**ChoresTracker**

<http://chorestracker.infinityfreeapp.com/chores_tracker/>

**Bevezetés, a téma ismertetése, témaválasztás indoklása**

A Chorestracker egy olyan szoftver, amely célja, hogy megkönnyítse a házimunkával kapcsolatos szervezési problémákat a család számára.

Ennek érdekében lehetőséget biztosít arra, hogy egy megbízott személy felügyelje és koordinálja a feladatokat. Természetesen ez a jog az anyát illeti meg, aki általában a háztartás szervezője.

A regisztráció során az admin jogosultságot kap, amely lehetővé teszi számára, hogy további tagokat hozzon létre és kezeljen.

Ez lehetővé teszi számára, hogy hatékonyan oszthassa szét a feladatokat és delegálja a felelősségeket a családtagok között.

Ebben a fejezetben szeretnénk egy kicsit írni a projekt születésének részleteiről.

Az alapötlet munkában használt jegykezelő rendszer adta. Az IT support területet vettük alapul és ott szükség van a megoltdott problémák dokumentálására, adminisztrációjára és a megoldási folyamat részletes követésére.

Ezt ültettük át egy másik kontextusba és egy otthon használt applikációt hoztunk létre. Beleképzeltük egy egy nagyon elfoglalt soktagú család helyzetébe ahol egy applikáció segíthet a ház körüli teendők ellátásának koordinálásában.

Így született meg a ChoresTracker egy elsődlegesen böngészőben használható program formájában. Az applikáció lehetőséget biztosít arra, hogy egy ház körüli teendőt, létrehozzanak, új részfeladattal lássanak el, vagy egyéb különböző fontos információkkal lássanak el a feladat elvégzése előtt, közben és természetesen után.

A következő fejezetekben szemléltetjük részletesen a fejlesztői és felhasználói dokumentációkat.

**Fejlesztői dokumentáció**

Ebben a fejezetben szeretnénk részletes képet adni a fejleszés menetérő és a felhasznált technológiákról amelyeket meg kellett tanulnunk és használnunk ahhoz, hogy a projekt elérje a jelenlegi formáját.

**Felhasznált technológiák**

A tervezés ideje alatt Figma-t használtunk, ez manapság egy népszerű ingyenesen is elérhető web-applikáció a web designer-ek körében. Ebben próbálkattuk először, hogyan nézne ki szépen és logikusan az elrendezés.

Ekkor még a HTML-be, CSS- be JavaScript-be kezdtük átültetni a terveinket, de hamar rájöttünk, hogy a technológia amit végül választottunk az a PHP lesz.

Frontend:

PHP, HTML, Javascript, Tailwind

Backend:

PHP, MySQL

A fejlesztéshez használt szoftverek

Visual Studio Code

XAMPP

Google Chrome

**Adatbázis szerkezet**

Az alkalmazás 1 MySQL adatbázist használ.

Az adatbázis 4 táblát használ: chores, comments, users, groups

chores / a feladatok tárolására /

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generatedcomments / a kommentek tárolására/

A screenshot of a computer

Description automatically generatedusers / a felhasználok tárolására /

A screenshot of a computer

Description automatically generatedgroups / a csoportok tárolására /

Az táblákat kezelését(CRU) az app/data/data.class.php ból initializált Data végzi,

Ami a MysqlDataProvider.class metódusait hívja meg.

A MysqlDataProvider.class PDO segítségével végzi az adatbázis műveleteket.

Könyvtár és fájlszerkezet

Az alkalmazás MVC (model-view-controller) minta alapján lett kialakítva.

A nézet fájlok a views mappában találhatók, az irányítók a felhasználóhoz

a root-ban, az adminhoz az admin mappában. Hogy biztosítsuk, egyes felhasználók

nem férnek hozzá az admin felületekhez és be tudjanak lépni a nekik szánt felületre

session php funkciókat használunk.

Az app mappa tartalmazza a config.php-t (db confighoz), app.php-t

(a szükséges fájlok betöltéhez), functions.php-t(a funkciók szeparált tárolásához)

És a data mappát, ahol az adatbázis műveletekhez szükséges osztályokat definiáló

fájlok találhatók.

**Felület kinézete, formázása**

Az oldalak keretét (header,footer,) a layout.view.php adja, amin belül

jelenik meg az adott view.php

A HTML oldalak dizájnjának kialakításához Tailwind css-t használtunk.

Ennek oka, hogy ne keljen külön css fájlokat létrehozni, azokban keresgélni.

**Algoritmusok, kódok**

Az alkalmazás belépési pontja az „/index.php” .

