

**LIA Alapítványi Óvoda és Szakképző Iskola
Kaposvár**

Szoftverfejlesztő
14. évfolyam

Tanév: 2014/2015

ZÁRÓDOLGOZAT

Zárárdolgozat címe: Futár

Készítette: Nyári Bálint

Dátum: 2015.03.26

Tartalom

Tartalom	2
Bevezető	3
Fejlesztői dokumentáció.....	4
Feladat specifikáció	4
Követelmények (Fejlesztői környezet, webszerver)	4
Képernyőtervek	5
Adatbázis tervek	5
Programkód felépítése (mappák, fájlok szerepe)	7
Továbbfejlesztés lehetőségei	8
Felhasználói dokumentáció	10
Ismertető.....	10
Futtatási környezet	10
A program használatának leírása.....	10
Irodalomjegyzék.....	11
Nyilatkozat	12

Bevezető

A záródolgozatomban egy ügynevezett futár cég honlapját/mintáját fogom elkészíteni. Hogy ez valójában mi is? Manapság igencsak sok az vagy „az ügynevezett futár cég futár cég, vagyis olyan cégekre kell itt gondolni, ahol egy regisztráció után felvihetünk csomagokat és ők ezt kiszállítják egy adott szállítási pontra vagy, ha igénylik, házhoz is. A mai világban igen nagy szükség van az ilyesféle honlapokra, hiszen egy elektronikus világban élünk. Sokan azt hinnénk, hogy az internetet csak a fiatalok használják, de ez nem igaz. Ezt bizonyítják az ilyen oldalak is. Egy ilyen oldalnak nincs célközönsége, ugyanis ez egy cégnek a saját weboldala. Ezt bárki használhatja, más cégek, vagy akár magánszemélyek is.

Fejlesztői dokumentáció

Feladat specifikáció

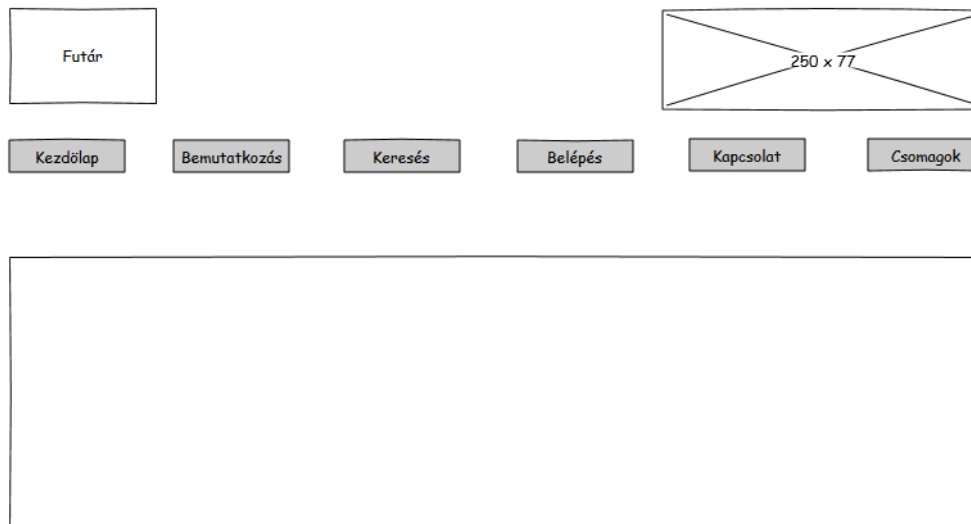
Az én záródolgozatom egy weblap, melynek a feladata egy működőképes futár, vagy szállítmányozási cég honlapja. Egy külön, ehhez a munkához szánt adatbázissal összeköttetésben áll, ezek teszik lehetővé a csomagok rögzítését, vagy éppen egy csomagról való adatok lekérését/kikérését. Az oldal bejelentkezés nélkül is használható ugyan, de nem teljes mértékben, ugyanis vannak olyan funkciói, amelyeket egy külső személy nem biztos, hogy láthat. Bejelentkezés után persze rendelkezésünkre állnak ezek a funkciók is. Ilyenek nem kell zárójel: Ilyenek pl.: Csomagok adatai, szállítási pontok felfedezése. Bejelentkezésre is szükség lesz és persze ahhoz, hogy be tudjunk jelentkezni, valahol felhasználót is csinálnunk kell. Így természetesen létezik egy regisztráció menü pont is.

