Game Principles Check List

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Principles | G | M | B | NT | Comments |
| Direction |  |  |  |  |  |
| **1. Focal Point** |  |  |  |  |  |
| **2. Anticipation** |  |  |  |  |  |
| **3. Announce Change** |  |  |  |  |  |
| **Behavior** |  |  |  |  |  |
| **4. Believable Events and Behavior** |  |  |  |  |  |
| **5. Overlapping Events and Behavior** |  |  |  |  |  |
| **6. Physics** |  |  |  |  |  |
| **7. Sound** |  |  |  |  |  |
| **Progression** |  |  |  |  |  |
| **8. Pacing** |  |  |  |  |  |
| **Environment** |  |  |  |  |  |
| **9. Spacing** |  |  |  |  |  |
| **Method** |  |  |  |  |  |
| **10. Linear Design versus Component Breakdown** |  |  |  |  |  |
| **Foundation** |  |  |  |  |  |
| **11. Player** |  |  |  |  |  |
| **12. Communication** |  |  |  |  |  |
| **13. Appeal** |  |  |  |  |  |

**1. Focal Point**

Never allow the player to guess what they should focus on. At the same time, always allow secondary subject matter, but it is the designer's job to clearly provide the primary focus at all times. This applies to both visual and visceral aspects of gameplay.

**2. Anticipation**

Time is needed to inform the player that something is about to happen. Always factor in Anticipation when designing and implementing events and behaviors.

**3. Announce Change**

Communicate all changes to the player. This short step occurs between Anticipation and the event itself.

**4. Believable Events and Behavior**

Every event or behavior must occur according to the logic and expectations of the player. Every action, reaction, results, emotion and conveyance must satisfy the players' subconscious acceptance test.

**5. Overlapping Events and Behavior**

Dynamic is lost if only one change occurs at a time. Discover the right amount of events to occur at any given moment of time.

**6. Physics**

The player's primary logic operates within the known possibilities of physics. Keep in mind gravity, weight, mass, density, force, buoyancy, elasticity, etc. Use this as the starting point, but do not be limited by it.

**7. Sound**

Ask yourself, "What sound does it make when \_\_\_\_\_\_\_\_ happens?" "Is the sound appropriate?" "Is the sound necessary?" "Does it benefit the experience or hinder it?" If players close their eyes, the sound alone should still achieve the desired affect.

**8. Pacing**

Keep in mind the desired sense of urgency, the rate in which events occur, the level of concentration required and how often events are being repeated. Spread out the moments of high concentration, mix up the sense of urgency, and change things wherever possible to achieve the proper affect.

**9. Spacing**

Understand how much space is available both on-screen and in-world, recognize the spatial relationship between elements and take into account the effects of modifying those spaces.

**10. Linear Design versus Component Breakdown**

Linear Design involves solving challenges as they come. All solutions and possibilities hold the same institutional value. Focus can be lost with this method, but it provides creative and spontaneous solutions.

Component Breakdown involves systemic categorization and forming a logical hierarchy of all solutions. This method can restrict innovation but preserves clarity of primary design objectives.

**11. Player**

How does the player factor into this? How does the player interact with everything that has been designed? More than just device input, address how the player contributes to the experience. If it's a good idea and you're able to convey it correctly but the player is not into it, change it or scrap it!

**12. Communication**

Is the appropriate team member correctly aware of the objective? Are the appropriate developers clear on the solution? If it's a good idea but you can't communicate it correctly, it might as well be a bad idea because it's very likely to be received as such.

**13. Appeal**

When addressing anyone, ask yourself, "Does this draw the audience in?" This applies to (but is not limited to) the player, the spectator, your fellow developers, the publisher, and their marketing team. If it's not a good idea, there's no need to continue until it becomes a good idea or is replaced by something better.