**Szekszárdi SZC Apáczai Csere János Szakgimnáziuma,**

**Szakközépiskolája és Kollégiuma**

**Az 54 213 05 számú Szoftverfejlesztő szakképesítés**

**záródolgozata**

**Benzinkút Webshop**

**Készítette:**

**Szalai Áron**

**Dombóvár**

**2019**

**Tartalomjegyzék**

[Bevezetés 4](#_Toc25262713)

[1. Témaválasztás indoklása 4](#_Toc25262714)

[2. Fejlesztői dokumentáció 4](#_Toc25262715)

[2.1. Igényfelmérés 4](#_Toc25262716)

[2.1.1. Összefüggések és célok 4](#_Toc25262717)

[2.1.2. Célközönség 5](#_Toc25262718)

[3. Futtatási környezet 5](#_Toc25262719)

[3.1.1. HTML 5](#_Toc25262720)

[3.1.2. CSS 5](#_Toc25262721)

[3.1.3. Javascript 5](#_Toc25262722)

[3.1.4. Ajax 6](#_Toc25262723)

[3.1.5. jQuery 6](#_Toc25262724)

[3.1.6. PHP 7](#_Toc25262725)

[3.1.7. SQL 7](#_Toc25262726)

[4. Fejlesztői környezet 8](#_Toc25262727)

[4.1.1. Visual Studio Code 8](#_Toc25262728)

[4.1.2. MariaDB 8](#_Toc25262729)

[5. Diagramok 9](#_Toc25262730)

[5.2.1. Regisztráció 12](#_Toc25262731)

[5.2.2. Belépés 14](#_Toc25262732)

[5.2.3. Rendelés 15](#_Toc25262733)

[5.2.4. Időpont kérés 16](#_Toc25262734)

[6. Tevékenységdiagramok 17](#_Toc25262735)

[7. Ábrák 21](#_Toc25262736)

[8. A program funciói 24](#_Toc25262737)

[8.1.1 Webes alkalmazás 24](#_Toc25262738)

[8.1.2 Főoldal 24](#_Toc25262739)

[8.1.3. Kapcsolat 24](#_Toc25262740)

[8.1.4. Belépés 24](#_Toc25262741)

[8.1.5. Regisztráció 24](#_Toc25262742)

[8.1.6. Profil 24](#_Toc25262743)

[8.1.7. Rendelés 24](#_Toc25262744)

[9. Irodalomjegyzék, hivatkozásjegyzék 25](#_Toc25262745)

# **Bevezetés**

A záródolgozatom témája egy vállalkozás weblapja lesz. Azért van erre szükség mert

a világ digitális, és nem maradhatunk le. Többen elérjék, és minden platformon

elérhető legyen. Egy webshop kiépítése a cél. Rendelési, és termék lefoglalási

lehetőséggel. Hogy megkönnyítsem az alkalmazottak, és a felhasználók dolgát.

# **1. Témaválasztás indoklása**

Szakdolgozatom a Benzinkút Webshop nevet kapta. Záródolgozatomban mindenre szeretnék kitérni. Mivel már szinte mindenki a telefonját használja, és rengeteg weblap elérhető. Meg persze közösségi oldalak, ezért szeretném hogy egy ilyen „kicsi” vállalkozás is helyet kapjon az Interneten. Igyekeztem minden funkcióra megoldást találni amit szerettem volna. A dolgozat további részeiben részletesen összefoglalom a weblapfejlesztés általánosan elfogadott modelljeit, és több éves szakértelemmel kifejlesztett technológiáit is.

# **2. Fejlesztői dokumentáció**

A fejlesztés első lépése az igényfelmérés és tervezés, amit szakszerűen el kell végezni, melynek során kiderül, hogy az ügyfél pontosan mit szeretne látni az oldalon, mit szeretne végezni. Ez alapján elkezdődhet a tervezés, az adatbázis felépítés, diagramok készítése, aminek alapján a fejlesztés meg fog valósulni.

## **2.1. Igényfelmérés**

## **2.1.1. Összefüggések és célok**

A cég szeretné hogy Interneten keresztül is elérhetőek legyenek. Minden termék, szolgáltatás látható, kérhető és rendelhető legyen. Továbbá a felhasználók naprakészen láthassák az üzemanyag és termék árakat is.

## **2.1.2. Célközönség**

* **felhasználók**: Megkeresik az általuk keresett termékeket, szolgáltatásokat. A belépés és rendelés regisztrációhoz kötött. Sikeres belépés után megtekinthetik a korábbi rendeléseiket.

* **alkalmazottak:**Ők töltik áruval a készletet. Felvételük után, ők is tudnak az oldalon rendezni, dolgozni. Ha rendelés történik azt ők igazolják vissza és készítik elő a terméket/termékeket a rendelő számára
* **adminok**: Teljes mértékben hozzáférnek a rendszerhez, a felhasználói módosításokat ők végzik el, beleértve az új alkalmazottak felvitelét is. És az oldal frissen tartását is.

