

**SAPIENTIA ERDÉLYI MAGYAR TUDOMÁNYEGYETEM
MAROSVÁSÁRHELYI KAR,
INFORMATIKA SZAK**



SAPIENTIA
ERDÉLYI MAGYAR
TUDOMÁNYEGYETEM

SapiMITevents: Tanszéki események dokumentálása

DIPLOMADOLGOZAT

Témavezető:
Dr. Osztián Erika,
Egyetemi docens

Végzős hallgató:
Bakos Csongor-Attila

2023

UNIVERSITATEA SAPIENTIA DIN CLUJ-NAPOCA
FACULTATEA DE ȘTIINȚE TEHNICE ȘI UMANISTE,
SPECIALIZAREA INFORMATICĂ



UNIVERSITATEA
SAPIENTIA

Evenimentele SapiMIT: Documentarea evenimentelor
departamentului

LUCRARE DE DIPLOMĂ

Coordonator științific:
Dr. Osztián Erika,
Conferențiar universitar

Absolvent:
Bakos Csongor-Attila

2023

**SAPIENTIA HUNGARIAN UNIVERSITY OF
TRANSYLVANIA
FACULTY OF TECHNICAL AND HUMAN SCIENCES
COMPUTER SCIENCE SPECIALIZATION**



SAPIENTIA
HUNGARIAN UNIVERSITY
OF TRANSYLVANIA

SapiMITevents: Documenting the events of the department

BACHELOR THESIS

Scientific advisor:
Dr. Osztián Erika,
Associate Professor

Student:
Bakos Csongor-Attila

2023

LUCRARE DE DIPLOMĂ

Coordonator științific:
dr. Osztián Erika

Candidat: Bakos Csongor Attila
Anul absolvirii: 2023

a) Tema lucrării de licență: Evenimentele SapiMIT: Documentarea evenimentelor departamentului

b) Problemele principale tratate:

- Proiectarea și dezvoltarea platformei SapiMITevents- Gândire logică și rațională
- Explorarea diferitelor funcționalități pe care platforma SapiMITevents le va oferi: crearea și gestionarea de evenimente, înregistrarea participanților
- Interfața utilizatorului și experiența utilizatorului

c) Desene obligatorii:

- Diagrama cazurilor de utilizare
- Diagrama de arhitectură a sistemului
- Modelul entitate-asociere

d) Softuri obligatorii:

- Aplicație web bazată pe tehnologii PHP și Vue.js care stochează evenimentele departamentului de Matematică-Informatică. Aplicația permite căutarea și sortarea evenimentelor pe diferite criterii, acesta este creat pentru profesorii departamentului, pentru a păstra mai eficient evidența evenimentelor.

e) Bibliografia recomandată:

- [1] M. Bean. Laravel 5 Essentials: Explore the Fundamentals of Laravel, One of the Most Expressive and Robust PHP Frameworks Available. Community experience distilled. Packt Publishing, 2015.
- [2] Documentație Laravel 10.x, 2023. <https://laravel.com/docs/10.x/>

f) Termene obligatorii de consultații: de 2-3 ori pe lună

g) Locul și durata practicii: Universitatea „Sapientia” din Cluj-Napoca,
Facultatea de Științe Tehnice și Umaniste din Târgu Mureș, sala / laboratorul 414

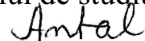
Primit tema la data de: 15.12.2022

Termen de predare: 15.06.2023

Semnătura Director Departament



Semnătura responsabilului
programului de studiu



Semnătura coordonatorului



Semnătura candidatului



Declarație

Subsemnatul/a BAKOS CSONGOR-ATTILA absolvent(ă) al/a specializării
INFORMATICA, promoția..... cunoscând
prevederile Legii Educației Naționale 1/2011 și a Codului de etică și deontologie profesională a
Universității Sapientia cu privire la furt intelectual declar pe propria răspundere că prezenta
lucrare de licență/proiect de diplomă/disertație se bazează pe activitatea personală,
cercetarea/proiectarea este efectuată de mine, informațiile și datele preluate din literatura de
specialitate sunt citate în mod corespunzător.

Localitatea,

Data:

TG-MUREȘ, 09.06.2023

Absolvent

Semnătura..... Bakos

Kivonat

Ez a szakdolgozat bemutatja egy eseménykezelő rendszer fejlesztését a Sapientie EM-TE Matematika-Informatika Tanszékére. A rendszer célja az egyetemi tagok, különösen a tanárok és munkatársak közötti koordináció és részvétel javítása. A weboldal lehetővé teszi az adminisztrátorok számára az események hozzáadását, frissítését és eltávolítását, míg a tanárok regisztrálhatják magukat résztvevőként és nyomon követhetik részvételi előzményeiket. A rendszer központosítja az események részleteit, ideértve az esemény típusát, helyszínét, dátumát, várható résztvevők számát és részletes leírásait. Az eseménykezelő rendszer hatékony eseményszervezést és kommunikációt biztosítva hozzájárul egy összetartó és elkötelezett akadémiai közösség kialakításához a tanszéken belül.

Az eseménykezelő rendszer központosított platformként szolgál az adminisztrátoroknak, hogy hatékonyan kezeljék a tanszéki eseményeket. Új események könnyedén hozzáadhatók, az eseményinformációk frissíthetők, és múltbeli események eltávolíthatók, ami átfogó képet nyújt a tanszék tevékenységeiről. A tanárok hasznát látják a rendszernek, mivel személyre szabott vezérlőpultot kínál számukra, ahol megtekinthetik a közelgő eseményeket, nyomon követhetik jelenlétük történetét és kezelhetik részvételüket különböző tevékenységekben. Ez a központosított megközelítés az eseménykezelésben kitölti a korábban fennálló űrt, lehetővé téve a tanárok számára, hogy könnyedén nyomon követhessék, hány eseményen vettek részt és milyen típusú eseményeken.

Kulcsszavak: eseménykezelő rendszer, részvétel, egyetem, informatika, tanári jelenlét.

Rezumat

Această lucrare de licență prezintă dezvoltarea unui sistem de gestionare a evenimentelor adaptat Departamentului de Matematică-Informatică de la Universitatea Sapiientia. Scopul sistemului este îmbunătățirea coordonării și implicării membrilor universității, în special a profesorilor și personalului. Website-ul permite administratorilor să adauge, actualizeze și elimine evenimente, în timp ce profesorii se pot înregistra ca participanți și pot urmări istoricul participării lor. Sistemul centralizează detalii despre evenimente, inclusiv tipul evenimentului, locația, data, numărul estimat de participanți și descrieri cuprinzătoare. Prin optimizarea organizării evenimentelor și îmbunătățirea comunicării, sistemul de gestionare a evenimentelor contribuie la crearea unei comunități academice mai coezive și implicare în cadrul departamentului.

Sistemul de gestionare a evenimentelor servește ca o platformă centralizată pentru administratori pentru a gestiona eficient evenimentele departamentului. Aceștia pot adăuga cu ușurință evenimente noi, actualiza informațiile despre evenimente și elimina evenimente trecute, oferind o privire de ansamblu asupra activităților departamentului. Profesorii beneficiază de sistem prin intermediul unui panou de control personalizat, în care pot vizualiza evenimentele viitoare, pot urmări istoricul participării și pot gestiona implicarea lor în diverse activități. Această abordare centralizată a gestionării evenimentelor acoperă lacuna existentă anterior, permițând profesorilor să urmărească ușor numărul și tipurile de evenimente la care au participat.

Cuvinte cheie: sistem de gestionare a evenimentelor, implicare, universitate, informatică, prezența profesorilor.

Abstract

This bachelor's thesis presents the development of an event management system tailored for the Department of Mathematics-Informatics at Sapientia Hungarian University of Transylvania. The system aims to enhance coordination and engagement among university members, particularly teachers and staff. The website provides administrators with the ability to add, update, and remove events, while teachers can register themselves as attendees and track their participation history. The system centralizes event details, including event type, location, date, expected number of attendees, and comprehensive descriptions. By streamlining event organization and improving communication, the event management system contributes to a more cohesive and engaged academic community within the department.

The event management system serves as a centralized platform for administrators to efficiently manage departmental events. They can easily add new events, update event information, and remove past events, providing a comprehensive overview of the department's activities. Teachers benefit from the system by having access to a personalized dashboard where they can view upcoming events, track their attendance history, and manage their participation in various activities. This centralized approach to event management fills the gap that previously existed, allowing teachers to easily track the number and types of events they have participated in.

Keywords: event management system, engagement, university, informatics, teacher attendance.

