

GAME DESIGN FUNDAMENTALS: A Critical Workshop for Practitioners

Core Theme:

The **Game Designer** is the primary author of the game's soul through living documents (GDD, pitch decks, spreadsheets, asset lists, iteration logs). These documents are not bureaucracy—they are the entire core of the game. Every mechanic, progression beat, and emotional moment exists first on paper (or in Google Docs) and is approved, revised, or cut at the joint discretion of Game Designers and Directors.

PART 1: Deconstructing the Game Designer's Role (20 min)

The Myth vs. Reality

Game designers are not lone geniuses. They are **document-driven problem-solvers**.

The Reality: Designers are Problem-Solvers, Not Visionaries

Game designers aren't lone auteurs conjuring masterpieces. They're:

- **Documenters:** Writing design docs, asset lists, spreadsheets
- **Collaborators:** Working with artists, programmers, writers, QA
- **Iterators:** Endlessly tweaking based on playtesting feedback
- **Communicators:** Pitching ideas, defending choices, compromising

The Documents That Create the Game

Document	Purpose	Who owns it	How the game progresses from it
Game Design Document (GDD)	The living bible: vision, core loop, mechanics, systems, progression, levels	Lead / Systems Designers	Milestones are built section-by-section
Pitch Deck	10–30 slides that sell the emotional contract to directors & publishers	Game Director + Lead Designer	Greenlight → vertical slice → full production
Balance Spreadsheets	Economy, progression curves, damage values, spawn tables	Systems Designers	Updated after every playtest; drives iteration
Asset Lists & Flowcharts	Exact list of animations, UI screens, dialogue lines, encounters	Level / Quest / Technical Designers	Artists & programmers work directly from these
Playtest Reports & Iteration Logs	"What broke, what felt wrong, what to change"	All designers → consolidated by leads	Only changes approved by Designer + Director enter the game

The game literally does not exist until these documents do.

Modern AAA Fragmentation

Role Fragmentation in Modern AAA:

- **Gameplay Designers:** Second-to-second interactions (combat feel, movement) → writes combat feel section of GDD
- **Systems Designers:** Overarching mechanics (progression, economy) → owns the spreadsheets that make progression feel fair
- **Level Designers:** Environmental design, encounter placement → authors encounter documents and gray-box maps
- **Quest Designers:** Narrative missions, branching paths
- **Technical Designers:** Bridge between design and code
- **UX Designers:** Interface, accessibility, player communication

All feed back into the master GDD; Directors arbitrate conflicts.

Modern Example, God of War Ragnarök (2022):

The game features combo finishers where players build up a stun meter, Pre-Fall attacks from ledges with different animations per weapon, and two shield types (Dauntless Shield for parrying and Stonewall Shield for blocking). Each of these systems required separate designers coordinating their work.

Modern Example, Red Dead Redemption 2 (2018):

RDR2 had designers focused exclusively on horse systems (bonding mechanics, saddle stats, breed behaviors). Most design work happens in gray box (untextured geometry). The game took 7 years to develop with massive team specialization.

The Uncomfortable Truth:

- Elden Ring represents the "peak of a very specific, influential video game formula" (polishing existing ideas rather than revolutionary innovation)

- When designers spend years on subsystems (RDR2's horse bonding, honor system, camp donations), they're solving micro-problems, not creating grand visions

Critical Question

When a GDD becomes 800 pages and 200 people touch it, who truly owns the vision?

Critical Lens:

How do AAA production pipelines constrain creative agency? When does specialization become creative fragmentation?

PART 2: The MDA Framework as Critical Tool (30 min)

Mechanics → Dynamics → Aesthetics: Reverse-Engineering Player Manipulation

How Designers Actually Use MDA

Understanding the Framework:

- **Mechanics:** Rules, systems, stats that exist in code
- **Dynamics:** How players act/behave in response to mechanics
- **Aesthetics:** Emotional responses players feel

Every major mechanic is written as an MDA table in the GDD:

Mechanic (what the code does)	Dynamic (what players actually do)	Aesthetic (intended emotion)
Manual save stations take 6 seconds	Players hide, panic, listen for the alien	Terror & isolation
Weapons break after ~40 hits	Constant experimentation, hoarding reduced	Resourcefulness, impermanence
Honor -8 to +8, store discounts at high end	Players greet everyone to farm honor	Moral dissonance

Designers can only directly edit the left column. The rest is prediction → playtest → revise document.