A screenshot of a computer program

Description automatically generated

Itt ha a user authentikált(is\_user\_authenticated funkciót használunk, ha a

$\_SESSION['user\_logged\_in']

létezik, akkor authentikáltnak vesszük), a dashboard-ra kerül, ha nem a „start.view-ra”

A start.view-n login és signup menüpontok láthatók.

Signup

A screenshot of a computer program

Description automatically generated

Minden inputot „megszűrünk” , hogy biztonságos legyen az alkalmazás.

Ha valami hiányzik vagy már létező adat, figyelmeztetjük a user-t.

Ha minden rendben van, létrehozzuk a csoportot és a usert a db-ben.

echo-val meghívunk egy js scriptet, ami 3500 sec után átirányít a

login felületre.

A signup felületen csak adminok regisztrálhatnak.

A sima „családtagokat” az adminok hozzák létre.

Adminból 2 lehet egy csoportban, családtagból 9.

Login

A computer screen shot of a program code

Description automatically generated

Itt is minden inputot megszűrünk.

Ha helyesek az adatok, beléptetjük a usert családtag

Vagy admin felületre.

Ha valami nem helyes, figyelmeztetjük.

**Admin felület**

Itt jelennek meg a létrehozott feladatok, létre tudunk hozni

admint, családtagot, feladatot és törölni is tudjuk őket.

A screen shot of a computer screen

Description automatically generated

Két foreach looppal előkészítjük az adminok és a

Családtagok email címet, hogy a view-ban meg tudjuk jeleníteni őket.

A screen shot of a computer program

Description automatically generated

Post metódussal tudunk új usert létrehozni vagy törölni.

A create\_user létrehozza a felhasználót a user táblában az insert\_member\_into\_group

Pedig felviszi a megfelelő helyre a groups táblában.

A remove\_user eltávolítja az adott usert az users és groups táblából.

Az utolsó admin nem tudja magát eltávolítani.

A feladatok létrehozása szintén post metódussal történik,

ugyanezen a nézeten. Az adatok szanitizációja után az add\_chore

metódus helyezi el a feladatot a chores táblában.

A feladatok mellett az edit-re kattinta van lehetőségünk szerkezteni az

adott feladatot.

Az adatok megváltoztatása után az update\_chores\_as\_admin

metódus végzi módosításokat a chores táblában.

Lehetőségünk van törölni a feladatot és kommentet írni hozzá.

(delete\_chore, add\_comment)

Tesztelés során jöttünk rá, hogy a comment hozzáadása és

A változtatások mentése a jelenlegi kód struktúrával egy időben

nem működik.

Családtag felület

Családtag(nem admin) esetében login után

A dashboard nézetre kerülünk, amit a dashboard.php irányít.

Itt megjelennek a user feladatai.

Ha a done-ra kattint, a quick\_done metódus elvégzettre állítja a

feladatot. A másik lehetőség, hogy megnyitja a részleteket.

Itt az update\_chore metódus frissíti a feladatot az adatbázisban.

**Tesztelés**

A tesztelés célja, az hogy részletesen átnézzük, pontosan mik az adott funkciók pontos reakciói. Ezen dokumentáció során szeretnénk erről egy részletes képet adni.

Mivel ez egy iskolai projekt, a domain ami kiszolgáltatja az oldalt, egy ingyenes szolgáltatás. A funkciók úgy működnek, mint amikor lokálisan vannak a forrásfájlok tárolva.

A következőkben a különböző funkciókról szeretnénk részletesen írni.

Az applikáció URL-je:

<http://chorestracker.infinityfreeapp.com/chores_tracker/>

Itt a fő cél az, hogy az applikáció sikeresen megnyílik egy webböngészőben. Az eredmény az, hogy böngésző mivoltától függetlenül megnyílik az oldal. Kipróbáltuk Google Chrome-ban (ez volt az elsődleges böngésző), Microsoft EDGE-ben és Mozilla Firefox-ban ezetúl Android-om is Google Chrome-ban. Az említett platformokon elfogadható sebességgel megnyílik az applikácó. Ezen eredmények a további tesztesetekre is jellemzőek.

Amikor a SIGN UP oldalra lépünk, egyértelműen a felhasználó dokumentációban is részletezett lehetőségek tárulnak elénk. A név szempontjából nincs szigorú megkötés, a jelszó szempontjából sem(ezt lehet a későbbiekben szigorítani), viszont az email címnél fontos, hogy a formátuma a bevitt email címnek megfelelő legyen, különben sikeresen megkapjuk a várt felszólítást, hogy az email címnek @ karaktert kell tartalmaznia.

