

Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

VISOKOŠOLSKI STROKOVNI ŠTUDIJ
Delovna praksa
ZAKLJUČNO POROČILO

CGS plus d.o.o.

Ime in priimek študenta: Denis Kotnik

Vpisna številka: 63100078

Čas izvajanja delovne prakse: 17. februar 2014 – 20. april 2014

Mentor na fakulteti: izr. prof. dr. Janez Demšar

Mentor v podjetju/ustanovi: dr. Rok Kršmanc

Datum: _____

Podpis študenta:

Podpis mentorja v podjetju/ustanovi:

Kazalo

1	Uvod	1
2	Glavni del	2
2.1	Opis podjetja oz. ustanove	2
2.2	Opis delovne prakse	3
2.2.1	Projekt METRoSTAT	3
2.2.2	Terensko delo	5
2.2.3	Programiranje NanosG20 računalnika	6
2.2.4	Cestno vremenski informacijski sistem	8
3	Sklep	10

Slike

1	Logotip podjetja.	2
2	Odrta omarica cestno vremenske postaje s komponentami.	5
3	Menjava merilnika, vgrajenega v cesto.	6
4	Merilnika Clima in Parsivel proizvajalcev Thies Clima in OTT.	6
5	NanosG20 računalnik s, priključenim merilnikom temperature in modemom.	7
6	Kartografska karta z merilnimi postajami v CVIS.	9
7	Uporabniški vmesnik z meritvami določene merilne postaje v CVIS.	9

1 Uvod

Podjetje CGS plus d.o.o. je vodilni slovenski ponudnik rešitev na področju računalniško podprtega načrtovanja (CAD) za arhitekturo, gradbeništvo, strojništvo in geodezijo.

Poleg vodstva, je podjetje razdeljeno na 8 oddelkov, med katerimi je tudi Oddelek okolje, kjer sem v sklopu visokošolskega študija Fakultete za računalništvo in informatiko pod mentorstvom izr. prof. dr. Janeza Demšarja opravljal strokovno prakso. V omenjenem oddelku mi je mentor v podjetju Rok Kršmanc zaupal delo na raziskovalnem projektu METRoSTAT, poleg tega pa sem v času opravljanja prakse postal skrbnik cestno vremenskega informacijskega sistema, katerega vlogo bom opravljal do nadaljnjega.

V nadaljevanju bom podrobneje predstavil podjetje, znanja ki sem jih osvojil ter znanja ki bi jih potreboval, na koncu pa podal še osebno mnenje o svojem opravljanju prakse v podjetju ter svojem visokošolskem študiju računalništva in informatike.

2 Glavni del

2.1 Opis podjetja oz. ustanove

Sedež podjetja CGS plus d.o.o., katerega polno ime je CGS plus, Inovativne informacijske in okoljske tehnologije, se nahaja na Brnčičevi ulici 13 v Ljubljani.

Poleg prej omenjenega področja, računalniško podprtega načrtovanja (CAD), se podjetje ukvarja tudi s partnerskimi pristopi podjetjem, s katerimi jim omogočajo doseganje boljših poslovnih rezultatov. Pomaga jim pri integraciji novih tehnologij, optimizaciji delovnih procesov, izobraževanju kadrov in skrbi za brezhibno delovanje njihove tehnične opreme.

Poleg zastopstev za številne svetovno znane ponudnike CAD programske opreme pa ima podjetje tudi več kot 20 let izkušenj na področju razvoja programske opreme. Ima status razvojnega partnerja za Autodesk in BricsCAD. CGS plus specializirane rešitve za načrtovanje arhitekturnih in infrastrukturnih objektov s pomočjo mreže tujih partnerjev uspešno trži po celem svetu.

Podjetje je dejavno na področju prostorskih in okoljskih informacijskih tehnologij, kjer uspešno združuje interdisciplinarno znanje s področja informatike, strojništva, arhitekture, gradbeništva in naravoslovnih ved. Razvija inteligentne informacijske sisteme, ki služijo vzdrževanju prometne infrastrukture, urejanju prometa, v energetiki in telekomunikacijah, zaščiti in reševanju ob naravnih nesrečah, za hidrološko opozarjanje, dejavnostim v kmetijstvu, gozdarstvu, turizmu in še v mnogih drugih panogah.



Slika 1: Logotip podjetja.

Registrirano je tudi kot raziskovalna organizacija. Leta 2010 je na 5. Slovenskem forumu inovacij prejelo prvo nagrado v kategoriji storitvenih inovacij za algoritem "Best fit".

