good when you need to see how everything ties together without losing the big picture.

COST

FREE for personal use

- One-time payment options available for publishing and advanced features, but not necessary for Studio use

USE

Worldbuilding, lore tracking, connected note systems, long-term creative thinking.

ROLES

Used mostly by creative and writing-heavy roles.

Especially useful for:

- **STORYWRITER & NARRATIVE DESIGNER** - Organizing plotlines, characters, lore
- **IDEATOR** - Mapping out ideas and how they connect
- **DOCUMENTATION MANAGER** - Building internal knowledge bases
- **LEAD** - Keeping track of Studio vision or story universes

MIRO

Miro is a digital whiteboard. It's built for brainstorming, mapping things out, and connecting ideas visually. We use it when we need to see everything laid out in one space, like workflows, mind maps, or systems. It's beneficial for game design and development planning, but it also works well for general creative sessions when things are still messy.

COST

FREE plan available

- Has limits on boards and collaborators, paid version unlocks more space, but the free plan usually does the job

USE

Brainstorming, flowcharts, game systems, visual planning, mind maps.

ROLES

Mostly used for mapping out structure-heavy or creative work.

Especially useful for:

- **GAME DESIGNER & SYSTEMS DESIGNER** - Designing mechanics or flowcharts
- **IDEATOR** - Visualizing messy ideas early on
- **PROJECT MANAGER** - Creating production flows or roadmaps
- **NARRATIVE DESIGNER** - Building branching story paths and logic maps

WRITING & LANGUAGE

These tools support the writing process, from finding the right word to refining phrasing, checking tone, or translating meaning. Great for keeping your language sharp, your dialogue clear, and your ideas flowing.

(illustration here)

WORD HIPPO

WordHippo is a flexible language tool that helps with everything from synonyms and antonyms to definitions, rhymes, translations, and sentence examples. It's especially helpful when you're stuck on phrasing, looking to fine-tune your writing, or trying to keep language consistent across a project.

COST

FREE

USE

Synonyms, antonyms, definitions, rhymes, translations, sentence examples.

ROLES

Mostly used by the writing Team.

Especially useful for:

COPYWRITER - Refining tone, phrasing, and clarity in short-form writing

STORYWRITER - Strengthening dialogue and narrative voice

NARRATIVE DESIGNER - Refining character tone and reworking lines

MARKETER - Tightening ad copy, taglines, and campaign writing

IDEATOR - Exploring phrasing to match tone or concept

GRAMMARLY

Grammarly is a writing assistant that helps catch grammar mistakes, spelling errors, awkward phrasing, and tone issues in real time. It's a quick way to clean up drafts, polish final text, or double-check clarity before publishing. Whether you're writing dialogue, documents, or captions, it keeps your writing sharp and readable.

COST

FREE basic version

- Paid plan unlocks tone suggestions, full rewrites, and advanced feedback

USE

Grammar checks, spelling corrections, tone adjustments, phrasing suggestions.

ROLES

Mostly used by the writing Team.

Especially useful for:

- **COPYWRITER** - Cleaning up headlines, captions, and marketing copy
- **STORYWRITER** - Proofreading scenes, dialogue, and prose
- **NARRATIVE DESIGNER** - Tightening in-game text and system prompts
- **MARKETER** - Polishing campaign text and social copy
- **IDEATOR** - Reviewing and editing pitch materials or documentation

LEX

Lex is a minimalist writing tool that helps you move through drafts faster by offering on-the-fly phrasing suggestions, tone tweaks, and autocomplete features. It's designed to assist your writing, not replace it. The goal isn't to generate everything for you, but to help when you hit a wall or want a second opinion without leaving the page.

NOTE: While we generally avoid AI-generated content, we make an exception for Lex because it supports the human writing method instead of automating it. Lex is built for writers who want to stay in control, not outsource the work. It doesn't use your writing for model training, and its privacy policy is stricter than most consumer-facing AI tools. For us, it's a writing assistant, not a shortcut.

COST

FREE with sign-up

- Paid plan (Lex Pro) unlocks premium AI models, longer generations, and early feature access

USE

Drafting, rewriting, AI-powered autocomplete, productivity-focused writing.

ROLES

Mostly used by the writing Team.

Especially useful for:

- **COPYWRITER** - Drafting blurbs, social posts, and marketing copy faster
- **STORYWRITER** - Speeding up scene writing and narrative flow
- **NARRATIVE DESIGNER** - Ideating character lines or branching text
- **MARKETER** - Quickly drafting and adjusting copy tone
- **IDEATOR** - Turning loose thoughts into presentable drafts or outlines

DEVELOPING & CREATION

These are the tools we use to design, build, animate, edit, and create the actual content that makes up our projects, whether it's a game, video, illustration, or sound.

(illustration here)

ADOBE CREATIVE CLOUD

Adobe Creative Cloud is a collection of creative tools we use across the Studio. It includes apps like Photoshop, Illustrator, After Effects, Premiere Pro, and more. If you work with visuals, video, animation, or layout, you'll probably end up using something from this suite. Since it's a cloud-based service, it also provides the option to save your files online, sync settings across devices, and access projects from anywhere, which is particularly helpful for working on different setups.

COST

PAID subscription (monthly or yearly)

- You may already have access through a school or work account. Let the Studio know if that's not the case.

USE

Design, photo editing, video editing, motion graphics, animation, layout, and cloud-based asset syncing.

ROLES

Used across creative departments.

Especially important for:

- **GRAPHIC DESIGNER** - Posters, UI, logos, branding
- **VIDEO EDITOR & SPECIAL EFFECTS ARTIST** - Editing, motion work, VFX
- **2D ARTIST & 2D ANIMATOR** - Illustration and animation
- **MARKETER & MARKETING MANAGER** - Layout design, pitch decks, branded assets

ADOBE AFTER EFFECTS

After Effects is used for motion graphics and visual effects. It's what we use when we want to animate text, create title sequences, add effects to videos, or build complex layered visuals. If it moves and isn't part of gameplay, it probably went through After Effects at some point. It's great for trailers, intros, overlays, and any moment where you want extra visual impact.

COST

PAID subscription (part of Adobe Creative Cloud)

USE

Motion graphics, video overlays, animated titles, transitions, and visual effects.

ROLES

Mostly used by roles focused on video or motion.

Especially useful for:

- **VIDEO EDITOR** - Adding motion elements to cuts
- **SPECIAL EFFECTS ARTIST** - Building layered effects and visuals
- **MARKETER & MARKETING MANAGER** - Creating trailers or social content

ADOBE ILLUSTRATOR

Illustrator is used for creating clean, scalable graphics. It's great for things like logos, icons, UI elements, and any visual that needs to stay sharp at any size. Unlike pixel-based programs like Photoshop, Illustrator works with vectors, which makes it perfect for design work that needs precision and flexibility. It's often used when we're building brand assets or visual systems for projects.

COST

PAID subscription (part of Adobe Creative Cloud)

USE

Logo design, UI elements, icons, vector illustrations, and visual layout.

ROLES

Mostly used by the visual Team.

Especially useful for:

- **GRAPHIC DESIGNER** - Branding, UI, and presentation assets
- **VISUAL MANAGER** - Developing or refining design systems
- **MARKETER & MARKETING MANAGER** - Creating visuals for promotional content
- **2D ARTIST** - Vector art or stylized layouts

ADOBE INDESIGN

InDesign is made for layouts. It's the tool we use when designing anything that feels like a document, pitch decks, studio guides, booklets, magazines, you name it. It's built to handle lots of text and images across multiple pages, so if you're working on something that needs to look clean and consistent from front to back, this is the tool for it.

COST

PAID subscription (part of Adobe Creative Cloud)

USE

Multi-page layouts, pitch decks, visual documents, formatting for print or digital.

ROLES

Used mostly for structured, layout-heavy work.

Especially useful for:

- **GRAPHIC DESIGNER** - Designing pitch decks, magazines, and multi-page visuals
- **LEAD & MARKETING MANAGER** - Preparing presentations or formatted content for sharing

ADOBE LIGHTROOM

Lightroom is used for photo editing and color correction. It's built for adjusting lighting, tone, and mood in photos without destroying the quality. We use it when working with photography or visual assets that need to feel more polished or cinematic. It's especially good for batch editing and keeping a consistent style across a set of images.

COST

PAID subscription (part of Adobe Creative Cloud)

USE

Photo editing, color correction, tone adjustments, batch processing.

ROLES

Used mainly by roles working with photos or promotional visuals.

Especially useful for:

- **GRAPHIC DESIGNER** - Polishing visuals for marketing or branding
- **MARKETER & MARKETING MANAGER** - Editing photos for campaigns
- **CINEMATOGRAPHER & VIDEO EDITOR** - Adjusting stills or reference shots

ADOBE PHOTOSHOP

Photoshop is our go-to for image editing and digital painting. It's one of the most flexible tools in the Creative Cloud, used for everything from quick retouches to full-on illustrations. Whether you're making thumbnails, editing concept art, building mockups, or cutting out assets, this is probably where it's happening.

COST

PAID subscription (part of Adobe Creative Cloud)

USE

Photo editing, concept art, asset creation, mockups, digital painting.

ROLES

Used across a lot of visual roles.

Especially useful for:

- **2D ARTIST** - Painting, asset creation, and illustration
- **GRAPHIC DESIGNER** - Photo edits, mockups, thumbnails
- **DOCUMENTATION MANAGER** - Formatting visuals for guides
- **MARKETER & MARKETING MANAGER** - Editing content for social posts, trailers, or web

ADOBE PREMIERE PRO

Premiere Pro is our main video editing software. It's where we cut trailers, devlogs, cinematics, and any other video content. It handles everything from basic edits to complex timelines with multiple audio and video layers. It also works well with other Adobe apps like After Effects and Photoshop, so you can bring everything together in one place.

COST

PAID subscription (part of Adobe Creative Cloud)

USE

Video editing, trailers, devlogs, cutscenes, and promotional clips.

ROLES

Used mainly by the video Team.

Especially useful for:

- **VIDEO EDITOR** - Cutting and assembling all types of video content
- **SPECIAL EFFECTS ARTIST** - Integrating motion or overlay work
- **MARKETER & MARKETING MANAGER** - Helping create content for campaigns
- **SCREENWRITER & CINEMATOGRAPHER** - Reviewing scene edits and timing

ADOBE XD

Adobe XD is used for designing interfaces and interactive layouts. It's helpful when you want to plan out how something looks and functions before actually building it, like menus, websites, or app screens. You can create mockups and even link things together to show how a user would navigate through it. It's simple, clean, and built for testing ideas fast.

COST

PAID subscription (part of Adobe Creative Cloud)

USE

UI/UX mockups, interactive prototypes, layout previews, web or menu flow planning.

ROLES

Used when designing or previewing interactive interfaces.

Especially useful for:

- **GRAPHIC DESIGNER** - Designing interfaces, layouts, and user flows
- **UI DEVELOPER** - Turning mockups into functioning sites
- **GAME DESIGNER** - Planning menus or in-game UI

ASEPRITE

Aseprite is made for pixel art. It's lightweight, simple to use, and great for creating sprites, tiles, and small-scale animations. If you're working on a game with a pixel art style, this is one of the best tools for building assets frame by frame. It also makes animation super intuitive, which is great for character loops or environmental effects.

COST

PAID one-time payment (affordable and worth it if you're doing pixel art regularly)

- A free trial version is also available on GitHub, although it has limited features.

USE

Pixel art, sprite animation, tilesets, and small-scale game visuals.

ROLES

Used mainly by artists working in a pixel style.

Especially useful for:

- **2D ARTIST** - Drawing pixel assets and characters
- **2D ANIMATOR** - Animating frame-by-frame pixel motion
- **GAME DESIGNER** - Prototyping with visual placeholders
- **TECHNICAL ARTIST** - Prepping assets for integration

BLENDER

Blender is our main tool for 3D work. It's used for modeling, sculpting, animation, rendering, and even basic VFX. Whether you're creating environments, props, characters, or cinematics, Blender can do it. It's open source, powerful, and works well for both stylized and realistic art styles.

COST

FREE

USE

3D modeling, animation, rigging, rendering, environment design, and basic VFX.

ROLES

Used heavily in 3D and technical visual work.

Especially useful for:

- **3D ARTIST** - Modeling characters, props, and environments
- **3D ANIMATOR** - Animating scenes and characters
- **TECHNICAL ARTIST** - Rigging, optimizing, and integrating assets
- **CINEMATOGRAPHER & VIDEO EDITOR** - Building or staging cinematic scenes

🔗 **CLIP STUDIO PAINT**

Clip Studio Paint is a popular tool for digital drawing and illustration, especially when it comes to comics, storyboards, and animation. It's known for its smooth brush engine, great pen stabilization, and animation features. It's a go-to for many 2D artists who want more control over their linework or are working on frame-by-frame animation.

COST

PAID, one-time payment or subscription (depending on version and platform)

USE

Digital illustration, storyboarding, comics, and 2D animation.

ROLES

Mainly used by the visual art Team.

Especially useful for:

- **2D ARTIST** - Sketching, painting, and creating clean illustrations
- **STORYBOARD ARTIST** - Laying out scenes and visual sequences
- **2D ANIMATOR** - Creating frame-by-frame animation
- **GRAPHIC DESIGNER** - Drawing assets or stylized visuals

DA VINCI RESOLVE

Da Vinci Resolve is a professional video editor known for its powerful color grading tools. It's a solid alternative to Premiere Pro, and a lot of people use it for editing trailers, short films, or anything that needs a polished finish. It also works well for sound editing and visual effects, all in one program.

COST

FREE version available with most features

- Studio version is a one-time payment with extra tools and effects

USE

Video editing, color grading, sound mixing, VFX, and final polishing.

ROLES

Mainly used by the video and post-production Team.

Especially useful for:

- **VIDEO EDITOR** - Cutting and finishing trailers or cinematic content
- **SPECIAL EFFECTS ARTIST** - Adding visual polish
- **AUDIO ENGINEER** - Editing and syncing sound
- **MARKETER & MARKETING MANAGER** - Finalizing content for campaigns

FIGMA

Figma is used for designing user interfaces, prototyping interactions, and collaborating in real time. It's a browser-based design tool (with a desktop app option) that makes it easy for teams to work on the same file at once. It's great for layout-heavy projects like websites, apps, or menus, and makes feedback quick and visual. You can also build interactive mockups without touching code.

COST

FREE tier available

- Paid plans for advanced features and Team management

USE

UI/UX design, wireframes, interactive prototypes, collaborative layout design.

ROLES

Mostly used by the visual and technical Teams.

Especially useful for:

- **GRAPHIC DESIGNER** - Layouts, component libraries, design systems
- **UI DEVELOPER** - Referencing and developing UI mockups and prototypes for frontend implementation
- **GAME DESIGNER** - Wireframing menus and in-game interfaces

FL STUDIO

FL Studio is a digital audio workstation (DAW) used for composing music and producing sound. It's especially popular for electronic and cinematic tracks, and it has a user-friendly interface that makes experimenting easy. If you're working on original music, background scores, or sound loops for a project, this is one of the tools you might reach for.

COST

PAID (one-time purchase with free lifetime updates)

- Different versions available depending on features

USE

Music composition, sound design, loops, and audio production.

ROLES

Used by the sound and music Team.

Especially useful for:

- **COMPOSER** - Creating music scores and tracks
- **SOUND DESIGNER** - Producing original audio or sound effects
- **AUDIO ENGINEER** - Mixing or editing stems

FRAMER

Framer is a tool we use to build functional websites through its visual editor. It's great for landing pages, portfolios, and interactive content that needs to look polished and feel smooth. You can build full sites visually, but there's also room to add custom code, which is where developers can step in if needed. It's flexible, fast, and gives us control over both design and behavior.

COST

FREE plan available

- Paid plans unlock more pages, custom domains, and advanced features

USE

Website building, landing pages, portfolios, and interactive web design with optional coding.

ROLES

Used by both designers and developers.

Especially useful for:

- **UI DEVELOPER** - Adding custom code or handling advanced features
- **GRAPHIC DESIGNER** - Designing layouts and visuals
- **MARKETER & MARKETING MANAGER** - Building campaign pages or one-pagers
- **LEAD & PROJECT MANAGER** - Reviewing and managing launches

FRAME IO

Frame.io is a cloud-based video collaboration tool built for reviewing, commenting, and sharing media during production and post. It lets teams upload drafts, leave frame-accurate feedback, and track version history — all in one place. Perfect for remote collaboration, quick turnarounds, and keeping everyone aligned on video edits or visual reviews.

