#019 Level Design - Process - Pre-production

> How to plan a game level with mechanics, experience goals, and pillars
> From *The Level Design Book*.

- ▼ *#019 Level Design - Process - Pre-production*
  - *What is pre-production?*
  - *Project planning*
  - ▼ *Mechanics*
    - *What if you haven't finalized mechanics yet?*
    - *Build test maps*
  - ▼ *Experience Goals*
    - ▼ *Advice for setting goals*
      - *BE SPECIFIC.*
      - *THINK FROM THE PLAYER'S PERSPECTIVE.*
      - *DEFINE MULTIPLE GOALS.*
    - *Pillars*
  - *Example: Psychonauts 2*
  - *Example: Dishonored 2, "Clockwork Mansion"*
  - *To review...*

What is pre-production?

Pre-production is the early phase of a project where you generate ideas and try to figure out what the project is about. It's about answering big design questions:

- What are you making?
  - What will the player be able to do? (Mechanics, Experience Goals, Pacing)
  - Which parts are most important? (Pillars)
- How are you going to make this?
  - How did other people solve similar problems? (Research)
  - How much time do you need? Is this feasible? (Scope)

Your answers to these questions will change throughout the project, and that's OK.

Planning is useful but it's not magic; you won't completely understand what you're making until you start making it.

Project planning

"Planning a level" can take many forms. Some people build small prototypes and test levels before committing to a concept, or you can write out design goals and constraints.

Minimal checklist

If you can answer these very basic questions, that's enough to start a quick solo jam project:

- ☐ **Elevator pitch.** In 1-3 sentences, describe the project concept and what makes it interesting.
- ☐ **Tools.** What engine and tools will you use? Will they work for this project? What questions do you have about functionality?
- ☐ **Scope (basic).** How long will it be? How many levels? Are these big or small levels? What does a "big level" mean? These are basic question about scope, the size and workload for a project.

Recommended checklist

But really, you should be able to answer these other basic questions about the design:

- ☐ **Pillars.** What are the 2-3 most important themes / features of this project?
- ☐ **Experience goals.** How should the player react throughout each level? What type of emotion or behavior?
- ☐ **Mechanics.** What are the frequent activities / interactions / skills the player will use?
- ☐ **Asset list (basic).** What kinds of models, textures, and sounds will you need for each level? Have you acquired these assets yet, or is someone going to make them?

Full checklist

If you're 3+ people or if it's a 3+ months project, then more detailed planning will help you a lot:

- ☐ **Pacing.** What happens in each level? Why? How do the levels fit together?
- ☐ **Research.** What are the inspirations / reference points for these levels? Does the group have a shared understanding and interpretation? How is this project in conversation with the larger culture and industry?
- ☐ **Worldbuilding.** What is the larger world / fictional universe of these levels? Who built these places? What logic / mood guided the construction and use?
- ☐ **Scope (detailed).** Every week, what will you work on, and when? What features / levels are "nice to haves" that can be cut from final release? Planning scope can get so complicated that

commercial studios hire one or more producers -- people whose main job is to oversee scope.

The core of preproduction planning: **write down the main ideas and then talk about it.**

Mechanics

A game mechanic is any activity, system, or tactic repeated in your game, basically any action a player routinely performs or uses. **Core mechanics** are the most basic frequent and fundamental actions / systems of a game, while less frequent activities can be called secondary mechanics.

Do not design for the genre, do not sleepwalk into familiar patterns and conventions. Instead, design for the actual game that exists. Get a feel for the game. If you're modding a pre-existing game, play that game for at least a couple hours, including other community modders' work. Watch YouTube footage of gameplay, and pay attention to how other players combine various mechanics and game systems.

What if you haven't finalized mechanics yet?

It's hard to build levels without knowing the mechanics. Both mechanics and levels depend on each other, one cannot exist without the other. You can't evaluate a mechanic without testing in a level, but you also can't quite evaluate a level without finalized mechanics.

If you change your mechanics, then you'll end up with a lot of wasted work and obsolete levels that relied on discarded mechanics. But if you don't build enough test levels to measure the mechanics' potential, then you might end up with a dud mechanic that isn't interesting enough beyond the first level.

Throughout your game's development, you'll likely have to throwaway or redo your levels. You might even have to remake a level multiple times! It's OK. Do not think of this as wasted time or work. Instead, think of each discarded version as valuable research. This is why we call it game development -- development is a gradual process that takes time. It will take time to figure out what makes for a good level and a good game.

Build test maps

As part of the planning process, game designers often build simple "playground" blockout-style test maps. These barebones level are intended for internal developer use only, and usually do not ship in the final public release.

For our purposes, these maps basically have zero level design, they are just big boxy courtyards filled with random objects and NPCs.

These simple playground-style debug levels are vital for testing game mechanics, physics, and general feel for the game. This is how you prototype a game.

Experience Goals

An **experience goal** is some kind of idea, feeling, or activity, that you want the player to understand or undergo while playing. To conceptualize goals, try starting with the phrase, "In this level, the player should [learn/feel/do]..."

- In this level, the player should...
  ... learn how to use the double jump ability. (Experience as tutorial)
- In this level, the player should...
  ... feel vulnerable, then reach the safehouse and feel relief. (Experience as emotion)
- In this level, the player should...
  ... dodge deadly traps and unlock a shortcut at the end. (Experience as activity)
- In this level, the player should...
  ... feel like they're escaping a medieval prison. (Experience as fantasy)

Advice for setting goals

BE SPECIFIC.