V letu 2012 je bil podjetju CGS plus d.o.o. prvič podeljeno priznanje za odlično poslovanje »Excellent SME Slovenia«. To je certifikat, ki ga Gospodarska zbornica Slovenije v sodelovanju z uveljavljeno bonitetno hišo Coface Slovenija izdaja najboljšim slovenskim srednjevelikim, malim in mikro podjetjem.

2.2 Opis delovne prakse

2.2.1 Projekt METRoSTAT

Zjutraj, na prvi dan opravljanja strokovne prakse 17. februarja 2014, me je mentor v podjetju Rok Kršmanc prijazno sprejel, razkazal prostore v podjetju in seznanil z direktorjem ter ostalimi zaposlenimi.

V novem delovnem okolju oz. pisarni mi je Rok razložil infrastrukturo podjetja, nato pa prišel z razlago plana oz. nalog, katere je imel v načrtu za moj sklop opravljanja prakse. Na delovnem računalniku, kateri je bil od tega dne rezerviran zame, me je seznanil s programsko opremo ter orodji, katere sem potreboval kasneje.

Sledil je nekaj urni opis projekta METRoSTAT oz. razlaga problema, iz česar sem ugotovil, da bom potreboval znanje iz programiranja v programskem jeziku Python, programskem okolju R ter znanje iz umetne inteligence. Cilj projekta je bilo s pomočjo umetne inteligence oz. uporabe nevronske mreže izboljšati izhodne podatke modela METRo, kateri kot vhodne podatke uporablja podatke modela METRo.

ALADIN je numerični meteorološki model za računanje bodočega stanja ozračja nad omejenim geografskim območjem za vsakodnevno operativno izvajanje.

Osnova modela ALADIN so fizikalne enačbe, ki opisujejo obnašanje nestisljivega zraka v vlažnem ozračju na vrteči se Zemlji: Navier-Stokesova enačba, kontinuitetna enačba in energijska enačba. Iz njih dobimo sistem diferencialnih enačb, ki nima analitične rešitve. Lahko ga rešujemo le s pomočjo numeričnih metod. Enačbe so zato prilagojene takemu reševanju. V ta namen ozračje nad območjem, kjer z modelom simuliramo bodoči razvoj vremena, razdelimo na posamezne prostorninske dele, ki so v horizontalni smeri običajno enakomerno razporejeni, v vertikalni smeri pa je računska mreža zgoščena blizu tal. Velikost ali dolžina posameznega prostorninskega dela v horizontalnih smereh predstavlja horizontalno ločljivost višine vertikalnih slojev pa vertikalno ločljivost. Na tak način razdelimo ozračje na diskretne prostorninske dele, ki predstavljajo računske točke, v katerih numerično rešujemo sistem diferencialnih enačb. Tako dobimo tridimenzionalno mrežo računskih točk. V vsaki mrežni točki računamo hitrost gibanja zraka (veter), temperaturo, vlago, pritisk in druge količine, kot so vsebnost oblačnih in padavinskih delcev, itd. S tipično horizontalno ločljivostjo modelske mreže, ki je danes v modelih za omejeno območje od nekaj kilometrov do nekaj deset kilometrov, lahko ustrezno simuliramo razvoj sinoptičnih meteoroloških procesov (cikloni, anticikloni, fronte) in meteoroloških procesov v mezoskali (organizirani konvektivni sistemi, gorska in priobalna cirkulacija). Od horizontalne ločljivosti je odvisna tudi izbira hidrostatične ali nehidrostatične verzije enačb. Pri hidrostatični verziji je zračni tlak v neki točki odvisen le od teže zraka nad njo in so vertikalne hitrosti mnogo manjše od horizontalnih. Pri nehidrostatični verziji pa je tudi vertikalna hitrost zraka modelska prognostična spremenljivka.

Model je precej manj enovit in konstanten kot se to morda zdi povprečnemu neodvisnemu uporabniku. Glede na izbire določenih parametrov obstaja več verzij modela (trenutno se v evropskem prostoru operativno uporabljajo vsaj tri), hkrati pa se model neprestano nadgrajuje z manjšimi ali večjimi spremembami v opisu določenih fizikalnih pojavov ali v spremembah nekaterih računskih algoritmov. Vsako leto izide približno ena večja verzija programske kode modela (imenovana tudi "cikel") in dve ali tri manj obsežne podverzije. [1]

V domeni projekta METRoSTAT je bilo zajemanje ter priprava podatkov za uporabo na nevronskih mrežah, na koncu pa prikazovanje ter analiza le-teh v grafični obliki.

