

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Sandi Mikuš

**Spletna aplikacija za nadzor
senzorskih podatkov v preskrbovalni
verigi**

DIPLOMSKO DELO

VISOKOŠOLSKI STROKOVNI ŠTUDIJSKI PROGRAM PRVE
STOPNJE RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: doc. dr. Mira Trebar

Ljubljana, 2014

Rezultati diplomskega dela so intelektualna lastnina avtorja. Za objavljanje ali izkoriščanje rezultatov diplomskega dela je potrebno pisno soglasje avtorja, Fakultete za računalništvo in informatiko ter mentorja

Besedilo je oblikovano z urejevalnikom besedil L^AT_EX.

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika naloge:

Preskrbovalna veriga hitro pokvarljivih živil vključuje zahteve po nadzoru temperatur, ki se običajno pridobivajo na osnovi zapisov izvajalcev v procesih priprave, skladiščenja, prevozov in prodaje. Sodobne tehnologije omogočajo uporabo različnih senzorskih sistemov, ki zagotavljajo shranjevanje podatkov na strežniku. Zbrani podatki omogočajo uporabnikom stalen nadzor, vendar jih je potrebno prej analizirati in ustrezno predstaviti.

Kandidat naj v diplomskem delu izdelava spletno aplikacijo za obdelavo in prikaz podatkov, ki so pridobljeni v postopku sledljivosti in shranjeni v podatkovni bazi. Izmerjene temperature naj analizira in predstavi v časovnem grafu, izpiše pomembne ugotovitve in izračuna koliko časa je izdelek še uporaben. Rešitev naj bo splošna in izdelana z uporabo spletnih tehnologij (HTML, CSS, JavaScript, PHP, ...) in podatkovno bazo MySQL. Implementacija in testiranje naj vključujeta predstavitev aplikacije z uporabo realnih podatkov podjetja, kjer so bile izvedene meritve temperatur v povezavi s proizvodi.

IZJAVA O AVTORSTVU DIPLOMSKEGA DELA

Spodaj podpisani Mikuš Sandi, z vpisno številko **63100383**, sem avtor diplomskega dela z naslovom:

Spletna aplikacija za nadzor senzorskih podatkov v preskrbovalni verigi

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvom doc. dr. Mire Trebar,
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela,
- soglašam z javno objavo elektronske oblike diplomskega dela na svetovnem spletu preko univerzitetnega spletnega arhiva.

V Ljubljani, dne 5. september 2014

Podpis avtorja:

Zahvalil bi se mentorici, doc. dr. Mira Trebar, za strokovno pomoč in usmerjanje med izdelavo diplomskega dela. Prav tako bi se rad zahvalil svoji družini in ostalim, ki so mi stali ob strani vsa leta študija in vsem, ki so mi ob študiju pomagali.

Kazalo

Povzetek

Abstract

1	Uvod	1
2	Uporabljene tehnologije	3
2.1	HTML in CSS	3
2.2	Javascript	4
2.3	PHP	4
2.4	MySQL	5
2.5	LAMP	5
2.6	Apache	5
3	Opis problema	7
3.1	Podatki	8
3.2	Analiza in predstavitev podatkov	10
3.3	Opis izračuna preostale dobe uporabnosti	12
4	Opis rešitve	15
4.1	Načrtovanje aplikacije	15
4.2	Opis podatkovne baze	16
4.3	Izvedba spletne aplikacije	18
5	Sklepne ugotovitve	27

Seznam uporabljenih kratic

kratica	angleško	slovensko
CSS	Cascading Style Sheets	Kaskadne stilske predloge
EPC	Electronic Product Code	Standardizirani globalni identifikator
HHVM	HipHop Virtual Machine	Navidezni stroj za poganjanje HipHop vmesne kode
HTTP	Hypertext Transfer Protocol	Protokol za prenos hiperteksta
JIT	Just-in-time (compiler)	predčasni prevajalnik
LAMP	Linux, Apache, MySQL, PHP, Pearl	Skupek programov za razvoj spletnih aplikacij
PHP	Hypertext preprocessor	splošno uporaben skriptni programski jezik, ki ga tolmači strežnik
MySQL	Structured Query Language	Strukturirani jezik za poizvedbe
RDBMS	Relational database management system	Sistem za upravljanje relacijskih podatkovnih baz

Povzetek

V okviru diplomske naloge je bila izdelana spletna aplikacija, ki omogoča analizo podatkov o temperaturah, ki so bile izmerjene v preskrbovalni verigi in izračun preostale dobe uporabnosti. Aplikacija je na voljo podjetjem in potrošnikom, ki imajo na voljo različno količino podatkov za pridobitev informacij o živilskih proizvodih.

S pomočjo aplikacije je mogoče pregledovati nihanja temperature izdelka v času skladiščenja in med transportom. Sama aplikacija omogoča uporabo dveh različnih dostopov oz. pregledov nad podatki. Osnovni - gostujoči pogled na podatke, ki je namenjen potrošnikom. Razširjen pogled je namenjen uporabnikom neposredno v stiku s hladno verigo in jim omogoča celovit pregled podatkov vključno z obveščanjem. Obveščanje je samodejno in se izvede med analizo. Uporabnik z razširjenim dostopom tako v obvestilu izve podatke o napaki.

Ključne besede: hladna veriga, temperatura, spletna aplikacija, doba uporabnosti.

Abstract

The thesis describes the web application used for analysing temperature data measured in a supply chain and calculation of the remaining shelf life. The application is used for both businesses and consumers who have various amount of data.

Fluctuations in temperature of the product during storage and transportation can be reviewed with the help of the web application. The application itself allows you to use two different types of accesses or reviews for data; guest access and expanded access. Guest access is designed for consumers of those products. Expanded access is designed for users in direct contact with the cold chain and allows them a comprehensive overview of data and information. Information is automatic and ran during the analysis allowing the user to receive detailed information of any error.

Keywords: cold chain, temperature, web application, shelf life.

Poglavje 1

Uvod

Vsakodnevno se srečujemo s problematiko hitro pokvarljivih živil v živilski industriji. V preskrbovalni verigi morajo biti živila hranjena v območjih priporočljivih temperatur (hladna veriga). Zagotavljanje neprekinjene hladne verige pomeni vzdrževanje teh temperatur, neprekinjeno od proizvodnje izdelka, do njegove porabe. S pomočjo sodobne tehnologije lahko z uporabo različnih senzorjev tem temperaturam sledimo in jih shranjujemo na strežniških sistemih. Dandanes imamo naprave, ki nam omogočajo sprotni nadzor temperature, beleženje trenutne lokacije in shranjevanje podatkov (o temperaturi, lokaciji ipd.) na strežniške sisteme. Ker pa ti podatki sami po sebi ne nosijo nobenih informacij, jih je potrebno analizirati in predstaviti uporabniku.

