



Jelenlét követő alkalmazás fejlesztése

Készítette

Györkis Tamás

Programtervező informatikus BSc

Témavezető

Dr. Király Roland

Egyetemi docens

EGER, 2024

Tartalomjegyzék

Bevezetés	3
1. Alkalmazás bemutatása	4
1.1. Adatbázis	4
1.2. Általános funkciók, információk	4
1.3. Adminisztrátori nézet, funkciók	4
1.4. Tanári nézet, funkciók	4
1.5. Hallgatói nézet, funkciók	4
2. Felhasznált technológiák, csomagok	5
2.1. Laravel	6
2.1.1. Keretrendszer alapjai	6
2.1.2. MVC architektúra	6
2.1.3. Eloquent ORM	6
2.1.4. Blade templating engine	6
2.1.5. Laravel Fortify	6
2.1.6. Események, queue	6
2.1.7. Email küldés	6
2.2. Tailwind	6
2.2.1. DaisyUI	6
2.3. LiveWire	6
2.3.1. Ismertető, a csomag működése	6
2.3.2. LiveWire és SPA	6
2.3.3. Komponensek és data binding	6
2.3.4. Események, lapozás	6
2.3.5. Alpine.js: LiveWire és JavaScript kapcsolata	6
2.3.6. WireToast: LiveWire alapú értesítések	6
2.4. Websocket és Pusher szerviz	6
2.4.1. Websocket jelentősége	6
2.4.2. Pusher	6
2.4.3. Websocket integrálása a keretrendszerbe	6
2.4.4. Csatornák, események létrehozása	6

2.4.5.	Események fogadása JavaScript, LiveWire esetén	6
2.5.	QR-kód generálás	6
2.6.	Naptár - FullCalendar.js	6
2.6.1.	Beállításai	6
2.6.2.	Kapcsolat a keretrendszerrel LiveWire segítségével	6
2.7.	Diagramok - Chart.js	6
2.7.1.	Beállításai	6
2.7.2.	Kapcsolat a keretrendszerrel LiveWire segítségével	6
2.8.	Progressive Web Apps	6
2.8.1.	A PWA jelentése, jelentősége	6
2.8.2.	Laravel PWA csomag	6
2.8.3.	A manifest fájl	6
2.8.4.	Service Worker	6
3.	Az alkalmazás tesztelése	7
3.1.	Manuális tesztelés	7
3.2.	Automatizált tesztelés	7
3.3.	Terheléses tesztelés	7
3.4.	Laravel Pint és Github actions	7
4.	Alkalmazás telepítése	8
4.1.	Laravel Forge	8
4.2.	Kézi telepítés lépései	8
	Összegzés	9
	Irodalomjegyzék	10

Bevezetés

Amikor szakdolgozati témaválasztás előtt álltam, sokat gondolkoztam a témán. Mindenképpen egy olyan alkalmazást szerettem volna elkészíteni, mely ténylegesen hasznos is lehet. Egyetemi éveim alatt demonstrátorként tanítottam több féléven át az egyetemen, és innen jött a felismerés, hogy jó lenne, ha a hallgatók jelenlétét, hiányzásait ne táblázatokban kelljen vezetni, hanem egy külön eszköz legyen rá készítve. Innen származik az alkalmazás ötlete.

A megvalósítás során törekedtem arra, hogy egy jól átlátható, ergonomikus weboldalt készítsek, amit – esetleges kisebb módosításokkal – ne csak egyetemi környezetben lehessen használni. Ezekből kifolyólag egy webalkalmazást készítettem, mivel ezt a leg-egyszerűbb elérni, hiszen csak egy webböngésző szükséges hozzá. A tanulók és tanárok által használt oldalak reszponzívan lettek elkészítve, ezáltal biztosítva, hogy mobiltelefonon is használni lehessen. Illetve az alkalmazás PWA funkciókkal rendelkezik, aminek jelentőségére később térek ki, de előjáróban annyit érdemes megemlíteni róla, hogy lehetővé teszi a weboldal alkalmazásként való telepítését számítógép és telefon esetén is, ezáltal egy rendes alkalmazás érzését keltve.

A megvalósításhoz a Laravel keretrendszert használtam, mely egy PHP alapú, MVC keretrendszer, amivel tanulmányaim során találkoztam, és egyből megkedveltem. Nem titkolt célom a szakdolgozatommal, hogy bemutassam, hogy a mai, JavaScript preferált világban, továbbra is lehetséges modern, a mai kort kielégítő weboldalt készíteni PHP segítségével, amihez különböző csomagokat alkalmaztam, amiknek működését, illetve egymással való működését a későbbiekben fogom kifejteni.

1. fejezet

Alkalmazás bemutatása

1.1. Adatbázis

1.2. Általános funkciók, információk

1.3. Adminisztrátori nézet, funkciók

1.4. Tanári nézet, funkciók

1.5. Hallgatói nézet, funkciók

2. fejezet

Felhasznált technológiák, csomagok

2.1. Laravel

2.1.1. Keretrendszer alapjai

2.1.2. MVC architektúra

2.1.3. Eloquent ORM

2.1.4. Blade templating engine

2.1.5. Laravel Fortify

2.1.6. Események, queue

2.1.7. Email küldés

2.2. Tailwind

2.2.1. DaisyUI

2.3. LiveWire

2.3.1. Ismertető, a csomag működése

2.3.2. LiveWire és SPA

2.3.3. Komponensek és data binding

2.3.4. Események, lapozás

2.3.5. Alpine.js: LiveWire és JavaScript kapcsolata

2.3.6. WireToast: LiveWire alapú értesítések

2.4. Websocket és Pusher szerviz

2.4.1. Websocket jelentősége

2.4.2. Pusher

2.4.3. Websocket integrálása a keretrendszerbe

2.4.4. Csatornák, események létrehozása

2.4.5. Események fogadása JavaScript LiveWire esetén

3. fejezet

Az alkalmazás tesztelése

3.1. Manuális tesztelés

3.2. Automatizált tesztelés

3.3. Terheléses tesztelés

3.4. Laravel Pint és Github actions

4. fejezet

Alkalmazás telepítése

4.1. Laravel Forge

4.2. Kézi telepítés lépései

Összegzés

Irodalomjegyzék

- [1] FAZEKAS ISTVÁN: *Valószínűességszámítás*, Debreceni Egyetem, Debrecen, 2004.
- [2] TÓMÁCS TIBOR: *A valószínűességszámítás alapjai*, Líceum Kiadó, Eger, 2005.