Razumevanje problema je bilo z moje strani sprva zelo slabo, saj sem šele po nekaj dneh preučevanja programske kode ter razlag problema začel pravilno razmišljati o le tem oz. o uporabljenih podatkih ter parametrih.

Prvi del projekta je bilo zbiranje oz. priprava podatkov, za kar je skrbela programska koda v Pythonu. Le ta je iz treh datotek, katere so bile v obliki CSV oz. *comma-separated values*, izbrala meritve istih časovnih oznak oz. meritve, ki so bile merjene ob istem času. V prvi datoteki so se nahajali podatki, ki so predstavljali izmerjene vrednosti okvirno 15-ih parametrov specifične cestno vremenske postaje. V drugi oz. tretji datoteki pa so podatki z istimi parametri predstavljali napovedane oz. modelirane vrednosti CVP (cestno vremenskih postaj) modela METRo.

Po pripravi podatkov v obliko za uporabo le teh na nevronskih mrežah, je bilo potrebno izpeljati formulo, primerno za našo problemsko domeno, katero sem uporabil oz. kot parameter podal programski funkciji v okolju R, ki kot rezultat vrne nevronske mreže, naučeno nad učnimi podatki. Učni podatki so predstavljali 80% vseh podatkov, ki sem jih v prejšnjem koraku pripravil s skripto v Python programskem jeziku. Podatkom sem dodal še parameter *napaka*, ki je pomenil razliko med modelirano vrednostjo modela METRo za nekaj ur vnaprej in dejansko izmerjeno vrednostjo temperature ceste za isto uro. Omenjeni parameter so vsebovali le učni podatki, na testnih pa ga je poskušala izračunati nevronska mreža s pomočjo naslednje formule:

$$\text{napaka} \sim \text{modelirana temperatura ceste} + \text{izmerjena temperatura ceste} + \text{izmerjena temperatura zraka} + \text{količina sončnega sevanja}$$

Zgornja formula tako v R okolju, kot tudi v mnogih drugih programskih okoljih in jezikih, simbolično opisuje da je parameter na levi strani odvisen od parametrov na desni strani kače oz. znaka \sim .

Ko sem z omenjeno metodo pridobil željeni parameter *napaka* tudi za testne podatke, sem lahko njegovo vrednost odštel od modelirane temperature ceste numeričnega modela METRo in jo tako izboljšal oz. popravil.

Ker so uporabljeni algoritmi za izdelavo nevronskih mrež v uporabljeni programski knjižnici nedeterministični, sem poskuse v programski kodi ponovil večkrat, izračunal standardni odklon rezultatov in varianco rezultatov ter jih prikazal na grafu.

Ker sem za izračun napake imel na voljo več datotek, vsako s pripravljenimi podatki za specifično cestno vremensko postajo ter specifično uro, za katero je model ALADIN modeliral vrednosti, je bilo v sklopu problemske domene potrebno odgovoriti na naslednja vprašanja:

- lahko eno nevronske mreže naučimo s podatki vseh CVP ter dobimo zadovoljive rezultate ali je potrebno za vsako CVP izdelati svojo nevronske mreže?
- lahko eno nevronske mreže naučimo s podatki vseh ur, za katere je model ALADIN modeliral vrednosti ali je potrebno za vsako specifično uro svoja nevronske mreže?
- kako sistemsko požrešna so računanja oz. izdelave nevronskih mrež, za posamezne primere navedene v zgornjih dveh vprašanjih?

- kako ostali parametri, ki jih poleg učnih podatkov še sprejme funkcija za izdelavo nevronskih mrež, vplivajo na rezultate in sistemsko požrešnost?

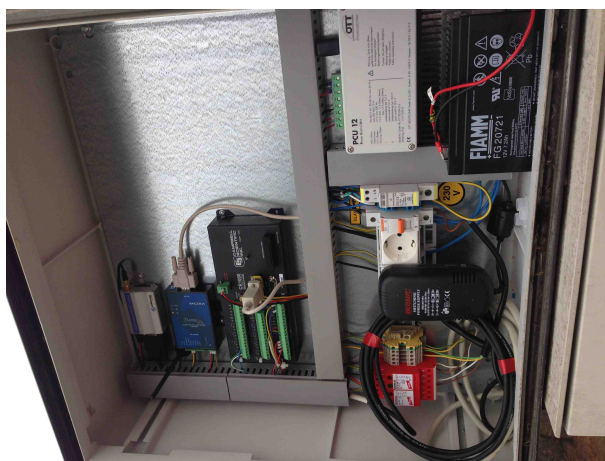
Po večtedenski raziskavi sem iz rezultatov sistemske zahtevnosti ter meritev prišel do sklepa, da je iz podatkov za vsako cestno vremensko postajo in vsako specifično uro najboljše izdelati oz. naučiti svojo nevronska mrežo.