V okviru diplomske naloge smo izdelali aplikacijo, ki uporabniku omogoča analizo in predstavitev podatkov. Uporabniku nudimo analizo temperatur, spremljanje logističnega postopka in kar je najbolj pomembno, izračun preostale dobe uporabnosti. Pri izdelavi aplikacije smo se srečali z vprašanjem, kaj je za posameznega uporabnika pomembno in kaj ne? Problem smo rešili z razdelitvijo uporabnikov v dve skupini, potrošnike in račune namenjene podjetjem. Potrošnikom smo omogočili dostop, do za njih najbolj pomembnih podatkov (pregled spreminjanja temperature skozi čas, lokacija produkta ter izračun preostale dobe uporabnosti). Uporabniškim računom namenjenim podjetjem, smo pa omogočili dostop do vseh podatkov. S tem smo jim omogočili lažji nadzor in pregled nad podatki.

Poglavje 2

Uporabljene tehnologije

Razvoj aplikacije je zahteval uporabo različnih tehnologij. Za samo obliko spletne aplikacije je bila potrebna uporaba HTML jezika in CSS stilskih predlog. Ker smo želeli dopustiti možnost spreminjanja stila in teme spletne aplikacije je bila izbira CSS stilskih predlog neizogibna. Za dinamičnost aplikacije skrbita PHP na strežniski strani in Javascript na odjemalčevi strani. S pomočjo MySQLi dodatka lahko v kombinaciji s PHPjem dostopamo do podatkov v bazi. Vse skupaj pa na strežniku podpira skupek programskih orodij LAMP in strežnik Apache.

2.1 HTML in CSS

HTML (Hyper Text Markup Language) je označevalni jezik namenjen označevanju besedilnih datotek [18]. S pomočjo označevalnega jezika določimo strukturo in semantičen pomen spletne strani. HTML značke so v glavnem sestavljene iz parov npr. `` in ``. Nekatere značke niso sestavljene iz parov in so same zase element. ``. Značke v parih so sestavljene iz začetne in končne značke. Spletni brskalnik prejme HTML datoteko in prikaže spletno stran. Ne prikazuje HTML značk, temveč jih uporabi za interpretacijo vsebine.

CSS (Cascading Style Sheets) oz. po slovensko kaskadne stilske predloge [18]. CSS opiše izgled in obliko dokumenta napisanega v označevalnem jeziku (npr. HTML). Z njim tudi določamo pravila, ki veljajo nad selektorji. CSS selektor je del CSS pravila, ki dejansko izbere vsebino, katero želi uporabnik oblikovati.

2.2 Javascript

Javascript (JS) je računalniški programski jezik [18]. Razvit je bil z namenom, da bi uporabnikom pomagal pri ustvarjanju interaktivnih spletnih strani. Uvrščen je v skupino skriptnih jezikov z dinamičnimi tipi. Jezik podpira več paradig (objektno, funkcijsko in imperativno). Javascript koda se, tako kot HTML in CSS prenese na odjemalca in se interpretira v brskalniku.

2.3 PHP

PHP (Hypertext Preprocessor) je skriptni jezik, ki se izvaja na strežniku [15]. Namenjen je razvoju spletnih aplikacij, ter povezavi spletne aplikacije s podatkovno bazo. Občasno se ga uporablja tudi kot splošno namenski programski jezik. Prvotni pomen kratice PHP je bil Personal Home Page (osebna spletna stran). PHP je bil napisan kot skupina CGI programov v programskem jeziku C. Napisal ga je dansko kanadski programer Rasmus Lerdorf leta 1994, zgolj za potrebe izgradnje enostavnih domačih spletnih strani. PHP interpreter se poganja na strežniški strani in kot vhod vzame PHP izvorno kodo, ter vrne odjemalcu zgenerirano HTML kodo.

Zelo znana je velika optimizacija PHP-ja s strani družbe Facebook, to je HipHop prevajalnik. HipHop prevajalnik (HPHPC) prevede PHP izvorno kodo v C++ kodo, ter to nato v binarno izvršljivo kodo [9][10]. Pri Facebooku so zatrdili, da uporaba HipHop prevajalnika omogoča bistveno pohitritev in manjšo porabo virov. Druga velika optimizacija zopet pripada družbi Facebook in sicer HHVM in Hack. HHVM (HipHop Virtual Machine) je namenjen poganjanju Hack in PHP kode [8] [10]. HHVM uporablja JIT (just-in-time) prevajanje.

Programski jezik je sam po sebi prijazen in omogoča netipizirano programiranje, zaradi česar se ga da hitro naučiti. V PHP-ju je nastalo veliko uspešnih projektov: Joomla, Facebook, WordPress, Moodle, Yahoo, tumblr.

2.4 MySQL

MySQL je odprtokodni sistem za upravljanje z relacijskimi podatkovnimi bazami (RDBMS) [13]. Ime je dobil po ustanoviteljevi (Michael Widenius) hčerki, My. Kratica SQL pomeni Structured Query Language oziroma Strukturirani Jezik za Poizvedbe. Družina podatkovnih baz MySQL omogoča kompatibilnost med posameznimi različicami strežnika MySQL. S tem nam je poenostavljena priprava razvojnega okolja, saj lahko uporabimo razvojno orodje LAMP. Danes sta najbolj znana dva podatkovna pogona za tabele in sicer InnoDB in NDB Cluster. V lokalnem razvojnem okolju se uporablja InnoDB v produkcijskem gručnem sistemu pa NDB Cluster.

