**ZÁRÓDOLGOZAT**

Készítették:

Nyenyestyán Bence – Szabó Dániel

Konzulens:

Horváth István

Miskolc

2025.

Miskolci SZC Kandó Kálmán Informatikai Technikum

Miskolci Szakképzési Centrum

**SZOFTVERFEJLESZTŐ- ÉS TESZTELŐ SZAK**

**Korházi kezelőfelület weboldal**

Egy online korházi kezelőfelület

Nyenyestyán Bence – Szabó Dániel

2024-2025

**Tartalomjegyzék**

Bevezetés 4

Programozási nyelvek és technológiák 5

MySQL adatbázis (Database)

Az adatbázis fogalma

Mi az SQL? (Strukturált lekérdező nyelv)

PHPMYADMIN és a XAMPP

HTML

CSS

Bootstrap

React (webes frontend)

Hogy működik a React?

Miért kedvelik a fejlesztők a Reactot?

Dokumentum Objektum Model (DOM) 1

Backend 1

ASP.NET

A backend feladata

Trello

A weboldal elképzelt váza:

Felhasználói dokumentáció

Főoldal

Termékek

Kosár

Regisztráció

Bejelentkezés

Rendelés

Tesztelés

Források:

Mellékletek

Bevezetés

Ezt a korházi kezelőfelület weboldalt elég könnyű volt kigondolni mivel Magyarországon országszerte baj van az egészségügyi ellátással ezért úgy gondoltuk megkönnyítjük az emberek és orvosok dolgát ezzel a weboldallal.

Az ötletet közösen gondoltuk ki és jó ötletnek tartottuk hogy egy egyszerű és használható dolgot csináljunk.

Ebben a dokumentációban megmutatjuk mivel dolgoztunk és hogyan készült el ez az egész lépésről lépésre.

4

Programozási nyelvek és technológiák

A mai világ elsődleges céljai között ott figyel az emberek egészsége, fentmaradása és a projektünk erre a témára épül fel.

Weboldalunkon az emberek egyszerűen és gyorsan tudnak időpontokat foglalni az internet segítségével, így az embereknek hatalmas segítség lehet az idejük beosztásában és a foglalások gyorsaságában.

Emellett korházunkban egy csapat jólképzett orvos várja betegeinket akik szél tudáskörökkel rendelkeznek és készenállnak minden ember ellátására.

**MySQL adatbázis (Database)**

Az adatbázis fogalma

Az adatbázis megtervezésével kezdtük a projektet, de mi is az adatbázis?

Az adatbázis strukturált információk vagy adatok szervezett gyűjteménye, amelyeket

jellemzően elektronikusan, számítógépes rendszerben tárolnak. Az adatbázisokat általában

adatbázis-kezelő rendszer (DBMS-az a rendszer mely összekapcsolja az adatbázist a

végfelhasználóval és kezelni tudják az adatokat benne) vezérli. Az adatokat és az adatbázis-

kezelő rendszert, valamint a hozzájuk kapcsolódó alkalmazásokat együttesen adatbázis-

rendszernek nevezzük, amit gyakran csak „adatbázis”-ként rövidítünk.

A ma használatos leggyakoribb adatbázis-típusokban az adatok jellemzően táblák soraiba és

oszlopaiba vannak rendezve, ami megkönnyíti az adatok feldolgozását és lekérdezését.

ezt használtuk az adatok tárolására a vizsgaremekben.



**Mi az SQL? (Strukturált lekérdező nyelv)**

Az SQL egy olyan programozási nyelv, amelyet szinte minden relációs adatbázis\*\* használ az

adatok lekérdezésére, kezelésére és meghatározására, valamint a hozzáférés vezérlésére. Az

SQL-t az IBM fejlesztette ki az 1970-es években az Oracle jelentős közreműködésével, ami az

SQL ANSI-szabvány bevezetéséhez vezetett, majd az SQL számos kiterjesztése követte olyan

vállalatoktól, mint az IBM, az Oracle és a Microsoft. Bár az SQL ma is széles körben

használatos, új programozási nyelvek is kezdenek megjelenni.

\*\* relációs adatbázis: A relációs adatbázisokban az elemek oszlopokkal és sorokkal

rendelkező táblákba vannak rendezve. A relációs adatbázisok technológiája biztosítja a

strukturált információk elérésének leghatékonyabb és legrugalmasabb módját. Az 1980-as

évektől kezdték meg uralkodásukat a piacon, mai napig is ez a legismertebb és

leghasználtabb típusa, ettől független rengeteg másik fajtája is van

(pl.: objektumorientált adatbázisok, elosztott adatbázisok, felhőalapú adatbázisok), de a mai

programozási piac és nyelvek a relációs adatbázissal dolgoznak túlnyomó részt. Ezért lett ez

az adatbázisunk.



**A mi SQL-ünk amit hasznátunk**

PHPMYADMIN és a XAMPP

Mi a projektünkhöz a PHPMYADMIN nevezetű adatbázis kezelő szoftvert használjuk. A

működéséhez szükségünk van a XAMPP nevezetű control panelre.

Mi az a XAMPP?

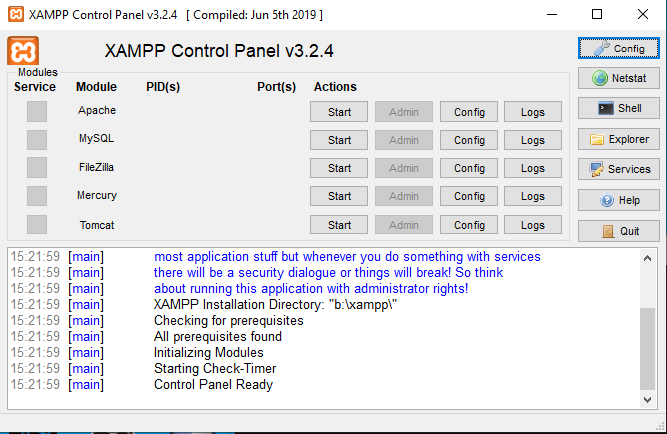
Mondhatnánk, hogy öt értelmetlen betű egymás mellé téve, de nem az.

