

SvelteKitでつくるゲーム音楽プレイヤー

Svelte(スベルト)とは

- フロントエンドフレームワークの一種
 - 3大フレームワーク(Angular/React/Vue)よりも比較的新しめ
 - 使用事例 → 大統領選挙の特設サイトなど
- Svelteはコンパイラ
 - `.svelte` ファイルからJavaScriptに変換
 - 仮想DOMを使わず、DOMを更新するための命令型コードを生成・実行

構文 (ステート)

`$state()` を使う

```
<script lang="ts">  
  let keyword = $state("");  
</script>  
  
<input type="text" bind:value={keyword} />
```

構文 (ステート)

Reactの書き方

```
const [keyword, setKeyword] = useState("");

const handleChange = (event) => {
  setKeyword(event.currentTarget.value);
}

return (
  <input type="text" value={keyword} onChange={handleChange} />
)
```

構文 (ループ)

eachブロックを使ってループを表現する

```
<script lang="ts">
  const users = [ { id: "12345", name: "TEST_USER" } ]
</script>

<div>
  {#each users as { id, name }}
    <p>{id}: {name}</p>
  {/each}
</div>
```

構文 (条件分岐)

if-elseブロックを使って条件分岐を表現する

```
<script lang="ts">
  let flag = $state(false);
</script>

{#if flag}
  <p>OK</p>
{:else}
  <p>NG</p>
{/if}
```

SvelteKitとは

- Svelteを拡張したフロントエンドフレームワーク
 - SSR・SSG
 - APIルート
 - ファイルベースルーティング
- React・Next.jsの関係性と同じ

- サーバーサイドでレンダリングを実行
 - データフェッチもサーバーサイドで行う
 - 外部APIキーをフロント側から参照させない

```
/** @type {import("./$types").PageLoad} */  
export async function load({ params }) {  
  const res = await axios.get("/api/users")  
  return { users: res.data }  
}
```

APIルート

- APIエンドポイントを作成できる
 - フロントとサーバーの間APIとして使う
 - 外部APIキーをフロント側から参照させない

```
/** @type {import("./$types").RequestHandler} */  
export async function POST({ params }) {  
  const res = await axios.post("https://hoge/api/user, { user: ... }");  
  return json({user: res.data});  
}
```

動的ルーティング

- 任意のパスパラメータでルーティングする
 - URLが `/users/任意の文字列` であればマッチする
 - それ以外は404エラー

```
•  
└── routes  
    ├── users  
        └── [userId]  
            ├── +page.server.ts  
            └── +page.svelte
```

作ったもの

- <https://game-music.vercel.app>
- <https://github.com/meuzi8122/game-music>

技術スタック

- SvelteKit
- TailwindCSS + daisyUI
- Playwright
- MicroCMS
- Vercel

- TailwindCSSのコンポーネントライブラリ
- 短いクラス名でデザインを適用できる

```
<!-- プライマリカラーのボタン -->  
<button class="btn btn-primary">送信</button>
```

Playwright

- E2Eテストのフレームワーク
- APIのモック機能が備わっている

```
test("ユーザー一覧画面のテスト", async ({ page }) => {
  await page.route("*/**/api/user", async ( route ) => {
    route.fulfill({
      status: 200,
      body: JSON.stringify({
        contents: [{ name: "TEST_USER" }]
      })
    })
  })
});

/* アサート文を書いていく */
});
```

その他

- MicroCMS
 - CMS（コンテンツ管理システム）を提供するサービス
 - SDKが公開されており、APIアクセスが容易
- Vercel
 - Webアプリのホスティングサービス
 - 商用利用でなければ無料で使える

感想・今後の課題

- ステートの記述が少ないので楽
- gotoメソッドを使ったページ遷移が遅い
 - Aタグからの遷移は遅くない
- プレビュー用の音源を取得しているので、再生がぶつ切りになる
 - Appleアカウント認証 → フル再生できる音源を取得
 - 曲の終わりが近づいたら徐々に音量を小さくする