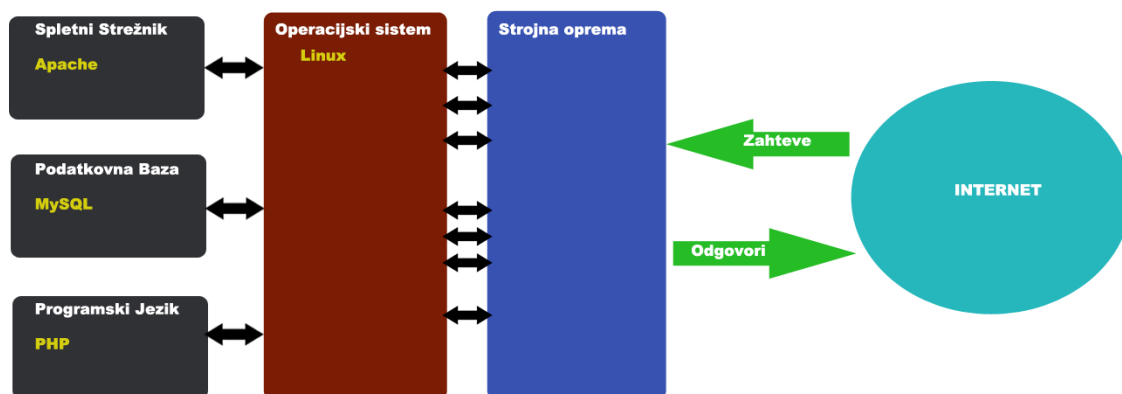
MySQLi, je razširitev sistema MySQL, ki omogoča uporabo večih funkcionalnosti v verziji MySQL 4.1 in novejših verzijah [14].

2.5 LAMP

LAMP je skupek programskih orodij namenjen lažjemu razvoju spletnih aplikacij. V osnovi je sestavljen iz štirih zamenljivih komponent: Linux, Apache HTTP Server, MySQL, PHP (Slika 2.1). Model LAMP je primeren za izgradnjo dinamičnih spletnih strani in spletnih aplikacij. Ekvivalentna verzija z operacijskim sistemom Microsoft Windows se imenuje WAMP [11].

2.6 Apache

Apache HTTP Server ali Apache, je spletni strežnik, ki igra ključno vlogo pri razvoju in širjenju spleta [3][4]. Bil je prva alternativa Netscapeovemu spletnemu strežniku. Ime "Apache" je bilo izbrano zaradi dveh razlogov. Prvi razlog je indijansko pleme Apači (znani po svojih vojaških sposobnostih in spretnostih), drugi razlog je pa izpeljanka iz besede "patches" (slov. set dodatkov), "a patchy". Iz tega je nastalo ime "a patchy server", kasneje spremenjeno v "Apache". S podatkov na spletni strani w3techs.com je razvidno, da je Apache uporabljen na 59.7 odstotkih strežnikov [19].



Slika 2.1: Predstavitev komponent, uporabljenih v skupku programskih orodij LAMP in njihovih povezav. [11].

Poglavje 3

Opis problema

S sodobno tehnologijo lahko brez večjih težav odčitamo temperaturo živilskih izdelkov, notranjosti hladilnika, skladišč, okolja ipd. ter jih shranjujemo na strežnikih. Ker so podatki sami po sebi brez pomena in ne nosijo nobenih informacij, jih je potrebno še analizirati in predstaviti prejemniku oziroma končnemu uporabniku. *”Informacija so podatki, katerih oblika in vsebina je primerna za določeno uporabo.”* (Alter, 1999) [17].

V našem primeru je bilo potrebno analizirati senzorske podatke v preskrbovalni verigi. S pomočjo analize smo lahko določili preostalo dobo uporabe izdelka. Nadzor in vzdrževanje predpisane temperature živila, da ohranimo varnost in čim boljšo kakovost živila v celotni živilski vrigi (od proizvodnje, prevoza, skladiščenja do porabe pri potrošniku), se imenuje hladna veriga (angl. cold chain) [5].

Doba uporabnosti (angl. shelf life) je obdobje, v katerem se lahko blago ali izdelek skladišči, da je primeren za uporabo ali porabo. To velja za živila, pijače, zdravila, druge farmacevtske izdelke, kemikalije ter mnoge druge pokvarljive izdelke [1].

3.1 Podatki

Za analizo, pregled in prikaz nadzora v preskrbovalni verigi smo uporabili realne podatke, ki se nahajajo v dveh podatkovnih tabelah [12] [16]. Shranjeni so v dveh bazah in vključujejo podatke o: (i) hladni verigi z izmerjenimi temperaturami; (ii) produktih, ki so bili nadzorovani v hladni verigi.

Iz teh tabel smo pridobili podatke in jih uporabili za analizo.

Celoten zapis podatkov o hladni verigi se nahaja v tabeli `data`, ki vsebuje 28 stolpcev.

`idrow` - id vrstice

`EPCU` - Uporabnik (EPC koda-NFC kartica)

`EPCSL` - START Lokacija (EPC koda-NFC nalepka)

`EPCCL` - END Lokacija (EPC koda-NFC nalepka)

`EPCTL` - NFC-TL (EPC koda)

`EPCPR` - EPC koda produkta

`EPCB` - EPC koda embalaže

`EPCP` - EPC koda zaboja

`EPCT` - EPC koda tovornjaka

`STATUS` - Bit, ki pove če je merjenje še v procesu (1) ali končano (0)

`ST` - Datum začetka logističnega postopka

`ET` - Datum konca logističnega postopka

`NM` - Število meritev

`LF` - Način beleženja (Dense, Out of limits...)

`INT` - Dolžina intervala (v sekundah)

`LL` - Spodnja meja temperaturnega območja

`ELL` - Skrajna spodnja meja temperaturnega območja

`UL` - Zgornja meja temperaturnega območja

`EUL` - Skrajna zgornja meja temperaturnega območja

`LLC` - lower limit count

`ELLC` - extreme lower limit count

`ULC` - upper limit count

`EULC` - extreme upper limit count

LATS - Zemljepisna širina začetne lokacije
 LONS - Zemljepisna dolžina začetne lokacije
 LATE - Zemljepisna širina končne lokacije
 LONE - Zemljepisna dolžina končne lokacije
 MEA - meritve (zapisane v obliki "Ti;i:Ti+1;i+1:...")

V stolpcu MEA so shranjene meritve temperatur in sicer v obliki $T_i;i:T_{i+1};i+1:T_{i+2};i+2:\dots T_n;n$, kjer je i število prve meritve in n število zadnje meritve.

Polja EPCU, EPCSL, EPCEL, EPCTL, EPCPR, EPCP, EPCB, EPCT imajo podatke shranjene v formatu GS1 (Tabela 3.1) [7].

Polje EPCU uporablja format GSRN (angl. Global Service Relation Number), polje EPCSL uporablja SGLN (angl. Serialized Global Location Number), polje EPCPR uporablja SGTIN (angl. Serialized Global Trade Item Number), ostala polja pa uporabljajo format GRAI (angl. Global Returnable Asset Identifier) [12].

