Rendszer dokumentáció

Telekommunikációs weboldal

Tartalom

[**1. A projekt feladatról** 2](#_Toc504051927)

[*1.1 – Feladat leírása, megfogalmazása* 2](#_Toc504051928)

[*1.2– A projektben résztvevő tagok* 2](#_Toc504051929)

[*1.3– Feladatok felosztása a tagok közt* 3](#_Toc504051930)

[*1.4 – Határidők* 3](#_Toc504051931)

[**2. - Felhasznált programok, nyelvek** 3](#_Toc504051932)

[2.1 - Programok 3](#_Toc504051933)

[2.1.1 - Atom code editor 3](#_Toc504051934)

[2.1.2 - XAMPP 4](#_Toc504051935)

[2.2 – Nyelvek 4](#_Toc504051936)

[2.2.1 – HTML5 4](#_Toc504051937)

[2.2.2 – CSS3 4](#_Toc504051938)

[2.2.3 – JavaScript 4](#_Toc504051939)

[2.2.4 – Material Design 4](#_Toc504051940)

[2.2.5 – SQL 4](#_Toc504051941)

[**3. - A rendszer futtatása** 4](#_Toc504051942)

[**4. – A rendszer felépítése** 4](#_Toc504051943)

[*4.1 – Back-end* 4](#_Toc504051944)

[*4.2 – Front-end* 5](#_Toc504051945)

[4.2.1 – Index.html, indexstyle.css 5](#_Toc504051946)

[4.2.2 – Intro.html 5](#_Toc504051947)

[4.2.3 – Reg.html 6](#_Toc504051948)

[4.2.4 – Sign\_in.html 6](#_Toc504051949)

# **1. A projekt feladatról**

## *1.1 – Feladat leírása, megfogalmazása*

Feladatunk egy telekommunikációs cég weboldalának létrehozása, azon belül a back-end és a front-end megtervezése és implementálása. A vállalat neve*, P-LID*, vagyis **Professzionális-Levelezés Internet Direktben**. A cég alapvetően 4 szolgáltatást nyújt ügyfeleinek, mégpedig: internet, telefon, televízió, és még egy 3-in-1 szolgáltatást, ami tartalmazza az összes, a cég által kínált szolgáltatást.

A megvalósítandó oldalak a következők: szükséges egy főoldal, egy login oldal, ahol az ügyfelek be tudnak jelentkezni az oldalra, az új ügyfeleknek egy regisztrációs oldal, hogy tudjanak regisztrálni. Ezen felül létre kell hozni az összes szolgáltatáshoz köthető oldalt is, ez szám szerint 4 lesz.

A belépés után a felhasználó a következő műveleteket tudja elvégezni: megnézni a csomagját, mennyi még a hűségideje, csomagot válthat, új jelszót adhat meg, adatot módosíthat, hűség időt hosszabbíthat csomagváltás nélkül.

Léteznie kell egy oldalnak, amin a felhasználó végig navigálva új szolgáltatást vásárolhat.

Egy külön oldal megalkotása is a követelmények között szerepel, ez pedig az adminisztrátori felület. Az adminnak jogában áll látni az összes felhasználót, azoknak minden adatait. Ezen felül képes új felhasználót felvenni az adatbázisba, meglévőt törölni, adatokat módosítani, új csomagot hozzáadni/cserélni és törölni.

A rendszer megvalósítása a következő nyelvekkel kell, hogy történjen.

* **Front-end:**
  + HTML5
  + JavaScript
  + CSS3
  + Material Design
* **Back-end:**
  + SQL

## *1.2– A projektben résztvevő tagok*

|  |  |  |
| --- | --- | --- |
| Név | Neptun-kód | Szak |
| Pap Réka | BFSE36 | Gazdaságinformatikus |
| Hegyi Balázs | W0EAGL | Mérnökinformatikus |
| Kápli Gergely | OGSRVZ | Gazdaságinformatikus |
| Rádli Richárd | QW7LE5 | Mérnökinformatikus |

## *1.3– Feladatok felosztása a tagok közt*

|  |  |  |  |
| --- | --- | --- | --- |
| **Feladat ID** | **Feladat neve** | **Rövid leírás** | **Végrehajtó személy** |
| **001** | Adatszerkezet megtervezése | A weboldal back-end részének gerincét adó relációs adatmodell megtervezése. | Egész csapat |
| **002** | Adatszerkezet megvalósítása | A papíron már megtervezett modell átültetése a valóságba. Fizikai adatbázis létrehozása. |  |
| **003** | Felhasználói interface megtervezése | A felhasználok által látott, ill. használt interface megtervezése. | Egész csapat |
| **004** | Felhasználói interface megvalósítása | A megtervezett UI implementálása. | Kápli Gergely  Rádli Richárd |

## *1.4 – Határidők*

|  |  |  |  |
| --- | --- | --- | --- |
| **Feladat ID** | **Kezdés** | **Határidő** | **Végrehajtó személy** |
| **001** |  |  | Egész csapat |
| **002** |  |  |  |
| **003** |  |  | Egész csapat |
| **004** |  |  | Kápli Gergely  Rádli Richárd |

