Álvaro Maldonado Peña

Entwickler und Game Designer mit Schwerpunkt auf Interaktivität und Game Feel

Web: alvaromaldonado.com/de
Schreib mir: alvaromaldonado.com/de/mailbox
Madrid, Spanien · Remote, hybrid oder vor Ort in Madrid; offen für einen Umzug

PROFIL

Entwickler und Game Designer mit Schwerpunkt auf Interaktivität und Game Feel. Ich entwerfe, programmiere und tune Spielmechaniken, spielbare Prototypen und Echtzeit-3D im Browser (three.js, WebGL, Godot) sowie komplette Web- und Mobile-Apps mit TypeScript und React. Ideen bringe ich vom Prototyp zum fertigen Produkt – mit automatisierten Tests, Barrierefreiheit nach WCAG AA und unabhängiger Prüfung jeder Version.

KERNKOMPETENZEN

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)

BERUFSERFAHRUNG

Softwareentwickler

Softwareentwicklungsunternehmen (Name vertraulich) · August 2026 – heute
Produktentwicklung in einem Softwareteam.

SPRACHEN

Spanisch: Muttersprache · Englisch: C2 · Japanisch: C1 · Deutsch: C1 (Niveaus nach dem GER)

ÜBER MICH

Mich interessiert, wie sich etwas anfühlt, nicht nur, wie es aussieht. Ich entwerfe die Mechanik, programmiere sie und stimme sie ab, bis man sie ohne Nachdenken versteht: das Gewicht eines Sprungs, das Federn eines Buttons, den genauen Moment, in dem etwas reagiert.

Ich arbeite wie ein kleines Studio: Zuerst kläre ich, welches Problem wir für wen lösen, dann baue ich in Teilen, die zusammenpassen, und eine unabhängige Prüfung nimmt nichts ab, bevor es stimmt. So bringe ich Ideen vom spielbaren Prototyp zum fertigen Produkt.

Ich spreche Spanisch, Englisch, Japanisch und Deutsch und bin offen für einen Umzug: Ich arbeite gern mit Teams in jedem Land.

WAS ICH FÜR DICH TUN KANN

Ich sorge dafür, dass sich Digitales gut anfühlt

Mir ist wichtig, wie sich jede Aktion anfühlt: Sie reagiert sofort, hat das richtige Gewicht und springt nie. Wie in einem guten Spiel, auch auf einer seriösen Website.

- Ich gestalte, wie sich ein Produkt anfühlt, nicht nur, wie es aussieht: sofortige Reaktion, Antizipation, Gewicht, Trägheit und Ausschwingen, ohne Teleportieren oder Sprünge.
- Bewegungskurven und dosierter juice: gerade so viel, dass die Aktion verständlich wird, kein Schnickschnack, mit visuellem und haptischem Feedback.
- Figuren mit glaubwürdigem Verhalten: Zustände, Aufmerksamkeit (dem Cursor nachschauen), Ruhe, Neugier und Reaktion auf direkte Manipulation (greifen, werfen).
- Prüfung jeder neuen Mechanik durch spezialisierte Prüfer für game feel und Animation, mit Maßstäben aus Film und Videospiel.
- Im Web genauso wie in 3D, mit Oberflächen, die sich wie ein physisches Objekt bedienen lassen, auch auf dem Handy und mit Rücksicht auf „Bewegung reduzieren“.
- Der Beweis steckt in genau dieser Startseite: MANCA, meine virtuelle Assistentin, lebt gut getarnt darin.

Ich entwerfe Spiele und mache sie spielbar

Von der Idee zum spielbaren Prototyp: klare Mechaniken und Regeln, Figuren, die natürlich reagieren, so lange abgestimmt, bis es Spaß macht und verständlich ist, auch auf dem Handy.

- Godot 4 (GDScript): Figuren mit Verhalten nach Aktivitätsstufen und prozeduraler Animation, geprüft von Animations-Prüfern.
- Art Direction mit Referenzen, Modell-Viewer zur Freigabe vor der Integration und Prüfung von Silhouette und Lesbarkeit auf kleinen Bildschirmen.

