

LovassyApp – Tanulást ösztönző rendszer

NemCsoport

Ocskó Nándor

Felkészítő: Báder Anikó

Lovassy László Gimnázium, 8200 Veszprém, Cserhát ltp. 11.

1. Bevezetés

A LovassyApp egy tanulást ösztönző webes szoftver, amely az iskolám egy korábban nem digitális rendszertét váltja le. Ennek a rendszernek a lényege az, hogy az iskola diákjai az órákon szerzett jegyeiket egy iskolai valutára válthatják be (ez esetünkben a LovassyLóvé, későbbiekben: LoLó) és a beváltott pénzüikkel a diákok különböző, tanárok által felajánlott, termékeket vásárolhatnak, majd használhatnak fel órán vagy órán kívül. Ilyen termék például a „kimentés”, amely egy órán felhasználható termék, a felajánló tanár egy óráján mentesíti a felhasználó diákot a be nem jelentett dolgozatok és felelések alól. Egy másik termék például a „zene az iskolarádióban”, amely egy órán kívül felhasználható termék és technikai feltételei közé tartozik, hogy beváltáskor a diák megad hasson különböző értékeket bemeneti mezőbe (ebben az esetben egy ilyen mező a zeneszám címe).

Egy másik fontos funkciója az alkalmazásnak a szavazó rendszer, amely szintén egy korábbi iskolai rendszert vált le, és lehetővé teszi a diákoknak és a tanároknak, hogy iskolai eseményeken képszavazásokban tudjanak részt venni.

Bár maga az alkalmazás a Lovassy László Gimnázium diákjainak készült, de az is fontos volt, hogy konfiguráción keresztül az alkalmazás adaptálható legyen új követelményekhez, a LoLó rendszer változásaihoz és más iskolák sajátos követelményeihez is.

2. A probléma megoldásának menete

2.1. A kitűzött célok

Fontos célkitűzésem volt, hogy a jegyek külön munka nélkül, vagy minimális külön munkával jussanak el az adatbázisába egy megbízható forrásból, valamint az alkalmazás legyen minden eszközön jól használható, és a rendszerrel visszaélő felhasználók legyenek kitilthatók, valamint csak az iskola diákjai tudjanak regisztrálni. Mindezek mellett lehetővé akartam tenni, hogy bárki saját alkalmazásokat tudjon fejleszteni, amelyek a LovassyApp funkcióit valósítják meg (a központi iskolai rendszert használva) vagy azzal

összhangban működnek. Végül pedig azt is fontosnak tartottam, hogy az alkalmazás minimális emberi beavatkozással és költségek nélkül működjön, karbantartást lehetőség szerint ne igényeljen, ezt el is értem.

2.2. A megvalósítás

A megvalósítás modulárisan történt. Minden funkció egy funkciós csoportba tartozik, és a funkciós csoportok egymástól teljesen független egységeket alkotnak egy monolitikus architektúra keretein belül. A funkciós csoportok mind egyenként ki- és bekapcsolhatók és a köztük történő kommunikáció egy Event-Listener megoldással történik.

Az első ilyen funkciós csoport az autentikációért és autorizációért felelős csoport. Ez foglalja magában a belépést, a jogosultságok kezelését, az e-mailes megerősítést és a jelszó visszaállítását is. A rendszer csak @lovassy.edu.hu-ra végződő email című diákokat enged regisztrálni és amikor valaki regisztrál, automatikusan bekerül a „Default” felhasználói csoportba és megkapja az ezzel járó jogosultságokat is. A rendszerrel visszaélőket is ezen felhasználói csoport megváltoztatásával lehet kitiltani.



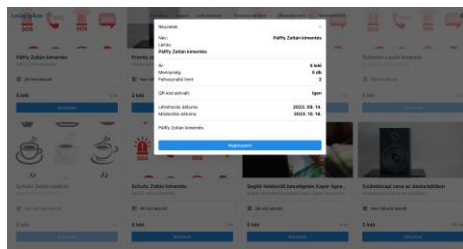
4. ábra: A jogosultságok kezelése

A második ilyen modul a szavazásokért felelős modul. Ez foglalja magában a képszavazások létrehozását, kezelését... Egy szavazás lehet inkrementáló (tetszik/nemetszik) típusú vagy választásos (az összes közül csak egy kép választható). Megszabható az is, hogy egy ember hány képet tölthet fel, mely felhasználói csoportok tölthetnek fel képeket, meg legyen-e jelenítve a diák neve és osztálya is a beküldött képek mellett és még sok más.

A harmadik ilyen modul a jegyek importálásáért felelős. Ez foglalja magában az „import kulcsok” kezelését (ilyen birtokában lehet csak jegyeket feltölteni), valamint ehhez tartozik egy asztali alkalmazás is, amely feldolgozza a jegyeket és titkosítva elküldi a szerveren futó alkalmazásnak. Minden héten egyszer az iskolánk igazgatóhelyettese tölti fel a krétából kimenthető összes jegyet tartalmazó Excel táblázatot. Korábban ez a rendszer a Kréta API-t használta és így nem is volt szükség emberi interakcióra a jegyek

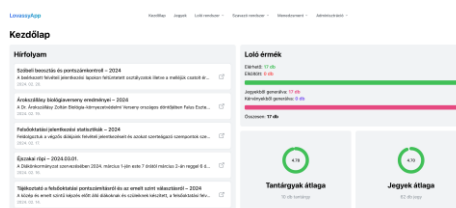
megszerzéséhez, de sajnos a Kréta egy kérésemre válaszul nem engedélyezte ezt, így ezt le kellett cserélnem a jelenlegi rendszerre.

