ANDREY VELKOV

Lead Game Designer

19 years · 9+ shipped titles · Systems, Progression & Combat Design

DESIGN PORTFOLIO
andreypvelkov@gmail.com

ABOUT THIS PORTFOLIO

This portfolio accompanies my CV and goes deeper into the design work behind the credits. Rather than listing what was built, each entry focuses on the design problem, what I personally owned, and what made the work interesting or difficult. Projects are ordered from newest to oldest.

A note on authorship: in every project listed here, I was part of a design team. Where I describe designing or owning something that reflects my personal responsibility for that feature or system, but all of it was delivered alongside colleagues, with input from across disciplines. The entries aim to be clear about what I led, what I contributed to, and what the team delivered together.

PROJECTS IN THIS DOCUMENT

Off the Grid

Black Phoenix Games / Gunzilla Games · 2024–Present · Lead Game Designer - Meta & Progression

---

OVERVIEW

Off the Grid is a cyberpunk battle royale shooter developed by Gunzilla Games. I joined through Black Phoenix Games as Lead Game Designer focused on meta progression, and continued in the same role following a studio transition to Gunzilla. The work centred on the systems that sit above the core match loop - how players grow, what they work toward, and how live content integrates into that structure.

MY CONTRIBUTIONS

- Audited the challenges system using analytics data to identify where players were dropping off - revising challenge design to better calibrate completion effort, resulting in improved completion rates across the system.
- Designed the initial campaign system framework, defining its structure, reward cadence, and integration with the broader meta loop. The structural foundation of that design informed the current live campaign system.
- Coordinated cross-functional teams (design, art, engineering) to produce and integrate live content on cadence.

Dying Light 2: The Beast

Techland · 2023–2024 · Lead Game Designer

CONTEXT

Dying Light 2: The Beast is a standalone sequel to Dying Light 2 Stay Human, built by Techland with a focused team. I joined as Lead Game Designer to head up a team of 5–6 designers across the full production cycle, from pre-production feature definition through to ship.

The Design Problem

The Beast needed to feel like a distinct experience from DL2 - sharper, more focused - while sharing much of the same engine and systemic foundation. The design challenge was to define what was genuinely new (new tools, new progression, new encounters) and ensure those additions changed how the game felt to play, not just how it looked on a feature list. With a focused team, every design decision had to earn its place.

What I Did

- Led a team of 6 designers through pre-production and production, communicating the design direction and managing delivery.
- Contributed to scope definition alongside the game director and production team - deciding what to cut, what to build, and in what order.
- Oversaw design of new combat tools, ensuring they created genuinely new gameplay possibilities rather than reskinning existing verbs.
- Led the redesign of the progression system, working with the team to define a structure that fit the standalone format and shorter expected playtime.
- Designed and delivered multiple boss encounters – as a part of a small multidisciplinary strike team - scoping each for clarity and impact while still offering memorable design moments.
- Grew the design team from 5 to 6 during production, managing onboarding without losing momentum.

What Made It Interesting

The most interesting discipline was focus. Designing boss encounters for a game of this scale can involve sprawling iteration cycles and layers of spectacle. Here I had to design encounters that were achievable quickly, could be built with shared assets, but still felt authored and intentional. That forced a clarity about what actually makes an encounter memorable - it’s almost never the spectacle, it’s the precision of the challenge and the moment of reading the player correctly. Working within that discipline sharpened my instincts considerably.

Solution Design

Dragons Lake Entertainment · 2022–2023 · Lead Game Designer

OVERVIEW

Dragons Lake is a game development services company. My role was solution design — taking client briefs across a wide range of scopes and turning them into two deliverables: a design proposal and a full production roadmap.

MY CONTRIBUTIONS

- Researched each client's product and competitive landscape to inform the design direction — every project required a different research approach depending on the domain.
- Produced design proposals spanning simple add-ons, gamification projects, and full game concepts — covering system design, core gameplay loops, economy, and progression.
- Delivered full production roadmaps for each proposal, including timelines, milestones, team composition, and ramp-up planning.

The most valuable part of the role was the variety. Each brief demanded fresh research and a different design lens, but every one had to converge into the same two deliverables: a design that solved the client's problem, and a roadmap a team could actually execute against.

