

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Matej Konobelj

**Možnosti sinhronizacije v
heterogenem poslovnem okolju**

DIPLOMSKO DELO

VISOKOŠOLSKI STROKOVNI ŠTUDIJSKI PROGRAM PRVE
STOPNJE RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: doc. dr. Mojca Ciglarič

Ljubljana 2012

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

Besedilo je oblikovano z urejevalnikom besedil \LaTeX .



Št. naloge: 00220/2012

Datum: 03.04.2012

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Kandidat: **MATEJ KNOBELJ**

Naslov: **MOŽNOSTI SINHRONIZACIJE V HETEROGENEM POSLOVNEM
OKOLJU**
**SYNCHRONIZATION OPTIONS IN HETEROGENEOUS BUSINESS
ENVIRONMENT**

Vrsta naloge: Diplomsko delo visokošolskega strokovnega študija prve stopnje

Tematika naloge:

Preučite protokole in orodja, ki se danes uporabljajo za sinhronizacijo podatkov različnih aplikacij, zlasti koledarja, z mobilnimi ali drugimi napravami. Opišite problematiko, ki jo v zvezi s sinhronizacijo srečuje izbrano podjetje. Opišite tehnološko heterogeno okolje, v katerem mora delovati sinhronizacija. Postavite ustrezno testno okolje in v njem preizkusite izbrane metode sinhronizacije. Primerjajte jih glede na način administracije, uporabnost spletnega vmesnika in glede na funkcionalnost za odjemalca. Utemeljite izbiro, izbrani protokol namestite v produkcijskem okolju, opišite svoje izkušnje in predlagajte možnosti za nadaljnje delo.

Mentor:

M. Ciglarič

doc. dr. Mojca Ciglarič



Dekan:

Z

prof. dr. Nikolaj Zimic

IZJAVA O AVTORSTVU DIPLOMSKEGA DELA

Spodaj podpisani Matej Konobelj, z vpisno številko **63040231**, sem avtor diplomskega dela z naslovom:

Možnosti sinhronizacije v heterogenem poslovnem okolju

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvom doc. dr. Mojce Ciglarič,
- 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 v zbirki "Dela FRI".

V Ljubljani, dne 12. oktobra 2012

Podpis avtorja:

Zahvaljujem se mentorici doc. dr. Mojci Ciglarič za pomoč pri izdelavi diplomskega dela. Zahvale gredo tudi vsem zaposlenim na Arnesu, ki so mi omogočili izdelavo diplomskega dela. Zahvalil bi se tudi družini in prijateljem, ki so mi vedno stali ob strani. Posebna zahvala gre Teji, ki je poskrbela za manjkajoče vejice, pike in me spodbujala ter motivirala pri izdelavi tega dela. Hvala vsem.

Noni v spomin.

Kazalo

Povzetek

Abstract

1	Uvod	1
2	Potreba po sinhronizaciji	3
2.1	Zahteve podjetja	3
2.2	Možnost vključitve odjemalca v obstoječi spletni vmesnik . . .	4
3	Pregled sinhronizacijskih protokolov	7
3.1	ActiveSync	7
3.2	SyncML	8
3.2.1	Funambol	8
3.3	CalDAV	9
3.3.1	DAViCal	11
3.3.2	Radicale	12
4	Primerjava orodij in protokolov	13
4.1	Strežnik LAMP in podporni servisi	13
4.1.1	Linux	13
4.1.2	Apache	14
4.1.3	Podatkovne baze	14
4.1.4	PHP	15
4.2	DAViCal	17

KAZALO

4.3	Radicale	19
4.4	AgenDAV	20
4.5	Roundcube vtičnik	24
5	Testiranje v realnem okolju	27
5.1	Načrt testiranja	27
5.2	Administracija uporabnikov	28
5.3	Spletna vmesnika AgenDAV in Roundcube	29
5.4	Odjemalci	31
5.5	Rezultati testiranja	31
6	Sklepne ugotovitve	33
	Seznam slik	34
	Literatura	37

Povzetek

Ob povečani uporabi pametnih mobilnih naprav v zadnjih letih lahko opazamo, da v veliki meri že nadomeščajo konvencionalne metode organiziranja in obveščanja. Težava pa se pojavi, ko želimo uporabljati več kot eno napravo, ki nam omogoča smotrno organizacijo našega časa. Zato je na tem področju potreba po sinhronizaciji teh naprav velika, pa naj gre za aplikacijo, ki jo imamo nameščeno na računalniku in jo uporabljamo na delovnem mestu, ali pa aplikacijo, ki je nameščena na mobilni napravi in jo imamo vedno pri roki. V nalogi smo tako primerjali protokole, ki se najpogosteje uporabljajo pri sinhronizaciji dogodkov, shranjenih na oddaljenem koledarskem strežniku. Na podlagi prvotnih primerjav smo izbrali protokol, ki je zadostil našim zahtevam. V nalogi smo tudi prikazali implementacijo odprtokodne rešitve, ki uporablja izbran protokol za sinhronizacijo podatkov, in preskusili delovanje rešitve v realnem okolju.