The Cascade Effect:

Designers can only directly control mechanics, but changes cascade down:

- Change ammo capacity → Changes player behavior → Changes emotional experience

Case Studies (GDD Evolution)

Deep Analysis, Alien Isolation's Evolution:

Original System:

- **Mechanic:** Automatic checkpoints (borrowed from Call of Duty/BioShock)
- **Dynamic:** Players walked around carelessly knowing death = minor progress loss
- **Aesthetic:** No fear (completely wrong for horror)

Final System:

- **Mechanic:** Manual save stations that take time to operate
- **Dynamic:** Players became terrified of dying; saving itself became tense
- **Aesthetic:** Terror and isolation (perfect for horror)

Lesson: Context destroys universality. What works in action games kills horror games.

Modern Example, Baldur's Gate 3 (2023):

Baldur's Gate 3

GDD explicitly states: "dice rolls must never block plot artifacts" → forces save-scumming dynamic → accepted because vision is "player expression".

Larian programmed BG3 to "twist and turn on itself to ensure that, regardless of the choices a player wanted to make, they'd always end up with an artifact in their inventory necessary to progress the game's plot."

- **Mechanic:** D&D 5e dice rolls for dialogue checks
- **Dynamic:** Players save-scum before important conversations, experiment with different character builds for checks
- **Aesthetic:** Turn-based combat "can be less overwhelming than quick beat-'em-ups and quick time events," allowing players to "analyze the situation and take my time when planning out a plan of action"

Modern Example, Zelda: Breath of the Wild's Breaking Weapons:

- **Mechanic:** All weapons degrade and eventually break
- **Dynamics:**
 - Players hoard less, experiment with variety
 - Constant engagement with world's resource systems
 - Creative problem-solving in combat
- **Aesthetics:**
 - Feeling of exploration in a decaying world
 - Resourcefulness rather than power fantasy
 - Every weapon feels temporary, precious

Counter-Question for Discussion:

Does this mechanic annoy hardcore players? Is frustration always bad design? (Answer: It depends on your vision)

Modern Example, Red Dead Redemption 2's Honor System:

Red Dead Redemption 2 – Honor System

GDD promised "internal morality vs. reputation split" but never clearly communicated mask behavior → players felt betrayed. Mechanical contradiction born in documentation.

How It Works:

- Scale from -8 (dishonorable) to +8 (honorable)
- Positive actions: Greeting NPCs, donating to camp (\$20 = big honor boost), helping strangers, sparing lives, doing camp chores
- Negative actions: Killing innocents, stealing, robbing, antagonizing

MDA Breakdown:

- **Mechanics:**
 - Honor affects NPC reactions (high honor = friendlier civilians, easier to defuse situations, witnesses accept bribes)
 - Store discounts at high honor (10% at rank 2, 25% at rank 4, 50% at ranks 7-8)
 - Better loot from corpses at low honor (Dead Eye tonics, weapons, alcohol)
 - Unique weapon grips unlock at +4 and -4
 - Chapter 6 has 1.5x honor multiplier
- **Dynamics:**
 - Players greet everyone in saloons to "farm" positive honor
 - Players wear masks when robbing (though this doesn't prevent honor loss, only witness recognition)
 - Players donate \$20 repeatedly instead of \$100 once (more honor per dollar)
 - Creates dissonance: "The game both wants you to be a criminal... but also not be a criminal"
- **Aesthetics:**
 - High honor = feeling of redemption, Arthur's self-reflection
 - Low honor = ruthless outlaw, darker path
 - **But creates confusion:** Required story missions force you to rob/kill, then free roam punishes you for the same actions

The Design Problem:

When the story demands that you behave one way and then in your free time you can act completely differently, what's meant to be a complex character instead smacks of ambivalence not by intent but by virtue of gameplay mechanics

This is **mechanical contradiction** in action, the honor system doesn't align with the game's mandatory criminal activities.

Activity (3 min)

Pick any mechanic from a recent game. Write its one-row MDA in the chat/notes.

PART 3: Vision-Driven Design vs. Design-by-Committee (25 min)

The Ideology Behind Every Mechanic

Vision is not a tagline. It is the first page of the GDD.

Defining Vision (Not Vague "Fun"):

Vision = Emotional contract with the player. What specific feeling are you promising?