Assassin’s Creed Mirage

Ubisoft Sofia · 2022 · Lead Game Designer

OVERVIEW

AC Mirage was developed by Ubisoft Bordeaux with support from Ubisoft Sofia’s design team. My role was to lead Sofia’s contribution - but the defining constraint was that the Sofia team had just shipped AC Valhalla and Dawn of Ragnarök. The goal was to take on as much meaningful work as possible without overloading a team that needed room to recover. That made scope negotiation the central challenge: identifying what was genuinely valuable to Bordeaux, what Sofia could realistically deliver at quality, and finding the overlap between the two.

MY CONTRIBUTIONS

- Identified the highest-value areas for Sofia to contribute to in collaboration with the Bordeaux lead team, focusing on combat design improvements.
- Negotiated scope, tasks, and deliverables across studios - balancing ambition with the realistic capacity of a team coming off a major ship.
- Managed 6 game designers through the remote cross-studio collaboration, maintaining quality and alignment with Bordeaux’s direction.

Assassin's Creed Valhalla: Dawn of Ragnarök

Ubisoft Sofia · 2021–2022 · Lead Game Designer

CONTEXT

Dawn of Ragnarök is a major expansion for AC Valhalla, developed primarily at Ubisoft Sofia. I co-led a team of 7 designers across the full DLC - new bosses, new enemies, new encounters, a new weapon - from the feature list and roadmap through to ship.

The Design Problem

The DLC needed to deliver a self-contained Norse mythological experience that felt meaningfully different from the base game while remaining coherent within Valhalla's established systems. The design challenge I personally owned most directly was the new weapon: the Atgeir. By the time of Dawn of Ragnarök, Valhalla and its expansions already had a large weapon arsenal - the question wasn't just what the Atgeir would be, but why a player would choose it over everything else. It needed to offer something genuinely distinct: a different way of engaging with tactical situations, not just a different set of numbers. Additionally, mid-production, I had to absorb the responsibilities of both a departing senior designer and my co-lead, which changed the shape of my role significantly.

What I Did

- Co-led a team of 7 designers, overseeing the feature list, roadmap, and delivery of new bosses, enemies, and encounter structures across the team.
- Personally designed the Atgeir - solving the problem of differentiation through a branching combo system that gave players a tactical choice mid-sequence: AOE to handle groups, a quick interrupt to punish enemy actions, or a heavy finisher for maximum damage. The weapon rewards players who read situations carefully rather than committing to a single attack pattern.
- Worked with engineering on the Atgeir's integration into the Anvil player attack system - a deep dive into the existing architecture that, once understood, allowed for a fairly systemic implementation.
- Oversaw the team's delivery of all major boss fights, a new set of minibosses, and a new enemy faction - existing archetypes reimagined with the fire powers of Muspelheim, requiring players to adapt familiar strategies to new threat profiles.
- Managed continuity through the mid-production departure of both a senior designer and my co-lead - stopping to reassess what truly mattered, delegating clearly, and asking for help rather than absorbing everything alone.

What Made It Interesting

The Atgeir started with a creative constraint: Valhalla already had a wide arsenal, so adding another weapon for the sake of it would have been noise. The branching combo was the design answer to that constraint - a system that made the weapon feel like a different mode of play rather than a reskin. The key was ensuring each branch felt like a genuine tactical option, not just a numerical variation: AOE for crowd control, interrupt for reactive play, heavy finisher for committed damage. Players who paid attention to what the situation demanded got more out of the weapon than those who didn't. The Anvil integration required a thorough understanding of the existing player attack system before any implementation could begin - but once that groundwork was done, the actual integration followed naturally from the system's own logic. The mid-production leadership changes were a different kind of challenge. The instinct when the structure collapses is to absorb everything yourself. What I learned is that the better move is to stop, decide what actually matters most, delegate the rest clearly, and ask for help without hesitation.

Assassin’s Creed Valhalla

Ubisoft Sofia · 2020–2021 · Lead Game Designer

OVERVIEW

Before the Dawn of Ragnarök DLC, I led a Sofia design team through a DLC concept exploration phase - developing 5–6 expansion concepts through paper design and paper prototypes before production commitments were made. The phase fed directly into Valhalla’s post-launch content roadmap, with some of the concepts going on to ship.

MY CONTRIBUTIONS

- Led a design team developing 5–6 DLC concepts through paper design and paper prototypes, covering new systems, encounter structures, and mechanical directions.
- Presented and iterated on concepts with broader creative leadership, contributing to the decisions about which directions to pursue into production.
- Established the conceptual groundwork that informed several of Valhalla’s shipped post-launch expansions.

Skull & Bones

Ubisoft Sofia · 2019-2020 · Lead Game Designer

OVERVIEW

Skull & Bones was in an early development phase when Ubisoft Sofia's design team was brought in to contribute. I led a team of 4 designers across two parallel design tracks: player progression systems and ship-based boss encounters.