# **2. - Felhasznált programok, nyelvek**

## 2.1 - Programok

A következő programokat használtuk a megvalósítás során:

### 2.1.1 - Atom code editor

Az Atom egy ingyenes és nyílt forráskódú kódszerkesztő, mely támogatja az feladatban meghatározott nyelveket, ezért esett erre a választás. Főbb jellemzői:

* Beépített csomag menedzser
* Cross-platform szerkesztő
* Fájlböngésző
* Kereső és fájlcserélő
* Többszörös lapok
* Okos auto-befejezés

### 2.1.2 - XAMPP

## 2.2 – Nyelvek

A következő nyelveket használtuk a megvalósítás során:

### 2.2.1 – HTML5

A HTML egy leíró nyelv, melyet [weboldalak](https://hu.wikipedia.org/wiki/Weboldal) készítéséhez fejlesztettek ki, és mára már internetes szabvánnyá vált a [W3C](https://hu.wikipedia.org/wiki/W3C) (World Wide Web Consortium) támogatásával. A nyelv legújabb verzióját a HTML5-öt alkalmaztuk.

### 2.2.2 – CSS3

A CSS a számítástechnikában egy [stílusleíró nyelv](https://hu.wikipedia.org/w/index.php?title=St%C3%ADlusle%C3%ADr%C3%B3_nyelv&action=edit&redlink=1), mely a [HTML](https://hu.wikipedia.org/wiki/HTML) vagy [XHTML](https://hu.wikipedia.org/wiki/XHTML) típusú strukturált dokumentumok megjelenését írja le. A CSS-t a [weblapok](https://hu.wikipedia.org/wiki/Weblap) szerkesztői és olvasói egyaránt használhatják, hogy átállítsák vele a lapok színét, betűtípusait, elrendezését, és más megjelenéshez kapcsolódó elemeit.

### 2.2.3 – JavaScript

A JavaScript [programozási nyelv](https://hu.wikipedia.org/wiki/Programoz%C3%A1si_nyelv) egy [objektumalapú](https://hu.wikipedia.org/w/index.php?title=Objektumalap%C3%BA&action=edit&redlink=1), prototípus alapú [szkriptnyelv](https://hu.wikipedia.org/wiki/Szkriptnyelv" \o "Szkriptnyelv), amelyet [weboldalakon](https://hu.wikipedia.org/wiki/Weboldal) elterjedten használnak.

### 2.2.4 – Material Design

Material design (anyagszerű megjelenés) a [Google](https://hu.wikipedia.org/wiki/Google) által fejlesztett, a 2014. június 25-i [Google I/O](https://hu.wikipedia.org/wiki/Google_I/O) konferencián bejelentett [formanyelv](https://hu.wikipedia.org/w/index.php?title=Formanyelv&action=edit&redlink=1) (design language). Az először a [Google Now](https://hu.wikipedia.org/wiki/Google_Now)-ban látott kartotéklap-motívumok továbbfejlesztve, letisztultabb formában jelennek meg, a rács-alapú elrendezés gyakoribb használatával, reszponzív animációkkal és áttűnésekkel, kitöltésekkel és a mélység hatását keltő effektekkel, mint a megvilágítás és az árnyékok vetése.

### 2.2.5 – SQL

Az SQL, azaz Structured Query Language (strukturált lekérdezőnyelv) [relációsadatbázis-kezelők](https://hu.wikipedia.org/wiki/Rel%C3%A1ci%C3%B3sadatb%C3%A1zis-kezel%C5%91_rendszer) lekérdezési nyelve. A [relációsadatbázis-kezelők](https://hu.wikipedia.org/wiki/Rel%C3%A1ci%C3%B3sadatb%C3%A1zis-kezel%C5%91_rendszer) 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](https://hu.wikipedia.org/wiki/Kompatibilit%C3%A1s_(technika)), emiatt számos SQL nyelvjárásról beszélhetünk.

# **3. - A rendszer futtatása**

# **4. – A rendszer felépítése**

## *4.1 – Back-end*

Egy adott rendszer alsóbb, a tényleges feldolgozást végző rétege. A back-end réteg feladata a front-end réteg felől érkező adatok feldolgozása, ill. a keletkezett eredmény a front-end számára történő visszajuttatása.

## *4.2 – Front-end*

A front-end, vagyis egy adott rendszer legfelsőbb, a felhasználóval vagy a csatlakoztatott további rendszerekkel a kapcsolatot tartó rétege. A front-end réteg feladata a rendszerből kijutó adatok prezentálása, illetve a bejövő adatok fogadása a felhasználó vagy a csatlakoztatott rendszer felől.

Most vegyük sorra a megvalósított oldalak felépítését.