A XAMPP egy ingyenes nyílt forráskódú webszerver szoftvercsomag, mely képes értelmezni a

PHP-t és főbb alkotóeleme az SQL. Enélkül a felületet nem tudjuk elérni, fontos hogy

startoljuk az Apache-t először, utána a MySQL-t, és az Admin gombra kattintva érhetjük el a

phpmyadmin felületét.



A PHPMYADMIN a XAMPP-hoz hasonlóan egy nyílt forráskódú, és ingyenes webes

adminisztrációs eszköz, mely az egyik legnépszerűbb ilyen alkalmazás, ami képes SQL

adatokat kezelni.

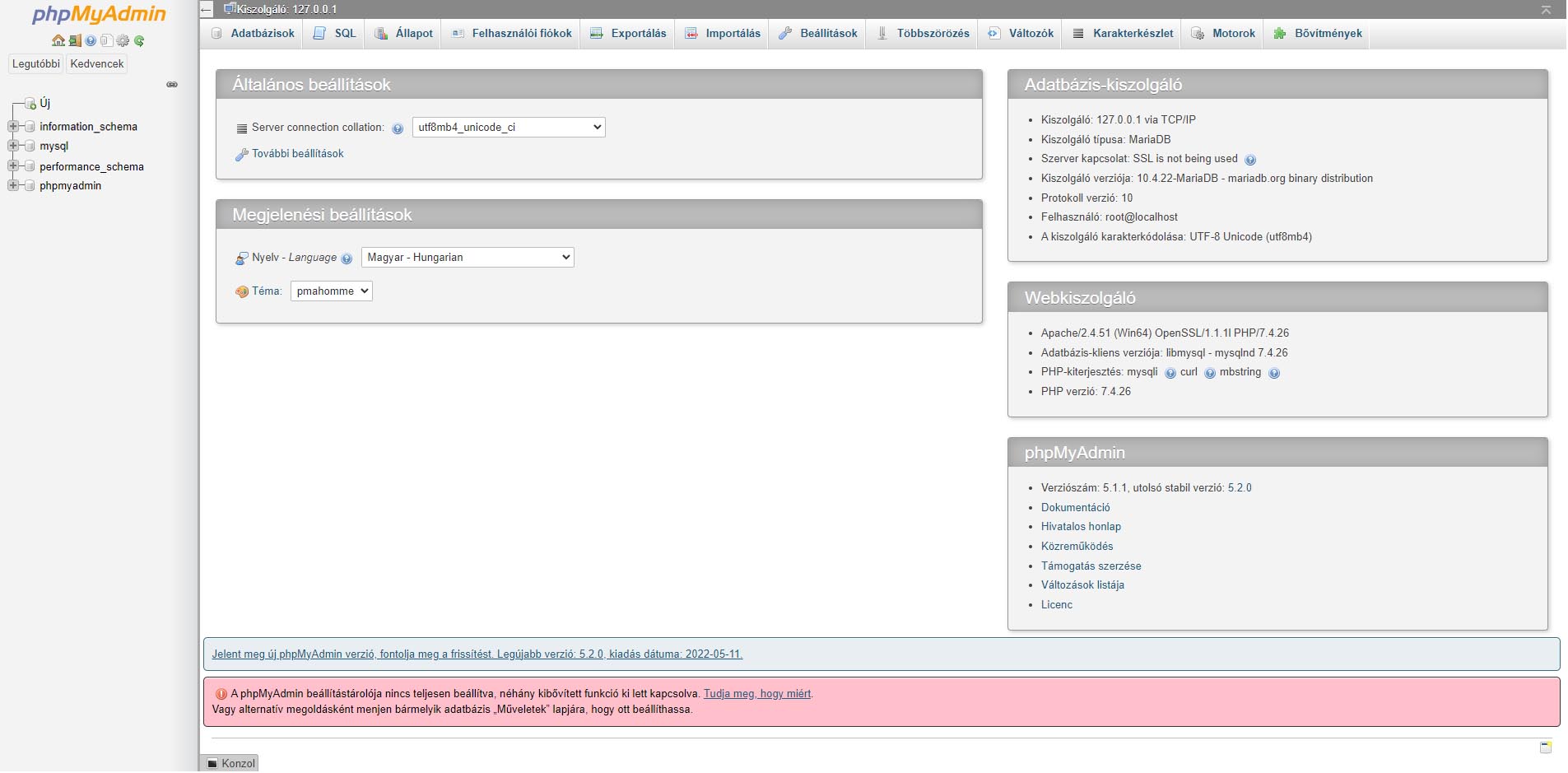
A felületen tudunk új adatbázist létrehozni, táblákat és listákat készíteni, fájlokat importálni

és exportálni különböző fájlokként, összetett lekérdezéseket tudunk végrehajtani,

adatkapcsolatokat és élő diagrammokat készíteni. SQL parancsokkal is végezhetjük ezeket,

de egyszerűbb „kattintgatós” módszerrel is használhatjuk a felületet, ennek köszönhetően

gyorsan és hatékonyan tudunk vele dolgozni.