Example vision statements that actually shipped:

Game	Vision Statement in GDD / Pitch Deck	Everything in the documents must serve this
Doom (2016)	"Push-forward combat – you are the predator"	Glory kills, no reload, fleeing demons
Subnautica	"Thrill of the unknown"	Darkness + danger curve
Hades	"Failure is progress"	Death = narrative reward
Flower	"Calm, meditative joy"	Removed levels, spells, timers, failure

Modern Examples of Clear Vision:

Subnautica: "Thrill of the unknown"

- Mechanic: Deep ocean becomes darker, more dangerous
- Dynamic: Players venture cautiously into depths
- Aesthetic: Fear mixed with curiosity

Resident Evil Village: "Struggle to survive"

- Mechanic: Limited resources, overwhelming enemies
- Dynamic: Players must manage inventory, choose fights carefully
- Aesthetic: Vulnerability, resourcefulness

Doom 2016: "Push forward combat"

- Mechanic: Glory kills restore health, enemies drop ammo
- Dynamic: Players sprint toward danger, not away
- Aesthetic: Unstoppable predator, power fantasy

FTL: "Recreate the atmosphere of commanding a Starship"

- Mechanic: Pause-and-play crew management during crises
- Dynamic: Players make captain-like tactical decisions under pressure
- Aesthetic: Star Trek-style command authority

Flower's Radical Subtractions

Originally had: level-ups, spells, resource management, time limits ("expected features of other video games")

All removed because they contradicted intended emotions of relaxation and calm.

Genova Chen: "We've played so many games growing up, these bad habits form. A lot of the time we like to make things very fun, but fun doesn't always help the emotion you want to deliver."

The Radical Act: Removing what "every game has"

When Everything Aligns, Doom 2016:

Vision: "Push forward combat"

Supporting mechanics:

- Rapid movement speed
- Glory kills require closing distance, restore health
- Demons programmed to flee from you
- No reloading (constant aggression)

Supporting non-mechanics:

- Heavy metal soundtrack
- Violent, empowering animations

- Doomslayer's confident personality
- Visual design emphasizes predatory nature

Genova Chen: "All the elements have to sing the same notes to make the impact strong"

The Uncomfortable Reality, Callisto Protocol:

- Mechanic 1: Limited ammo (creates fear, vulnerability)
- Mechanic 2: One-button instant stealth kills (creates power, confidence)
- Result: **Mechanical contradiction** (which emotion wins?)

Modern Example, Hades (2020):

"What if there was a roguelike with narrative continuity, where every time you run into a boss, they remember, you start keeping a tally of who won this time, who won last time."

Vision: "Failure is progress" (narrative advancement regardless of skill)

- The goal is "not to get good," shifting focus "away from mastery and towards a more holistic, thematic approach"
- Progress and narrative advancement isn't tied to winning like in many other roguelikes
- Death becomes a narrative opportunity, not punishment

Red Dead Redemption 2 Camp System – Vision vs. Documentation Failure

GDD Vision: "Feel part of a living family; contributions matter"

Reality in docs: Donations never change story outcome → false choice.
The documents promised an emotional contract the code could not keep.

Modern Example, Red Dead Redemption 2's Camp System:

Vision: Create feeling of being part of a community, a gang member contributing to collective survival

Mechanics:

- Donation box accepts money, valuables, pelts, meat
- Ledger shows gang contributions, allows camp upgrades
- Camp has supply wagons (food, medicine, ammunition) with three tiers: Red (depleted), White (stocked), Gold (fully stocked)
- Upgrades include: Dutch's tent (encourages others to donate), Arthur's lodging (unlocks fast travel), Pearson's wagon (food variety), ammunition wagon, horse station (\$300), leather working tools (better satchels)
- Camp chores: carrying hay, chopping wood, feeding animals (small honor boosts)

Dynamics:

- Players donate \$20 repeatedly to farm honor
- Players rarely use camp supplies (abundant elsewhere)
- Players prioritize fast travel upgrade (\$325 + previous tier), ignore cosmetic upgrades
- Other gang members contribute automatically through story

Aesthetics:

- **Intended:** Feeling of community, collective responsibility
- **Actual:** Feels like busywork with minimal impact

The Design Failure:

The game does a poor job of letting the player feel the influence of their actions on the camp. Regardless of whether the player buys all RDR2's camp upgrades, does all the chores, and talks to every NPC, the Van der Linde camp will still be there for the player to return to and continue the story as if nothing happened

Why This Matters:

When mechanics don't meaningfully affect the world, players see through the simulation. RDR2's camp donations are a **false choice**, the game continues identically whether you engage or ignore the system.

BREAK (5 min)

PART 4: The Political Economy of Borrowing Mechanics (30 min)

Why the Industry Copies Itself to Death

Borrowing is Mandatory – Documentation is the Filter

Section A: The Borrowing Imperative (10 min)

Game design exists in **intertextual space** (no mechanic is truly original).

Every designer keeps a "reference GDD" folder of mechanics they love.
The disciplined ones write:

1. What is the original MDA?
2. Does that aesthetic serve MY vision document?
3. How must I change the mechanic (and its spreadsheet) to fit?

