

SZAKDOLGOZAT

Csuzdi Bianka

2022

Magyar Agrár- és Élettudományi Egyetem
Károly Róbert Campus

Webalkalmazás készítése
a VI-NI TRANSZ Fuvarozó Kft.
részére

Konzulens

Sike Zoltán

mesteroktató

Készítette

Csuzdi Bianka

gazdaságinformatikus szak

nappali tagozat

2022

Tartalomjegyzék

1.	Bevezetés	3
1.1.	Témaválasztás	3
1.2.	A projekt rövid leírása	4
2.	Irodalmi áttekintés	5
2.1.	Webprogramozás	5
2.2.	Webalkalmazáshoz használt technológiák	7
2.2.1.	PHP	7
2.2.2.	Python	7
2.2.3.	HTML	7
2.2.4.	CSS	8
2.2.5.	JavaScript	9
2.2.6.	MySQL	9
2.2.7.	Frontend keretrendszer: Bootstrap	10
2.2.8.	Python-nal használható webes keretrendszerek	10
2.2.8.1.	Django	10
2.2.8.2.	Flask	11
2.2.9.	Visual Studio Code	11
2.3.	A webalkalmazások gazdasági oldalról	12
2.3.1.	E-kereskedelem	12
2.3.2.	Online Marketing	12
2.3.2.1.	Keresőmotor-optimalizálás (SEO).	12
2.3.2.2.	Keresőhirdetés	13
2.3.2.3.	Közösségi média marketing	13
2.3.3.	Mikro- és kisvállalkozások helyzete	14
2.3.3.1.	A vállalkozások típusai	14
2.3.3.2.	Kisvállalati marketing	14
2.3.4.	Vállalati példák	15
3.	Anyag és módszer	17
3.1.	VI-NI Transz Fuvarozó Kft. bemutatása	17
3.2.	Webalkalmazás	18
3.2.1.	Programtervezés a vállalkozással	18
3.2.2.	Elvárások informatikai oldalról	20
3.2.3.	Adatbázisterv	21

3.2.4.	Felhasznált eszközök és technológiák	22
3.2.4.1.	Kódszerkesztő: Visual Studio Code	22
3.2.4.2.	Programnyelvek	23
3.2.4.3.	Keretrendszerek	23
3.2.5.	Az időpontfoglaló rendszer működési folyamata	23
4.	Eredmények és értékelésük	25
4.1.	A projekt állapota	25
4.2.	Technológiák értékelése	25
4.3.	A kritériumok megvalósítása	26
4.3.1.	Nyitó oldal	26
4.3.2.	Fuvarozó	27
4.3.3.	Műhely	29
4.3.4.	Admin felület	34
4.4.	Értékelés és következő lépések	39
5.	Következtetések és javaslatok	42
6.	Összefoglalás	45
7.	Irodalomjegyzék	47
8.	Táblázatok és ábrák jegyzéke	50
9.	Nyilatkozatok	51
9.1.	Nyilatkozat a szakdolgozat nyilvános hozzáféréséről és eredetiségéről	51
9.2.	Konzultációs nyilatkozat	52
10.	Mellékletek	54
10.1	2004. évi XXXIV. törvény	54

1. Bevezetés

1.1. Témaválasztás

A Webprogramozás I. és Webprogramozás II. kurzusok elvégzése közben mindig is érdekelt, hogy milyen lehet egy olyan webalkalmazást létrehozni, amit az emberek ténylegesen használnak és nem csak egy tesztfelület. Egy-egy weboldal meglátogatásakor sokszor elgondolkodtam, hogy milyen egyedi szempontokat kell figyelembe venni egy-egy speciális webalkalmazás elkészítésénél. Emellett kíváncsi voltam, hogy milyen lehet együtt dolgozni egy vállalkozással a fejlesztés során. Érdekelt, hogy a vállalkozások jellegzetes megjelenését hogyan lehet implementálni a weboldalra úgy, hogy az a márkához hű maradjon. Szakdolgozatom fő témája éppen ezért a webprogramozásra esett.

Többek között az egyetemi kurzusokon megismert technológiákat szeretném használni az alkalmazás fejlesztése során, de szeretném megvizsgálni, hogy milyen programnyelv alkalmazható még a webprogramozáshoz, illetve ezt milyen keretrendszerrel lehet kiegészíteni. Ennek keretében pedig célom, hogy a meglévő ismereteimet kiegészítve, akár az újonnan megismert programnyelvvel és keretrendszerekkel létrehozzak egy jól működő webalkalmazást.

Mivel egy működő cégnek szerettem volna fejleszteni, ezért a szakdolgozat elkészítése előtt megkerestem egyet, amelynek még nem volt saját weboldala. Az alkalmazást egy kis hazai vállalkozásnak, a VI-NI TRANSZ Fuvarozó Kft.-nek fejleszttem. A cég székhelye a Heves vármegyei Petőfibányán található. A fő tevékenysége a közúti áruszállítás, de várhatóan a következő hónapok során új tevékenységgel egészül ki a vállalkozás profilja. Gépjármű alkatrészekkel is szeretnének foglalkozni, e célból alakítanak ki egy gumis szervízt.

A dolgozatom további részében részletesen kitérek az általam alkalmazott informatikai megoldásokra és webes alkalmazással kapcsolatban kitűzött igényekre, célokra és azok megvalósítására. Mivel napjainkban az interneten rendkívül elterjedt a közösségi média és sok vállalkozás ezeket a platformokat is aktívan használja, ezért szeretném megvizsgálni, hogy a közösségi médiában történő megjelenések mellett miért fontos a saját weboldal üzemeltetése. Illetve, hogy napjainkban a kisvállalkozások számára milyen új gazdasági lehetőségeket ad egy weboldal létrehozása. Végül, de nem utolsónak pedig, hogy milyen előnyökkel és hátrányokkal járhat számukra egy webalkalmazás üzemeltetése.

Ez a mellékes kutatás fontos szereppel bír, ugyanis így megfelelő háttér tudással tudom támogatni a vállalkozást a projekt elkészítése során.

1.2. A projekt rövid leírása

Mielőtt átbeszéltük a webalkalmazás részleteit a vállalkozással jó magam is igyekeztem ötletelni. Ugyanis a fejlesztésnél az ügyfél nem mindig tudja milyen lehetőségei vannak. Ezért összegyűjtöttem ötleteket és funkciókat, amik alapján elkezdtünk beszélgetni az igényekről. A webalkalmazás programozása előtt a funkciókat és az egyéb szükséges információkat egyeztettem a vállalkozással. A fejlesztést ez alapján kezdtem el.

Az igények szerint a webalkalmazás négy részre osztható szét: a két tevékenységi körhöz kapcsolódó felületre, egy nyitóoldalra és egy admin felületre. A nyitóoldaltól a felhasználó tovább léphet arra a területre, ami miatt rákeresett a vállalkozásra.

A cég már több éve foglalkozik fuvarozással, így ehhez kapcsolódóan az igény egy tájékoztató jellegű weboldal, ahol a látogatók információt tudnak szerezni a cég szolgáltatásairól és elérhetőségeiről.

A szerviz esetében is fontos a tájékoztatás, de egy sokkal interaktívabb, látványosabb felületre van szükség. Mivel ezen a területen a vállalkozás új belépőnek számít és főként magánszemélyeket szólít meg, ezért nagyobb figyelmet kell fordítani a potenciális ügyfelekkel való kapcsolat kiépítésére. Ehhez megfelelő részletességgel kell bemutatni a vállalkozást, elérhetővé kell tenni a referencia munkákat, illetve biztosítani kell a lehetőséget a kapcsolat felvételéhez. A felsoroltakon kívül a cég szeretne egy időpontfoglaló rendszert kialakítani a szervizhez. Ezen a felületen az ügyfelek kiválaszthatják a szabad időpontokból, hogy mikor szeretnék leadni az autót és milyen szolgáltatásokat vennének igénybe, illetve a foglalások státuszait online nyomon követhetik.

Az admin felület kialakításánál az a cél, hogy könnyen és átláthatóan hozzáférjen a vállalkozás az adatbázishoz és az időpontfoglalásokat kezelni tudja. A későbbiekben pedig lehetősége legyen referencia munkák feltöltésére és fontosabb információk módosítására. Emellett az ügyfelekkel való kapcsolattartás megkönnyítése is célként fogalmazódott meg benne.

2. Irodalmi áttekintés

2.1. Webprogramozás

A hálózatok hálózata, avagy az internet, amelyhez kezdetben még csak számítógépek, napjainkban már különböző szenzorok és eszközök is képesek csatlakozni. (FEHÉR, 2017) A mai világban rutinszerűvé vált az internet mindennapi használata és a weboldalak böngészése. Bárki, aki rendelkezik számítógéppel, lappal vagy telefonnal és van internet kapcsolata, az már böngészett webalkalmazásokat, amik a webprogramozás eredményeként jöttek létre.

A webprogramozás egy rendkívül komplex terület. A world wide web alapjait Tim Berners Lee az 1990-es évek elején találta fel. Ezen elemek közé tartozik a webszerver, a kliens és feladataik. Továbbá a HTML nyelv, az URL címzés és a HTTP protokoll is a world wide web alapjaihoz tartozik. Kezdetben még csak statikus weboldalak jöttek így létre, ahol a fejlesztő határozta meg a tartalmat. Később megjelentek a dinamikus weblapok, ahol a tartalom már a programtól függött. (SZABÓ, 2013)

A weboldalak létrehozása több programnyelv ismeretét igényli. Elmondható, hogy az informatika és az elektronikus eszközök fejlődésével párhuzamosan a mai napig is folyamatosan új lehetőségek és opciók jelennek meg a fejlesztésben.

A különböző alkalmazások, így a webalkalmazások is három fő részre bonthatók szét: backend, frontend és adatbázis. Az applikációk felhasználói a frontend oldalt használják és látják. A backend szolgálja ki a a frontendről érkező igényeket és kapcsolja össze szükség esetén az adatbázissal őket. Az adatbázisban a szükséges adattáblák vannak eltárolva. Ahhoz, hogy a korábban említett részek összehangoltan tudjanak működni, a fejlesztők több programnyelvvel, keretrendszerrel és egyéb technológiákkal dolgoznak egyszerre. Ezen eszközök kiválasztása tudatosan történik. Például a keretrendszerek kiválasztása is egy bonyolultabb feladat. Fontos, hogy megfelelő keretrendszert válasszunk az applikációhoz, ugyanis a nem megfelelőből hibák adódhatnak vagy elhúzódhat a fejlesztés időtartama. A fejlesztőnek célszerű ismernie az általa használt programnyelvekhez kapcsolható keretrendszereket és az ezekkel kapcsolatos bevált technikákat. (M. DEL SALES-ZÁRATE et al, 2015)

Ahhoz, hogy egy jól működő webalkalmazás népszerű is legyen a felhasználók körében fontos követni az aktuális trendeket, irányvonalakat és figyelembe venni néhány szempontot. Ezek többnyire a frontend oldal megjelenését befolyásolják. Milyen kritériumok vannak pontosan, amik a weblapunk sikerét segítik elő? Az első és legfontosabb, hogy a célcsoportnak megfelelő információk és funkciók legyenek megtalálhatóak a weboldalon. A kommunikáció fontos szerepet tölt be az interneten, ezért a weboldalon érdemes kialakítani különböző lehetőségeket erre. Emellett nagyon általános dolog, de mégis szükséges kiemelni, hogy a weblap frontend oldalának szépnek, esztétikusnak és aktuálisnak kell lennie. Fontos, hogy a weboldal felépítése és a menüsor átlátható, logikus legyen és segítse a felhasználót. Végül, de nem utolsó sorban figyelembe kell venni a célcsoport igényeit. Például napjainkban sokan okostelefonról böngésznek, így érdemes reszponzív weboldalt készíteni, ami alkalmazkodik a felhasználó eszközéhez. (REKETTYE et al, 2016)

2.2. Webalkalmazáshoz használt technológiák

2.2.1. PHP

A PHP programnyelvet azért szükséges használni a webprogramozáshoz, mert lehet kódolni azokat a funkciókat benne, amikkel a webszerver a kliensnek válaszol. Amennyiben ezen programnyelv használata mellett dönt a fejlesztő, akkor a legáltalánosabb módszer, hogy Apache-PHP-MySQL alapú webfejlesztést alkalmazzák. (NAGY, 2011)

2.2.2. Python

A Python egy nyílt forráskódú programozási nyelv, ami egyre népszerűbb. Objektumorientált szkriptnyelv-ként tartják számon. Fontos megemlíteni, hogy a Python kódok könnyen olvashatóak, így az informatika több területén is használják a fejlesztők. Többek között data science, IT biztonság és webprogramozás területeken is előszeretettel alkalmazzák. Számos könyvtárat tudunk importálni az alkalmazásba, amik nagyban elősegítik a fejlesztői munkát. (LUTZ, 2001)

2.2.3. HTML

A HTML azaz a Hyper Text Markup Language egy leíró nyelv, ami tulajdonképpen a weboldalak szerkezetét írja le. A HTML állomány jelölőtagok segítségével írja le az oldal felépítését. Ezeket .html kiterjesztésű fájlokban kell menteni. A HTML állomány létrehozásához nem feltétlen szükséges fejlesztői környezet, ugyanis egy egyszerű szövegszerkesztő programmal is megírható, de sokkal célszerűbb az IDE-ben megírni. Ugyanis ezekben több billentyűparancs gyorsítja a munkát és a tagolást is megkönnyíti.

Mint korábban is említettem a HTML állomány jelölő tagokat tartalmaz. Ezek egy-egy elemet jelölnek a weboldalunkban. A jelölők jellemzően két tagból állnak: nyitó és záró tag, de vannak kivételek, amik csak egy tagból állnak. Kéttagú jelölő esetében a kettő között helyezkedik el a szöveg vagy tartalom. A HTML dokumentumok egy `<!DOCTYPE html>` címkével kezdődnek és az első jelölő tagja a `<html>` nyitó tag, utolsó jelölő tagja pedig a `</html>` záró tag. A jelölő tagok egymásba ágyazhatók, így ezen belül további jelölőket adhatunk meg. A `<html>` `</html>` jelölő tagokba ágyazva szükséges még használni a `<head>` `</head>` és `<body>` `</body>` jelölő tagokat is. A `<head>` nyitó és `</head>` záró tag között adhatóak meg alapvető információk az oldalról, például a címe. De itt kapcsolhatunk hozzá

egyéb fájlokat is, amik az oldal működéséhez szükségesek például egy CSS fájl, Bootstrap vagy JavaScript fájl. A `<body>` `</body>` jelölő tagok közé kerül az a tartalmi rész, amit a felhasználó lát a weboldalon. Ezeken kívül számos jelölő létezik. Vannak jelölő tagok a konténerekhez, címsorokhoz, bekezdésekhez, hivatkozásokhoz, űrlapokhoz, de vannak olyan elemek is, amivel formázni tudjuk a weblapunkat. Bár utóbbi kevésbé használatos, ugyanis ezt CSS segítségével sokkal igényesebben lehet megoldani. (NAGY, 2011)

2.2.4. CSS

Míg a HTML a weboldal szerkezetét írja le, addig a CSS azaz a Cascading Style Sheet a kinézetét definiálja. A CSS fájlok rangsorolt stíluslapok, ami azt jelenti, hogy az ebben megírt szabályok a HTML fájl elemeire hierarchikusan alkalmazza a rendszer. A stíluslapokat megírhatjuk a HTML dokumentumon belül is vagy külön fájlban is. Utóbbi esetben .css kiterjesztéssel kell lementeni a fájlt. Ezt a fájlt a HTML dokumentumokkal a HTML dokumentum `<head>` tagjában lehet összekapcsolni.