Ključne besede:

Koledarski strežnik, CalDAV, DAViCAL, AgenDAV, sinhronizacija, SyncML, Radicale, WebDAV

Abstract

With increased use of smart mobile devices in recent years, we can note that the latter have already largely replaced conventional methods of organization and communication. The problem occurs when we want to use more than one device, which allows us to efficiently organize our time. Therefore, in this field the need for synchronization of these devices is great, whether we are talking about an application installed on a local computer and used at workplace or an application that is installed on a mobile device and is always within our reach. In this thesis we also compare the protocols most commonly used in the synchronization of events located on a remote calendar server. Based on initial comparisons we chose a protocol, which fulfilled our requirements. In the paper we also show the implementation of an open source solution that uses the selected protocol for data synchronization, and operation of the solution tested in a real environment.

Key words:

Calendar server, CalDAV, DAViCAL, AgenDAV, synchronization, SyncML, Radicale, WebDAV

Poglavje 1

Uvod

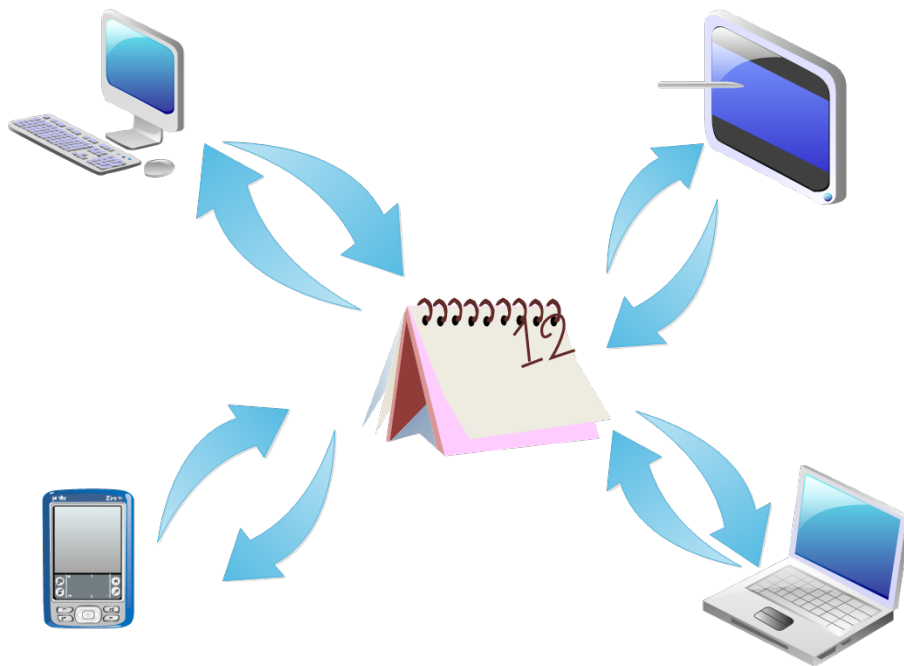
V zadnjih letih opažamo porast uporabe elektronskih mobilnih naprav, ki nam omogočajo boljše in kvalitetnejše izkoriščanje ter organizacijo našega časa. Večina ljudi v sodobnem svetu dnevno uporablja vsaj eno mobilno napravo, kot je na primer telefon ali tablica. Telefoni so bili včasih namenjeni samo klicanju, sedaj pa se spreminjajo v tako imenovane pametne naprave. Sam prenos govora tako ni več prvotnega pomena.

Večina nas uporablja različne mobilne naprave za brskanje po spletu, hrambo kontaktov, pošiljanje sporočil, uporabo socialnih omrežij. Ker so te naprave zaradi priročnosti in modernega načina življenja vedno z nami, smo jih začeli uporabljati tudi za bolj učinkovito razporejanje našega časa. Na mestu, kjer so bile včasih beležnice, koledarji in svinčniki, so sedaj aplikacije, ki jih imamo uporabniki nameščene na svojih napravah, pa naj bo to telefon, tablica ali osebni računalnik.

In prav tu se pojavi težava. V kolikor uporabljamo več kot eno elektronsko napravo na različnih lokacijah z nameščenimi različnimi koledarskimi aplikacijami, moramo zagotoviti zanesljivo sinhronizacijo le-teh.

V diplomskem delu bomo predstavili možne rešitve težave sinhronizacije med različnimi aplikacijami za organizacijo posameznikovega časa. Zahteve, ki jih je postavilo podjetje, so odprtost rešitve, povezljivost z namiznimi aplikacijami, povezljivost z mobilnimi aplikacijam ter samostojen sple-

tni vmesnik. Raziskali pa bomo tudi možnost vključitve servisa v obstoječo ponudbo podjetja, bolj natančno, raziskali bomo možnost integracije rešitve v vmesnik za spletno pošto Roundcube.



Slika 1.1: Pogled s strani uporabnika

Poglavje 2

Potreba po sinhronizaciji

Za začetek vzemimo v razmislek, koliko časa zaposleni v večjih ustanovah izgubijo za samo prizadevanje, da se skliče seja, sestanek ali organizira kakšna druga aktivnost med zaposlenimi. Trenutno to vključuje telefonske klice, nekaj elektronske pošte in veliko usklajevanja. Običajno je za vsak posamezen sestanek odgovorna ena oseba, ki mora poskrbeti, da se sestanka udeleži čim več povabljenih. Obdelati pa mora tudi primere, ko čas sestanka ali srečanja ne dopušča udeležitve enega ali večih udeležencev.