Economic Pressure: Publishers fund what's proven

Modern Example, Far Cry 2 vs Far Cry 4:

Same franchise, same setting, radically different experiences:

Far Cry 2:

- Weapons jam, malaria attacks, jeeps break down
- Saves are rare, far apart
- Result: Harrowing struggle for survival

Far Cry 4:

- Abundant ammo, reliable weapons
- Track enemies easily
- Permanent progress
- Result: Power fantasy action movie

Critical Insight: Your game's "feel" comes from mechanical choices, not narrative window-dressing.

The Right Way to Steal:

1. Analyze through MDA: What does this mechanic make players DO? How does it make them FEEL?
2. Ask: Does that feeling serve MY vision?
3. Adapt ruthlessly (don't just import)

Modern Example, Elden Ring (2022):

FromSoftware has been "working on the same formula since Demon's Souls in 2009," with "major changes to setting and mechanics," but continuously "refining and experimenting with each successive outing"

Elden Ring combines "the design pillars that FromSoftware popularized back in Dark Souls," representing "the peak of a very specific, influential video game formula" rather than radical reinvention.

Elden Ring's GDD is literally titled "Refinement of the Souls Formula Since 2009".

Section B: When Copying Becomes Derivative (10 min)

The Soulslike Problem:

Early wave just copied:

- Bonfires for checkpoints
- Estus flasks for healing
- Souls as currency
- Stamina-based combat

Treated genre as **checklist**, not philosophy.

If you consider a genre "a long list of must-have mechanics," you'll "end up making the same thing over and over again"

Dead Space's Sonic Identity Crisis:

EA initially requested "predictable sci-fi soundtrack" (electronics and drums, because that's what space games had)

Made players feel **heroic**, not scared

Composer had to fight to align music with horror vision

Gary Napper (Alien Isolation): "Like any development team we all play a lot of games and each have our favorites. Often our decisions and choices are colored by the games we play" but those choices must fit your specific game.

Section C: Mechanical Contradictions (10 min)

When Borrowed Mechanics Fight Each Other:

Examples of Borrowing Gone Wrong (Document Mismatch)

- Alien: Isolation – auto-checkpoints copied from CoD → killed horror → rewritten
- RDR2 masks – disguise mechanic copied but honor effect not explained → player confusion
- Callisto Protocol – limited ammo (fear) + one-button stealth kills (power) → contradictory docs

Alien Isolation's Degenerate Strategy:

- Automatic checkpoints (convenient, forgiving) vs. deadly persistent alien (scary)
- Result: Players rushed to next checkpoint, got killed, respawned safely
- Borrowed "convenient" mechanic undermined "scary" mechanic

Solution: Manual save stations that take time and involve risk

- "Saving became tense, looking for a save became tense"
- "The simple act of saving had become supportive to the game's driving factors of terror and isolation"

The Flower Philosophy Revisited:

Sometimes "fun doesn't always help the emotion you want to deliver"

Have the courage to remove what "should" be there according to genre conventions.

Modern Example, The Last of Us Part II (2020):

"Non-disabled players are using these features to adjust the game to their preferences and eliminate frustration. Accessible design is good universal design."

Combat accessibility settings fundamentally alter difficulty:

- "Invisible while prone" feature, slow motion while aiming, reduced enemy perception
- These could "clash" with survival horror vision
- Solution: "Combat accessibility settings are designed to make combat accessible for all players. As such, they can significantly alter the gameplay experience" (explicit warning that they change the intended experience)

Modern Example, Red Dead Redemption 2's Mask System:

The Borrowed Mechanic:

- Masks should hide identity (like in heist games)
- Players expect: wear mask → no honor loss, no witness recognition

The Reality:

Wearing a bandana does not grant the player amnesty from changes in honor: it only prevents eyewitnesses from immediately recognizing the player. This is because, in this game, honor is more about Arthur's self-reflection instead of his reputation

The Problem:

- Game never explains this
- Players feel betrayed: "The hell's the point of masks if I still lose honor?"
- Borrowed mechanic (disguise system) doesn't serve the vision (internal morality system)

The Lesson:

When you borrow a mechanic, players bring expectations from the original context. If your implementation differs, you must communicate that clearly, or risk player confusion and frustration.