Ključ	Tip	Filter	Podjetje	Razred	Serijska številka
GSRN	biti 0-7	biti 8-10	biti 14-43	0 bitov	biti 44-71
GRAI	biti 0-7	biti 8-10	biti 14-43	biti 44-56	biti 58-95
SGTIN	biti 0-7	biti 8-10	biti 14-43	biti 44-56	biti 58-95
SGLN	biti 0-7	biti 8-10	biti 14-43	biti 44-54	biti 55-95

Tabela 3.1: Razčlenitev formata GS1 na pomembne podatke

3.2 Analiza in predstavitev podatkov

Pred pričetkom razvoja aplikacije je bilo podatke potrebno analizirati in pregledati. Ker so vse temperature v zapisane v obliki niza, smo vse razbrane temperature iz niza spravili v polje (angl. array). Iz take podatkovne strukture smo jih veliko lažje analizirali in jih prikazali v obliki grafa (v obliki spreminjanje temperature v odvisnosti od časa). Med pregledovanjem podatkov prebranih iz podatkovne baze, preverimo še pravilnost temperaturnih območij. Primer narobe definiranega temperaturnega območja - polje z višjo temperaturo ima shranjeno temperaturo nižje od polja z nižjo temperaturo. Temperature razdelimo v tri območja:

- priporočljivo,
- pogojno dobro in
- kritično

temperatirno območje. Pri produktu pogledamo koliko časa je bil izpostavljen vsakemu izmed treh naštetih območij. V kolikor je bil produkt izpostavljen kritičnemu območju 5% celotnega časa ali več, se uporabniku pojavi obvestilo, ki ga o tem obvesti. Za pridobitev podatkov o lokaciji, aplikacija uporablja Googlovo aplikacijo Geocoding [6]. Aplikacija iz podatkovne baze prebere zemljepisni širini (angl. latitude) in dolžini (angl. longitude), ter s pomočjo Googlove aplikacije prebere podatke o lokaciji (kraj, pošta, država).

Za lažji pregled podatkov smo omogočili več možnosti. Podatki so urejeni po ID-jih (stolpec idrow) in po datumu (stolpec ST). Uporabnik lahko izbira tudi v spustnem seznamu, filtriranem po statusu (stolpec STATUS). Pri tej možnosti lahko uporabnik izbira med že zaključenimi meritvami in meritvami v izvajanju. S tem ima lahko uporabnik (predvsem administrator) nadzor, katere meritve so že zaključene in katere še ne.

EPC kodo produkta izvemo iz stolpca EPCPR (EPC PRoduct). Podatke o temperaturah preberemo s stolpca MEA, kjer so shranjene vse meritve za izbrani produkt. Temperaturna območja pa določimo z uporabo podatkov prebranih s stolpcev LL (spodnja meja), ELL (skrajna spodnja meja), UL (zgoranja meja), EUL (skrajna zgoranja meja). Priporočljivo območje je med mejama LL in UL. Pogojno območje je med mejama ELL in LL ter med mejama UL in EUL. Ime produkta, izvemo iz tabele `product` [16].

Podjetje Fonda ima v senzorski podatkovni bazi data shranjen naslednji zapis.

urn:epc:idpat:sgtin:383004776.0042.1006268

Številka 383004776 predstavlja ID podjetja. Zadnja številka 1006268 je serijska številka produkta in jo lahko izpustimo pri nadaljnjem iskanju.

Da dobimo ime produkta, poiščemo ujemajoče polje `pureIdentity` v tabeli `product`.

Tako najdemo vrednost stolpca `pureIdentity`, *urn:epc:idpat:sgtin:383004776.0042.**.

Dobljeni rezultat nam vrne slovensko ime "Brancin - očiščen 400-600g" in angleško ime "Seabass - cleaned 400-600".

Za predstavitev podatkov smo se odločili za graf s prikazom temperatur v odvisnosti od časa in uporabo treh barv. Vsaka izmed barv predstavlja svoje temperaturno območje. Zelena barva predstavlja priporočljivo območje, oranžna pogojno in rdeča kritično temperaturno območje. Za izbiro teh barv smo se odločili zaradi konsistentnosti z mobilno aplikacijo, uporabljeno za odčitavanje podatkov s senzorjev [12].

3.3 Opis izračuna preostale dobe uporabnosti

Algoritem za izračun preostale dobe uporabnosti deluje tako, da analizira koliko časa je bil produkt na dani temperaturi in potem s pomočjo tabele (Tabela 3.2) izračuna preostalo dobo uporabnosti. Preostala doba uporabnosti je izračunana iz količnika najdaljša doba uporabnosti / trenutna doba uporabnosti glede na trenutno temperaturo, pomnoženega z dolžino intervala v dnevih. Tako izračunano preostalo dobo uporabnosti odštejemo od trenutne dobe uporabnosti in dobimo novo dobo uporabnosti produkta [2]. Tabela s podatki o spreminjanju dobe uporabnosti je v podatkovni bazi shranjena v tabeli `shelflife`.

Temperatura [°C]					0	1	2	3	4	5	6
Doba uporabnosti [dan]					8.00	6.50	5.00	4.00	3.50	3.35	3.15
7	8	9	10	11	12	13	14	15	16	17	18
3.00	2.70	2.35	2.00	1.75	1.40	1.40	1.60	1.20	1.10	1.00	0.92
19	20	21	22	23	24	25	26	27	28	29	30
0.89	0.85	0.83	0.80	0.78	0.75	0.73	0.71	0.69	0.67	0.62	0.60

Tabela 3.2: Spreminjanje dobe uporabnosti.

```
foreach loggedTemperature do

    tmpTemp = round(loggedTemperature);

    /* baseTemp: tabela spreminjanja dobe uporabnosti */
    size    = sizeof(baseTemp);
    /* inicializacija indexa */
    closestI = 0;

    /* iskanje soležnega indexa Temperatura - Dan */
    if(tmpTemp >= maxTemperature)
        closestI = maxIndex;

    if(tmpTemp <= minTemperature)
        closestI = minIndex;

    for(c = 0; c < size; c++)
        if(baseTemp[c] == tmpTemp)
            closestI = c;
            break;

    /* tmpDays: soležni dan v tabeli */
    tmpDays = baseDays[closestI];

    /* x = pretvorba dolzine intervala iz sekund v dneve*/
    x = interval / 86400;

    /* izracunana preostala doba uporabnosti */
    /* shelfLifeMax: najdaljsa doba iz tabele
       spreminjanja dobe uporabnosti */
    SLRemain = SLRemain - x*(shelfLifeMax/tmpDays);
```


Poglavje 4

Opis rešitve

S pomočjo aplikacije, je mogoče analizirati podatke, prebrane iz podatkovne baze, le ti pa so bili prebrani s senzorjev, uporabljenih v hladni verigi. Analiza nam prikazuje spreminjanje temperature izdelka med skladiščenjem in transportom. V aplikaciji smo želeli s pomočjo podatkov, ki jih imamo na voljo ponuditi uporabniku informativni izračun preostale dobe uporabnosti izdelka.