**HTML**

A HTML (HyperText Markup Language) egy olyan jelölőnyelv, amelyet első sorban webhelyek

készítéshez használnak. Magyarul hiperszöveges jelölőnyelvet jelent.

A HTML-el egy webhely szerkezetét, illetve tartalmát szokás meghatározni. Létrehozhatók

vele például bekezdések, címsorok, táblázatok, elhelyezhetők képek, átméretezhetőek,

szövegeket lehet vele alakítani

A HTML dokumentumoknak több részre oszlanak, attól függően, hogy az melyik részre utal.

Az első zárójelben lévő kód kezdi, az utolsó pedig lezárja a parancsot.

**CSS**

Cascading Style Sheets (CSS) a HTML különböző elemeinek a kinézetét és formáit

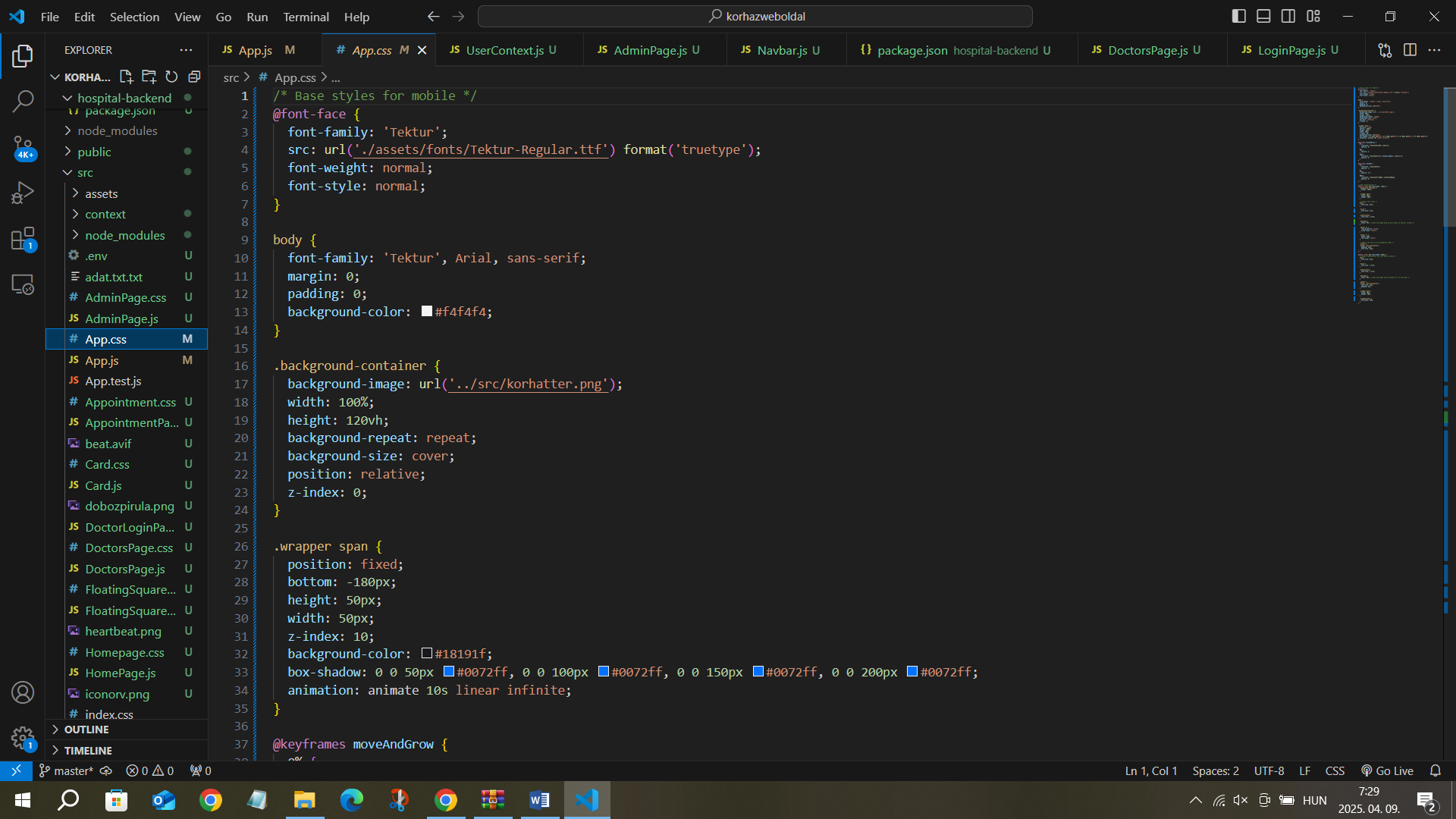
határozhatja meg, többek között a bekezdések színét változtathatjuk, tehát színezhetünk,

vagy éppen betűtípusokkal játszhatunk, képeket és betűket méretezhetünk, margókat

állíthatunk, lebegtethetünk, erősebbé vagy picit beolvadóbbá tehetjük a tartalmat a HTML-

en.

Ennél a résznél csináltuk meg a teljes weboldalunk kinézetét effecteket,színeket,betűformákat.



**Bootstrap**

A Bootstrap egy nyílt forráskódú keretrendszer a webes alkalmazások és weboldalak gyors

és hatékony fejlesztéséhez. Elsősorban HTML, CSS és Javascript segítségével dolgozik, és

előre elkészített sablonokat, stílusokat és komponenseket biztosít a webfejlesztőknek. Ennek

eredményeként a Bootstrap lehetővé teszi a fejlesztők számára, hogy könnyen

létrehozzanak reszponzív, jól kinéző és különböző eszközökön használható webes

alkalmazásokat és weboldalakat.