# **3. Futtatási környezet**

A **futtatókörnyezet** programok futtatásához szükséges szoftverkörnyezet. Ennek részét képezi az operációs rendszer, illetve a telepített programkönyvtárak, segédprogramok, esetenként rendszerbeállítások.

## **3.1.1. HTML**

A **HTML** (angolul: HyperText Markup Language=hiperszöveges jelölőnyelv) egy leíró nyelv, melyet weboldalak készítéséhez fejlesztettek ki, és mára már internetes szabvánnyá vált a W3C (World Wide Web Consortium) támogatásával.

Az aktuális változata az 5, mely az SGML általános jelölőnyelv egy konkrét alkalmazása. Ezt tervek szerint lassan kiszorította volna az XHTML (ami a szintén SGML alapú XML leíró nyelven alapul.

## **3.1.2. CSS**

A CSS ( Cascading Style Sheets, magyarul: egymásba ágyazott stíluslapok) a számítástechnikában egy stílusleíró nyelv, mely a HTML vagy XHTML típusú strukturált dokumentumok megjelenését írja le. Ezenkívül használható bármilyen XML alapú dokumentum stílusának leírására is, mint például az SVG, XUL stb.

A CSS specifikációját a World Wide Web Consortium felügyeli.

## **3.1.3. Javascript**

A **JavaScript** [programozási nyelv](https://hu.wikipedia.org/wiki/Programoz%C3%A1si_nyelv) egy [objektumorientált](https://hu.wikipedia.org/wiki/Objektumorient%C3%A1lt_programoz%C3%A1s), prototípus alapú [szkriptnyelv](https://hu.wikipedia.org/wiki/Interpreteres_nyelvek" \o "Interpreteres nyelvek), amelyet [weboldalakon](https://hu.wikipedia.org/wiki/Weblap) elterjedten használnak.

A JavaScript kód vagy a [html](https://hu.wikipedia.org/wiki/Html" \o "Html) fájlban vagy külön (jellemzően .js kiterjesztésű) szövegfájlban van. Ezek a fájlok tetszőleges [szövegszerkesztő](https://hu.wikipedia.org/wiki/Sz%C3%B6vegszerkeszt%C5%91) (nem [dokumentumszerkesztő](https://hu.wikipedia.org/wiki/Dokumentumszerkeszt%C5%91)) programmal szerkeszthetőek.

A JavaScript esetében a futási környezet jellemzően egy [webböngésző](https://hu.wikipedia.org/wiki/Webb%C3%B6ng%C3%A9sz%C5%91) ([JavaScript-motorja](https://hu.wikipedia.org/wiki/JavaScript-motor)).

[Windowsos](https://hu.wikipedia.org/wiki/Windows) környezetben futtatható a [wscript.exe](https://hu.wikipedia.org/w/index.php?title=Wscript.exe&action=edit&redlink=1) segítségével vagy a [node](https://hu.wikipedia.org/wiki/Node" \o "Node)-dal a [Node.js](https://hu.wikipedia.org/wiki/Node.js) telepítésével. [linuxos](https://hu.wikipedia.org/wiki/Linux" \o "Linux) környezetben pedig [node](https://hu.wikipedia.org/wiki/Node" \o "Node)-dal és szintén a [Node.js](https://hu.wikipedia.org/wiki/Node.js" \o "Node.js) installálása után futtatható.

## **3.1.4. Ajax**

Az Ajax (Asynchronous JavaScript and XML) interaktív webalkalmazások létrehozására szolgáló webfejlesztési technika.

A weblap kis mennyiségű adatot cserél a szerverrel a háttérben, így a lapot nem kell újratölteni minden egyes alkalommal, amikor a felhasználó módosít valamit. Ez növeli a honlap interaktivitását, sebességét és használhatóságát.

Az Ajax a következő technikák kombinációja:

* **HTML** (vagy HTML) és CSS a tartalom leírására és formázására.
* **DOM**kliens oldali script nyelvekkel kezelve a dinamikus megjelenítés és a már

megjelenített információ együttműködésének kialakítására.

* **XMLHttpRequest** objektum az adatok aszinkron kezelésére a kliens és a webszerver

között. Néhány Ajax keretrendszer esetén és bizonyos helyzetekben IFrame- et használnak

XMLHttpRequest objektum helyett.

* **XML** formátumot használnak legtöbbször az adattovábbításra a kliens és a szerver

között, bár más formátumok is megfelelnek a célnak, mint a formázott HTML vagy a

sima szöveg.

Mint a DHTML, a LAMP vagy az SPA, az Ajax sem egy technológia önmagában, hanem egy kifejezés közösen használt technológiákra.

