

Webstudio Customization & Hosting – Execution Brief

1) Initial Setup & Understanding

- Clone and run Webstudio locally.
- Understand its architecture, editor flow, authentication system, project storage structure, and publishing mechanism.
- Identify how projects are stored and how user access is controlled.

The goal at this stage is full familiarity before making modifications.

2) Branding & UI Customization

We need to completely align the builder UI with Cambliss branding.

Required Changes:

- Replace all Webstudio branding with Cambliss branding.
- Update logo, favicon, page titles, and meta tags.
- Apply Cambliss brand color palette across:
 - Buttons
 - Sidebar
 - Active states
 - Hover states
 - Background colors
- Ensure UI consistency with our main Cambliss dashboard.
- Remove any demo or open-source promotional references from UI (subject to license compliance).

The builder should look like a native Cambliss product – not a third-party tool.

3) Authentication Strategy

Preferred approach:

- Only logged-in Cambliss users should access the builder.
- Integrate it with our existing authentication system (SSO/JWT-based access).
- Subscription should be activated, If a user's subscription is inactive, access should be restricted.

4) Multi-Tenant Consideration

Since Cambliss is a SaaS platform:

- Each client must have isolated projects.
- No cross-client project visibility.
- Project ownership must be tied to tenant ID.
- Data isolation must be strictly enforced.

Even if Webstudio does not natively support this, we must architect tenant-level separation before production deployment.

5) Deployment Plan

Once branding and authentication are complete:

- Deploy on VPS under subdomain

- Configure secure HTTPS.
- Ensure reverse proxy routing works correctly.
- Ensure builder does not expose internal APIs publicly.
- Production environment must be stable and optimized.

6) Security & Stability

Before going live, team must ensure:

- HTTPS enforced
- Proper session validation
- No development logs in production
- Error handling implemented
- File uploads secured
- Environment variables protected

7) Scalability Preparation

Architecture must be ready for:

- Custom domain mapping per client
- Publishing websites to client domains
- Storage for media assets
- CDN integration
- Subscription-based builder access

This should not be treated as a static tool, but as a scalable SaaS component.

8) Legal & License Check

Before removing branding or modifying UI:

- Confirm Webstudio license terms.
- Ensure compliance with open-source obligations.
- Document any required attribution.

9) Expected Outcome

At the end of this phase:

- A fully branded Cambliss Builder.
- Hosted under cambliss domain
- Accessible only to authorized Cambliss users.
- Architected for multi-tenant SaaS expansion.

link :

④ GitHub - webstudio-is/webstudio: Open source website builder and Webflow alternative. Webstudio is an ad...