A Bootstrap segítségével a fejlesztőknek nem kell minden egyes projekt esetén teljesen újra

írniuk a kódot az alapvető funkciókhoz és elrendezésekhez. Ehelyett a Bootstrap sablonok és

komponensek széles választékát kínálja, amelyek egyszerűen testreszabhatóak és

kombinálhatók az adott projekt igényeinek megfelelően. Ez jelentősen felgyorsíthatja a

fejlesztési folyamatot és lehetővé teszi a konszolidált, egységes megjelenésű weboldalak

létrehozását.



**React**

Jelenleg a legnépszerűbb, és több interneten található adat alapján véve egyszerűbb és

kényelmesebb, mint egyéb programozási nyelv (Angulart hozzák többen erre példának, mivel

az is egy elterjedt nyelv.)

Első körben a typescriptet kell megismernünk, mivel a React egy typescript nyelvű könyvtár.

A typescript egy objektum-orientált programozási nyelv, mely a Javascript egy bővített és

tovább gondolt változata. A TypeScript olyan, más nyelvekből ismert gyakori fogalmakat ad

hozzá a JavaScripthez, mint például osztályok, interfészek, generikusság, és (opcionális)

statikus típusosság, ezáltal sokkal nagyobb eszköztárral rendelkezik és lehetőséggel.

A React először a Facebook házon belüli projektje volt, majd 2013-ban tették nyílt

forráskódúvá. Itt készült el a weboldal főbb része és a teljes frontendet ebbe készítettük el.

**Hogy működik a React?**

A ReactJS segítségével épített alkalmazások az egyszeri újrafelhasználhatóság elvét

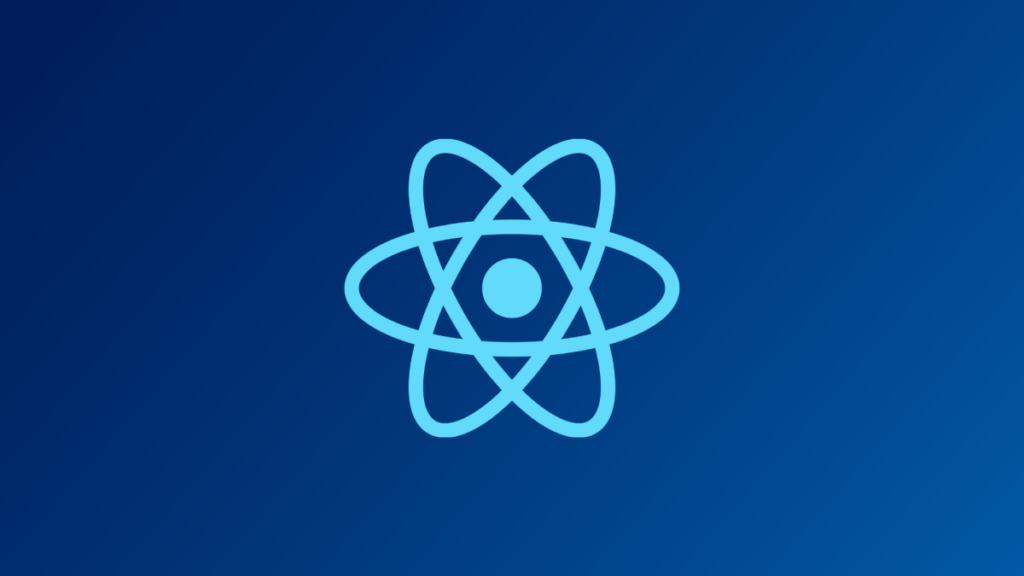
alkalmazzák. Ez arra az elvre épül, hogy a weboldalak és alkalmazások komponensekből,

egyirányú folyamattal épüljenek fel. A Reactben létezik az állapotok és a

megváltoztathatatlanság fogalma. A szülő és a gyermek komponensek tekintetében a

komponensek szigorú hierarchiával rendelkeznek. A React esetében egy komponens egy

meglévő kódrészlet, amely megírása a tiszta kód (Clean Code) elvén alapult.



**Dokumentum Objektum Model (DOM)**

A Dokumentum Objektum Modell (Document Object Model / DOM) egy platform - és nyelvfüggetlen standard objektummodell, amely a HTML, XHTML, XML, valamint rokon

formátumaiknak a szerkezetét és az objektumaikkal történő interakciókat modellezi. A DOM

egymással gyerek-szülő kapcsolatban álló objektumok rendszere. A dokumentum tartalmát,

illetve a dokumentum valamennyi összetevőjét magában foglalja. A beépített objektumok

kezelése böngészőnként eltérő lehet, továbbá plusz tulajdonságok is lehetnek különböző

böngészők esetén.

Az 1994-ben alapított World Wide Web Consortium, amely nyílt szabványokat támogat a

világháló számára, összehozta a Netscape Communications-t, a Microsoft-ot és más cégeket,

hogy kifejlesszenek egy szabványt a webböngészőkben használatos szkript nyelvek számára,

így alakult ki az „ECMAScript”. A szabvány első verziója 1997-ben látott napvilágot.

Az ECMAScript megjelenése után a W3C elkezdett dolgozni a DOM szabványosításán. A

kezdeti eredmények voltak az ún. „DOM Level 1”, amelyet 1998 végén terjesztett elő a W3C.

Ez a verzió már teljes modellt biztosított a teljes HTML és XML dokumentum számára,

beleértve a dokumentumok bármely részének megváltoztatásához szükséges eszközöket is.

2000 végén jelent meg a DOM Level 2. Bevezette a getElementById funkciót, valamint az

eseménymodellt, és támogatta az XML névtereket és a CSS-t. A webböngészők rendszerint a DOM-hoz hasonló belső modellt használnak a

dokumentumok (például egy HTML oldal) megjelenítéséhez. A DOM API-kat weboldalak,

illetve egyes részeik JavaScript kódból történő vizsgálására, vagy módosítására is használják.