## **3.1.5. jQuery**

A **jQuery** népszerű JavaScript könyvtár, mely a HTML kód és a kliensoldali JavaScript közötti kapcsolatot hangsúlyozza. 2006 januárjában jelentette meg a Mozilla Alapítvány népszerű JavaScript evangelistája, John Resig. A függvénykönyvtár MIT és GNU kettős licenc alatt jelent meg. A jQuery ingyenes, nyílt forrású szoftver.

Számos ismert IT cég is alkalmazza a jQuery-t saját projektjeiben, például a Microsoft erre építette a Visual Studióban is elérhető [ASP.NET](http://asp.net/) AJAX platformját.[2]

A jQuery célja, hogy segítsen minél inkább leválasztani a JavaScript kódot a HTML-ről, és kényelmes kommunikációt biztosítson a weblap elemeivel – eseményvezérlők és azonosítók (ún. CSS szelektorok) használatával.

## **3.1.6. PHP**

A PHP egy általános szerveroldali szkriptnyelv dinamikus weblapok készítésére. Az első szkriptnyelvek egyike, amely külső fájl használata helyett HTML oldalba ágyazható. A kódot a webszerver PHP feldolgozómodulja értelmezi, ezzel dinamikus weboldalakat hozva létre. Rasmus Lerdorf 1995-ben indította útjára. Ma a The PHP Group tartja fenn és fejleszti. A PHP szabad szoftver, de licence nem csereszabatos a GNU licenccel, mivel megkötéseket tartalmaz a PHP név használatára.

A hagyományos HTML lapokkal ellentétben a kiszolgáló a PHP-kódot nem küldi el az ügyfélnek, hanem a kiszolgáló oldalán a PHP-értelmező motor dolgozza fel azt. A programokban lévő HTML elemek érintetlenül maradnak, de a PHP kódok lefutnak. A kódok végezhetnek adatbázis-lekérdezéseket, létrehozhatnak képeket, fájlokat olvashatnak és írhatnak, kapcsolatot létesíthetnek távoli kiszolgálókkal. A PHP-kódok kimenete a megadott HTML elemekkel együtt kerül az ügyfélhez.

A PHP-t parancssori alkalmazásként is telepítik, így alkalmas parancsfájlok készítésére. Számos rendszergazda automatizálási célokra is a PHP-t használja, pedig ezt a feladatot hagyományosan Perl- vagy shell scriptekkel oldották meg.

## **3.1.7. SQL**

Az SQL, azaz Structured Query Language (strukturált lekérdezőnyelv) relációsadatbázis- kezelők lekérdezési nyelve.

Angol nyelvterületen 'eszkjuel' a kiejtése. A hagyományokhoz való hűség jegyében sokan 'szíkvel'-nek ejtik, ugyanis korábban Structured English Query Language (SEQUEL) volt az elnevezés, és ezt rövidítették le.

A relációsadatbázis-kezelők általában az SQL nyelven programozhatók. Az SQL alapvető utasításait közel egyformán valósítják meg, de a később beépült nyelvi elemek körében nagyon nagy az eltérés, az inkompatibilitás, emiatt számos SQL nyelvjárásról beszélhetünk.

Jellegét tekintve ez a szakterület-specifikus nyelv részben procedurális, részben deklaratív.

# **4.** **Fejlesztői környezet**

A fejlesztői környezet programozási eszközök, könyvtárak és beállítások csoportja, amelyekkel a szoftverfejlesztés során a felhasznált programozási nyelven vagy nyelveken létrehozott forráskódokat futáskész állapotba készítjük és azt teszteljük. Ez jelentheti a fordítást, vagy nem önálló programok futtatási környezetbe helyezését. Mindkét esetben a környezet általában tartalmazza a futtatókörnyezetet is a tesztelés miatt.

Sok esetben a felhasznált programozási nyelv rendelkezik egy olyan csomaggal, melyet telepítve a fejlesztői gépen ez a környezet automatikusan létrejön.

Sok nyelv rendelkezik saját forráskód szerkesztővel, fordítóval, ezt hívják integrált fejlesztői környezetnek.

A fejlesztői környezet nagyobb méretű, szerteágazóbb programok esetében nem merül ki a fent említett csomagok összességében. A fejlesztés során rengeteg felhasznált program, segédprogram, esetleg a fejlesztési lépések vagy a tesztelés segítéséreszolgáló rövid egyedi segédprogramok, illetve egy szerteágazó jegyzékrendszer alkotja.

A fejlesztői környezet csak a fejlesztést végző személy/ek által használt számítógép/ek-en szükségesek. A kész program csak a futtatókörnyezetet igényli.