Tartalomjegyzék

1. Bevezető	11
2. Célkitűzések	12
3. Elméleti megalapozás és szakirodalmi tanulmány	13
3.1. Hasonló alkalmazások	13
3.1.1. Eventbrite	13
3.1.2. Meetup	13
3.1.3. Eventzilla	13
3.1.4. Hasonlítás a már létező alkalmazásokhoz	13
3.2. Felhasznált technológiák	14
3.2.1. Szoftverek	14
3.2.2. Szerver oldali technológiák	15
3.2.3. Kliens oldali technológiák	17
4. A rendszer specifikációja	19
4.1. Felhasználói követelmények	19
4.1.1. Használati esetek leírása	20
4.2. Rendszerkövetelmények	20
4.2.1. Funkcionális követelmények	21
4.2.2. Nem funkcionális követelmények	21
5. Tervezés	23
5.1. A rendszer architektúrája	23
5.2. Adatbázis terv	24
5.3. Felhasználói interfész	25
6. Kivitelezés	26
6.1. Az alkalmazás bemutatása	26
6.1.1. Bejelentkezés	28
6.1.2. Főoldal	28
6.1.3. Rendezvény részleteinek oldala	31
6.1.4. Rendezvények/Tanárok kezelőfelülete	32
6.1.5. Tanárok oldala	34
6.1.6. Profil	35
Összefoglaló	36

6.2. Következtetések	36
6.3. Továbbfejlesztési lehetőségek	36
Ábrák jegyzéke	37
Irodalomjegyzék	37

1. fejezet

Bevezető

A hatékony eseménykezelés rendkívül fontos szerepet játszik a felsőoktatási intézményekben annak érdekében, hogy koordinációt, részvételt és zökkenőmentes kommunikációt teremtsen az egyetemi tagok között. Felismerve annak szükségességét, hogy központi platformot biztosítson az események kezelésére és nyomon követésére, a szakdolgozat célja egy dedikált weboldal fejlesztése a Matematika-Informatika Tanszék számára a Sapientia egyetemen. A weboldal elsődleges célja a tanszék specifikus igényeinek kielégítése egy intuitív eseménykezelő rendszer létrehozásával, amelyet elsősorban az egyetem tanárai és munkatársai számára fejleszttek.

A szakdolgozat fő célja egy felhasználóbarát eseménykezelő rendszer tervezése és megvalósítása, amely lehetővé teszi a tanároknak és a munkatársaknak, hogy hatékonyan szervezzék, nyomon kövessék és részt vegyenek a tanszéki eseményekben. Hagyományosan hiányzott egy központi eszköz, amely lehetővé tette volna az eseménystatisztikák nyomon követését, a tanárok jelenlétének nyilvántartását és az események részleteinek hatékony kezelését. Ez a weboldal azon igyekszik segíteni, hogy áthidaljuk ezt a hiányt, és átfogó megoldást nyújtsunk, amely javítja a koordinációt, növeli a részvételt és egyszerűsíti az eseménykezelési folyamatokat.

2. fejezet

Célkitűzések

Főbb funkciók:

A tervezett eseménykezelő rendszer több kulcsfontosságú funkciót foglal magában, amelyek kifejezetten a Matematika-Informatika tanszék tanárainak igényeit célozzák:

- Események nyomon követése és kezelése: A weboldal lehetővé teszi az adminisztrátorok számára az események könnyű hozzáadását, frissítését és eltávolítását, egy központi helyen megőrizve a tanszék tevékenységeit. A tanároknak hozzáférésük lesz egy vezérlőpulthoz, ahol megtekinthetik a közelgő eseményeket, nyomon követhetik jelenlétük történetét és kezelhetik részvételüket.
- Tanári jelenlét kezelése: Az eseménykezelő rendszer segítségével a tanárok regisztrálhatják magukat az egyes eseményekre való résztvevőként. Ez a funkció pontos nyilvántartást biztosít és segíti a hatékony eseménytervezést, valós idejű jelenlétinformációval szolgálva az adminisztrátoroknak.
- Események részletei és információi: Minden eseménynek lesz egy dedikált oldala, amely tartalmazza a lényeges információkat, mint az esemény típusa, helyszíne, dátuma, résztvevők száma és részletes leírása. Ez a központi adatbázis átfogó áttekintést nyújt a múltbeli és közelgő eseményekről a tanárok számára.

3. fejezet

Elméleti megalapozás és szakirodalmi tanulmány

3.1. Hasonló alkalmazások

3.1.1. Eventbrite

Az **Eventbrite**[6] egy online eseménykezelő platform, amely lehetővé teszi az események létrehozását, promoválását és jegyértékesítését. Az eseményeket kategóriákba lehet sorolni, és részletes információkat lehet megadni, mint például a dátum, a helyszín, a jegyárak és a leírás. A felhasználók regisztrálhatnak az eseményekre és jegyeket vásárolhatnak.

3.1.2. Meetup

A **Meetup**[14] egy közösségi alapú platform, amely lehetővé teszi az emberek számára, hogy találkozókat szervezzenek és csatlakozzanak másokhoz közös érdeklődési körök alapján. A platform lehetővé teszi az események létrehozását, a részletes információk megadását, a résztvevők regisztrációját és a kommunikációt az esemény résztvevői között.

3.1.3. Eventzilla

Az **Eventzilla**[7] egy online eseménymenedzsment platform, amely széles körű események kezelését teszi lehetővé, beleértve a konferenciákat, műhelyeket, koncerteket és sporteseményeket is. Segít az események regisztrációjában, jegyértékesítésben, az eseményadatok kezelésében, a résztvevők kommunikációjában és más adminisztratív feladatokban.

3.1.4. Hasonlítás a már létező alkalmazásokhoz

A felsorolt alkalmazásokon kívül még nagyon sok létezik, ezek hasonló funkciókat kínálnak, például lehetőséget nyújtanak az események létrehozására és kezelésére, részletes információk megadására, regisztrációra és kommunikációra a felhasználók között. Azonban fontos megjegyezni, hogy minden szolgáltatásnak megvannak a sajátosságai és

funkciói, így a SapiMITevents is egy specifikus célra készült, kifejezetten a Sapiientia EM-TE Matematika-Informatika tanszék számára.

3.2. Felhasznált technológiák

3.2.1. Szoftverek

A **Git**[8] egy elosztott verziókezelő rendszer, amely lehetővé teszi a fejlesztők számára a változtatások nyomon követését és kezelését a projektekben, az előzmények ellenőrzését és az esetleges problémák visszavonását. A Git rendkívül rugalmas és hatékony, lehetővé téve az ágak kezelését, a változások egyesítését és a visszatérési pontok létrehozását.

Egy könnyű és gyors rendszer, amely lehetővé teszi a fejlesztők számára a helyi gépükön végzett munka könnyű kezelését. Az elosztott jellege miatt minden fejlesztő saját másolattal rendelkezik a teljes kódbázisról, és ez lehetővé teszi a hatékony párhuzamos fejlesztést.

A **GitHub**[9] egy felhőalapú Git tároló és együttműködési platform, amely lehetővé teszi a központi tárolást és a fejlesztők közötti könnyű csapatmunkát. A GitHub könnyen hozzáférhető és felhasználóbarát felületet biztosít, ahol a fejlesztők megoszthatják a kódot, együtt dolgozhatnak másokkal, és kezelhetik a projekteket.

A két rendszer kombinációja hatékony eszköztár a verziókezeléshez, a kódmegosztáshoz és a fejlesztői együttműködés támogatásához, hozzájárulva egy hatékonyabb fejlesztési folyamathoz.

A SapiMITevents kidolgozásakor a két rendszer kombinációját használtam, ami lehetővé tette, hogy több eszközzel is lehessen a projektet fejleszteni, szinkronizálva a kódbázist ezek között.

A **PhpStorm**[16] egy professzionális integrált fejlesztői környezet (IDE¹) PHP-hez, amelyet a JetBrains fejlesztett. Az IDE erőteljes funkciókat kínál a fejlesztéshez, például kódkiegészítést, hibakeresést, refaktorálást és verziókezelési támogatást. Kényelmes és hatékony fejlesztési környezetet biztosít a Laravel keretrendszerrel való munkához, segítve a kódírást és a projekt kezelését.

A program speciálisan a PHP fejlesztők igényeire van optimalizálva, és számos fejlett funkciót és eszközt kínál a hatékonyabb fejlesztés érdekében. Intelligens kódkiegészítést biztosít, amely segít a gyorsabb és pontosabb kódírásban. A beépített kódsablonokkal és automatikus kódkiegészítéssel időt és erőfeszítést lehet megtakarítani. A hibakeresés és a hibajavítás során számos eszközt és lehetőséget kínál. Például a beépített hibakeresővel lépésről lépésre lehet követni a kód végrehajtását, és ellenőrizni a változók értékeit.

Támogatja a verziókezelést, különösen a Git-et. Beépített integrációval rendelkezik, amely lehetővé teszi a változtatások nyomon követését, a commitok kezelését és az ágak közötti navigálást.

Lehetővé teszi a projektkezelést és a fájlrendszer navigálását. Könnyen át lehet tekinteni a projekt struktúráját, hozzá lehet férni az összes fájlhoz és mappához, és gyorsan lehet keresni a kódban illetve a fájlok között.

¹Integrated Development Environment

Erős refaktorálási lehetőségeket kínál, amelyek megkönnyítik a kód átszervezését és a kódminőség javítását. Például át lehet nevezni az osztályokat, metódusokat vagy változókat, és az IDE automatikusan frissíti az összes hivatkozást.

Az alkalmazás fejlesztése során a PhpStorm-ot használtam, ami nagyban megkönnyítette és felgyorsította a fejlesztést.

A **Docker**[5] egy nyílt forráskódú konténerizációs platform, amely lehetővé teszi az alkalmazások gyors és egyszerű csomagolását, szállítását és futtatását. A konténerek könnyen hordozhatóak és konzisztensek az összes környezetben, függetlenül attól, hogy milyen rendszeren vannak futtatva, lokálisan fejlesztés történik vagy éppen éles környezetben üzemel.

