

# Playtest / Code Review

---

## Outward Appearance

---

- Does the interactive card "make sense"?
  - Can you tell what you are supposed to do?
  - Can you tell when you are done?
- Can the visuals be improved? For example:
  - Is the text easily readable? Find somebody with "imperfect" eyesight to test for you.
  - Is there adequate contrast between elements?
  - Are the colors well chosen (remember that about colorblindness is common, about 10% of the population, mostly men, has some form of colorblindness).
- General observations:
  - Is the layout attractive? Is there anything that could be done to make it more so?
  - Do dynamic elements overlap other elements? Is that ok?
  - Do you notice any "glitches" -- anything that seems like a bug?

## Testing

---

Read through the code to understand what it is supposed to do. As you are reading, pay particular attention to the if statments and see if you can come up with ideas for how to test them.

- How do you interact with the card?
  - Do all of the interactions work?
- What is supposed to happen when you're done?
  - Does it happen?
- Are there other behaviors described in the code?
  - Do they work?

## Code Review

---

Things to look for (and which should be corrected).

- Error (red square) or warning (yellow triangle) markers.
- Variables named "sprite" -- rename with a meaningful name.
- Animations without meaningful names -- for example: `cat` and `*cat_copy_1` should be renamed, something like `"cat_sitting"` and `"cat_standing"` would help the reader to understand why there were two cat animations and what was different about them.
- Duplicate / redundant function calls. Look for:
  - Setting an object property that never changes in the `draw( )` function. Make those calls outside of the `draw( )` function.
  - Multiple calls to `drawSprites( )` -- in general you only need one call, usually in the `draw( )` function. If you think you need more than one call to `drawSprites( )` there must be a comment explaining why you need more

than one call.

- Calls to `background( )` and other functions that need to be called for every frame *outside* of the `draw( )` function.
- Placing the `draw( )` function in the middle of the program.
  - JavaScript won't have a problem with this, but it will make the code harder to read and reason about. Move it to the end of the program.
- Inconsistent capitalization.
- "Magic" numbers. Any value that you use more than once is a good candidate for conversion to a variable. Especially:
  - If the value doesn't mean the same thing everywhere it's used -- for example if it is used for both an X and a Y value.
  - If the value is used a lot. Or if the value is derived from another value.
  - If the meaning of the value is not very clear from context. For example, hex color values (`#11AA2D`) are hard for most people to visualize, give them a name that describes the color or its use.
  - If the value might need to be changed to tune the behavior of the program.
- Too few, or too many, comments.