Zato smo se v našem podjetju odločili vpeljati sistem, ki bi zaposlenim omogočal uporabo enotnega strežnika za hrambo in izmenjavo koledarjev. Nakup licence za naš primer ne pride v poštev. Zato je bilo potrebno raziskati možnosti, ki nam jih ponujajo odprtokodne rešitve. V ožji izbor so tako prišli trije bolj znani načini sinhronizacije strežnika z mobilnimi in namiznimi aplikacijami.

2.1 Zahteve podjetja

Zahteve, ki jih je podjetje postavilo, so bile:

- prostodostopnost rešitve,
- odprtokodnost rešitve,

- povezljivost rešitve z obstoječimi aplikacijami,
- samostojen spletni vmesnik.

Uporabniku bi radi z rešitvijo omogočili:

- ustvarjanje lastnega koledarja,
- distribucijo svojega koledarja,
- uvoz drugega javnega koledarja,
- sinhronizacijo koledarja med različnimi napravami.

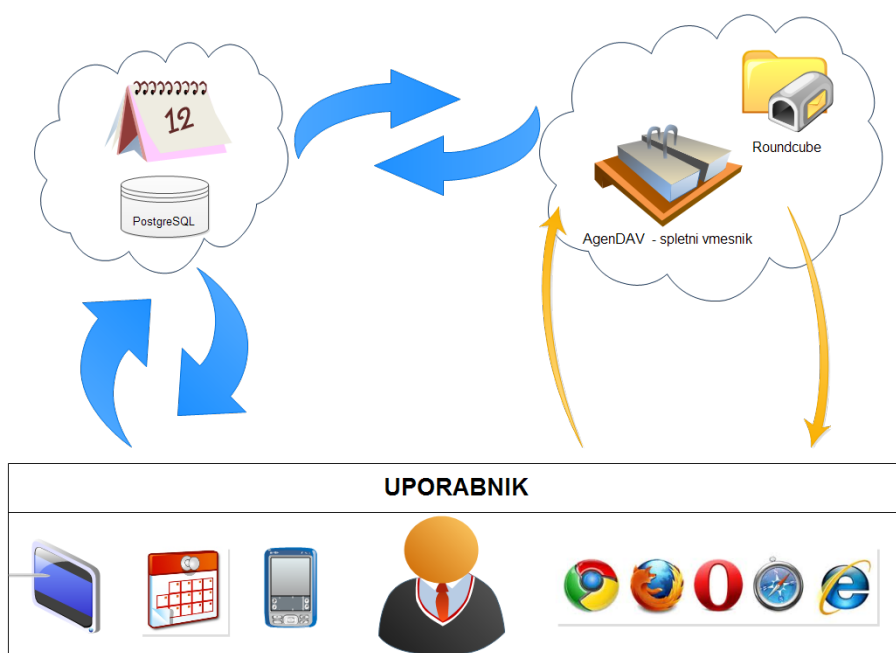
Glavno vodilo naloge je bilo, da s čim manjšimi stroški in čim nižjo porabo resursov zagotovimo čim večjo interoperabilnost. To je zmožnost sistema ali izdelka, da sodeluje z drugimi sistemi ali izdelki brez posebnega truda uporabnika storitve. Interoperabilnost je zelo pomembna kakovost za izdelke informacijske tehnologije, ko zamisel "omrežje je računalnik" postane resničnost.

2.2 Možnost vključitve odjemalca v obstoječi spletni vmesnik

Podjetje ima kot internetni ponudnik tudi namen razširjanja obstoječe ponudbe, ki je na voljo našim uporabnikom. V podjetju uporabljamo spletni vmesnik Roundcube, ki našim uporabnikom omogoča pregledovanje svoje elektronske pošte prek spleta. Vmesnik Roundcube omogoča tudi dodajanje vtičnikov, ki jih lahko uporabimo za razširitev funkcionalnosti programa. Nekaj funkcionalnosti vmesnika Roundcube že sedaj zagotavljamo z uporabo vtičnikov.

Za našo nalogo je bistvenega pomena, da dosežemo vključitev koledarskega odjemalca v vmesnik Roundcube. V kolikor ta cilj dosežemo, je namen naloge izpolnjen, saj to pomeni, da bomo lahko iz interne aplikacije razvili in

dodali v produkcijo novo storitev, po kateri naši uporabniki že povprašujejo, predvsem iz razloga nezaupanja ostalim ponudnikom podobnih rešitev, katerih strežniki se v večini primerov ne nahajajo v Sloveniji, temveč nekje izven mej. Tako uporabniki nimajo nikakršnega zagotovila, da se njihova zasebnost ohranja.