Key Takeaway:

- Borrowing is inevitable and good
- But you must understand the mechanic's emotional purpose
- Test if it serves YOUR vision, not just the game you borrowed from
- Be willing to cut mechanics that don't fit, even if they're "standard"
- Communicate when borrowed mechanics work differently than expected

PART 5: Design for Humans, Not Theory (20 min)

Players Will Break Everything

Playtesting – The Moment Documents Meet Reality

Your brain is a terrible playtester.
Only real players reveal:

- Degenerate strategies (RDR2 Dead Eye trivialises combat)
- Misunderstood mechanics (RDR2 masks)
- False choices (RDR2 camp donations)

The Foundational Problem:

Your brain is a "truly bad video game simulator" (everything seems fun in your mind's eye).

You can only spot problems, design flaws, and unintended issues when the game is real.

A game idea is "worthless until you've proven its value through a prototype"

What Playtesting Actually Reveals:

Modern Example, Baldur's Gate 3:

Larian "made excellent use of the 'early access' period, using it to address many game-breaking bugs, much-needed balance changes, and incorporating the feedback of fans"

"Intelligent enemy AI responds to your actions so even repeated combat encounters against the same enemies feel distinct, with your choices shaping fights to play out in myriad ways"

Players find:

- Exploits you never imagined
- Ways to sequence-break
- Misunderstandings of "obvious" mechanics
- Degenerate strategies that undermine your goals

Sometimes It's Emergent Fun:

- Rocket League's flying cars (unintended but perfectly fit the vision)

Sometimes It's Degenerate:

- Alien Isolation's checkpoint rushing (undermined horror experience)

Every playtest produces an iteration log → Designer rewrites GDD sections → Director approves or rejects.

Modern Example, Red Dead Redemption 2's Dead Eye System:

The Mechanic:

- Slow-motion targeting system
- Level 1: Slow time manually
- Level 2: Mark targets
- Level 3: Stay in Dead Eye after kills
- Level 4: Identify fatal areas
- Level 5: Critical hit markers

The Player Behavior:

Players discovered Dead Eye trivializes most combat encounters. With enough Dead Eye tonic items (easily farmable), players can:

- Clear entire enemy camps without risk
- Win duels instantly
- Hunt legendary animals with no challenge

The Design Problem:

The game gives you a powerful tool but doesn't balance encounters around it. Result: combat becomes either trivially easy (with Dead Eye) or tediously difficult (without it).

What Playtesting Should Have Revealed:

Players will always optimize the fun out of a game if given the chance. If Dead Eye is too powerful and resources are abundant, skilled players will feel forced to self-impose limitations.

The Target Audience Problem:

Target Audience is also decided in the documents

Spider-Man (2018) GDD explicitly says "accessible power fantasy for casual audience" → simple swinging is a feature, not a bug.

Modern Example, Spider-Man (2018):

Critic Perspective: Web-swinging too simple, no challenge, impossible to fail

- Wanted complexity, high skill ceiling, meaningful failure stakes

Actual Target Audience: Casual PS4 gamers

- Got an "incredible sensation of being Spider-Man"
- Accessible power fantasy without frustration

The Lesson: A game is built for a "target audience, a specific type of person or player" (you can't please everyone).

Your job as a critic: Recognize when a game isn't made for you vs. when it failed its actual audience.

Modern Example, God of War Ragnarök (2022):

Ragnarök is "more of a polished iteration of 2018's God of War than a meaningful evolution," adding dog-sled vehicles and grappling via Blades of Chaos for traversal, but "that's about all that Ragnarök brings to the table on the traversal front"

Question: Is this iterative refinement or creative stagnation? Answer depends on whether you're the target audience for polished experiences vs. innovation.

The Accessibility Question:

Modern Example, Celeste's Assist Mode:

Explains how the game is "supposed to be played" but offers difficulty reduction if needed:

- For new players
- For players with disabilities
- For players more interested in story than challenge

Critical insight: Options don't threaten vision if you communicate the intended experience.

Can accessibility options live in the core GDD without breaking vision?

Modern Example, The Last of Us Part II:

Over 60 accessibility features including three presets for vision, hearing, and motor accessibility, with text-to-speech, high-contrast view, combat audio cues, awareness indicators, and fully customizable controls

Full control customization allows remapping "every command to a different controller input, including touch pad swipes and controller shake," with options to change "every button hold into a toggle, and every rapid press into a hold"

The team approached it as "a completely blank canvas," comparable to "the switch to 3D with PlayStation 1 and Nintendo 64 when designers had to figure out how to let players control the camera"

Result: Considered "by some field experts, as the most accessible game ever produced"

Modern Example, Red Dead Redemption 2's Travel System:

The Design Challenge:

Large open world (29 square miles), slow travel speed, long distances between missions

What Players Do:

- Fast travel from camp (requires \$300+ in upgrades)
- Use cinematic camera for auto-pilot on roads
- Take trains/stagecoaches between towns
- But still spend significant time in transit

The Playtest Insight:

Some players love the slow, contemplative travel (supports immersion vision). Other players find it tedious busywork (conflicts with pacing expectations).