Experience goals don't have to be complicated or profound, but a clear and specific experience goal helps you decide what to build and what is unnecessary.

For example, "the player should have fun", is a vague goal that can't drive design decisions. What type of fun? A light and casual type of fun, a deep engaging type of fun?

Or if the goal is to "make the player feel fear," then what type of fear? There's body horror, existential horror, fear of failure, fear of rejection, mid-life crisis, etc.

THINK FROM THE PLAYER'S PERSPECTIVE.

Avoid overly abstract experience goals, and try to think of it from the player's perspective. Ideally, how would a player describe their feelings or understandings while playing the level?

For example, "the player should enjoy non-linearity" focuses on an abstract structural aspect that most would not grasp nor even consider. It is not a useful experience goal. What do you actually mean by

enjoyment or non-linearity?

Imagine interviewing the player after a playtest. What do you hope they say about your level? You would feel insulted if they said, "wow, that was really... um... non-linear..."

DEFINE MULTIPLE GOALS.

Levels rarely have just one experience goal. Even something straightforward like a tutorial level benefits from additional experience goals about setting up some storytelling, fantasy, or mood. It might be less important than teaching the player how to play the game, but it still helps to guide design decisions.

For example, "the player should fight five enemies at once" is a combat-oriented mechanical goal for a level. But so much is still undefined here. Is five supposed to feel easy or difficult? Where is this level set, will we need a bigger room to support this higher enemy count?

Let's add additional experience design goals: "the player should feel overwhelmed but then discover a powerup" and "the player should feel like they're defending their campsite from zombies" ... we now bring in themes of resourcefulness, defense, survival, and home. The concept of a campsite suggests a medium-sized outdoor area, and now we can research more specific types of campsites (military? hiker? winter?) to make our plan more specific.

Pillars

The most important experience goals across your entire project are **pillars** -- the most vital ideas that structurally support and justify your entire design. Design pillars help teams maintain a cohesive vision for the project, and resist "feature creep" work tasks that don't actually support your main goals.

Pillars are a design planning tool to help you conceptualize short-term experience goals for each level or area. Each of your level's smaller experience goals and small gameplay beats should contribute to your pillars somehow, and that culminates in the grander arc of your entire game's experience.

Pillars help us define what is important about our projects, and thus make more consistent and coherent decisions during development.

Example: Psychonauts 2

Psychonauts 2 is a 3D platformer about exploring peoples' brains. In November 2016, Double Fine Studios made a pre-production with a vertical slice style "art test" prototype -- 5 years before the game's release in August 2021.

As depicted in their official dev documentary video series "PsychOdyssey" episode 8, the pre-production team focused on building out a core movement set, art style, and level design in Unreal Engine 4. While much of the core platforming mechanics carried over from the original Psychonauts 1, the team wanted to experiment with new ways the player's various psychic powers could interact with the platforming gameplay -- resulting in a telekinesis-with-movable-tightrope prototype mechanic that never appeared in the final game.

By the end of the dev cycle, there's still a lot of uncertainty and unanswered questions. Will this game be good? No one knows. The journey is just beginning.

Example: Dishonored 2, "Clockwork Mansion"

Dishonored is a first person stealth action RPG series about exploring complex places. In Dishonored 2, there is a "Clockwork Mansion" level where the layout dynamically changes.

Arkane Studios designer Dana Nightingale built a proof of concept pre-production prototype: (*Twitter thread*)

> "I created this proof of concept early in 2013, long before the map was greenlit to be included in the game, basically to say "yes this could be amazing" and "yes I should work on it." It did both, but it would still be over a year before the map was officially OK'd.
>
> But sure, I could make a bunch of blocks animate however I wanted. What is possible using real level geometry? That's why I made this prototype. A bit less wild? Yes. But still clearly do-able.
>
> That prototype introduced the idea of rooms that were moved around like cargo containers and slotted into place, making the map's layout totally dynamic, but without transforming rooms. I am glad we didn't go in that direction.
>
> [...]
>
> After nearly a year of working on other prototypes it was time to revisit the Clockwork Mansion and finally get it validated. I made a new version with fewer twisty rooms, and more emphasis on the "behind the scenes" areas..."
>
> -- *Dana Nightingale (@DanaENight), 14 October 2021*

To give you a sense of time scales here: this level spent 7 months in pre-production, then entered final layout / blockout about 18 months later.

To review...

- Mechanics are the repeated systems and interactions of a game.
  - Core mechanics are the most frequent, while secondary mechanics add occasional variety and finesse for more experienced players.
- Experience goals are the player-facing design goals for your levels
  - Pillars are the main design goals for your entire game. What do you want the player to learn, feel, or understand?
- How much planning do you need? It depends.
  - Small casual projects will be OK with small (or nonexistent) plans, big long-term projects may require a huge internal wiki.
  - **Pre-production planning helps developers coordinate and trust each other.**
  - If you're prototyping new mechanics or game design elements, then **blockout a test map** with a big playground-style space.
- Plans help you make things, but **plans always change.** Even after preproduction, while working in the middle of the project, you will almost always adjust your plans and improvise a redesign. That is OK.

> ***write by Hellinus - 2023.9.14***