Slika 2.1: Zastavljen nivo interoperabilnosti rešitve

Poglavje 3

Pregled sinhronizacijskih protokolov

Danes pogosto uporabljamo svetovni splet samo za brskanje in pridobivanje informacij. Vendar pa to ni bil prvotni namen spleta. Že leta 1990 je bil predstavljen koncept uporabe spleta kot bralno-zapisljiv medij. Čeprav takrat še ni bilo zanimanja za koncept, ideja ni zamrla. V letu 1995 so razvijalci uvideli potrebe po razvoju in razširitvi protokola HTTP (Hyper text transfer protocol). WebDAV je v osnovi razširitev protokola HTTP, ki omogoča sodelovanje med uporabniki pri urejanju dokumentov in datotek, shranjenih na strežnikih, povezanih v svetovni splet. WebDAV nam je tako omogočil uporabo svetovnega spleta kot bralno-zapisljiv medij. Uporabnikom omogoča okvir, v katerem lahko ustvarjajo, spreminjajo in premikajo datoteke na spletnem strežniku. Najpomembnejša lastnost protokola WebDAV je vključitev tako imenovanih metapodatkov. To so informacije o avtorju, datumu spreembe, zbirkah ter bralno-pisalni zaščiti.

3.1 ActiveSync

ActiveSync je Microsoftova programska oprema za sinhronizacijo e-pošte, koledarjev, stikov in ostalih podatkov med namiznimi računalniki in mobilnimi

napravami, kot so žepni računalniki, dlančniki in pametni telefoni. Je tudi poglobljena sinhronizacijska komponenta, ki omogoča sinhronizacijo v sistemu za elektronsko pošto Microsoft Exchange. V tem poglavju jo omenjamo zgolj kot predstavnika zaprtokodne rešitve, ki bi se jim radi v naši nalogi v čim večji meri izognili.

3.2 SyncML

SyncML je kratica za "Synchronization Markup Language", kar pomeni sinhronizacijski označevalni jezik. Je standard za sinhronizacijo podatkov, neodvisen od platforme. Uporablja se za sinhronizacijo podatkov, kot so stiki, sezname, urniki in koledarji, med različnimi napravami (telefoni, dlančniki, računalniki).

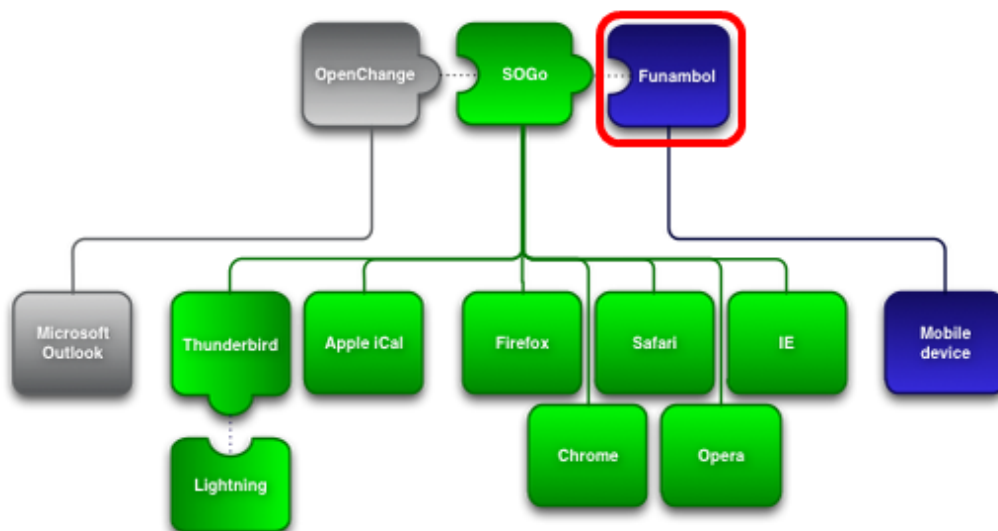
SyncML zagotavlja standardni format, ki temelji na jeziku XML. Sinhronizacija z uporabo standarda SyncML lahko deluje preko različnih vrst povezav, kot so brezžični internet, povezava bluetooth in infrardeča povezava. Protokol SyncML je leta 2000 razvilo združenje Open Mobile Alliance. Je standardiziran, odprtokoden in brezplačen odgovor na protokol ActiveSync.

3.2.1 Funambol

Ena od rešitev, ki smo jo kot možno vzeli pod drobnogled, je bila tudi Funambol. Funambol za sinhronizacijo podatkov med napravami uporablja ravno protokol SyncML. Vendar se je rešitev že v začetku izkazala za ne-perspektivno; čeprav je strežnik, ki omogoča sinhronizacijo, brezplačen in prostodostopen, ni zanj razvite nobene odprtokodne aplikacije, ki bi nam omogočala oblično (frontend) komunikacijo s strežnikom samim.

Vse rešitve okoli Funambol strežnika so v večini primerov zapakirane v celovito rešitev za podjetja, ki morajo za uporabo rešitve plačati licenčnino. Tak primer je plačljiva rešitev SoGo, ki sicer vključuje strežnik Funambol, vendar je uporabljen le kot posredovalno sinhronizacijsko orodje, ki omogoča sinhronizacijo koledarja, kontaktov in ostalih datotek le na mobilne naprave,

kar je lepo razvidno tudi iz spodnje slike.