Rockstar's Solution:

Provide multiple travel options, but make fast travel deliberately inconvenient (must return to camp, can only travel to discovered locations, costs money to unlock).

The Design Philosophy:

"We want you to experience the journey, not skip it" (but we'll give you an out if you really want it).

PART 6: The Realities of Getting Into Game Design (15 min)

Beyond the Theory

Getting Into Game Design – Your Portfolio IS Your Documents

What hiring managers actually want to see:

1. A clean, well-structured GDD (even 5–10 pages)
2. Balance spreadsheets with revision history

3. Before/after playtest notes showing you iterated
4. A prototype + devlog proving the document became reality

What Actually Matters:

1. **Portfolio over credentials:** Make stuff, show your ability to design; put projects in a portfolio
2. **Design thinking:** Can you analyze mechanics critically?
3. **Soft skills:** Collaboration, handling criticism, pitching ideas
4. **Experience** (even tangential)

Portfolio Reality Check:

- Small projects, vertical slices, game jam prototypes ("don't need to be full games")
- Must have "clear documentation for how you dreamt up, designed, implemented, and refined your ideas"
- Focus on "best work" and "quality over quantity"
- Use Unity, Unreal, Dreams, Tabletop Simulator
- Mod existing games with level editors

Paths: game jams → mods → QA → junior design → indie or AAA.

Modern Example, Level Designer Success Story:

One designer got hired because they'd made a level for the firm's previous game (showed initiative and understanding of the studio's design language).

Alternative Paths:

QA/Testing Route:

- Start in QA ("not uncommon for qa testers... to impress their bosses and get moved into design roles")
- **Important:** "QA is a vitally important role in of itself and shouldn't be seen as just a stepping stone"

Reality Check:

- Your first job "probably won't be designing games at Blizzard or Bungie"
- Get experience at smaller studios first (mobile games, kids games, even gambling games)
- Build up to AAA

University: Worth It?

Almost everyone warned: "The college diploma itself isn't the important bit... it's about making the most of the opportunities and connections"

A game design degree is:

- "Not a guarantee of a good job"
- "Not a requirement" for many positions

What universities DO provide:

- Time to build portfolio
- Industry connections
- Access to internships/work experience
- Collaborative environment

Interview Warfare:

Design Thinking Tests:

- "What would happen if you remove one of the options in rock paper scissors?" (testing your understanding of game balance)
- They'll dismantle your portfolio pieces "to see how you'll react to criticism"

Personality Tests:

- Are you a team player?
- Can you adapt to someone else's vision?
- Can you communicate effectively?
- Are you resilient to feedback?

Design Tests:

- Take an existing game and add a new mechanic
- Design a level/encounter for the studio's game
- Often done on-site under time pressure

Red Dead Redemption 2 Example:

If interviewing at Rockstar, you might be asked:

- "How would you redesign RDR2's honor system to better align with the story's mandatory criminal actions?"
- "Design a camp interaction system that makes player donations feel more impactful"
- "Create a new random encounter type that fits RDR2's reactive world"

Industry's Dark Side:

The Challenges:

- Crunch (extreme overtime before milestones, RDR2 dev mentioned "100-hour weeks")
- Mass layoffs/studio closures (volatile industry)
- Online abuse and threats
- Harassment in workplace

The Response:

"While these problems exist they should not discourage you from joining the business and... strides are being made to improve these practices"

Self-Protection:

- Research studios on Glassdoor before accepting offers
- Talk to current/former employees
- Look for warning signs of toxic culture

Geographic Limitations:

- Few developers in India, South America, parts of Asia
- May need to relocate for best opportunities
- Remote work expanding options

The Indie Alternative:

Thousands of indie games released every year prove individuals and small teams can make games without industry involvement.

Solo designers of Axiom Verge, Stardew Valley, and Gunpoint had "way more creative control than the dude working on Red Dead Redemption's horses"

But: Not easier, no guaranteed income, extremely competitive market.

PART 7: Key Takeaways & Final Q&A (10 min)

The Essential Lessons for Critics-as-Designers

1. The game is the documents. Everything else is implementation.
2. Vision lives on page 1 of the GDD; every mechanic must justify its existence against it.
3. Borrowing without MDA documentation = mechanical contradiction.
4. Playtesting is the only truth serum for design documents.
5. Great critics already reverse-engineer GDDs when they analyse games – you're 60% of the way to being a designer.