Érdemes a fenti opciókból a külön fájlt választani ugyanis, ha egy komplex webalkalmazást szeretnénk létrehozni több lappal és egy design-nal, akkor ez az optimális választás. Mert elég egyszer egy CSS fájlban megírni a stílus szabályokat és csak ezt kell hozzákapcsolni a HTML dokumentumokhoz. Egy CSS fájlban eltárolhatjuk egy cég stílusát és a későbbiekben, ha szükséges egy fájlban átírhatjuk az összes lap kinézetét. Így például, ha megváltozik a cég által használt betűtípus vagy a színvilágban történik módosítás, akkor ezeket egyszerűen és gyorsan át lehet írni.

A CSS-ben számos megjelenéssel kapcsolatos dolgot szabályozhatunk. Vannak elrendezési és formázási tulajdonságok. Az elrendezési tulajdonságok segítségével határozhatjuk meg, hogy az adott HTML jelölő tag hol jelenjen meg, míg a formázási tulajdonsággal meghatározhatjuk hogyan jelenjen meg azaz milyen színnel, stílussal és így tovább. Ezeket a szabályokat nem feltétlen kell HTML jelölő tagra megadni, class vagy id segítségével megadhatjuk speciálisan egy-egy elemre is. Emellett kezelhetünk különböző eseményeket is. Például mi történjen, ha a felhasználó egy adott elem felé viszi az egér mutatóját. (MELONI, MORRISON, 2010)

2.2.5. JavaScript

A JavaScript egy szkriptnyelv, amit a böngésző értelmez. Jellemzője, hogy interaktivitást biztosít a weboldalaknak. A webprogramozáshoz használt JavaScript kód megírható külön fájlba, de akár a HTML forráskódba építhető. Neve hasonlít a Java programozási nyelvre, de ennek ellenére nincs szoros kapcsolatban a korábban említett programnyelvvél. Azonban fontos tudni, hogy C++ és Java szintaxisra alapoz. (NAGY, 2011)

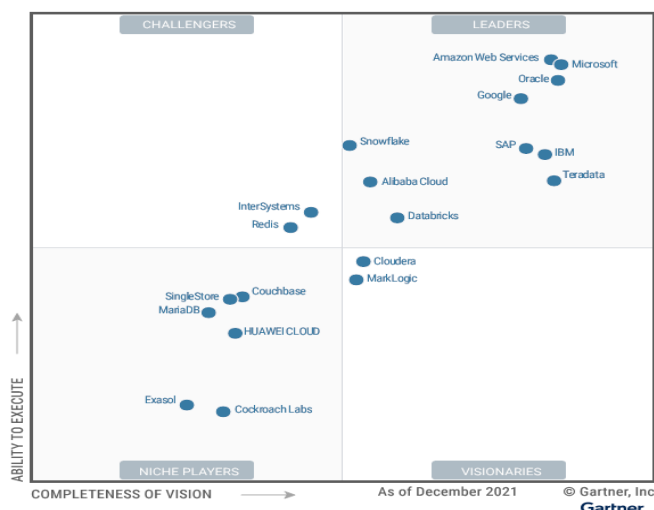
2.2.6. MySQL

Az adatbázis integrált, logikailag összetartozó információk összessége. Az adatbázisban adattáblákban tárolhatóak el a rekordok. Az adattáblákban mezőket definiálhatunk. Megadhatjuk a mező nevét és típusát. Kijelölhetünk kulcsokat, amik által egy-egy adatsort megkülönböztethetünk. Ez azért lehetséges, mert a kulcsok egyedi értékek, így egyértelműen azonosítja a rekordokat. Az adatbázisban található táblák között szükség esetén létrehozhatunk kapcsolatokat. Ezek a kapcsolatok lehetnek egy-egy, egy-sok vagy sok-sok típusúak. (RADVÁNYI, 2014)

Az adatbázisokhoz kapcsolódóan az egyik legnépszerűbb rendszer a MySQL. Ez egy nyílt forráskódú adatbázis kezelő szoftver, amiben relációs adatbázisokkal egyszerűen lehet dolgozni. A szoftver sikerességét mi sem bizonyítja jobban, hogy a Sun Microsystems (a jelenlegi Oracle Corporation) felvásárolta. (ARRINGTON, 2008)

Az Oracle egy sikeresnek mondható vállalat. A Gartner 2021-ben kiadott felhős adatbázis elemzésében továbbra is a vezetők között található. A következő ábrán látható, hogy az élen szerepel olyan szolgáltatók mellett, mint a Microsoft és Amazon. De a korábbi években is hasonlóan a vezetők között volt megtalálható a cég. (COOK et al, 2021)

1. ábra: Gartner féle Mágikus kvadráns az adatbázis kezelő rendszerekről



Forrás: Gartner (december 2021)

2.2.7. Frontend keretrendszer: Bootstrap

A HTML, CSS és JavaScript egyik, hanem a legnépszerűbb keretrendszere a Bootstrap. A Bootstrap egy frontend oldali keretrendszer, aminek a sikere több okból ered. Nyílt forráskódú keretrendszer, amit megfelelő HTML és CSS tudással egyszerűen lehet használni, ugyanis nagyszerű, átlátható dokumentáció érhető el hozzá. (GAIKWAD et al 2019)

A Bootstrap használatának több előnye is van. Egyrészt a legnépszerűbb böngészőket támogatja, másrészt a „Mobile first approach” szerint működik. Ami azt jelenti, hogy a design elsőként mobil képernyőre készül el és utána alakul tovább és igazodik a számítógép képernyőjéhez. Végül, de nem utolsó sorban a Bootstrap használatával egyszerűen létre lehet hozni reszponzív felületeket. Ez egy nagyon hasznos funkció, ugyanis így az elkészített weboldal felépítése a felhasználó eszközéhez alkalmazkodik. Ez a Bootstrap-ben a rács rendszer segítségével valósul meg. (GAIKWAD et al 2019)

2.2.8. Python-nal használható webes keretrendszerek

2.2.8.1. Django

A Django egy nyílt forráskódú webes keretrendszer. A projektet 2005-ben publikálták. A keretrendszer megírása Adrian Holovaty és Simon Willison nevéhez köthető. A Django a Python programnyelvre épül, ezért a használatához a Python ismeretére és magára a programnyelvre is szükség van. Napjainkra a Django közkedvelt keretrendszer lett

nemcsak a Python-ra épülő keretrendszerek között, hanem a webes keretrendszerek között is. Több népszerű cég is alkalmazza, mint például a Google, The Washington Post és a Tabblo. (FORCIER et al 2008)

2.2.8.2. Flask

A Flask a Django-hoz hasonlóan egy Python-ra épülő webes keretrendszer. A Flask különlegessége, hogy a fejlesztőnek a többi keretrendszerhez képest szabad kezet ad. Például az applikációk fejlesztése során támogatja a relációs és nem relációs adatbázisok használatát is. De adatbázis használat nélkül is működtethető. A keretrendszer célja, hogy felgyorsítsa a webes alkalmazások létrehozását. (GRINBERG, 2018)

2.2.9. Visual Studio Code

A Visual Studio Code egy kódszerkesztő alkalmazás. A Visual Studio Code elérhető Windows, macOS és Linux operációs rendszerekre is. Számos programnyelvet és bővítményt támogat. Használata rendkívül egyszerű. Nagy előnye, hogy az IntelliSense és Debug funkciók elősegítik a hatékonyabb, gyorsabb kódolási élményt. Az IntelliSense egy kódkiegészítő segédprogram, ami gyorsítja a kódolási folyamatot. Számos hasznos funkciója van. Kiegészíti a kódban előforduló parancsokat, így azokat nem kell végig begépelni, de a paraméterezés és változók használatában is segít. A debug funkció a különböző hibák kezelésében segíti a fejlesztőt.

2.3. A webalkalmazások gazdasági oldalról

2.3.1. E-kereskedelem

Napjainkban egyre több folyamat zajlik online. Ezeket a folyamatokat 3 csoportba sorolhatjuk: információszerzés, kommunikáció és üzleti tranzakciók végrehajtása. Ahhoz, hogy ezek a folyamatok működjének alkalmazásokra van szükség. Ilyen alkalmazások például az elektronikus levelezés, blog, csevegőprogramok, keresőmotorok és webáruházak. (KACSUKNÉ BRUCKNER et al, 2019) Már a felsoroltakból is látható, hogy az internetben rengeteg lehetőség van, természetesen gazdaságilag is.

Az e-business (elektronikus üzletvitel) népszerűsége az elmúlt években folyamatosan növekszik. Ennek nagy részét az e-kereskedelem teszi ki. Az e-kereskedelem azokat a tranzakciókat és folyamatokat foglalja magába, ahol az összeköttetés a gazdálkodó szervezet és a fogyasztó között elektronikus úton történik meg. (AVORNICULUI et al, 2019)

Tehát az elektronikus kereskedelem is eladó-vevő kapcsolatról szól a hagyományos kereskedelemhez hasonlóan. Azonban mégis számos előnye van a hagyományos kereskedelemhez képest. Az online téren a sokkal egyszerűbb az információközlés és reklámozás. A folyamatok gyorsabbak, de a költségük kevesebb. Az e-kereskedelem lehetőséget ad az eladónak, hogy új értékesítési lehetőségei legyenek és betörhessen a még ki nem aknázott piacokra. (AVORNICULUI et al, 2019)

2.3.2. Online Marketing

Az online térben egyre több versenytárs van jelen, ezért a marketingre is érdemes figyelmet fordítani. Az informatika fejlődésével a marketingen belül is kialakult az online marketing. Vannak olyan online marketings eszközök, amik kifejezetten a weboldalak megtalálását segítik elő, ilyen a keresőmotor-optimalizálás is.

2.3.2.1. Keresőmotor-optimalizálás (SEO).

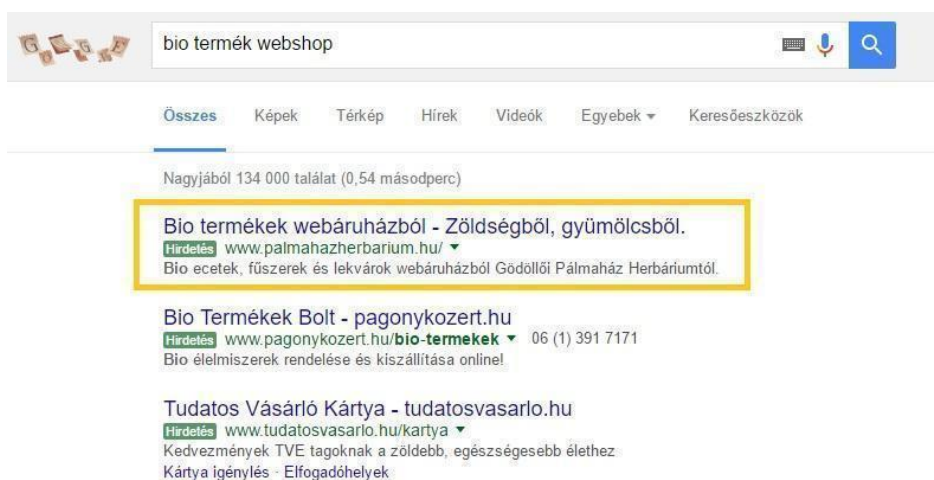
A felhasználók az internetes keresőkben általában szavakra vagy kifejezésekre keresnek. Az eladó célja, hogy a találati listán minél előrébb kerüljön. A keresők három szempontot vesznek figyelembe a találatok rangsorolásakor. Megvizsgálja a weboldalra

mutató hivatkozás helyét, tartalmát és számát. Felügyeli, hogy mi található a weboldalon és milyen kifelé mutató hivatkozások vannak a weblapon. (BÁNYAI et al, 2016)

2.3.2.2. Keresőhirdetés

A keresőket hirdetési felületként is fel lehet használni. Lehetőség van arra egy-egy keresőmotornál, hogy bekerüljenek a weboldalak egy fizetett listára. Ahogyan a következő képen is látható, ez egy kiemelt és elkülönített területet biztosít a találatok előtti megjelenésnél. Ezáltal a felhasználó nagyobb valószínűséggel talál rá a weboldalra.

2. ábra: Keresőhirdetés a Google-ben



Forrás: <https://marketing21.hu/2016/10/11/keresohirdetes-online-marketing-amit-ismerni-kell/>

Keresőhirdetés létrehozásánál nagy figyelmet kell fordítani a hirdetés szövegére és a kulcsszavak kiválasztására. A célzás során lehet megadni kizáró szavakat is, így még jobban tudjuk szabályozni, hogy kit érjen el a hirdetés. (BÁNYAI et al, 2016)

2.3.2.3. Közösségi média marketing

A közösségi média platformjai web alapú szolgáltatások, amelyek lehetővé teszik az önkefejezést, megosztást és tartalomgenerálást. A platformokon az eladóknak lehetősége van marketing tevékenységet is folytatni. A közösségi média marketing napjainkra egy elterjedt hirdetési forma. A hirdető kapcsolatba tud lépni a fogyasztókkal, párbeszédet és témát tud velük generálni. A platformon zajló fogyasztói párbeszéd alapján lehetősége van formálni a cég marketingkommunikációját. (BÁNYAI et al, 2016)

2.3.3. Mikro- és kisvállalkozások helyzete

2.3.3.1. A vállalkozások típusai

A 2004. évi XXXIV. törvényben meghatározták, hogy egy vállalkozás mikor minősül mikro-, kis- vagy középvállalkozásnak. Mikro-, kis- és középvállalkozásnak (a továbbiakban: KKV) minősül létszám alapján az a vállalkozás, aminek a létszáma 250 főnél kevesebb. Az 50 fő alatti vállalkozások kisvállalkozásnak, míg a 10 főnél kevesebb alkalmazottat foglalkoztató cégek mikrovállalkozásoknak minősülnek.

2.3.3.2. Kisvállalati marketing

Rekettye Gábor a Marketing a magyar kisvállalatoknak című könyvében nagyszerűen összefoglalta a kisvállalati marketing lényegét és eszközeit. A kisvállalatok marketing tevékenysége eltér a nagyvállalatok tevékenységétől. Jellemzően nem állnak rendelkezésre olyan nagy mértékű pénzügyi eszközök, ezért sokkal olcsóbb és egyszerűbb, mégis hatékony módszerekhez és eszközökhöz kell folyamodniuk. Ez az online marketing esetében is hasonlóan igaz.

Napjainkra a kisvállalatoknak is elkerülhetetlen terület az internet. Mondhatni a piacon maradás egyik feltételévé nőtte ki magát. A cég internetes jelenléte új lehetőségeket ad. Bemutatja az általuk kínált termékeket és szolgáltatásokat, emellett információt biztosít a vevőnek. Megfelelő internetes jelenléttel a vevők bizalmát is könnyebb elnyerni.

Több online marketing eszköz is könnyen elérhető a kisvállalatok számára. Az egyik legnépszerűbb az email, ami reklámozási, kapcsolattartási és értékesítési eszköz is lehet a vállalkozás számára.

A másik legnépszerűbb eszköz a weboldal. az író álláspontja szerint a honlap a vállalkozás virtuális megjelenítése. Éppen ezért fontos, hogy kialakítsanak egy virtuális arculatot a vállalkozásnak, ami szakértelmet sugároz és a cég megjelenéséhez illeszkedik. Emellett a honlap előnye, hogy rengeteg funkcióval lehet ellátni. Ha szükséges, akkor a cég kialakíthat egy webáruházat vagy akár időpontfoglaló rendszert is. Megválaszthatja a különböző kapcsolattartási funkciókat. Végeredményként egy egyedi webalkalmazáshoz jut a vállalkozás, ami tökéletesen képviseli a céget az internetes világban, ahol jóval szélesebb potenciális vevőkörhöz képes eljutni a vállalkozás.