COST

PAID subscription (free trial available)

USE

Video review, edit feedback, version tracking, collaborative approval.

ROLES

Mostly used by the visual and sound Teams.

Especially useful for:

- **CINEMATOGRAPHER & VIDEO EDITOR** - Uploading drafts, managing revisions, and receiving feedback
- **SOUND DESIGNER** - Syncing audio to updated visuals or reviewing timing
- **VISUAL MANAGER** - Reviewing and approving visuals across production
- **PROJECT MANAGER** - Coordinating timelines and overseeing review stages

GAMEMAKER

GameMaker is a 2D game engine that's great for making games quickly without needing to build everything from scratch. It's beginner-friendly but still powerful enough to handle complex mechanics. You can use drag-and-drop tools or code in GameMaker Language (GML) if you want more control. We use it when working on pixel-based games, narrative prototypes, or small-scale interactive projects.

COST

FREE version available

- Paid license required for full export options (like desktop or console builds)

USE

2D game development, prototyping, pixel-style games, and small interactive experiences.

ROLES

Used by Teams focused on 2D or simplified gameplay systems.

Especially useful for:

- **PROGRAMMER** - Writing gameplay logic in GML
- **GAME DESIGNER** - Building mechanics and systems
- **2D ARTIST & 2D ANIMATOR** - Implementing visual assets

GITHUB

GitHub is a platform for hosting and collaborating on code. It's primarily used for version control (via Git), but also supports project tracking, issue management, and documentation through features such as Issues, Pull Requests, and Wikis. Whether you're developing games, tools, or websites, GitHub helps teams stay organized and in sync, especially when working across branches or remotely.

COST

FREE for individuals and public repos

- Paid plans unlock private repos, advanced permissions, and team tools

USE

Code hosting, version control, collaboration, bug tracking, documentation.

ROLES

Mostly used by the technical Team.

Especially useful for:

- **PROGRAMMER** - Versioning, debugging, and merging code
- **UI DEVELOPER** - Pushing and syncing frontend/backend projects
- **GAME DESIGNER** - Accessing dev builds and commenting on gameplay scripts
- **PROJECT MANAGER** - Reviewing progress through Issues or project boards



Godot is a free, open-source game engine that supports both 2D and 3D development. It's flexible, lightweight, and gives you full control over how things work. We use it for projects that need more technical freedom, whether that's in mechanics, visuals, or structure. It's perfect for indie-style games where custom systems or experimental ideas are a big part of the vision.

COST

FREE and open-source

- No license fees, even for commercial use

USE

2D and 3D game development, systems-driven gameplay, and experimental game ideas.

ROLES

Used for both creative and technical development.

Especially useful for:

- **PROGRAMMER** - Scripting mechanics and building systems
- **GAME DESIGNER** - Prototyping features and fine-tuning gameplay
- **TECHNICAL ARTIST** - Integrating assets and optimizing visuals



Krita is a free digital painting program made for artists. It's especially good for illustration, concept art, and frame-by-frame animation. The brush engine feels smooth, the interface is clean, and it has all the essentials for drawing and painting. It's a great alternative to Photoshop or Clip Studio if you're focusing on expressive, detailed artwork.

COST

FREE and open-source

USE

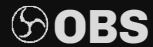
Digital painting, illustration, concept art, and 2D animation.

ROLES

Used by visual artists and animators.

Especially useful for:

- **2D ARTIST** - Painting, sketching, and building assets
- **2D ANIMATOR** - Frame-by-frame sequences or loops
- **STORYBOARD ARTIST** - Quick visual layouts and drafts
- **GRAPHIC DESIGNER** - Stylized visuals or hand-drawn textures



OBS (Open Broadcaster Software) is used for screen recording and live streaming. We use it mainly to record project demos, gameplay footage, tutorials, or anything that needs to be captured straight from your screen. It's lightweight, customizable, and works well even on lower-end setups.

COST

FREE and open-source

USE

Screen recording, gameplay capture, live streams, video walkthroughs.

ROLES

Used across different types of documentation and content capture.

Especially useful for:

- **CINEMATOGRAPHER & VIDEO EDITOR**- capturing raw footage for trailers or devlogs
- **DOCUMENTATION MANAGER**- recording tutorials or walkthroughs
- **GAME DESIGNER & PROGRAMMER** – Showing prototypes or mechanics
- **MARKETER & MARKETING MANAGER** – Collecting in-game footage for campaigns

MOHO

Moho is a 2D animation tool focused on rigging and vector-based animation. It's known for making character rigging, bone systems, and smooth motion more manageable compared to frame-by-frame animation. We use it when a project requires complex character animation that needs to remain consistent across scenes.

COST

PAID license (one-time payment for standard or pro versions)

USE

2D character rigging, vector animation, motion cycles, and reusable animated assets.

ROLES

Used across projects that require structured 2D animation.

Especially useful for:

- **2D ARTIST** - Rigging and animating characters or props
- **TECHNICAL ARTIST** - Prepping and optimizing rigs
- **VIDEO EDITOR & SPECIAL EFFECTS ARTIST** - Integrating animated sequences

REAPER

Reaper is a digital audio workstation (DAW) used for recording, editing, and mixing sound. It's lightweight, powerful, and super customizable. Whether you're working on music, voiceovers, sound effects, or anything else audio-related, Reaper gets the job done without being bloated. It's a solid choice for both beginners and pros.

COST

FREE trial available

- One-time payment for a full license (affordable compared to most DAWs)

USE

Audio editing, voice recording, sound mixing, and mastering.

ROLES

Used by everyone working with sound.

Especially useful for:

- **AUDIO ENGINEER** - Mixing and mastering tracks
- **VOICE ACTOR** - Recording and editing takes
- **COMPOSER & SOUND DESIGNER** - Layering and fine-tuning audio

ROBLOX STUDIO

Roblox Studio is the game development platform used to create Roblox experiences. It features a built-in editor, asset library, and Lua scripting support, making it accessible for beginners while being flexible enough to build full games. We use it for game projects for the Roblox platform.

COST

FREE

USE

Game development for Roblox, multiplayer games, and platform-based events.

ROLES

Used on specific Roblox-based projects.

Especially useful for:

- **PROGRAMMER** - Scripting gameplay and systems in Lua
- **GAME DESIGNER** - Building mechanics and world layouts
- **3D ARTIST & 3D ANIMATOR** - Importing and applying assets
- **MARKETING MANAGER** - Organizing community events or public releases



Unity is a widely used game engine for both 2D and 3D development. It's flexible, scalable, and supports everything from small indie projects to full-blown commercial games. We use Unity for projects that require more advanced systems, real-time interaction, or cross-platform builds. It supports both visual scripting and traditional code, making it adaptable depending on the user's preferences.

COST

FREE for personal and small-team use

- Paid tiers for larger-scale commercial releases

USE

Game development, interactive experiences, real-time rendering, cross-platform projects.

ROLES

Used heavily on larger or more complex projects.

Especially useful for:

- **PROGRAMMER** - Coding gameplay systems and logic
- **GAME DESIGNER** - Building levels, testing mechanics
- **TECHNICAL ARTIST, 3D ARTIST & 3D ANIMATOR** - Asset integration and optimization

VISUAL STUDIO

Visual Studio is a full-featured code editor and development environment. It's mainly used for writing, testing, and debugging code. We use it when working on larger or more complex programming tasks, especially for game development, tool creation, or system logic that goes beyond quick scripts.

COST

FREE (Community edition)

- Paid (Professional and Enterprise editions with additional features)

USE

Programming, scripting, debugging, and building tools or systems for games and projects.

ROLES

Used across different types of technical development work.

Especially useful for:

- **PROGRAMMER** - Writing and testing game code, tools, or systems
- **SYSTEMS DESIGNER** - Working on complex systems logic
- **TECHNICAL ARTIST** - Building or refining custom tools for the Team
- **GAME DESIGNER** - Prototyping mechanics or gameplay logic (when code is involved)

ASSET LIBRARIES

These tools provide ready-to-use resources, including fonts, templates, and components, that are incorporated into the final product, not just as placeholders.

(illustration here)

GOOGLE FONTS

Google Fonts is a free font library offering hundreds of typefaces for use on the web and in design. All fonts are open-license, easy to browse, and downloadable for local use or embeddable via the web. It's a solid go-to when you need something clean, readable, and accessible, without worrying about licensing issues.

COST

FREE

USE

Typography, web-safe fonts, brand systems, design mockups.

ROLES

Mostly used by the visual Team.

Especially useful for:

- **GRAPHIC DESIGNER** - Selecting typefaces for layouts, posters, and UI
- **UI DEVELOPER** - Embedding web fonts for clean, consistent styling
- **VISUAL MANAGER** - Maintaining font consistency across brand or project designs
- **MARKETER** - Applying brand-aligned fonts in promo visuals or presentations

COOLORS

Coolors is a fast, intuitive tool for generating, exploring, and organizing color palettes. You can create custom palettes from scratch, extract colors from images, or browse trending combos from the community. It's especially helpful when you need to lock in a visual tone or build consistent color systems across a project.

COST

FREE with limited features

- Paid plan unlocks saved palettes, brand kits, and extra tools

USE

Color palette generation, tone setting, visual system building.

ROLES

Mostly used by the visual Team.

Especially useful for:

- **GRAPHIC DESIGNER** - Building palettes for layouts, posters, or UI
- **VISUAL MANAGER** - Developing color systems for branding or project consistency
- **MARKETER** - Choosing brand-aligned color combos for campaigns
- **2D ARTIST & 3D ARTIST** - Applying palettes to illustrations, environments, and stylized assets

COLORFFY

Colorffy is a sleek all-in-one tool for creating, converting, and exploring color palettes. It includes generators for gradients, themes, text colors, and mesh blends, as well as features for converting color formats, exporting files, and saving branded sets. With its lightweight interface and built-in creative tools, it serves as a great starting point for both quick ideation and comprehensive design preparation.

COST

FREE plan available

- Paid plan unlocks additional export options, AI features, and advanced tools

USE

Color palette creation, gradient design, color conversion, theme building, asset-ready exports.

ROLES

Mostly used by the visual Team.

Especially useful for:

- **GRAPHIC DESIGNER** - Building color combos, gradients, and UI-ready palettes
- **VISUAL MANAGER** - Prepping brand kits, theme references, and color systems
- **MARKETER** - Making visuals that align with campaign aesthetics
- **2D ARTIST & 3D ARTIST** - Applying or referencing color sets for stylized illustrations or assets

STOCK MEDIA

These are the tools we use to obtain stock media when we lack the resources to produce specific content ourselves.

(illustration here)

PEXELS

Pexels is a free stock media library with high-quality photos and videos. We use it when we need placeholder visuals, reference images, or footage to support presentations, mockups, or internal pitches. Everything on Pexels is free to use, including for commercial projects.

COST

FREE

USE

Placeholder visuals, mockups, pitch decks, reference images, stock video clips.

ROLES

Used across different types of visual and marketing work.

Especially useful for:

- **GRAPHIC DESIGNER & 2D ARTIST** - Adding placeholder images or references
- **VIDEO EDITOR & MARKETER** - Sourcing stock footage for edits or promos
- **IDEATOR & PROJECT MANAGER** - Pulling visuals for concepts or presentations

TEXTURE LABS

Texture Labs is a free library of high-resolution textures designed for creative projects. We use it to add detail and depth to artwork, mockups, and designs without having to create textures from scratch. It's a handy resource for quick, high-quality surface detail.

COST

FREE

USE

Adding textures to 2D or 3D assets, creating mockups, enhancing designs with surface detail.

ROLES

Used across different types of visual creation.

Especially useful for:

- **2D ARTIST & 3D ARTIST** - Applying textures to illustrations, models, or renders
- **GRAPHIC DESIGNER** - Using textures in layouts, backgrounds, or branding
- **TECHNICAL ARTIST** - Testing or applying materials in projects

FREE SOUND

Freesound is a large, free library of user-submitted sound effects, ambiences, and audio clips. We use it to source placeholder sounds, background noise, or inspiration for custom sound design. It's great for quick audio needs or prototyping sound ideas.

COST

FREE (requires a free account for downloading)

USE

Placeholder sound effects, ambient audio, sound references, and prototyping.

ROLES

Used across different types of audio and multimedia work.

Especially useful for:

- **COMPOSER & SOUND DESIGNER** - Sourcing or referencing audio elements
- **VIDEO EDITOR & MARKETING MANAGER** - Adding temporary sound to trailers or promos
- **GAME DESIGNER & TECHNICAL ARTIST** - Testing sounds in early builds

SVG REPO

SVG Repo is a massive library of free, downloadable SVG graphics. It's packed with icons, illustrations, and simple vector assets that can be customized and reused across various projects. Everything is open-license, which makes it a great go-to when you need clean visuals fast without licensing headaches.

COST

FREE

USE

Icons, vector illustrations, placeholders, scalable graphic elements.

ROLES

Mostly used by the visual Team.

Especially useful for:

- **GAME DESIGNER** - Finding and customizing vector assets for layouts or UI
- **UI DEVELOPER** - Sourcing scalable icons and visuals for frontend design
- **MARKETER** - Grabbing quick visuals for decks, posts, or campaigns
- **VISUAL MANAGER** - Curating asset libraries and speeding up production

OTHER

Didn't see your go-to tool on the list? No problem. We have a tool suggestion forum where you can submit a request. After submitting, make sure to ping a higher-up so it gets noticed; we don't check that thing religiously. If the tool aligns with our workflow and quality standards, we may add it to the list.

[FORUM HERE →](#)

BLACKLISTED TOOLS

These are tools we don't allow in Studio projects. Each tool listed here has a specific reason for being blacklisted, whether it's due to licensing issues, security risks, quality concerns, or compatibility problems. If you're unsure about a tool not listed in this guide, ask before using it.

(illustration here)

CAPCUT

REASON - Its terms of use give the company rights to reuse, distribute, and monetize your content without notice or compensation, which creates serious legal and ownership risks for our projects.

MOST GENERATIVE AI TOOLS

REASON - We do not permit the use of generative AI for production in Studio projects, particularly when it comes to visuals and sound. Our work is built on original human-created content; however, when it comes to writing and code, it is debatable in specific scenarios.

Read our guidelines on the usage of generative AI in the [GENERATIVE AI GUIDELINES](#) section if you plan on using any generative AI.

COMMUNICATION

COMMUNICATION

This chapter is here to show you how we communicate as a Team, how you can contribute to that, and what we expect along the way.

COMMUNICATION TOOLS

We utilize a limited set of tools to ensure clear, simple, and easy-to-follow communication. Each one has a purpose. Here's how they fit into our workflow.

You can click on the headings of the Tools to learn more about them in the [TOOLS](#) chapter.

DISCORD

This is where most of our communication happens. We use Discord for chat, quick updates, voice calls, meetings, and sharing files. We have a main server that serves as a hub, where all our general information and resources are stored. We maintain separate servers for each project to keep things organized.

GMAIL

Gmail is for external communication or formal messages. If we're reaching out to collaborators or sending files that don't fit on Discord, this is what we use. It's not something you'll use daily, but it matters when formal contact is needed.

GOOGLE CALENDAR

Google Calendar keeps us on the same page about meetings, deadlines, and key dates. To check what's scheduled, refer to Google Calendar. Check it so you know when things are happening, and don't miss important events.

FEEDBACK

Feedback helps us make improve work and stay aligned as a Team. It's how we improve, solve problems together, and keep projects moving forward. This section outlines how feedback integrates into our process, how we provide it, and how we utilize it to achieve better results.

FEEDBACK CYCLES

We don't save feedback for the end of a project. We share, review, and adjust as we go. A feedback cycle helps us identify issues early, stay on the same page, and avoid last-minute problems.

