

Exercicio 67: SvelteKit na creación de aplicacións web

Obxectivo: Crear unha aplicación de busca de películas.

Preparativos: Antes de nada temos que asegurarnos de que temos instalado NodeJS no ordenador. Tamén deberemos de crear unha carpeta chamada *Exercicio67* onde gardaremos o proxecto.

E recomendable abrir a páxina web da [documentación oficial de SvelteKit](#) para ter acceso rápido a ela e consultala en caso de dúbida.

1. Configuración do proxecto:

- Crea un novo proxecto SvelteKit usando a plantilla oficial: ``npx sv create my-movie-app``. Despois disto deberás seguir os pasos que vai marcando o terminal.
- Navega ao directorio do proxecto: ``cd my-movie-app``.
- Instala os paquetes de NPM: ``npm install``
- Inicia o servidor de desenvolvemento: ``npm run dev``.
- Abre a aplicación no teu navegador e asegúrate de que todo está funcionando correctamente.

2. Interface de usuario inicial:

- Creamos o ficheiro ``+layout.svelte`` en ``src/routes`` e dentro do ficheiro engade a etiqueta `<slot>`.
- En ``src/routes/+page.svelte``, crea unha interface de usuario simple cun campo de busca e un botón de busca.

3. Integración coa API de TMDb:

- Rexístrate no sitio web de TMDb e obteña unha chave de API (<https://developer.themoviedb.org/reference/intro/getting-started>).
- No teu proxecto SvelteKit, crea un arquivo ``.env`` na raíz e garda a túa chave de API: ``VITE_TMDB_API_KEY=your_api_key_here``.
- Agora podes importar a túa API KEY en ``src/routes/+page.svelte`` usando ``const API_KEY = import.meta.env.VITE_TMDB_API_KEY``.
- En ``src/routes/+page.svelte``, utiliza ``fetch`` para realizar solicitudes á API de TMDb e obter resultados de busca de películas.

4. Mostrar resultados de busca:

- Amosa os resultados da busca na interface de usuario. Podes amosar información como o título, a descrición e o póster das películas.

5. Detalles da película:

- Implementa a funcionalidade para amosar detalles completos dunha película cando se fai clic nela.
- Crea unha nova páxina en ``src/routes/[id].svelte`` para amosar os detalles.

6. Estilo:

- Engade estilos para facer que a aplicación sexa visualmente atractiva.
- Utiliza clases CSS e/ou estilos en liña segundo sexa necesario.

7. Manexo de erros:

- Implementa manexo de erros para casos nos que a busca non devolva resultados ou se haxa algún problema coa conexión á API.

8. Melloras adicionais (Opcionais):

- Implementa filtros para clasificar as películas por xénero ou ano.
- Garda as películas favoritas do usuario no almacenamento local (*localStorage*).

Entrega: Unha vez completado todo **subiremos a carpeta *Exercicio67*** ao espazo para entrega de exercicios que temos no OneDrive de Microsoft Teams.

Ejercicio 67: SvelteKit en la creación de aplicaciones web

Objetivo: Crear una aplicación para buscar películas.

Preparativos:

Antes de comenzar, asegúrate de tener NodeJS instalado en tu ordenador. También deberás crear una carpeta llamada *Ejercicio67* donde guardarás el proyecto.

Es recomendable abrir la página web de la [documentación oficial de SvelteKit](https://svelte.dev/docs) para tener acceso rápido y consultarla en caso de dudas.

1. Configuración del proyecto:

- Crea un nuevo proyecto SvelteKit utilizando la plantilla oficial:

```
npx sv create my-movie-app
```

Sigue los pasos indicados por el terminal.

- Navega al directorio del proyecto:

```
cd my-movie-app
```

- Instala los paquetes necesarios con NPM:

```
npm install
```

- Inicia el servidor de desarrollo:

```
npm run dev
```

- Abre la aplicación en tu navegador y verifica que todo funcione correctamente.

2. Interfaz de usuario inicial:

- Crea el archivo `+layout.svelte` en `src/routes` y añade la etiqueta `<slot>` dentro del archivo.
- En `src/routes/+page.svelte`, diseña una interfaz simple con un campo de búsqueda y un botón para buscar.

3. Integración con la API de TMDb:

- Regístrate en el sitio web de TMDb y obtén una clave API (<https://developer.themoviedb.org/reference/intro/getting-started>).

- En tu proyecto SvelteKit, crea un archivo `.env` en la raíz y guarda tu clave API:

```
VITE_TMDB_API_KEY=your_api_key_here
```

- Importa tu clave API en `src/routes/+page.svelte` usando:

```
const API_KEY = import.meta.env.VITE_TMDB_API_KEY;
```

- Utiliza `fetch` en `src/routes/+page.svelte` para realizar solicitudes a la API de TMDb y obtener resultados de búsqueda.

4. Mostrar resultados de búsqueda:

- Muestra los resultados en la interfaz, incluyendo información como el título, descripción y póster de las películas.

5. Detalles de películas:

- Implementa la funcionalidad para mostrar detalles completos de una película al hacer clic sobre ella.
- Crea una nueva página en `src/routes/[id].svelte` para mostrar los detalles.

6. Estilo:

- Añade estilos para que la aplicación sea visualmente atractiva.
- Usa clases CSS y/o estilos en línea según sea necesario.

7. Manejo de errores:

- Implementa manejo de errores para casos donde no haya resultados o surjan problemas con la conexión a la API.

8. Mejoras adicionales (Opcionales):

- Implementa filtros para clasificar las películas por género o año.
- Guarda las películas favoritas del usuario en el almacenamiento local (*localStorage*).

Entrega:

Una vez completado, sube la carpeta *Ejercicio67* al espacio de entrega de ejercicios disponible en OneDrive a través de Microsoft Teams.