Követelmények (Fejlesztői környezet, webszerver)

A záródolgozatom megírásához a NetBeans 8.0-es verzióját használom. A web szerver futtatását az XAMPP 3.1.0 Beta 6 Apache és MySQL szerverének a segítségével hajtom végre. Verziókezelésre GitHub-ot alkalmazok, így záródolgozatom nyomon követhető a fejlesztési folyamat, megkönnyíti az esetleges korábbi verzióban jól működő elemek visszakeresését, így a frissítéseket követő hibák könnyedén észrevehetőek. Háttérképek szerkesztésére, a vízjelek eltávolítására az Adobe Photo shop CS6-os verzióját vettem segítségül.

Képernyőtervek

Hogy mi is az a képernyőterv? A képernyőterv az előre elkészített megjelenés, vagyis az oldal gombjainak, stílusának elkészített terve. Ez elengedhetetlen az oldal felépítéséhez. Képernyőterv tartalmaz gombokat, textfieldeket, boxokat password fieldet. linkeket labeleket.



Adatbázis tervek

A weboldalamhoz készített adatbázis négy táblából áll (csomagok, feladók, users, szállítási _pontok). Ebben az adatbázisban kerül tárolásra minden felvett adat. A táblák felépítése a következő módon néz ki:

- **Csomagok:**

- id (integers, see auto increment)
- felado_id (integer)
- suly(integer)
- címzett_neve (varchar:60 character hosszú)
- szallitasipont_id (integer)
- feladás_idelye (datetieme)
- torekeny (tinyint)
- szallitas_kezdet (date time)
- érkezés_idelye (date time)
- átvétel_idelye (date time)
- azonosito (varchar:20 character hosszú)
- szallito_id (integer)

- **Feladók:**

- id (integer)
- nev (varchar :60 character hosszú)
- cegnev (varchar :60 character hosszú)
- adoszam (varchar :13 character hosszú)
- irszam (varchar :4 character hosszú)
- telepules (varchar :60 character hosszú)
- utcahaz (varchar :100 character hosszú)
- email (varchar :100 character hosszú)
- telefon (varchar :15 character hosszú)
- maganszemely (tinyint)

- **szallitasi_pontok:**

- id (integer)
- megnevezes (varchar:60 character hosszú)
- irszam (varchar:4 character hosszú)
- varos (varchar:100 character hosszú)
- utcahaz (varchar:100 character hosszú)
- telefon (varchar:15 character hosszú)
- email (varchar:100 character hosszú)
- kapcsolattarto (varchar:60 character hosszú)

- **users**

- id (integer)
- name (varchar:20 character hosszú)
- uname (varchar:25 character hosszú)
- upass (varchar:50 character hosszú)
- email (varchar:80 character hosszú)
- rights (tinyint)
- aktiv (tinyint)

A users táblába kerülnek a regisztráció során felvett adatok. Az id mező auto increment, ez azt jelenti, hogy automatikus sorszámozást kapnak. A uname mező varchar típusú egyedi mező, ami 25 karaktert tartalmazhat maximálisan. A felhasználói jogosultságok a rights mezőbe kerülnek, ami tinyint típusú mező.

A rights table nem tartalmaz túl sok adatot. Csak az id és a description mezőket tartalmaz. A description ötven karakteres varchar mezőtípus.