Additional Key Takeaways:

6. Mechanics create experience, not narrative
 - Far Cry 2 vs Far Cry 4: Same setting, different mechanics = completely different games
 - RDR2's honor system creates moral confusion through mechanical contradiction
7. "It depends"
 - Design choices must serve your specific vision
 - No universal "right" or "wrong" mechanics
 - High scores, lives, permadeath (all context-dependent)
8. Borrowing mechanics is essential
 - Elden Ring and BG3 succeeded through "iteration" (decades of "refining and experimenting")
 - But understand WHY they work first
 - Adapt, don't copy blindly
9. Test if borrowed mechanics fit YOUR vision
 - Alien Isolation's checkpoints worked in Call of Duty, killed horror in their game

- Dead Space's heroic music worked in space games, killed horror in theirs
- RDR2's mask system borrowed disguise mechanics but served different purpose (internal morality vs. external reputation)

10. Mechanics can fight each other

- Callisto Protocol: Limited ammo vs. instant stealth kills
- RDR2: Story demands crime, honor system punishes crime
- Creates conflicting player feelings, undermines coherent vision

11. Sometimes the best choice is removing "standard" features

- Flower removed levels, spells, resources, time limits
- "Fun doesn't always help the emotion you want to deliver"
- RDR2 could have removed camp donations entirely if they didn't meaningfully impact the world

12. Your brain is a terrible game simulator

- Prototype everything
- "A game idea is worthless until you've proven its value through a prototype"
- RDR2's Dead Eye seemed like empowerment tool, but playtesting should have revealed it trivializes combat

13. Design for your target audience

- Spider-Man was "made for casual gamers," not hardcore critics
- BG3 offers "freedom and flexibility" that rivals "old school text-based RPGs" but in a polished package for modern audiences
- RDR2's slow travel serves immersion vision for some, frustrates others seeking pacing

14. Playtesting reveals everything

- Players will surprise you
- Design is problem-solving, not just ideation
- Iteration is the core of game design
- BG3's early access caught issues before launch

15. Keep questioning

- Even these lessons aren't universal truths
- "Any lesson you've learned needs to be constantly re-evaluated"
- Never accept design wisdom "as gospel, especially if it's coming from a YouTuber"

Final Designer + Director Mantra

"Does this change make the player feel what page 1 promised? If not, cut it—no matter how cool."

The Designer's Mantra:

"It depends on the game you are trying to create, the emotions you're trying to evoke, and the experience you want your players to have." (Yacht Club Games)

For Critics Specifically:

Your Advantage:

- Analysis ≠ Design, but critical thinking is 50% of the job
- Your ability to deconstruct games is your competitive advantage
- Use analytical skills to reverse-engineer successful mechanics

Your Challenge:

- Understanding why something works ≠ ability to execute it
- Must learn collaboration, communication, iteration
- Must move from observer to participant mindset

The Meta-Lesson:

Game design is:

- Not about having great ideas (ideas are worthless until prototyped)
- Not about following rules (all rules are context-dependent)
- Not about pleasing everyone (design for target audience)

Game design IS about:

- Understanding how mechanics create player behavior and feelings
- Testing relentlessly to find what actually works
- Solving problems that emerge from playtesting

- Communicating vision and collaborating with others
- Iterating based on feedback
- Making informed choices about what to borrow and what to cut

Red Dead Redemption 2: The Design Autopsy

What RDR2 Teaches Us:

The Good:

- MDA in action: Honor system, camp mechanics, Dead Eye all create specific dynamics
- Vision-driven world building: Every system supports "living, breathing world"
- Attention to detail: Horse bonding, NPC reactivity, random encounters

The Bad:

- Mechanical contradictions: Story demands crime, honor punishes crime
- False choices: Camp donations don't meaningfully impact world
- Borrowed mechanics without adaptation: Mask system confuses players

The Ugly:

- Development cost: 7 years, 100-hour weeks, massive team specialization
- When does perfectionism become diminishing returns?

****The Lesson:****

Even masterpieces have design flaws. Your job as a critic-turned-designer is to:

1. Identify these contradictions
2. Understand why they happened
3. Learn how to avoid them in your own work

Final Thought:

As critics, you already ask "why does this work?" when analyzing games.

As designers, you'll ask "how can I make this work?" when creating games.

The transition requires:

- Humility (your first designs will be bad)
- Collaboration (you can't do it alone)
- Iteration (nothing works on the first try)
- Resilience (most of your ideas will fail)

But your critical eye (your ability to see WHY games succeed or fail) is the foundation everything else builds on.