Slika 3.1: Primer integracije Funambol strežnika v SOGo

3.3 CalDAV

CalDAV je oznaka za Calendaring Document Authoring and Versioning in pomeni protokol oziroma jezik, s katerim se različne naprave dogovorijo o podatkih v koledarju. Prve specifikacije CalDAV so se pojavile v letu 2003 kot internetni osnutek, ki je bil posredovan IETF (Internet Engineering Task Force).

CalDAV je bil nato zasnovan kot orodje, ki bi omogočilo sodelovanje med programsko opremo različnih razvijalcev, pa naj bo to odjemalec ali strežnik, ki mora vzdrževati ažurno stanje s sinhronizacijo med lokalno aplikacijo in oddaljenim strežnikom, do katerega lahko dostopa več uporabnikov z uporabo različnih naprav in orodij. CalDAV temelji na WebDAVu in ga uporablja kot transportni mehanizem. Protokol definira zbirka RFC 4791. CalDAV protokol nam tako omogoča dostop različnih odjemalcev do enotnega strežnika.

Že ob razvoju je CalDAV posvojil računalniški format iCalendar (".ics");

”.ical”; “.ifb”; “.icalendar”), ki močno temelji na starejšem formatu “.vCalendar” in dovoljuje uporabnikom pošiljanje zahtev in nalog drugim uporabnikom z uporabo spletne pošte ali koledarske aplikacije. Uporabnik, ki je prejel datoteko iCalendar, lahko tako enostavno odgovori pošiljatelju s spremembo v datoteki, ki mu jo nato posreduje nazaj prek elektronske pošte ali koledarske aplikacije.

Podatki v formatu “.ical” so zapisani z uporabo golega besedia (plain text) in vsebujejo podatke o vrsti koledarja, imenu, lokaciji, datumu, ko smo dogodek ustvarili in datumu, ko smo ga zadnjič spremenili. Primer zapisa v iCal datoteki:

```
BEGIN:VCALENDAR
PRODID:-//davical.org//NONSGML AWL Calendar//EN
VERSION:2.0
CALSCALE:GREGORIAN
X-WR-CALNAME:Tretji koledar
BEGIN:VEVENT
UID:DB5123DB-3DF2-491C-9A7A-DDE896B2CC5A
DTSTAMP:20121004T221702Z
CLASS:PUBLIC
CREATED:20121004T221702Z
DTSTART;TZID=Europe/Ljubljana:20121017T003000
DTEND;TZID=Europe/Ljubljana:20121017T013000
LAST-MODIFIED:20121004T221709Z
SEQUENCE:1
SUMMARY:Test dodajanje
TRANSP:OPAQUE
END:VEVENT
BEGIN:VTIMEZONE
TZID:Europe/Ljubljana
X-LIC-LOCATION:Europe/Ljubljana
BEGIN:STANDARD
DTSTART:20121028T030000
TZOFFSETFROM:+0200
TZOFFSETTO:+0100
TZNAME:CET
END:STANDARD
BEGIN:DAYLIGHT
DTSTART:20130331T020000
TZOFFSETFROM:+0100
TZOFFSETTO:+0200
TZNAME:CEST
```

```
END:DAYLIGHT
END:VTIMEZONEF
END:VCALENDAR
```

3.3.1 DAViCal

DAViCal je strežnik, namenjen hrambi in izmenjavi koledarjev med različnimi napravami. Je preprosto vendar zelo zmogljivo orodje za administracijo koledarskih vnosov. DAViCal implementira protokol CalDAV, ki skrbi za pravičen dostop do strežnika. V osnovi je to servis, ki skladišči koledarje in nudi dostop odjemalcem, kot so Evolution, Thunderbird, Apple iCal ali iPhone, ter odjemalcem CalDAV na platformi Android.

Strežniško aplikacijo DAViCal odlikuje dejstvo, da ni potrebe po dodatnem spletnem strežniku, kot je Tomcat, ki se z zahtevno javansko aplikacijo ves čas povezuje s strežniško aplikacijo. Tak primer smo si ogledali pri rešitvi Badework, ki ni zadovoljila potrebe po čim bolj lahkotnem servisu. Zato je tudi izpadla iz nabora možnih rešitev. Za strežnik DAViCal tako zadostujeta že Apache in PHP, ki s svojo lahkotnostjo ne obremenjuje samega fizičnega strežnika in ga je možno namestiti tudi na virtualni strežnik z malo resursi. Strežnik pa nam tudi zagotovi uvoz in izvoz datotek v formatu ".ics", ki zagotavlja varnost v obliki preproste varnostne kopije.

Administracija strežnika DAViCal poteka preko spletnega vmesnika, kjer je mogoče ustvarjati in brisati uporabnike, združevati uporabnike v skupine in določati bralno-pisalne pravice vseh uporabnikov, kakor tudi posameznika. Uporabniki so lahko na strežniku v različnih razmerjih:

- Administrator - daje uporabniku vse pravice nad koledarjem drugega uporabnika, vključno z zasebnimi dogodki.
- Pomočnik (is assisted by) - daje uporabniku bralno-pisalne pravice na koledarju drugega uporabnika.
- Lahko bere (Can read from) - daje uporabniku bralne pravice na koledarju drugega uporabnika.