## **4.1.1. Visual Studio Code**

A **Visual Studio Code** (rövidítve: **VSCode** vagy **VS Code**) egy ingyenes, nyílt forráskódú kódszerkesztő, melyet a Microsoft fejleszt Windows, Linux és OS X operációs rendszerekhez. Támogatja a hibakeresőket, valamint beépített Git támogatással rendelkezik, továbbá képes az intelligens kódkezelésre (intelligent code completion) az IntelliSense segítségével. Ezen felül testre szabható, így a felhasználók megváltoztathatják a kinézetet (témát), megváltoztathatják a szerkesztő gyorsbillentyű-kiosztását, az alapértelmezett beállításokat és még sok egyebet.

## **4.1.2. MariaDB**

A **MariaDB** egy többfelhasználós, többszálú, SQL-alapú relációsadatbázis-kezelő szerver, egy MySQL-fork. Mivel kompatibilis a MySQL-lel, a MariaDB is lehet a LAMP adatbázisszervere.

# **5. Diagramok**

Ebben a részben a különböző folyamatokat láthatjuk lerajzolva.

Ilyenek például a regisztráció, belépés vagy az időpont kérés.

Mindennek megvan a saját rajza. Ezeken az ábrákon keresztül szemléltetem a folyamatok lépéseit.

Nagyon egyszerű volt a dolgom az ábrák elkészítése során.

A program amit használtam hozzá az a Visual Paradigm volt.

Mi is az a Visual Paradigm?

A Visual Paradigm (VP-UML) egy [UML](https://en.wikipedia.org/wiki/Unified_Modeling_Language)[CASE](https://en.wikipedia.org/wiki/Computer-aided_software_engineering) eszköz, amely támogatja az [Object Management Group (OMG)](https://en.wikipedia.org/wiki/Object_Management_Group" \o "Objektumkezelő csoport) UML 2, [SysML](https://en.wikipedia.org/wiki/Systems_Modeling_Language" \o "Rendszert modellező nyelv) és [Business Process Modeling Notation (BPMN](https://en.wikipedia.org/wiki/Business_Process_Modeling_Notation" \o "Üzleti folyamat modellezés jelölés)[) szolgáltatásait](https://en.wikipedia.org/wiki/Object_Management_Group) . A modellezési támogatás mellett jelentést generál és kódmérnöki képességeket is kínál, beleértve a [kód generálást](https://en.wikipedia.org/wiki/Automatic_programming) . Meg lehet [visszafordítani](https://en.wikipedia.org/wiki/Reverse_engineer) diagramokat kódot, és [oda-vissza mérnöki](https://en.wikipedia.org/wiki/Round-trip_engineering" \o "Oda-vissza mérnöki munka) különböző [programozási nyelvek](https://en.wikipedia.org/wiki/Programming_language).

|  |  |  |  |
| --- | --- | --- | --- |
| **Ki** | **Milyen gyakran** | **Mit** | **Megjegyzés** |
|  |  |  |  |
| - | 1 | Rendel |  |
| - | 1 | Rendelése | Online |
| F, A | 1 | Regisztráció | E-mail, Login, Password |
| F, A | 1 | Felhasználó beléptetése | Login, Password |
| F | 1\_N | Termék/ek kiválasztása |  |
|  |  | Termék választás | típusa, mennyisége |
| A | 1\_N | Termék mutatása | Alternatív mutatása |
| A | 1\_N | Kosárba helyezés | További adatok megadása |
| F, A | 1 | Kosár ellenőrzése | Mennyiség helyes, termék helyes |
| F | 1\_N | Rendelés folytatása | Adatok megadása/Bejelentkezés |
| F, A | 1 | Fizetés | Helyszínen |
| K | 1 | Helyszínen termék előkészítés | Megkeresés, csomagolás, számla ha kell akkor készítése |
| F | 1\_N | Termékhez jutás | A helyszínen igazolva magát átveszi |
| F, K | 1\_N | Fizetés | készpénz, bankkártya |
| K | 1\_N | Termék átadása | Bónuszpontok hozzáadása/felülírása |

F: felhasználó A: admin K: kutas

|  |  |  |  |
| --- | --- | --- | --- |
| **Ki** | **Milyen gyakran** | **Mit** | **Megjegyzés** |
|  |  |  |  |
| - | 1 | Időpont kérés | Online |
| - | 1 | Gumis műhely időpont | Bejelentkezés/Regisztráció |
| F, A | 1 | Felhasználó beléptetése | Login, Password |
| F, G, A | 1\_N | Időpont megadása |  |
| A, G | 1\_N | Időpontok mutatása | Ideje, ára, mikor |
| F | 1 | Kiválasztása | Mikor, ideje |
| G, A | 1 | Elfogadás, visszaigazolás | E-mail, megfelel-e az időpont |
| F, G, A | 1\_N | Pontosítás |  |
| F, G | 1\_N | Megérkezés | Igazolva magát, és az időpontot |
| F, G | 1\_N | Folyamat | Javítás, Csere stb. |
| G | 1 | Fizetés | Helyszínen, készpénz |
| G | 1 | Nyugta adás | Visszaigazolás |

F: felhasználó A: admin G: gumis

„A rendszer modellezése a felhasználó, megrendelő szemszögéből. Ez a modell a fejlesztés

kezdeti fázisaiban lényegében kialakul, és végigkíséri a teljes fejlesztést.

