

Elias Hogstvedt

Developer

Bac Ninh, Vietnam

+47 9479 3143

✉ elias@mewb.no

in www.linkedin.com/in/elias-hogstvedt-4b1a4a156

🔑 CapsAdmin



Summary

Developer with over 15 years of experience spanning systems programming, frontend and backend, with a specialization in TypeScript, React, React Native and C#. Known for translating complex designs into polished, pixel-perfect interfaces, working closely with designers and stakeholders, asking the right questions, and staying engaged until the original vision is fully realized. A strong open source background, including game engines, type system for Lua, and LLM inference from scratch. Brings low-level technical depth that surfaces in professional work when it matters most. Comfortable across the full stack, with a track record of delivering across both product teams and consultancy environments.

Experience

2021–present **MEWB**, *Developer*, Oslo (remote)

Fullstack developer working across greenfield client projects in domains ranging from maritime automation to social media and indoor farming. Typically the sole developer on a project, taking full ownership from initial build through to delivery while maintaining direct communication with clients. Work spans frontend and backend, with a focus on producing polished, well-crafted interfaces.

2019–2021 **Posten/Bring**, *Developer*, Oslo

Worked within Posten's digital innovation department, a team primarily comprised of UX designers and business developers. I was one of the few developers, and at times the only, developer. Built several products from the ground up while also producing demos, prototypes, internal tooling and data visualizations through workshops and rapid iteration. Worked in close collaboration with designers and non-technical stakeholders throughout.

2017–2019 **Tøtsj**, *Developer*, Oslo

Developer at a small Oslo-based agency (now MEWB), taking senior ownership across a varied range of client projects. Work spanned different technologies and problem domains depending on the client, from mobile apps to interactive experiences. Responsible for significant parts of each project end to end.

Selected Project Deliveries

2022–2022 Høglund HMI Web, MEWB - Høglund

Responsibility: Høglund delivers integrated automation for the maritime industry. As part of a team, developed and continuously evolved a web-based HMI using C# and Blazor, enabling real-time remote monitoring and control of ship systems from a browser, replicating and extending the functionality of their existing native HMI written in Pascal.

Work spanned the full depth of the application: building a playback system with timeline scrubbing and time-travel through historical ship data, an alarm and event history panel with real-time streaming updates, a signal and data layer using Redis, and a range of custom UI components including datagrids with virtualization, and write-capable dialogs. Also contributed to infrastructure and deployment on AWS, including Docker configuration.

Result: Running in production on real ships.

Technology: C#, Blazor, Redis, AWS, Docker

2022–2022 Rift Labs / Photosynthetic, MEWB - Rift Labs

Responsibility: Stepped in near the end of development for Photosynthetic, an HMI for controlling grow environments in indoor farming. Took ownership of the remaining scope: resolved the outstanding bugs, completed the unimplemented functionality, and brought the project to a shippable state. The frontend communicates with a local hardware controller over HTTP/REST, and gives growers real-time and time-lapse views of plant growth alongside controls for tuning light spectrum, CO₂, and oxygen levels per growth stage.

Result: Prototype completed and delivered.

Technology: Typescript, React

2022–2022 Cliq, MEWB - Amara Thompson

Responsibility: Sole developer on a React Native prototype for collaborative short-form video creation. Users join private groups, capture or upload 15-second clips, and assemble a shared group movie on-device using a combination of recording and editing libraries, with finished videos uploaded to AWS. Also covered profile and account flows, phone-based authentication, thumbnail generation, and a Firebase-backed data layer with a schema-validated abstraction on top. A strong emphasis throughout on feel: smooth gestures, snappy transitions, and polished interactions were a core part of the project, working closely with designers to get the details right.

Result: Finished working prototype delivered; put on hold by the client.

Technology: React Native, TypeScript, Firebase, AWS

2021–2021 Truss, MEWB - Russ

Responsibility: Built a platform for fans to book live one-on-one video calls with artists, initially centred around rap artist Russ. Covered the full product: a React Native iOS app for artists to manage their availability by enabling time slots, a React web app for fans to browse open slots and join calls, and a management dashboard. The fan-side flow included a pre-call lobby for testing camera and audio before the session started; fans received an SMS link ahead of time and could join when ready, with the countdown timer starting once the artist joined. Video calls were handled via Twilio. Started on the frontend and gradually took over the backend as well, with a MEWB colleague handling the initial backend groundwork. Payments were handled via Stripe.