Más szóval, a Dokumentum Objektum Modell az a mód, ahogyan a JavaScript látja a

webböngésző állapotait és az általa megjelenített HTML oldalt.

Amikor egy böngésző megjelenít egy HTML oldalt, akkor legelőször betölti az oldalt a web-

szerverről, majd értelmezi a jelöléseit (pl. HTML) és létrehozza a modellt a memóriában

tárolt DOM formájában.

**Backend**

**(Web API, ASP.NET CORE)**

Az API jelentése és felhasználása

Az API az angol Application Programming Interface rövidítést takarja, amit magyarra

alkalmazásprogramozási felületnek fordíthatunk. Egyáltalán nem számít új találmánynak,

hiszen már jóval a személyi számítógépek megjelenése előtt is létezett. Ekkor azonban még

csak lokálisan, egy-egy szerveren látott el különféle feladatokat. Az internet elterjedésével

vált az API igazán megkerülhetetlenné nagyjából a 2000-es évek közepetáján.

Kicsit konyhanyelvebben megfogalmazva: az API teszi lehetővé, hogy a szoftverek egymás

között kommunikáljanak, adatot cseréljenek, módosítsanak vagy előhívjanak egyes

funkciókat. Fontos, hogy az API nem egy programozási nyelv, és nem is korlátozódik egyetlen

nyelvre. Bármilyen programozási nyelven létrehozható és alkalmazható – és kismillió féle

programban és app-ban használható.

Nézzük először fejlesztői szemszögből az API előnyeit. Nos, az alkalmazásprogramozási

interfész egyszerűbbé teszi a munkájukat, és rengeteg időt megspórol nekik. Nem kell

minden funkciót az alapoktól saját kezűleg megírniuk:

hogy néhány fejlesztés nem kerülne be a programba, ha egyedileg kéne létrehozni, egy API-

val azonban mindez nem jelent akadályt. Végül a fejlesztőknek könnyebbséget jelent, hogy

az API platformfüggetlen, azaz bármilyen környezetbe, operációs rendszerbe integrálhatják.

Ami pedig a felhasználókat illeti, az API az ő életüket is jelentősen megkönnyíti, és ezt a

cégünknél is kiaknázhatjuk. A felhasználók ugyanis nem szeretnek tucatnyi programot

használni és megnyitogatni. Az API-val elérhetjük, hogy egy helyen, például a weboldalunkon

érjenek el számos funkciót. Már említettük, hogy bejelentkezhetnek közösségi média

fiókjukkal vagy fizethetnek külső alkalmazással.

Az API-val ezen túl arra is lehetőségünk nyílik, hogy valós raktárkészlet-információkat

biztosítsunk az érdeklődőknek vagy éppen arra, hogy időpontot foglaljanak a Google Naptár

segítségével – és a lefoglalt időpont azonnal bekerüljön a személyes naptárjukba. Ha pedig

egy ügyfél csetelni szeretne velünk, egy API támogatásával ezt megteheti az oldalunkon

elérhető WhatsApp vagy Messenger használatával. A Backend részünk folyamatos fejlesztés

alatt áll ezért nem tökéletes minden benne ezért szeretnénk majd minden apró hibát kijavítani.

**ASP.NET**

Az ASP.NET egy nyílt forráskódú, szerveroldali webalkalmazás-keretrendszer, amelyet

webfejlesztésre terveztek, dinamikus weboldalak előállítására. A Microsoft fejlesztette ki,

hogy lehetővé tegye a programozóknak dinamikus webhelyek, alkalmazások és

szolgáltatások létrehozását. A név az Active Server Pages Network Enabled Technologies

rövidítése.

Először 2002 januárjában adták ki a .NET-keretrendszer 1.0-s verziójával, és a Microsoft

Active Server Pages (ASP) technológiájának utódja. Az ASP.NET a Common Language

Runtime-ra (CLR) épül, lehetővé téve a programozók számára, hogy ASP.NET kódot írjanak

bármely támogatott .NET nyelv használatával. Az ASP.NET SOAP kiterjesztési keretrendszer

lehetővé teszi az ASP.NET összetevők SOAP üzenetek feldolgozását.

Az ASP.NET utódja az ASP.NET Core. Ez az ASP.NET újbóli megvalósítása moduláris webes

keretrendszerként, más keretrendszerekkel, például az Entity Frameworkkal együtt.

A backend feladata

A backend a hátsó, felhasználónak szempontból láthatatlan, de a programokban fontos és

tényleges számításokat végző része.

Feladata a frontend felől érkező adatok gyors feldolgozása, és az eredmények visszajuttatása

a frontendnek, de szerver és kliens kapcsolatot is a backend működteti, illetve az adatbázis

feldolgozását, és az adatok megfelől védelmi tárolásáért és módosításaiért is a backend

felelős.

**Trello**

Ahhoz, hogy minél egyszerűbb legyen a közös munkánk és a folyamatok nyomon követése, a

Trello-t hívtuk segítségül.