4.1 Načrtovanje aplikacije

Aplikacijo smo izdelali v svojem PHP ogrodju (angl. framework). Ogrodje je tako narejeno, da omogoča lažji nadzor nad spletno aplikacijo. Aplikacija je bila v času razvoja in testiranja postavljena na strežnik, na naslovu *http://sltemp.avatar-rpg.net*. Za delovanje aplikacije je bila potrebna tudi podatkovna baza od koder bo aplikacija poizvedovala in prenašala podatke. Podatkovna baza se je v času razvoja in testiranja nahajala na istem naslovu kot aplikacija. Pri izbiri podatkovne baze smo se odločili za relacijsko podatkovno bazo, ki temelji na sistemu in jeziku MySQL.

4.2 Opis podatkovne baze

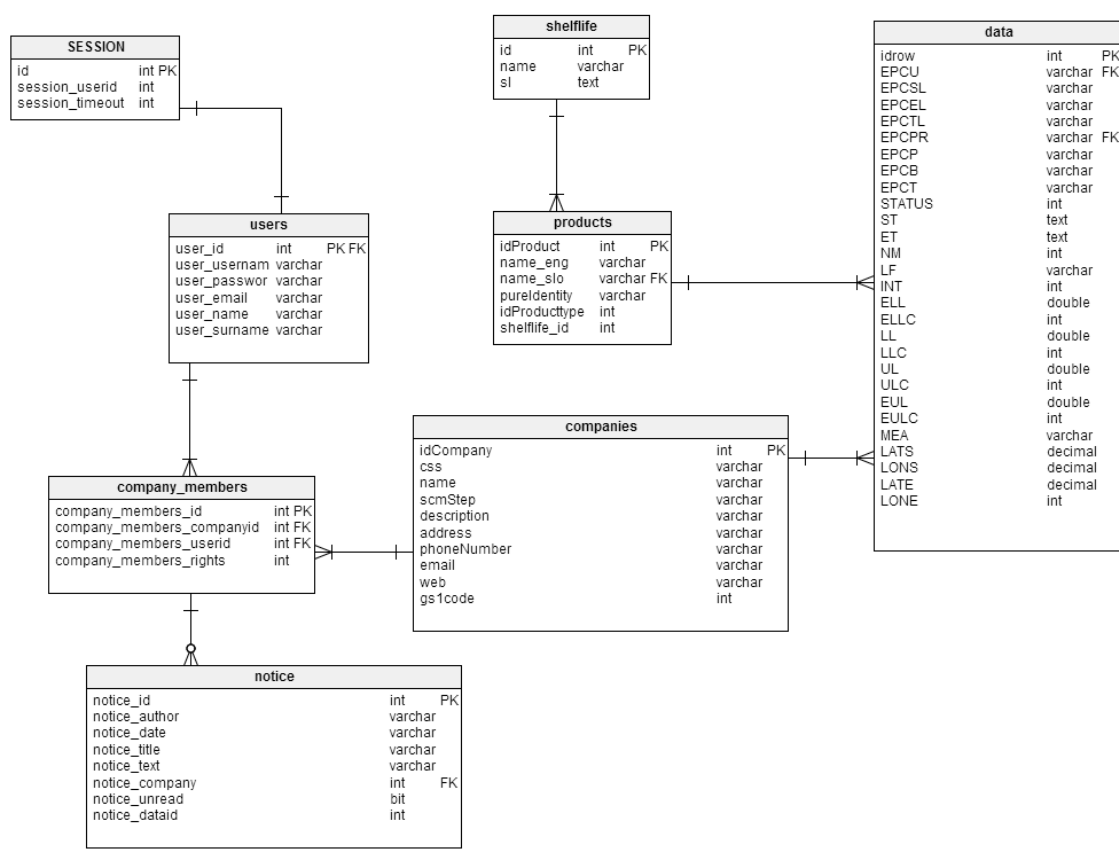
Podatkovna baza je sestavljena iz 8 tabel, ki pokrivajo podatke o uporabniku, podjetjih, izdelkih ter relacij med njimi (Slika 4.1).

- Tabela `SESSION` je namenjena zgolj vodenju evidence prijavljenih uporabnikov.
- Tabela `users` ima shranjene osnovne podatke o uporabniku (ID uporabnika, uporabniško ime in geslo, ime, priimek ter email naslov).
- Tabela `notice` ima shranjena obvestila, ki so prikazana uporabnikom v aplikaciji.
- Tabela `companies` vsebuje osnovne podatke podjetij.
- Tabela `company_members` povezuje tabele `users`, `companies` in `notice` med seboj in določa kater uporabnik pripada kateremu podjetju ter kakšne pravice ima v vsakem izmed njih.

Ostale so še tri tabele, katere vsebujejo podatke pomembne za analizo in izračun preostale dobe uporabnosti. Te tabele so (`shelflife`, `products` in `data`).

- Tabela `shelflife` vsebuje podatke o spreminjanju dobe uporabnosti produkta. Tabela ima tri stolpce (`id`, `name`, `sl`). V stolpcu `name` je ime produkta, v stolpcu `sl` so pa shranjeni podatki o spreminjanju dobe uporabnosti. Podatki so v formatu `T1:DU1<>T2:DU2<>...`, kjer je `T` temperatura in `DU` doba uporabnosti. Stolpec `id` je pa zaporedna številka vrstice.

Tabeli `products` in `data` sta bili vnaprej podani in vključeni v podatkovni model naše aplikacije.



Slika 4.1: Model podatkovne baze.

4.3 Izvedba spletne aplikacije

Registracija od uporabnika zahteva zgolj osnovne podatke in za uporabnika ni zahtevna (Slika 4.2). Z izbiro podjetja pri registraciji, dobite dostop do podatkov tega podjetja. Po uspešno opravljeni registraciji se izbrani podatki vnesejo v podatkovno bazo (v tabelo *users*). Dodano varnost v podatkovni bazi omogoča zgoščevalna funkcija izvedena nad uporabniškim geslom. S tem preprečimo shranjevanje gesel v obliki *golega besedila* (angl. plain text). Registracija je potrebna samo pred prvo uporabo, za nadaljno uporabo ni potrebno, ker so podatki že shranjeni v podatkovni bazi.