What a feedback cycle looks like here:

- **SHARE EARLY PROGRESS** - This could be a sketch, a draft, a prototype, or whatever makes sense for your role. The point is to give people a chance to help while there's still time to make changes.
- **ASK FOR FEEDBACK IN THE RIGHT PLACE** - Keep it in the project server, document comments, or shared boards, not just in private DMs unless it's sensitive. This way, the whole Team stays in the loop.
- **THE TEAM GIVES INPUT** - Feedback should be clear, specific, and focused on improving the work. We don't nitpick for the sake of it.
- **YOU REVISE AND UPDATE** - You make adjustments based on the feedback. If you're unsure about something, ask for clarification.
- **YOU CHECK BACK IN** - Share the updated version so people can see what's changed and give a final look if needed.

And you keep repeating this cycle until the finished product is as intended. Development and work are often iterative processes that require a lot of back-and-forth. Sometimes it even requires you to go back to the drawing board.

WHY WE DO IT THIS WAY

Feedback cycles keep projects moving, prevent last-minute issues from arising, and help everyone grow and improve. The goal is always to make the work better, together.

CONSTRUCTIVE CRITICISM

Constructive criticism is about helping the work improve in a way that's clear, useful, and respectful. The point isn't just to spot what's wrong, but to give input that helps someone make the work improve and easier for the Team to build on.

What constructive criticism means at HELLINGTON:

- **BE CLEAR AND DIRECT** - If something doesn't work, say exactly what and why. Instead of "this feels off," explain what isn't landing or what could be stronger. The goal is to help the person identify what needs changing without requiring them to guess.
- **FOCUS ON THE WORK, NOT THE PERSON** - We're here to build the project, not critique each other as individuals. Keep feedback focused on the task at hand, whether it's the design, writing, code, or whatever is being shared.
- **POINT OUT WHAT'S WORKING TOO** - It's just as important to highlight what's good. This helps others understand what to retain, build upon, or further develop. Balanced feedback yields a better final result and maintains a positive process.
- **ENSURE YOUR FEEDBACK HELPS THE WORK MOVE FORWARD** - If your comment doesn't help the person improve the work, rethink how you're framing it. Feedback should give someone a clearer idea of what to do next.
- **BE KIND AND RESPECTFUL IN YOUR APPROACH** - It costs nothing, and it keeps collaboration smooth. Tone matters, even when you're being honest.
- **WHEN RECEIVING FEEDBACK, STAY OPEN-MINDED** - Feedback is about the work, not about you as a person. It's okay to explain your choices or ask for clarity if something doesn't make sense. The goal is to improve the project together, not to "win" the discussion.
- **ASK IF YOU'RE UNSURE** - If you're not sure how to give feedback in a way that helps, or if the feedback you got isn't clear, just ask.

MEETINGS

Meetings at HELLINGTON are for sharing updates, planning, problem-solving, checking progress, and making decisions as a group.

How often do we meet?

It depends on the project and circumstances. Sometimes we have meetings on a weekly or bi-weekly basis. Other times, we meet as needed when there's something specific to cover. Studio-wide meetings are rare and are always announced in advance. They usually occur more frequently in the lead-up to a new project.

Where are meetings hosted?

Meetings happen in Discord voice channels. The schedule and links are shared in announcement channels and sometimes scheduled further in advance through Google Calendar.

Who attends?

Only the people who are needed for that meeting. If you're invited, it's because you're expected to share, contribute, or help make decisions. However, sometimes meetings are hosted simply to inform the Team or share updates, and anyone can attend optionally, by announcing their attendance in advance so they won't need to read Meeting logs to catch up.

Why we meet:

- To align the Team on goals and progress
- To make decisions that need group input
- To inform the Team about updates and changes

How we keep meetings focused:

- Every meeting has a clear purpose
- We (try to) stay on topic
- We log key decisions and next steps afterwards

CHECK-INS

Check-ins are smaller and more personal than full meetings. They're about staying aligned, unblocking work, and ensuring support is available when needed.

How often do check-ins happen?

Usually on a weekly or bi-weekly basis, depending on the project and role. You might also have extra check-ins during busy periods or when something needs closer attention.

Who are check-ins with?

Typically, your Manager, Advisor, or Project Lead, whoever is guiding the work you're doing.

Why we have check-ins:

- To track progress without micromanaging
- To catch problems early
- To provide direct support or feedback
- To keep communication open

How they work:

Check-ins are short and focused. We cover what's done, what's in progress, and what's next. If you need help, this is the place to bring it up.

COMMUNICATION CONVENTIONS

We use a few simple systems to make communication cleaner and easier to track. These systems aren't here to add extra steps; they're here to help ideas, tasks, and updates move without confusion or wasted effort.

GREYSCALE SYSTEM

The Greyscale System replaces the idea of “greenlighting” or outright rejecting ideas. It's built around the belief that timing matters (kairos). An idea that isn't used now might still be valuable later.

-  **RED** - This idea is so great that it should be highlighted (essentially favorited)
-  **WHITE** - This idea or project will/should be used
-  **GREY** - This idea or project might be used
-  **BLACK** - This idea or project won't be used right now

Think of it like a light bulb; the more it lights up, the more excited you are to work on it, and if you really love the idea, it unlocks your inner creativity.

The idea is that no idea gets devalued. The focus is on finding the right time for everything. All ideas are saved, so we have a library of ideas to draw from, slowly building over time.

FLAGGING SYSTEM

The Flagging System is for reporting progress quickly and clearly. It also helps others see where help is needed.

- **WHITE** - Task has started
- **BLACK** - Task is in progress
- **BLUE** - Task is complete
- **RED** - There's an issue or roadblock
- **YELLOW** - Volunteering to help or take on the task

This makes it easy to track work without lengthy explanations, and easy for others to jump in if needed.

VERSION CONTROL & BACKUPS

- Link the version history when you make significant changes, so the Team can follow what changed
- Duplicate documents or files before big edits if you're not sure
- Follow naming conventions so nothing gets lost in clutter

Learn more about how to follow our version control and backup conventions accurately, as well as our other communication conventions, in [INSERT HERE](#).

PROCESS FLEXIBILITY

We provide best practices, templates, and guides to help the Team stay clear and consistent. A shared system makes collaboration easier. When everyone knows where things are and how things work, it's faster and smoother to get things done.

That said, we don't believe one system fits everyone all the time, even when we try our best to be flexible with our options. If you have a process that differs from what we present and genuinely helps with the work, we're open to it. What matters is that:

- It doesn't harm the overall workflow or create extra confusion
- It brings a real benefit to the work or the Team
- You provide a reference sheet so others can understand and follow your method, just like we provide references for ours

We want systems that work. If you can help us improve how we do things, challenge what's there and share your approach. The goal is to support the Team, not stick to a process just because it's written down.

Here is how to have your method implemented:

Use our template for making your own method reference sheet:

[METHOD REFERENCE TEMPLATE HERE →](#)

Use this forum to submit your method reference sheet for review (make sure to alert a higher-up; we don't usually check our forms):

[SUBMIT METHOD REFERENCE SHEET HERE →](#)

COMMUNICATION TIPS

Here are a few extra tips that help keep communication smooth at HELLINGTON:

- **KEEP IT IN THE RIGHT PLACE** - Use the proper channels, threads, or docs. Important info should be shared where the whole Team can see it, not hidden in private DMs.
- **BE CLEAR AND DIRECT** - Say what you mean without unnecessary filler. The clearer your message, the less chance for confusion.
- **DON'T SIT ON BLOCKERS** - If something's slowing you down, flag it early. It's easier to solve problems together before they escalate into bigger issues.
- **RESPECT PEOPLE'S TIME** - Updates should be useful, not noise. Meetings should have a purpose. Keep check-ins focused.
- **ASK IF YOU'RE UNSURE** - It's always better to check than to guess. No one expects you to have all the answers.
- **CONSIDER THE INTENT WHEN GIVING OR RECEIVING FEEDBACK** - Ask what was intended by the work and how it was meant to be perceived. This gives you the proper framework for offering helpful critique or steering someone back on track if they've missed the mark of the project brief.
- **USE COMMENTS IN DOCS RATHER THAN OVERWRITING** - When collaborating on shared documents, leave comments or suggestions instead of changing someone's work without context. It keeps things clear and avoids confusion.

(illustration here)

PROJECT FRAMEWORK

PROJECT FRAMEWORK

This chapter outlines how we handle projects at HELLINGTON, from pitching an idea to wrapping things up and reflecting on what we've created. It provides a clear picture of how our projects flow, what is expected in each phase, and how to navigate the process, whether you're leading or contributing.

You'll learn:

- Our formula for creating a HELLINGTON classic project
- How to pitch a project idea that actually gets considered
- The different ways we structure and manage projects (not everything is linear)
- What phases does each project go through
- What happens during each phase, and who's usually involved
- How we document, wrap, and reflect on the work

We don't lock ourselves into one strict process. Some projects move step by step. Others grow and shift with time. What matters is that we're aligned and that we stay adaptable without losing momentum.

(illustration here)

PROJECT FORMULA

Evoke. Provoke. Impact.

This is the core of what we make at HELLINGTON. Every project should hit all three, not in the same way, or at the same intensity, but they should all be there in some form. It's how we know an idea has substance. It's how we know it's worth making.

These are creative checkpoints, a formula that shapes how we evaluate, refine, and ultimately decide which projects move forward.

EVOKE

Pulls people in.

It makes someone stop and pay attention. Maybe it's the concept. Maybe it's the tone, the world, or the characters. It's the feeling that something here matters, that there's something beneath the surface. This is the hook, the thing that stirs curiosity or emotion and gets someone invested right away.

PROVOKE

Challenges them.

It makes people think. It reframes something they thought they knew, or exposes something they didn't. It can be subtle or aggressive, loud or quiet, but it has to confront. Provoke doesn't mean being edgy for the sake of it. It means pressing on something real, something worth questioning.

IMPACT

Leaves a lasting impression.

After the story ends, the game fades out, the credits roll, and it stays with you. Impact is the part people remember. Not because it told them what to think, but because it made them feel something. Projects that impact are the ones that echo. They shift perspective. They linger.

PROJECT PITCH

Every project starts with an idea, and anyone can pitch one. You don't need to be in charge or have everything figured out right away. You just need to know what you're trying to make and why it's worthwhile.

The pitch stage isn't about planning the whole thing. It's where you capture the core of the idea. If it has a solid foundation, the rest will fall into place naturally.

WHAT TO INCLUDE

A good pitch doesn't ramble. It gives us just enough to see the potential and understand where it's headed. Make sure it includes:

- **THE CONCEPT** - What's the idea in a nutshell? Summarize it in a singular paragraph if possible. Think genre mashups, emotional hooks, or intriguing setups.
- **THE USP (UNIQUE SELLING POINT)** - What makes this idea different from something we've already seen?
- **THE PURPOSE** - What makes this idea interesting? What could it leave people thinking or feeling?
- **THE MEDIUM** - Game? Film? Interactive web piece? Series? This helps frame the format.
- **THE GENRE(S)** - What category does it fall under? (Drama, horror, dark comedy, surreal mystery...)
- **THE AUDIENCE** - Who is this for? Who would enjoy it or connect with it?
- **PROOF OF CONCEPT** - Show the idea in action. This could be a rough sketch, storyboard, short prototype, sample scene, or any other visual representation that gives a glimpse into how the idea could come to life. It doesn't need to be polished, just enough to help us get the vision.

THINGS TO AVOID

It's great to have ideas, just don't drown your pitch in them. The goal is to sell the core concept, not the entire thing. Keep it focused and easy to take in.

AVOID

- Writing a full-on essay
- Overloading it with details we don't need yet
- Dropping every bit of reference or research
- Trying to answer every possible question upfront

If you've got extra stuff like moodboards, lore, or research, that's good. Just link it at the bottom in a "Resources" section. Keep the main pitch simple, straightforward, and easy to get behind. Think of it like an elevator pitch: you've got one short ride to convince someone it's worth their time. Make it count.

SUBMIT YOUR PITCH

What actually happens when you submit a pitch? Here's the whole timeline, from start to finish.

1. PREPARE YOUR PITCH

Use the template and follow the guide in the previous section. Keep it focused and sell the core idea without overwhelming it with unnecessary details. You can include supporting material (moodboards, research, etc.) in a "References" section at the end.

Reminder: You must agree to the submission terms before sending. We cannot guarantee that your idea will or won't be used or referenced in the future if we don't proceed with it at the time. Only submit if you're okay with it being used directly or as inspiration.

PROJECT PITCH TEMPLATE:

 [HS Project Pitch Template](#)

2. SUBMIT IT

Please complete the official Pitch Form and attach your supporting document. Once submitted, a Manager or Lead will send your pitch to the Team Server for voting.

[PITCH SUBMISSION FORM →](#)

3. TEAM VOTE

All Team Members get to vote on whether the pitch should move forward using the [GREYSCALE SYSTEM](#). We don't approve ideas based solely on hierarchy; we want the Team to actually enjoy the project. A pitch that gets enough support moves to review.

4. REVIEW & EVALUATION

Leads and Managers take a closer look. We evaluate the pitch based on the Studio's Mission, values, and the Project Pass Iceberg. You may be asked to revise or clarify specific parts before we're sure it's a good fit for our Studio.

5. PRESENTATION

Before we officially move forward, the idea gets presented to the Team. If you're comfortable, you'll present it yourself; otherwise, a higher-up will do it for you. This helps the Team better understand the vision and ask any final questions before we bring it to a final vote.

6. DECISION

You'll receive one of the following outcomes:

- ☒ **APPROVED** - The project enters pre-production
- ☒ **NEEDS REVISION** - You'll be given notes and can resubmit
- ☐ **NOT MOVING FORWARD** - The project isn't the right fit right now, but future pitches are always welcome

7. ONWARD TO PRE-PRODUCTION

If the project pitch is approved for pre-production, the project officially commences. We'll assign roles and begin the process as described in the rest of this chapter.

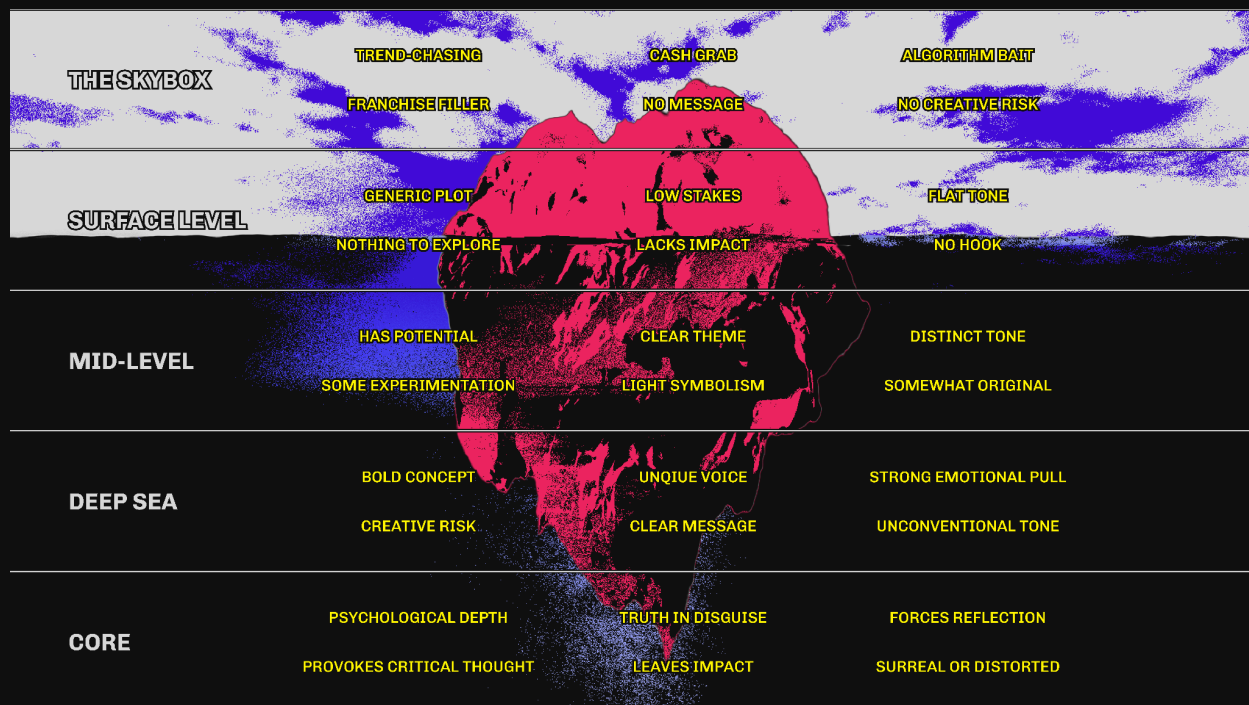
THE PROJECT PASS ICEBERG

Not every pitch gets picked up; that's just how it goes. But from the start, we've been clear about the kind of work we want to make. That's where the Project Pass Iceberg comes in.