Ezen eszközökön kívül érdemes lehet még különböző cégekatalógusokba regisztrálni a vállalkozásunkat. Ezekben megadhatók a cég elérhetőségei, tevékenységi körei, esetenként még véleményezhetik is a felhasználók a vállalkozást. Elérhetőek ingyenes és fizetős cégekatalógusok is. Végül, de nem utolsó sorban a közösségi média is egy olcsó és egyszerű marketing eszköznek minősül a kisvállalkozások körében. A közösségi portálokon így kapcsolatot lehet teremteni a vevőkkel és potenciális vevőkörrel. Azonban ahhoz, hogy felfigyeljenek a cégre, aktív tartalomgyártásra van szükség. Érdemes aktuális információkat és érdekességeket megosztani a követőkkel. (REKETTYE, 2016)

2.3.4. Vállalati példák

Egyre több kisvállalkozás ismeri fel az online marketing fontosságát, illetve, hogy mennyire fontos a saját honlap megléte. A keresések során egyre több kisvállalkozás weboldalát találhatjuk meg. A Cápák között című reality show-ba is évről évre számos kisvállalkozás jelentkezik, akik közül néhányan weboldal létrehozásával szeretnék bővíteni vevőkörüket. Nem véletlen, hogy a Shoprenter kiemelt stratégiai partnere lett a műsornak. A DigitalHungary cikkéből derül ki, hogy a Shoprenter cég 2011-ben indult azzal a céllal, hogy segítse a kereskedőket az online értékesítésben. A cég nagyszerű kurzusokat és lehetőségeket kínál a vállalkozásoknak annak érdekében, hogy megerősítsék az online jelenlétüket. A műsorban láthattunk néhány vállalkozást, ahol a honlap megléte számos versenyelőnyt biztosított.

Több tanulmány is elérhető már a kisvállalkozásokkal és az online értékesítéssel kapcsolatban. Bernschütz Mária, Deés Szilvia, Kenéz András marketing esettanulmányai között is található erre példa. Egy zöldbolt, azaz öko kis- és nagykereskedés, ami az online kereskedelem segítségével vált igazán sikeressé. A vállalkozás meglepő módon először csak egy webshopot üzemeltetett és csak utána nyitott hagyományos boltokat. (BERNSCHÜTZ et al, 2016)

Napjainkra már vannak azonban olyan vállalatok, amelyek csak a virtuális térben vannak jelen. Ezekre a cégekre virtuális vállalatként hivatkozhatunk. Chikán Attila megfogalmazása szerint a virtuális vállalatok olyan vállalatok, amelyek a hagyományos működési formákat, fizikai tevékenységeket az információs technológia és menedzsment eszközeivel helyettesíti. Állítása szerint a virtuális vállalatoknak több formája lehet. Előfordulhat akár az is, hogy nincs székhelye vagy állandó alkalmazottja, de mégis versenyképes. (CHIKÁN, 2020)

Összességében kijelenthetjük, hogy az internet gazdasági szempontból rengeteg lehetőséget biztosít a hagyományos vállalatoknak és a virtuális vállalatoknak egyaránt. Segít a vevőkör bővítésében és tájékoztatásában. Emellett a vállalkozás egy vagy több tevékenysége is működtethető online felületeken. Néhány esetben nagyszerű eredményeket produkálhat az is, ha a hagyományos és virtuális világ adta lehetőségeket ötvözik a vállalkozások.

3. Anyag és módszer

3.1. VI-NI Transz Fuvarozó Kft. bemutatása

A VI-NI TRANSZ Fuvarozó Korlátolt Felelősségű Társaság 2009.10.12-én lett alapítva. Egyetlen telephellyel rendelkezik, ami Petőfibányán található. A cég alapítója 1990 óta áruszállítással foglalkozott más társasági formában, viszont korlátolt felelősségű társaságként csak a 2009-es évben kezdte meg működését két kamionnal.

A KKV száz százalékban magyar tulajdonban van, fő tevékenysége a közúti áruszállítás. Belföldi és nemzetközi fuvarokat teljesítenek. 2021-es árbevétel 134 millió forint volt, ami mindössze 7 fő bevonásával került elérésre. A cég különlegessége, hogy magyar tulajdonú vállalként TIR vám-okmánnyal rendelkezik. A TIR azaz Trafic International Routier eljárás szerint az átmenő, avagy tranzit országokban a fuvarozott áruk mentesítést nyernek a vámok és egyéb illetékek fizetése alól. Illetve a TIR-okmánnyal rendelkező szállítmányokat általában nem nyitják fel a határok átlépésekor. Ez az egyezmény kedvező kereteket biztosít a nemzetközi szállítmányozásra.

A cég, noha még csak 13 éve létezik korlátolt felelősségű társaságként, de már túlesett az első generáció váltáson is. Ez nagyban hozzájárult az elmúlt évek dinamikus növekedéséhez. Az alapításkor idősebb, főként lízingelt gépjárműpark fokozatos cseréje és újítása is megtörtént az elmúlt időszakban. Jelenleg 6 saját kamionból áll a flotta, amit egy bérelt kamion és három saját kisteherautó egészít ki.

A 2020/2021-es években is a fejlesztések mellett döntött a VI-NI Transz Fuvarozó Kft., így a koronavírus okozta nehézségeken túl is tovább tudott növekedni, megerősítve piaci pozícióját. A termelés visszaesése előtt a cégvezető felismerte a vírus okozta ellátási láncbeli problémákat, így a korábban főként jármű és elektronikai szállítmányokat részben élelmiszer alapanyag szállításra váltotta. Ezzel biztosította a gyárleállások esetén is folytonos megkereséseket.

A KKV célja a hosszútávon fenntartható és hatékony működés, ami biztosítja a partnerek számára az elvárt minőségű szolgáltatást és kollégák számára a biztos hátteret, stabil munkahelyet.

Az idei évben a kettő új kamion mellett a cég telephelye is teljes körű felújításon esett át. A szervizeléshez és üzemben tartáshoz korszerű gépek biztosítottak. Az adminisztráció

számára külön iroda került kialakításra. Az épület napelemeket nem, de teljes energetikai korszerűsítést kapott.

A felújítás lehetőséget biztosít, hogy a KKV több évtizedes szervíz tudását kamatoztassa. Ennek érdekében gumis műhely kialakítása történik a következő hónapokban. Az új tevékenység népszerűsítését segítő a szakdolgozatom témájaként megjelölt weboldal készítése, amit a közösségi médiába való szereplés fog kiegészíteni az induláskor.

A jelenlegi projekt célja, hogy egy olyan digitális felülettel gazdagodjon a vállalat, ami egyrészt a cég jelenlegi profilját ismerteti, referenciákat sorakoztat fel, másrészt lehetőséget ad új partnereknek a kapcsolatfelvételre. Az új tevékenység esetében pedig lehetőséget ad többek között a partnereknek az online időpontfoglalásra. Mindezek mellett pedig segíti a vállalkozás működését és adminisztratív tevékenységeit.

3.2. Webalkalmazás

3.2.1. Programtervezés a vállalkozással

Ahhoz, hogy a fejlesztés elkezdődhessen pontosan meg kellett beszélni a webalkalmazással kapcsolatos igényeket. Ahogyan azt a bevezetés során már említettem, ezeket pontosan egyeztettem a vállalkozással. A weboldalakat négyfelé tudjuk csoportosítani. Ebben a fejezetben részletesen is szeretném ismertetni a részekkel kapcsolatosan megfogalmazott igényeket.

Mivel a vállalkozáshoz két eltérő tevékenységi kör is kapcsolódik, ezért szükséges egy landing page, azaz nyitó oldal, ahol a felhasználó röviden megismerheti a vállalkozást és tovább léphet ahhoz a tevékenységi körhöz kapcsolódó weboldalra, ami miatt rátalált az oldalra. Ez a két tevékenységi kör a közúti áruszállítás és a gumiszervíz.

A fuvarozáshoz, mint korábban is említettem az elvárás egyetlen egy weboldal, ami inkább tájékoztató jellegű. Ezen a lapon a vállalkozás egy rövid bemutatkozást, elérhetőségeket, szolgáltatások bemutatását szeretné vizsgálni. Emellett az elérhetőségeknél biztosítani szeretnék azt, hogy az ügyfél akár egy űrlap kitöltésével is fel tudja velük venni a kapcsolatot. Mivel ehhez a területhez minimalista, átlátható tartalmat terveztek, így a végső elképzelés is egy egyoldalas weboldal maradt, ahol a menüpontok az adott a weboldal adott részére vezetik át a felhasználót.

A szerviz esetében egy sokkal komplexebb elképzelés volt. Ezért itt a megvalósításhoz már több weboldalra volt szükség. Az első megbeszélések szerint a következő témákhoz kapcsolódóan van szükség aloldalakra: bemutatkozás, szolgáltatások, portfólió, elérhetőségek és időpontfoglalás. Mivel ez a része a webalkalmazásnak a vállalkozás új tevékenységi köréhez készül és más vevőkört szólít meg, ezért egy interaktívabb felület kialakítása a cél, ahol például felhasználók kérdezhetnek a vállalkozástól, időpontot foglalhatnak vagy megnézhetik a foglalásuk részleteit. Ezek megfelelő kialakításához azonban jól fel kell mérni a folyamatokat, ami jelen esetben még nem lehetséges, mert még nem indult el a vállalkozás ezen része. Ebből az okból csak egy tesztrendszer kiépítése a cél, amit a későbbiekben könnyen lehet módosítani a folyamatnak megfelelően. Az időpontfoglaló rendszer kezdetben egy űrlapra épül, ahol a vevő kiválaszthatja a szabad időpontokból, hogy mikor szeretné leadni az autóját. Illetve a szolgáltatás részleteit megadhatja és a nyilvántartáshoz szükséges személyes adatait. A foglalást követően kapnak egy azonosítót, amit az erre létrehozott űrlapra beírva le tudják kérdezni a foglalásuk állapotát és adatait.

A foglalás elküldése után a vállalkozás ezt az admin felületen amennyiben el tudja vállalni jóvá tudja hagyni, szükség esetén módosíthat rajta vagy törölheti azt. Emellett az átláthatóság miatt a vállalkozás azokat a foglalásokat is hozzá szeretné adni az adatbázisához, amelyek személyesen vagy telefonon keresztül történtek, így az adminisztrátor számára is fontos kialakítani egy olyan felületet, ahol az adatbázishoz hozzáadhat új rekordokat, nemcsak módosíthatja vagy törölheti a meglévőket. A vállalkozás a foglalások státuszait az admin felületen szeretné módosítani és ellenőrizni.

Az időpontfoglaló rendszer felépítéséről is látszik, hogy az első verziós igények kezdetlegesebbek az admin felülettel kapcsolatban. Az alapvető információkat, mint például az elérhetőségek és nyitvatartás legyen képes a vállalkozás saját magának módosítani. Illetve az időpontfoglaló rendszerhez van szükségük hozzáféréshez. A későbbiekben, amikor a vállalkozás ténylegesen elindítja a gumiszervizt ezeket a felületeket párhuzamosan szeretnék tovább fejleszteni. Újabb és komplexebb funkciókat is szeretnék használni a felületen. Többek között a portfólióban látható referenciamunkák feltöltését, módosítását és törlését is erről a felületről szeretnék intézni. Hosszú távon cél, hogy a weboldal nagy részén lévő tartalmat az admin jogosultsággal rendelkező felhasználók tudják módosítani, így egy dinamikus testreszabható projekt létrehozása is célként fogalmazódott meg.

A weboldalnak fontos része a kinézet, így az ezzel kapcsolatos igényeket és elvárásokat is fontos volt egyeztetni. A dolgozatomban korábban kitértem rá, hogy olyan vállalkozást kerestem, amelynek még nem volt működő weboldala. Illetve fontos tényező az is, hogy a közúti áruszállítás miatt eddig főként business-to-business (B2B) kereskedés jellemezte a vállalkozást. Ezekből a tényekből az látszik, hogy a vállalkozás eddig nem fordított aktív figyelmet a vizuális megjelenésére, tehát nem volt egy brand guide, ami tartalmazta volna az arculattervezéshez szükséges információkat. Itt az a döntés született, hogy a teszt verziót online elérhető sablonokból építem fel és az éles verziónál alakítunk ki egy egységes arculatot a vállalkozás számára.

Ettől függetlenül a webalkalmazás frontend oldala reszponzívan lesz felépítve. Ez azért fontos szempont, mert napjainkra az emberek különböző eszközökről interneteznek és az oldal megjelenése ezáltal alkalmazkodni tud a felhasználó eszközhöz. A vállalkozásnak ez egy fontos szempont, ugyanis könnyen előfordulhat, hogy egyes vevők az okostelefonjukról böngészik majd a weboldalt és foglalnak időpontot, míg mások tabletről vagy laptopról. Illetve a reszponzivitás mellett fontos szempont a vállalkozás számára, hogy felhasználóbarát webalkalmazást működtessen. Ezért az alkalmazás felépítését a lehetőleg logikusan és átláthatóan szeretnék kialakítani, hogy a vevőknek a használata egyértelmű és egyszerű legyen.

3.2.2. Elvárások informatikai oldalról

A weboldalak felépítése a korábban megemlített csoportonként változik. De, ahol lehetséges ott igyekszem kialakítani egy egységes keretet a webalkalmazáson belül. Mivel egyedül az admin és a szerviz részhez kapcsolódik több weboldal, így ezeknél például a menüsört, lábléceket, de még a tartalmat eltároló törzs részt is újra fel lehet használni. Ezzel is gyorsítva a fejlesztési folyamatot.

A hosszabb terjedelmű weboldalaknál szeretnék beépíteni egy gombot, amire rákattintva a felhasználó visszatérhet az oldal tetejére, ezzel is megkímélve őt a felesleges vissza görgetéstől. A menüsoron megkülönböztetve jelölöm azt a pontot, amelyiket éppen a felhasználó nézi. Ez segíti a felhasználó tájékozódását a weboldalon belül.

A weboldalon megjelenő űrlapokból beérkező adatokat egy adatbázisban tárolom. A weboldal egyik, ha nem legfontosabb része az adatbázis. Ezért már a teszt verziónál is nagy hangsúlyt kell fordítani rá azért, hogy a későbbiekben könnyebben lehessen tovább bővíteni.

Mivel a vállalkozás a weboldalt főként a jövőbeli gumiszerviz miatt szeretné létrehozni, ezért első körben egy demo verziót alakítok ki, amiben a fentiek felül szabadkezet kaptam.

3.2.3. Adatbázisterv

A megbeszéltek alapján szintén elkezdtem kidolgozni az adatbázistervet. A webalkalmazás első verziójában az adatbázishoz 9 táblát tervezek létrehozni. A főhangsúly az időpont (appointments) táblán van. Ez tartalmazza a foglaláshoz kapcsolódó azonosítókat, amik a kapcsolatot biztosítják az autók (cars), naptár (calendar) és szolgáltatások (service) tábla között. Illetve az időpontra vonatkozó státuszt, amit a vállalkozás az admin felületről képes lesz módosítani.