Po tem ko se uspešno registriramo, se lahko z uporabniškim imenom in geslom prijavimo v aplikacijo in jo začnemo uporabljati (Slika 4.3). V kolikor se poskušamo prijaviti z neregistriranim računom, nas na to aplikacija opozori in vrne napako pri prijavi. Funkcionalnosti aplikacije so uporabniku omogočene zgolj ob uspešni prijavi.

Aplikacija je zasnovana tako, da omogoča dva tipa uporabnikov, to sta Administrator podjetja in Uporabnik, ki je pravzaprav potrošnik. V nadaljevanju sta predstavljeni obe možnosti, ki prikazujeta razlike v posredovanju informacij na spletni strani.

SLTemp

Uporabniško ime Geslo

Ime

Priimek

Uporabniško ime (dovoljene samo črke)

Geslo
Dolžina gesla mora biti vsaj 8 znakov.

Geslo Ponovno

E-Mail

Podjetje

2014 © SLTemp

Slika 4.2: Registracija uporabnika.

SLTemp

Uporabniško ime Geslo

 Cold chain
Food safety

2014 © SLTemp

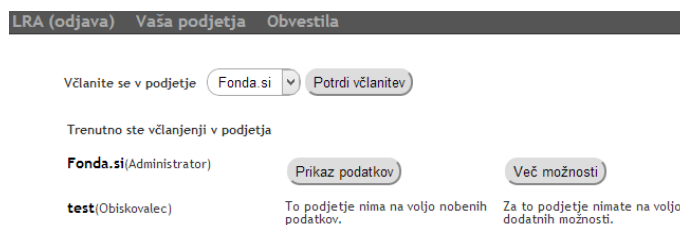
Slika 4.3: Prijavna stran v aplikacijo.

4.3.1 Administrator podjetja

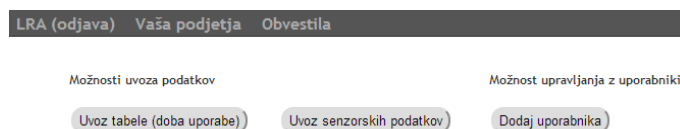
Administrator ima v aplikaciji na voljo naslednje funkcionalnosti:

- dodajanje uporabnikov,
- uvoz senzorskih podatkov,
- uvoz podatkov o spreminjanju dobe uporabnosti,
- dostop do obvestil,
- dostop do vseh shranjenih podatkov, ki so analizirani in prikazani v aplikaciji.

V zavihku podjetja, je seznam podjetij v katera ste včlanjeni ter možnost včlanitve v nova podjetja. (Slika 4.4). Pri podjetjih, v katerih imate administratorske pravice, imate ob strani na razpolago gumb *Več možnosti* (Slika 4.5). V meniju zo dodatnimi možnostimi imamo dve možnosti. Uvoz podatkov in dodajanje uporabnika. Pri uvozu podatkov lahko izbiramo med uvozom senzorskih podatkov in uvozom podatkov o spreminjanju dobe uporabnosti. Uvoz vseh podatkov poteka z uvozom *.csv* datotek.

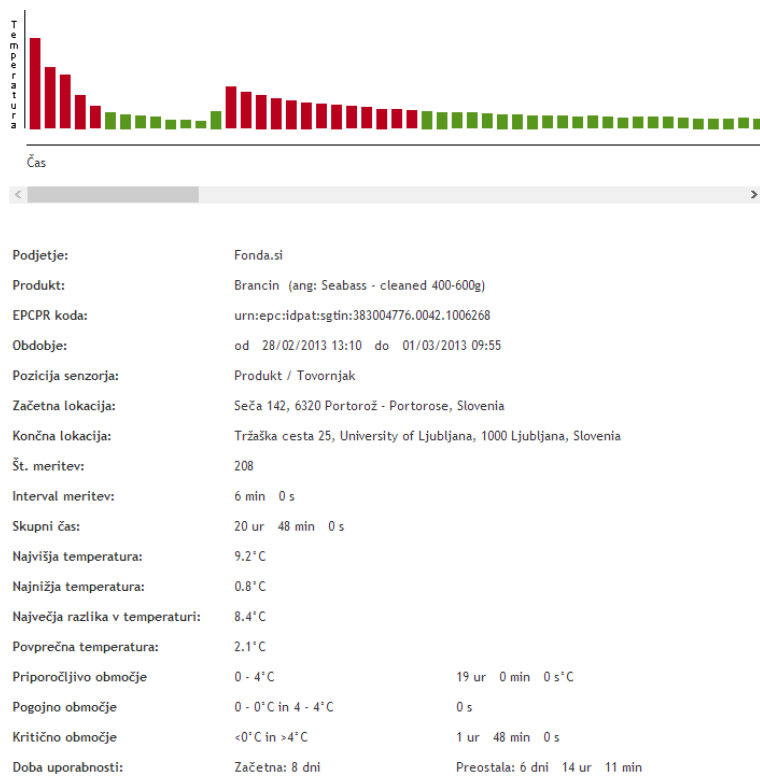


Slika 4.4: Izbira podjetij. Ob imenu podjetja je napisana vloga uporabnika v tem podjetju.



Slika 4.5: Meni z dodatnimi možnostimi.

Pri predstavitvi podatkov uporabniku, imajo administratorji omogočen pogled nad vsemi podatki (Slika 4.6).

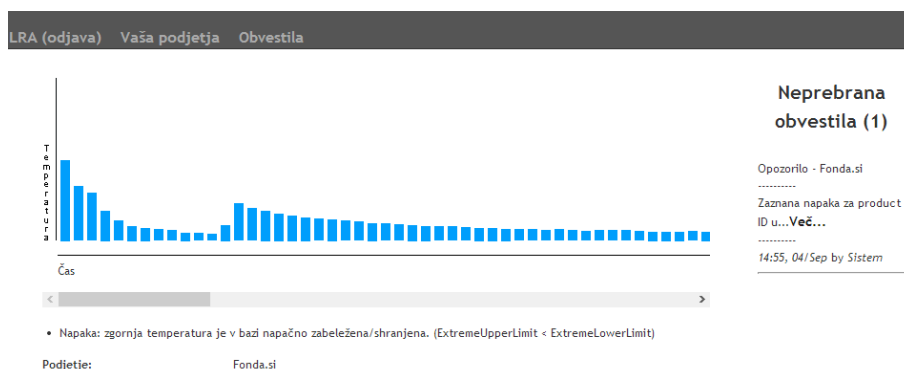


Slika 4.6: Predstavitev podatkov z razširjenim dostopom.