A csomagok table igen csak sok adatot tartalmaz. Ez tartalmazza a címzett nevét, a csomag súlyát és azt, hogy törékeny-e a csomag. Ezek mellett persze sok más szükséges dátumot is tartalmaz (feladás_ideje, szállítás_ideje, érkezés_ideje, átvétel_ideje). És többek közt egy azonosító mezőt, ami a csomagok keresésében vesz részt. A programkódban van egy kódgenerátor. Generál egy kódot, amit akkor kapunk meg, ha felvittük a csomagot. E azonosító alapján tudunk rákeresni csomagunk adataira.

A feladok táblába értelemszerűen a feladók adatai találhatók. Ilyenek a feladók neve, adószáma, magánszemély-e vagy nem, az esetleges cégnév, irányítószám, település, utca-, ház-, és telefonszám.

A szállítási_pontok tartalmazza a szállítási pont címét. Ezek pedig: megnevezés, irányítószám, város, telefon, email, és egy úgynevezett kapcsolattartót.

A users table tartalmazza a már regisztrált felhasználók adatait, vagyis azokat, akik legalább egyszer igénybe vették oldalunk szolgáltatásait. Ezek pedig a name, username, password, email.

Programkód felépítése (mappák, fájlok szerepe)

A weboldalam 3 különféle részre van osztva: egy fejlécből (header), egy úgynevezett contentből (ez tartalmazza a különféle tartalmakat, vagy táblázatokat), és egy láblécből. Ennek köszönhető az a tulajdonság, hogy az egyes részekben történt apróbb változtatásokkor nem kell mindegyik oldalt változtatni. Ez úgy értendő, hogy, ha egy gombot meg akarok változtatni a headerben, akkor nem kell a footer programkódján változtatásokat eszközölnöm. Emellett minden egyes oldal két különböző fájlra van bontva, ezek pedig a controllers és a view.

Minden oldal behívásakor az Index.php töltődik be. A többi lap az URL paraméterek alapján jelenik meg. Az index.php fájlba vannak behívva a configból az adatbázissal összekapcsoló adatok. A config.php fájl az úgynevezett változókat tartalmazza. Ezek a változók az adatbázis kapcsolatot tárolják. A config.default.php lényegében megegyezik a config.php-val, ugyanis ez csak azért van jelen, mert a githubos verziókezeléshez ehhez volt szükség. Annyiban tér el a tartalmuk, hogy a változóknak nem szerepelnek az adatok.

A controllers mappa tartalmazza a controllers fájlokat. Minden itt szereplő fájlban van egy, a megjelenítésért felelős része is, ez a views mappa tartalmazza. A controllers tartalmazza a különféle SQL lekérdezéseket, itt gyűjtjük ki a viewsban megadott adatokat. FrontPage.php változókat tárol, ide kerül bele az aktuális dátum, a bejelentkezett felhasználó id-je, és a textarea-ban létező szöveg, majd az ezekben lévő adatok egy lekérdezés segítségével

bekerülnek egy adatbázisba, majd a legfrissebb hírek id szerint rendezésre kerülnek. A regPage.php-ban változók találhatók. Egy SQL lekérdezés felviszi az adatbázisba a változóban található adatokat, az adatbázis megfelelő mezőibe.

A views mappában tárolt fájlok felelnek az oldalak megjelenéséért. A FrontPage.php ezen része kiírat néhány hírt, vagy változást, esetleg újítást a honlapon, vagy a cég szolgáltatásaival kapcsolatban. A regPage.php oldal 3 db input mezőt tartalmaz, (hiányzó vessző) melyek segítségével lehet regisztrálni a weboldalra. A contactPage.php a kapcsolat oldal megjelenítését végzi. Ez a kapcsolat a szerkesztők felé irányul az esetleges hibák, vagy információk szerzéséhez.