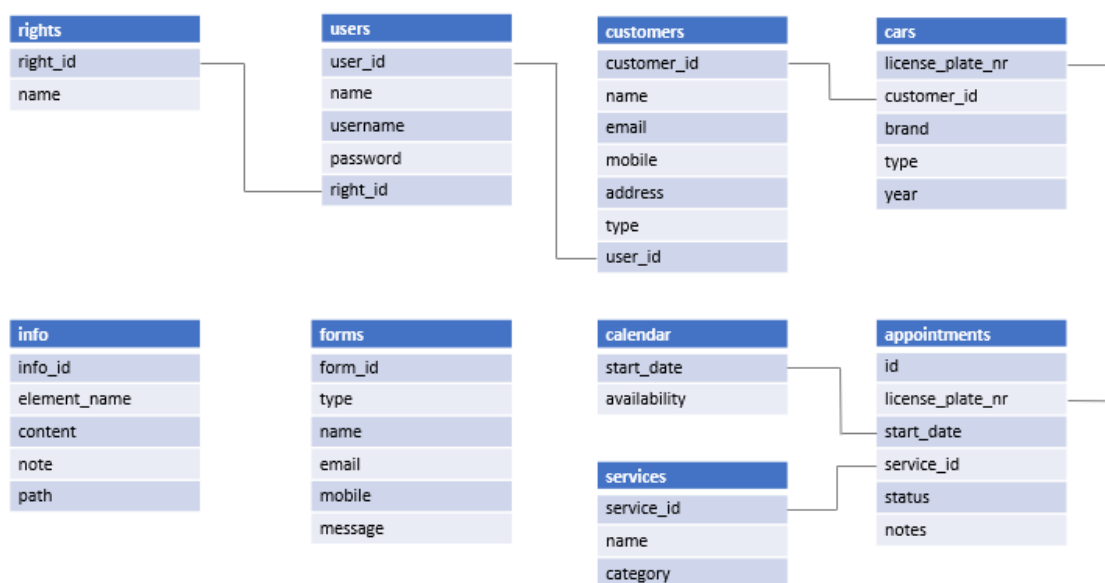
Az autók táblában alapvető információkat tárolok el az ügyfél autójáról. Ebben a táblában az egyedi kulcs a rendszám, ugyanis az minden autón egyedi. Ezenfelül ez a tábla kapcsolódik a tulajdonos azonosítón (customer_id) keresztül a vevők (customers) táblához. A vevők táblában főként az ügyfelek elérhetőségeit tárolom el. Emellett a későbbi fejlesztésekre felkészülve a felhasználó azonosítón keresztül tovább kapcsolódik a tábla a felhasználók (users) táblához. A későbbiekben ezáltal könnyebb lesz kialakítani a felhasználós rendszert. A felhasználók táblában az azonosító, név, felhasználó és jelszó mezők mellett került egy jogosultság azonosító. Ezáltal a mező által lehet megkülönböztetni majd a vállalkozáshoz tartozó személyeket, akik admin joggal rendelkeznek, illetve az ügyfeleket. Ezeket a jogosultságokat a rights táblában tárolom el.

A naptár táblában kerülnek eltárolásra azok az időpontok, amiket a vállalkozás megad, mint autó leadási időpontok. Illetve ezek néhány foglalás bekövetkezte után betelnek. Ezt az availability mezőben lehet követni.

A szolgáltatások is egy külön táblában kerülnek tárolásra. Ebben első körben a szolgáltatás azonosítója, amivel egyébként az időpontokhoz kapcsolódik, megnevezése és kategóriája került be. Későbbiekben valószínűleg ez is részletesebb tábla lesz.

Végül, de nem utolsó sorban van 2 különálló tábla. Az info táblában azok a weblap elemek adatai kerülnek, amiket az admin felületen lehet módosítani. Illetve létrehoztam egy táblát az űrlapból beérkező kitöltéseknek is. A következő ábrán látható az előző táblákból kiépített adatmodell.

3. ábra: Az adatbázisisterv



Forrás: saját szerkesztés

3.2.4. Felhasznált eszközök és technológiák

3.2.4.1. Kódszerkesztő: Visual Studio Code

Diplomamunkám fejlesztéséhez szükségem volt egy kódszerkesztő programra. A Visual Studio Code nevű kódszerkesztőt esett. Azért választottam ezt, mert számos bővítményt lehet benne telepíteni, így a frontend megvalósításához, keretrendszerek telepítéséhez és a különböző funkciók megírásához egyaránt alkalmas. A korábban megemlített IntelliSense és Debug segédprogramjai pedig extra előnyt jelentenek a fejlesztés során.

3.2.4.2. Programnyelvek

A webalkalmazás elkészítéséhez több programnyelv használatára is szükség van. A frontend elkészítése HTML-ben és CSS-ben történik. Fontos megjegyezni, hogy ezek nem tényleges programnyelvek. A HTML a szerkezetét írja le a weboldalaknak. A CSS stíluslap, ami a megjelenését, avagy kinézetét határozza meg a weboldalnak. Ahhoz, hogy a webalkalmazást a felhasználók ténylegesen használni tudják a backend oldalon több funkciót kellett megírni. Ezekhez főként a Python-t használtam. A fejlesztésem ideje alatt a Python 3.9 volt a legfrissebb verzió, így ezt alkalmazom. A projektben előfordulnak olyan elemek is, amiknek a működését a JavaScript biztosítja. A webalkalmazáshoz tartozik egy adatbázis. Az adatbázis kezeléséhez MySQL programnyelv használata szükséges. Ez az adatbázis egy szerveren lesz eltárolva. Ahhoz, hogy ez működni tudjon a XAMPP-ot fogom használni. Ez a szoftver lehetővé teszi, hogy Apache és SQL szervereket futtassak, amikre a rá tud kapcsolódni a megírt alkalmazás.

3.2.4.3. Keretrendszerek

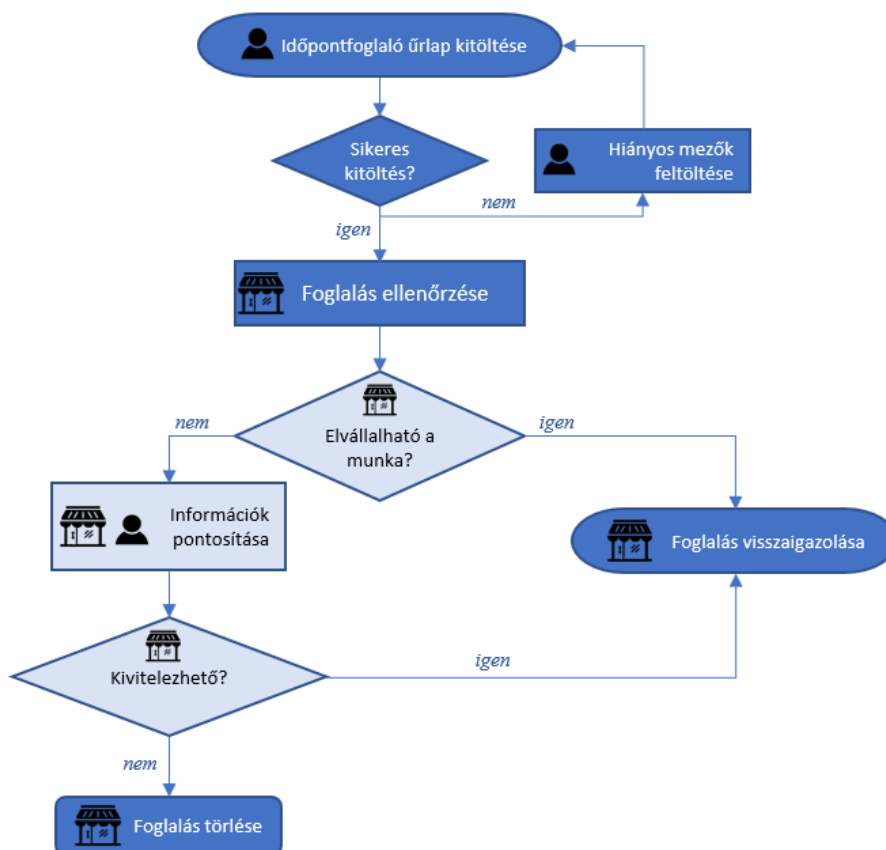
A diplomamunkám elengedhetetlen részei a keretrendszerek. Az alkalmazásban felhasznált keretrendszerek között van frontend és webes keretrendszer is. Frontend keretrendszer esetében a Bootstrap-re esett a választásom, ami az egyik legnépszerűbb keretrendszer ezen a téren. A fejlesztés ideje alatt a Bootstrap 5 volt a legfrissebb verzió, így a projekt kivitelezése során ezt fogom alkalmazni. A Python-hoz a kutatásom során két webes keretrendszert találtam a Django-t és a Flask-et. Végül a Flask-et fogom használni a fejlesztésben, mert a többi keretrendszerhez képest viszonylag szabad kezet nyújt.

3.2.5. Az időpontfoglaló rendszer működési folyamata

A weboldal legértékesebb része az időpontfoglaló rendszer, így ennek a folyamatát szeretném részletesebben is bemutatni. A folyamat a vevőtől indul, aki kitölti az űrlapot a foglaláshoz. Az űrlapban kiválasztja a számára megfelelő időpontot, szükséges szolgáltatást és megad további adatokat, mint például az elérhetőségeit és az autójának típusát vagy a szolgáltatás elvégzéséhez szükséges információkat. Szükség esetén megjegyzést is tehet a foglalása mellé. Sikeres kitöltés után a foglalás beérkezik a vállalkozáshoz. A vállalkozás ellenőrzi, hogy tudja-e kivitelezni a munkát. Amennyiben igen, akkor jóváhagyja a foglalást.

Ha tisztázó kérdések szükségesek, akkor felveszi a vevővel a kapcsolatot és az egyeztetés során módosíthatja a foglalást jóváhagyás előtt. Amennyiben nem sikerül az egyeztetés a vállalkozás törölheti a foglalást. A teszt verzióban a foglalás állapotát a weboldalon lehet lekérdezni, a foglaláskor kapott azonosítóval. Így kezdetben a folyamatból még nem vonhatóak ki az emberek a vállalkozás oldaláról, azonban a későbbiekben igény van további automatizációkra és értesítési lehetőségekre a foglalással kapcsolatban.

4. ábra: Az időpontfoglalás folyamata



Forrás: saját szerkesztés

A fenti ábrán a teszt verzióban lekódolt folyamat látható. A folyamatok mellett feltüntetett ábrák mutatják, hogy az ügyfélnek vagy a vállalkozásnak van-e teendője az adat cselekvés során. Illetve világoskékkel vannak jelölve azok a folyamatrészek, ahol még egyáltalán nincs bevonva a webalkalmazás.

4. Eredmények és értékelésük

4.1. A projekt állapota

Ebben a fejezetben ismertetni és értékelni fogom a szakdolgozatom eredményét, azaz az általam fejlesztett webalkalmazást. Ez a webalkalmazás egy valós projekt része. Ezért fontos megjegyezni, hogy a projekt az eddig elkészült alkalmazással még nem zárult le. Továbbra is kerülnek be a jelenlegi tesztverzióba az újabb és újabb funkciók. Az értékelés során a funkciókat három csoportba sorolom: kezdetleges kritériumok, hamarosan bekerülő funkciók és tervek a jövőre nézve.

4.2. Technológiák értékelése

A fejlesztés során a korábban ismertetett technológiákkal dolgoztam. Ahhoz, hogy ez a folyamat el tudjon kezdődni a már telepített Visual Studio Code-ban telepítettem a Pythont, majd a Flask-et. Ezt követően a kódszerkesztőben már tudtam kódolni a különböző programnyelveken. A webalkalmazás backend oldala Flask keretrendszer és Python használatával valósult meg. Ebben a kódban minden weboldalhoz funkciókat hoztam létre. Ezekben a metódusokban meghatároztam, hogy rendelkezik-e GET és POST metódussal a funkció. Amennyiben igen, akkor egy if elágazásban lekódoltam, hogy mi történjen ezek során. A GET metódus célja a dokumentumok visszakeresése. A POST módszert általában módosításokra használják, mint például törlés vagy frissítés.

A Flask keretrendszert meglepően egyszerűen össze lehetett kapcsolni az általam létrehozott MySQL adatbázissal. Ehhez a kód elején a flask_mysqldb könyvtárból kellett importálnom a MySQL modult. Ezt követően megírtam a kódot, amivel az applikációt összekapcsoltam az adatbázissal. Meg kellett adnom néhány információt az adatbázisról, mint például a felhasználónevet, jelszót és az adatbázis nevét. Ezután a funkciókban már könnyedén adhattam az adatbázis tábláihoz új rekordokat vagy módosíthattam azokat. Összességében a webalkalmazás fejlesztéséhez sikeresen tudtam alkalmazni a kiválasztott programnyelveket és keretrendszereket.

A fejlesztés során sikerült további bővítményeket találnom, amik tovább segítették a munkámat. Ezek közül egyet alkalmaztam is, ami a Flask keretrendszer egy bővítménye, a Session. Ezt főként az admin felület fejlesztésénél alkalmaztam. A Session bővítmény

segítségével könnyen megvalósítható, hogy a felhasználók csak belépés után férjenek hozzá különböző oldalakhoz. Emellett a menüsoron lévő menüpontok megjelenését is lehet korlátozni. Például, ha a felhasználó be van jelentkezve újabb menüpontok lesznek számára elérhetőek. Illetve a Harvard Egyetem egyik online elérhető kurzusából, a CS50-ből is több apró fejlesztési technikát elnéztem ennek kapcsán. a kurzusban látott anyag szerint könnyedén el tudtam készíteni az űrlapoknál felmerülő hibaüzeneteket, illetve még tovább korlátozni azon weboldalak elérhetőségét, amikhez bejelentkezés szükséges.

Ezen felül a template-k működtetésében óriási előnye volt a keretrendszernek, hogy a jinja2 sablon motort is könnyen lehet használni. Ezáltal a weboldalak fejlécéből, menüsorából és láblécéből készítettem egy sablont, amit mindig csak az aktuális oldal tartalmával kellett bővíteni.

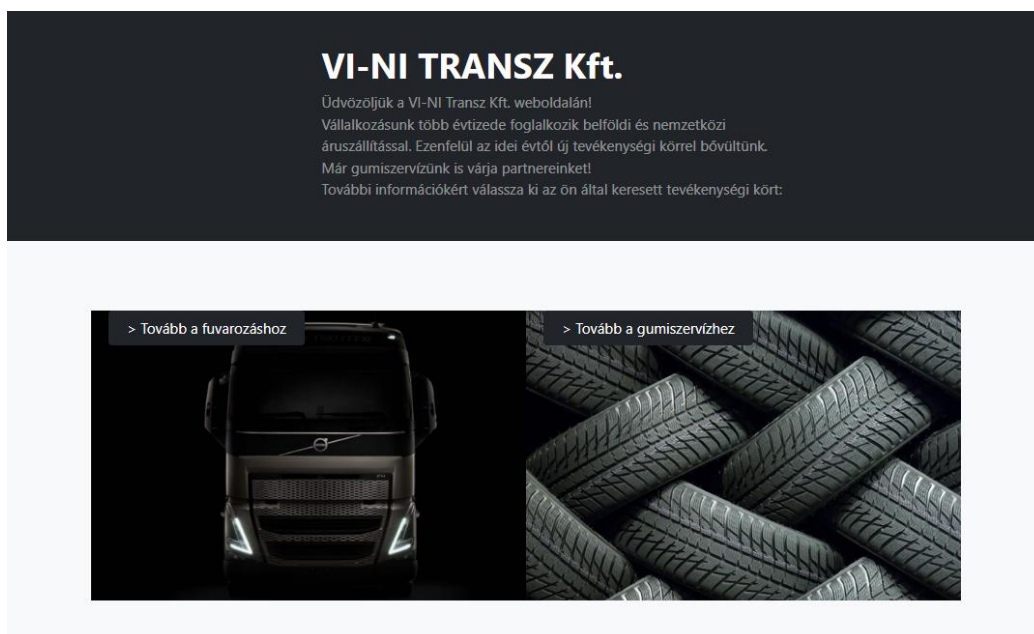
Az alkalmazás front-end részét a teszt verzióban a bootstrap keretrendszer segítségével valósítottam meg. Emellett a startbootstrap.com oldalon ingyenesen elérhető modern business template átalakításával készítettem el az oldalakat. Ezeket a következő alfejezetben bővebben is bemutatom a projekt leírás során említett csoportosítás szerint.