Ich mache 3D-Erlebnisse im Browser

Interaktive 3D-Welten, die sich per Link öffnen, ohne Installation, mit glaubwürdiger Physik und flüssig auch auf dem Handy.

- three.js mit BVH für Kollisionen und Physik mit Rapier (WebAssembly): greifen, werfen und Strukturen, auf denen man sich bewegen kann.
- Multiplayer mit autoritativem Server über WebSocket und derselben Simulation auf Client und Server.
- Performance: feste Ziel-Framerate, automatische Messung und Jagd auf Ruckler und Flackern.
- Sorgfältig abgestimmtes Steuerungsgefühl: Kurven, Gewicht, Ausschwingen und Ego-Kamera, ohne Sprünge oder Doppelungen.
- Welten als Daten definiert (Komponenten, Kataloge und Stile in YAML) und Bauen im Raster.

Ich entwickle komplette Web- und Mobile-Apps

Vom Server bis zum Handydisplay: Apps, die sich installieren lassen, benachrichtigen, offline funktionieren und sich zwischen deinen Geräten synchronisieren.

- TypeScript durchgängig: Node (Fastify, WebSocket, natives SQLite), React und Vite.
- Installierbare PWA mit Workbox, Web-Push-Benachrichtigungen und Offline-Betrieb.
- OAuth-Integrationen mit Google-Diensten und Echtzeit-Synchronisierung zwischen Geräten.
- Infrastruktur am Edge mit Cloudflare (Workers, D1, Turnstile, Tunnel) und privaten Netzwerken mit Tailscale.
- Lernalgorithmen wie verteilte Wiederholung (FSRS), mit Tests.

Ich bringe Ideen in Produktion

Ich entscheide, was zählt, baue es in Teilen, die zusammenpassen, und nichts gilt als fertig, bis eine unabhängige Prüfung es bestätigt. Ich arbeite wie ein kleines Studio.

- Zuerst klare Regeln: Die gemeinsame Sprache zwischen den Teilen (TypeScript-Typen und ein Änderungsprotokoll, das nur ergänzt und nie bricht) steht fest, bevor die Arbeit verteilt wird.
- Parallele Arbeit, jeder Teil in einem eigenen Branch (ein Git-worktree) mit gemeinsamem Statusprotokoll.
- Unabhängige Prüfungen ohne Einblick in die Überlegungen der Entwickler (UX/UI, Game Feel, Animation, Grafik, Barrierefreiheit, Sicherheit, Lizenzen), in Runden, bis keine schweren oder mittleren Fehler mehr übrig sind.
- Lebendige Dokumentation: jede Entscheidung samt Empfehlung, dazu Versions- und Designprotokolle.

Ich liefere Qualität: getestet, barrierefrei und sicher

Was ich liefere, ist getestet, für alle nutzbar (auch mit Screenreader oder nur per Tastatur) und geht sorgsam mit den Daten der Menschen um.

- Automatische Tests: Vitest, node:test und Playwright, auch End-to-End mit mehreren Clients.
- Barrierefreiheit nach WCAG AA und UX/UI-Audits, Mobile First.
- Sicherheit und Datenschutz von Anfang an: nichts offen, was nicht nötig ist, Berechtigungen pro Gerät, mehrschichtiger Spamschutz und DSGVO.
- Fremdlizenzen vor der Veröffentlichung geprüft (Schriften, Musik, Daten unter CC BY-SA).
- Desktop-Automatisierung unter Windows: Start mit dem System, Selbstwiederherstellung und Anbindung physischer Controller.

Ich denke an die Menschen, die es nutzen

Ich höre nicht beim Prototyp auf: Von Anfang an überlege ich, wer es nutzt und wie es sich als Produkt tragen kann.

- Produkt- und Geschäftssicht: vom Prototyp zu etwas, das sich verkaufen lässt.
- Prioritäten mit Blick darauf, wer es nutzt und wer dafür zahlt.