Könnyedén használható felület, ahol munkateret hozhatunk létre, azon belül pedig táblákat,

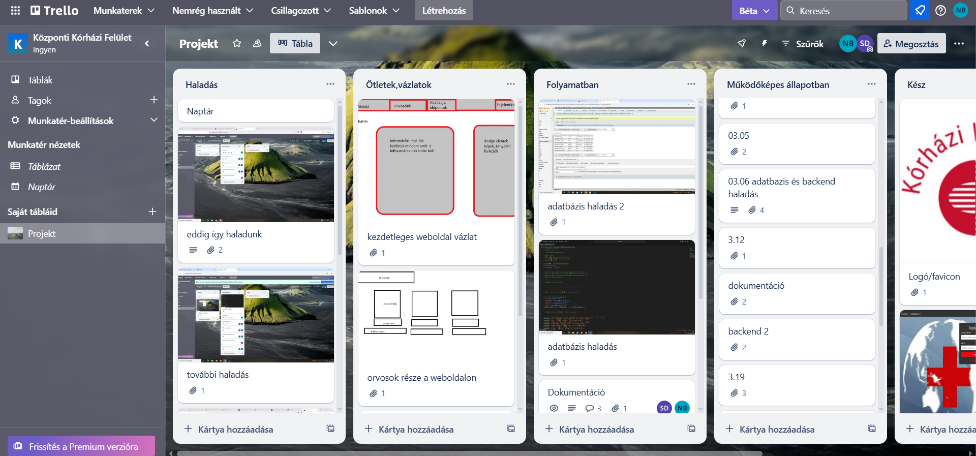
illetve listákat készíthetünk.

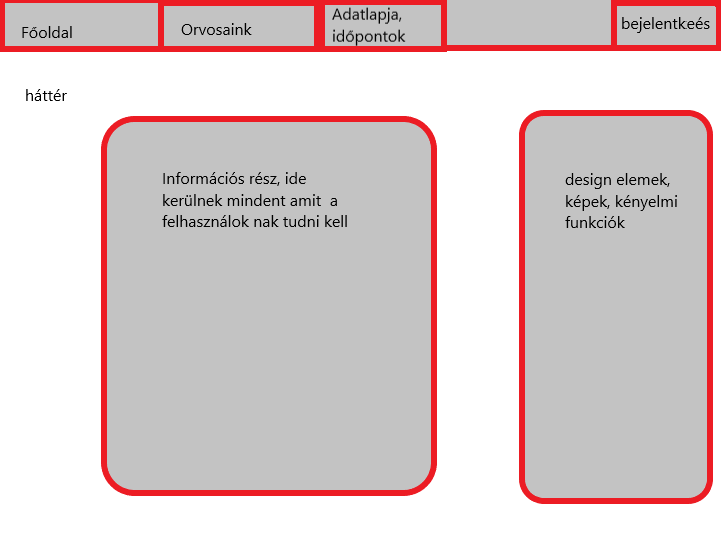
Ezzel a technológiával követtük nyomon magunkat, hogy hol tartunk, mi az, amit szeretnénk,

mi az, ami épp megvalósítás alatt van, illetve mi az, amivel már készen vagyunk és

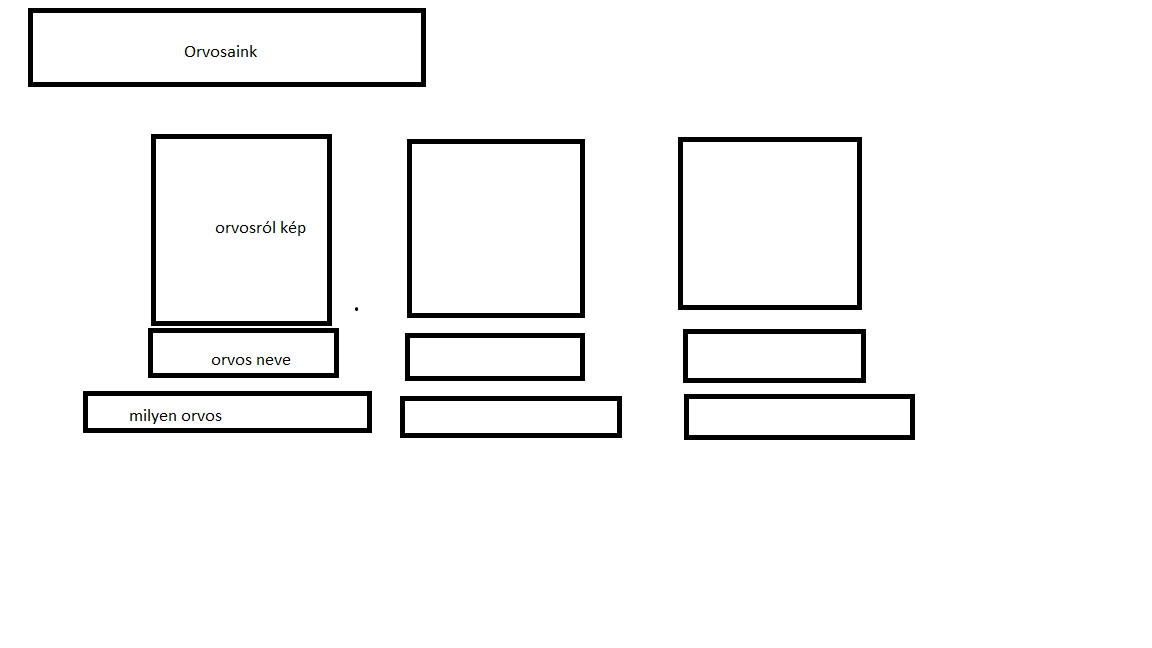
fellélegezhetünk.