- Lahko vidi (Can see the availability of) - daje uporabniku vpogled v stanje zasedenosti koledarja drugega uporabnika.

3.3.2 Radicale

Radicale je predvsem koledarski strežnik napisan z uporabo jezika Python. Uporablja poenostavljen protokol CalDAV za oddaljeni dostop za branje, ustvarjanje in spreminjanje koledarjev. Povezavo med aplikacijo in strežnikom lahko tako kot pri strežniku DAViCal šifriramo s protokolom SSL.

Strežnik Radicale smo v naši nalogi vzeli kot možno rešitev, ker strežnik ni zahteven za vzdrževanje in porablja malo resursov.

Poglavje 4

Primerjava orodij in protokolov

V poglavju bomo v testno okolje namestili dva koledarska strežnika in storitve, ki bodo zagotavljale testnim uporabnikom dostop do njiju. Primerjali bomo strežnika DAViCal ter Radicale, ki oba temeljita na sinhronizacijskem protokolu CalDAV. Protokol CalDAV smo izbrali zaradi obilice prostodostopne in odprtokodne programske opreme. Poleg tega nam le ta protokol zagotavlja visok nivo interoperabilnosti med napravami. Poskusili bomo tudi namestiti dva različna spletna vmesnika in ju povezati s koledarskim strežnikom.

4.1 Strežnik LAMP in podporni servisi

Testni strežnik smo pripravili v okviru še ene Arnesove testne storitve Ablak. Storitve Ablak temelji na virtualizacijskem ogrodju Ganeti. Ganeti je programsko orodje za upravljanje z gručami virtualnih strežnikov, razvito s strani Googla. V njem imamo uporabniki pripravljene predloge, izmed katerih ob nameščanju operacijskega sistema eno izberemo.

4.1.1 Linux

Za potrebe postavitve sistema, ki nam bo omogočal hranjenje dveh različnih baz in nekaj različnih servisov, smo izbrali strežniški sistem Ubuntu 10.04.

Izbira operacijskega sistema je temeljila na dejstvu, da ima trenutno Ubuntu med vsemi brezplačnimi sistemi Linux najboljšo podporo za odpravljanje težav.

4.1.2 Apache

Za serviranje aplikacij smo uporabili spletni strežnik Apache. Za namestitvev spletnega strežnika Apache potrebujemo pravice korenskega uporabnika. Namestitev strežnika Apache izvedemo z ukazom:

```
\$ sudo apt-get install apache2
```

4.1.3 Podatkovne baze

Pred namestitvijo ostalih komponent sistema smo morali zagotoviti delovanje dveh neodvisnih podatkovnih baz in določiti vloge uporabnikov teh baz. Strežnik DAViCal potrebuje za delovanje podatkovno bazo PostgreSQL. Za podporo ostalim servisom smo namestili tudi podatkovno bazo MySQL. Preden ustvarimo bazo, moramo urediti zapis v datoteki

```
/etc/postgresql/9.1/main/pg_hba.conf
```

Datoteka `pg_hba.conf` je konfiguracijska datoteka baze PostgreSQL. Zapisi v tej datoteki določajo vrsto povezave, odjemalčev naslovni prostor IP, s katerega lahko dostopa, ime baze, uporabniško ime in avtentikacijsko metodo. S spodnjo dopolnitvijo konfiguracijske datoteke baze omogočimo dostop do baze za uporabnika `davical_dba`, ki bo uveden za lastne potrebe baze in tabel, pa tudi za uporabnika `davical_app`, ki bo ustvarjen za povezavo spletne aplikacije z bazo.

```
local    davical    davical_app    trust
local    davical    davical_dba    trust
```

Spremembe v nastavitvah podatkovne baze uveljavimo z ukazom za ponovni zagon:

```
\$ sudo /etc/init.d/postgresql-9.1 restart
```

Po ponovnem zagonu servisa PostgreSQL moramo še ustvariti uporabnike, ki jih bo strežnik DAViCal potreboval za dostop in napolnitev baze s tabelami, potrebnimi za pravilno delovanje servisa. Za delo s PostgreSQL se najprej prijavimo v sistem kot korenski uporabnik, nato pa lahko izvajamo ukaze nad bazo kot uporabnik "postgres".

```
sudo su
su postgres -c "createuser davical_app"
su postgres -c "createuser davical_dba"
exit
```

Bazo napolnimo s skripto, ki jo je pred zagonom potrebno ročno popraviti, da naslavlja pravilne lokacije datotek na strežniku.

```
\$ su postgres -c /usr/share/davical/dba/create-database.sh
```

4.1.4 PHP

PHP (trenutno tričrkovni rekurzivni akronim za "PHP Hypertext Preprocessor", izvorno pa "Personal Home Page Tools" – orodja za osebno spletno stran) je razširjen odprtokodni programski jezik, ki se uporablja za strežniške uporabe oziroma za razvoj dinamičnih spletnih vsebin. Lahko ga primerjamo z Microsoftovim sistemom ASP/VBScript/JScript, sistemom JSP/Java podjetja Sun Mycosystem in sistemom CGI/Perl.

