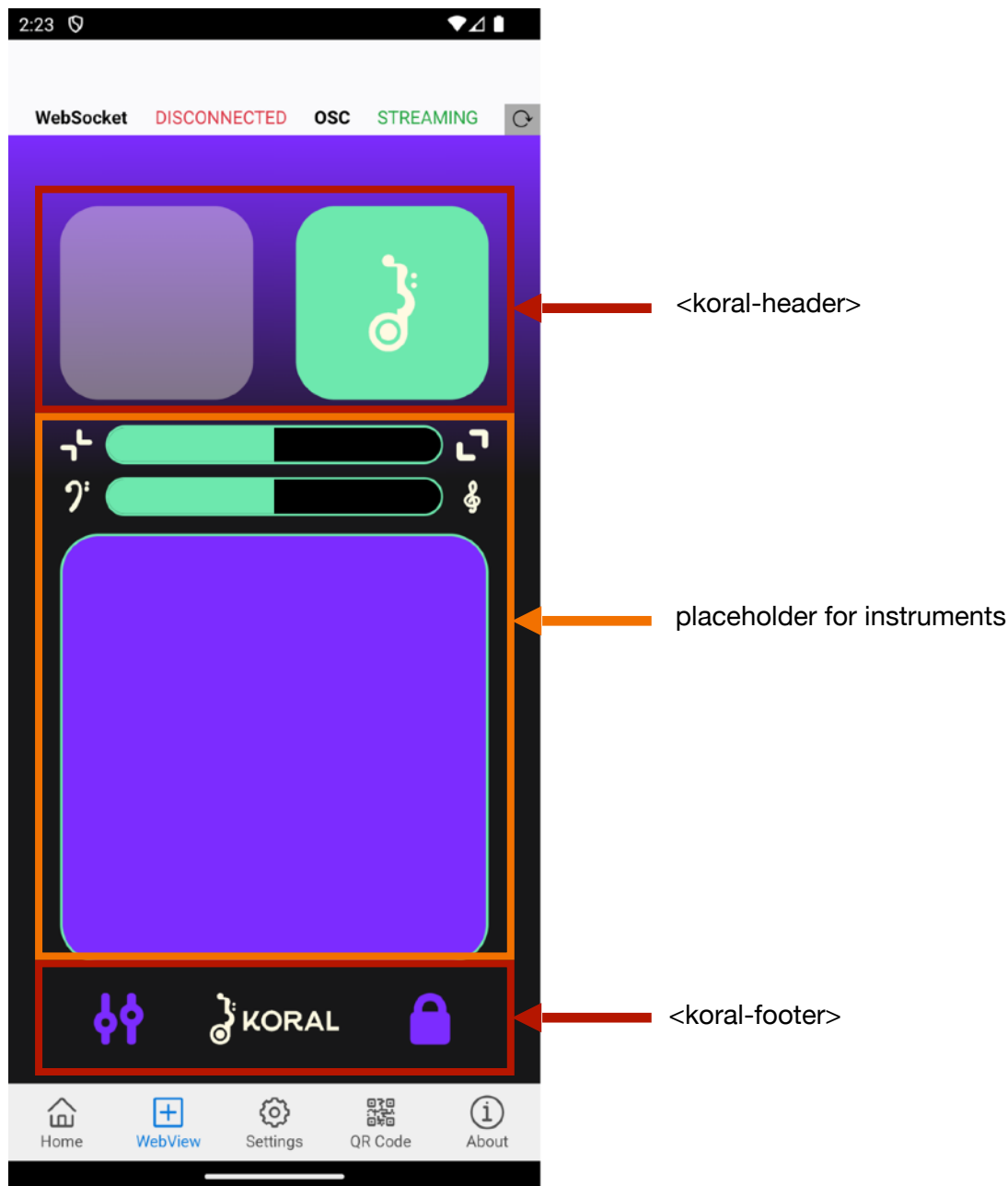


CoMote-Koral, WebViews

Toutes les sources sont dans le dépôt <https://github.com/ircam-ismm/comote-koral/>, dans le dossier `koral-server-src`. Les changements sont à faire dans ce dossier de sources et pas dans le dossier de build (koral-server)

Layout



La structure globale de la page est définie dans `src/clients/instruments/index.js` lignes 111 à 138. Les instruments sont importés et définis au début du même fichier.