Segítségével egyszerűen lehet elvégezni az alkalmazások telepítését, konfigurálását és skálázását, valamint hatékonyan lehet kezelni az alkalmazásfüggőségeket és a verziókezelést. A konténerizáció technológiája lehetővé teszi a környezetek izolációját, így egymástól függetlenül lehet futtatni több alkalmazást anélkül, hogy konfliktusok vagy erőforrásproblémák jelentkeznének.

Rendkívül népszerű a fejlesztői közösségben és a vállalati környezetekben egyaránt, mivel segít egyszerűsíteni és felgyorsítani a fejlesztési folyamatokat, valamint biztosítja a rugalmasságot és skálázhatóságot az alkalmazások kezelése során.

Sokat segített a Docker az alkalmazás bármilyen környezetben való gyors futtatása során.

3.2.2. Szerver oldali technológiák

A **MariaDB**[13] egy nyílt forráskódú relációs adatbázis-kezelő rendszer, amely az eredeti MySQL adatbázis-kezelő rendszerre épül. Fő célja az adatok hatékony tárolása, lekérdezése és kezelése. Az adatok strukturált táblákban vannak tárolva, és a rendszer számos lehetőséget kínál a táblák és az adatok kezelésére. A MariaDB támogatja a tranzakciókat, amelyek lehetővé teszik az adatok konzisztenciájának fenntartását és a hibák helyreállítását. Emellett különféle indexelési és lekérdezési mechanizmusokkal rendelkezik, amelyek gyors és hatékony adatelérést biztosítanak. Széles körben használt adatbázis-kezelő rendszer, amelyet a fejlesztők és az üzleti felhasználók egyaránt kedvelnek a könnyű kezelhetősége és a fejlett funkcionalitása miatt.

Az **Apache HTTP Server**[2] (röviden Apache) a világ egyik legnépszerűbb nyílt forráskódú webkiszolgáló szoftvere. Erőtéljes és megbízható kiszolgálóplatformot nyújt, amely lehetővé teszi a weboldalak és alkalmazások hosztolását és szállítását.

Több operációs rendszeren is futtatható, beleértve a Linuxot, a Windows-t és az macOS-t is. Könnyen konfigurálható és testreszabható, ami lehetővé teszi az alkalmazások és weboldalak optimális működését. A HTTP² protokollt használja a kliensek (böngészők) és a szerver közötti kommunikációra.

Magas szintű teljesítményt és skálázhatóságot biztosít, ami különösen fontos a forgalmas webhelyek és alkalmazások esetében. Támogatja a többszálú működést, amely

²Hypertext Transfer Protocol

lehetővé teszi a párhuzamos kiszolgálást és a nagy terhelés kiegyensúlyozását.

A **PHP**[15] (Hypertext Preprocessor) egy szerveroldali, szkriptelhető programozási nyelv, amelyet kifejezetten webfejlesztésre terveztek. Dinamikus jellegű nyelv, ami azt jelenti, hogy a kód futása közben interpretálódik, és lehetővé teszi a weboldalak és alkalmazások dinamikus generálását.

Sokoldalú és széles körben használt nyelv a webfejlesztésben. Támogatja a szerveroldali logikát, az adatbázis-kapcsolatokat, a fájlkezelést és a felhasználói műveleteket. A PHP kód beágyazható HTML³-dokumentumokba, és lehetővé teszi a dinamikus tartalmak létrehozását és manipulálását.

Nagy közösséggel és bőséges fejlesztői ökoszisztémával rendelkezik, amely számos kész modult, keretrendszert és eszközt kínál a gyorsabb és hatékonyabb fejlesztés érdekében. Támogatja a objektumorientált programozást (OOP), amely lehetővé teszi a kód újrafelhasználását és a fejlesztési folyamat strukturáltabbá tételét.

Könnyen tanulható és alkalmazható, ami az egyik oka annak, hogy olyan népszerű a webfejlesztők körében. A nyelv folyamatosan fejlődik és frissül, így a legújabb verziókban további funkciók és fejlesztői eszközök érhetők el. Lehetővé teszi a webes projektek gyors és hatékony megvalósítását, így egy ideális választás az interaktív és dinamikus webalkalmazások létrehozásához.

A **Laravel**[12] egy kiterjedt, nyílt forráskódú PHP-alapú webes alkalmazásfejlesztési keretrendszer, amely célja a gyors és hatékony webalkalmazások létrehozása. Számos funkcionalitást és eszközt biztosít a fejlesztők számára, hogy egyszerűen és strukturáltan készítsenek modern webalkalmazásokat.

Ez a keretrendszer moduláris és skálázható. Tartalmazza az alapvető műveleteket és eszközöket, mint például az útvonalkezelés, a sémaépítő, az adatbázis-migráció és a felhasználókezelés. Emellett számos beépített funkciót és komponenst tartalmaz, például az autentikációt, az e-mail küldést, a cache-kezelést és az események kezelését.

Támogatja az objektumorientált programozást és a MVC⁴ tervezési mintát, amely segít, hogy a kód tiszta és strukturálható maradjon. Egyszerű és elegáns szintaxissal rendelkezik, amely lehetővé teszi a hatékony kódírást és a fejlesztési idő csökkentését. A modern webes fejlesztési gyakorlatokat támogatja, mint például a REST⁵ API⁶-k, a tesztelés, a verziókezelés és a fejlesztési környezetek automatizálása.

Könnyen tanulható és alkalmazható, ami miatt népszerű a webfejlesztők körében. A keretrendszer hatékonyan kezeli a gyakori feladatokat és a kódbázis szervezését, ezáltal segítve a hatékony és rugalmas fejlesztést. Egy erőteljes eszköz a webalkalmazások létrehozásához, és lehetővé teszi a fejlesztők számára, hogy gyorsan és megbízhatóan hozzanak létre modern és robusztus webalkalmazásokat.

³Hypertext Markup Language

⁴Model-View-Controller

⁵Representational State Transfer

⁶Application Programming Interface

A **REST API**[17] az egyik leggyakrabban használt megközelítés az alkalmazások közötti kommunikációra. A Laravel támogatja ennek a kialakítását és implementálását.

Beépített eszközökkel rendelkezik a REST API-k kényelmes kialakításához, például az útválasztás és a vezérlők segítségével. A Laravel nyújtja a könnyű adatelérési réteget, például az Eloquent ORM⁷-t, ami lehetővé teszi az egyszerű adatbázis műveletek kezelését a REST API-n keresztül. Egyszerűen definiálhatók a különböző HTTP metódusokhoz tartozó útvonalak, és azokhoz a megfelelő vezérlők rendelkeznek, így lehetővé téve a válaszok dinamikus generálását az API hívásokra.

A Laravel és a REST API együttes használata lehetővé teszi a könnyű és hatékony adatcserét különböző alkalmazások között, és egy jól strukturált, könnyen bővíthető API-t eredményez.

3.2.3. Kliens oldali technológiák

A **Laravel Blade**[10] egy beágyazott PHP sablonmotor, amelyet a Laravel keretrendszer részeként fejlesztettek ki. Lehetővé teszi a kód és a HTML egyszerű kombinálását a dinamikus weboldalak létrehozásához. Lehetőséget biztosít a sablonok öröklésére és a részleges nézetek használatára, ami lehetővé teszi a kód újrafelhasználását és a fejlesztési idő csökkentését. A sablonmotor továbbá támogatja a sablonok elrendezését és a szekciók definiálását, ami segíti a tartalomstruktúra hatékony kezelését.

Kihasználja a Laravel keretrendszer többi funkcióját és eszközét. Egyszerű beillesztetőséget és kapcsolatot kínál a vezérlőkkel, modellekkel és adatbázissal, ami lehetővé teszi a dinamikus adatok kezelését és megjelenítését.

A **Vue.js**[18] egy progresszív, nyílt forráskódú JavaScript keretrendszer, amelyet a felhasználói felületek fejlesztésére terveztek. Lehetővé teszi a dinamikus és interaktív webes alkalmazások létrehozását, amelyek gyorsan és hatékonyan működnek a böngészőben.

Kiválóan alkalmas a felhasználói interfészek összetett felépítésére, mivel moduláris megközelítést kínál. A keretrendszer komponensekre osztja fel az alkalmazást, amelyek önállóan működnek és újrahasznosíthatók. Ez lehetővé teszi a kódbázis könnyű karbantartását és bővítését.

Rendelkezik egy kifejező sablonrendszerrel, amely lehetővé teszi a HTML és JavaScript kombinálását a komponensek létrehozásához. Ez átlátható és könnyen olvasható kódot eredményez, és lehetővé teszi a fejlesztők számára, hogy hatékonyan dolgozzanak a felhasználói interfészekkel.

Egy reaktív adatkezelési rendszert kínál, amely lehetővé teszi az alkalmazás állapotának könnyű nyomon követését és frissítését. A változások automatikusan frissülnek a felhasználói felületen, ami gyors és interaktív élményt nyújt.

Széles körben használt a webes fejlesztésben, és rengeteg közösségi támogatást és bővítményt kínál. A keretrendszer gyorsan tanulható és egyszerűen integrálható más projektekbe. Emellett jól skálázható, ami lehetővé teszi az alkalmazások növekedését és a nagyobb projektek kezelését.