Suggested Q&A Starters

1. How would you rewrite RDR2's honor system GDD to remove contradiction?
2. When should a Director overrule a Designer's document?
3. Can accessibility options live in the core GDD without breaking vision?

****Additional Discussion Questions:****

4. When does iteration become derivative? (Elden Ring vs. early soulslikes)
5. Can accessibility features compromise artistic vision? (TLOU2 vs. Celeste vs. Dark Souls)
6. Is narrative continuity in roguelikes (Hades) the future or a novelty?
7. Do critics have an ethical obligation to consider target audience in reviews?
8. Should studios prioritize innovation or refinement? (BG3 vs. GoW Ragnarök approaches)
9. When is "false choice" acceptable in game design? (RDR2's camp donations)
10. Should games respect player time or force intended pacing? (RDR2's travel system)

APPENDIX A – Games Referenced

Elden Ring • Baldur's Gate 3 • God of War Ragnarök • Red Dead Redemption 2 • Hades • Doom (2016) • Alien: Isolation • The Last of Us Part II • Flower • Subnautica • Celeste • Spider-Man (2018) • Resident Evil Village • FTL • Far Cry 2 • Far Cry 4 • Dead Space • Callisto Protocol

APPENDIX B – One-Page GDD Template (Handout)

1. Vision (one sentence)
2. Core Loop (diagram)

3. MDA Table (top 5 mechanics)
4. Progression Overview (spreadsheet snippet)
5. Open Questions / Risks

The Core Loop Analysis Framework

Use this framework to analyze any game's core loop:

Step 1: Identify the Core Loop

- What does the player do repeatedly?
- Example (RDR2): Travel → Encounter event → Make choice → Receive reward/consequence → Return to camp

Step 2: Break Down Through MDA

- Mechanics: What rules/systems govern this loop?
- Dynamics: How do players actually behave?
- Aesthetics: What emotions result?

Step 3: Identify Contradictions

- Do any mechanics fight each other?
- Do borrowed mechanics serve the vision?
- Are there false choices that don't impact the game?

Step 4: Evaluate Against Vision

- Does this loop support the intended experience?
- What could be removed?
- What needs iteration?

Practice: Analyze These Core Loops

Hades:

- Loop: Leave underworld → Clear rooms → Collect boons → Die → Conversation → Upgrade → Repeat
- Vision: "Failure is progress"
- Question: How do the boon choices create different dynamics each run?

Baldur's Gate 3:

- Loop: Explore → Encounter → Dialogue/Combat choice → Consequence → Progress story
- Vision: "Player agency and emergent narrative"
- Question: How does the dice roll mechanic support or undermine player agency?

Red Dead Redemption 2:

- Loop: Travel → Random event → Choice (help/ignore/rob) → Honor change → NPC reactivity
- Vision: "Living, breathing world with consequences"
- Question: Do the consequences feel meaningful or superficial?

APPENDIX C – Further Learning

- GDC Vault (search "GDD")
- "The Art of Game Design" – Jesse Schell (lenses)
- Game Maker's Toolkit (YouTube)
- Start a 48-hour game jam this weekend and write the GDD first.

Additional Resources:

Books:

- *Blood, Sweat, and Pixels* by Jason Schreier (RDR2 development stories)
- *The Art of Game Design: A Book of Lenses* by Jesse Schell
- *Rules of Play* by Katie Salen and Eric Zimmerman

YouTube Channels:

- Game Maker's Toolkit (Mark Brown)
- AI and Games (AI system breakdowns)

- Noclip (Developer documentaries)

Online Resources:

- GDC Vault (free talks from Game Developers Conference)
- Gamasutra/Game Developer (industry articles)
- [itch.io](#) game jams (practice prototyping)

Portfolio Building:

- Start with game jams (Ludum Dare, GMTK Game Jam)
- Mod existing games (Skyrim, Fallout, Half-Life)
- Use accessible tools (Unity, Unreal, Dreams, Tabletop Simulator)
- Document your process (devlogs, design docs, video breakdowns)

Next Steps:

1. Pick a favorite game and do a full MDA analysis
 2. Identify one mechanical contradiction in a AAA game
 3. Prototype a simple core loop (1-minute playable)
 4. Join a game jam and collaborate with others
 5. Start building your portfolio with documentation
-

Workshop Materials Created From:

- GMTK video transcripts on game design fundamentals
- Industry research on modern game design (2020-2024)
- Case studies from award-winning titles
- Accessibility innovation documentation
- Red Dead Redemption 2 design analysis