4.3. A kritériumok megvalósítása

4.3.1. Nyitó oldal

A landing page egy nagyon egyszerű koncepciót kapott. Felépítését tekintve 3 blokkból áll az oldal. A tetején a látogató egy rövid bemutatkozást olvashat a vállalkozásról. Az oldal alsó felén pedig továbbléphet választása szerint a fuvarozással vagy a gumis műhellyel kapcsolatos tartalmakhoz. A felhasználók a 2 tevékenységi körhöz kapcsolódó oldalakról bármikor visszaléphetnek a nyitó oldalra a főoldal menüpont segítségével. A következő kép a nyitó oldalt ábrázolja.

5. ábra: A nyitólap megjelenése



Forrás: saját szerkesztés

4.3.2. Fuvarozó

Ehhez a tevékenységi körhöz az igényeknek megfelelően egy egyoldalas weboldalt készítettem. Az oldalon található menüsor, de ez nem új oldalakra irányít át, hanem az adott menüponthoz kapcsolódó tartalomhoz görget le a weboldalon. Ahogy a lentebb szereplő ábra is mutatja, a megvalósításához a különböző témakörökhöz kapcsolódó elemeknél egyedi azonosítót, avagy id-t adtam meg. Ezt követően a menüpontokhoz tartozó hiperhivatkozásoknál ezekre az azonosítókra hivatkoztam.

6. ábra: A navigáció működését biztosító kód

```
<div class="collapse navbar-collapse" id="navbarSupportedContent">
  <ul class="navbar-nav ms-auto mb-2 mb-lg-0">
    <li class="nav-item"><a class="nav-link" href="#index">Főoldal</a></li>
    <li class="nav-item"><a class="nav-link" href="#anchor-about">Rólunk</a></li>
    <li class="nav-item"><a class="nav-link" href="#anchor-products">Szolgáltatások</a></li>
    <li class="nav-item"><a class="nav-link" href="#anchor-contact">Kapcsolat</a></li>
  </ul>
</div>
</nav>
<!-- Bemutakozás/Rólunk - Fuvarozás-->
<header class="py-5">
  <div class="container px-5">
    <div class="row justify-content-center">
      <div class="col-lg-8 col-xxl-6">
        <div class="text-center my-5" id="anchor-about">
```

Forrás: saját szerkesztés

Az oldal másik különlegessége, a lap tetejére visszatérés funkció. Mivel egy hosszabb terjedelmű oldal készült el, ezért egy kevés görgetés után a jobb alsó sarokban megjelenik egy gomb. Ezzel a felhasználó könnyedén vissza tud lépni a lap tetejére. Mivel ez az elem a frontend-en van jelen, ezért a működéséhez szükséges kódsort javascript-ben írtam meg. A gomb formai megjelenését pedig CSS-ben. Ezt a w3schools oldalon elérhető tananyagok alapján készítettem el. A következő képen összefoglalva látható a gombhoz tartozó JavaScript és CSS kód, illetve a gomb különböző megjelenési formái.

7. ábra: Vissza a lap tetejére gomb működése és megjelenési formái

```
// Vissza a lap tetejére - gomb
// (Forrás w3schools)

let mybutton = document.getElementById("myBtn");

// Görgetés
window.onscroll = function() {scrollFunction()};

// Ha 20px-el lentebb görget a felhasználó, akkor megjelenik a gomb
Complexity is 4 Everything is cool!
function scrollFunction() {
  if (document.body.scrollTop > 20 || document.documentElement.scrollTop > 20) {
    mybutton.style.display = "block";
  } else {
    mybutton.style.display = "none";
  }
}

// Ha rákattint a gombra, akkor felgörget az oldal tetejére
function topFunction() {
  //A Safari böngészőhöz ezt másképpen kell lekezelni
  document.body.scrollTop = 0; // Safari
  document.documentElement.scrollTop = 0; // Chrome, Firefox, IE and Opera
}

/*A gomb tulajdonságai
(vissza a lap tetejére)*/
#myBtn {
  /*elhelyezés*/
  display: none;
  position: fixed;
  bottom: 75px;
  right: 30px;
  z-index: 99;

  /*megjelenés*/
  border: none;
  outline: none;
  background-color: blue;
  color: white;
  cursor: pointer;
  padding: 15px;
  border-radius: 10px;
}

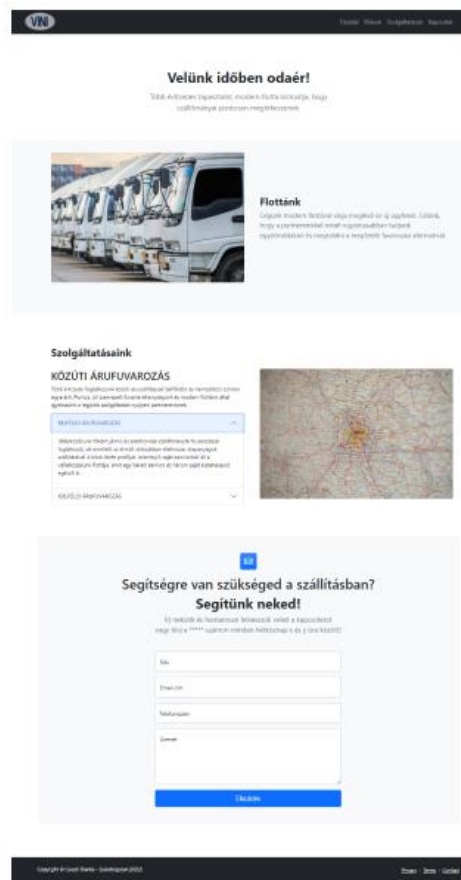
#myBtn:hover {
  background-color: #555;
}
```



Forrás: saját szerkesztés

A weboldal tartalmát tekintve, a látogató olvashat a vállalkozásról, a flottáról és az elérhető szolgáltatásokról. Emellett az oldal alján egy űrlap is elérhető számukra, amin keresztül felveheti a kapcsolatot a vállalkozással. Az űrlapon az ügyfél megadhatja az adatait és megírhatja kérdéseit a cég felé. Miután ezt elküldi, az űrlapba beírt mezők bekerülnek az adatbázisba. Ezt a cég az admin felületen látja, így könnyen fel tudja venni a kapcsolatot a potenciális partnerrel.

8. ábra: Fuvarozás weboldal layout



Forrás: saját szerkesztés

Mint korábban is említésre került, ezek a tartalmak egy egyoldalas weboldalban szerepelnek, ezért a weboldalt böngészve nem látható a teljes layout. A képen ez a felépítés összesítve látható.

4.3.3. Műhely

Ahogy ez várható volt, jelenleg az alkalmazásnak ez a legterjedelmesebb része, ugyanis ehhez a területhez tartozik a legtöbb oldal és funkció. Fontos megjegyezni, hogy mivel még a vállalkozás ezzel a tevékenységi körével még nincs a piacon, ezért a webalkalmazásban jelenleg teszt adatok vannak, amik az alkalmazás használhatóságának tesztelését teszik lehetővé.

A menüpontokat tekintve a következő oldalak lettek kialakítva:

Rólunk: Ezen az oldalon a látogató a cég bemutatkozását tekintheti meg. Olvashat a cég történetéről, telephelyéről, értékrendjéről és megismerheti az ott dolgozókat.

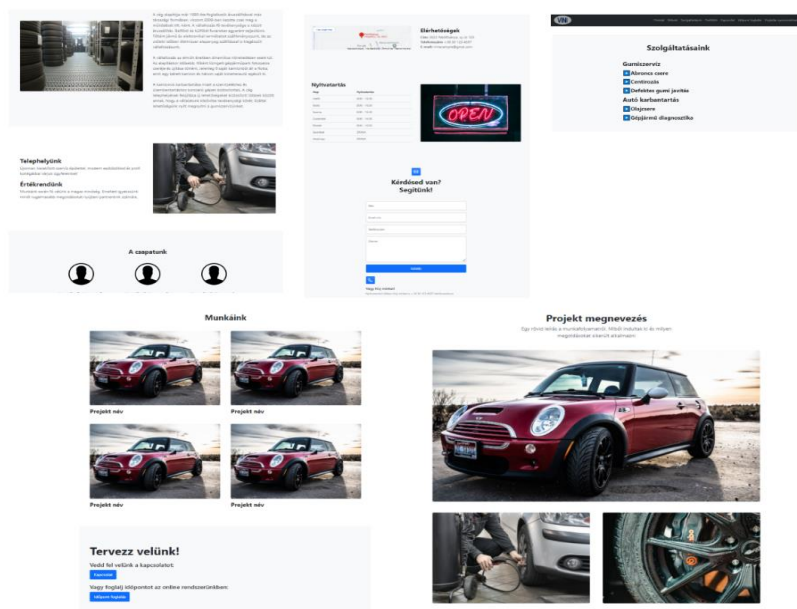
Szolgáltatások: Itt a felhasználó a szolgáltatásokat csoportokra bontva megtekintheti.

Portfólió: A referencia munkáknak kialakított album, ahol egy-egy munkára kattintva részletesebb képet és információt is kaphat a látogató a munkákról.

Kapcsolat: A felhasználó itt megtalálja az elérhetőségeket, nyitvatartást. Emellett lehetősége van kapcsolatfelvételre egy űrlapon keresztül a fuvarozásnál látható kérdőívhez hasonlóan.

Mint a következő képek is mutatják, ezek az oldalak még viszonylag statikusak. Az oldalakon olvasható tartalom jelenleg még a HTML kódban található. Azonban a későbbi fejlesztéseknél szeretném összegyűjteni azokat a tartalmakat, amik folyamatosan változhatnak. Ezt követően pedig létre szeretnék hozni az admin felületen egy platformot, ahol a vállalkozás ezeket tudja frissíteni.

9. ábra: A műhelyhez kapcsolódó weboldalak szerkezetei
(Balról jobbra: Rólunk, Kapcsolat, Szolgáltatások, Portfólió, Referencia Munka 1)



Forrás: saját szerkesztés

A következő oldalak között már több dinamikus oldalt is szeretnék bemutatni. Ilyen felületek például az időpontfoglaló rendszerhez kialakított foglaló, ellenőrző oldalak és a későbbiekben bemutatandó admin felület legtöbb része.

Időpontfoglaló rendszer:

Ezen az oldalon csak egy űrlap található, de mégis az alkalmazás legfontosabb részét alkotja. A felhasználó ezen keresztül tud időpontot foglalni. Ki kell választania az időpontot és meg kell adnia az autóval, választott szolgáltatással kapcsolatos információkat, illetve az elérhetőségeit. Ezáltal az űrlapban számos input mező található. Vannak szövegdobozok és lenyíló listák. A lenyíló listák többségébe az adatbázisból töltődnek be a választási lehetőségek. A következő képeken látható az említett időpontfoglaló űrlap a legördülő listákkal, illetve a foglalást követő megerősítő üzenet.

10. ábra: Időpontfoglalás és a kapott megerősítő üzenet

The image shows a web form titled 'Időpontfoglalás' (Appointment Booking). The form contains several input fields and dropdown menus. The 'Foglalható:' (Bookable) dropdown is set to '2022-09-20 08:00:00'. The 'Szolgáltatások:' (Services) dropdown is set to 'Gumiabroncs javítás' (Tire repair). The 'Forgalmi rendszám:' (Traffic license plate) field is empty. The 'Márka:' (Brand) field is empty. The 'Típus:' (Type) field is empty. The 'Autó évjárata:' (Car age) field is empty. The 'Vevő kategória:' (Customer category) dropdown is set to 'Magánszemély' (Private person). The 'Név:' (Name) field is empty. The 'Email cím:' (Email address) field is empty. The 'Telefonszám:' (Phone number) field is empty. The 'Cím:' (Address) field is empty. The 'Üzenet:' (Message) field is empty. A blue button at the bottom is labeled 'Foglalás leadása' (Submit booking). To the right of the form, a confirmation message is displayed: 'Foglalás elküldve, a foglalásod azonosítója: 10' (Booking sent, your booking ID is: 10).

Foglalható:	2022-09-20 08:00:00
Szolgáltatások:	2022-09-20 08:00:00
	2022-09-20 08:30:00
	2022-09-21 08:00:00
	2022-09-21 08:30:00
Forgalmi rendszám:	2022-09-22 08:00:00
Rendszám	2022-09-22 08:30:00

Forrás: saját szerkesztés

Ahhoz, hogy a weboldalon megtalálható űrlapokban a felhasználó a legördülő listákban tartalmakat láthasson a weboldal megjelenítését tartalmazó funkciónak már a GET metódusában össze kellett gyűjteni, majd át adni a template-nek a megfelelő listákat. Ehhez első lépésként az adatbázisból le kellett kérnem a megfelelő adatokat a listákhoz. Például

lekérdezést kellett írnom ahhoz, hogy egy tömbben megkaphassam a szolgáltatás táblából a megnevezéseket és az azonosítókat vagy a szabad időpontokat. Ezeket az adatokat a Jinja2 segítségével át tudtam adni a template fájl-nak, amiben egy for ciklus beolvasta és kiírta ezeket az adatokat a dropdown lista elemeibe, ahogy ezt a következő kódsorok is mutatják.

11. ábra: Legördülő listák működését biztosító kód

Kód az applikációban:

```
# GET metódus
# dropdown listában szereplő adatok összegyűjtése

cur = mysql.connection.cursor()

services= cur.execute("SELECT * FROM services")
if services>0:
    services=cur.fetchall()

dates= cur.execute("SELECT * FROM calendar")
if dates>0:
    dates=cur.fetchall()

cur.close()

return render_template('calendar.html',services=services,dates=dates)
#####
```

Adatok betöltése a sablonba:

```
<div class="input-group mb-3">
  <div class="input-group-append">
    <label class="input-group-text" for="inputDate">Foglalható:    </label>
  </div>
  <select class="custom-select" id="inputDate" name="date">
    {% for date in dates %}
    <option value="{{date[0]}}">{{date[0]}}</option>
    {% endfor %}
  </select>
</div>

<!-- Szolgáltatás input-->
<div class="input-group mb-3">
  <div class="input-group-append">
    <label class="input-group-text" for="inputService">Szolgáltatások:</label>
  </div>
  <select class="custom-select" id="inputService" name="service">
    {% for service in services %}
    <option value="{{service[0]}}">{{service[1]}}</option>
    {% endfor %}
  </select>
</div>
```

Forrás: saját szerkesztés

Ha a felhasználó az űrlapon leadja a foglalását, akkor ugyanezen funkció POST metódusa fog lefutni. Ebben az esetben az input mezők adatait eltárolom különböző változóba. Legördülő listák esetében a szolgáltatások azonosítói szerepelnek az elemnél megadott értékeknél. Ezáltal a felhasználók nem tudnak különböző módszerekkel nem létező szolgáltatásokra időpontot foglalni, mert a form leadásánál az id értékével fog a kód tovább dolgozni. Majd a változókat felhasználva az adatbázis időpontok táblájához hozzáadom az

új foglalást. Ez a gyakorlatban azonban nem ilyen egyszerű. Ellenőriznem kellett először, hogy az autó és a vevő már szerepel-e az adatbázisban. Amennyiben igen, akkor az azonosítókat lekérdeztem és ezeket adtam hozzá a foglalás részleteihez. Ha nem, akkor az időpont létrehozása előtt az ügyfél és az autóját is hozzá kellett adnom a megfelelő táblákhoz, hogy ezt követően az azonosítókat hozzá tudjam adni a foglalás adataihoz.