A LOGIN oldalesetében szintén az email formátuma egy fontos eset, amire figyelni kell, hogy nem megfelelő és természetesen nem regisztrált email cím és jelszó nem kerül elfogadásra. Itt különbséget kell tenni a helytelen email cím esetében:

„Please enter a valid email address” - amikor az email formátuma nem megfelelő

“The provided credentials are not correct” – amikor a formátuma az email címnek megfelelő, de nem egyeznek az adok regisztrált értékekkel.

A továbbiakban nézzük meg az adminként beregisztrált felület eredményeit. (DASHBOARD)

A az első fontos eredmény az ADMIN-ok és MEMBER-ek esetében kell meghatároznunk, méghozzá az adminok és felhasználók számának behatárolása. Két admin után a lehetőség újabb admin hozzáadására sikeresen megszűnik. A felhasználók esetében (MEMBER) ez a szám kilenc után szűnik meg, ez után nem lehetséges további tagokat hozzáadni. (az ADD MEMBER és értelemszerűen a REMOVE MEMBER gombokkal).

Az ALL CHORES szekcióban láthatjuk megjelent regisztrált feladatok eredményeit. Ezekről részletesebben egy későbbi bekezdésben írnánk.

A következők ben a CREATE CHORE oldalról szeretnénk írni. Itt szabadon tudunk megadni adatokat egy bizonyos feladatról, cím is opcionális, bár logikus megadni, az email cím csak a meglévők közül lehetséges és határidő és leírás sem fontos, hogy meg legyen adva. Amint elmentettük teljesen lehet a részleteket adminként változtatni ez nagyon előnyös az admin szempontjából, viszont a megbízott szempontjából kevésbé. Itt végül is a cél a fontos, mindkét fél számára fontos, hogy jó legyen a kommunikáció, de mégis ad egy flexibilis mozgásteret.

Itt ragadnánk meg az alkalmat, hogy írjunk a meglévő feladatok részleteinek eseteiről.

Az EDIT CHORE oldalon megmarada a flexibilitás a feladat részleteinek módosítására és a COMMENTS szekció teret ad egy részletes folyamat és kommunikáció dokumentálására ez alapján lehet a későbbiekben információt gyűjteni és tanulni egy feladat egyszerűbbé tételéről.

A továbbiakban nézzük meg a MEMBER jogosultság esetleges kevésbé flexibilis helyzetét.

A MEMBER DASBOARDON egy kicsit átláthatóbbak a feladatok, mivel itt kettéoszlik meglévő és elvégzett feladatokra. Ez azért lehet előnyös a megbízott számára, mert az ADMIN abból indul ki, hogy a megbízott igenis foglalkozik a rábízott feladatokkal és az elvégzett feladatok listáját elkülönítve láthatja.

A MEMBER EDIT CHORE oldalán logikusan van módosíthatóvá téve a meglévő részletek listája, és a részletes és flexibilis kommunikáció megbízott oldalról is kitűnően lehetséges. Ami a hozzászólások(COMMENT) számát illeti nincs limit amennyi az adatbázisban elfér, lehetséges..

Ami még a kijekentjkezét illeti, egyszerű LOGOUT gombbal van erre lehetőség és ezután akár ADMIN akár MEMBER, megfelelően működik és újra authentikációt és belépést igényel ha ezt az opciót választjuk.

Utolsó témáink egyike lenne a még a további statikus oldalak megjelenítése, névszerint:

ABOUT, CONTACT, TERMS OF USE és PRIVACY POLICY. Ezek informatív fukcióikat töltik be sikeresen.

Még egy paragrafusban a SETUPról írnánk egy kicsit:

Ez sikeresen teszi lehetővé a továbbfeljesztést, mivel lokálisan van ennek az oldalnak funkciója, ezáltal könnyen létrehozhatjuk az adatbázisunkat localhost-on.

Legutolsó sorban a reszponzívitásról írnánk, igyekeztünk minden oldalon a lehető legjobban láthatóvá tenni a legalább 340px szélességnél is az oldal tartalmát.

**Felhasználói dokumentáció**

A Chorestracker egy olyan szoftver, amely célja, hogy megkönnyítse a házimunkával kapcsolatos szervezési problémákat a család számára.

Ennek érdekében lehetőséget biztosít arra, hogy egy megbízott személy felügyelje és koordinálja a feladatokat. Természetesen ez a jog az anyát illeti meg, aki általában a háztartás szervezője.