Itt történt a kommunikáció ahol a csapaton belül megbeszéltük hogy hogyan is nézzenek ki a

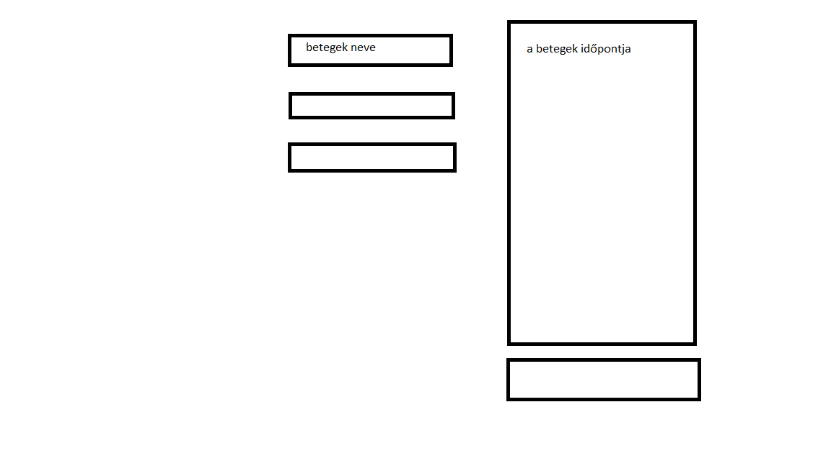
dolgok és milyen új ötletek legyenek még benne .



ez volt a weboldalunk első elképzelése konkrétan a „prototipus” hogyan is fog kinézni a weboldal maga.



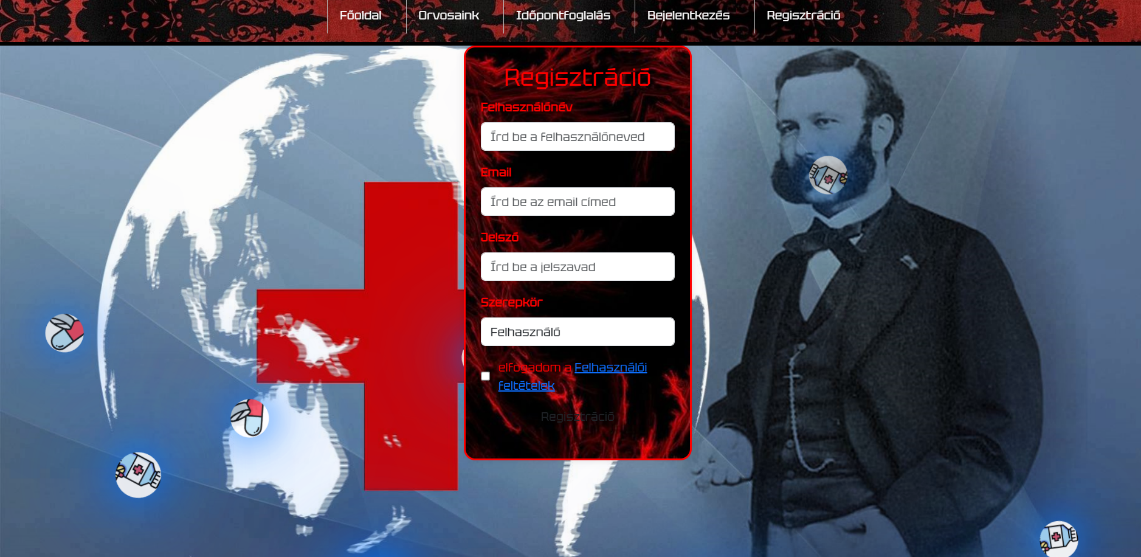
Ez az orvosok részen belüli elképzelés volt hogy kártyánként oldjuk meg az orvosokat.



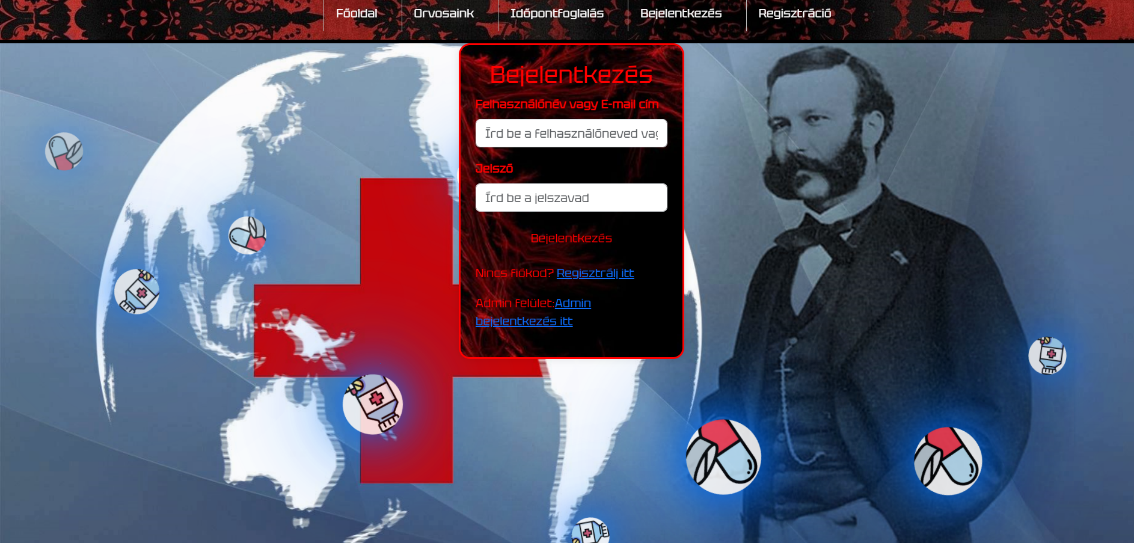
Ez az időpontfoglalás első változata volt azóta az is fejlődött.