Jelenleg az űrlapban csak az email mező tartalmát ellenőrzi a kód, a többi szövegdobozt nem. Ezt a későbbiekben tovább szeretném fejleszteni. Például lehetne az telefonszám formátumokat ellenőrizni. Ezáltal több gépelési problémát meg lehetne előzni és valószínűleg a felhasználók is magabiztosabban töltenék ki az űrlapot.

Foglalás nyomon követése:

Ez a felület azért jött létre, hogy az ügyfél a foglalás után kapott azonosítóját megadva ellenőrizni tudja a foglalását és annak státuszát. Technikailag nézve ezen az oldalon egy űrlap van elhelyezve. A szövegdobozban a felhasználó megadja a foglalási azonosítót. Majd a lekérdezés elindítását követően a kód lefuttat egy lekérdezést, aminek eredménye az adott azonosítóval rendelkező foglalás adatai. A funkció eközben átirányít egy oldalra, amin ezek az adatok kerülnek kiírásra.

12. ábra: Státusz ellenőrzés és az ellenőrzés után kapott adatok

The screenshot shows a web interface for checking booking status. At the top, it says 'Státusz ellenőrzés' and 'Kérem adja meg a foglalási azonosítót!'. Below this is a text input field containing '10' and a blue button labeled 'Lekérdezés'. Below the button, there is a section titled 'Foglalás adatai' which contains a table with the following data:

Foglalás azonosító	Rendszám	Leadási dátum	Státusz
10	XYZ-123	2022-09-20 08:00:00	Elküldve

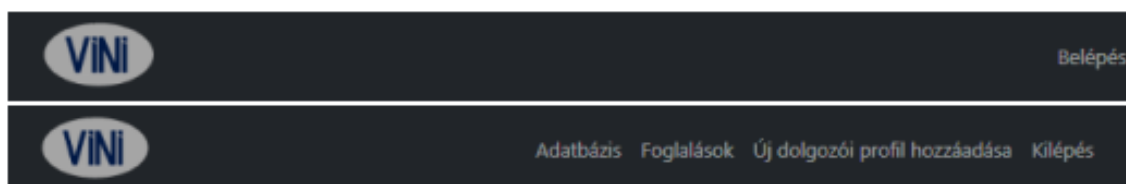
Forrás: saját szerkesztés

A fenti képeken is látszik, hogy ez a felület még nagyon kezdeti állapotban van. Későbbiekben érdekes kiegészítés lehet akár a korábbi foglalások lekérdezése is vagy még több információ biztosítása az ügyfél számára.

4.3.4. Admin felület

Az admin felület nem érhető el közvetlen a menüsorról. Ehhez a felhasználóknak tudniuk kell az elérési útvonalat. Ahhoz, hogy különböző weboldalakhoz hozzáférjenek be kell lépniük. Ezért belépés nélkül hiába ismerik az elérési útvonalat, nem fogják tudni használni az érintett oldalakat. Ahhoz, hogy ez működjön a korábban említett Session bővítményt használtam. A belépés során elmentésre kerül a felhasználó azonosítója session id-ként és ezáltal hozzáférhet a korlátozott tartalmakhoz. Az admin felület még csak 4 oldalból áll: Belépés, regisztráció, adatbázis és időpontfoglalások. A felhasználó belépés nélkül csak a belépés menüpontot látja a menüsoron. Bejelentkezést követően a menüsor tartalma lecserélődik. Hozzáférést kap az adatbázishoz, időpontfoglaló rendszer kezeléséhez, létrehozhat új dolgozói profilt és a munkamenet végén ki is jelentkezhet. Amennyiben valaki ezeket az oldalakat URL cím alapján próbálja megnyitni és nincs bejelentkezve, akkor automatikusan a bejelentkezési felületre irányítja az applikáció. A következő képeken látható a menüsor belépés előtt és után elérhető menüpontjai.

13. ábra: A navigáció belépés előtt és után



Forrás: saját szerkesztés

A belépés és regisztráció során különböző feltételek alapján ellenőriztem az űrlapok tartalmát. Eszerint nem megfelelő kitöltés után különböző hibaüzeneteket kap a felhasználó. Ilyen esetek lehetnek például az eltérő jelszavak, hiányos mezők vagy a hibás felhasználónév és jelszó kombináció.

Az admin felület használatához létrehozott felhasználók az adatbázisban a felhasználók táblában tárolódnak el. Ebben a táblában elhelyeztem egy jogosultság mezőt. A dolgozók admin jogosultsággal rendelkeznek. Ez a mező főként a webalkalmazás későbbi életében fog fontos szerepet játszani. Ugyanis, ha a későbbiekben a vállalkozás a vevőinek is szeretne felhasználói rendszert kialakítani, akkor a jogosultság mezővel a felhasználókat megfelelően el lehet lehet különíteni.

A jelszó az adatbázisban jelenleg még nem biztonságosan van eltárolva, ugyanis a konkrét értéket kapja meg az adatbázis. Azonban a `werkzeug.security`-t importálva a kódban már tesztelem, hogyan lehet a jelszót hash-elni. Ezáltal különböző kalkulációk alapján egy kódsor tárolódna el az adatbázisban a jelszó mezőben. Ezt a bejelentkezés során a későbbiekben valószínű, hogy egy függvény segítségével tudnám összehasonlítani, hogy érvényes jelszót adott-e meg a felhasználó.

Belépés:

Ezen az oldalon tudnak belépni a vállalkozás munkavállalói az admin felületre a felhasználónevük és jelszavuk megadásával. Amennyiben új dolgozót szeretnének regisztrálni, a regisztráció menüpontra kattintva átléphetnek a regisztrációs oldalra.

Új dolgozói profil hozzáadása

Amennyiben új dolgozó érkezik vagy új felhasználót kell létrehozni, ezen az oldalon regisztráció segítségével megethetik. A regisztráció során meg kell adni a felhasználónevet, nevet és a jelszót kétszer. Sikeresen regisztráció után a belépés oldalra visszalépve tud a felhasználó bejelentkezni.

14. ábra: A belépési és regisztrációs űrlapok

The image shows two web forms side-by-side. The left form, titled 'Belépés', contains two input fields: 'Felhasználónév:' and 'Jelszó:', followed by a blue 'Belépés' button. The right form, titled 'Regisztráció', contains four input fields: 'Felhasználónév:', 'Név:', 'Jelszó:', and 'Jelszó megint:', followed by a blue 'Regisztráció' button.

Forrás: saját szerkesztés

Adatbázis:

Itt a cég webalkalmazásához kapcsolódó adatbázis tábláinak tartalmát böngészhetik át a dolgozók. Jelenlegi formájában még csak egyszerű táblázatokba vannak kiírva a táblák rekordjai a template motor segítségével.

15. ábra: Adattáblák megjelenése az admin felületen

Szolgáltatások

Megnevezés	Főcsoport
Gumiabroncs javítás	Gumiabroncs
Gumiabroncs csere	Gumiabroncs

Beérkezett üzenetek és foglalások

Foglalások

Foglalás azonosító	Rendszám	Leadási dátum	Szolgáltatás azonosító	Státusz	Megjegyzés
1	SQL-123	2022-09-20 08:00:00	1	Teljesítve	Téli nyári gumi csere

Forrás: saját szerkesztés

A képen látható, hogy ez még nem a leghasznosabb formája az adatbázis megjelenítésének. Úgy gondolom, hogy a későbbiekben ezt a felületet érdemes lesz még további funkciókkal ellátni. Hasznos lenne, ha a dolgozók tudnák a táblákat szűrni, keresni bennük vagy szükség esetén kiexportálni adatokat.

Időpont foglalások:

Az oldalon a dolgozók megnézhetik a foglalásokat és azok adatait. Ehhez az adatbázis tábláinál használt módszerrel egy táblázatba tettem a lekérdezést eredményét. Ahhoz, hogy a foglalások kiírásánál ne csak azonosítókat lássanak a dolgozók egy összetett lekérdezést kellett írnom, amiben az azonosítókat összekötöm a megfelelő táblákkal. Ezáltal az összes szükséges adat, amit az űrlap kitöltésénél az ügyfél megadott, az a táblázatban megjelenik az admin számára, ahogyan ezt a következő kép is szemlélteti.

16. ábra: Foglalások kilistázva a felületen

Beérkezett foglalások

Foglalások

Foglalás azonosító	Státusz	Leadási dátum	Vevő azonosító	Neve	Email	Mobil	Cím	Kategória	Rendszám	Márka	Típus	Évjárat	Szolgáltatás azonosító	Szolgáltatás	Megjegyzés
1	Teljesítve	2022-09-20 08:00:00	1	Test Elek	testelek@gmail.com	+36301234567	3200 Gyöngyös, Mátrai út 43.	Magánszemély	SQL-123	BMW	Q4	2020	1	Gumiabroncs javítás	Téli nyári gumi csere
2	Megerősítve	None	1	Test Elek	testelek@gmail.com	+36301234567	3200 Gyöngyös, Mátrai út 43.	Magánszemély	SQL-123	BMW	Q4	2020	2	Gumiabroncs csere	test

Forrás: saját szerkesztés

Emellett lehetőségük van új foglalás hozzáadására, foglalás törlésére és a státusz szerkesztésére is. Ezekhez különböző űrlapokat hoztam létre. Az új foglalás hozzáadása az ügyfelek foglalási űrlapjával megegyezően működik. Ezek a funkciók lenyitható listákban vannak elrejtve. A következő kép ezt a listát ábrázolja összecsukva.

17. ábra: A lenyitható lista űrlapokkal

Időpont hozzáadása	▼
Státusz módosítás	▼
Időpont törlése	▼

Forrás: saját szerkesztés

A foglalás törlésénél a dolgozó kiválasztja azt a foglalási azonosítót, amit törölni szeretne, majd az űrlapnál található gomb segítségével a törlés meg is történik. Az űrlap leadásával lefut egy POST metódus, ami az adatbázisból törli a legördülő listából kiválasztott azonosítójú foglalást. Ez egy nagyon egyértelmű és egyszerűen kezelhető űrlap, ahogy ezt az ábra is mutatja.

18. ábra: Űrlap az időpont törléséhez

Időpont hozzáadása	▼
Státusz módosítás	▼
Időpont törlése	^

Időpont törlése

Foglalás azonosítója: 1 ▼

Foglalás törlése

Forrás: saját szerkesztés

Az előző folyamat logikáját követi a státusz módosítás is. A következő képen ez látható. A törléshez képest csak egy szövegdobozzal egészült ki az űrlap. Az átíráshoz ki kell választani az azonosítót, majd egy legördülő listából ki kell választani az új státuszt és az űrlapnál található gombra kattintva lefut a POST metódus, amiben megtalálható az a kódrész, ami frissíti az adatbázisban a kiválasztott azonosítóval rendelkező foglalás státuszát.

19. ábra: Űrlap az időpont státuszának módosításához

Státusz módosítás

Időpont - státusz módosítás

Foglalás azonosítója: 36 ▼

Státusz: Elküldve ▼

- Elküldve
- Feldolgozás alatt
- Megerősítve
- Teljesítve

Státusz módosítása

Forrás: saját szerkesztés

Ezeket az űrlapokat a későbbiekben tovább szeretném fejleszteni, hogy még praktikusabbak legyenek. Jelenleg a legördülő listákban alapértelmezett értéként a legkisebb azonosító jelenik meg. Ez azért nem ideális, mert a mellette lévő üres szövegdobozt figyelmetlenségből a dolgozó rá tudja publikálni a jelenlegi tartalomra. Ebből kifolyólag érdemes lenne a státuszokat vizsgálnom a státusz tényleges lefrissítése előtt vagy pedig egy legördülő listát létrehoznom az előre definiált státusz megnevezésekből. Illetve érdemes lehet egy üres mezőre átváltanom az azonosító kezdőértékét, mert a törlés esetében a figyelmetlenség egy foglalás törléséhez is vezethet. Továbbgondolva a szituációt érdemes lehet az éles verziónál biztonsági mentéseket létrehozni.

Továbbá a státusz módosítás alapján el kezdtem kialakítani a foglalás módosítására lehetőséget adó űrlapot is. Ez a funkció első ránézésre a hozzáadáshoz létrehozott űrlaphoz nagyon hasonlít, mert ugyanaz az űrlap került felhasználásra annyi kiegészítéssel, hogy meg kell adni a foglalás azonosítóját. Az űrlap input elemeiben szeretném megjeleníteni az aktuálisan kiválasztott foglalás adatait, hogy a dolgozónak ezeket ne egyesével kelljen visszakeresnie. Az űrlap beküldését követően pedig lefutna egy frissítés, ami a kiválasztott rekord mezőinek tartalmát frissíti az itt megadott adatokra.

4.4. Értékelés és következő lépések

Ebben az alfejezetben szeretném listászerűen összegezni a projekt elején megadott igényeket, majd ezt követően értékelni azok megvalósítását. Ez az első nagyobb webes projektem és úgy gondolom, hogy ebben a pontban nem lehet minden kritériumnál egy egyszerű igennel vagy nemmel eldönteni, hogy az teljesült-e. Ezért az igényeken végig haladva szeretném röviden értékelni ezeket.

Az oldal legyen négy részre bontva

Ez a kritérium könnyen teljesíthető volt., ugyanis ez alapján kezdtem el az egész projektet.

A nyitó oldal felépítése

Ahogy az igényeknél meg is fogalmazódott, jelenleg úgy épül fel a nyitólap. A tetején a látogató olvashat a vállalkozás, majd az oldal alsó fele meg van felezve és tovább léphet a fuvarozáshoz vagy a műhelyhez kapcsolódó oldalakra.

Legyen egy egyoldalas weboldal a fuvarozással kapcsolatos tevékenységi körnek

Sikeresen létrehoztam ezt a weboldalt egy menüsorral, amiben a menüpontok az adott tartalomhoz irányítják az oldalon a felhasználót.

Bemutakozás, elérhetőségek, szolgáltatások, kapcsolat menüpontok a fuvarozáshoz kapcsolódóan

A látogató jelenleg ezeket a menüpontokat látja a menüsoron, illetve ezekhez található tartalom is a weboldalon.

Bemutakozás, szolgáltatások, portfólió, elérhetőségek, időpontfoglaló menüpontok a gumis műhelyhez kapcsolódó oldalakon

Bár a menüsoron ezek a menüpontok jelen vannak és weboldalak is vannak. A megoldás a tökéletestől távol van, ugyanis jelenleg próbaként teszt adatok szerepelnek az oldalakon.

Belépés és regisztráció az admin felületre

A regisztrációs és belépő űrlap is működik a webalkalmazásban. Szükség esetén különböző hibaüzeneteket ad, illetve sikeres belépés után a korlátozott tartalmak is elérhetővé válnak a menüsorról.

Az adatbázis táblái láthatóak az admin felületen

Maga ez a tény teljesül, azonban véleményem szerint a platform így magában még nem eléggé praktikus. A továbbiakban érdemes lesz megvitatnom a korábban említett plusz funkciók bevezetését.

Alapvető információk módosítása

Sajnos ez a funkció még nem elérhető az admin felületen, azonban az adatbázis elő van készítve arra, hogy ez kivitelezhető legyen.