Result: Tested in production with artist Russ; put on hold by the client.

Technology: React Native, React, TypeScript, Twilio, Stripe

2020–2021 **Bring CO2 Emissions Estimator**, *Posten/Bring*

Responsibility: Built a CO2 and emissions estimation system for Bring's large business customers such as H&M, integrated into Bring's main customer portal. Built a node backend with API that ingested delivery event data automatically sourced from legacy internal systems in CSV-like formats, normalised and aggregated it into a Postgres database, then served it to the Java/Spring customer portal backend. Due to the volume of data, hand-rolled binary protocols were used in places for performance. Also built the frontend dashboard within the existing customer portal. The emissions algorithm was defined by logistics domain experts; worked closely with them throughout to ensure the calculations were correctly implemented.

Result: Live in production as part of Bring's customer portal.

Technology: Node.js, Typescript, Java, Spring, React, PostgreSQL

2019–2022 **Last Mile Delivery App**, *Posten/Bring*

Responsibility: Sole full-stack developer on a PWA for last-mile delivery drivers, adding gamification to the delivery experience. Drivers could opt into a leaderboard and track personal stats including total weight carried, packages delivered per hour, and streaks. Research was deeply embedded: rode alongside drivers, visited warehouses, and ran interviews throughout. The game designer on the project was himself a driver of ten years and knew the driver community well, which gave the product an unusually grounded perspective.

Result: Delivered and used by drivers in production.

Technology: React, TypeScript

2017–2018 **Smakeappen**, *Tøtsj - Kapital Vin / Jon Trygve Hegnar*

Responsibility: Smakeappen is a Norwegian wine discovery app, a Vivino-style product built specifically for the Norwegian market and Vinmonopolet's catalogue. It is now known as Vinify. It included a barcode scanner for looking up wines in Vinmonopolet's catalogue, ratings, user made "playlists" of wines among other things. Sole frontend developer, building the full app in Fuse while a separate team handled the backend. Fuse was Tøtsj's in-house specialit. The agency was founded by former Fuse developers, and was in many ways a precursor to React Native, offering a similar declarative mobile model before React Native had gained traction.

Result: Delivered. <https://www.smakeappen.no/>

Technology: TypeScript, Fuse

2017–2018 **Telia Soundmaker**, *Tøtsj - Telia / Kygo*

Responsibility: Telia Soundmaker was a music-creation app, a collaboration between Telia and Kygo, a creative/experimental experience designed to let anyone make music easily. Given the app's creative nature, the design was animation-heavy. I implemented the complex UI animations specified in the original designs, as well as a real-time sound effects API that let loops and melodies have effects applied as part of the composition experience.

Result: Delivered.

Technology: C#, Unity

Recent Hobby Projects

NattLua (2019-present) A typed dialect of Lua, similar in spirit to TypeScript. Adds a static type system to Lua while remaining compatible with the ecosystem. <https://github.com/CapsAdmin/NattLua/>

LuaJIT LLama3 (2024)	Inference for the Llama3 model implemented in scratch in LuaJIT. Supports a subset of the gguf format and runs on either Cuda or CPU threads. https://github.com/CapsAdmin/luajit-llama3/
Game Engine/Framework (2015-present)	Game engine written entirely in LuaJIT with the goal of only having Vulkan as a direct dependency while the rest, such as window management, font loading, texture loading, etc is implemented entirely in LuaJIT. Has a very complete 2d rendering system with the ability to run love2d games through a compatibility layer like Wine. Also has a full UI system with complex theming support.
PAC3 (Lua) (2010-2020)	Allows players to create and wear custom outfits, accessories, animations, and visual effects on their in-game character model. One of the most widely used mods in the Garry's Mod ecosystem. https://github.com/CapsAdmin/pac3/ I handed the torch for this project to another maintainer in 2020, but I still contribute occasionally.

Skills

Languages	Typescript, C#, C, C++, Java, LuaJIT, Python
Frameworks	React, React Native, Node/Bun/Deno, Unity
Infrastructure	Docker, Linux, Firebase
Other	NixOS/Linux, OpenGL, Vulkan, GLSL, various agent harnesses, PyTorch, setting up and using local LLMs