

# PWA Banner Fix - Complete Removal from All Pages

---

## Issue Summary

---

PWA install banner was still showing on authenticated pages (Login, Dashboard, Marketplace, User Management) even though it was removed from public pages (homepage).

## Root Cause Analysis

---

### Previous Fix (Incomplete)

- The install banner in `PWAManager.tsx` was commented out (lines 453-467)
- This fixed the banner on public pages

### Why Banner Still Showed on Authenticated Pages

1. **Separate Component:** The `PWAInstallButton` component in `src/components/pwa/PWAInstallButton.tsx` was still active
2. **DashboardLayout Integration:** `DashboardLayout.tsx` (used for authenticated pages) imported and rendered `<PWAInstallButton>` on line 822
3. **State Management:** `PWAInstallButton` gets its `isInstallable` state from the `PWAContext` provided by `PWAManager`
4. **Event Listener:** The `beforeinstallprompt` browser event was still being listened to and setting `isInstallable = true`, even when PWA was disabled

## The Fix

---

### 1. Updated `PWAManager.tsx`

Added environment variable check in the `beforeinstallprompt` event handler:

```
// Effect to handle beforeinstallprompt event for install button
useEffect(() => {
  // Check if PWA is enabled via environment variable
  const pwaEnabled = (process.env['NEXT_PUBLIC_PWA_ENABLED'] || '').toString() ===
    '1';

  // Don't handle install prompts if PWA is disabled
  if (!pwaEnabled) {
    console.info('[PWAManager] PWA disabled - install prompts suppressed');
    return;
  }

  const handleBeforeInstallPrompt = (e: Event) => {
    // ... rest of the code
    setIsInstallable(true);
  };

  window.addEventListener('beforeinstallprompt', handleBeforeInstallPrompt);
  // ...
}, []);
```

**Impact:**

- When `NEXT_PUBLIC_PWA_ENABLED` is not set to '1', the `beforeinstallprompt` event listener is never registered
- This prevents `isInstallable` from ever becoming `true`
- As a result, `PWAInstallButton` will not render (it returns `null` when not installable)

## 2. Updated .env File

Added explicit documentation:

```
# PWA Configuration (Progressive Web App)
# Set to '1' to enable PWA features (install prompts, offline support, service work-
ers)
# Currently DISABLED to avoid UI clutter and focus on core functionality
# NEXT_PUBLIC_PWA_ENABLED=0
```

## How It Works

### PWA Control Flow

1. **Service Worker Registration** (PWAManager.tsx lines 97-162):
  - Already had kill-switch: when `NEXT_PUBLIC_PWA_ENABLED !== '1'`, it unregisters all service workers
2. **Install Prompt Event** (PWAManager.tsx lines 206-230):
  - **NEW:** Now also checks `NEXT_PUBLIC_PWA_ENABLED` before registering event listener
  - If disabled, `isInstallable` never becomes `true`
3. **PWAInstallButton Rendering** (PWAInstallButton.tsx lines 67-125):
  - Component checks `isInstallable` from PWA context
  - Returns `null` if not installable (line 125)
  - Since we never set `isInstallable = true`, the button never renders


## Files Modified

---

1. `src/components/pwa/PWAManager.tsx` - Added environment check to suppress install prompts
2. `.env` - Added documentation for PWA configuration

## Testing Checklist

---

- [x] Banner removed from public pages (homepage)  (already fixed)
- [ ] Banner removed from authenticated pages:
- [ ] Login page
- [ ] Dashboard
- [ ] Marketplace
- [ ] User Management
- [ ] Admin pages

## Verification Steps

---

1. Check that `NEXT_PUBLIC_PWA_ENABLED` is not set or not set to '1' in environment variables
2. Visit authenticated pages and confirm no install button/banner appears
3. Check browser console for log: `[PWAManager] PWA disabled - install prompts suppressed`
4. Verify that service workers are not registered (check Application tab in DevTools)

## Deployment Notes

---

- No environment variables need to be changed on Render
- Default behavior is PWA disabled (secure by default)
- To enable PWA in future, set `NEXT_PUBLIC_PWA_ENABLED=1` in environment

## Related Components

---

- `src/components/pwa/PWAManager.tsx` - Main PWA provider with context
- `src/components/pwa/PWAInstallButton.tsx` - Install button component
- `src/components/pwa/InstallPrompt.tsx` - Banner/modal prompt component (not used in layouts)
- `src/components/layout/DashboardLayout.tsx` - Uses `PWAInstallButton` (now suppressed)
- `src/app/layout.tsx` - Root layout that wraps app with `PWAManager`

## Summary

---

The fix ensures that when PWA is disabled (default state), the install prompt event is never listened to, which prevents any PWA install UI from appearing anywhere in the app, including authenticated pages.