Feladatunk:

- Felhasználók információs igényeinek elemzése.

- Funkcionális követelmények elemzése.

A modell tartalmazza a rendszerrel szemben támasztott felhasználói követelményeket;

- kik, és mire akarják használni a rendszert (Felhasználók információs igényeinek elemzése.).

- Megadja, hogy a rendszert mire akarják majd használni (funkcionális követelmények),

- milyen egyéb feltételeknek kell teljesülniük (nem funkcionális követelmények; pl.képernyőn való megjelenés, nyomtatás, teljesítmény, tesztelés).

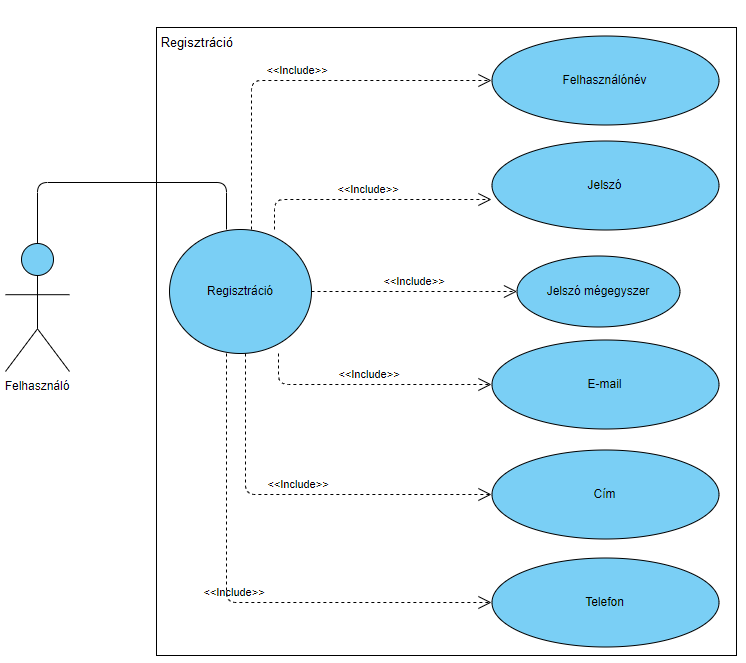
## **5.2.1. Regisztráció**

Ezen az ábrán láthatjuk egy rajz segítségével a regisztráció folyamatát.

Három fél elengedhetetlen a regisztrációhoz. Ilyenek az admin, és maga a felhasználó.

A regisztrációhoz a következő adatokra lesz szükség. Felhasználónév, jelszó, DE a jelszó mégegyszer kelleni fog. Kell még az E-mail címünk, lakcímünk és telefonszámunk is.

Majd véglegestjük a regisztrációt. Erről a folyamatról egy E-Mailt fogunk kapni.

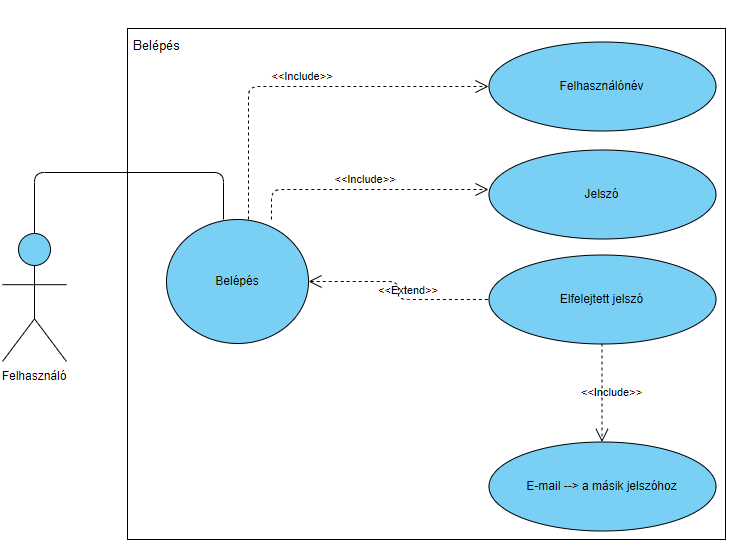


## **5.2.2. Belépés**

A belépésnek is megvan a maga folyamata. Ezt a következő ábra szemlélteti.

Kellenek a belépéshez az adataink. Ilyen a felhasználónév, jelszó.