It's not a checklist. It's a visual tool that shows how closely a pitch aligns with the kind of stories we're drawn to. The deeper your idea sits on the iceberg, the more likely it is to feel like something we'd actually make. That said, don't let it scare you off, even if your idea floats closer to the surface, it's still worth sharing. We are flexible, and even if your idea isn't exactly what we were looking for from the get-go, it may convince us in other unexpected aspects.

One last thing: The iceberg doesn't judge the quality of your pitch. It simply reflects whether it aligns with what we're interested in producing. Even the traits we label as "not for us" can be valuable in their own right. There's a market for everything, we're just not trying to sell to all of them.

Here's how it breaks down:



THE SKYBOX

Ideas that drift way outside of what we do. These are often cash grabs, trend hoppers, or completely hollow concepts. They're not necessarily bad, just not the kind of project we want to work on.

SURFACE LEVEL

Safe, standard, or just plain uninteresting. This is where generic slice-of-life stuff usually ends up, especially if it doesn't bring anything new to the table.

MID-LEVEL

Projects with decent interest, but missing something; maybe they lack a unique voice, a strong twist, or just don't go deep enough thematically. They have potential, but they need work.

DEEP SEA

These are ideas that feel bold, strange, risky, but interesting. They might not be fully fleshed out yet, but they're pushing in the right direction.

THE CORE

This is as iconic to our Studio as it gets. Psychological horror, layered messaging, surreal or experimental, unexpected emotional impact, ideas that challenge perception and encourage thought.

PROJECT MANAGEMENT

Not every project needs the same approach. Some thrive with structure, others need room to breathe and adapt. This section outlines the primary methods we use to manage projects at HELLINGTON, including how we plan, organize, and progress projects once they have been approved.

Whether it's a short-term video, a massive game prototype, or a collaborative document, the way we handle the process can shift depending on the scope, complexity, and flexibility.

This section outlines the four primary management methods we employ, selecting the most suitable approach for each situation.

If you want to find out more about project management methodologies we recommend reading Asana's resource going over the 12 most popular frameworks:

[12 MOST POPULAR PROJECT MANAGEMENT METHODOLOGIES →](#)

(illustration here)

WATERFALL (LINEAR)

The Waterfall method is the most straightforward: you complete one phase, then move on to the next. No turning back, no jumping ahead. It works best when the project is short, the outcome is crystal clear, and every task is tightly defined from the start.

We rarely use this at HELLINGTON, but it can be helpful for simpler deliverables where flexibility isn't needed, like polishing a pre-written script or exporting final edits from a cut.

How it works:

- Plan → Build → Deliver, in a straight line
- Each stage is finalized before the next one begins
- Requires clear requirements upfront

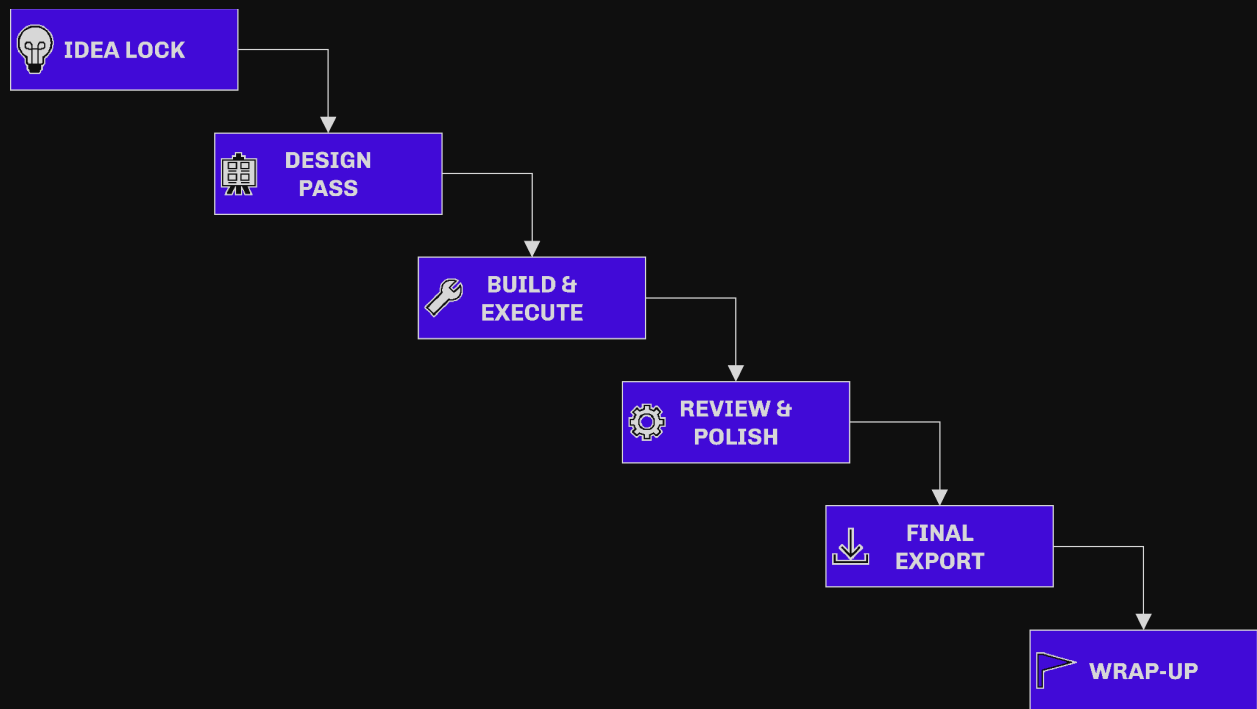
Good for:

- One-off tasks with no room (or need) for feedback loops
- Projects where timing and steps are already locked
- Simple visual outputs, internal documents, or static media

Things to keep in mind:

- Not flexible, changes are complex to make once you're deep in
- If you miss something early, it'll cost you later
- Doesn't work well for exploratory or creative-heavy work

Here's how it breaks down:



(Used occasionally, for simple, linear projects with precise requirements)

1. **IDEA LOCK** - The core concept is clear, scoped, and locked in.
2. **DESIGN PASS** - Visuals, story, or systems are planned and outlined.
3. **BUILD & EXECUTE** - Everyone creates what they're responsible for.
4. **REVIEW & POLISH** - The Team checks for quality and consistency.
5. **FINAL EXPORT** - The last version is rendered, exported, or published.
6. **WRAP-UP** - Archive files, reflect, and prep for showcasing.

ITERATIVE / AGILE

This is the method we use the most. Instead of doing everything in one big sweep, we work in cycles, testing, refining, and improving as we go. It's built for creative chaos: perfect for projects that evolve or need flexibility.

Each cycle (or “iteration”) allows us to reevaluate what’s working and what needs to change. You try something, test it, then move forward wiser.

How it works:

- Break the project into smaller tasks or features
- Prioritize what’s most important to build first
- Make something functional → get feedback → improve
- Polish and repeat

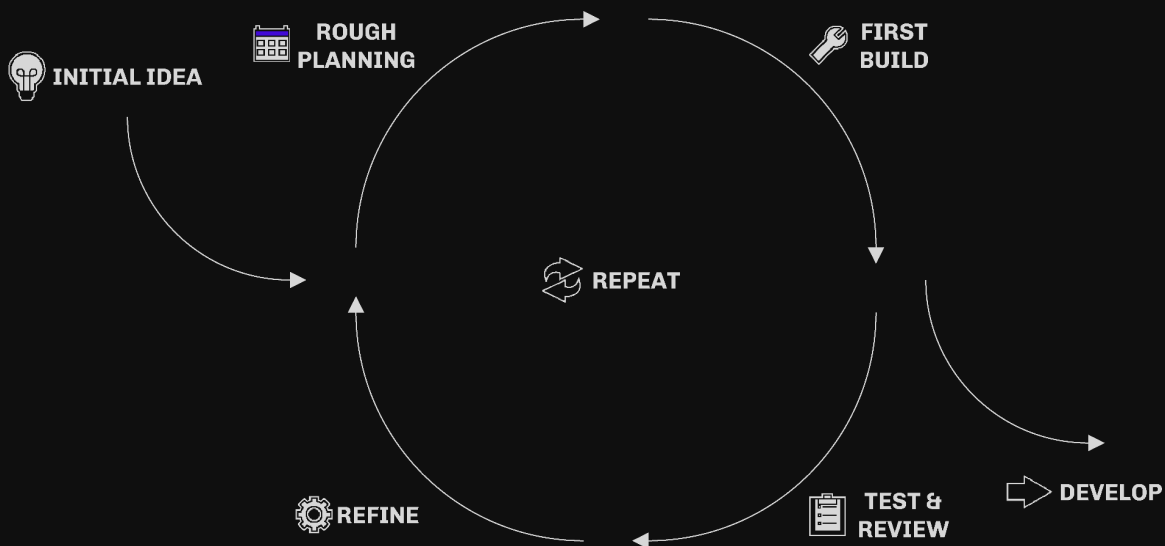
Good for:

- Creative projects that shift as they’re made
- Prototyping new ideas
- Long-term work with room to grow and evolve

Things to keep in mind:

- Needs structure, or it can spiral
- Feedback loops are essential
- You’re never really “done” until you decide to be

Here's how it breaks down:



(Used for big, creative, or evolving projects)

1. **INITIAL IDEA** - You've got the core idea and a rough direction.
2. **ROUGH PLANNING** - Map out tasks, scope, and what we're aiming for in this round.
3. **FIRST BUILD** - Create the first working version, even if rough.
4. **PLAY & TEST** - Try it out, review with the Team, and collect feedback.
5. **REFINE** - Improve what's working, cut what isn't, rework what needs it.
6. **REPEAT OR DEVELOP** - Loop again with new insight and direction. If satisfied, move into full development.

HYBRID

Sometimes the project needs a mix. Maybe one part is super experimental, while another is already nailed down. A hybrid method allows you to treat each part differently, using linear steps where appropriate, and iterating where necessary.

You might follow a clear roadmap for the visual design, while the story or mechanics get built and revised over time. It's flexible without being chaotic.

How it works:

- Pick the method that fits each part of the project
- Keep everyone in sync on how different parts are progressing
- Combine structure with room to experiment

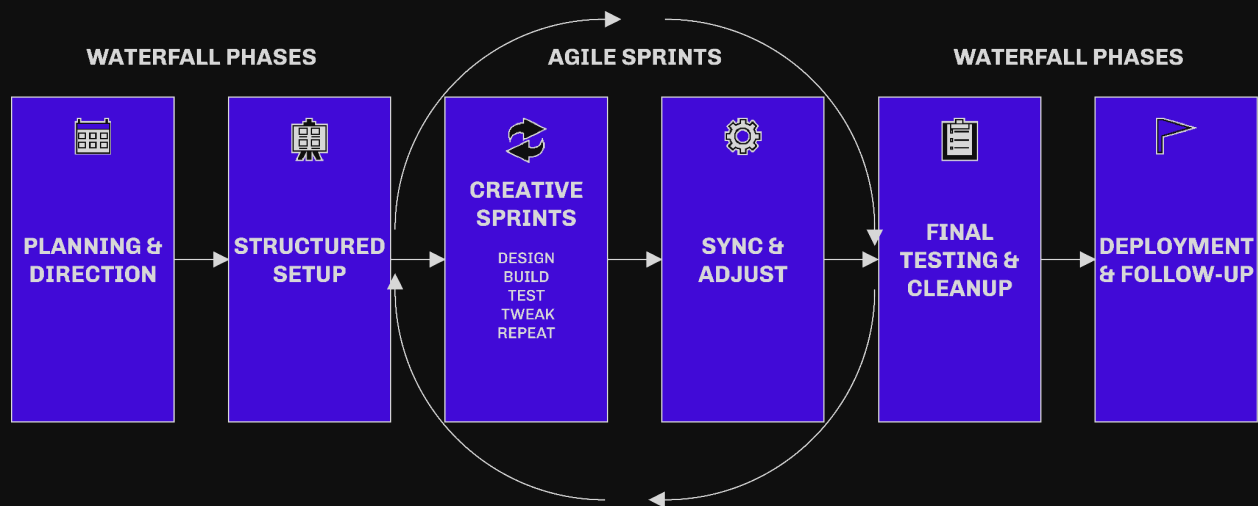
Good for:

- Bigger projects with multiple teams or departments
- Projects that are part-new, part-refined
- Adapting to different workflows or creative needs

Things to keep in mind:

- Needs strong coordination
- Everyone should understand which parts are fixed and which are still evolving
- Miscommunication can cause friction if one part moves faster than another

Here's how it breaks down:



(Used for big, creative, or evolving projects that need both structure and flexibility)

1. PLANNING & DIRECTION

- Get the core idea down
- Set expectations and timeline
- Outline what's flexible and what's fixed

2. STRUCTURED SETUP

- Lock down the parts that don't need experimentation (e.g. visual branding, format, style rules)
- Prep initial tasks and tools
- Establish Team roles and goals

3. CREATIVE SPRINTS

- Work in sprints for anything that benefits from iteration (e.g. story, game mechanics, interactive sequences)
- Design → Build → Test → Tweak
- Repeat as needed based on feedback and gut

4. SYNC & ADJUST

- Weekly check-ins to align progress
- Flag blockers or slowdowns
- Adjust the approach if needed

5. FINAL TESTING & CLEANUP

- Run tests across all departments (visual, audio, technical)
- Smooth out the rough edges
- Prepare for publishing/launch

6. DEPLOYMENT & FOLLOWUP

- Upload, publish, or share
- Make post-launch adjustments if needed
- Reflect and archive

FREEFORM

This is the most unstructured method: no locked-in phases, no set expectations. You move as you go, make things up along the way, and follow whatever direction feels right.

Sometimes, structure is more of a hindrance. It could be a passion project, a weird experiment, or a quick one-person thing. The process isn't defined; the end result is.

How it works:

- Start wherever you want
- Work in whatever order you want
- Let the idea guide the process, not the other way around

Good for:

- Quick solo projects
- Experiments and prototypes
- Projects without deadlines or expectations

Things to keep in mind:

- Easy to get lost or stuck
- No system = no safety net
- Only works if you're self-directed and self-managed

MAYBE ADD CHART OR SUMN HERE

(Used rarely, for small, experimental, or personal projects)

1. **INITIAL IDEA** - You've got something you want to try, no pitch, no plan, just a concept you want to explore.
2. **DO WHAT FEELS RIGHT** - There's no strict order. Perhaps you start by designing first, or maybe you just write, sketch, or build whatever's calling to you.
3. **ADAPT** - The idea will likely change, let it. Be flexible, scrap what's not working, and follow what's interesting.
4. **REFLECT MIDWAY** - At some point, take a step back. What's working? Where's it going? What do you actually want from this?
5. **FINISH WHEN IT FEELS DONE** - Wrap it up when it feels right. Clean up anything messy. If you want to polish, polish. If not, you're good.
6. **OPTIONALLY WRAP-UP** - Decide what to do with it, publish it, archive it, show your peers, or just let it be what it was.

DECIDING METHOD

Here's a quick guide to help you figure out which method fits your project best:

Choose Waterfall if ...

- The project is short, simple, and you know exactly what needs to be done
- You don't expect many changes or revisions
- You just need to get from A to B with minimal fuss

Choose Iterative if ...

- The idea is still evolving or needs to be tested along the way
- The project is big or creatively complex
- You want constant feedback and room to reiterate

Choose Hybrid if ...

- Some parts are locked, others still need exploration
- You're working across departments (e.g. story is flexible, visuals are fixed)
- You need structure, but still want the freedom to iterate where it matters

Choose Freeform if ...

- It's a personal project, passion piece, or quick experiment
- You don't need structure, deadlines, or a Team
- You're exploring, not producing

PROJECT PHASES

Every project we undertake undergoes three main phases: Pre-Production, Production, and Post-Production. Each phase has its own focus, challenges, and purpose, and each one builds on what came before.