### 4.2.1 – Index.html, indexstyle.css

A kezdőoldal, ezt látja meg először a felhasználó, ha meglátogatja az oldalt. 4 jól elkülönülő részre bonthatjuk a html részt.

Első rész a *MAIN\_NAV\_BAR*, ez a felső sáv, itt kapott helyet az oldalsó SIDE\_BAR, a Bejelentkezés BUTTON, egy a szolgáltatásokat tartalmazó DROPDOWN, valamint, egy regisztrációs TOOLTIPPED\_BUTTON.

Második rész a FIRST\_BANNER, ez tartalmaz egy képet, aminek pozíciója nem fix, tehát mozog, ha görgetünk. Egy szöveg is található itt, ami a vállalat nevét és szlogenjét írja ki. A szöveg alatti „Bemutatkozás” gombbal az intro.html oldalra navigálhatunk át.

A harmadik részben található a SECOND\_BANNER, itt ismertetjük a cég szolgáltatásait. Struktúráját tekintve mindegyik szolgáltatás egy oszlopban van benne, melyhez jön egy kép, egy rövid szöveges leírás, és egy gomb, ami az adott szolgáltatás oldalára irányít át.

A negyedik rész a FOOTER, vagyis az oldal alja. Egy oszlopban soroljuk fel a cég elérhetőségeit, mindezt egy listaként.

Az oldalon helyet kapott még egy TAP\_TARGET, melyre ha rákattint a felhasználó, egy negyedkör jelenik meg, melyben megjelenítjük az ügyfélszolgálat telefonszámát.

### 4.2.2 – Intro.html

A bemutatkozó oldal, itt mutatjuk be a céget. 6 jól elkülönülő részre bonthatjuk a html részt.

Első rész a **MAIN\_NAV\_BAR**, ez a felső sáv, itt kapott helyet az oldalsó SIDE\_BAR, egy ügyfélkapu nevű DROPDOWN amiben a felhasználó további 2 opciót talál, a regisztrációt és a belépést. Az utolsó elem egy FLOATING\_BUTTON, mellyel a főoldalra térhetünk vissza.

Második rész a **FIRST\_BANNER**, ez tartalmaz egy képet, aminek pozíciója nem fix, tehát mozog, ha görgetünk. Egy szöveg is található itt, ami a vállalat nevét és szlogenjét írja ki.

Harmadik rész az **INTRODUCTION**, melyben a vállalatról olvashatunk röviden. Két paragrafusba törtük a szöveget, a végén egy elválasztó vonalat alkalmaztunk.

Negyedik rész a **WE\_ARE\_THE\_BEST** melyben egy lenyíló menüben olvashatunk a cég erényeiről. Mind a 4 tulajdonsághoz egy címszó tartozik, egy ikonnal, és egyedi szöveggel.

Ötödik rész a **TEAM**, mely a P-LID csapatát mutatja be. Mindegyik csapattag egy kártyán szerepel, ahol megjelenik a képe, neve, beosztása, szerepe a cég életében, és az elérhetőségei.

A hatodik rész a **FOOTER**, vagyis az oldal alja. Egy oszlopban soroljuk fel a cég elérhetőségeit, mindezt egy listaként.

Az oldalon helyet kapott még egy TAP\_TARGET, melyre ha rákattint a felhasználó, egy negyedkör jelenik meg, melyben megjelenítjük az ügyfélszolgálat telefonszámát.

### 4.2.3 – Reg.html

A regisztrációs oldal, ahol a látogatók regisztrálni tudnak, és így felhasználói státuszba kerülhetnek. Egy form-boksz-ot használunk itt, melyet felosztottunk 2 részre, egy felső és egy alsó részre. A felső rész csak a Regisztráció szöveget írja ki, az alsóban pedig maga a regisztrációs form van. A látogatónak meg kell adnia a felhasználó nevét, jelszavát (biztonság miatt kétszer), az e-mail címét, a telefonszámát, és a születési dátumát. A regisztráció gombbal regisztrálhatja magát, ezután pedig automatikusan visszatér a főoldalra.

Két FLOATING\_BUTTON van még az oldalon ami segíti a navigálást, a bal sarokban lévővel visszatérhetünk a főoldalra, a jobb sarokban lévővel pedig a bejelentkezés oldalra térhetünk át.

### 4.2.4 – Sign\_in.html

Az oldal, ahol be lehet jelentkezni a rendszerbe. Egy form-boksz-ot használunk itt, melyet felosztottunk 2 részre, egy felső és egy alsó részre. A felső rész csak a Bejelentkezés szöveget írja ki, az alsóban pedig maga a login form van. A felhasználónak meg kell adnia felhasználó nevét, és jelszavát, hogy beléphessen. A Belépés gombra nyomva pedig visszakerül a főoldalra, immár bejelentkezve.

Két FLOATING\_BUTTON van még az oldalon ami segíti a navigálást, a form-boksz bal sarkában lévővel visszatérhetünk a főoldalra, a jobb sarokban lévővel pedig a regisztrációs oldalra térhetünk át.