A views-on belül fellelhető egy css nevű mappa, abban pedig egy base.css fájl. Ennek feladata az oldal külső kinézetének beállítása, egy viszonylag a mai modern világ stílusához alkalmazkodó, de nem túlságosan feltűnő koncepció felvétele. Többek közt itt kerülnek beállításra az oldal színei és elrendezése. A fejlécbe egy képet fogok beszúrni és az ott található linkek, gombok, betűinek színét ennek megfelelően fogom beállítani, vagyis hogy könnyen és egyszerűen olvashatóak legyenek. Az oldal háttéréhez is egy képet fogok használni, természetesen egy olyan, vagy hasonló stílusú kép, mint amit a headerben, vagyis a fejlécben használtam. E mappában található még egy includes nevű mappa. Ami még 3 fájlt tartalmaz, azok a fájlok a header.php, footer.php, navigation.php. A header már korábban említésre került, ez tartalmazza az oldal logóját és nevét. Navigation.php, mint neve is mondja, egy úgynevezett navigációs rendszert tartalmaz. Ez azt jelenti, hogy ez tölti be a kért oldalt. Ez a navigation.php a header.php fájlban be van hívva egy include paranccsal.

Documents-en belül van a Logó mappa, ahol a css-ben használt háttérképeket tárolja és a wireframes, ami a képernyőtervek tárolására szolgál.

Ezekon kívül található egy keresesPage.php egy packetPage.php, regPage.php. A keresesPage.php olyan változókat tartalmaz, melyek egy előre megadott csomag azonosítót fognak keresni az adatbázisunkban. A packetPage.php azokat a változókat tartalmazza, melyek egy csomag felviteléhez szükségesek. Néhány példa: Feladó, Címzett neve, Szállítási cím.

A regPage.php egy formon belül néhány labelt tartalmaz, ezeket kell az oldalon kitölteni és ennek folyamán tudunk regisztrálni. E mezők kitöltése elengedhetetlen.

Található egy szalitPage.php, ez változókat tartalmaz, ahol a szállítási pontokat tudjuk megkeresni.

További fejlesztési lehetőségei

További fejlesztési lehetőségek: az oldalam továbbfejlesztéséhez nem elegendő a programkód

megváltoztatása, bővítése. Ehhez egyértelműen szükség van az adatbázis bővítésére is.

A regisztráció menüponthoz például a nagyobb biztonság érdekében elhelyezhetnénk egy úgynevezett csapcsát. Ennek feladata pedig az lenne, hogy tudnánk, hogy a regisztrált felhasználó nem csak egy erre a célra tervezett bot, hanem tényleg látja a csapcsában kért üzenetet és ezzel megerősít minket személyéről.

Ezekon felül azt is végre tudnánk hajtani fejlesztés képpen, hogy az ügyfelek GPS segítségével nyomon tudják követni csomagjaik útvonalát. De akár azt is meg tudjuk valósítani, hogy ha egy ügyfél csomagja elindulásra kerül, akkor a rendszer egy előre megírt e-mailt küld a felhasználó megadott e-mail címére, ha esetleg már a kiszállítási ponton van a csomag, és az ügyfél nem tudja hogy mikor ér oda, így a rendszer akkor is küld egy e-mailt, amikor a csomag a kiszállítási ponton van.

Egy igencsak hasznos bővítményt is tudunk alkalmazni. Ez pedig az lenne, hogy regisztrációkor a rendszer kéri a települést, város nevét ahol él. Ennek megfelelően a rendszer kiválasztaná a felhasználó lakhelyéhez legközelebb fekvő szállítási pontot, ahol átveheti csomagját.

Az is egy hatalmas előrelépés lenne az oldal folyamán, ha a csomagok kezelése az admin oldalon lenne elérhető. Ez azt eredményezné, hogy csak azok látnák az oldalt, akiknek már van felhasználójuk. Ez most is így működik, csak nem használók hozzá egy admin oldalt.