Podjetje	Ime podjetja kateremu pripadajo podatki o produktu.
Produkt	Ime produkta v slovenščini in angleščini.
EPCPR koda	EPC koda produkta.
Obdobje	Obdobje od prve do zadnje opravljene meritve.
Pozicija senzorja	Kje je bil senzor vstavljen (produkt, embalaža, tovornjak...).
Začetna lokacija	Začetna lokacija, kjer je bil senzor vstavljen.
Končna lokacija	Ciljna lokacija v logističnem procesu.
Št. meritve	Število izmerjenih meritev (intervalov).
Interval meritev	Dolžina intervala posamezne meritve.
Skupni čas	Skupni čas meritev.
Najvišja temperatura	Najvišja izmerjena temperatura.
Najnižja temperatura	Najnižja izmerjena temperatura.
Največja razlika v temperaturi	Razlika med najvišjo in najnižjo temperaturo.
Povprečna temperatura	Povprečje izmerjenih temperatur.
Priporočljivo območje	Priporočljivo temperaturno območje.
Pogojno območje	Pogojno temperaturno območje.
Kritično območje	Kritično temperaturno območje.
Doba uporabnosti	Doba uporabnosti v začetku logističnega procesa in preostala doba uporabnosti, po koncu logističnega procesa.

Tabela 4.1: Opis podatkov, ki jih vidi administrator podjetja.

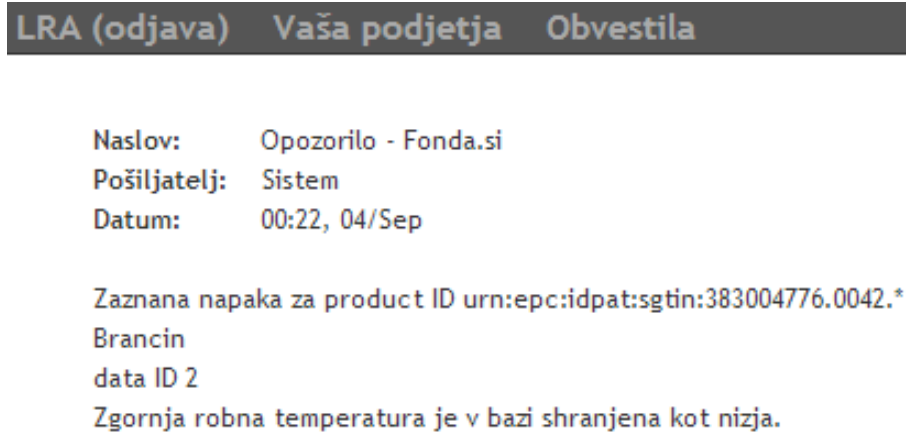
Administratorjem podjetij je na voljo tudi obveščanje. Aplikacija uporabnika obvešča o napakah, ki jih zazna med analizo (Slika 4.7). Do istih podatkov lahko dostopa več uporabnikov naenkrat in en uporabnik je lahko včlanjen v več podjetij. Ker v podatkovni bazi ne želimo imeti podvojenih vnosov, so obvestila vezana na podjetja. Vsakdo, ki je včlanjen v podjetje in ima razširjen dostop lahko prebere obvestila. Vsa obvestila so shranjena v zavihku **Obvestila** (Slika 4.8). V obvestilih so osnovni podatki o napaki (Slika 4.9).



Slika 4.7: Prikaz obvestila o neprebranih obvestilih.



Slika 4.8: Nabiralnik z obvestili.



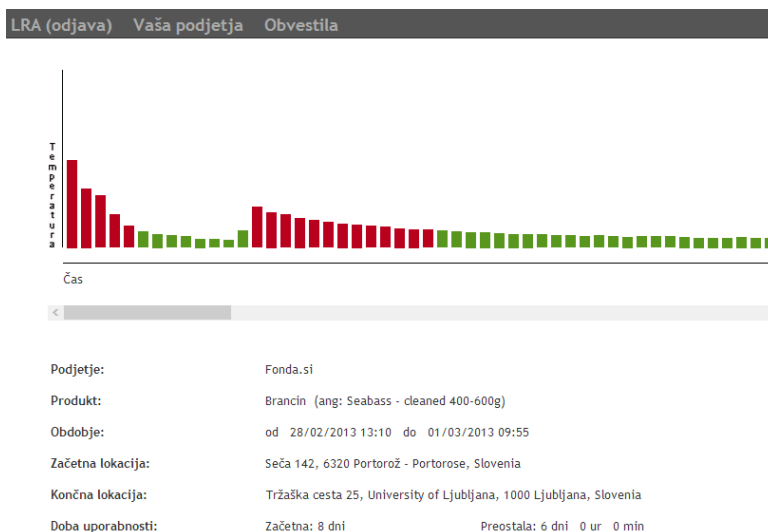
Slika 4.9: Obvestilo s podatki o napaki in produktu.

4.3.2 Uporabnik

Navadni uporabniki oz. potrošniki imajo rahlo okrnjen izpis podatkov. Potrošnikom prikazujemo zgolj osnovne podatke (podjetje, produkt, obdobje, lokacijo in dobo uporabnosti). Zaradi lažje predstavitve potrošnikom, graf uporablja samo dve barvi (zeleno in rdečo). S tem omogočimo potrošnikom lažje razumevanje grafa. To lahko naredimo zato, ker je pogojno dobro območje zelo blizu priporočljivega območja.

Opis podatkov, ki jih vidi potrošnik.