⁷Object-relational mapping

A **Vuetify**[19] egy Material Design alapú komponenskönyvtár a Vue.js-hez, amely lehetővé teszi a gyors és egyszerű felhasználói felületek kialakítását. Egy készletet kínál olyan előre elkészített komponensekből, amelyek tartalmazzák a Material Design stílusát és funkcióit. Ezek a komponensek könnyen testreszabhatók és kombinálhatók, így rugalmasan alkalmazhatók a projekt igényei szerint. Továbbá tartalmaz kényelmes segédprogramokat és eszközöket a reszponzív elrendezések, a tipográfiák, a színek és az animációk kezeléséhez.

A Vue.js és a Vuetify együtt egy erőteljes kombinációt alkotnak a modern, felhasználóbarát és esztétikus webalkalmazások fejlesztéséhez.

A **Bootstrap**[4] egy népszerű, nyílt forráskódú front-end keretrendszer, amelyet a webfejlesztők széles körben használnak a reszponzív és modern webalkalmazások készítéséhez.

Lehetővé teszi a gyors és hatékony webalkalmazásfejlesztést a beépített stílusok, oszloprendszer és komponensek segítségével. Az előre elkészített stílusok, mint például a gombok, a táblázatok, a formák és a navigációs elemek, lehetővé teszik a felhasználóbarát felhasználói felületek egyszerű kialakítását. A reszponzív oszloprendszer lehetővé teszi a tartalom rugalmas elrendezését és alkalmazkodását a különböző eszközök képernyőméreteihez.

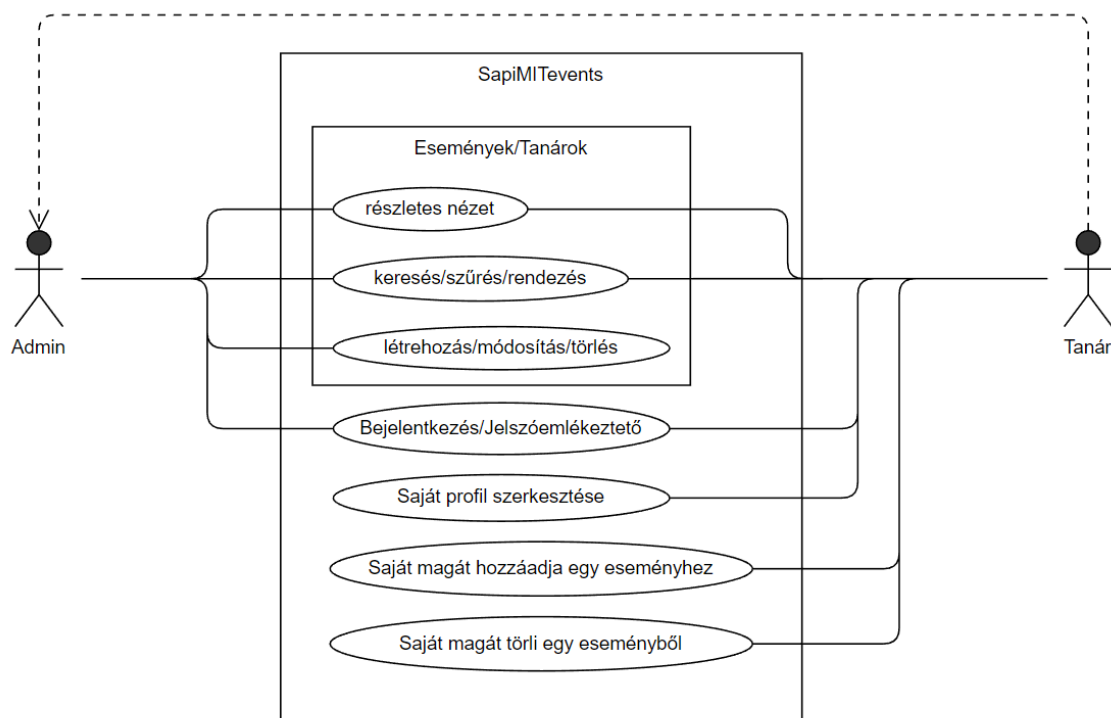
Az egyértelmű dokumentáció és a könnyen használható CSS és JavaScript komponensek révén gyorsan és könnyedén építhetünk felhasználóbarát felületeket. Továbbá támogatja a testreszabást és a kibővítést a saját stílusok és komponensek hozzáadásával.

4. fejezet

A rendszer specifikációja

4.1. Felhasználói követelmények

A következő fejezetben részletesen bemutatom a rendszer specifikációját a felhasználói követelmények szempontjából. Az alábbiakban felsorolom a rendszer által elvárt funkciókat és szolgáltatásokat, amelyeket a felhasználói igények kielégítése érdekében fejlesztettem. Két típusú felhasználó létezik: tanár és adminisztrátor, ezeknek különböző jogosultságaik vannak.



4.1. ábra. Használati eset diagram.

4.1.1. Használati esetek leírása

- Bejelentkezés:
 - Az adminisztrátorok által előzetesen regisztrált felhasználók be tudnak jelentkezni a saját fiókjukba.
- Események kezelése:
 - Az adminisztrátorok hozzá tudnak adni a rendszerhez új eseményeket, szerkeszteni illetve törölni tudják ezeket.
 - Az események tartalmazznak információkat, mint például típus, helyszín, dátum, résztvevők száma, képek és részletes leírás.
- Események szűrése:
 - Az események szűrhetőek szöveges keresés segítségével.
 - Előre megadott értékek alapján is lehet keresni, amik figyelembe veszik a szöveges dobozba írt kulcsszót.
 - Rendezni lehet dátum és név szerint, csökkenő és növekvő sorrendbe.
- Részvétel nyomon követése:
 - A tanárok jelentkezni tudnak az egyes eseményekre résztvevőként illetve törölni is tudják magukat.
 - Az adminisztrátorok is meg tudják adni, hogy melyik tanár melyik eseményen vesz részt.
 - A rendszer nyomon követi a tanárok részvételét és megjeleníti azokat az esemény oldalán.
- Személyre szabott felhasználói felület:
 - A felhasználóknak van egy egyéni vezérlőpultjuk, ahol áttekintést kapnak a részvételi előzményeikről.
 - A rendszer testre szabható beállításokat biztosít, hogy a felhasználók személyre szabják adataikat.

Ezek a felhasználói követelmények a rendszer alapját képezik, hogy hatékonyan kezelni lehessen az eseményeket és nyomon követhető legyen a tanárok részvétele. A rendszer célja a felhasználói élmény javítása.

4.2. Rendszerkövetelmények

A következőkben bemutatok néhány rendszerkövetelményt a tervezett eseménykezelő rendszerhez, amelyek segítenek a funkcionalitás és hatékonyság biztosításában:

4.2.1. Funkcionális követelmények

- Bejelentkezés:
 - A rendszer lehetőséget kell biztosítson a felhasználók számára a bejelentkezésre.
 - A felhasználók számára egyedi felhasználói fiókokat kell létrehozni, amelyekhez hozzáférhetnek az események kezeléséhez és részvételéhez, ezeket az adminisztrátor kezeli, ő hozza létre őket.
- Események kezelése és szerkesztése:
 - Az adminisztrátoroknak engedélyezni kell az események létrehozását, szerkesztését és törlését.
 - Az eseményeknek tartalmazniuk kell olyan adatokat, mint típus, helyszín, dátum, résztvevők várható száma, képek és leírás.
- Felhasználói részvétel kezelése:
 - A tanároknak biztosítani kell az eseményeken való részvételük jelzését.
 - A rendszernek nyomon kell követnie a tanárok részvételét az eseményeken és meg kell jelenítenie ezeket az adatokat.
- Események keresése, szűrése és rendezése:
 - A rendszer a felhasználók számára keresést, szűrést és rendezést kell biztosítson. Az eseményekre való keresés szöveges formában is történhet. Szűrni típus, dátum vagy helyszín alapján lehet.
 - Az eseményeket rendezni is lehet különböző szempontok szerint, például dátum, vagy résztvevők száma alapján.

4.2.2. Nem funkcionális követelmények

- Teljesítmény:
 - Az alkalmazás gyorsan és hatékonyan kell működjön, még nagy adatforgalom vagy többidejű felhasználói aktivitás esetén is.
- Biztonság:
 - Megfelelő kell legyen a biztonság, bejelentkezés nélkül nem lehet hozzáférni az oldalhoz, így biztosítva, hogy a felhasználók adatai biztonságban maradnak.
- Megbízhatóság:
 - Az alkalmazás stabil és megbízható kell legyen, minimális rendszerhiba és leállás mellett.

- Felhasználói élmény:
 - Felhasználóbarát kell legyen, intuitív felhasználói felülettel és könnyen érthető navigációval, hogy a felhasználók könnyedén és hatékonyan megtalálják amire szükségük van.
- Karbantarthatóság:
 - Az alkalmazás jól strukturált és könnyen karbantartható kell legyen, a kód olvashatóságával és dokumentálásával.
- Skálázhatóság:
 - Képes kell legyen a bővíthetőségre, új funkciók hozzáadására vagy meglévő funkciók módosítására a jövőbeli igények kielégítése érdekében.
- Kompatibilitás:
 - Kompatibilis kell legyen különböző böngészőkkel és eszközökkel, hogy a felhasználók különböző platformokon könnyedén használhassák.