Felhasználói dokumentáció

Ismertető

Ez a weboldal egy szállítmányozással foglalkozó cég weboldala. A nevéből is kiderül, hogy mivel foglalkozik, nyilvánvalóan csomagokat szállít egy adott helyről egy megadott szállítási pontra, ahol címzett át tudja venni csomagját, elérhető benne egy keresési funkció, mely egy csomag azonosítón alapul. A CsomagPage.php tartalmaz egy úgynevezett kódgenerátort. Ez a kódgenerátor generál egy kódot az ügyfél számára, aki ennek a kódnak a segítségével hozzájut csomagjainak adataihoz. Létezik egy belépés menüpont, ehhez természetesen tartozik egy regisztráció funkció is. A csomagok kezelésében egy táblázatot látunk az aktuálisan felvett csomagokról és itt tudunk felvinni újabb csomagokat.

Futtatási környezet

A záródolgozatomban elkészített weboldalam tetszőleges operációs rendszer alatt, tetszőleges böngészővel futtatható, nem igényel semmi egyéb programot vagy kiegészítőt a böngészőben.

A program használatának leírása

Ez a weboldal használható regisztráció, vagy bejelentkezés nélkül is, de így csak a kezdőlap, a keresés, és a kapcsolat használható teljes mértékben. De ez sem igaz teljesen, mert a keresés funkció ugyanis egy azonosító kód alapján működik, amit természetesen nem kap minden egyes úgynevezett vendég, aki a weboldalra téved. Ennek a funkció teljes mértékű használatához szükséges az azonosító kód. A kezdőlapon a legfrissebb híreket fogja találni a böngésző személy. Ezeknek szerepe figyelemfelkeltés, szüksége lenne a cég szolgáltatásaira, itt tájékozódhat a legfrissebb hírekről, vagy esetleges szállítási módszerekről. Ha regisztrált és be is jelentkezik az úgynevezett felhasználó nevéen, az oldal köszönti. A keresés menüpontban egy kapott azonosítóra kell rákeresni. Így fogja megtalálni csomagja adatait. A csomagok kezelésében tudja felvinni a felhasználó a csomagot kizárólag az után, hogy bejelentkezett egy már korábban létrehozott felhasználóval. A belépés menüpont alatt létezik egy regisztráció menüpont, ami átirányít minket a regisztráció oldalra. Itt tudjuk létrehozni felhasználói accountunkat. Emellett a csomagok kezelésében van egy táblázat, ami visszaadja az eddigiekben felvitt csomagok adatait. A bemutatkozás menüpontban a cég írja le az eddigiekben elért sikereit és néhány felhasználó véleményét a cégről. A kezdőlapon a legfrissebb hírek láthatóak, amik a cég újdonságairól, vagy következő időkben várható fejlesztésekről szólnak. A kapcsolatok nevű menüpont a felhasználók véleményét, vagy esetleges fejlesztésekhez való ötleteiket tartalmazza, vagy esetleges hibákat, információszerzést adhat, vagy juttathat el a szerkesztőkhöz.

Irodalomjegyzék

<http://www.w3schools.com/php/default.asp>

-W3School PHP Tutorial: igencsak hasznos információkkal szolgált a zárdolgozatom elkészítésében.

<http://www.w3schools.com/sql/default.asp>

-W3School SQL Tutorial: W3School SQL tutorialt is igénybe vetünk a zárdolgozatom adatbázisának kezeléséhez.

Nyilatkozat

Alulírott Nyári Bálint kijelentem, hogy a záródolgozat saját munkám eredménye. A dolgozat elkészítéséhez felhasznált valamennyi forrást az irodalomjegyzékben feltüntettem a hivatkozás szabályainak megfelelően. A dolgozat elkészítéséhez meg nem engedett segédanyagokat nem használtam fel.

Záródolgozatomat más intézményben nem adtam be.

Kelt: Kaposvár, 2015.03.31.

.....
Aláírás

Szoftverfejlesztő szakos tanuló