

# Patent Abstract and Claims Summary

**Patent Title:** Method and system for providing a website and a standalone application on a client device using a single code base

**Assignee:** Easywebapp Inc.

**Jurisdiction:** United States Patent and Trademark Office (USPTO)

**Status:** Granted (Filed 2021, Granted 2022)

**Patent No.:** US11526577B2

## What the Patent Covers

This invention enables a **single Progressive Web App (PWA)** to serve two **distinct user experiences (UXs)** based on how the app is launched:

- **Web UX:** If the PWA is accessed via a browser or direct link, the app presents a traditional web interface with browser UI (address bar, tabs).
- **App UX:** If the PWA is launched as an installed application (from home screen, taskbar, or OS launcher), it delivers a fullscreen, gesture-native UX akin to native apps.

This **dynamic, context-aware UX switching** happens at runtime from a single codebase using environmental cues such as `window.matchMedia('(display-mode: standalone)'),` install state, and service worker context.

## What Rights Are Protected

- **Method Claims:** The detection and automatic rendering of UX modes depending on install context.
- **App Behavior:** Rights to deliver and manage two UXs from the same PWA instance without requiring separate builds or routes.
- **UI/UX Duality:** The system-level response and treatment of the app across browser, OS, and install modalities.

This covers any implementation where **UX switching is tied to how the PWA is opened**, regardless of framework or platform (e.g., Chrome, Edge, Safari).

## Why It's Novel and Non-Obvious

Before this invention:

- PWAs could detect install context, but did not meaningfully **change UX, layout, or feature availability**.
- Developers often maintained **separate builds or layouts** for app vs. browser access, creating overhead and inconsistent UX.

This patent formalizes a **lightweight and scalable method** to deliver two optimized experiences **from a single deployed app**, reducing complexity and enhancing the installable app value proposition.

It introduces a programmable **dual-mode UX layer**, which adapts dynamically to:

- Browser-initiated sessions (navigational UX)
- OS-integrated sessions (immersive UX)

The innovation lies in how **context triggers dynamic layout and behavior switching**, enabling better platform integration, discoverability, and engagement without fragmenting the codebase.