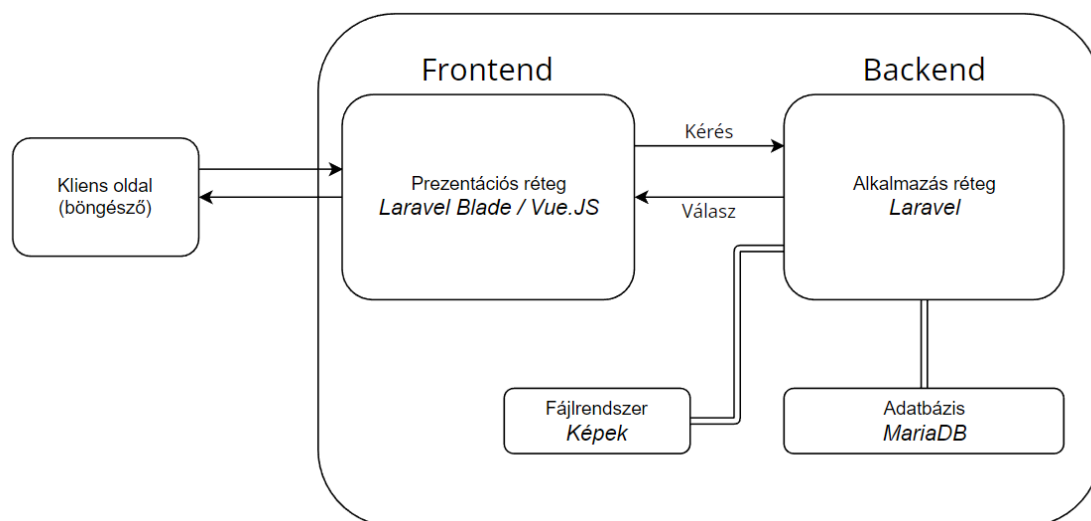
5. fejezet

Tervezés

5.1. A rendszer architektúrája

Három fő komponensből tevődik össze a rendszer: kliens oldal (frontend), szerver oldal (backend), adatbázis.

- Kliens oldal: A legfontosabb elem, amely aktív kapcsolatba lép a felhasználóval, fogadja a bemenetet és kezeli a prezentációs logikát, miközben a felhasználó és az alkalmazás közötti interakciókat irányítja. A felhasználói bemeneteket validálja is.
- Szerver oldal: A webkiszolgáló kezeli az üzleti logikát és feldolgozza a felhasználói kéréseket azáltal, hogy a kéréseket a megfelelő komponenshez irányítja, és kezeli az alkalmazás teljes működését. A backend szintén validálja a felhasználói bemeneteket.
- Adatbázis: Az adatbázis-kiszolgáló biztosítja a szükséges adatokat az alkalmazás számára. Az adatokkal kapcsolatos feladatokat kezeli.



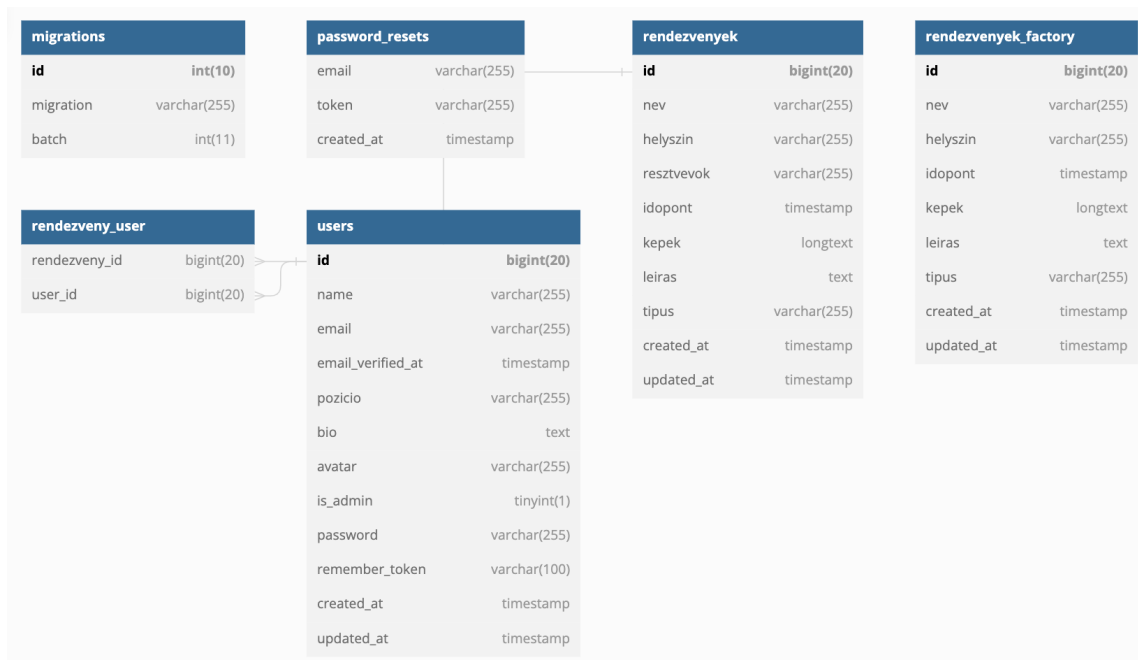
5.1. ábra. Rendszer architektúra diagram.

5.2. Adatbázis terv

Az adatbázis 6 táblából tevődik össze:

- *rendezvenyek*: A rendezvényeket és ezek részleteit tárolja: ID, név, helyszín, résztvevők száma, időpont, leírás, típus, illetve képek, amiknek a relatív elérési útvonalai JSON¹ formátumban vannak tárolva. Két dátum is szerepel benne, ezek a *created_at* és az *updated_at*, amik az adott modell létrehozásának illetve utolsó módosításának dátumát tárolják, ezeket a Laravel automatikusan kezeli.
- *rendezvenyek_factory*: Ebben találhatóak a rendszer által, tesztelés szempontjából kigenerált rendezvények. Struktúrája megegyezik a rendezvények tábláéval. Ez csak egy tartalék, a rendszer bármikor átváltható erre a táblára, ha tesztelésre lenne szükség, hogy ne sérüljenek az éles adatok.
- *users*: A felhasználók adatait illetve a bejelentkezéshez szükséges adatokat tartalmazza. Ilyenek az: ID, név, email, titulus, profilkép elérési helye, jelszó, egy token ami a bejelentkezésnél választható *"Jegyezz meg"* funkció működéséhez szükséges. Megtalálható egy *is_admin* nevű oszlop is, ami meghatározza, hogy egy felhasználó adminisztrátor jogosultságokkal rendelkezik-e. Ez egy boolean típusú változó ami alapértelmezetten hamis, a rendszer úgy volt tervezve, hogy egy adminisztrátora legyen. Itt is megtalálható a rendezvények táblánál említett *created_at* és *updated_at* mező.
- *rendezveny_user*: Ez a tábla biztosítja a kapcsolatot a rendezvények és felhasználók között, két idegen kulcs (*rendezveny_id*, *user_id*) segítségével jönnek létre a *több-a-többhöz* kapcsolatok.
- *password_resets*: Ennek segítségével követi nyomon a rendszer a jelszó visszaállítási kérélmeket.
- *migrations*: A rendszer által használt tábla, amely lehetővé teszi az adatbázis sémájának változtatását és a táblák létrehozását/módosítását a Laravel keretrendszerben. A migrációk egyszerű PHP osztályok, amelyekben az adatbázis változtatásainak utasításai vannak definiálva, segítve az adatbázis integritásának fenntartását és az alkalmazás adatbázisstruktúrájának kezelését.

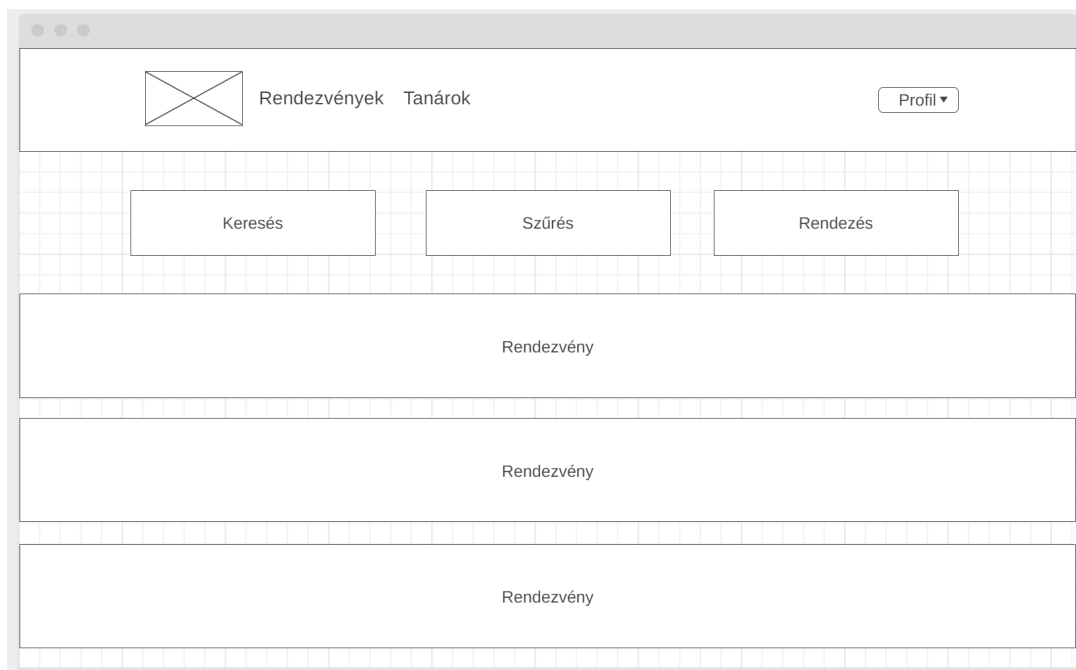
¹JavaScript Object Notation



5.2. ábra. Adatbázis egyed-kapcsolat modell.