De ha a jelszavunkat elfelejtettük volna, akkor kérhetünk másikat.

E-mail címünk segítségével jutunk hozzá.

## **5.2.3. Rendelés**

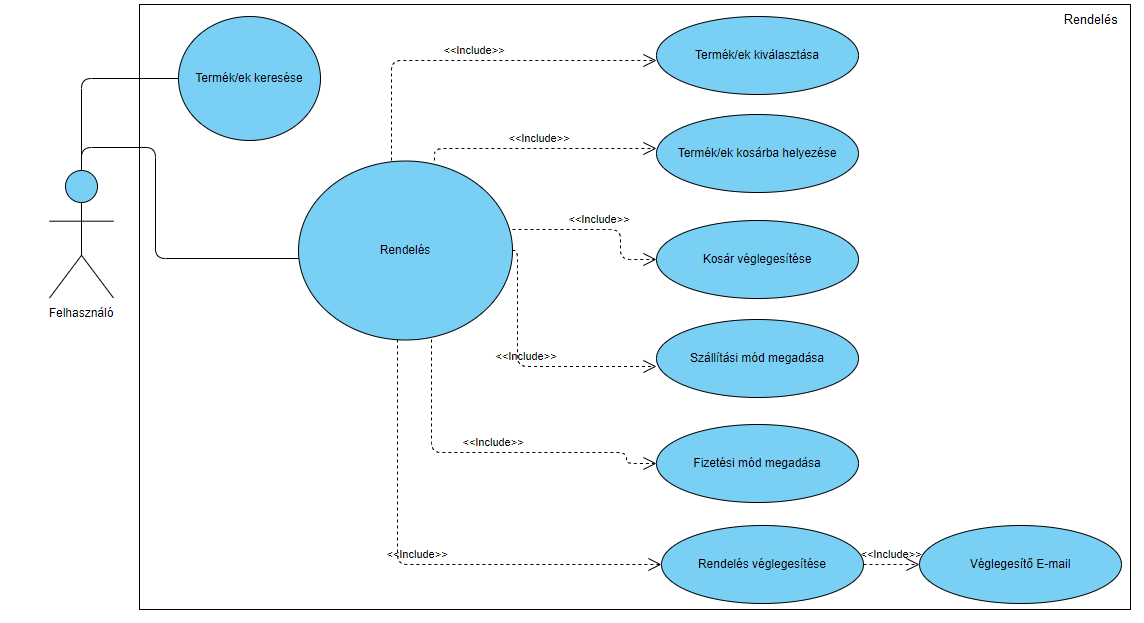
Mivel az oldalon tudunk rendelni is ezért ehhez is készítettem ábrát.

Adataink megadásával tudunk csak rendelni.

Meg kell keresni a rendelni kívánt terméket. Majd kiválasztjuk, kosárba helyezzük.

Majdnem de nem végül megadjuk a fizetési módot. Ebből többféle is van. Bankkártyás és készpénzes.

Utolsó lépés pedig a véglegesítés. Erről kapunk egy E-mailt. Ebben láthatjuk a rendekésünket.



## **5.2.4. Időpont kérés**

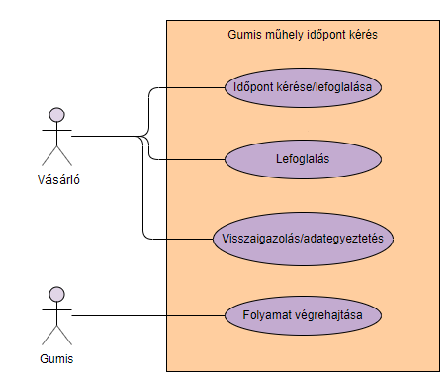
A cég része a gumis műhely is. Ebben a részben szeretném szemléltetni az időpont egyeztetés menetét.

Ehhez szükségünk van az adatainkra. Kell a nevünk, címünk.

Különböző időpontok közül tudunk választani. Ezek fel vannak sorolva az oldalon.

Kiválasztjuk, majd egyeztetjük adatainkat.

Innen már csak a szolgáltatást kell igénybe vennünk. Mindenről tehát erről is kapunk egy E-mailt.



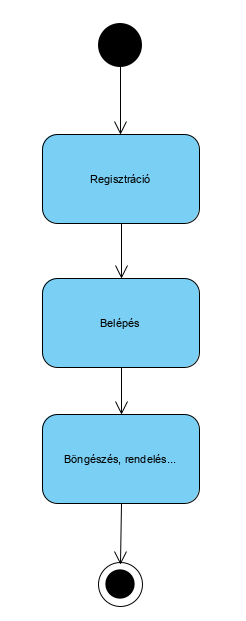
# **6. Tevékenységdiagramok**

Ebben a részben a folyamatok és tevékenységek ábráit láthatjuk a rajzokon.