This section breaks down what happens in each phase, how we approach it at HELLINGTON, and what to expect along the way.

(chart/timeline illustration here)

PRE-PRODUCTION

This is where we bring the idea to life. Before anyone starts creating, we take time to explore, align, and build a solid direction together.

What it usually includes:

1. **BRAINSTORMING** - We throw around ideas, challenge them, combine them, and start locking in what we actually want to make, and why. The concept pitch is our launchpad, but we build outward from there.
2. **RESEARCH** - We collect references, gather context, and dig deeper into the themes, audience, or formats we're exploring. This helps us stay grounded while pushing creative boundaries.
3. **CONCEPTUALIZATION** - We define the tone, style, and direction of the project. This could mean moodboards, theme prompts, test dialogue, sketches, or visual concepts; it all depends on the medium.
4. **EXPERIMENTATION** - We build rough tests to see if the concept actually works in practice. That might be a design mockup, storyboard, demo, or proof-of-concept scene. Doesn't have to be pretty, just real enough to test.
5. **PLANNING** - We set up a central document (or something similar) that outlines the project's creative and production plan. It could be a full Design Document or just a simpler outline, depending on the scope. Either way, this document becomes our anchor and evolves with the project.

The point of Pre-Production is to get aligned. We don't move forward until we know what we're creating, why we're creating it, and how we plan to bring it to life.

PRODUCTION

This is where the project begins to take real shape. Ideas become assets, scenes, systems, or sequences, depending on the medium. The core of the work happens here.

What it usually includes:

6. **SETUP** - Before diving in, we get everything organized. This involves breaking down tasks, assigning roles, setting up the file structure, and synchronizing tools. *PROJECT MANAGERS* play a significant role here, along with Team Leads in their respective departments.
7. **PROTOTYPING** - With the full concept, plan and setup in place, we now prototype the core elements of the project as a final check before development. This isn't about testing small pieces in isolation, it's about making sure the main mechanics, visuals, systems, or structure all hold up when combined. Whether it's a polished pre-alpha, a full test scene, or a detailed proof-of-concept, this step helps us validate the foundation before committing to the full build.
8. **DEVELOPMENT** - The core of it all. This is where we build, whether it's writing scripts, designing visuals, animating scenes, composing music, or coding gameplay. *WRITERS, VISUAL ARTISTS, DEVELOPERS, SOUND DESIGNER*, and others carry out the production work defined in the Planning phase.
9. **REFINEMENT** - While we develop, we improve. Feedback is gathered regularly, internally from the Team or externally through playtests, previews, or internal reviews. Nothing is ever perfect on the first try, so this stage helps smooth out the rough edges.
10. **DOCUMENTATION** - Throughout this whole phase, we update our logs, timelines, and the central project document. This keeps everyone in sync, shows how far we've come, and helps identify where more focus is needed. Everyone's responsible for logging their progress; it's how we avoid bottlenecks and confusion.

Production isn't just "doing the work." It's about collaborating effectively, staying organized, and building with purpose. Good preparation makes this phase smooth. Good communication makes it sustainable.

POST-PRODUCTION

This is where we clean up, finalize, and reflect on our work. The main content is complete; now we refine, test, and finalize it properly.

What it usually includes:

- 10. TESTING** - For games, that means bug testing. For film, it could be reviewing edits or checking timing. For comics or books, it's proofreading and layout checks. This step is about catching issues before launch, not after.
- 11. POLISHING** - Now that major issues are resolved, we go back and refine the details. This might involve tightening visuals, cleaning up audio, refining writing, or enhancing flow. If something still feels off, awkward, or incomplete, this is the time to fix it.
- 12. ARCHIVING** - Once the project is complete, everyone logs their contributions, updates their portfolios, and helps organize the files. Everything should be tidy and documented in case we return to the project later (or need to show it off). This is also a chance to reflect on what worked, what didn't, and what we'd do differently next time.
- 13. PRESENTATION** - After archiving your work and adding it to your portfolio, you'll have the chance to present it to the Team. This isn't mandatory, but it's highly encouraged. It provides everyone with insight into what went into your contribution, your thought process, and your growth and development. It helps build mutual respect across the Team and allows us all to learn from each other's approaches.

Post-production is about closing the loop. We don't just finish the work, we finalize, test, and archive it with care.

PROJECT TIMELINE

This chapter walks you through the full lifecycle of a HELLINGTON project, from the first brainstorming session to the final polish and showcase. It expands on the Project Phases chapter by digging into the step-by-step flow of how we work, how roles interact, and what deliverables are expected along the way.

You'll find a detailed look at each stage of development, examples across different mediums, and tips on how to stay aligned, adapt your workflow, and keep things moving forward.

All templates, guides, and extended documentation will be linked in the relevant section, along with a comprehensive overview in the [RESOURCE LIBRARY](#) chapter.

HOW TO USE OUR PROJECT TIMELINE

Sometimes, the steps to a project aren't like rigid checklists. They are more like building blocks. Depending on the format, scale, or creative flow, you might rearrange steps, skip some entirely, or revisit others more than once. The blocks can be reused, remixed, or stacked differently depending on what the project calls for.

That said, most projects still move through the same three main phases. It's usually the steps within those phases that shift around. Use the timeline as a flexible tool to shape your process, not a rulebook to follow line by line.

Need help figuring out what your project actually needs? Check out our guide for assessing scope and scaling your process:

[PROJECT SCOPE & STEP SCALING GUIDE](#)

NOTE ON TEAM SCOPE

We only forward projects that match the skills and capacity of our current Team. We don't build projects around roles we wish we had. Since we're not profit-driven, we create what we can, not what we hope to outsource.

PRE-PRODUCTION

Before anything is built, it's shaped. Pre-production is where we explore ideas, define direction, and establish a strong foundation that will carry the entire project.

1. BRAINSTORMING

This is where we start expanding the project pitch. The core idea is already there, now we explore what it can become.

That could mean shaping the world, defining the tone, figuring out key mechanics, or outlining the characters. It's about exploring what makes the idea interesting, finding the angles that stand out, and building creative momentum.

IDEATORS usually lead the process, but *WRITERS*, *VISUAL ARTISTS*, and others often join in early as well. Whether you're doing this solo or as a Team, the goal is the same: generate ideas, test directions, and stay open.

TIP: Keep the project in the back of your mind, even when you're not actively working on it. Ideas tend to show up when you least expect them.

What makes brainstorming worthwhile?

- **NO JUDGMENT UPFRONT** - All ideas are welcome at first. We'll figure out what sticks later.
- **MORE IS BETTER** - Even the bad ideas help shape the good ones. Just keep going.
- **IDEAS BUILD IDEAS** - You can often improve someone else's idea just by bouncing off of it.
- **IT'S ABOUT EXPLORING** - Not every idea will make it in, but exploring them helps shape the final direction.

This stage is loose on purpose. We're still figuring things out.

BRAINSTORMING METHODS

There's no one way to brainstorm. These are just some methods that tend to work well:

- **MIND MAPPING** - Start with the pitch in the middle. Expand outward with related ideas, characters, themes, visuals, story beats, mechanics, and see what branches connect.
- **FREEWRTING** - Just write. Don't think too hard. You'll be surprised what comes out when you stop trying to be "right."
- **WHAT-IF CHAINS** - Ask "what if ..." and keep going. Follow where the idea leads. This works well for story ideas or world-building.
- **THEME EXPLORATION** - Break down the theme and think about how it could show up in tone, visuals, character arcs, or gameplay.
- **CHARACTER WEBS** - Think about how your characters could relate, interact, or clash. Connections between them often reveal the story and motivation.
- **MOODBOARDING** - A visual board with colors, textures, art, references, or music that captures the feeling of the project. It helps align the tone early on.
- **PROMPT CATALYSTS** - Short, focused prompts that encourage your thinking to delve deeper within the pitch's concept. These can help with specific aspects, such as character ideas, lore, settings, or dilemmas.
Example: "A villain who used to be good" or "A place no one comes back from."
- **FIRST PRINCIPLES BREAKDOWN** - Instead of working with existing genre tropes or story formats, break the concept down to its most basic truths. Then build from scratch with logic and creativity.
Example: "What's really required to make this story work?"
- **SIX THINKING HATS** - Each "hat" represents a different perspective: logic, emotion, caution, optimism, creativity, and process. It helps teams examine an idea from all angles without getting stuck in a single mindset.
- **SCAMPER** - A structured method where you improve or reshape ideas by asking: Substitute, Combine, Adapt, Modify, Put to another use, Eliminate, Reverse.
Example: "What happens if we reverse the villain's role?" "What if we combine two settings?"
- **RAPID-FIRE ROUNDS** - Set a timer and push out as many ideas as possible around one question or problem in 3–5 minutes. Helps reduce overthinking and unlock fast, weird ideas.
- **EMPATHY MAPPING** - A tool for stepping into your audience's (or character's) head. What do they think? Feel? See? Hear? Helps build deeper, more grounded ideas.
- **TENSION MAPPING** - Map out opposing forces or conflicts at play, either within a story, character, setting, or theme. Conflict creates drama, which creates interest.

For more ideas and in-depth breakdowns, check our [BRAINSTORMING & IDEATION METHODS](#) Guide, which provides detailed information on specific techniques and exercises.

BRAINSTORMING IN TEAMS

When you're brainstorming with others, it can get messy fast, and that's okay. The trick is to keep it creative without losing sight of the good stuff. Here's how we recommend handling it:

- **START IN YOUR OWN SPACE** - If you're joining a group session, start by writing down your own ideas in a personal document. This gives you room to think without interruption and ensures nothing gets lost.
- **USED A SHARED DOCUMENT** - After gathering your own thoughts, copy them into a shared brainstorming document. Everyone does the same, which creates a collective pool of ideas.
- **ORGANIZE AS YOU GO** - Once everything's dumped into the shared doc, go through it together. Group similar ideas, add comments, highlight what stands out, and question what requires further consideration.
- **DON'T DELETE, ARCHIVE** - If something doesn't make the cut, move it to the bottom of the document or into an "extra" section. Just because it doesn't fit now doesn't mean it won't be useful later.
- **DOCUMENT CHANGES** - If a rough idea turns into something more refined, copy it into a separate "refined concepts" section. This helps track how your thinking has developed.

Need help setting this up? We've created a few brainstorming templates to get you started, whether you're working solo or as a Team:

BRAINSTORMING TEMPLATES →

This system ensures that everyone's voice is included while keeping the process clean and trackable, which is especially useful when revisiting things later in planning or documentation.

TOOLS WE USE

- **GOOGLE DOCS** - To write down loose ideas or outline everything you've been thinking about
- **MIRO & MILANOTE** - For visual brainstorming or mapping complex connections
- **PINTEREST** - For collecting and organizing visual tone references
- **DISCORD** - For fast-paced group brainstorming, voice calls, and bouncing ideas off teammates

You can learn more about these tools in the **TOOLS** chapter, which includes information on how we utilize them.

WRAPPING-UP

Once your ideas are flowing and directions are starting to take shape, it's time to slow down and strengthen the foundation with research.

2. RESEARCH

Research is a vital part of the process. It helps us make informed creative choices, represent things accurately, and build worlds that actually make sense. If you want your concept to take hold, rather than fall apart the moment someone challenges it, this is where it begins.

This step, while led by *RESEARCHERS*, often includes input from *IDEATORS*, *WRITERS*, *VISUAL ARTISTS*, and sometimes *TECHNICAL DESIGNERS*, depending on what's being explored. While everyone can contribute, these roles are primarily responsible for pushing research that informs core decisions moving forward.

WHY WE RESEARCH

Before we dive into how we do it, let's be clear on why we do it in the first place. These are the driving purposes behind research at HELLINGTON:

- **TO MAKE INTENTIONAL CREATIVE CHOICES** - We don't rely on gut feeling alone. When we know how things work, we make better decisions. Everything from worldbuilding to tone becomes more purposeful.
- **TO REPRESENT THINGS ACCURATELY** - If the project touches on real cultures, time periods, or systems, we want to approach them with care. That means getting familiar with the facts so we don't misrepresent or oversimplify.
- **TO GIVE FICTION STRUCTURE AND CONSISTENCY** - A fictional world needs rules, too. Research helps us determine what those are, how they function, and how they influence the story. That's how a made-up world still feels real, by basing it on what is real.
- **TO INSPIRE BETTER IDEAS** - Research is a breeding ground for creativity. You might discover strange facts, forgotten history, or design details that completely reframe your project, in the best way.

TYPES OF RESEARCH

There's no one way to research. The type of research you do depends on the project's focus and what you need to understand better before you move forward.

Here are a few types of research you'll likely engage in:

CONTEXTUAL RESEARCH

Understand the bigger picture. This could refer to the world, time period, or culture in which your story is set, whether real or fictional. You're aiming to ground your ideas in something that makes sense, feels believable, or hits the right tone.

USEFUL FOR

Worldbuilding, tone, settings, backstory, immersion

ROLES

IDEATOR, NARRATIVE DESIGNER, VISUAL ARTIST

THEMATIC RESEARCH

Dive into topics tied to your project's themes. If you're working on a project that explores memory, justice, or identity, consider how these concepts manifest in the real world and media. This helps deepen the meaning and add layers of complexity.

USEFUL FOR

Symbolism, story depth, tone, message

ROLES

IDEATOR, STORYWRITER, NARRATIVE DESIGNER

MECHANICAL RESEARCH

How do similar systems work? What makes a mechanic engaging or frustrating? Whether you're building a branching dialogue system, combat loop, or interactive scene, studying how others have done it can save you time and bad design choices.

USEFUL FOR

Gameplay, UX, systems, pacing

ROLES

GAME DESIGNER, SYSTEMS DESIGNER, PROGRAMMER

AESTHETIC RESEARCH

This includes visual mood, art style, lighting, sound, or even layout references. It could be period-accurate, genre-inspired, or something uniquely experimental. Gather examples that reflect the look, feel, and sensory tone you want to build toward.

USEFUL FOR

Visual direction, audio identity, creative tone

ROLES

VISUAL ARTIST, GRAPHIC DESIGNER, SOUND DESIGNER, COMPOSER

REFERENCE GATHERING (REAL & FICTIONAL)

Sometimes you need to pull from both ends, realism and imagination. Maybe you're researching the Appalachian region for a horror setting. One teammate might look into the real geography and history, while another explores folklore, myths, and legends. Both are valid, and together they form a richer whole.

USEFUL FOR

Hybrid settings, speculative fiction, believability

ROLES

RESEARCHER, IDEATOR, NARRATIVE DESIGNER, STORYWRITER

RESEARCH STRATEGIES & METHODS

You don't need to be an academic to do good research. You just need a solid approach, one that keeps you focused, organized, and digging in the right places. This section breaks down different research strategies that help your creative process, along with tips for staying on track.

1. DEFINE YOUR FOCUS

Start by narrowing down what you actually need to know. This could be a theme, a mechanic, a setting, or a concept you want to build out. If you're stuck, try asking:

- What feels unclear or underdeveloped right now?
- What's too vague or surface-level?
- What's the missing piece between our idea and something that makes sense?

For creative teams, this doesn't always mean forming a "research question"; sometimes it's just identifying what needs more depth.

Example: You're making a story set in a fictional cult. Maybe you research real-world cult psychology, group dynamics, or obscure belief systems for inspiration.

Write down a few focused goals. They'll help guide you as you search. If you're stuck, try mind mapping, asking "why?" until you hit something meaningful, or listing out what's missing from the pitch or current concept.

2. EXPLORE & GATHER

Now that you know what you're looking for, it's time to start digging. This is where you explore your topic from every angle, collecting sources, examples, and insights that can help shape your decisions moving forward.

You're not here to skim and move on. You're here to build a solid base of understanding.

Here are a few ways to go about it:

- **GUIDED BROWSING** - Use targeted keywords tied to your concept and keep going past the first search result. Dig deeper than surface level.
- **COMPARISON BOARDS** - Find 3-5 different takes on a similar idea (say, vampire depictions across media) and break down how they differ, tone, style, logic, etc.
- **SOURCE STACKING** - Don't rely on one article or video. Cross-reference multiple angles to build a more comprehensive picture, especially for subjects such as history, psychology, or belief systems.
- **FLAG & TAGGING SYSTEMS** - Label what you find. Create buckets like tone, mechanics, world logic, or visuals. Makes it way easier when you start organizing your findings into concept documents later.