5.3. Felhasználói interfész

A felhasználói interfész tervezése során wireframe-et, azaz egy szerkezeti vázat használtam arra, hogy a kezdeti fázisban megtervezsem az alkalmazás kinézetét, majd a későbbiekben ezt kibővítve, egy egyszerű design-t hozzak létre a felhasználók számára.



5.3. ábra. Wireframe.

6. fejezet

Kivitelezés

6.1. Az alkalmazás bemutatása

Az alkalmazás írása során a Laravel dokumentációja[11] és a *Laravel 5 Essentials*[1] nevű könyv sokat segített, ezek segítségével nagyon gyorsan megtaláltam mindent amire szükségem volt.

Laravelben az útvonalak a *"web.php"* fájlban vannak tárolva. A 6.1 ábra tetején látható sor automatikusan generálja a hitelesítéshez tartozó útvonalakat, kivételt képez a regisztráció és a jelszó megerősítési útvonal, mivel ezekre nincsen szükség ebben az alkalmazásban.

```
Auth::routes(['register' => false, 'confirm' => false]);

Route::group(['middleware' => ['auth']], function () {
    Route::get(uri: '/', [RendezvényController::class, 'home'])->name(name: 'rendezvenyek.home');
    Route::get(uri: '/tanarok', [TanarController::class, 'home'])->name(name: 'tanarok.home');

    Route::post(uri: '/tanarok/search', [TanarController::class, 'search'])->name(name: 'tanarok.search');
    Route::post(uri: '/rendezvenyek/search', [RendezvényController::class, 'search'])->name(name: 'rendezvenyek.search');

    Route::get(uri: 'profile', [ProfileController::class, 'index'])->name(name: 'profile');
    Route::post(uri: 'profile', [ProfileController::class, 'store'])->name(name: 'submit-profile');
});

Route::group(['middleware' => ['auth'], 'prefix' => 'admin'], function () {
    Route::get(uri: '/rendezvenyek/{rendezveny}/toggle-relation', [RendezvényController::class, 'toggleTanarRelation'])
        ->name(name: 'rendezvenyek.toggle-relation');

    Route::resource(name: 'rendezvenyek', controller: RendezvényController::class)->parameters([
        'rendezvenyek' => 'rendezveny'
    ]);

    Route::resource(name: 'tanarok', controller: TanarController::class)->parameters([
        'tanarok' => 'tanar'
    ]);
});
```

6.1. ábra. Laravel útvonalak.

Egy route tartalmazza az útvonal típusát (POST, GET, PUTH, PATCH, DELETE), egy *controller*-en belüli metódus nevét amire mutat, és a nevét, ami segít, hogy a későbbiekben nagyon egyszerűen le lehessen kérni az útvonalat akár a kliens-, akár

a szerver-oldalon. Ez átláthatóságot és rugalmasságot eredményez, segítségével újra felhasználhatóak az útvonalak.

A *route* függvény szerver oldalon használható, de egy Ziggy[20] nevű könyvtár segítségével ez használható JavaScript-ben is, a kliens oldalon. Ez a segédfüggvény előre kigenerálja a Laravel által létrehozott útvonalakat, így bárhol könnyedén lehet használni őket.

Létezik egy *resource* típusú route is, amit 1-1 modell controllerjéhez tudunk hozzárendelni, ami a CRUD¹ mintát segíti, egyetlen sorral létre tudunk hozni 7db útvonalat.

A keretrendszer egyik funkciója a *middleware*, ennek a segítségével megadhatjuk, hogy egy útvonal milyen feltételekkel érhető el, pl. csak bejelentkezett felhasználók érhetik el. Ez a funkció már előre meg van írva a rendszerben, de saját middleware-eket is létrehozhatunk, ilyen pl. az *IsAdmin*, ami segít, hogy csak adminisztrátorok férjenek hozzá az adott útvonalhoz, ellenkező esetben egy átirányítás történik a főoldalra. Egy route-hoz tartozhat több middleware is, ezek mind ellenőrizve vannak a hozzáférés előtt.

```
class IsAdmin
{
    • Csongor Bakos *
    public function handle(Request $request, Closure $next)
    {
        if (Auth::user() && Auth::user()->is_admin) {
            return $next($request);
        }

        return redirect(B: RouteServiceProvider::HOME);
    }
}
```

6.2. ábra. IsAdmin middleware.

¹Create, Read, Update, Delete

6.1.1. Bejelentkezés

Az alkalmazás kifejezetten a Sapientia egyetem egy adott szakának készült, azon belül is a tanárok számára, ezért a rendszer teljesen zárt, csak bejelentkezés után férhetnek hozzá az adatokhoz a felhasználók, vendégeknek nincs jogosultságuk használni.

Külső regisztrációra nincsen lehetőség, ezért a felhasználókat az adminisztrátor tudja létrehozni. Létrehozás után egy jelszó változtatására alkalmas hivatkozást küld a rendszer a felhasználó email címére. Erre kattintva tud a felhasználó jelszót állítani magának.

The image displays two web forms side-by-side. The left form, titled 'Bejelentkezés', contains input fields for 'E-mail' and 'Jelszó', a checkbox labeled 'Jegyezz meg', and two buttons: 'Bejelentkezés' and 'Elfelejtett jelszó?'. The right form, titled 'Jelszó helyreállítás', has an 'E-mail' field with the text 'admin@sapimitevents.ro' and a 'Küldés' button. Below this, there is another section with 'Jelszó' and 'Jelszó megerősítés' fields, and a 'Jelszó helyreállítás' button.

6.3. ábra. Bejelentkezés/Jelszóemlékeztető űrlap.

6.1.2. Főoldal

Bejelentkezés után a főoldalon látható fent a menüsor, itt lehet navigálni a tanárok oldalára illetve vissza a főoldalra, jobb oldalon látható a bejelentkezett felhasználó neve. Erre kattintva egy legördülő menü jelenik meg. Itt a saját profilt lehet megtekinteni és szerkeszteni. Kijelentkezésre is itt van lehetőség. Adminisztrátor esetén két újabb menüpont jelenik meg, itt lehet elérni a rendezvények és tanárok kezelőfelületét.

A menü alatt megjelenik a rendezvények listája, itt található egy szöveges mező amivel a rendezvények között lehet keresni, mellette az év és kategória legördülő menük, amik segítségével szűrni lehet. A legutolsó menü pedig rendezésre van, itt a rendezvények neve alapján lehet ábécé sorrendbe rendezni a listát. Az alapértelmezett rendezés a rendezvények dátuma szerint történik.

Minden rendezvénynek megjelenik a neve, a dátuma, egy lerövidített leírása, a résztvevőinek létszáma és a helyszíne. Alapértelmezetten 10 rendezvény jelenik meg, az oldal aljára görgetve megjelenik egy *további találatok* nevű gomb, ami segítségével újabb rendezvények kerülnek lekérésre az adatbázisból a szerveren keresztül.

Rendezvények			
Keresés...	×	Év ▾	Kategória ▾
			Időpont ▾
06 máj. 2023		Emter https://emter.ms.sapientia.ro/... Létszám: 65	Sapientia EMTE, C06/C07
17 márc. 2023		Make IT easy Hackathon! 10 csapat...	Sapientia EMTE, Aula

6.4. ábra. Rendezvények listája

Keresés

A keresés résznél is segítségemre volt a Vue.js, azon belül pedig a Vuetify-t használtam, ami egy Material Design alapú UI² keretrendszer, segítségével modern és stílusos felhasználói felületeket lehet létrehozni. Komponenseit felhasználva nagyon egyszerűen és gyorsan lehet egy letisztult applikációt létrehozni az előre megírt elemek felhasználásával. A 6.1 kódrészletben látható az a *\$watch* nevű funkció ami figyeli ha változik a beírt keresendő szó, egy *debounce* nevű funkció segítségével késlelteti ezt, így a benne levő függvényhívás csak *500ms* után fut le, ezzel megakadályozva azt, hogy a függvény minden beírt betű után lefusson.

```
this.$watch('searchTerm', _.debounce(function() {
  this.search()
}, 500))
```

6.1. kódrészlet. debounce funkció

Az Axios[3] könyvtár segítségével POST kéréseket tudok küldeni a szerver felé, ez segít abban, hogy az oldal újratöltése nélkül is lehessen keresni, és nagyon hamar megjelenjenek a találatok. Ebben a kódrészben az új eseményeket a *rendezvenyek* tömbbe tárolom el, illetve a következő oldal URL³-jét, amit a szerverről visszakapott *pagination* funkció ad meg, valamint egy logikai változóba azt, hogy van-e még több oldal vagy nincs, ezáltal tudom, hogy a keresés eredményeinek végére értem.

```
axios.post(this.getEventsUrl).then((response) => {
  if (response && response.data) {
    this.rendezvenyek.push(...response.data.data)
    this.getEventsUrl = response.data.next_page_url
  }
})
```