Következő nagyobb funkciós csoport az a bolt rendszer. Ez foglalja magában a „bazárt” és a „kincstárt” (minden felhasználó megvásárolt termékeit) és ezen keresztül lehet termékeket létrehozni, kezelni és beváltani, valamint kérvényeket kezelni, ugyanis nem csak jegyekért, hanem versenyeredményekért is lehet LoLót kapni, és ehhez a diáknak egy kérvényt kell benyújtania. A termékek lehetnek órán aktiválhatók, ekkor a diáknak egy QR kódot kell beszkenneálnie az órán, amely egyedi és a tanárnál van kinyomtatva, és lehetnek nem órán aktiváltak, ekkor csak egy gombot kell megnyomniuk. A termékek tartalmazhatnak még bemeneti mezőket is megadható az is, hogy egy ember hányat vehet az adott termékből.



2. ábra: A bazár

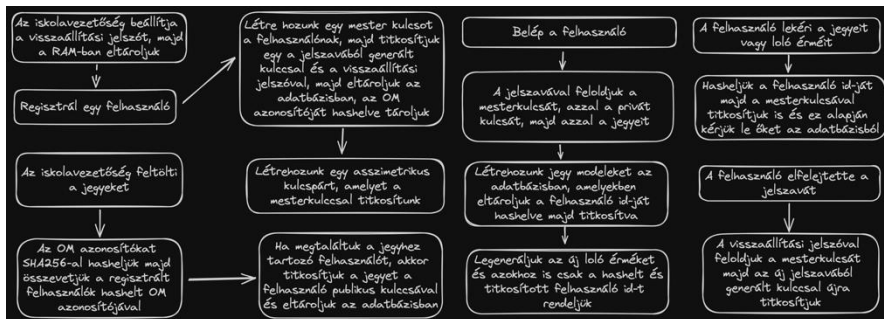
Végül pedig a kisebb modulok: a felhasználók modul, a hírfolyam, az „iskolai” modul és a státusz modul. Ezek felelősek a felhasználók kezeléséért, regisztrációért, az iskola híreinek jelzéséért, azért, hogy a diákok meg tudják nézni a jegyeiket az alkalmazásban és a rendszer állapotának kijelzéséért is.



3. ábra: A kezdőlap, ahol láthatók az iskola hírei

A LovassyAppnak sok biztonsági kikötésnek is meg kellett felelnie, így egy extenzív biztonsági rendszert is tartalmaz. Elvárás volt, hogy a diákok jegyeit senki se lássa, még az sem, akinek hozzáférése van az adatbázishoz, valamint, hogy a jegvek titkosíthatatlanul soha ne hagyják el az

igazgatóhelyettesünk számítógépét (és így is csak a regisztrált felhasználók jegyei). A rendszer működését az alábbi ábra mutatja:



4. ábra: A biztonságért felelős folyamatok rendszere

2.3. A felhasznált technológiák

Mivel költségmentesen akartam megvalósítani az alkalmazást, az iskola erőforrásaival kellett dolgoznom, és ez mindössze egy nagyon gyenge régi linux szervert jelent, emiatt sok teljesítmény optimalizáció is kellett.

Az alkalmazás központját, a szerveren futó részét ehhez igazítva ASP.NET Core keretrendszerben írtam meg, mivel elképesztően gyors nagyon kevés erőforrás mellett is, és websocketeken keresztül lehetővé tette, hogy minden, ami az oldalon látható élőben (újra töltés nélkül) frissüljön. A külső alkalmazásokat támogatva ez egy publikus API-t tesz elérhetővé.

A webes felületet React keretrendszerben írtam TypeScript segítségével és az asztali alkalmazást Tauri és React használatával írtam meg.

Az egész alkalmazás teljes mértékben nyílt forráskódú, forráskódja elérhető itt: <https://github.com/LovassyApp/LovassyApp>

3. Elért eredmények

Az alkalmazás nyújtotta előnyök már az első hetekben elérték, hogy a diákság több mint 90%-a regisztráljon, és a LoLó rendszert az alkalmazás előtti évek 5-6 fős felhasználói táboráról egy jelenleg 537 fősre bővítette, ez által is motiválva a diákokat a tanulásra. Az iskolánkon túlmutató potenciálját az is bizonyítja, hogy már a fejlesztés alatt is érkezett kérés (a Pannon Egyetemtről), hogy csináljuk meg a Neptun-hoz is, hogy egyetemek is használhassák.

Néhány statisztika az alkalmazás kihasználtságáról: 105 terméket vásároltak az idei tanévben, ebből 87-et fel is használtak, és 3597 LoLót gyűjtöttek a diákok (3 ötös/5 négyes = 1 LoLó), és 56 tanár ajánlott fel termékeket.