Az időpontfoglaló rendszer kezelése

Az admin felületen az időpontfoglaló rendszerben leadott foglalások egy táblázatban látható az összes rendelkezésre álló információval. Funkciókat tekintve: a státuszt lehet módosítani, a foglalásokat lehet törölni és lehet új foglalásokat hozzáadni a rendszerhez.

Emellett már zajlik a foglalás módosító funkció bevezetése, illetve az előző alfejezetekben említett apró részletek implementálása, amik még egyszerűbbé teszik a felület használatát.

Az adatbázisterv alapján létrehozott adatbázis jól működik a webalkalmazással

Az adatbázistervben elkészített adatbázis modell megfelelően működik. Azonban fontos megjegyezni, hogy a lehetséges távoli fejlesztések miatt ez az adatbázis még bővülni fog különböző mezőkkel.

Reszponzív felület

A Bootstrap frontend keretrendszer segítségével könnyedén kialakítottam a rezponzív felületet.

Felfelé görgető gomb

Ezt a kritériumot a w3schools tananyagai alapján gyorsan meg tudtam oldani. A gomb egyelőre a fuvarozással kapcsolatos oldalon elérhető.

Sablon alkalmazása

A Jinja-t template motort alkalmazva ez könnyedén kivitelezhető volt.

Felhasználóbarát kivitelezés

Véleményem szerint ezen a ponton még van mit fejleszteni. Több helyen látok fejlesztési potenciált. Például az űrlapok töltését is lehet tovább egyszerűsíteni. Emellett kellő tesztelés után a különböző menüpontok és tartalmak elhelyezését is érdemes lehet átgondolni, mert nem valószínű, hogy a potenciális vevőkör is ugyanazzal a szemlélettel nézi a weboldalt, mint a weboldal megrendelője.

Céges design kialakítása

Ahogy a szakdolgozatomban korábban kitértem rá, a cégnek nincs jelenleg saját style guide-ja, ami alapján a weboldal kinézetét szerkeszteni tudtam volna. Ezért egy kezdetleges logót és színvilágot kezdtünk közösen kialakítani a vállalkozással annak érdekében, hogy jobb felhasználói élményt nyújtson a teszt oldal is. A következő kollázs ez eddig meglévő terveket, színvilágot mutatja. Azonban ezek még nem kerültek teljesen bele a webalkalmazásra.

20. ábra: Inspiráció a cég arculattervezéséhez



Forrás: saját szerkesztés

Az űrlapok működése

A webalkalmazás jelenlegi állapotában a benne lévő űrlapok működnek. Azonban vitathatatlan, hogy lehet rajta fejleszteni. Egyes űrlapok az adatbázis tartalmát módosítják. Ezeknek a tartalmát fontos lenne, jobban felülvizsgálni. Emellett a lehetséges elírásokat vagy hibalehetőségeket kiküszöbölni. Azonban, hogy több ilyen scenario-t összegyűjtsék a weboldalt teszteltetnem kell több emberrel.

Összességében úgy gondolom, hogy sikerült egy jól működő webalkalmazást létrehoznom. Azonban, ha már adatott a lehetőség, hogy a projekt egy teszt verzióval induljon az éles helyett ezt érdemes kihasználnom akár tesztelésre, akár még több funkció bevezetésére.

5. Következtetések és javaslatok

A webalkalmazás elkészítését egy projektként éltem és élem meg jelenleg is. Ebből kifolyólag nemcsak a webprogramozáshoz köthető következtetéseket és javaslatokat szeretnék tenni ebben a fejezetben, hanem a projektezésre vonatkozóan is.

Ez volt az első nagyobb webprogramozással kapcsolatos projektem, így várható volt, hogy rengeteg tanulságot le fogok tudni vonni ebből. Ezekből néhányat most szeretnék ismertetni.

Számomra a legnagyobb tanulság az volt, hogy az igényeket a projekt elején pontosan meg kell határozni. A kritériumokat és igényeket bár meghatároztuk, de nem eléggé pontosan. Így azt vettem észre, hogy sokszor nem igazán pontosan tudtam, hogy milyen konkrét output-ra lesz szüksége a vállalkozásnak 1-1 funkciót illetően. Illetve a pontos meghatározások hiányában sokszor túl sok apró részletre koncentráltam, ami nem is fogalmazódott meg kritériumként. Leegyszerűsítve a konklúziómat: nem elég meghatározni a tárgyat, annak pontos tulajdonságait is definiálni kell, hogy ténylegesen egy dologra gondoljon a fejlesztő és a megbízó is.

Továbbá a frontend kialakítása sem ment olyan könnyen, mint azt gondoltam. Ez több okból adódott. A tesztverzióban egyelőre lebutított teszt adatokkal dolgoztam, így nem voltak tényleges adatok ezáltal a fejlesztés során hiányzott a tényleges tartalom megléte. Ahhoz, hogy megkapjam a vázat sokszor különböző placeholder-eket kellett használnom. Úgy gondolom, hogy ez a része a fejlesztésnek sokkal hatékonyabb lehetett volna, ha már a valós tartalommal készítek egy pontos vázlatot akár a különböző stílus elemekkel együtt. Utólag értékelve a vázak elkészítése akár Figma-ban nem vett volna el sok időt a projekttől és még akár a kritériumok pontos definiálásának hiánya is kibukott volna belőle.

A folyamatok modellezése is hasonlóan fontos része a fejlesztésnek. Sajnos az időpontfoglaló esetében egyelőre még csak egy fiktív folyamatot tudtam lemodellezni és az alapján lefejleszteni a rendszert. Nyilván ennek is meg van az előnye, mert így már ismerem milyen további fejlesztési lehetőségeket tudok beépíteni a rendszerbe. Azonban megvan a hátulütője is. Előfordulhat, hogy a műhely elindulása után kiderül, hogy nem ez az optimális folyamat és át kell alakítani a rendszert.

Az eredmények értékelésénél a teljesített kritériumokat és a hamarosan megvalósításra kerülő funkciókat helyeztem előtérbe. Ennek a fejezetnek a hátralévő részében viszont

szeretnék a távolabbi fejlesztési lehetőségekre fókuszálni. A kutatás és fejlesztés során rengeteg ötlet merült fel bennem, amivel színesíteni lehetne a projektet.

Mivel a hétköznapijaimban adatvizualizációs szakértőként dolgozom, így az első gondolatom is ezzel kapcsolatos. A fuvarozáshoz kapcsolódó weboldal vizuális élményét lehetne gazdagítani egy térképes vizualizációval. Ehhez az elmúlt jelentősebb útvonalakat, úti célokat összegyűjtve könnyen ki lehet alakítani egy adatforrást, amiből egy adatvizualizációs szoftverrel (például: Tableau) könnyen egy látványos ábrát készíthetünk.

Természetesen informatikai oldalról is számos lehetőség van még a projektben. Amennyiben lesz kereslet az időpontfoglaló rendszerre, érdemes lehet a későbbiekben felhasználói rendszert kialakítani. Ezáltal a vevőknek nem kell minden foglalásnál megadniuk az adataikat, illetve az autók specifikus tulajdonságait. Elég lenne egyszer regisztrálniuk, majd ezt követően felvihetik a rendszerbe az autóikat, amiket a foglalás során egyszerűen hozzáadhatnak a foglaláshoz. Emellett a felhasználói profil lehetővé tenné, hogy a vevő visszanézhesse foglalásait.

Továbbá az időpontok megnyitását és lezárását is lehetne automatizálni. Ebben az esetben érdekes lehet az új időpontok automatikus hozzáadása, illetve az időpontokra érkező foglalások korlátozása is.

A felhasználói rendszer kialakításánál a további lehetőségeket is figyelembe kell venni. Napjainkra az emberek rengeteg weboldalon rendelkeznek felhasználói fiókokkal, amit képtelenség számontartaniuk sok esetben, ezért érdemes lehet megfontolni a fiókok összehangolását a Google fiókokkal. Ez egyre több weboldalon elterjedt funkcióvá nőtte ki magát.

Végül, de nem utolsó sorban a marketing. Ahhoz, hogy minél többen és gyorsan rá találjanak a vállalkozás weboldalára a cégnek marketing tevékenységeket kell folytatnia. A diplomamunkám elején folytatott kutatásból is látszott, hogy a kisvállalkozásoknak érdemes a közösségi médián aktívan jelen lenni. Ezek a platformok viszonylag olcsó marketing lehetőségeket rejtenek magukban. Amellett, hogy kapcsolatot tudunk tartani a vevőkkel rajta és népszerűsíteni tudjuk a vállalkozásunkat, még a weboldal meglétére is fel tudjuk hívni az ügyfelek figyelmét. Emellett akár különböző promóciókkal is még tovább népszerűsíthetjük a weboldalt. Előbb-utóbb ezen tevékenységek miatt az ügyfélkör szokásává válik az

időpontfoglaló rendszer használata. Így kialakul egy kör, akik miatt érdemes lesz még tovább fejleszteni a rendszert.

A fent említett fejlesztési lehetőségeken kívül még számos ötletet fel lehetne sorolni, ugyanis a témát rengeteg szemszögből meg lehet közelíteni. Akár még a vállalkozás által használt más rendszerekkel való szinkronizálásig is tovább lehet vinni a projektet. Azonban fontos, hogy mielőtt egy komplexebb funkciót vagy fejlesztést elindítunk a webalkalmazásban előtte fel kell mérjük, hogy az az új projekt megtérül-e vagy több időt, energiát és pénzt vesz-e igénybe, mint amekkora hasznot hoz a vállalkozás számára. Összességében ez a projekt rávilágított, hogy miért van fontos szerepe a gazdaságinformatikus szaknak. Úgy gondolom, hogy a szakmánkból adódóan képesek vagyunk megítélni, hogy ezek a beruházások megtérülnek-e gazdaságilag, illetve szükség esetén le is tudjuk fejleszteni ezeket.

6. Összefoglalás

A szakdolgozatom témája a webalkalmazás készítés. Ennek keretében egy hazai kisvállalkozás számára készítettem egyet. A dolgozatom elkészítése előtt egy rövidebb kutatást végeztem a weboldalak gazdasági lehetőségeiről, különös figyelmet fordítva a mikro-, kis- és középvállalatokra és a lehetőségeikre. A KKV-k jellemzően kevés összeget tudnak marketing tevékenységekre fordítani. Emiatt fontos, hogy olyan eszközöket válasszanak, amik hatékonyak és nem túl drágák. Az online marketinges eszközök között több ilyen megoldás is létezik, mint például a keresőmotor optimalizálás, kereső hirdetés vagy a közösségi média. Azonban ezek mellett egy weboldal működtetése több új gazdasági lehetőséget ad a vállalkozásnak, még ha a kialakítása nem feltétlen olcsó is.

Mivel a diplomamunkám eredménye egy webalkalmazás, emiatt informatikai oldalról is végigjártam egy weboldal elkészítésének folyamatát. Először a webprogramozáshoz szükséges alapvető információkat, majd az ehhez használható programnyelveket és keretrendszereket kutattam. A webalkalmazások frontend oldalának elkészítéséhez mindenféleképpen HTML és CSS használata szükséges. Ezek adják a weboldal szerkezetét és kinézetét. Ezt tovább lehet szépíteni például JavaScript-ben lekódolt interakciókkal vagy különböző frontend oldali keretrendszerek használatával. Backend oldalról ezzel szemben sokkal több opciót találtam. A fejlesztő több programnyelv közül is választhat, mint például a PHP és a Python. Illetve ezekhez különböző keretrendszereket tud felhasználni a fejlesztés során. A webalkalmazáshoz kapcsolódó adatbázis kérdése sem olyan egyértelmű, ugyanis számos adatbázisrendszer létezik.

Ezen információk birtokában elkezdtem a saját projektemen dolgozni. A webalkalmazást egy fuvarozó cégnek készítem, ami a közeljövőben új tevékenységgel szeretné bővíteni a profilját. Mivel a vállalkozás a kamionjait magas színvonalon tartja karban, ezért úgy döntött, hogy új piacra tör be és kialakít egy gumis műhelyt, ahol a kamionok karbantartása mellett az ügyfeleknek is új szolgáltatásokat kínál. A vállalat bemutatása után részletezem a webalkalmazással kapcsolatosan felmerült igényeket és kritériumokat. A webalkalmazás azon túl, hogy bemutatja a vállalkozást, lehetőséget ad az ügyfeleknek a kapcsolat felvételre. Emellett a webalkalmazás legkomplexebb része az időpontfoglaló rendszer, amit a gumis műhelyhez készítek. Illetve igyekszem egy komplex admin felületet kiépíteni. Ezzel megalapozni azt, hogy a projekt végterméke minél dinamikusabban tudjon működni.

Ezt követően kiválasztottam a számomra megfelelő technológiákat a fejlesztéshez és a diplomamunkám nagy részét a webalkalmazás készítése jelentette. A webalkalmazást végül a Python, HTML, CSS, SQL és JavaScript programnyelveket használva készítettem el. Emellett több keretrendszert is alkalmaztam. Frontend oldalról a Bootstrap-et, backend oldalról pedig a Flask-et. Korábban ilyen nagy webes projektet nem készítettem Python-ban a Flask keretrendszert alkalmazva, így rengeteg új információra leltem szert és több könyvtárat is megismertem. Ennek következtében dolgoztam többek között a Session bővítménnyel és a Jinja template motorral is.

A dolgozatom elkészítése során több következtetést és javaslatot is levontam. A diplomamunkámra én több szempontból tekintek. Nagy tudást ad informatikai oldalról a webprogramozás ismertetése által. Emellett kiegészül egy kis gazdasági oldallal, ahol a KKV-k világába és a kisvállalkozások marketing tevékenységébe nyújt betekintést. Illetve az alkalmazás elkészítése során megtapasztaltam, hogy milyen egy webprogramozással kapcsolatos projekt, így ennek az egyes fázisairól is igyekeztem levonni a megfelelő következtetéseket. Például megértettem, hogy miért fontos, hogy a projekt elején pontos kritériumokat és igényeket határozzunk meg vagy hogy mennyire fontos a folyamatok részletes ismerete.

Végül, de nem utolsó sorban pedig szerettem volna felhívni a figyelmet az aktuális trendekre és irányvonalakra. Napjainkra nem elég, ha egy vállalkozás csak készít magának egy weboldalt. Szükséges, hogy legyen egy saját arculata, amiről a vevő könnyen felismeri a céget. Legyen ez akár egy logó, akár egy színekombináció. Emellett az online marketing is egyre nagyobb szerepet tölt be a vállalkozások életében, ugyanis az emberek rengeteg időt töltenek el az interneten, így napjainkra már majdnem könnyebb őket ezen a platformon elérni, mint a hagyományos médiában. Illetve a webalkalmazások elkészítésénél fontos figyelembe venni a felhasználók szokásait és igényeit is. A kezdetben ehhez lelkes tesztelőket kell toborozni több korosztályból, több területről. Majd ezt követően az internetes szokásaikat is érdemes figyelniük. Manapság az emberek nagy részének például rengeteg felhasználói fiókja van különböző weboldalakon. Szinte már nem is tudják ezeket számontartani. A szakdolgozatomban többek között erre és hasonló felvetésekre is kitérek, mint jövőbeli tervek az általam fejlesztett webalkalmazás életében.