²User Interface

³Uniform Resource Locator

```

        this.hasMoreEvents = !!response.data.next_page_url
    }
    this.isLoadingMore = false
}).catch((e) => {
    console.log(e)
    this.isLoadingMore = false
})

```

6.2. kódrészlet. axios hívás

A 6.5 ábrán látható, hogy milyen eredményt küld vissza a Laravel a kliens oldalra, ebben benne vannak a rendezvény adatai mellett az oldalakra vonatkozó részletek, előző/következő oldal URL-je, hány oldal van összesen, hányadik oldalt kértük le éppen.

```

$rendezvenyek = $query->paginate(10);

return $rendezvenyek->withQueryString();

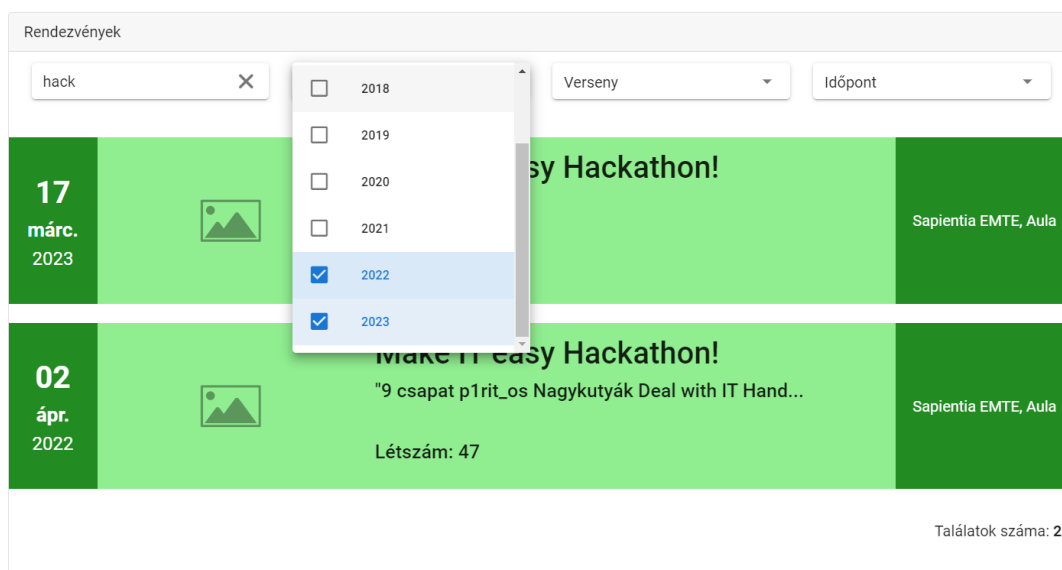
```

```

current_page: 1
▼ data: [{id: 57, nev: "Emter", helyszin: "Sapientia EMTE 114, 230, 231, 128", resztvevok: "~100",...}]
  ▼ 0: {id: 57, nev: "Emter", helyszin: "Sapientia EMTE 114, 230, 231, 128", resztvevok: "~100",...}
    created_at: "2023-06-02T12:46:39.000000Z"
    ev: "2019"
    helyszin: "Sapientia EMTE 114, 230, 231, 128"
    honap: "ápr."
    id: 57
    idopont: "2019-04-06"
    kepek: null
    leiras: "https://emter.ms.sapientia.ro/"
    nap: "06"
    nev: "Emter"
    resztvevok: "~100"
    tipus: "verseny"
    updated_at: "2023-06-02T12:46:39.000000Z"
  ► 1: {id: 46, nev: "Emter", helyszin: "Sapientia EMTE 114, 230, 231", resztvevok: "~90",...}
  ► 2: {id: 41, nev: "Emter", helyszin: "Sapientia EMTE 114, 230, 231", resztvevok: "84",...}
  first_page_url: "http://localhost/rendezvenyek/search?order_by=idopont&years%5B0%5D=2018&years%5B1%5D=2019&years%5B2%5D=2020"
  from: 1
  last_page: 1
  last_page_url: "http://localhost/rendezvenyek/search?order_by=idopont&years%5B0%5D=2018&years%5B1%5D=2019&years%5B2%5D=2020"
  links: [{url: null, label: "&laquo; Előző", active: false}, {url: null, label: "Következő &raquo;", active: false}]
  next_page_url: null
  path: "http://localhost/rendezvenyek/search"
  per_page: 10
  prev_page_url: null
  to: 3
  total: 3

```

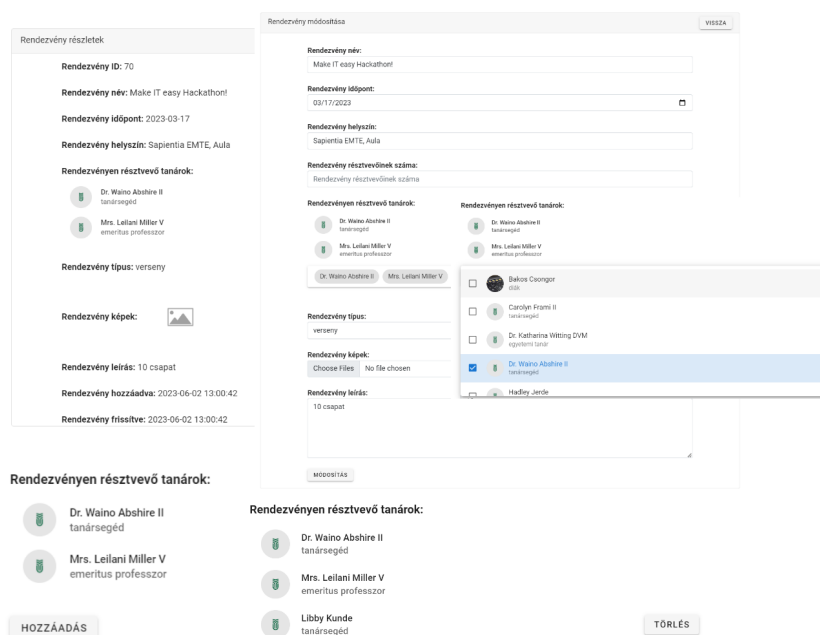
6.5. ábra. Lapszámozás



6.6. ábra. Kereső

6.1.3. Rendezvény részleteinek oldala

A rendezvény részleteinél megtalálható a név, időpont, helyszín, típus, képek, leírás, a létrehozás és utolsó módosítás időpontja és a résztvevő tanárok listája, itt a felhasználók hozzáadhatják saját magukat a rendezvényhez illetve törölhetik is, az adminisztrátoroknak fent megjelenik egy *Módosítás* nevű gomb, ahol módosítani tudják a rendezvény adatait.



6.7. ábra. Rendezvény részletei, módosítása, résztvevő tanárok hozzáadása/törlése

```
@if (Auth::user()->is_admin)
<v-btn
  small
  href="{{ route('rendezvenyek.edit', $rendezveny->id) }}"
>
  Módosítás
</v-btn>
@endif
```

6.8. ábra. if függvény Blade-ben

```
usage: Csongor Bakos
public function toggleTanarRelation(Rendezveny $rendezveny)
{
    $tanar_id = Auth::user()->id;

    $toggle = $rendezveny->tanarak()->toggle($tanar_id);

    return redirect()->route('rendezvenyek.show', $rendezveny->id)
        ->with('success', $toggle['attached'] ?
            'Sikeresen hozzáadtad magad ehhez a rendezvényhez!' :
            'Sikeresen törölted magad erről a rendezvényről!');
}
```

6.9. ábra. Tanár-Rendezvény kapcsolat váltása



A keretrendszer Blade nézeteiben nagyrészt HTML kód szerepel, de léteznek kiegészítő PHP funkciók is amiket fel lehet használni, ilyen pl. a 6.8 ábrán látható *if* függvény.

A tanárok és rendezvények közötti kapcsolatot nagyon egyszerűen lehet kezelni Laravel-ben, az MVC minta modelleinek segítségével egy egyszerű függvény használatával lehet összekötni vagy szétválasztani a több-a-többhöz kapcsolható modelleket. Ez látszik a 6.9 ábrán.



6.1.4. Rendezvények/Tanárok kezelőfelülete

A rendezvényeknek/tanároknak van egy külön oldaluk, ahol az adminisztrátor tudja ezeket létrehozni/módosítani/törölni.