Tous les composants sont définis comme WebComponent standards, cf. https://developer.mozilla.org/en-US/docs/Web/API/Web_components

Pour simplifier le développement, ils sont implémentés avec la bibliothèque <https://lit.dev/> développée par Google.

Variables de style globales

`src/clients/styles/app.scss` en haut du fichier, eg.

```
:root {  
  --koral-font-family: 'Bricolage Grotesque';  
  
  --koral-gray-light: #d5d5d5;  
  --koral-gray-normal: #929292;  
  --koral-color-1: #6DE8AE;  
  --koral-color-1-dimmed: #254f3c;  
  --koral-color-2: #FF9C66;  
  // ...  
}
```

Components communs

<koral-header>
- `src/clients/instruments/views/shared/koral-header.js`

<koral-footer>
- `src/clients/instruments/views/shared/koral-footer.js`

Réglages

La page de réglages est définie dans le fichier `src/clients/instruments/views/shared/koral-settings.js`.

Le composant <koral-settings> est affiché ou caché en fonction de l'état d'une variable dans le fichier `src/clients/instruments/index.js`

Instruments

Les instruments sont injectés dynamiquement entre le header et le footer en fonction du nom de l'instrument donné.

Les fichiers des instruments sont situés dans `src/clients/instruments/views/`

Exemple de `koral-kgesture.js`

```
import { html, LitElement } from 'lit';
import { KoralInstrument } from './shared/KoralInstrument.js';

import '@ircam/sc-components/sc-button.js';

b-ma, il y a 5 mois | 1 author (b-ma)
class KoralKgesture extends KoralInstrument {
  static properties = {
    instrument: { type: String },
    freeze: { type: Boolean },
  };

  render() {
    if (!this._state) {
      return nothing;
    }

    return html`
      <div class="screen column" style="justify-content:center">
        <div class="column buttons">
          <sc-button
            class="zone set "
            style="background-image: url(images/gesture.png)"
            @press=${e => sendEvent(`${this.instrument}/gesture/set`, 1)}
            @release=${e => sendEvent(`${this.instrument}/gesture/set`, 0)}
          ></sc-button>
          <sc-button
            class="zone reset "
            style="background-image: url(images/undo.png)"
            @input=${e => sendEvent(`${this.instrument}/gesture/reset`, 1)}
          ></sc-button>
        </div>
      </div>
    `;
  }
}

if (customElements.get('koral-kgesture') === undefined) {
  customElements.define('koral-kgesture', KoralKgesture);
}
```

Tous les changements de structure HTML se font à l'intérieur du `return html` (cf. carré rouge).

Les vues utilisent des composants génériques de la bibliothèque [sc-components](https://ircam-ismm.github.io/sc-components/) <https://ircam-ismm.github.io/sc-components/> : les éléments HTML correspondant commencent par `<sc-*`. (cf. carré vert)

Les styles appliqués au composant sont définis dans des classes CSS (cf. carré orange). Les classes à appliquer sont définies dans le fichier CSS `src/clients/styles/app.scss`.

Modifications

Dans le dossier `koral-server-src`, il faut d'abord installer les dépendances de node :

```
cd koral-server-src  
npm install
```

Il est ensuite possible de faire des modifications. Il faut alors faire un nouveau build pour max

```
./max-build.bash
```

Il est possible de lancer un environnement de travail pour effectuer un build à chaque changement. Il suffit de recharger la WebView pour voir les modifications. Tout d'abord, il faut arrêter les autres serveurs de WebView (notamment dans Max ou dans l'application autonome). Puis, lancer la commande :

```
npm run dev
```

Lorsque les changements sont validés, ne pas oublier de faire un dernier build pour Max, puis de faire une version sur git (commit et push).