**Szerző**

Csapat: **Pap András, Farkas Péter, Pardi Polett**

Gyakorlatvezető tanár neve: **Várkonyi Tibor**

Feladat megnevezése: **Repülünk!**

**Tartalom**

Szerző 1

Tartalom 2

Felhasználói dokumentáció 3

Feladat 3

Környezet 3

Használata 3

Alkalmazások 3

Menüpontok 3

Kezdőlap 3

Hivatkozások 3

Reptér választás 3

Fekete / Fehér háttér 3

Adatok frissítés 4

Fejlesztői dokumentáció 5

**Felhasználói dokumentáció**

**Feladat**

Egy repülőgép felügyeleti szolgáltatást valósítottunk meg a weboldalunkon. A kiválasztott reptérnek láthatod a valós idejű indulási és érkezési, késési adatait az elkövetkezendő 8 órára. A programot egy ingyenes API szolgáltatással alkottuk meg PHP-ban. Felhasználóbarát, valamint a docker segítségével könnyen módosítható.

**Környezet**

Az oldal bármilyen böngészőben futtatható.

**Használata**

***Alkalmazások***

Alkalmazások: Docker Desktop, Visual Studio Code, GitHub Desktop

Letöltjük a GitHub-ról a projektünket, megnyitjuk a Visual Studio Code-ban. A menüpontokból kiválasztjuk a Terminalt majd New Terminalt. Eközben elindítjuk a docker programot. Ha ez megtörtént, visszalépünk a kódunkhoz és beírjuk a következőt: docker-compose up -d web. Ahhoz, hogy sikerüljön kell a projekt mappánkban lennie egy conf és egy www mappának, valamint egy .env, egy docker-compose.yml valamint egy Dockerfile-nak.

***Menüpontok***

*Kezdőlap*

Az oldal tetején egy menüsáv található, ahol az alábbi menüpontok vanna: Rólunk, Hivatkozások (ezek átirányítanak egy másik oldalra), egy lenyíló lista (ahol ki lehet választani a repülőteret, majd a hozzátartozó adatok megjelennek), egy gomb (amin ki lehet választani, hogy fehér vagy fekete színű legyen a háttérszín), egy ikon gomb (az adatok frissítését szolgálja).

A felhasználó, az oldalon láthatja, a kiválasztott repülőtérhez tartozó valós idejű 8 órán belüli indulási, érkezési, késési adatokat.

*Hivatkozások*

Az oldalon a felhasznált légitársaságok honlapjai vannak összeszedve egy táblázatba.

*Lenyíló lista*

A lenyíló listában találhatóak a repterek összeszedve. Ha 8 órán belül nem indul/érkezik járat automatikusan „Jelenleg egyik reptérről sem indul/érkezik járat 8 órán belül!” üzenet lesz a repterek kilistázása helyett.

*Fekete / Fehér háttér*

A felhasználó kiválaszthatja, az oldal háttérszínét, hogy fekete vagy fehér legyen.

*Adatok frissítése*

A képre kattintva az adatok lefrissülnek és betöltődnek a legfrissebb állapotra.

**Fejlesztői dokumentáció**

**Felület**

*HTML*

Az oldal megfelel a HTML5-ös szabványnak.

*CSS*

Az oldal megfelel a CSS 3-as szabványnak.

*PHP*

Az oldal megfelel a PHP 8-as szabványnak.

**Forráskód**

A teljes fejlesztői anyag a www nevű könyvtárban található meg.

**Szerkezet**

A futtatható oldalak a www mappán belül találhatóak. (index.php, links.php)

Az oldal a mentett adatait settings.json fájlban tárolja, illetve az API lekéréseket a data.json fájlba menti le, az 100 lekérésből kifutás elkerülése érdekében.

A style mappában találhatóak a kinézetért felelős .css fájlok. (main.css, dark.css, light.css)

A parts mappában a főbb oldalakon többször meghívott tartalmakat tároló fájlok találhatóak (header.php, footer.php)

A js mappában a Javascript fájl kapott helyet.

Az img mappában az oldal által használt képek vannak elhelyezve.

A php mappában pedig a vezérlésért felelős fájlok találhatóak. (airport.php, Datas.php, download.php, request.php, sheet.php, style.php)

*airport.php*

Ez a fájl felelős azért, hogy az index.php lenyíló listájában kiválasztott reptér adatai legyenek megjeleníve. A legutóbb kiválasztott reptér neve el van tárolva a settings.json fájlban, hogy frissítés után ne kelljen megint kikeresni. Ez a reptér van beolvasva a $settings változóba.

*Datas.php*

Itt található a Datas osztály, amely feldolgozza az API által lekért adatokat, illetve egy TimeDifference függvényt, ami majd a 8 órán belüli repülők számítását végzi.

private $airportsData; 🡪 Ebben tárolódnak a járatok részletes adatai.

private $airports; 🡪 Ebbe gyűjtjük ki az összes repteret (Ahol indul 8 órán belül járat)

private $airlines; 🡪 Ebben vannak tárolva a légitársaságok.

private $intervalHour; 🡪 Osztály szinten tárolja az időkorlátot, a módosíthatóság kedvéért.

Konstruktor 🡪 Értéket az a fenti adattagoknak.

GetDatas() 🡪 Visszaadja az $airportsData tömböt.

GetAirports() -> Visszaadja a $airports tömböt.

GetAirlines() ->Visszaadja az $airlines tömböt.

GetIntervalHours() / SetIntervalHours() 🡪 Visszaadja / beállítja az $intervalHour változót.

*download.php*

Elvégzi az API kérést, és lementi az adatait a data.json fájlba.

*request.php*

CallAPI() függvény van benne, ez kommunikál a REST API-val.

*sheet.php*

Na igen, ez egy papír. :D Itt olyan php kódok vannak amik nem illettek bele a többi fájl tartalmába.

*style.php*

A főoldali Fekete/Fehér stílus kiválasztását és beállítását végzni.