

Sane, not Safe

Making a High Quality Game in a Low Quantity Time

Be Sane, not Safe

This is a collection of tips, complaints, good ideas, and booboos that I've seen in my time involved with this class.

The short version:

You should be sane; you don't have to be safe.

Cite Your Sources!

Brian Upton, 30 Things I Hate About Your Game Pitch

<https://youtu.be/4LTtr45y7P0>

Butterscotch Shenanigans, Shenanijam HOW-TO

<http://bit.ly/howtogamejam>

Did You Read This?

<http://robert.zubek.net/docs/games-studio-2018/lecture-notes/2-player-experience.pdf>

If you didn't, don't worry (I won't tell Rob). You should read it again; it holds a lot of useful information to help guide you.

1. Be Free (at First)

How to come up with a whole lot of ideas in like twenty minutes:

1. Everyone split up and come up with as many ideas as possible in ten minutes. Write *everything* down; self-censoring comes later.
2. Join up and read everyone's lists. Write ideas on a whiteboard, grouping by "similarity."
3. Spend five more minutes coming up with ideas. Write them all down.
4. Everyone pick your favorite few and discuss those until you have a winner.

You can be realistic later (in like two slides).

For now, just come up with what sounds like fun.

2. Do Your Homework

Questions to think about:

- Has someone made this game before?
- How would you describe your game?
 - can you do it without mentioning another game?
- What are your genre's gameplay conventions?
- What are your genre's UI/UX conventions?

It's fine to push (or even ignore) boundaries, but you need to know where the boundaries are.

3. Know What Experience You're Delivering

- Who's going to play your game?
- Where are they going to play it?
- What sort of experience do you want them to have?
 - light-hearted, spooky, existentially dreadful...?
 - how long will it be?
- What are you asking them to do?
 - physically, mentally, morally...





High Total Time



Few Mechanics

Your Shit???

Many Mechanics

Something
Weird...?

Low Total Time

4. Plan It All Out

Make a fairly detailed schedule. You don't have to follow it, but it will force you to *think about everything*, which is more the point, anyway.

- What (exactly) are you going to do?
 - specific, quantified descriptions like “add 2 new types of melee weapon” are better than “more weapons”
- When are you going to do it?
 - date (and maybe time, if your schedule works like that)
- Who's going to do it?
 - there are three of you - who's best suited to take care of it?
- How long will it take?
 - descriptions like “3 hours” are fine (but be realistic!)

5. Think About (Design) Everything

A *bad* design decision implies that you have some learning to do. That's fine.

A *negligent* design decision implies that you were lazy. That's not.

Or, in other terms:

- It's fine to look like Guy Fieri on purpose
- It's not fine to look that way by accident



6. Focus on the Key (Unique) Parts of Your Game

Two things:

1. How important something seems to a player is a function of how much design time appears to have been put into it.
 - a. Or: a more complicated feature seems more important.
 - b. Or: how much *you* focus on it is how much *they* focus on it.
2. You have limited time, don't spend it on parts of your game that don't matter.

Or, in other terms:

1. Don't make me invest time in something secondary.
2. Don't invest your time in something secondary.

6b. And Do It Now

How early you work on something should be proportional to how paramount it is to your game.

*Don't spend the first weeks getting idle animations to work.
Show me your goddamn game.*

7. Don't Sweat the Small Stuff

On the other hand, little things add up in the end. Spend the last 20% of your time polishing. Some things you can do:

- Start menu. You can make one in five minutes.
- Standalone executable. It's nice not to have to look at the Unity UI...
- The ability to restart. Don't make this complicated, just restart the scene.
- Non-textual UI elements (meters, gauges, indicators, etc.). Don't make your players look at "Health: 100."
- A "custom" font, or UI icons. Sources at the end.
- Controller input (if relevant).
- Sounds effects and background music. Surprisingly simple.

8. Be Consistent

Consistency of aesthetic contributes a lot of quality to your game.

This harkens back to my complaint about “Health: 100,” but it also means that you should attempt to combine assets that look like they belong.

Or, in other terms:

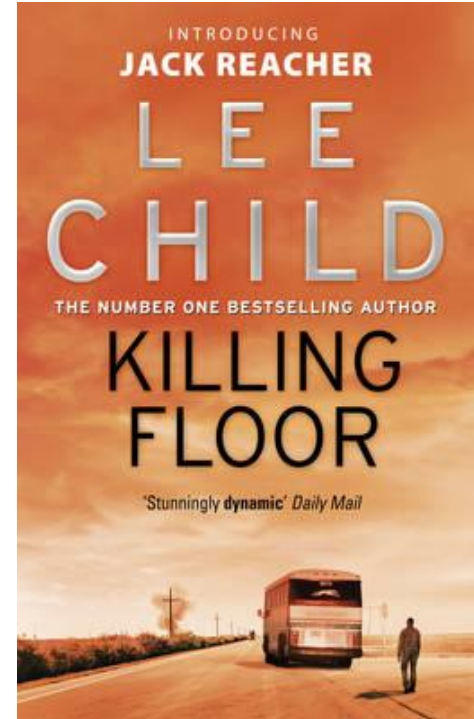
Don't mix a “Space” font with cutesy animals (unless your game is about kittens in space)

9. No Plan Survives First Contact with the Enemy

Nothing that you implement is going to go over the way that you expect.

This isn't a good or a bad thing - people (including you) are just really bad at predicting how other people are going to behave.

The secret to success (no really) is to put everything in front of someone as soon as possible.



10. Shoot for Roughly a Half an Hour a Day

Think of it like practicing the piano or going to the gym:

7 Days/Week x 0.5 Hours/Day =

3.5 Hours/Week

This is **NOT** the same thing as doing one long practice/workout each week.



Remember

Get your core loop up and running *as soon as possible*. People can't comment on what isn't there.

Consistent, incremental change is crucial. You can't cram, sorry.

Don't polish too early, but make sure to polish!

Playtesting time (and players) is limited: take notes and pay attention to comments.

You have a lot of creative freedom, so:

Make something exciting!

Places to Get Assets

- Unity Store (duh) - lots of paid and free assets, scripts, etc.
- Kenney - a substantial quantity of free and paid assets, all high-quality.
<http://kenney.nl/>
- Opengameart - lots of free, open-source art of... variable quality. Also some SFX. <https://opengameart.org/>
- Deviantart - a little difficult to find things on here, but you'll strike gold now and then. <https://www.deviantart.com/>
- Google fonts - great source of fonts. <https://fonts.google.com/>
- BFXR - make sound effects. <https://www.bfxr.net/>