ÚJ RENDEZVÉNY

ID	Rendezvény neve	Rendezvény időpontja	Rendezvény helyszíne	Rendezvény résztvevőinek száma	Rendezvény típusa	Rendezvény képek	Rendezvény leírása	Műveletek
71	Emter	2023-05-06	Sapientia EMTE, C06/C07	65	verseny		https://emter.ms.sapientia.ro/	<div>RÉSZLETEK</div> <div>MÓDOSÍTÁS</div> <div>TÖRLÉS</div>
70	Make IT easy Hackathon!	2023-03-17	Sapientia EMTE, Aula		verseny		10 csapat	<div>RÉSZLETEK</div> <div>MÓDOSÍTÁS</div> <div>TÖRLÉS</div>

ÚJ TANÁR

ID	Tanár neve	Tanár email	Tanár pozíciója	Tanár avatar	Műveletek
50	Carolyn Frami II	ethel.kozey@example.org	tanársegéd		<div>RÉSZLETEK</div> <div>MÓDOSÍTÁS</div> <div>TÖRLÉS</div>
49	Hadley Jerde	ikonopelski@example.com	emeritus professzor		<div>RÉSZLETEK</div> <div>MÓDOSÍTÁS</div> <div>TÖRLÉS</div>

6.10. ábra. Rendezvények/Tanárok admin nézet

Adatok bevitele

A rendezvények és tanárok létrehozásakor az adatok validálva vannak szerver oldalon, ez segít abban, hogy a kliens oldalról a megfelelő adatok érkezzenek a szerverre. Így tudjuk megadni, hogy melyek azok a mezők amik kötelezőek vagy éppen opcionálisak, illetve bizonyos mezők mennyire felelnek meg a kért adatoknak. Az időpont dátum kell legyen, a feltöltött fájl kép legyen és ne más kiterjesztésű fájl, a mérete maximum 4 megabájt lehet.

```
$request->validate([
    'nev' => 'required',
    'idopont' => 'required|date',
    'helyszin' => 'required',
    'resztvevok' => 'nullable',
    'tipus' => 'required',
    'leiras' => 'required',
    'kepek.*' => 'image|max:4096',
], [
    'kepek.*.image' => 'Csak képfájlt lehet feltölteni!',
]);
```

6.11. ábra. Validálás

Ha nem megfelelőek az adatok, a szerver hibaüzenetként visszaküldi azokat a mezőket, amik nem tartották be a validálás szabályait, ezek az üzenetek a felhasználó számára megjelennek a megfelelő mezők alatt.

Rendezvény típus:

Rendezvény típus

A(z) típus megadása kötelező!

Rendezvény képek:

Choose Files

No file chosen




Csak képfájlt lehet feltölteni!

6.12. ábra. Validálási hibaüzenetek

6.1.5. Tanárok oldala

A tanszék tanárainak is létezik egy külön oldal, ahol szintén lehet keresni, név, email cím, és titulus szerint. Ezekre szintén rá lehet külön-külön kattintani és megjelenik a részletes oldaluk. Itt megjelennek a tanár adatai és a rendezvények, amiken részt vett, az adminisztrátor pedig módosítani tudja ezeket. Ez hasonló módon működik mint a rendezvények.

A tanár törlésénél automatikusan törlődnek a hozzá kapcsolódó rendezvények, ez fordított esetben is ugyanígy történik.

Tanárok		
<input type="text" value="Keresés..."/>		
	Dr. Waino Abshire II delphia40@example.org	tanársegéd
	Carolyn Frami II ethel.kozey@example.org	tanársegéd
	Hadley Jerde ikonopelski@example.com	emeritus professzor

6.13. ábra. Tanárok listája

6.1.6. Profil

A menüből elérhető a profil, a felhasználók át tudják írni a nevüket, a titulusukat illetve új avatart tudnak feltölteni, az email cím biztonsági okokból csak az adminisztrátor által változtatható meg.

Bakos Csongor

Név*

Bakos Csongor


Email*

csongji@test.com

Pozíció

diák

Avatar



Choose File

No file chosen

KÜLDÉS

6.14. ábra. Saját profil módosítása

Összefoglaló

6.2. Következtetések

A Matematika-Informatika Tanszék számára kifejlesztett eseménykezelő rendszer segítségével a szakdolgozat törekszik a koordináció, a részvétel és a kommunikáció javítására az egyetemi tagok között. A tervezett weboldal intuitív platformot kínál az adminisztrátoroknak az események zökkenőmentes kezeléséhez, míg a tanároknak lehetőséget nyújt a részvételi előzményeik egyszerű nyomon követésére és a közelgő tevékenységekhez történő tájékozódásra. Ez a központi megközelítés az eseménykezelésben hatékony tervezést, pontos nyilvántartást eredményez és végső soron hozzájárul egy összetartó és elkötelezett akadémiai közösség kialakításához a tanszéken belül.

6.3. Továbbfejlesztési lehetőségek

Továbbfejlesztési lehetőségként meg lehet említeni egy értesítés rendszert ami a felhasználókat értesítené az események változásáról, akár email, akár push értesítések formájában. Hasonló módon lehetne készíteni egy hírfolyamot, ahol a tanszék tanárait lehetne az eseményekhez köthető fontos információkról értesíteni.

Továbbá a rendezvények összköltségéről is lehetne tartani egy nyilvántartást, így lehetne követni a tanszék eseményekre szánt költségvetését.

A későbbiekben akár publikussá is lehetne tenni az oldalt, hogy a diákok is láthassák az eseményeket, ennek feltétele, hogy bizonyos adatok előttük rejtve maradjanak. Egy rendszert is lehetne fejleszteni ami a jogosultságokat kezelné, több típusú felhasználó létrehozásával.

A rendezvényen résztvevők számára a képek feltöltését is publikussá lehetne tenni, így megkönnyíteni az adminisztrátorok dolgát, akiknek feladatuk, hogy ezeket összegyűjtsék különböző forrásból.

Forráskód

Az alkalmazás forráskódja elérhető az alábbi linken: <https://github.com/bcsa/sapi-tanszek-web>

Ábrák jegyzéke

4.1. Használati eset diagram.	19
5.1. Rendszer architektúra diagram.	23
5.2. Adatbázis egyed-kapcsolat modell.	25
5.3. Wireframe.	25
6.1. Laravel útvonalak.	26
6.2. IsAdmin middleware.	27
6.3. Bejelentkezés/Jelszóemlékeztető űrlap.	28
6.4. Rendezvények listája	29
6.5. Lapszámozás	30
6.6. Kereső	31
6.7. Rendezvény részletei, módosítása, résztvevő tanárok hozzáadása/törlése	31
6.8. if függvény Blade-ben	32
6.9. Tanár-Rendezvény kapcsolat váltása	32
6.10. Rendezvények/Tanárok admin nézet	32
6.11. Validálás	33
6.12. Validálási hibaüzenetek	33
6.13. Tanárok listája	34
6.14. Saját profil módosítása	35

Irodalomjegyzék

- [1] M. Bean, *Laravel 5 Essentials: Explore the Fundamentals of Laravel, One of the Most Expressive and Robust PHP Frameworks Available*, Community experience distilled, Packt Publishing, 2015, ISBN: 9781785283017, URL: <https://books.google.ro/books?id=LBBwrgEACAAJ>.
- [2] *Apache HTTP Server hivatalos weboldala*, <https://httpd.apache.org/>, 1995, (elérés dátuma 2023. 06. 10.).
- [3] *Axios hivatalos dokumentáció*, <https://axios-http.com/docs/intro>, 2014, (elérés dátuma 2023. 06. 10.).
- [4] *Bootstrap hivatalos weboldala*, <https://getbootstrap.com/>, 2011, (elérés dátuma 2023. 06. 10.).
- [5] *Docker hivatalos weboldala*, <https://www.docker.com/>, 2013, (elérés dátuma 2023. 06. 10.).
- [6] *Eventbrite weboldala*, <https://www.eventbrite.com/>, 2008, (elérés dátuma 2023. 06. 10.).
- [7] *Eventzilla weboldala*, <https://www.eventzilla.net/us/home>, 2009, (elérés dátuma 2023. 06. 10.).
- [8] *Git hivatalos weboldala*, <https://git-scm.com/>, 2005, (elérés dátuma 2023. 06. 10.).
- [9] *GitHub hivatalos weboldala*, <https://github.com/>, 2008, (elérés dátuma 2023. 06. 10.).
- [10] *Laravel Blade dokumentáció*, <https://laravel.com/docs/10.x/blade>, 2011, (elérés dátuma 2023. 06. 10.).
- [11] *Laravel 10.x dokumentáció*, <https://laravel.com/docs/10.x/>, 2023, (elérés dátuma 2023. 06. 10.).
- [12] *Laravel hivatalos weboldala*, <https://laravel.com/>, 2011, (elérés dátuma 2023. 06. 10.).
- [13] *MariaDB hivatalos weboldala*, <https://mariadb.org/>, 2009, (elérés dátuma 2023. 06. 10.).
- [14] *Meetup weboldala*, <https://www.meetup.com/>, 2002, (elérés dátuma 2023. 06. 10.).
- [15] *PHP: Hypertext Preprocessor hivatalos weboldala*, <https://www.php.net/>, 1995, (elérés dátuma 2023. 06. 10.).

- [16] *PhpStorm hivatalos weboldala*, <https://www.jetbrains.com/phpstorm/>, 2009, (elérés dátuma 2023. 06. 10.).
- [17] *REST API wikipédia oldala*, https://en.wikipedia.org/wiki/Representational_state_transfer, 2000, (elérés dátuma 2023. 06. 10.).
- [18] *Vue.js hivatalos weboldala*, <https://vuejs.org/>, 2014, (elérés dátuma 2023. 06. 10.).
- [19] *Vuetify hivatalos weboldala*, <https://vuetifyjs.com/en/>, 2016, (elérés dátuma 2023. 06. 10.).
- [20] *Ziggy hivatalos forráskód*, <https://github.com/tighten/ziggy>, 2017, (elérés dátuma 2023. 06. 10.).