Where to look:

- Articles, essays, and interviews
- Books, journals, and academic sources
- Documentaries, podcasts, and lectures
- Firsthand accounts, blogs, or wikis
- News pieces, cultural references, or creative works

3. CHOOSE YOUR RESEARCH METHODS

What you're working on, and the kind of answers you're after, will shape how you go about it.

Some examples:

- If you're exploring culture or storytelling, consider using qualitative methods, such as comparative reading, myth studies, and community forums, to gain a deeper understanding.
- If you're examining behavior, data, or patterns, lean into quantitative methods, such as statistics, charts, timelines, and breakdowns.

If you're not sure, mix it. That's often the best route.

4. ORGANIZE & TRACK SOURCES

Keep everything accessible. Always track where your sources came from, it helps you:

- Avoid misinformation
- Reference things properly in your documents
- Double-check facts later if needed

Use it at the end of your document or a separate document, spreadsheet, or built-in comment system to credit links and summarize key takeaways. Don't just copy-paste blocks of text.

5. INTERPRET & APPLY

Once you've collected enough, zoom out and ask:

- What patterns or contradictions do you see?
- What surprised you?
- What helped reshape the concept or confirm something?

This is where your research turns from “info dump” into something meaningful. The goal isn't just to collect facts, it's to inform the creative direction. Use what you've learned to shape design, story, tone, mechanics, or whatever part of the project you're helping build.

Make sure your insights carry over into your Planning documents or Design Document drafts. Link research where it matters. Use quotes, visuals, or summaries to support your ideas. Let the research speak through the choices you make.

6. PRESENT YOUR FINDINGS

If you're working with a Team, keep your research organized and easy to navigate. Summarize the key points in your concept document, then link to a separate research document where the full details can be found.

You can find more in-depth explanations for strategies and guidance on how to do comprehensive research in our complete [RESEARCH STRATEGIES](#) Guide.

RESEARCHING AS A TEAM

Research doesn't always happen solo. Often, it's collaborative, especially when the project has many moving parts. Here's how to keep things organized and effective.

- **SPLIT THE WORKLOAD WITH PURPOSE** - Assign people to different topics based on their roles or strengths. Someone handles visuals, another dives into tone, and someone else tackles systems.
- **USE DUAL MINDSETS IN SMALL TEAMS** - If you're working as a duo, try assigning one person to focus on grounded and factual research. At the same time, the other explores fictional or speculative elements. This approach works exceptionally well when exploring myths, legends, or cultural beliefs that are built on both real and imagined phenomena.
- **START PERSONAL, THEN GO SHARED** - Record your own findings first however you like; your notes don't need to be polished. Then move what's worth sharing into a shared doc or space.
- **STAY ORGANIZED** - Use headers, tags, folders, and visual reference boards to keep everything easy to navigate. Clutter kills momentum.
- **EXPLAIN YOUR FINDINGS** - Don't just dump links. Add context. Why is this research relevant? What insight does it give us for our concept?
- **REFLECT AS A TEAM** - Once research is collected, go over it together. Talk about what clicked, what didn't, and what direction it's pointing you in before jumping into conceptualizing.

RESEARCHING TOOLS & TEMPLATES

We've got a few tools and templates to help keep your research organized, accessible, and actually useful down the line. You're not required to use all of them, just the ones that make sense for your project and workflow:

- **RESEARCH DOCUMENT TEMPLATE** - This is your go-to if you want a more structured format for research. It includes sections for your focus questions, collected sources, key takeaways, and how it connects to the project. Good for both solo and team research.
- **RESEARCH LOG TEMPLATE** - Use this to track everything you're finding. It helps you link sources, write down relevant notes, and summarize why the information matters. Think of it like a personal knowledge base for the project.
- **FLAG & TAG REFERENCE SHEET** - We use this system a lot during early-stage research. It helps you categorize your material (like visuals, mechanics, world logic, etc.) so nothing gets buried or forgotten. Especially helpful when you revisit things during conceptualization.
- **SOURCE STACK SHEET** - Not all sources say the same thing, and that's the point. This doc helps you compare different perspectives, map contradictions, and figure out what makes the most sense for your concept.
- **BRAINSTORMING & RESEARCH WORKSPACE** - Sometimes, you're bouncing between ideas and research constantly. This combo sheet helps you organize both in one place. Use it if you're still figuring out what your concept needs.

SUGGESTED TOOLS

We recommend *GOOGLE DOCS* or *NOTION* for text-based research, and *MILANOTE* or *PINTEREST* for visual boards. You'll find more in the **TOOLS** chapter if you want to explore what's approved for use.

3. CONCEPTUALIZATION

This is where we take everything we've been building, ideas, inspiration, and research, and start shaping it into something real. We're not finalizing every detail yet, but we're making decisions. This is where we define the concept in a way that can guide the rest of the Team forward.

The goal? A clear, creative outline that captures what the project is, what it's trying to do, and how it's going to stand out, without diving into full production planning just yet.

This step is inherently collaborative; *IDEATORS*, *RESEARCHERS*, *WRITERS*, *DESIGNERS*, and *VISUAL ARTISTS* may all be involved depending on the medium and direction the concept is heading in.

REFINING THE CORE CONCEPT

By now, your project has evolved. Between brainstorms and research, it's likely taken on a new shape, maybe a sharper tone, a stronger hook, or a deeper theme. This is your moment to revisit the original pitch and tighten the vision.

Start by asking:

- What's still working from the pitch?
- What has changed or improved based on what we've explored?
- What direction feels most promising or exciting?

This doesn't mean throwing everything out. It means sharpening the edges. Focus on the idea until it feels solid enough to build on, but open enough to allow for creative flexibility down the line.

Use your research to ground your decisions. Revisit things like:

- What feels authentic, realistic, or fitting for the world?
- What parts of the idea need clarification or simplification?
- What makes this idea ours, and not something anyone else could make?

This sets the tone for the rest of the step. Once the concept is refined, we can start building around it.

Use our template as a tool to help you summarize and pressure test your concept before moving on to nailing the creative direction:

CONCEPT REFINEMENT TEMPLATE →

DEFINING THE CREATIVE DIRECTION

With your core concept locked in, it's time to make some creative calls. You're not laying out every detail yet, but you are shaping the identity of the project, the stuff that will guide future decisions, inspire teammates, and keep everything aligned as things grow.

Some elements you'll want to define:

- **TONE** - What overall feeling or vibe does the project give off? Is it dark and serious, light and humorous, mysterious and eerie, fast-paced and intense, or calm and emotional?
- **THEMES** - What's the heart of the story? What ideas or questions are we exploring?
- **VISUAL MOOD** - Any clear aesthetic? What kind of world are we seeing? Realistic or stylized?
- **NARRATIVE/STRUCTURAL FEEL** - What kind of story are you telling, and how is it told? Is it focused on characters and their journey? Does it jump around in time or stay in a straight line? Is it tightly structured or more open and exploratory?

This step isn't just for writers or artists; it sets the vibe for everyone. A solid creative direction gives designers, editors, developers, and others a shared language to work from. Think of it like moodboarding, but across multiple layers, visual, emotional, structural, and even philosophical.

Use our template to organize your moodboard, define tone and theme, and collect reference material in one place:

[CREATIVE DIRECTION TEMPLATE →](#)

SUMMARIZING THE STRUCTURE

Now that your direction is taking shape, it's time to outline how the project will function at a high level. This doesn't mean writing a full script or design document; it just means breaking the idea into understandable parts.

Think big-picture structure. How does the concept unfold? What's the rhythm or format? If it's a narrative piece, maybe it's a three-act arc. If it's a game, maybe it's a loop or level progression. If it's a mixed-format concept, think in phases, beats, or episodic flow.

This part of the process is typically shaped by *NARRATIVE DESIGNERS* laying out story arcs, *GAME DESIGNERS* sketching out mechanics or loops, and *IDEATORS* guiding how it all fits together. Even *STORYWRITERS* might chime in here to define key moments or themes. The goal is to give everyone a clear sense of how the project is meant to flow, without diving into detailed planning just yet.

What this might include:

- Act or chapter breakdowns
- Narrative beats or gameplay loops
- Flow diagrams or story maps
- Scene or section summaries
- Interactive structure or branching paths (if applicable)

Use our template to map out the project's flow, structure, and pacing across formats:

STRUCTURE & FLOW TEMPLATE →

HIGHLIGHTING THE UNIQUE ELEMENTS

What makes this idea a HELLINGTON project? Now that the concept is taking form, it's time to zoom in on the pieces that make it stand out. This is where you clearly define your Unique Selling Point (USP), but also where you unpack the deeper traits that give your project identity and edge.

It's not just about being different. It's about being intentional, impactful, and creatively bold. Think of this as your project's creative fingerprint.

What to include:

- A clear USP in one sentence
- Key creative risks or stylistic flourishes
- Psychological themes or narrative subversions
- How it evokes, provokes, or impacts (see [PROJECT FORMULA](#))
- Any aspects that lean into or break from genre conventions

This section should help others “get it” fast. It’s the answer to: “Why this project, and why now?”

You can use our template to help pinpoint your project's edge and define what gives it a HELLINGTON theme:

[PROJECT USP TEMPLATE →](#)

CREATING THE OUTLINE DOCUMENT

Now that your concept has taken shape, it's time to package it into a clear and usable outline. This isn't just for you; this is for everyone else who might join the project later. It should communicate what the project is, what it aims to do, and what makes it stand out, all in a way that's digestible and easy to build on.

This is not your full Design Document. Think of it as a bridge between the pitch and the full plan. It's a snapshot of the creative core, flexible enough to grow, but defined enough to guide future decisions.

What to include in the outline:

- A clear summary of the project concept and goals
- The defined creative direction: tone, theme, visual style, and format
- A high-level structure of the story, gameplay, or flow
- Your unique angle: what makes this a HELLINGTON project (see [THE PROJECT PASS ICEBERG](#))
- **Optional:** key characters, scenes, mechanics, or inspirations

Once written, this outline becomes your anchor. If things shift later (and they will), you'll always have this to return to, the purest version of what you set out to make.

In smaller projects, this outline might be all you need; no need for a full-blown Design Document. But for anything bigger, it sets the foundation for the next phase: Planning.

Use our template to organize your summary, creative direction, and high-level structure in one place:

[PROJECT OUTLINE TEMPLATE →](#)

4. EXPERIMENTATION

Before we start planning full-scale production, we need to make sure the concept actually works. That's what experimentation is for. It's not about making something polished, it's about building rough, small-scale tests that help us see if our ideas hold up in practice.

This is where we stress-test the parts of the concept we're still unsure about. Maybe it's a gameplay mechanic, a visual style, or an interactive story moment. Whatever it is, this step helps us avoid wasted effort later by figuring out what's solid and what needs fixing early on.

Not every project needs early experimentation. But for bigger, more experimental, or interactive ideas, it's a smart checkpoint before moving into planning.

IDENTIFYING WHAT NEEDS TESTING

Before building anything, step back and ask: What are we unsure about? Not every part of the concept needs prototyping. The goal here is to pinpoint what actually needs to be proven.

This usually falls into one of a few categories:

- **MECHANICS OR SYSTEMS** - If the project involves gameplay, interactivity, or logic systems, test the core loop or dynamic. Will it feel good to play? Is it understandable? Is it engaging over time?
- **NARRATIVE OR EMOTIONAL BEATS** - If the experience leans on story or emotional impact, you might need to test how those moments land. Is the arc compelling? Does it hit the tone you're aiming for?
- **VISUAL STYLE** - If your direction is stylistically bold, untested, or experimental, test the art direction, framing, animation, or moodboards. Does it capture the feeling you want?
- **FORMAT OR FLOW** - Some projects need to prove their format, like whether an episodic structure, interactive layer, or split narrative actually works in practice.

You don't have to test everything at once. Start by identifying the riskiest parts of the concept, the stuff that could make or break the project if it doesn't work. Those are your priorities.

Use our template to map out what needs testing, why it matters, and how you might approach testing it:

[EXPERIMENTATION PRIORITY PLANNER TEMPLATE →](#)

MAKING LOW-STAKES EXPERIMENTS

Once you've identified what needs testing, it's time to build it. This doesn't mean jumping into a full prototype, we're keeping things small, scrappy, and focused.

The goal here is to create fast, low-pressure test pieces that help you answer specific questions. Think of these as mini experiments. They're not for show, they're for clarity. You're testing tone, function, feeling, not polish.

What this might look like:

- **MECHANICS** - A quick greybox version of a core system or gameplay loop (e.g. basic movement, obstacle and goal).
- **NARRATIVE** - A rough storyboard or mock scene to test a dialogue style or emotional beat.
- **VISUALS** - A small collection of frames, animations, or mock UI screens to try out the style, mood, or composition.
- **FORMAT & PACING** - A short edit or flow diagram showing how the structure would play out across episodes, levels, or pages.

These are usually built separately, it's rare that everything clicks together right away. You're testing ingredients, not baking the whole cake yet.

You may see roles like GAME DESIGNERS, NARRATIVE DESIGNERS, VISUAL ARTISTS, STORYBOARDERS, TECHNICAL ARTISTS, and COMPOSERS take the lead here, but it all depends on what's being tested.

Use our worksheet to help define what you're trying to test, how you're going to build it, and what success looks like:

[EXPERIMENTATION PLANNING TEMPLATE →](#)

GATHERING REACTIONS

Once you've got something working, even if it's rough, it's time to put it in front of a few trusted eyes.

This isn't about public feedback or polish, it's about getting honest reactions from people who understand the vision and can help you assess whether your experiments are hitting the mark. Ideally, this means team members who weren't directly involved in the tests. They bring fresh eyes.

Ask targeted questions like:

- What did you feel?
- What confused you?
- What stood out, for better or worse?
- Does this capture what we're aiming for?

You're not looking to impress, you're looking for insight. Did the mechanic feel natural? Did the tone come through? Did the moment land emotionally?

For more sensitive or story-driven tests, a smaller group of 2–3 reviewers is usually enough. In more functional tests (like gameplay systems), you might benefit from a slightly larger test group.

PROJECT LEADS often help facilitate feedback sessions or review notes, but everyone involved in shaping the direction, like IDEATORS, DESIGNERS, or WRITERS, should be part of interpreting the feedback.

If you want to formalize the process, you can use the following template to gather structured feedback:

[FEEDBACK SNAPSHOT TEMPLATE →](#)

ITERATING AS NEEDED

Not every experiment will work the first time. That's the point. This step is about making changes, trying again, and figuring out how to strengthen the parts that didn't land.

You're still in a flexible phase, so take advantage of that. Nothing is too precious yet.

Some things to keep in mind:

- Did your tests reveal anything unexpected?
- Were certain ideas stronger in theory than in execution?
- Is it a quick fix or something deeper that needs rethinking?

Iterating doesn't mean endlessly tweaking. It means taking the insight you got from feedback and making focused improvements. In some cases, that might mean refining the prototype. In others, it could mean shifting direction slightly, or even scrapping something that's not working.

It's also a good time to update your Concept Outline or start drafting key points for your Design Document, based on what you've learned.

This is a team effort. DESIGNERS, WRITERS, ARTISTS, COMPOSERS, and other creatives should work together with the PROJECT LEAD to figure out what stays, what changes, and what moves forward into planning.

Use our template to track what you tested, what worked, what didn't, and how you're improving it:

[ITERATION LOG TEMPLATE →](#)

5. PLANNING

Turning vision into a buildable blueprint.

You've got a strong concept, tested your assumptions, and now it's time to get serious about making it real. Planning involves expanding the outline into a fully detailed Design Document (or DD), breaking the project into phases, and figuring out how we're going to build what we've envisioned, within the time, tools, and Team we actually have.

Not every project needs a massive document. But every project does need clarity, structure, and alignment. That's what this step is for. We don't start producing until the plan is solid.

SETTING UP THE DESIGN DOCUMENT

Your project's blueprint.

Now that the concept is solid, it's time to map out how it's going to be executed thoroughly. The Design Document (DD) is where we turn vision into structure, outlining the creative direction, key features, content, timeline, Team roles, and production approach in one central place. This document becomes our foundation during development, keeping everyone aligned and on track.