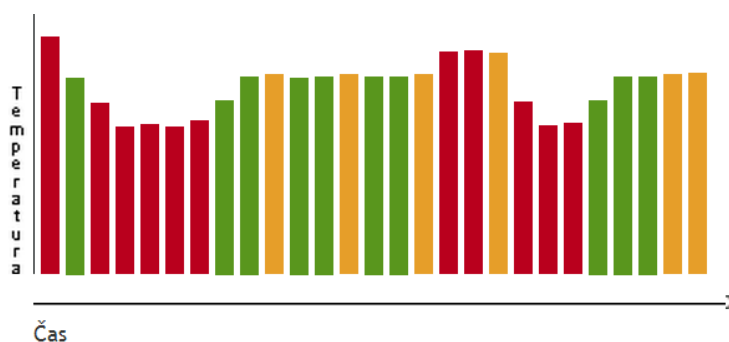
Podjetje	Ime podjetja kateremu pripadajo podatki o produktu.
Produkt	Ime produkta v slovenščini in angleščini.
Obdobje	Obdobje od prve do zadnje opravljene meritve.
Začetna lokacija	Lokacija kjer je bil senzor vstavljen.
Končna lokacija	Ciljna lokacija v logističnem procesu.
Doba uporabnosti	Doba uporabnosti v začetku logističnega procesa in preostala doba uporabnosti, po koncu logističnega procesa.



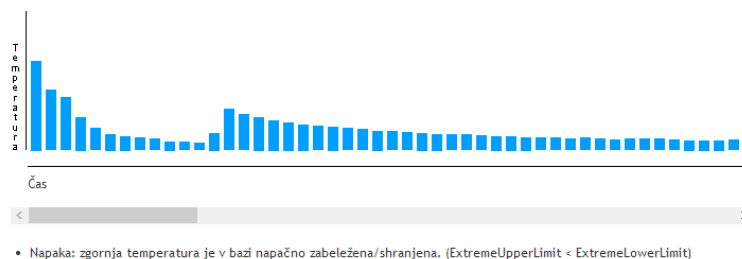
Slika 4.10: Predstavitev podatkov navadnemu uporabniku.

4.3.3 Opis dostopov

Oba dostopa (dostop kot administrator ali potrošnik) imata omogočen pogled grafa, ki prikazuje spreminjanje temperature skozi čas. Graf uporablja 4 različne barve (zeleno, oranžno, rdečo in modro) (Slika 4.11). Zelena barva sporoča, da je s temperaturo vse v redu, ter je znotraj predpisane temperaturega intervala. Oranžna barva označuje manjši odklon od predpisane temperature, toda še vedno ni kritično ali nevarno. Rdeča barva sporoča uporabniku nevarnost. Produkt je v tem času bil izpostavljen temperaturam višjim ali nižjim od predpisane meje, oziroma je izven temperaturnega intervala, ki je še toleriran. Modra barva je uporabljena v primeru napak, da barvna legenda ne zmede uporabnika (Slika 4.12).



Slika 4.11: Uporaba barv v grafu.



Slika 4.12: Modro obarvan graf z opisom napake.

Poglavje 5

Sklepne ugotovitve

V okviru diplomske naloge smo želeli analizirati podatke, ki se zbirajo v procesih hladne verige. Na podlagi dobljenih podatkov lahko aplikacija uporabniku izriše graf spreminjanja temperature v odvisnosti od časa. S pomočjo grafa lahko vidimo, kako se je spreminjala temperatura v logističnem procesu (od skladiščenja, transporta pa do cilja). Med samo analizo podatkov, aplikacija pregleduje za morebitne napake in uporabnika o tem obvesti. Z aplikacijo lahko izračunamo preostalo dobo uporabe živil.

S pomočjo napredka tehnologije je implementacija hladne verige dandanes cenejša in bolj enostavna kot pred leti. V aplikaciji imamo že nekaj testnih podatkov podjetja Fonda. Za zdaj imamo zgolj peščico podatkov, toda upamo, da jih bo v prihodnosti več. Aplikacijo bi v prihodnosti bilo mogoče nadgraditi. Ena izmed nadgradenj je analiza podatkov v realnem času (angl. real-time data analysis). V kolikor bi bilo veliko zahtevkov za analizo istih podatkov, bi aplikacijo lahko nadgradili s predpomnjenjem podatkov.

Prototipna aplikacija je tudi podlaga za nadaljni razvoj te ali drugih samostojnih aplikacij. V kolikor si lahko sami doma beležimo podatke, jih lahko uporabimo v aplikaciji in si sami analiziramo naša živila in mogoče celo optimiziramo dobo uporabnosti.

Literatura

- [1] John P. Doyle, "Seafood Shelf Life as a Function of Temperature", 1989
- [2] Dr. William Roberts, Dr. James L. Cox, "Proposal for Standardized Core Functionality in Digital Time-Temperature Monitoring SAL Devices", April 2003
- [3] (2014) Apache, HTTP Server, dostopno na:
<http://httpd.apache.org>
- [4] (2014) Apache, dostopno na:
<http://www.apache.org>
- [5] (2014) NIJZ, dostopno na:
http://ivz.si/Mp.aspx?ni=23&pi=5&_id=1664&_PageIndex=1&_groupId=193&_newsCategory=&_action=ShowNewsFull
- [6] (2014) The Google Geocoding API, dostopno na:
<https://developers.google.com/maps/documentation/geocoding/>
- [7] (2014) GS1 Slovenija, dostopno na:
<http://www.gs1si.org>
- [8] (2014) Hack, dostopno na:
<http://hhvm.com>
- [9] (2014) HipHop Virtual Machine, dostopno na:
<https://www.facebook.com/notes/facebook-engineering/the-hiphop-virtual-machine/10150415177928920>

-
- [10] (2014) HipHop Virtual Machine, dostopno na:
<http://hhvm.com>
- [11] (2014) LAMP, dostopno na:
<http://linuxsolutions.org.in/lamp.html>
- [12] (2014) Matic Odar, "Nadzor temperatur z uporabo tehnologije NFC – Android aplikacija v sistemu sledljivosti", dostopno na:
<http://eprints.fri.uni-lj.si/2352/>
- [13] (2014) MySQL, dostopno na:
<http://dev.mysql.com/doc/refman/5.7/en/>
- [14] (2014) MySQLi, dostopno na:
<http://php.net/manual/en/intro.mysqli.php>
- [15] (2014) PHP: Hypertext Preprocessor, dostopno na:
<http://php.net>
- [16] (2014) Rok Erjavec, "Aplikacija za upravljanje identifikatorjev in podatkov v sistemu z NFC nadzorom temperatur", dostopno na:
<http://eprints.fri.uni-lj.si/2219/>
- [17] (2014) e-računalništvo, dostopno na:
<http://www.s-sers.mb.edus.si/>
<http://goo.gl/Pj1Ht3>
- [18] (2014) w3schools dostopno na:
<http://www.w3schools.com/>
- [19] (2014) w3techs, dostopno na:
http://w3techs.com/technologies/overview/web_server/all
- [20] GNU General Public Licence, dostopno na:
<https://www.gnu.org/copyleft/gpl.html>