A regisztráció során az admin jogosultságot kap, amely lehetővé teszi számára, hogy további tagokat hozzon létre és kezeljen.

Ez lehetővé teszi számára, hogy hatékonyan oszthassa szét a feladatokat és delegálja a felelősségeket a családtagok között.

Az applikáció angol nyelven íródott, ez a dokumentáció igyekszik részletesen bemutatni a funkciókat a felhasználó számára és ezen fejezet elolvasása megmagyarázza az össze funkciót magyar nyelven.

Kétféle felhasználó lehetséges az alkalmazásban, a kiterjesztett jogokkal rendelkező, továbbiakban ADMIN és az általános jogokkal rendelkező MEMBER.

Az applikáció megnyitása a következő internetes linken érhető el és a Landing Page/Főoldal

-ra vezet:

<http://chorestracker.infinityfreeapp.com/chores_tracker/>

**Landing Page/Főoldal:**

A főoldalon több lehetőség tárul elénk, itt a két legfontosabb a LOGIN (bejelentkezés) és SIGN UP(regisztráció) gombok. Ezekről később többet.

Ezentúl az ABOUT/ABOUT US (rólunk) gomb elvezet egy információs lapra a fejlesztőkről. A CONTACT(kapcsolat) gomb elvezet egy oldalra ami további információt ad a fejlesztők elérhetőségéről (jelnleg ez az iskolánk címét mutatja az iskolai email címekre, mivel egy ez iskolai vizsgaprojekt).

A TERMS AND CONDITIONS és PRIVACY POLICY (Felhasználási feltételek/adatvédelem) is fontos az oldal számára, ha élő környezetben használjuk(jelenleg csak példaként szolgál).

A SETUP gomb a fejlesztőknek fontos, felhasználói szempontból, nem jelentős, kérjük ezt a gombot ne használják, a megfelelő feltételek nélkül nem működőképes.

Visszatérés a főoldalra a ’ChoresTracker’ logó ikonjára való kattintással lehetséges.

**SIGN UP(regisztráció)**

Ha az oldadon a felhasználó a **SIGN UP** gombra kattint, akkor egy űrlap jelenik meg számára, ahol meg kell adni a Nevet (Name), az Email címét (Email) és a jelszavát kétszer (másodszor megerősítésként), valamint az elküldés (Submit) gombra kattintva elküldheti a regisztrációt.

Ezen az oldalon szükséges a megfelelő email cím formátumot használni, különben a felhasználó kap egy hibaüzenetet és egy felszólítást helyes email cím formátum használatára ([nev@domain.hu](mailto:nev@domain.hu)).

A jeszó kétszeri megadása gépelési hibák elkerülése végett szükséges.

Ezután a SUBMIT(beküldés) gomb megnyomásával lehet a regisztrációt elküldeni/megerősíteni.

A további lépések csak sikeres regisztráció során lehetségesek. Fontos megjegyezni, hogy az új profil létrehozása automatikusan ADMIN jogosultságot ad a felhasználónak.

A sikeres regisztráció utan az ADMIN átkerül a bejelentkezési oldalra(LOGIN).

**LOGIN(bejelentkezés)**

A sikeres regisztráció után lehetősége van az ADMIN-nak a bejelentkezésre.

Fontos megjegyezni, hogy a regisztrált email címmel (EMAIL) lehet belépni nem a megadott névvel.

A megadott jelszó megjegyzése fontos, mert a továbbiakban ezzel lesz a felhasználó hitelesítve.

A MEMBER jogosultsgú felhasználók is tudják ezt az oldalt használni, mivel az ő profiljukat az ADMIN fogja létrehozni.

**ADMIN DASHBOARD**

Ezen az oldalon adatik meg a lehetőség a ház körüli feladatokkal kapcsolatos minden nemű művelet elvégzésére.

Az eddigi alapvető informatív gombon kívül megjelennek a fontos funkciók gombjai.

A legfontosabb az első gomb a CREATE CHORE (ház körűli feladat létrehozása). Itt van lehetőség arra, létrehozzunk egy feladatot.

Az ADD MEMBER gomb megnyomásával ugrik fel egy kis ablak ahol hozzá lehet adni legfeljebb egy társ ADMIN-t vagy maximum 9 tagot (MEMBER).

Fontos, hogy az ADMIN-nak kell megadni az új tagok belépési adatait. EMAIL NAME, PASSWORD(jhelszó) és ADMIN jog megadásával.

Természetesen lehetséges el is lehet távolítani a tagokat a REMOVE MEMBER gombbal és az érintett email cím megadásával.