Skupaj z ostalimi rezultati oz. ugotovitvami sem grafe, izrisane v programskem okolju R iz zgoraj omenjenih primerov, uporabil za izdelavo nekaj 10-stranskega poročila v angleščini, katerega sva skupaj z mentorjem poslala na slovaški hidrometeorološki inštitut, ki v projektu sodeluje kot partner.

2.2.2 Terensko delo

Kot je omenjeno v uvodnem delu, se v podjetju CGS plus ukvarjajo tudi z vzdrževanjem cestno vremenskih postaj. Nekajkrat sem tako sam ali pa z mentorjem Rokom odšel na teren, kjer sem opravil manjše servisiranje omenjenih postaj. Največkrat je bil potreben samo ponovni zagon ene ali večih izmed naslednjih komponent postaje:

- napajalni akumulator
- modem, za pošiljanje podatkov na strežnik
- RS232 / RS485 pretvornik
- merilnik za merjenje parametrov ceste
- kompaktna vremenska postaja Clima Sensor proizvajalca Thies Clima
- merilnik za merjenje ostalih vremenskih parametrov
- računalnik za sprejemanje podatkov iz merilnikov

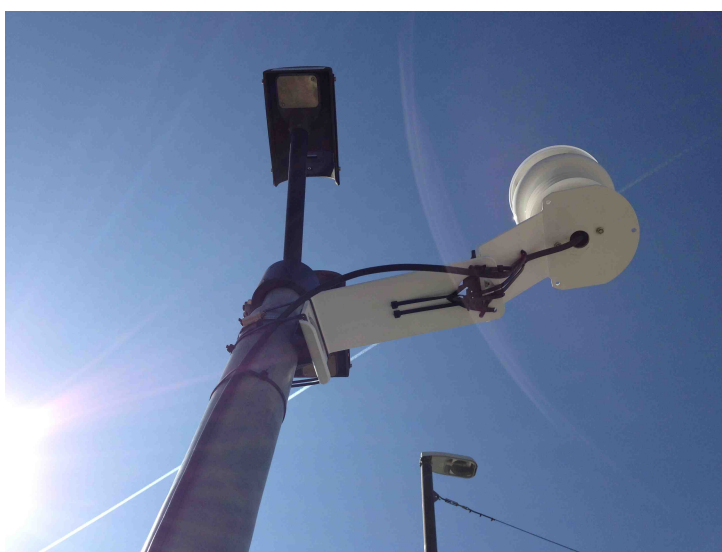


Slika 2: Odprta omarica cestno vremenske postaje s komponentami.

Ko ponovni zagon ni rešil težave, je bila potrebna zamenjava specifične komponente. Lastnoročno sem zamenjal merilnika za merjenje parametrov na cesti oz. vetrovnih parametrov, pri čemer mi je malo pomagalo tudi srednješolsko znanje iz elektrotehnike.



Slika 3: Menjava merilnika, vgrajenega v cesto.



Slika 4: Merilnika Clima in Parsivel proizvajalcev Thies Clima in OTT.

2.2.3 Programiranje NanosG20 računalnika

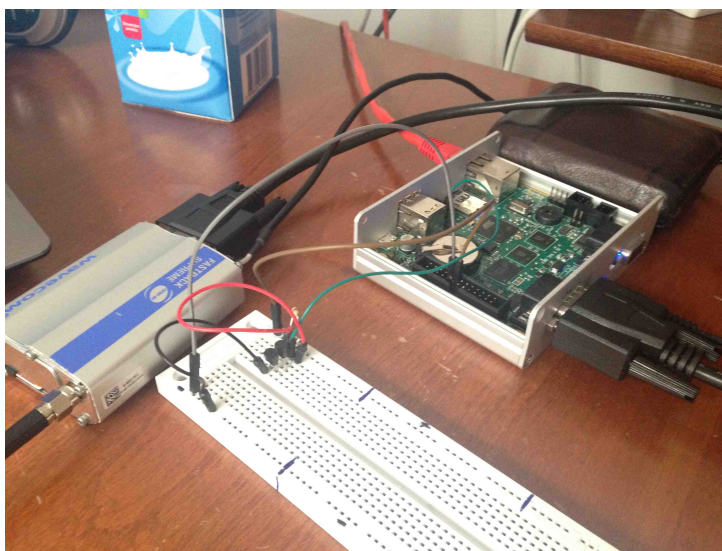
Ker je računalnik v prej omenjeni cestno vremenski postaji draga komponenta, so se v podjetju CGS plus usmerili tudi v razvoj alternativne oz. cenejše rešitve, v svojo programsko opremo na NanosG20 mini računalniku.

