

Überarbeitetes Schema mit internationalen Kürzeln

```

<!DOCTYPE html>
<html lang="de">
<head>
  -> HEAD_META_LINK_SCRIPT
  -> STYLE
</head>
<body>
  -> NAVBAR
  -> SIDEBAR
  -> CONTENT

  <script>
    -> GLOBALS
    -> F_OPENPROGRAM
    -> F_SHOWOPTIONS
    -> F_BRINGTOFRONT
    -> F_CLOSEPROGRAM
    -> F_REMOVEFROMOPENPROGRAMS
    -> F_RENDEROPENPROGRAMS
    -> SETINTERVAL

    -> EV_DL_Sidebar-Links

        -> EV_KH_SIDEBAR

    -> EV_ENDE

    -> HF_SHOWCONTENTBYID

    -> HF_GETVISIBLESIDEBARLINKS

    /* ===== Navigation: Aktives Sidebar-Item setzen ===== */
    document.addEventListener('DOMContentLoaded', () => {
      -> HF_UPDATEACTIVEITEM
    });

    -> EV_SIDEBAR

  </script>
</body>
</html>

```

Änderungen und Verbesserungen

1. Internationale Kürzel:

- **GL_**: Global variables or configurations.

- **FN_**: Functions that implement specific logic (e.g., FN_OPENPROGRAM).
- **CL_**: Click handlers, replacing the previous EV for events.
- **BL_**: Logical blocks, grouping related functionality together.

2. Nummerierte Blöcke (Optional):

- Falls du noch mehr Struktur möchtest, kannst du Blöcke wie folgt nummerieren:

```
-> BL_01_SIDEBAR_LINKS /* Sidebar link handling */
-> BL_02_NAVIGATION /* Active navigation logic */
```

-

3. Kommentare prägnant und englisch:

- Die Kommentare sind kurz, aber beschreibend, und auf Englisch gehalten, um internationale Zusammenarbeit zu erleichtern.

4. Ende von Blöcken markieren:

- Du verwendest jetzt klare **Ende-Markierungen** wie -> BL_SIDEBAR_LINKS_END, was die Lesbarkeit erhöht.

Was macht diesen Ansatz so effektiv?

1. Klarheit durch Präfixe:

- Jedes Kürzel hat eine klare Bedeutung, z.B. CL_ für Clickhandler, FN_ für Funktionen und BL_ für Blöcke.

2. Einheitliche Kommunikation:

- Im Büro kannst du jetzt leicht fragen: „Hast du den CL in BL_SIDEBAR_LINKS fertig?“ oder „Benötigst du eine FN für BL_NAVIGATION?“

3. Flexibilität für Änderungen:

- Neue Blöcke oder Logik können leicht hinzugefügt werden, ohne die bestehende Struktur zu stören.

Weiterer Vorschlag: Dokumentation mit Kürzeln

Um die Kürzel auch für neue Teammitglieder verständlich zu machen, könntest du eine kurze **Referenzliste** erstellen:

Harbourino Kürzel

| Kürzel | Bedeutung | Beispiel |
|--------|---|-------------------|
| GL_ | Global variables/configurations | GL_GLOBALS |
| FN_ | Function for a specific task | FN_OPENPROGRAM |
| CL_ | Click handler | CL_SIDEBAR |
| BL_ | Block of related logic or functionality | BL_SIDEBAR_LINKS |
| EV_ | Event-related handler (optional) | EV_SIDEBAR_TOGGLE |

Fazit

Mit diesen Anpassungen und der konsequenten Nutzung der Kürzel kannst du nicht nur deinen Code, sondern auch die Zusammenarbeit im Team optimieren. Die Struktur bleibt übersichtlich, und die Kommunikation wird präzise und effizient.