A meglévő csoporttagok(Group Members) megjelenítése ezen az oldalon látható, továbbá a csoport aktuális és elvégzett feladatai.

Innen kétféle egyéni oldal érhető el. Az első a CHEATE CHORE(feladat létrehozása) és az EDIT(a meglévő feladat megtekintése és további műveletek elvégzése).

**CREATE CHORE**

Kitöltendő sorok ezen az oldalon a következőek(elsőször is amit a bal oldalon látunk):

TITLE – ez a feladat megnevezése

ASSIGNED TO – az a tag akire rá lesz bízva a feladat

DUE DATE – az elvégzendő feladat határideje

DESCRIPTION - a feladat részletes leírása

A SAVE gombbal lehet a feladatot elmenteni és ezzel elküldeni a megbízottnak.

A BACK TO DASHBOARD gombbal léphetünk vissza az ADMIN főoldalára.

**ADMIN EDIT(DETAILS)**

Ezen lehet egy bizonyos CHORE-ral mindennemű adminisztrációt elvégezni.

Látható vagy változtatható adatok

TITLE – az adott neve a feladatnak, ezt itt már nem lehet megváltoztatni.

CREATION DATE – a feladat létrehozásának pontos időpontja, ez sem változtatható elem.

ASSIGNED TO – a megbízott, ezt átadható egy másik tagnak ha szükséges

DUE DATE – a határidő ez változtatható, természetesen indokoltan.

STATUS – alapértelmezetten not started(nem elkezdett)

további státuszok: in progress(folyamatban), delay(késés), done(elkészült)

DESCRIPTION – ez a részletes leírás ami változtatható

Ez alatt a mentések változtatása lehetséges (SAVE CHANGES), továbbá itt lehet kitörölni a feladatot (DELETE CHORE)

COMMENT – ez az egyik fontos funkció, mivel itt lehet esetleges hozzászólásokat írni.

Ezek a hozzászóláksok a COMMENT-ben lesz látható.

**MEMBER DASBOARD**

Itt lehet a meglévő és elkészült feladatok megnézni MEMBER jogosultsággal.

**MEMBER EDIT(DETAILS)**

Ami itt különbség az adminhoz képest az hogy bizonyos mezők nem változtathatóak:

DUE DATE(határidő), DESCRIPTION(leírás),

Ami lehetséges:

STATUS változtatása

COMMENT hozzáadása

SAVE CHANGES(változtatások mentése) is lehetséges.

**LOGOUT**

Az oldal elhagyása előtt természetesen fontos lépés kijelentkezni.

Hasznos időtöltést kívánunk a ChoresTracker használata során!

**Összefoglalás**

Összefoglalásként szeretnénk írni tapasztalaltainkról a fejleszést kitevő előző pár hónapban. Mint első közös projektkünk sok új élmény ért minket. Természetesen sok akadály került elénk és sok olyan megoldást kívánó részfeladat ami sok munkát és újragondolást vett igénybe.

Többször kellt újragonodolnuk a projektet menet közben, valalmilyen szinten újrakezdeni. A tervezés a Figma szolgáltatás segítségével készült, nyilván a végső megvalósításban születtek máshogy kinéző megoldások. A folyatás a HTML CSS és Javasscipt segítségével egy idő után kicsit követhetetlenné tette a folyamatokat.

Amikor elkezdtük áltültetni PHP-ba a projektet elsőzör a technológi megismerése jelentette a nagy kihívást ezentúl a Talilwind CCS is egy újdonság volt, őszintén szólva kissé összezavaró amikor az ismert CSS elemeket is igyekezni kellett a könyvtár szabályai szerint értelmezni.

Továbbá teljes mértékben állythatjuk, hogy sokat tanultunk a fejlesztési folyamat során bár nem eleget, természetesen nem kizárt, hogy a későbbiekben igyekszünk az eredetileg eklképzelt funkciókat megvalósítani akár közösen is. Az itt szerzett tapasztalat hasznos lesz későbbi projekteink esetében is.

Az sem véletlen, hogy az applikáció angol nyelven íródott, hiszen akármilyen projekt amelely angolul kerül közzétételre, nagyobb számú közönséget tud megszólítani. Itt ismét említeném a multinacionális IT cégeket, ott elképzelhetetlen, hogy egy szoftverfejlesztőnek ne kelljen közös munkát végezni más nemzetisélghez tartozó kollégákkal.

Csapatunk tagjainak többynire ezen szempontokat tartják szem előtt a jövőben elképzelt „álommunkávát”-t tekintve is.