Mindennek megvan a maga sorrendje, helye, ideje. Pl.: regisztráció

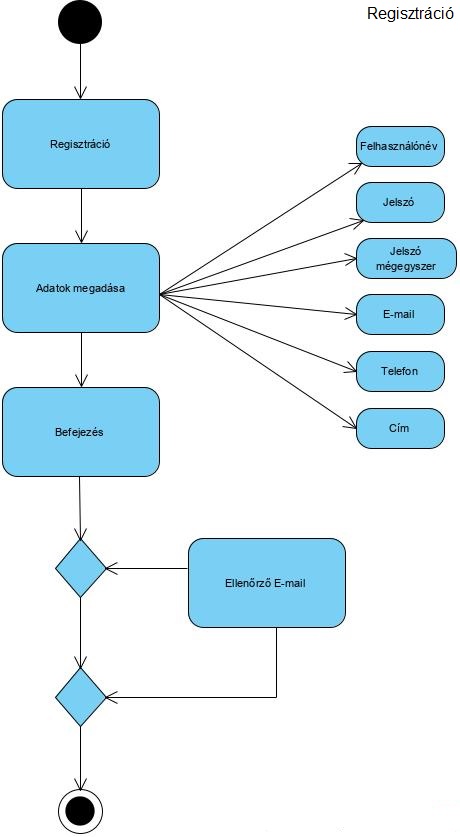
De különböznek az előbb látott diagramoktól. Itt a lépések nem aktorokhoz vannak kötve, hanem sorban egymásután jelölöm őket.

Első a regisztráció.



Láthatjuk a regisztrációt részletesebben is.

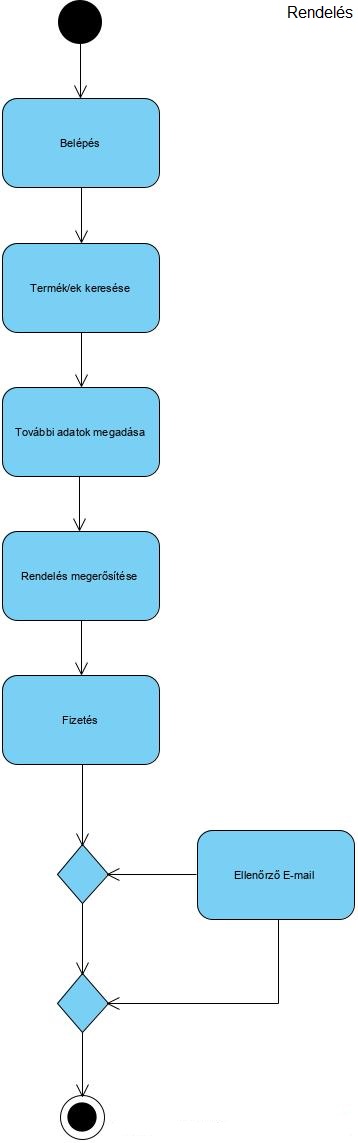
Itt minden folyamat lépés látható.



Következő lépés pedig a rendelés folyamata lesz.

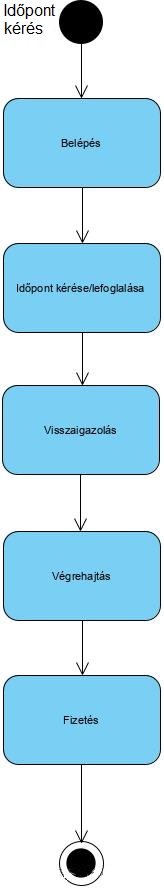
Itt is nagyon részletesen lehet látni a lépéseket, egymás után.

Itt is törekedtem a legjobb ábrázolásra.



Az említett gumis műhely időpont kérése folyamata is látható a következő ábrán.

Rendszerint minden lépés megvan, és szerepel.



# **7. Ábrák**

Ebben a fejezetben láthatjuk lépésről-lépésre a folyamatokat. Első lesz a regisztráció. Minden lépés leírva látható.

|  |  |  |
| --- | --- | --- |
| Azonosító  001 | Név  Regisztráció | |
| Rövid leírás  A felhasználó regisztrál, majd böngész az oldalon. Rendel, foglal. Információkhoz, hírekhez jut. | | |
| Érintett aktorok  Felhasználó (F), admin (A) | | |
| Kiváltó esemény  Amikor az FH regisztrálni és rendelni szeretne | | |
| Paraméterek  Felhasználónév, jelszó, E-mail, cím, telefon | | |
| Standard lefutás   1. Regisztráció 2. Adatok megadása 3. Véglegesítés | | Kivételek és alternatívák   1. Jelszó mégegyszer 2. Elfelejtett jelszó |
| Utófeltétel  Adataink ellenőrzése  Véglegesítés | | |
| Eredmény  Regisztráció elkészült | | |
| Átlagos időtartam  20 perc | | |
| Utalások  001-1, 001-2, 001-3, 001-4, 001-5 | | |
| Megjegyzések, nyitott kérdések  Rendelni csak így lehet | | |

A következő pedig a rendelés folyamata lesz.