MY CONTRIBUTIONS

- Led a team of 4 designers, setting direction and reviewing deliverables across both workstreams.
- Produced paper design explorations for player progression systems outside of combat - how the player grew and engaged with the game world beyond ship battles.
- Built playable engine prototypes for ship-based boss encounters. The core challenge was that every boss was a ship - making multiple encounters feel distinct when they share the same basic form required deliberate differentiation. We explored three axes: different weapon loadouts, unique abilities per boss, and varying locations and reinforcement patterns to change the tactical context of each fight.

AC Origins: Curse of the Pharaohs

Ubisoft Sofia · 2018 · Senior Game Designer

CONTEXT

Curse of the Pharaohs is the second major DLC for Assassin's Creed Origins, built on the new design foundation that Origins established for the franchise. The DLC featured four resurrected pharaohs as story bosses, each with their own identity. I led the combat portion of the DLC, coordinating the designers working on each pharaoh while personally owning the Ramesses boss fight and a system that extended the pharaohs' presence into the open world as a narrative tool.

The Design Problem

The central design challenge was making four boss encounters that each felt distinct and memorable, while remaining readable and fair to the player. Each pharaoh was built around a set of design keywords - a deliberate way of keeping the encounter's identity coherent across mechanics, behavior, and feel. Ramesses was defined as the 'big guy' of the four: a physically dominant presence whose fight incorporated the sandstorm as both a visual and mechanical element. Beyond the story encounters, we also wanted the pharaohs to serve as a narrative tool in the open world - appearing, terrorizing NPCs, and making their presence felt before the player ever faced them in a boss fight.

What I Did

- Coordinated the designers working on each of the other three pharaohs - giving feedback, reviewing milestone deliveries, and ensuring consistency of quality and player readability across all four encounters.
- Personally designed and implemented the Ramesses boss fight - building his identity around physical scale and the sandstorm as a core mechanic, from concept through iteration to ship.
- Designed a system governing how the pharaohs appeared in the open world between story encounters: when and where they showed up, how they interacted with civilian and guard NPCs, and how their presence was communicated to the player - extending them from boss fights into a narrative presence across the world.
- Ensured each boss met the dual standard of being engaging and distinctive while remaining legible and fair - a player encountering a pharaoh for the first time should be able to read the threat.

What Made It Interesting

The keyword approach to boss design was one of the more useful tools on this project. When you're building four distinct bosses in parallel across four separate designers, you need a shared language for what makes each one different - not just visually, but in how they play. Keywords gave each designer a clear north star and made it easier to identify when a mechanic or behavior was pulling the encounter away from its intended identity. Ramesses was a good example: 'big guy' and 'sandstorm' as keywords immediately suggested a fight that was about scale, environmental threat, and the feeling of being overwhelmed - rather than precise, technical combat. The open-world presence layer was equally important and often underestimated. A boss that only exists inside a formal encounter feels like a menu option. Having the pharaohs roam, terrorize NPCs, and disrupt the world before the player faced them in the story made the fights feel earned rather than triggered.

Assassin’s Creed Origins

Ubisoft Sofia · 2016–2017 · Senior Game Designer

CONTEXT

AC Origins was a landmark reinvention of the Assassin's Creed franchise - a deliberate break from the previous generation that required rethinking core design approaches from the ground up. New combat, new open world, new systems. The old assumptions about how players engaged with an AC game didn't fully apply anymore, which made every design decision more interesting and more uncertain. I worked at Ubisoft Sofia on two shipped features - the desert overheat experience and the Trial of the Gods events.

The Design Problem

The desert overheat experience needed to serve the spirit of exploration rather than work against it. The design direction was toward a narrative and experiential system: the desert as a place that altered perception, not just punished carelessness. The Trial of the Gods events were live encounters for the game - mythological boss fights that needed to feel distinct from standard combat, and distinct from each other, with three bosses each built around different attack patterns and identities.

What I Did

- Led the design direction of the desert overheat experience, shaping it into a hallucination-based experiential system.
- Facilitated a wide collaborative effort to generate the actual hallucination content, with many team members contributing ideas for individual mirages across three categories: purely visual atmospheric experiences; visions guiding the player toward points of interest; and sequences connecting to the game's narrative.
- Prototyped and designed the Trial of the Gods events system, establishing the structure and rules for how the live encounters worked across three distinct bosses.
- Personally designed one of the three Trial of the Gods encounters through full implementation and iteration.