NanosG20 je računalnik s 400 MHz procesorjem, 512 MB flash in 128 MB SDRAM pomnilnika ter ethernet, RS232, RS485 priključkom ter režo za spominsko kartico, na kateri naj bi bil nameščen Linux operacijski sistem. Glavna prednost računalnika je, da porabi manj kot 1 W električne energije ter široka podpora programskim rešitvam oz. programskim jezikom v Unix

okolju.

Z razvojem cenejše rešitve na omenjenem računalniku je lansko leto v sklopu prakse pričel že drug študent. Tako sem imel že vzpostavljeno delovno okolje oz. nameščene vse potrebne programske pakete. Na računalnik sem dostopal preko protokola *Secure Socket Shell* oz. SSH, ki je podprt v odprtokodnem programu Putty. Pričel sem s fizično povezavo cenениh temperaturnih senzorjev z Nanos računalnikom po protokolu *I2C*.

Najprej sem podatek v obliki tekstovnega niza oz. *stringa* z implementacijo rešitve v C programskem jeziku prejel iz senzorja, nato pa preko modema, ki je bil z Nanos-om povezan preko RS232 vmesnika, podatek poslal v podatkovno bazo na strežniku v podjetju. Pri tem sem si veliko pomagal z znanjem, ki sem ga pridobil pri fakultetnem predmetu sistemska programska oprema.



Slika 5: NanosG20 računalnik s, priključenim merilnikom temperature in modemom.

V Linux okolju sem najprej nastavljal lastnosti RS232 vmesnika, nato pa le tega z lastnim programom omogočil ter šele nato pričel s pošiljanjem podatkov. Med drugim sem uporabil naslednji skelet programske kode v C prog. jeziku, katerega sem dopolnil oz. prilagodil nastavitvam modema:

```
#include <fcntl.h>
#include <termios.h>
#include <linux/ioctl.h>
#include <linux/types.h>
#include <linux/serial.h>
```

```
void rs485_mode(int on)
{
    struct serial_rs485 rs485conf;
```

```

        if (on)
        {
            rs485conf.flags |= SER_RS485_ENABLED;
            rs485conf.delay_rts_before_send = 0x00000004;
            ioctl(fd, TIOCSRS485, &rs485conf);
        }
        else
        {
            rs485conf.flags &= ~SER_RS485_ENABLED;
            ioctl(fd, TIOCSRS485, &rs485conf);
        }
    }
}

void rs485_receiver(int on)
{
    struct termios options;

    tcgetattr(fd, &options);
    if (on)
        options.c_cflag |= CREAD;
    else
        options.c_cflag &= ~CREAD;
    tcsetattr(fd, TCSANOW, &options);
    fcntl(fd, F_SETFL, FNDELAY);
}

```

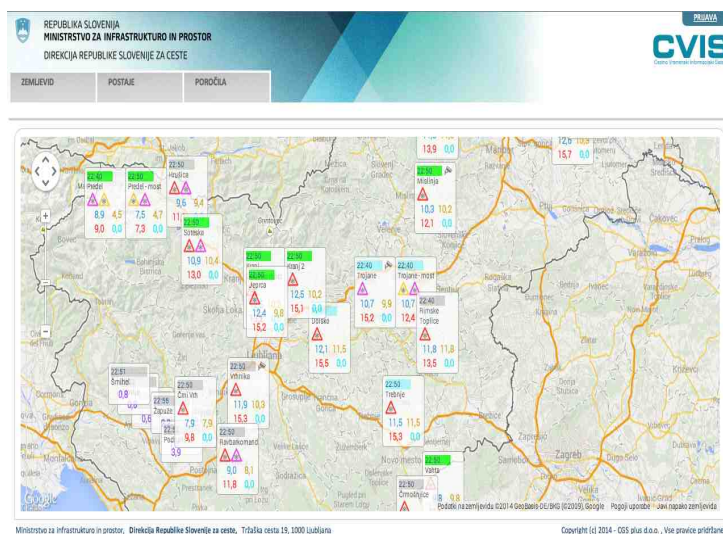
Po uspešnem zapisu izmerjene temperature v bazo na strežniku sem isto storil še s podatki iz kompaktne vremenske postaje proizvajalca Thies Clima, za kar sem uporabil podobno metodo oz. programsko kodo kot za RS232 vmesnik.