Itt is minden lépés látható.

|  |  |  |
| --- | --- | --- |
| Azonosító  002 | Név  Rendelés | |
| Rövid leírás  A felhasználó termékeket keres, majd választ. Rendel, foglal. | | |
| Érintett aktorok  Felhasználó (F) | | |
| Kiváltó esemény  Amikor az FH rendelni szeretne | | |
| Paraméterek  E-mail, cím, telefon | | |
| Standard lefutás   1. Keresés 2. Kiválasztás 3. Mennyiség megadása 4. Kosárba helyezés 5. Véglegesítés | | Kivételek és alternatívák   1. Kosárból vehetek ki termékeket, de rakhatok is bele 2. Szállítási cím választás 3. Kedvezmények fogadása 4. Számlázási cím választás |
| Utófeltétel  Készlet ellenőrzés | | |
| Eredmény  Megrendelés rögzítésre került | | |
| Átlagos időtartam  1 óra | | |
| Utalások  002-1, 002-2, 002-3, 002-4, 002-5 | | |
| Megjegyzések, nyitott kérdések  Regisztráció kell a rendeléshez | | |

A gumis műhely időpont foglalásának lépéseit itt láthatjuk.

Lépések sorban láthatóak. Részletesen leírva minden.

|  |  |  |
| --- | --- | --- |
| Azonosító  003 | Név  Időpont foglalás | |
| Rövid leírás  A felhasználó időpontot foglal, majd választ. De tudnak új gumiaink közül is válogatni, vagy rendelni | | |
| Érintett aktorok  Felhasználó (F), gumis (G), admin (A) | | |
| Kiváltó esemény  Amikor az FH időpontot szeretne foglalni | | |
| Paraméterek  E-mail, cím, telefon | | |
| Standard lefutás   1. Időpont keresés 2. Kiválasztása 3. Véglegesítés | | Kivételek és alternatívák   1. Új gumi vásárlása 2. Időpont modosítása |
| Utófeltétel  Van-e szabad időpont | | |
| Eredmény  Foglalás rögzítésre került | | |
| Átlagos időtartam  30 perc | | |
| Utalások  0023-1, 003-2, 003-3, 003-4, 003-5 | | |
| Megjegyzések, nyitott kérdések  Regisztráció kell a foglaláshoz | | |

# **8. A program funciói**

## **8.1.1 Webes alkalmazás**

A webes felület a külső felhasználók, könyvhasználók részére lett kialakítva, amiben tájékozódni tudnak az aktuális akciókról, termékejről valamint megtekinthetik a saját rendelési állapotukat.

## **8.1.2 Főoldal**

Ez a nyitóoldal, ahol bárki kilistázhatja az aktuális termékeket, nézheti az aznapi üzemanyag árakat. Segégitségükre lesz: különféle szűrőfeltételek, és utána kiválasztás után részletes lista jön elő az aktuális termékekről

## **8.1.3. Kapcsolat**

A benzinkútról tartalmaz általános információkat

## **8.1.4. Belépés**

Itt tud bejelentkezni a felhasználó felhasználónév és jelszó megadása után

## **8.1.5. Regisztráció**

A regisztrációt itt lehet elvégezni, ahol meg kell adni a jövőbeli könyvhasználó felhasználónevét, jelszavát, e-mail címét, nevét, születési dátumát, mobilszámát, lakcímét

## **8.1.6. Profil**

A belépett felhasználó tudja módosítani a regisztráció során megadott adatait, vagy esetlegesen törölni fiókját.

## **8.1.7. Rendelés**

A felhasználók tudnak rendelni és szolgáltatásokat igénybe venni. Ezeket az alkalmazottak kezelik. És segítik vele a munkát. Ők készítik elő a termékeket a rendeléshez. Számla, regisztráció kötelező lesz.

# **9. Irodalomjegyzék, hivatkozásjegyzék**

<https://hu.wikipedia.org/wiki/HTML>

<https://hu.wikipedia.org/wiki/Cascading_Style_Sheets>

<https://hu.wikipedia.org/wiki/JavaScript>

<https://hu.wikipedia.org/wiki/Ajax_(programoz%C3%A1s)>

<https://hu.wikipedia.org/wiki/JQuery>

<https://hu.wikipedia.org/wiki/PHP>

<https://hu.wikipedia.org/wiki/SQL>

<https://hu.wikipedia.org/wiki/Visual_Studio_Code>

<https://hu.wikipedia.org/wiki/MariaDB>

<https://en.wikipedia.org/wiki/Visual_Paradigm>