Na sistemu Ubuntu je namestitev mogoča z uporabo repozitorija z ukazom:

```
\$ sudo apt-get install php5
```

Namestitvi pa moramo dodati tudi knjižnico libapache2-mod-php5. To storimo z ukazom:

```
\$ sudo apt-get install libapache2-mod-php5
```

Po uspešni namestitvi moramo ponovno zagnati spletni strežnik Apache. To storimo z ukazom:

```
\$ sudo service apache2 restart
```

4.2 DAViCal

Namestitev strežnika DAViCal je mogoča preko upravljalca paketov z ukazom "apt-get install davical", pred tem pa moramo v datoteko "/etc/apt/sources.list.d/davical.list" dodati zapis

```
"deb http://debian.mcmillan.net.nz/debian lenny awm" ,
```

ki bo sistemu povedal, kje na spletu se nahaja zadnja verzija programa. Ta zapis v repozitoriju nam bo prišel prav tudi v prihodnje, ko bomo izvajali posodobitve servisa. Ob posodobitvi se napredno paketno orodje (apt - Advanced Packaging Tool) poveže z repozitorijem in prenese novejšo verzijo datotek na strežnik. Po prenosu novih datotek moramo samo še posodobiti povezavo z bazo, kar najlažje storimo prek grafičnega administratorskega vmesnika za administracijo strežnika DAViCal. Za namestitev potrebujemo pravice korenskega uporabnika. Po opravljeni namestitvi moramo strežniku Apache za lažji dostop do grafičnega vmesnika za upravljanje določiti virtualnega gostitelja. To lahko storimo preko orodja Webmin, ki ga imamo nameščenega na sistemu in nam omogoča administracijo strežnika v grafičnem načinu. Lahko pa dodamo spodnji zapis za virtualnega gostitelja v mapo "/etc/apache2/sites-available":

```
<VirtualHost *:443>
  ServerName  davical.domena.si:443
  ServerAdmin "'ime.priimek@organizacija.si"
  ErrorLog   /var/log/apache2/davical_error.log
  LogLevel  warn
  CustomLog  /var/log/apache2/davical_access.log combined
  ServerSignature On
  DocumentRoot /usr/share/davical/htdocs
  DirectoryIndex index.php index.html
  Alias /images/ /usr/share/davical/htdocs/images/
  <Directory /usr/share/davical/htdocs/>
```

```

    AllowOverride None
    Order allow,deny
    Allow from all
    php_admin_flag engine on
</Directory>
php_value include_path /usr/share/awl/inc
php_value magic_quotes_gpc 0
php_value register_globals 0
php_value error_reporting "E_ALL & ~E_NOTICE"
php_value default_charset "utf-8"
<IfModule mod_rewrite.c>
    RewriteEngine On
    RewriteCond %{REQUEST_URI} !^/$
    RewriteCond %{REQUEST_URI} !\.(php|css|js|png|gif|jpg)
    RewriteRule ^(.*)$ /caldav.php$1 [NC,L]
</IfModule>
SSLEngine on
SSLVerifyClient none
SSLCertificateFile /opt/psa/var/certificates/cert-YOURS
SetEnvIf User-Agent ".*MSIE.*" nokeepalive ssl-unclean-shutdown
</VirtualHost>

```

Po dodajanju virtualnega gostitelja moramo uveljaviti nove nastavitve tako, da ustvarimo simbolično povezavo v mapi `/etc/apache2/sites-enabled`, ki kaže na ravnokar ustvarjeno datoteko. To najlažje storimo z ukazom `ln -s`

```

sudo ln -s ../sites-available/mydavicalsite
      ../sites-enabled/mydavicalsite

```

Spremembe moramo uveljaviti s ponovnim zagonom strežnika Apache.

Preden se lahko prijavimo v spletni vmesnik za administracijo uporabnikov, nas čaka še nekaj osnovnih nastavitvev aplikacije. Na namestitveni loka-

ciji programa moramo ustvariti konfiguracijsko datoteko "davical.domena.si-conf.php". Vanjo zapišemo nekaj osnovnih informacij, kot so tehnični kontakt in ime sistema. Najpomembnejši zapis v tem delu je informacija o imenu baze in vratih, preko katerih se strežnik DAViCal povezuje.

```
<?php
    $c->admin_email = 'ime.priimek@organizacija.si';
    $c->system_name = "DAViCal CalDAV Server";
    $c->pg_connect[] = 'dbname=davical port=5432 user=davical_app';
```

V kolikor smo namestitev izvedli pravilno, se nam bo v spletnem brskalniku po navigaciji na stran "https://davical.domena.si" prikazalo prijavno okno, v katerega se moramo vpisati z administratorskim računom, ker trenutno nimamo ustvarjenega še nobenega drugega računa. Po prijavi lahko pregledamo, ali smo pozabili na katero od konfiguracijskih datotek v zavihku "Administration" (Administracija) s klikom na možnost "Setup" (Nastavitve).