2.2.4 Cestno vremenski informacijski sistem

Cestno vremenski informacijski sistem (CVIS) je informacijski sistem, razvit v sodelovanju oz. za potrebe Družbe za avtoceste Republike Slovenije (DARS) in Direkcije Republike Slovenije za ceste (DRSC). Slednji omogoča javno uporabo omenjenega sistema oz. pregled osnovnih podatkov na povezavi <http://drsc.cgsplus.si>.

CVIS skrbi za prenos podatkov oz. parametrov iz cestno vremenskih postaj, ki se nahajajo povsod po Sloveniji, v podatkovno bazo. Celoten sistem, katerega sestavljajo omenjene podatkovne baze, Windows storitve za prenos podatkov ter spletna aplikacija za grafični prikaz podatkov, je skupek več tisoč vrstic programske kode v Microsoftovi .NET tehnologiji.

Sistem je porazdeljen med 15 računalniki v več omrežjih. Za dostop do teh računalnikov je potrebno z avtentifikacijsko kartico vzpostaviti povezavo z VPN oz. virtualno zasebno omrežje DARS, ter se s programom *Windows Remote Desktop* povezati na željeni računalnik.



Slika 6: Kartografska karta z merilnimi postajami v CVIS.

Razvil ga je Andrej Beden, dipl. inž. gradb., kateri mi je pomagal pri razumevanju delovanja sistema in s strani katerega sem dobil tudi prvo zahtevo za novo funkcionalnost. Ta je zahtevala implementacijo nove storitve v programu *Microsoft Visual Studio*, katero sem kasneje na DARS-ovem strežniku namestil z orodjem *InstallUtil*.

V zadnjih dveh tednih opravljanja prakse sem imel precej sestankov z Andrejem Bedenom, katerih namen je bil predvsem predaja skrbništva nad CVIS. V okviru teh sem spoznal tudi nekaj drugih ljudi, s katerim bova z Rokom Kršmancem do nadaljnjega ostala v kontaktu.



Slika 7: Uporabniški vmesnik z meritvami določene merilne postaje v CVIS.

3 Sklep

Če bi moral opravljanje strokovne prakse v podjetju CGS plus subjektivno primerjati s predmeti na fakulteti, bi bil to eden boljših predmetov, saj so mi tako mentor kot tudi ostali zaposleni ponudili veliko novega znanja, izkušenj ter idej.

Spoznal sem veliko prijaznih ljudi, predvsem pa to, kako skupaj delujejo kot podjetje na zelo produktiven ter zaposlenemu prijazen način. Udeležil sem se tudi konference *Danube region strategy* ter predavanja o mehkih veščinah, za katere že iz preteklih izkušenj menim, da so v poslovnem življenju še posebej pomembne.

Kot sem omenil, sem odnesel tudi ogromno novega tehničnega znanja, predvsem iz Microsoftovega orodja *Visual Studio* ter s področja umetne inteligence.

Večino časa sem preživel v pisarni skupaj z mentorjem Rokom Kršmancem, s katerim nama res nikoli ni manjkalo novih tem za pogovor. Med njimi so bile tudi računalniške teme, kar mi je poleg projekta METRoSTAT dalo navdih za izbiro teme diplomske naloge v povezavi z umetno inteligenco.

Med opravljanjem prakse sem spoznal oz. videl, da sem na fakulteti pridobil veliko uporabnega znanja, ki sem ga lahko izkoristil pri skoraj vsaki stvari, ki je imela povezavo z računalništvom. Ob študiju sem pridobil mnenje, da *več kot veš, bolje veš, da veliko ne veš*, ter sposobnost, da se lahko vsaj osnov znanja, katerega nimam, naučim tudi sam, v kar pa sem se prepričal ob opravljanju strokovne prakse v podjetju CGS plus d.o.o..

Literatura

- [1] Neva Pristov, Jure Cedilnik, Jurij Jerman, and Benedikt Strajnar. 20 let modela ALADIN. *Slovensko meteorološko društvo*, 2001. Dostopno na: http://www.meteo-drustvo.si/data/upload/Vetrnica0412_Pod_drobnogledom.pdf.