It doesn't need to be perfect or final. It's a living document that can evolve over time, but it should be complete enough to guide production from start to finish.

READ OUR IN-DEPTH GUIDE

We've created an in-depth Design Document Guide that walks you through each section. It covers what to include for all project types, from outlining Team roles, creative direction, and storytelling, to defining mechanics, tone, milestones, and production scope. It also includes medium-specific breakdowns for games, films, books, comics, and more.

[DESIGN DOCUMENT GUIDE →](#)

THEN USE THE TEMPLATE

Once you understand how it works, use our template to structure your project document from top to bottom.

[DESIGN DOCUMENT TEMPLATE →](#)

TIMELINE & MILESTONES

Planning the path forward.

Now that the project is fully scoped in the design document, it's time to map out how it'll unfold. This step is about more than just marking dates on a calendar; it's about breaking the work into realistic phases, setting checkpoints, and keeping the Team aligned as things move forward.

Not everything needs to be scheduled down to the last task, that's for the setup at the start of production. The focus here is on defining key milestones that guide production, post-production, and any subsequent activities. These can include internal reviews, Team check-ins, public deliverables, or feedback rounds, whatever helps track progress and maintain momentum.

What to include:

- **PHASES** - Development, polish, testing, release, etc.
- **MILESTONES** - Major deliverables like demos, trailers, first cuts, or drafts
- **CHECK-INS** - Internal review windows or retrospectives
- **DEPENDENCIES** - Highlight which tasks rely on others to be completed first
- **BUFFER TIME** - Build in space between phases to allow for delays or iteration

A flexible timeline beats a rigid one. Keep scope, Team availability, and tool limitations in mind. PROJECT LEADS usually take the lead here, but should collaborate with the core contributors to make sure the plan is actually doable.

Use our Project Timeline Guide to help with this step. It walks you through how to break down deliverables, structure a flexible timeline, and align your milestones with the way your Team works. Includes examples across different types of media.

[PROJECT TIMELINE GUIDE →](#)

Once you understand how to structure the timeline, use our template to map out your internal roadmap and key dates:

[PROJECT TIMELINE TEMPLATE →](#)

TEAM ASSIGNMENTS & SCOPE

Figuring out who's doing what, and how much they can handle.

With the plan shaping up, it's time to lock in roles, responsibilities, and workload. This step ensures everyone knows what they're contributing, when they're expected to deliver it, and how their work connects to the rest of the project.

This is where the PROJECT LEAD collaborates closely with the Team to:

- Assign roles based on availability, skillset, and interest
- Define what deliverables each person is responsible for
- Make sure the overall workload is balanced and realistic
- Flag any dependencies (e.g. sound can't start until the animation is drafted)
- Identify any potential capacity issues early

This isn't just a to-do list. It's a structure that helps keep the Team organized and accountable throughout production.

Some things to consider:

- Is the scope manageable with your current Team?
- Are any key roles missing or overloaded?
- Do responsibilities line up with the timeline?

Use our Team Planning Template to organize who's doing what, map deliverables, and connect team members to specific milestones:

[TEAM PLANNING TEMPLATE →](#)

TOOL & WORKFLOW PLANNING

Making sure the pipeline makes sense.

Now that you know what you're making and how big it is, it's time to figure out how the work will flow. That means deciding what tools the project will rely on, how the pipeline will work, and how TEAM MEMBERS will stay aligned as production kicks off.

This isn't about setting up folders or installing software just yet; that comes later. Right now, the focus is on identifying needs and making informed decisions before diving into the build.

Some things to cover:

- Which tools are required for writing, art, sound, dev, feedback, etc.
- Where everything will be stored and tracked (e.g. Drive, Notion, GitHub)
- How the pipeline works: Who hands what off to whom, and in what format?
- What needs version control, what doesn't, and how it'll be handled
- Who needs access to what, and how permissions will be managed

This step is handled primarily by PROJECT LEADS, TECHNICAL ROLES, and anyone responsible for asset handoffs or tool-based workflows. In collaborative projects, it's helpful to bring in other departments to make sure their needs are met.

Use our Tool Setup Planning document to help identify tool needs, organize the pipeline, and note any technical dependencies or limitations:

[TOOL & PIPELINE PLANNING TEMPLATE →](#)

PRODUCTION

This is where things get real. Production is when we take everything we've planned and start building it into a working, playable, viewable, or tangible product. It's where ideas turn into assets, systems, scenes, and sequences, and the project starts to take its final form.

6. SETUP

Putting the plan into motion.

This is where we shift from thinking to doing. After all the outlining, decision-making, and prototyping, it's time to prep the workspace. Setup involves organizing everything so that production can run smoothly, including folders, tools, systems, pipelines, and communication channels.

It's the calm before the storm. A moment to make sure everyone knows where things are, what they're using, and how to move forward, before real work kicks off.

FOLDER & FILE SETUP

Before anyone creates a single asset, we need to establish a clear structure for organizing everything. This step involves setting up a consistent, shared file system that the entire Team can follow, encompassing visuals, writing, code, audio, and any other relevant content.

You'll want to:

- Set up a shared project folder (typically in Google Drive or a synced cloud solution)
- Create consistent subfolders for each department or content type
- Define a simple naming convention (versioning, dates, initials, etc.)
- Set up templates for common file types (e.g. PSD, DOCX, TXT, AE)

The goal is to make things easy to find, update, and track. No more digging through mystery folders or duplicated files. If multiple projects are running, make sure each has a clean structure with no overlap or confusion.

For a full breakdown of how we organize our folder structure and naming system, see the [DOCUMENTATION](#) section in the GUIDELINES chapter.

Once you're ready to apply it, use this template as your base structure:

[PROJECT FILE STRUCTURE TEMPLATE →](#)

TOOL & SOFTWARE CONFIGURATION

Pick your tools, then make sure everyone's using them right.

Before production begins, the team needs to agree on what tools and software will be used, and how they'll be used. If you're confused, earlier in the planning step we determined what tools were needed, now we determine which will be used. Anyway, this step ensures compatibility, consistency, and a smoother workflow across the board.

This includes:

- **CORE PRODUCTION TOOLS** - Game engines, editing software, DAWs, illustration tools, etc.
- **COLLABORATION PLATFORMS** - Shared drives, task boards, calendars, and chat tools.
- **FILE FORMATS & EXPORT SETTINGS** - Agree on what's standard to avoid issues later.
- **PIPELINES & HANDSOFF** - For projects with multiple departments (e.g. concept to modeling to rigging), define how files move between people.

Take the time to do it now so you don't waste time fixing it later.

How we set this up:

See our [TOOL PIPELINE GUIDELINES](#) for detailed recommendations, best practices, and team-specific setups.

TASK BOARD SETUP

Turning the plan into actionable tasks.

With the production timeline and deliverables in place, it's time to break things down into actual tasks based on our project timeline from planning. This is where a task board comes in, where we visualize and track the timeline. It gives the team a shared place to track progress, see what's next, and know who's responsible for what.

We typically use a tool like Notion for full customizability and control. The key is clarity. You don't want people guessing what to do next or where to find something.

A typical task board setup includes:

- Columns or statuses like "To Do," "In Progress," "Review," and "Done"
- Tasks broken down from milestones or the timeline
- Clear assignees for each task
- Deadlines, dependencies, and notes when relevant
- Tags for roles, categories, or priority

Each task should be small enough to be actionable, but meaningful enough to track progress. If something feels vague, break it down.

PROJECT LEADS usually create and maintain the board, but everyone on the team should be regularly checking in, moving their tasks, and updating progress as they go.

TEAM ONBOARDING

Getting everyone on the same page before production kicks off.

By this point, the plan is in place, the tools are ready, and the folder structure's set up. Now it's time to make sure the entire team is aligned.

Everyone should know:

- What the project is
- What their role is
- Where everything is
- What's expected next

This part of the setup isn't about creating, it's about clarity. A short kickoff meeting can go a long way. It helps introduce TEAM MEMBERS to each other, walk through the setup, and address any confusion before things get started.

What this usually involves:

- Giving access to the tools, folders, and boards
- Walking through the timeline, milestones, and first tasks
- Reviewing roles, collaborators, and points of contact
- Answering questions about direction or structure
- Optional: running a mini "alignment session" where everyone reflects on the core concept and how their role contributes

For most projects, we also set up a dedicated Discord server. This becomes the central hub for communication, announcements, and collaboration. It also acts as the entry point for new contributors: If someone joins the project later, they'll be onboarded through this server using a simple application system to determine their fit and assign a role.

Even on small teams, this sync helps prevent bottlenecks and miscommunication down the line. It also builds momentum, once people know where to go and what to do, they're more likely to dive in confidently.

7. PROTOTYPING

Building the foundation.

This is where things start coming to life. Whether or not you ran early experiments in pre-production, prototyping is where we test the actual features the project is built on. These aren't just ideas, they're the core mechanics, systems, or sequences that the rest of production will rely on.

It's not about figuring out what might work, it's about proving that what we've planned does work, or refining it until it does. That means building early versions of key features and making sure everything plays well together before we move into full-scale content creation.

CORE FEATURE PROTOTYPING

Testing what matters most.

At this point, we're not experimenting, we're building out the essential parts of the project that we already know will be included. This step is about prototyping the core elements that define how the project works, feels, or flows.

These early versions help us confirm that our ideas hold up once they start taking real shape. They aren't fully polished, but they're functional enough to prove that the direction works in practice.

Whether you're working on a story-heavy film, a systems-driven game, or a stylized animated piece, this is where you make sure the core experience actually translates from idea to execution.

What this usually includes:

- **TESTING CORE MECHANICS OR SYSTEMS** - For interactive work, this means early builds of gameplay loops, navigation, dialogue systems, or other essential functionality.
- **BUILDING KEY MOMENTS** - Could be a playable scene, an animated sequence, or a filmed segment that captures tone, pacing, or audience experience.
- **APPLYING VISUAL DIRECTION** - Taking your chosen style and applying it to a few pages, shots, screens, or scenes to see how it plays out in context.
- **EXPLORING STRUCTURE & FLOW** - Mapping or mocking up how the audience moves through the piece: scene order, level pacing, transitions, or layout.
- **ALIGNING CREATIVE ELEMENTS** - Making sure visuals, writing, sound, and interactivity (if present) are working in harmony, not clashing.
- **VALIDATING PRODUCTION SETUP** - Checking that tools, workflows, export formats, or pipelines are actually working and ready for scaling.

NOTE FOR INTERACTIVE OR GAME PROJECTS

We recommend prototyping mechanics and visuals separately at first. That way, you can focus purely on functionality without getting distracted by the presentation. A prototype doesn't need to look good, it just needs to work. Use simple placeholders (like colored blocks or basic shapes) to test gameplay, logic, or interaction before bringing in visual polish. And always make sure to save a backup of the mechanics prototype for future reference before implementing the visuals.

These prototypes should be created using the actual tools or software you plan to use moving forward. The goal is to build something real, not throwaway. You're laying the groundwork for what becomes the final version.

The goal: Start building with confidence. If something isn't working, this is the best moment to fix it, not halfway through development.

Common roles involved:

- **GAME DESIGNERS or SYSTEMS DESIGNERS** - Lead interactive and mechanic prototyping
- **VISUAL ARTISTS, UI DESIGNERS, or STORYBOARDERS** - Build early visuals, layout tests, or scene mockups
- **SOUND DESIGNERS & COMPOSERS** - Lay groundwork for audio direction and tone
- **VIDEO EDITORS** - Prototype cuts, motion sequences, or pacing
- **PROJECT LEADS** - Oversee the process and ensure each prototype aligns with the overall plan

Use our planner to figure out what to build and how to go about it:

[CORE FEATURE PROTOTYPING PLANNER →](#)

SYSTEMS SYNCING & INTERGRATION CHECKS

Making sure everything connects.

Once you've got your core features in place, it's time to see if they actually work together. This step is about syncing the individual parts, gameplay, visuals, narrative, audio, logic, layout, and checking that they behave as expected when connected.

That doesn't mean everything needs to be polished or fully functional yet. But it does mean you should be able to run through the core loop or flow of your project without things falling apart. The goal here is to surface integration issues early, before they become blockers later.

What this usually includes:

- Connecting different systems or content streams (e.g. dialogue triggering events, UI interacting with gameplay)
- Syncing pipelines between departments (e.g. how visual assets get implemented in-engine)
- Testing narrative flow alongside mechanics or level pacing
- Checking if your tools and file structures support the current direction

This step is especially useful for PROGRAMMERS, TECH ARTISTS, and GAME DESIGNERS in interactive projects, but also for VISUAL ARTISTS, UI DESIGNERS, and NARRATIVE ROLES to ensure their parts align and slot into the broader system.

Even in non-interactive projects, like film or comics, this step still matters. It might be syncing shot sequences with pacing, visual transitions with story beats, or layout with dialogue flow. Whatever the medium, make sure the moving parts are actually moving together.

FEEDBACK LOOP TESTING

Running through the rough build.

Once your features are built and systems are synced, it's time to step back and test how the whole thing comes together. Not just piece by piece, but as one connected flow.

Feedback Loop Testing is about trying out the core experience in its current, unfinished form. Scenes might be rough. Timing might be off. But the structure is there — enough to walk through and feel how it plays out.

Why this matters:

If the experience holds up now, without polish, that's a good sign it's going to land when it's finished.

- Addictive gameplay still feels addictive, even in a greybox.
- Emotional scenes still hit, even within a rough storyboard.

If it works in this state, it'll work even better later.

What you're testing:

This isn't about polish. You're testing whether the structure holds up, even in its rough, unfinished form.

Watch for:

- Moments where someone gets stuck or doesn't know what to do next
- Parts that feel confusing, unmotivated, or directionless
- Emotional or narrative beats that fall flat or don't land
- Sections that feel too long, too short, or poorly timed
- Features or interactions that are clunky or distracting

You're looking for friction, anything that breaks the experience or pulls someone out of it. If you catch those things now, you'll avoid bigger problems later.

How we run it:

We bring in the full team, not just the ones who built it. A mix of fresh eyes and familiar folks helps discover what's working and what's not.

Feedback is gathered using GOOGLE FORMS, which keeps everything simple, fast, and organized.

For those running the test round:

Check the guide below for best practices, detailed steps, and examples on how to collect strong feedback, including how to define goals, pick the right method, ask the right questions, and iterate on the results.

[USER EXPERIENCE FEEDBACK GUIDE →](#)

What happens next:

Look for patterns in the feedback. If something isn't landing, adjust. If it's already working, even in this rough state, you're ready to move into full development.

8. DEVELOPMENT

The core of it all.

This is where we start building the actual thing. Based on everything established so far, the plan, the prototypes, the feedback, we begin producing real content at scale. Scenes, systems, assets, dialogue, animations, layouts, sound, UI, whatever the format calls for.

Unlike prototyping, which is about testing if the foundation works, development is about building it out fully. We're no longer just checking if something makes sense, we're making it.

It doesn't have to be final yet. But it does have to be functional, connected, and consistent with the direction we locked earlier. The focus now is momentum: steadily producing, assembling, and adjusting as we go.

CONTENT PRODUCTION

Building the actual content.

This is where teams start creating the real stuff: the writing, visuals, audio, systems, animation, interaction, whatever the project needs. The focus here is on producing the final (or near-final) assets, not placeholders or mockups.

Each department works from the plan and prototypes. You're not starting from scratch, you're scaling up what was already tested. Scripts get written. UI is designed. Animations are blocked out. Sound effects are recorded. Systems are coded with actual logic and structure.

The pacing will vary. Some teams work ahead, others catch up later. The important thing is that everyone stays aligned, communicates often, and shares progress so we can keep the momentum moving.

PROJECT LEADS should help remove blockers and make sure departments are building toward the same version of the project, not separate ideas of what it should be.

SYSTEM IMPLEMENTATION

Making it all actually work.

This is where features, mechanics, and interactions from prototyping get implemented for real. Placeholder content gets replaced with actual assets. Systems are built out using final logic and connected to the rest of the project.

The goal here is consistency and functionality. Everything needs to work as intended, across tools, formats, and departments. That includes UI behavior, audio triggers, camera systems, dialogue integration, and any other structural pieces the experience depends on.