What Made It Interesting

The desert overheat experience is a design I think about often. The desert in Origins isn't just a biome, it's a mythological space - and the system needed to reflect that. Shifting to hallucinations reframed everything: the desert became somewhere that did something to the player rather than just threatened them. The three hallucination categories gave the system range - pure atmosphere, exploration reward, and narrative depth - so different players could have different relationships with it. One of the most satisfying outcomes was post-release: players started creating their own interpretations of the visions, some close to the intended meaning, some completely their own. That kind of player authorship is hard to design for deliberately - it tends to emerge when a system is evocative rather than explicit.

Assassin’s Creed Rogue

Ubisoft Sofia · 2013–2014 · Game Designer

CONTEXT

AC Rogue was developed primarily at Ubisoft Sofia and shipped alongside AC Unity - an unusual production context that meant a smaller team working at pace. My work centered on making the world feel bigger and more alive than Black Flag: expanding the crowd simulation, improving enemy AI behaviors, contributing to the new air rifle weapon, and coordinating the Present Day segment with Ubisoft Bucharest.

The Design Problem

Rogue built on Black Flag's foundation, so the challenge wasn't starting from scratch - it was pushing existing systems further. The goal for the crowd simulation was a busier, more reactive world: one where NPCs didn't just populate the streets but responded to what the player had actually done, both in the main story and in side activities. On the AI side, the challenge was identifying the most impactful improvements achievable within a limited production timeframe - the result of a series of brainstorms and workshops rather than a predefined list of fixes. The two behaviors that emerged were guards searching hiding spots and civilians raising alarms. The Present Day coordination with Bucharest added a cross-studio production layer on top of the design work.

What I Did

- Worked alongside a level designer and engineering to expand the crowd simulation - pushing for a bigger, livelier crowd than Black Flag and adding new NPC stations and encounter types.
- Designed a system that extended crowd dialogue to include progression-sensitive lines reacting to the player's story progress and side activity engagement, so the world's population reflected what the player had actually been doing.
- Designed an improved guard search behavior - when a guard lost sight of the player, they would actively move to and check nearby hiding spots, making the old strategy of diving into the nearest cover unreliable.
- Designed an alarm propagation behavior - a civilian witnessing a crime would react, drawing in nearby guards and creating a more organic and believable alert chain.
- Contributed to the design of the air rifle, a new weapon for the game.
- Coordinated the Present Day segment with Ubisoft Bucharest via regular cross-studio calls with both studios and HQ, maintaining design alignment across teams.

What Made It Interesting

The crowd simulation work on Rogue was less about individual features and more about density and ambition - pushing the simulation further than the previous game through close collaboration with level design and engineering. The progression-aware dialogue system was the most satisfying part of that. The idea was straightforward: the crowd should know who you are and what you've done. Tying the lines to story progression and side activities meant the world felt like it was keeping up with the player rather than running on a loop. The challenge was curation - deciding which actions were dramatic enough from the world's perspective to be worth a reaction, without the system becoming noise. The guard search behavior solved a problem players had exploited since the early AC games: the hide-and-wait. Once guards actively checked spots rather than giving up at the entrance, the player had to actually think about evasion rather than just find the nearest cover. The alarm propagation change worked similarly - routing it through a civilian reaction rather than a direct guard-to-guard trigger made it feel grounded in the world's logic rather than a game mechanic.

Assassin’s Creed IV: Black Flag

Ubisoft Sofia · 2012–2013 · Game Designer

CONTEXT

AC Black Flag introduced a dual-world structure - the pirate open world of the 18th-century Caribbean, and a Present Day sequence set inside the modern Abstergo Entertainment office. I contributed to the Present Day progression system, owned the Crowdlife simulation for the office environment, and designed two of the game's hacking minigames. Black Flag was also my first project working directly with Ubisoft HQ, receiving design input from the central team.

The Design Problem

The core ambition of the Black Flag Present Day was to put the player inside the game as themselves - not as a historical character, not as a named protagonist, but as the person holding the controller. First-person perspective was the primary tool for achieving that. Through narrative, the player became a mole inside Abstergo, secretly working with the Assassins, which meant the space needed to feel genuinely corporate and slightly unsafe, not just a menu screen. The progression system had to give that experience structure without breaking the immersion: mandatory story beats that moved the player forward, and optional content for those who wanted to go deeper. The hacking minigames existed to fulfil the fantasy of the role - if you're a mole inside a tech corporation, hacking should feel like something.