7. Irodalomjegyzék

Szakirodalom

1. Fehér K. (2017). *Digitalizáció és új média* [on-line]. Akadémiai Kiadó, Budapest. Elérhetőség/hozzáférés: <https://mersz.hu/feher-digitalizacio-es-uj-media/> [olvasva: 2022. szeptember 10.]
2. Szabó B. (2013). *Webprogramozás I.* [on-line]. Eszterházy Károly Egyetem, Eger. Elérhetőség/hozzáférés: <https://mek.oszk.hu/14000/14068/pdf/14068.pdf> [olvasva: 2022. szeptember 10.]
3. M. del Sales-Zárate, G. Alor-Hernández, R. Valencia-García, L. Rodríguez-Mazahua, A. Rodríguez-González, J. L. López Cuadrado (2015). *Analyzing best practices on Web development frameworks: The lift approach* [on-line]. Elérhetőség/hozzáférés: <https://www.sciencedirect.com/science/article/pii/S0167642314005735> [olvasva: 2022. szeptember 10.]
4. Rekettye G., Törőcsik M., Hetesi E. (2016). *Bevezetés a marketingbe* [on-line]. Akadémiai Kiadó, Budapest Elérhetőség/hozzáférés: <https://mersz.hu/rekettye-torocsik-hetesi-bevezetes-a-marketingbe/> [olvasva: 2022. szeptember 10.]
5. Nagy G. (2011). *Web programozás alapismeretek* [on-line]. Ad Librum Kiadó, Budapest Elérhetőség/hozzáférés: <https://mek.oszk.hu/09700/09713/09713.pdf> [olvasva: 2022. szeptember 10.]
6. M. Lutz (2001). *Programmin Python, Second Edition* [on-line]. O'Reilly & Associates, Inc., USA. Elérhetőség/hozzáférés: <https://books.google.hu/books?id=c8pV-TzyfBUC&lpg=PR11&dq=python&lr&hl=hu&pg=PR1#v=onepage&q=python&f=false> [olvasva: 2022. szeptember 10.]
7. J. C. Meloni, M. Morrison (2010). *Tanuljuk meg a HTML és a CSS használatát 24 óra alatt* [on-line]. Kiskapu Kft. – Budapest Elérhetőség/hozzáférés: https://people.inf.elte.hu/zirtaai/html_ebooks/html_css_24ora_alatt.pdf [olvasva: 2022. szeptember 10.]
8. Radványi T. (2014). *Adatbázisrendszerek* [on-line]. Hallgatói Információs Központ Elérhetőség/hozzáférés: https://dtk.tankonyvtar.hu/xmlui/bitstream/handle/123456789/3476/2011-0038_22_radvanyi_hu.pdf?sequence=1&isAllowed=y [olvasva: 2022. szeptember 10.]

9. H. Cook, M. Adrian, R. Greenwald, A. Ronthal, P. Russom (2021). *Magic Quadrant for Cloud Database Management Systems* [on-line]. Elérhetőség/hozzáférés: <https://www.gartner.com/doc/reprints?id=1-28F5QL1U&ct=211213&st=sb> [olvasva: 2022. szeptember 10.]
10. S. S. Gaikwad, P. Adkar (2019) *A Review Paper on Bootstrap Framework* [on-line]. Elérhetőség/hozzáférés: <https://irejournals.com/formatedpaper/1701173.pdf> [olvasva: 2022. szeptember 10.]
11. J. Forcier, P. Bissex, W. Chun (2008). *Python Web Development with Django* [on-line]. Elérhetőség/hozzáférés: <https://books.google.hu/books?id=M2D5nnYlmZoC&lpg=PT31&dq=django&lr&hl=hu&pg=PP1#v=onepage&q=django&f=false> [olvasva: 2022. szeptember 10.]
12. M. Grinberg (2018). *Flask Web Development* [on-line]. O'Reilly Media , USA Elérhetőség/hozzáférés: <https://books.google.hu/books?id=cVIPDwAAQBAJ&lpg=PT25&dq=flask&lr&hl=hu&pg=PT2#v=onepage&q=flask&f=false> [olvasva: 2022. szeptember 10.]
13. Kacsukné Bruckner L., Kiss T. (2019). *Bevezetés az üzleti informatikába.* [on-line]. Akadémiai Kiadó, Budapest. Elérhetőség/hozzáférés: <https://mersz.hu/kacsukne-bruckner-kiss-bevezetes-az-uzleti-informatikaba//> [olvasva: 2022. szeptember 10.]
14. Avornicului M., Gubán Á., Seer L., Szőcs I. (2019). *Az internet és lehetőségei.* [on-line]. Akadémiai Kiadó, Budapest. Elérhetőség/hozzáférés: <https://mersz.hu/avornicului-guban-seer-szocs-az-internet-es-lehetosegei//> [olvasva: 2022. szeptember 10.]
15. Bányai E., Novák P. (2016). *Online üzlet és marketing.* [on-line]. Akadémiai Kiadó, Budapest. Elérhetőség/hozzáférés: <https://mersz.hu/banyai-novak-online-uzlet-es-marketing//> [olvasva: 2022. szeptember 10.]
16. Rekettye G. (2016). *Marketing a magyar kisvállalatoknak.* [on-line]. Akadémiai Kiadó, Budapest. Elérhetőség/hozzáférés: <https://mersz.hu/rekettye-marketing-a-magyar-kisvallalatoknak//> [olvasva: 2022. szeptember 10.]
17. Bernschütz M., Deés Sz., Kenéz A.. (2016). *Marketing esettanulmányok.* [on-line]. Akadémiai Kiadó, Budapest. Elérhetőség/hozzáférés: <https://mersz.hu/bernschutz-dees-kenez-marketing-esettanulmanyok//> [olvasva: 2022. szeptember 10.]

18. Chikán A. (2020). *Vállalatgazdaságtan*. [on-line]. Akadémiai Kiadó, Budapest.
Elérhetőség/hozzáférés: <https://mersz.hu/chikan-vallalatgazdasagtan/> [olvasva: 2022. szeptember 10.]

Egyéb források:

19. M. Arrington (2008). *Sun Picks Up MySQL For \$1 Billion; Open Source Is A Legitimate Business Model* [on-line].
Elérhetőség/hozzáférés: https://techcrunch.com/2008/01/16/sun-picks-up-mysql-for-1-billion-open-source-is-a-legitimate-business-model/?guccounter=1&guce_referrer=aHR0cHM6Ly93d3cuZ29vZ2xlLmNvbS8&guce_referrer_sig=AQAAAIcCjMsdAEyQ1Ppy3ti3XfV_WWY_dRaq_naPpOI2G5UQI97pLQJC598qk6kGI0yzMETAmDMF22GSjLi6_WSswxSeEBwVap0vo67K1purPmNW-O5GlwBkh3nmQ4k7JcXJvnoxkA5IgWak_843C9GaJvgCNztgFwKhweVkX8l6daH [olvasva: 2022. szeptember 10.]
20. Visual Studio Code dokumentáció
Elérhetőség/hozzáférés: <https://code.visualstudio.com/docs>
[olvasva: 2022. szeptember 10.]
21. 2004. évi XXXIV. törvény:
Elérhetőség/hozzáférés: <https://net.jogtar.hu/jogszabaly?docid=a0400034.tv>
[olvasva: 2022. szeptember 10.]
22. Kishalakat segít a Shoprenter (cikk)
Elérhetőség/hozzáférés: <https://www.digitalhungary.hu/e-kereskedelem/Kishalakat-segit-a-Shoprenter/14431/>
[olvasva: 2022. szeptember 10.]
23. How To Create a Scroll back to top button
Elérhetőség/hozzáférés:
https://www.w3schools.com/HOWTO/howto_js_scroll_to_top.asp
[olvasva: 2022. szeptember 24.]
24. CS50x: Introduction to Computer Science
Elérhetőség/hozzáférés: <https://cs50.harvard.edu/x/2022/>
[olvasva: 2022. szeptember 24.]

8. Táblázatok és ábrák jegyzéke

1. ábra: Gartner féle Mágikus kvadráns az adatbázis kezelő rendszerekről	10
2. ábra: Keresőhirdetés a Google-ben	13
3. ábra: Az adatbázisterv	22
4. ábra: Az időpontfoglalás folyamata	24
5. ábra: A nyitólap megjelenése	27
6. ábra: A navigáció működését biztosító kód	27
7. ábra: Vissza a lap tetejére gomb működése és megjelenési formái	28
8. ábra: Fuvarozás weboldal layout	29
9. ábra: A műhelyhez kapcsolódó weboldalak szerkezetei	30
10. ábra: Időpontfoglalás és a kapott megerősítő üzenet	31
11. ábra: Legördülő listák működését biztosító kód	32
12. ábra: Státusz ellenőrzés és az ellenőrzés után kapott adatok	33
13. ábra: A navigáció belépés előtt és után	34
14. ábra: A belépési és regisztrációs űrlapok	35
15. ábra: Adattáblák megjelenése az admin felületen	36
16. ábra: Foglalások kilistázva a felületen	36
17. ábra: A lenyitható lista űrlapokkal	37
18. ábra: Űrlap az időpont törléséhez	37
19. ábra: Űrlap az időpont státuszának módosításához	38
20. ábra: Inspiráció a cég arculattervezéséhez	41

9. Nyilatkozatok

9.1. Nyilatkozat a szakdolgozat nyilvános hozzáféréséről és eredetiségéről

NYILATKOZAT

a szakdolgozat nyilvános hozzáféréséről és eredetiségéről

A hallgató neve: **Csuzdi Bianka**
A Hallgató Neptun kódja: **WNWH6X**
A dolgozat címe: **Webalkalmazás készítése a VI-NI TRANSZ Fuvarozó Kft. részére**
A megjelenés éve: **2022**
A tanszék neve: **Műszaki Intézet**

Kijelentem, hogy az általam benyújtott szakdolgozat egyéni, eredeti jellegű, saját szellemi alkotásom. Azon részeket, melyeket más szerzők munkájából vettem át, egyértelműen megjelöltem, s az irodalomjegyzékben szerepeltettem.

Ha a fenti nyilatkozáttal valótlan állítottam, tudomásul veszem, hogy a Záróvizsga-bizottság a záróvizsgából kizár és a záróvizsgát csak új dolgozat készítése után tehetek.

A leadott dolgozat, mely PDF dokumentum, szerkesztését nem, megtekintését és nyomtatását engedélyezem.

Tudomásul veszem, hogy az általam készített dolgozatra, mint szellemi alkotás felhasználására, hasznosítására a Magyar Agrár- és Élettudományi Egyetem mindenkor szellemi tulajdonkezelési szabályzatában megfogalmazottak érvényesek.

Tudomásul veszem, hogy dolgozatom elektronikus változata feltöltésre kerül a Magyar Agrár- és Élettudományi Egyetem könyvtári repozitóri rendszerébe.

Kelt: 2022 év október hó 27 nap


Hallgató aláírása

9.2. Konzultációs nyilatkozat

Konzultációs nyilatkozat

Hallgató neve: Csuzdi Bianka
Szak, szakirány: Gazdaságinformatika
Konzulens tanár neve: Sike Zoltán

Az első személyes konzultáció

Időpont: 2022.09.13

Téma: A tervezett felépítés és fejezetek véglegesítése; az elkészült munka áttekintése

A konzulens tanár aláírása:



A második személyes konzultáció

Időpont: 2022.09.27

Téma: Az alkalmazás működésének áttekintése, a szükséges módosítások, finomítások megbeszélése

A konzulens tanár aláírása:



A harmadik személyes konzultáció

Időpont: 2022.10.13

Téma: A szakdolgozat végleges formájának áttekintése; irodalomjegyzék és ábrajegyzék pontosítása

A konzulens tanár aláírása:



A nyilatkozatot a szakdolgozathoz/diplomadolgozathoz kötelezően csatolni kell!

KONZULTÁCIÓS NYILATKOZAT

A **Csuzdi Bianka** (hallgató Neptun azonosítója: **WNWH6X**) konzulenseként nyilatkozom arról, hogy a szakdolgozatot¹ áttekintettem, a hallgatót az irodalmi források korrekt kezelésének követelményeiről, jogi és etikai szabályairól tájékoztattam.

A szakdolgozatot a záróvizsgán történő védeésre javaslom / nem javaslom².

A dolgozat állam- vagy szolgálati titkot tartalmaz: igen nem^{*3}

Kelt: Gyöngyös év október hó 27. nap


Belső konzulens

¹ A megfelelő dolgozattípus meghagyása mellett a többi típus törlendő.

² A megfelelő aláhúzendő.

³ A megfelelő aláhúzendő.

10. Mellékletek

10.1 2004. évi XXXIV. törvény

2004. évi XXXIV. törvény - a kis- és középvállalkozásokról, fejlődésük támogatásáról MKOGY

Hatály: közlönyállapot (2004.V.1.)

1. oldal

2004. évi XXXIV. törvény

a kis- és középvállalkozásokról, fejlődésük támogatásáról¹

Az Országgyűlés a kis- és középvállalkozások tőkeerejének növekedése melletti elkötelezettség szem előtt tartásával, fejlődésük előmozdítása, verseny- és foglalkoztatási képességük megőrzése, hazai és uniós szinten történő növelése, ezáltal a gazdaság és a társadalom kiegyensúlyozott fejlesztésének elősegítése, továbbá a statisztikai adatgyűjtésnek az uniós módszerekkel való összehangolása, és az adatok, gazdasági folyamatok összehasonlíthatósága, valamint a kisvállalkozók közötti kapcsolatok áttekinthetőbbé, átláthatóbbá tétele érdekében a következő törvényt alkotja:

I. Fejezet

BEVEZETŐ RENDELKEZÉSEK

A törvény célja

1. § A törvény célja a mikro-, kis- és középvállalkozások meghatározása, valamint a fejlődésük előmozdítását szolgáló állami támogatások összefoglalása, és ezáltal olyan gazdasági feltételek kialakítása, amelyek hosszú távon, hazai és uniós szinten egyaránt biztosítják a verseny- és foglalkoztatási képesség növekedését, a versenyhátrányok csökkenését, valamint a vállalkozások Európai Unió követelményeihez való felzárkózását.

2. § A törvény hatálya a mikro-, kis- és középvállalkozásokra (a továbbiakban: KKV), valamint a KKV-k támogatására és az azzal kapcsolatos adatszolgáltatásra terjed ki.

A kis- és középvállalkozások meghatározása

3. § (1) KKV-nak minősül az a vállalkozás, amelynek
a) összes foglalkoztatotti létszáma 250 főnél kevesebb, és
b) éves nettó árbevétele legfeljebb 40 millió eurónak megfelelő forintösszeg, vagy mérlegfőösszege legfeljebb 27 millió eurónak megfelelő forintösszeg, továbbá
c) megfelel a (4) bekezdésben foglalt feltételeknek.
(2) Kisvállalkozásnak minősül az a vállalkozás, amelynek
a) összes foglalkoztatotti létszáma 50 főnél kevesebb, és
b) éves nettó árbevétele legfeljebb 7 millió eurónak megfelelő forintösszeg, vagy mérlegfőösszege legfeljebb 5 millió eurónak megfelelő forintösszeg, továbbá
c) megfelel a (4) bekezdésben foglalt feltételeknek.
(3) Mikrovállalkozásnak minősül az a vállalkozás, amelynek összes foglalkoztatotti létszáma 10 főnél kevesebb, és megfelel a (2) bekezdés b)-c) pontjában foglaltaknak.
(4) Egy vállalkozás akkor minősül KKV-nak, ha abban az állam, az önkormányzat vagy az (1) bekezdés szerinti vállalkozáson kívül eső vállalkozások tulajdoni részesedése - jegyzett tőke vagy szavazati jog alapján - külön-külön vagy együttesen sem haladja meg a 25%-ot.
(5) Nem kell alkalmazni a (4) bekezdésben meghatározott korlátozást abban az esetben, ha az (1)-(3) bekezdés szerinti vállalkozás tulajdonosai olyan befektetők, amelyek nem rendelkeznek többségi irányítást biztosító befolyással.

¹ A törvényt az Országgyűlés a 2004. április 26-i ülésnapján fogadta el.