**Dokumentácia - GPU Comparison Platform**

**1. Opis architektúry riešenia**

Aplikácia **GPU Comparison Platform** je navrhnutá pomocou **modulárnej architektúry** a využíva moderné technológie na separovanie frontendovej a backendovej logiky.

* **Frontend:**
  + Aplikácia je postavená na **React.js**, ktorý poskytuje dynamické a interaktívne používateľské rozhranie.
  + Na správu stavu aplikácie používame **React hooks** a **Context API**.
  + **Tailwind CSS** je použitý na tvorbu responzívneho dizajnu s podporou prepínania medzi svetlou a tmavou témou.
* **Backend:**
  + Backend je implementovaný pomocou **PHP**.
  + Komunikácia medzi frontend a backendom je realizovaná prostredníctvom **REST API**, ktoré zabezpečuje požiadavky na spracovanie údajov a autentifikáciu používateľov.
  + Backend spravuje aj databázové operácie a vykonáva validáciu údajov.
* **Databáza:**
  + Aplikácia využíva **MySQL** databázu na ukladanie informácií o používateľoch, GPU parametroch a výsledkoch porovnávania.
  + V databáze sú vytvorené tabuľky na ukladanie údajov o používateľoch.

**2. Inštrukcie na spustenie**

**Požiadavky:**

* **Node.js 16+**
* **PHP 7.4+**
* **MySQL 5.7+**
* **XAMPP/WAMP/LAMP**

**Frontend**

1. **Klonovanie repozitára:**  
   Skopírujte repozitár do vášho lokálneho prostredia:

git clone https://github.com/Propagandalfthered/WTP.git

1. **Inštalácia závislostí:**  
   Prejdite do priečinka s frontend aplikáciou a nainštalujte závislosti:

cd gpu-comparison/frontend

npm install

1. **Spustenie vývojového servera:**  
   Spustite frontend aplikáciu:

npm run dev

Aplikácia bude dostupná na http://localhost:5173.

**Backend**

1. **Nainštalujte XAMPP/WAMP/LAMP:**  
   Pre backend budete potrebovať lokálny server, napríklad **XAMPP**, ktorý obsahuje Apache, PHP a MySQL.
2. **Skopírovanie backend súborov:**  
   Skopírujte priečinok /backend do adresára htdocs v XAMPP:

/htdocs/wtp/backend

1. **Vytvorenie databázy:**  
   Otvorte MySQL a vytvorte databázu s názvom registracia:

mysql -u root -p

CREATE DATABASE registracia;

1. **Importovanie schémy:**  
   Naimportujte databázovú schému do vytvorenej databázy:

mysql -u root -p registracia < /path/to/backend/database/schema.sql

1. **Úprava konfigurácie:**  
   Upravte konfiguráciu databázy v súbore db\_config.php podľa vašich lokálnych nastavení (názov databázy, používateľ, heslo).
2. **Spustenie backendu:**  
   Backend aplikácia bude bežať na serveri Apache cez XAMPP a bude dostupná na http://localhost/wtp/backend.

**3. Použité technológie**

* **Frontend:**
  + **React.js** pre vývoj dynamického užívateľského rozhrania
  + **Tailwind CSS** pre responzívny dizajn
  + **React Router** pre navigáciu medzi stránkami
  + **React Hooks** na správu stavu
* **Backend:**
  + **PHP 7.4+** pre serverovú logiku
  + **MySQL** pre databázové operácie
* **Nástroje a knižnice:**
  + **Node.js** pre vývoj frontend aplikácie
  + **Postman** pre testovanie API

**4. Popis testovania**

**Testovanie aplikácie sa delí do niekoľkých častí:**

* **Frontend testovanie:**
  + **Unit testy:** Testovanie jednotlivých React komponentov pomocou **Jest** a **React Testing Library**.
  + **Integračné testy:** Testovanie interakcie medzi komponentmi a backendom.
  + **End-to-end testy:** Testovanie celkového chodu aplikácie a jej používateľských interakcií (napríklad pomocou **Cypress**).
  + **Testovanie prístupnosti:** Overovanie, či aplikácia spĺňa **WCAG 2.1** štandardy prístupnosti.
* **Backend testovanie:**
  + **API testy:** Testovanie REST API endpointov na overenie správnej funkcionality a správneho spracovania požiadaviek.
  + **Záťažové testy:** Testovanie, ako backend reaguje pri vysokom zaťažení, aby sa zabezpečila jeho stabilita a výkon.
  + **Bezpečnostné testy:** Testovanie ochrany proti **SQL Injection**, **XSS** a **CSRF**.
* **Výkonnostné testovanie:**
  + Testovanie času načítania stránky a optimalizácie databázových dotazov.
  + Monitorovanie výkonu sieťovej komunikácie medzi frontendom a backendom.

**Výsledky testovania:**

* Všetky **unit testy** a **integračné testy** prešli úspešne.
* API endpointy vracajú očakávané výsledky pri všetkých scenároch.