What I Did

- Contributed to the Present Day progression system, built around three categories of workstation: Story stations with moderate difficulty that drove mandatory narrative progression; Side stations with optional content for players who wanted more; and Challenge stations - also optional - using the hardest minigame levels and containing the most interesting pieces of content as a reward for the most engaged players.
- Designed the Crowdlife simulation for the Abstergo office - adapting the NPC behavior system from historical settings to a contemporary corporate environment, giving the space a lived-in feel while maintaining the sense of surveillance and unease.
- Designed both hacking minigames (Sphere and Wavel) from paper design through C# prototype to level implementation, built around the fantasy of operating as a covert hacker inside a corporation.
- Collaborated with Ubisoft HQ on design direction for the Present Day, incorporating their input while delivering the systems from Ubisoft Sofia.

What Made It Interesting

The 'player as themselves' concept was the most interesting design constraint I'd worked within at that point. Most game design assumes a character with defined traits, history, and abilities. Designing for a blank protagonist - someone the player was meant to project themselves onto - meant every system had to serve immersion rather than character expression. The first-person perspective was doing a lot of that work, but the Crowdlife simulation and the progression structure had to support it too. The office couldn't feel like a game hub; it had to feel like a place you'd been planted in. The three-tier workstation structure was a clean solution to a real tension: mandatory progression risks feeling like homework, fully optional content risks feeling ignored. Anchoring the most rewarding content in the Challenge tier gave players a genuine reason to engage beyond the story path without forcing them. The hacking minigames were the most direct expression of the fantasy - the brief was essentially 'make the player feel like a hacker,' which is a tone problem as much as a mechanical one.

Assassin's Creed III: Liberation

Ubisoft Sofia · 2011–2012 · Game Designer

CONTEXT

AC Liberation was Ubisoft Sofia's first lead development role on an AC title - a PS Vita game with its own protagonist, Aveline de Grandpré, set in colonial New Orleans. The central mechanic was the Outfit System: Aveline could adopt three distinct personas, each with different abilities and a different relationship with the world. I contributed to the overall game design and owned the Outfit System.

The Design Problem

The ambition was to create something genuinely new for AC - a system that hadn't been done in the franchise before. The approach was to take the full set of assassin abilities and deliberately divide them across three personas, then add unique abilities to each, so that every outfit felt like a different way of playing the game rather than a costume with minor stat differences. The world also had to respond to each persona coherently - guards and civilians needed to read Aveline's outfit and behave accordingly, which meant designing an AI reaction layer on top of the ability split.

What I Did

- Contributed to the overall game design alongside the broader team.
- Designed the Outfit System - dividing the assassin's core abilities across three personas and defining the unique tools that gave each one its own identity.
    - The Lady persona bypassed guard posts entirely - guards would not interfere with her. She could also bribe guards to temporarily disable posts or patrols. This compensated for her lack of acrobatics and very limited combat capability.
    - The Slave persona had full athletic ability but limited combat. Her unique ability was crate blending - blending in with workers across the city, causing guards to ignore her and guard posts to let her through.
    - The Assassin persona had full combat capability and full acrobatics, but stood out in the crowd and drew suspicion in civilian areas.
- Designed the AI reaction layer - how guard and civilian NPCs read outfit states and adjusted their behaviour accordingly across all three personas.
- Designed Crowdlife updates to support the outfit-aware world.

What Made It Interesting

The design principle of dividing abilities rather than adding them was what made the system work. If each outfit had the full assassin toolkit plus extras, the choice between them would be trivial. By making each persona genuinely incomplete on its own - the Lady can't climb, the Slave can't fight, the Assassin can't hide in plain sight - the player had to actually think about which tool fitted the situation. The unique abilities then gave each persona a ceiling: the Lady's bribe and the Slave's crate blend weren't just compensations for their weaknesses, they were capabilities the Assassin couldn't replicate. The AI reaction layer was the systemic glue that made it all feel real - a world that reads your outfit and responds accordingly is what turns a mechanical split into a genuine social stealth system. The edge cases in that layer were some of the most interesting design problems on the project: what happens when a guard has seen Aveline commit violence while dressed as a Lady? How does the system handle a persona switch mid-pursuit? Those state-management questions had no obvious answers and required careful iteration.

Ghost Recon Shadow Wars

Ubisoft Sofia · 2008–2010 · Game Designer

CONTEXT

Ghost Recon Shadow Wars was a turn-based tactical game for the Nintendo 3DS - Ubisoft Sofia's launch title for the platform and one of the highest-rated games at the 3DS launch. It was an early role for me with an unusually broad scope: contributing to core gameplay, building production tooling, AI design, and mission delivery.