For games, this might mean gameplay systems coming together with UI and player interaction. For film or animation, it could involve assembling cuts, syncing audio, or finalizing scene transitions. For mixed media, it often means stitching different types of content into one working flow.

This step usually involves close coordination between **TECHNICAL ROLES**, **DESIGNERS**, and **ARTISTS** to make sure everything connects smoothly, no missing logic, mismatched content, or implementation errors.

If something that worked in prototype form isn't holding up now, flag it early so it doesn't carry over into refinement.

ASSEMBLY

Putting it all together.

Now that content is being produced and systems are getting implemented, it's time to assemble the actual structure of the project. This is where everything starts to take shape as a full, connected experience.

Depending on the format, this could mean:

- Laying out levels, chapters, or scenes in order
- Building full gameplay sequences
- Editing together animated shots or video segments
- Structuring timelines, transitions, or progression flows

You're no longer working in isolation. Visuals, writing, sound, logic, it all needs to play well together. This step helps reveal how each piece supports (or disrupts) the bigger picture.

The point isn't to perfect it. The point is to see it functioning as a whole, so we can adjust what doesn't fit before refining begins.

9. REFINEMENT

Coming Soon ...

10. DOCUMENTATION

TRY FINISHING THIS.

POST-PRODUCTION

Coming Soon ...

11. TESTING

Coming Soon ...

12. POLISHING

Coming Soon ...

13. ARCHIVING

Coming Soon ...

14. PRESENTATION

Coming Soon ...

DOCUMENTATION & REFLECTION

TRY FINISHING THIS.

TRAINING

TRAINING

Coming soon ...

Ye, kind of lame, I know, but I promise to expand upon this when the time is right and we have the right resources. I don't want to rush this chapter because I feel like it's more important to get it right.

I am still doing research and noting any useful creators or resources for the future, so once again, I promise you I am working on it.

DOCUMENTATION

DOCUMENTATION

This chapter explains how we keep track of what's been done, what decisions were made, and where things stand.

You're not expected to write a lot. Just enough so your part of the work is visible and the rest of the team can stay aligned. We also appreciate a short explanation behind decisions when possible. It helps us know we're making informed choices, not just reacting.

Documentation also gives you a way to look back, reflect on what's changed, and understand how your work connects to the bigger picture. We'll cover what kinds of things we document, how often, and where to find the templates that make it simple.

WHY DOCUMENT?

Here are a couple of reasons why you should care about documenting your progress:

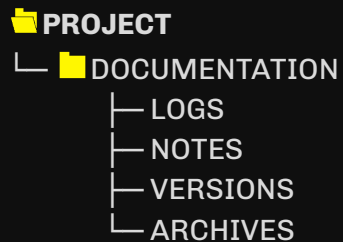
- **You get to reflect on your own decisions.** Writing things down helps you notice patterns, rethink choices, and become more intentional over time.
- **You can see your growth.** Logging your work creates a trail of progress, what you've tried, changed, and figured out along the way.
- **It makes teamwork smoother.** When everyone logs what they're doing, no one's left guessing. We waste less time repeating ourselves or untangling confusion.
- **It keeps projects clear and trackable.** Documentation gives structure to the chaos. It helps us understand what's been done and what still needs attention.
- **It provides context when you return later.** Whether it's been a week or a few months, good notes let you pick up where you left off without having to re-learn everything.

WHERE WE DOCUMENT

We don't keep all our documentation in one giant folder. Each project or related folder has its own DOCUMENTATION subfolder in Google Drive. That's where everything goes, logs, notes, revisions, feedback, etc.

This keeps things close to the work it's about. You don't have to dig through a central archive to find what you need.

Here is a simple example of how a documentation folder could be set up:



We also have a shared folder with templates, guides, and documentation tools. You'll be linked to it when needed, but you can always find it here:

[DOCUMENTATION RESOURCES →](#)

If you're ever unsure where to put something, ask, but most of the time, if it lives in the same folder as the work, you're doing it right.

TYPES OF DOCUMENTATION

We organize documentation into four main categories. Each one covers a different kind of work, but they all follow the same structure: clear, consistent, and useful.

1. PROJECT DOCUMENTATION

This covers anything related to the development, planning, or execution of a project. It helps track progress, organize ideas, and record important decisions as the project evolves.

EXAMPLES

- Project logs and timeline updates
- Brainstorms and moodboards
- Creative development (story drafts, design ideas, level planning, etc.)
- Feedback logs from internal reviews or testing
- Bug and issue tracking
- Task tracking and assignments

Head to our resources for PROJECT DOCUMENTATION for guides and templates:

[PROJECT DOCUMENTATION RESOURCES →](#)

2. TEAM DOCUMENTATION

This focuses on the people, what each person is doing and how the Team as a whole is moving forward.

There are two categories:

- **INDIVIDUAL LOGS** - Every member writes a short log of what they've contributed. These are typically written on a weekly, biweekly, or monthly basis, depending on the work pace or project.
- **SHARED TEAM LOGS** - All individual entries are collected into a single document by the end of the logging period and summarized.

Team documentation helps with internal awareness, reflection, and cross-role visibility.

Head to our resources for TEAM DOCUMENTATION for guides and templates:

[TEAM DOCUMENTATION RESOURCES →](#)

3. STUDIO & WORKFLOW DOCUMENTATION

This keeps track of how we work, the systems, tools, and rules we follow as a Studio.

EXAMPLES

- Internal workflows and method breakdowns
- Setup checklists for new projects or roles
- Tool usage guides
- Folder structure references
- How we name, version, and organize our files

Head to our resources for STUDIO & WORKFLOW DOCUMENTATION for guides and templates:

[STUDIO & WORKFLOW DOCUMENTATION RESOURCES →](#)

4. COMMUNITY & EXTERNAL DOCUMENTATION

This covers anything that involves people or platforms outside of the internal Team, from public feedback to community events and outreach.

EXAMPLES

- Community suggestion logs and surveys
- Event planning documents and post-event summaries
- Promotion and marketing tracking
- Notes from outside collaborations or partners

[COMMUNITY & EXTERNAL DOCUMENTATION RESOURCES →](#)

EXPECTATIONS & RESPONSIBILITY

Everyone contributes to documentation. If you've been involved in something, a project, a meeting, a task, anything, there should be some kind of record of what you did.

There's a saying: "If it isn't written down, it didn't happen." While we don't take it as gospel, we do believe there's truth to it.

We're not looking for detailed breakdowns, just a simple overview of what's been done, any key decisions, and anything worth remembering later. If you have additional details, you can add them at the end of your log in a "RESOURCES" or "EXTRAS" section.

How often you document depends on the work:

- **ONE-DAY TASKS** - Document on the same day
- **ONGOING TASKS** - Log updates weekly or biweekly
- **SLOW OR LONG-TERM TASKS** - Monthly updates are fine
- **MANAGERS & LEAD** - Help track team-wide progress and compile shared logs

If you're unsure where or how to log something, ask. It's always better to clarify than leave it undocumented. The goal is to keep things clear, for yourself and the people you're working with.

HOW TO DOCUMENT

We provide templates for everything we can. They're here to guide you, not confine you. Each template provides a clear starting point, so you're not left staring at a blank document. They're designed to maintain consistency, readability, and ease of use across the Studio.

You'll find them in:

- The [DOCUMENTATION RESOURCES](#) from earlier
- Inside most project folders under /DOCUMENTATION/TEMPLATES
- Eventually (still a W.I.P) in resources #library on Discord

Most are Google Docs, you copy to edit. Some include examples, brief guides or notes to help you complete them.

If the template doesn't fit what you're doing, that's fine. As long as your log includes the key info and follows a similar structure, it works.

Every log should include:

- **DATE** - When the work was done or the log was written
- **WHO** - Your name and role
- **WHAT** - What you did or what changed
- **WHY** - Context or reasoning behind the work
- **NEXT STEPS (IF RELEVANT)** - What's happening next or what still needs doing
- **EXTRAS OR RESOURCES (IF RELEVANT)**- Any files, links, visuals, or longer notes that don't fit into the main log

Keep it clear and readable. That's what matters most.

VERSIONING & FILE NAMING

We use a simple system to keep track of changes, whether you're working on a doc, game build, design file, or anything else.

VERSIONING FORMAT

We version our files so it's easy to track what's changed, how significant the change was, and which version is the most current.

This keeps everyone on the same page, especially when projects are moving quickly or multiple people are working on them simultaneously.

We use three numbers:

1.0.0 → Major.Minor.Patch

Each number reflects the scale of the update:

- **1st digit** = Complete version or full overhaul
- **2nd digit** = Significant update or added content
- **3rd digit** = Small fix, patch, or adjustment

Whenever the first or second digit changes, the numbers after it reset to zero.

EXAMPLE

- 1.1.1 → 1.2.0
- 1.1.1 → 2.0.0

This system works across everything:

Documents, websites, game builds, scripts, audio files, and any other content that evolves over time.

We also keep a Version Control Log to track big updates.

It includes:

- Version number
- File name
- Short, specific description of what changed

NAMING FILES

Consistent file names make it easier to find, share, and organize files. We keep names short, clear, and searchable, because no one wants to dig through a mess of random titles. Depending on the project, we typically agree on how to handle this early on with a shared document for guidance.

We keep file names:

- **SIMPLE** - Short, clear words
- **CONSISTENT** - Same format across folders
- **SEARCHABLE** - Easy to find later

We use prefixes to show what the file is for:

- **HS-** = General HELLINGTON STUDIO files
- **PROJECT ABBREVIATION (PA-)** = files tied to a specific project
(PA = Short for “Project Abbreviation” and is a placeholder for the abbreviation of your actual project name)

EXAMPLES

- HS-TeamLog-V1.0.0
- PA-Storyboard-V1.2.1

FINAL NOTES

Documentation doesn't need to be perfect; it just needs to be useful.

A brief log can go a long way in helping you track your own progress and keeping others informed. It saves time, prevents confusion, and makes it easier to keep things moving.

Use the templates. Follow the structure. Ask if something's unclear. If you've worked on something, make sure there's a record of it. That's all we need.

For a comprehensive guide on how to properly handle logging progress in-depth, version control, file naming and documentation in general, refer to our guidelines:

[DOCUMENTATION GUIDELINES →](#)

And for any resources like templates or related files on documentation, head to our resource library:

[DOCUMENTATION RESOURCES →](#)

GUIDELINES

GUIDELINES

Coming soon ...

NOTE: Some guidelines will be determined during and after our first project, while others are planned and will be added here once major sections of the TEAM GUIDE are completed.

TABLE OF CONTENTS

As a shortcut, you can click on the chapter you wish to visit below.

COMMUNICATION GUIDELINES

CONDUCT GUIDELINES

WORKFLOW GUIDELINES

DOCUMENTATION GUIDELINES

WORKFLOW GUIDELINES

TOOLS GUIDELINES

MEETING GUIDELINES

CREATIVE GUIDELINES

CONTENT GUIDELINES

RESEARCH GUIDELINES

SUBMISSION GUIDELINES

AVAILABILITY GUIDELINES

LEARNING GUIDELINES

CONFIDENTIALITY GUIDELINES

HSE GUIDELINES

COMMUNICATION GUIDELINES

Coming soon ...

CONDUCT GUIDELINES

Coming soon ...

WORKFLOW GUIDELINES

Coming soon ...

DOCUMENTATION GUIDELINES

Coming soon ...

TOOLS GUIDELINES

Coming soon ...

MEETING GUIDELINES

Coming soon ...

CREATIVE GUIDELINES

Coming soon ...

CONTENT GUIDELINES

Coming soon ...

GENERATIVE AI

TRY TO FINISH.

RESEARCH GUIDELINES

Coming soon ...

SUBMISSION GUIDELINES

Coming soon ...

AVAILABILITY GUIDELINES

Coming soon ...

LEARNING GUIDELINES

Coming soon ...

CONFIDENTIALITY GUIDELINES

Coming soon ...

HSE GUIDELINES

Coming soon ...

SUPPORT

SUPPORT

You're not expected to figure everything out on your own.

Whether you're stuck on something, unsure where to find a resource, or just need clarity, support exists for a reason. This chapter shows you where to turn when you need help, and how we make sure no one's left in the dark.

From asking a staff member to browsing guides in the library, you've got options. And if you're feeling overwhelmed, we've got space for that too.

WHO TO ASK

If something's unclear or you need support, reach out. You're not expected to navigate everything alone.

You can contact:

- **LEAD** - for big-picture guidance, project direction, or general support
- **MANAGER** - for help with scheduling, task tracking, or project flow
- **ADVISOR** - for feedback, outside perspective, or second opinions

DMs are always open, or you can bring it up in the relevant Team or project channel. If it's about something specific or sensitive, try to check if there's a *LEAD* or *MANAGER* connected to that project first.

RESOURCES

Not every question requires a person; many answers already exist within our internal resources.

Most of our resources live right here in the Team Guide in the [RESOURCE LIBRARY](#) chapter. This includes templates, tools, guides, documentation, and walkthroughs, everything you need to stay on track.

If you need quick access or just want to grab something fast, check the #library channel on our Discord Team Server. It's organized into threads for easy browsing and links back to the original resources in the guide or Drive.

Use the guide for structure. Use the Discord channel for quick access.

HEALTH & WELLBEING

Creative work takes energy, and sometimes, it takes more than you've got. Burnout, exhaustion, and mental blocks happen. You're not expected to push through in silence.

If you're struggling, let someone know. Talk to your *LEAD, MANAGER*, or someone you trust on the Team. You don't need to share details, just flag that you need space, support, or time.

We care more about your well-being than your output. What matters most is communication. If we know what's going on, we can adjust. You're allowed to step back. You're allowed to take breaks.

This is a Team, and support includes making sure people aren't running on empty. For more on how we approach mental health and sustainable work, please refer to our [HSE GUIDELINES](#).

FINAL NOTE

Support isn't a last resort; it's part of how we work. You're not expected to have all the answers or carry everything on your own.

If you're stuck, confused, or just need a second set of eyes, speak up. We're here to help each other. That's the point of working in a Team.

(illustration here)

DICTIONARY OF TERMS

DICTIONARY OF TERMS

Coming soon ...

I am still noting down terms I will expand upon later, once I have a comprehensive list, I will get it done.

Notes for this section:

- Words commonly used within the Studio and their precise meanings in the context of our usage.

List of words to add:

- Skillset - Refers to the set of skills one possesses.
- Mission - Refers to tasks usually leading up to a project or what our short-term goal is for a task or project.
- Vision - Refers to the long-term goal we're trying to reach, for instance, for a project.
- Recruit - Refers to the process of recruiting new members.
- DAW - Digital Audio Workstation.
- DD - Design Document.
- Logging - Documenting one's progress.
- QA (Quality Assurance) - Quality testing.
- Sprint Board - A way of organizing/managing a timeline of tasks within a project.
- MVP (Minimum Viable Product) - The simplest version of a new product that includes just enough essential features to be usable.
- Campaign Deck - A way of pitching a marketing campaign.
- Design Document - A document that details and breaks down everything about a project.
- Road Map - A live to-do list of what's planned, ever changing based on priorities.
- Feedback Cycles - Giving and Receiving Feedback in Structured Rounds.
- Project Brief - Describes the most important details about a project.
- Double Diamond - A method of thinking in terms of open and closed scope, depending on the phase of a project you're in.
- Computational Thinking - Mindset useful especially for programming.
- Iterative Process - Not working linearly, instead revising and repeating steps as needed.
- Waterfall Model - A linear approach, particularly useful in software development.
- Workflow - The tasks/activities leading up to achieving a set goal.
- Handoffs - Transferring work from one person to another during the production process. For example: 2D artist finish character design, 3D artist models the character.

- Dependencies - Usually, we refer to tasks that are dependent on others completing it first.
- Treatment - In Video Production, it's a detailed document outlining the concept, storyline, and stylistic approach for a video project
- Pipeline - A series of connected stages through which something (data, materials, or a process) moves sequentially from one point to another
- Frontend - The user-facing part of a website or application, the part that users directly interact with
- Backend - The server-side of an application, encompassing the code, databases, and servers that handle user requests and manage data

RESOURCE LIBRARY

RESOURCE LIBRARY

RESOURCE LIBRARY

Coming soon ...

SHOULD START ADDING RESOURCES AND ORGANIZING IT BETTER LATER.