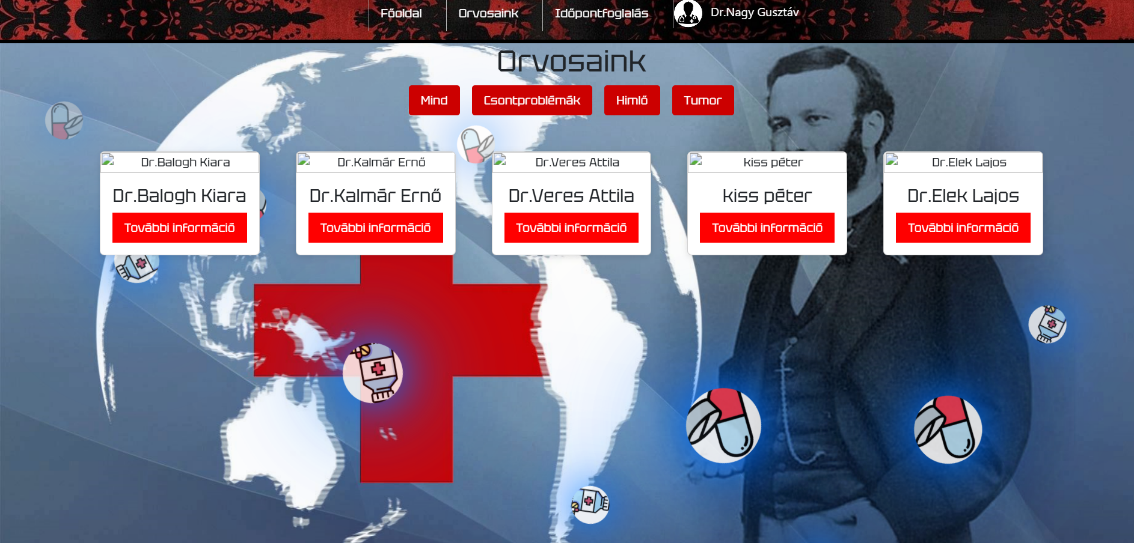
Először is ha fellépünk az oldalra az üdvözlő oldalunk ami egyben a főoldalunk is viszont ez egy kicsi egyszerű része csak a weboldalnak.



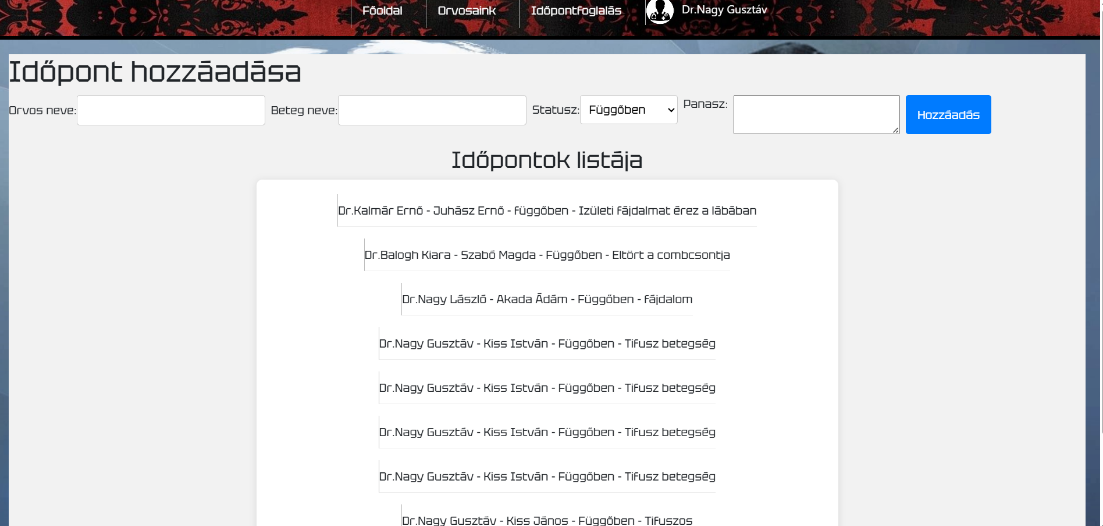
Itt látható a regisztrációs felület amin a felhasználó illetve egy újónnan érkezett orvos(admin)is tud regisztrálni.



Itt a bejelentkezés felület látható ahol a felhasználó és az admin tud bejelentkezni az oldalra ha nem jelentkeznek be nem tudnak semmit csinálni az oldalon .



Itt látható az orvosok része a weboldalnak itt meggyűlt a bajunk mivel a képek az orvosainkhoz nem nagyon akarnak működni viszont dolgozunk a megoldáson.



Ez az időpontfoglalás rész ahol a felhasználó tud időpontot foglalni magának és az admin az orvos tudja ezt kezelni.

**Források:**

<https://www.oracle.com/hu/database/what-is-database>, letöltés dátuma: 2025.04.09.

<https://www.linuxportal.info/enciklopedia/p/phpmyadmin>, letöltés dátuma: 2025.04.09.

<https://www.linuxportal.info/enciklopedia/p/phpmyadmin>, letöltés dátuma: 2025.04.09.

<https://www.prooktatas.hu/hirek/front-end-webfejlesztes-react-tel-miert-jo-valasztas>, letöltés dátuma: 2025.04.09.

[https://hu.wikipedia.org/wiki/Document\_Object\_Model forrás](https://hu.wikipedia.org/wiki/Document_Object_Model%20forrás), letöltés dátuma: 2025.04.09.

<https://kiszervezettmarketing.hu/weboldal-keszites/bootstrap/>, letöltés dátuma:

2025.04.09.

<https://hu.wikipedia.org/wiki/ASP.NET>, letöltés dátuma: 2025.04.09.

<https://learn.microsoft.com/hu-hu>, letöltés dátuma: 2025.04.09.