The Design Problem

The game went through several major iterations before finding its final form. The lead designer's vision for the core tactical combat loop arrived on a set of printed pages - a paper prototype that defined how the game actually played, before the story or campaign structure existed at all. From that foundation, the team began fleshing out the design. The challenge was proving the core loop worked before investing in everything built around it, then building the production tools needed to actually make the game - a map editor, a scripting language, and the mission pipeline that would use them.

What I Did

- Updated C# prototypes in parallel with the design process, translating the evolving paper design into playable form to test the core tactical combat loop as it developed.
- Built a basic map editor in C# when the team reached the point of needing to create maps and levels - the first tooling that allowed the team to author the game's environments.
- Researched existing scripting systems - including Starcraft and several other titles - and designed a scripting language for mission logic built on the best elements of each: a straightforward, robust system of conditions and actions that designers could use without a programming background.
- Handed the map editor and prototype off to the programming team, who extended them into the full production toolset used to develop all the game's missions.
- Designed and delivered bonus PVP missions - content outside the main campaign, allowing two players to share a single 3DS.
- Identified and fixed a C++ code issue during iteration - a submission had broken the line-of-fire check. The similarity between the C++ codebase and my own C# work made it straightforward to locate and submit a fix.

What Made It Interesting

The paper prototype origin is the detail I find most interesting in retrospect. The lead arrived with the core combat loop worked out on printed pages — no story, no campaign, no levels. Just here is how this game plays. Everything else came after the team was satisfied that the core loop actually worked. That sequencing — prove the game before building around it — is something I've thought about on every project since. The C# prototyping was the most formative part of this project for me. Building playable versions of the design alongside the design process meant I was seeing the game from both sides — how it read on paper and how it actually felt in practice. That dual perspective gave me a much deeper understanding of how games are made, and it's informed how I approach design work ever since. The PVP missions were a different kind of design problem — designing for two players sharing one handheld device required thinking carefully about how turns, information, and the physical screen were divided between players.

Chessmaster: Art of Learning

Ubisoft Sofia · 2007–2008 · Game Designer

OVERVIEW

Chessmaster: Art of Learning was my first project - a mobile entry in the long-running Chessmaster franchise, built around an educational learning system. I started by researching the existing titles to understand what made them work, then moved into designing the educational minigames and the learning system that tied everything together.

MY CONTRIBUTIONS

- Designed educational minigames in collaboration with the design team, then built C# prototypes that allowed the concepts to be played and tested. Three minigames I recall: Fork the Fruit - use chess pieces to fork fruit elements on the board for points, teaching the fork tactic; Chain Reaction - connect three or more of the same chess piece type in the pattern of their attack, with longer chains scoring more points; Knight Charge - use a knight to dismantle a pawn formation, teaching knight movement and tactics.
- Designed the Art of Learning scoring system - each completed challenge (minigame or classical chess puzzle) awarded points distributed across three learning attributes, tying the minigames and puzzle content into a unified progression framework.
- Curated and structured the chess puzzle selection and difficulty progression alongside the broader team.

A MEMORABLE MOMENT

On the day I distributed one of the minigame prototypes to the team, I discovered at the end of the day that several team members had kept playing it through their working hours, competing to beat their own scores. For a first project, that was a meaningful signal.

Endyval RPG Series

Independent · Published tabletop RPG series (Bulgarian)

OVERVIEW

Endyval is a published 3-book pen & paper tabletop RPG series. My involvement grew with each book - from player and tester on the first, to system contributor on the second, to co-author and coordinator on the third.

MY CONTRIBUTIONS

- Book 1 - Played as part of the core group of friends whose regular sessions formed the testing ground for the system. Provided feedback on rules, balance, and feel from the player's perspective.
- Book 2 - Helped flesh out the combat system and contributed narrative writing for one of the fantasy races.
- Book 3 - Took on system design and coordination responsibilities. Worked on combat and progression systems - ensuring skills provided meaningful values and that the dice-rolling mechanics were as tight and clear as possible. As coordinator, gathered material from all the authors, assisted with editing where needed, and kept the system parts coherent across contributors.

The core design challenge was the same one that comes up in any progression system: making sure every choice a player makes with their character feels like it matters. Working on a rules system with no engine to hide behind - everything lives or dies on the clarity of the writing and the tightness of the numbers - was a useful early lesson in design.