Álvaro Maldonado Peña

Developer & game designer specialised in interactivity and game feel

Web: alvaromaldonado.com/en
Write to me: alvaromaldonado.com/en/mailbox
Madrid, Spain · Remote, hybrid or on-site in Madrid; open to relocation

SUMMARY

Developer and game designer specialised in interactivity and game feel. I design, code and tune mechanics, playable prototypes and real-time 3D experiences in the browser (three.js, WebGL, Godot), as well as complete web and mobile apps in TypeScript and React. I take ideas from prototype to finished product with automated testing, WCAG AA accessibility and independent review of every release.

KEY SKILLS

Game design · Game feel · Interaction design · Gameplay prototyping · Animation & motion · Real-time 3D · three.js · WebGL · Godot 4 (GDScript) · TypeScript · React · Node.js · PWA · Accessibility (WCAG AA) · Automated testing (Playwright, Vitest)

EXPERIENCE

Software developer

Software development company (name confidential) · August 2026 – present
Product development within a software team.

LANGUAGES

Spanish: Native · English: C2 · Japanese: C1 · German: C1 (CEFR levels)

ABOUT

I care about how using something feels, not just how it looks. I design the mechanic, code it and tune it until it reads without thinking: the weight of a jump, the bounce of a button, the exact moment something responds.

I work like a small studio: first I make clear what problem we're solving and for whom, then I build in pieces that fit together, and an independent review signs nothing off until it's right. That's how I take ideas from a playable prototype to a finished product.

I speak Spanish, English, Japanese and German, and I'm open to relocation: I'm happy working with teams in any country.

WHAT I CAN DO FOR YOU

I make digital things feel good in your hands

I care about how every action feels: it responds instantly, has just the right weight and never jumps. Like a good video game, even on a serious website.

- I design how a product feels, not just how it looks: instant response, anticipation, weight, inertia and settling, with no teleporting or jumps.
- Motion curves and measured juice: just enough for the action to read, no decoration, with visual and haptic feedback.

- Characters with believable behaviour: states, attention (watching the cursor), rest, curiosity and response to direct handling (grab, throw).
- Reviews by specialised game feel and animation auditors, with film and video-game criteria, for every new mechanic.
- The same on the web as in 3D, with interfaces you handle like a physical object, on phones too and respecting “reduce motion”.
- The proof is on this very home page: MANCA, my virtual assistant, lives camouflaged in it.

I design games and make them playable

From the idea to a prototype you can play: clear mechanics and rules, and characters that react naturally, tuned until they're fun and easy to read, on phones too.

- Godot 4 (GDScript): characters with activity-level behaviour and procedural animation, reviewed by animation auditors.
- Art direction from references, model viewers for sign-off before integration, and silhouette and readability reviews on small screens.

I create 3D experiences in the browser

Interactive 3D worlds that open from a link, nothing to install, with believable physics and smooth on phones too.

- three.js with BVH collisions and Rapier physics (WebAssembly): grabbing, throwing and structures you can move through.
- Multiplayer with an authoritative server over WebSocket and the same simulation on client and server.
- Performance: a fixed frame-rate target, automatic measurement and hunting down hitches and flicker.
- Carefully tuned feel: easing, weight, settling and first-person camera, with no jumps or duplicates.
- Worlds defined as data (components, catalogues and styles in YAML) and tile-based building.

I build complete web and mobile apps

From the server to your phone screen: apps that install, send notifications, work offline and sync across your devices.

- TypeScript end to end: Node (Fastify, WebSocket, native SQLite), React and Vite.
- Installable PWA with Workbox, Web Push notifications and offline support.
- OAuth integrations with Google services and real-time sync across devices.
- Infrastructure at the edge on Cloudflare (Workers, D1, Turnstile, tunnels) and private networks with Tailscale.
- Learning algorithms such as spaced repetition (FSRS), with tests.

I take ideas into production

I decide what matters, build it in pieces that fit together, and nothing is done until an independent review says so. I work like a small studio.

- Clear rules first: the language between pieces (TypeScript types plus a changelog that only adds and never breaks) is set before the work is split up.
- Parallel work, each part on its own branch (a git worktree) with a shared status log.
- Independent reviews that don't see the maker's reasoning (UX/UI, game feel, animation, art, accessibility, security, licensing), looping until no serious or medium issues remain.
- Living documentation: every decision with its recommendation, plus version and design logs.

I deliver quality: tested, accessible and secure

What I deliver is tested, anyone can use it (including with a screen reader or keyboard only) and it looks after people's data.

- Automated tests: Vitest, node:test and Playwright, including end-to-end tests with several clients.
- WCAG AA accessibility and UX/UI audits, mobile first.
- Security and privacy by design: nothing exposed that doesn't need to be, per-device permissions, layered anti-spam and GDPR.
- Third-party licences checked (fonts, music, CC BY-SA data) before publishing.
- Windows desktop automation: start with the system, self-recovery and hardware controller integration.

I think about who will use it

I don't stop at the prototype: from day one I think about who will use it and how it can stand on its own as a product.

- Product and business vision: from prototype to something you can sell.
- Priorities set with both the user and the buyer in mind.