4.3 Radicale

Strežnik Radicale lahko prenesemo z uporabo ukaza "apt-get install radicale". Ko to storimo, se nam na sistem prenese namestitev in že imamo pripravljen strežnik z osnovnimi nastavitvami. Nastavitve smo morali v naši namestitvi prilagoditi, da smo lahko uporabili domeno namesto naslova IP. Po namestitvi moramo strežnik še zagnati s klicem servisa "radicale". Sedaj se lahko na strežnik povežejo uporabniki, ki že imajo račun na strežniku samem in niso odvisni od dodatne administracije. Kar je slabo, ker bi kot ponudnik želeli več nadzora nad uporabniškimi računi. Uporabniki se lahko povezujejo s svojih s CalDAV združljivih naprav tako, da pri dodajanju koledarja izberejo metodo sinhronizacije CalDAV ter vpišejo lokacijo in uporabniško ime na strežniku v obliki:

```
https://radicale.domena.si:5232/user/calendar/
```

The screenshot shows the DAViCal Setup page. At the top is a navigation menu with 'Home', 'User Functions', 'Administration', and 'Help'. Below it is a 'Setup' section with 'Upgrade Database' and 'Tools' links. A paragraph explains that the page checks the environment for DAViCal. Below that is a 'Dependencies' table with 15 rows, each showing a dependency, its status (all 'Passed'), and a link to an explanation on the DAViCal Wiki. At the bottom is a 'Configuring DAViCal' section with a note that if the user can read this, things should be mostly working already.

Dependency	Status	
Current DAViCal version	Passed	Explanation on DAViCal Wiki
AWL Library version	Passed	Explanation on DAViCal Wiki
PHP not using Apache Filter mode	Passed	Explanation on DAViCal Wiki
PHP PDO module available	Passed	Explanation on DAViCal Wiki
PDO PostgreSQL drivers	Passed	Explanation on DAViCal Wiki
Database is Connected	Passed	Explanation on DAViCal Wiki
DAViCal DB Schema version	Passed	Explanation on DAViCal Wiki
GNU gettext support	Passed	Explanation on DAViCal Wiki
PHP iconv support	Passed	Explanation on DAViCal Wiki
PHP DateTime class	Passed	Explanation on DAViCal Wiki
Suhosin "server.strip" disabled	Passed	Explanation on DAViCal Wiki
PHP Magic Quotes GPC off	Passed	Explanation on DAViCal Wiki
PHP Magic Quotes runtime off	Passed	Explanation on DAViCal Wiki
PHP calendar extension available	Passed	Explanation on DAViCal Wiki
PHP curl support	Passed	Explanation on DAViCal Wiki

Slika 4.1: DAViCal - namestitev

4.4 AgenDAV

AgenDAV je spletni odjemalec za CalDAV, ki ponuja vmesnik, razvit v tehnologiji AJAX, ki omogoča uporabnikom, da upravljajo in delijo koledarje. AgenDAV je izdan pod licenco GPLv3. Pri razvoju spletnega odjemalca AgenDAV so bili razvijalci osredotočeni na združljivost odjemalca s strežnikom CalDAV DAViCal. AgenDAV nam omogoča ustvarjanje, razvrščanje in deljenje koledarjev in je v naši nalogi ena od poglobitvenih zahtev, okoli katerih smo zgradili rešitev.

Sama namestitev vmesnika AgenDAV od nas ne zahteva veliko napora. Imeti pa moramo predpripravljen ogródje na strežniku. V tem ogródju morajo biti izpolnjene zahteve po strežniku CalDAV, kar smo zagotovili z namestitvijo in konfiguracijo strežnika DAViCal. Pripravljen moramo imeti tudi spletni strežnik, kar smo zagotovili z namestitvijo strežnika Apache. Name-

stiti pa moramo "PHP mbstring extension", ki ni v privzeti PHP namestitvi in nam bo omogočal pravilno prikazovanje odjemalca v spletnem brskalniku. V kolikor tega dodatka za PHP ne namestimo, se prijavno okno ob odpiranju odjemalca sicer pojavi, vendar se po prijavi oblika koledarja ne izriše.

V kolikor smo okolje pripravili, lahko s spletne strani <http://www.agendav.org> prenesemo datoteko "tar.gz" in jo po prenosu na strežnik razširimo v željeno mapo. Zaradi varnostnih razlogov je dobro, da je odjemalec nameščen v mapi, ki ni javno dostopna (primer: var/www/agendav), ampak imamo nanjo ustvarjeno samo simbolično povezavo. Vanjo preusmerimo tudi virtualnega gostitelja, ki nam omogoča navigacijo do spletnega vmesnika za koledar. Ko imamo datoteke na željenem mestu, je potrebno v bazi MySQL dodati novega uporabnika, ki mora imeti vse privilegije za dostop in urejanje baze "agendav". Ko to storimo, moramo bazo "agendav" napolniti s shemo, ki jo najdemo v mapi /agendav/sql. Tu imamo na voljo dve shemi ".sql". Ena je namenjena za delo z bazo PostgreSQL, druga pa za delo z bazo MySQL. Za našo rešitev smo izbrali slednjo zaradi lažje administracije baze v primeru težav, in sicer z orodjem phpMyAdmin, ki smo ga kot podporni servis dodali naši rešitvi.

Pri namestitvi smo uporabili verzijo 1.2.6, ki pa ima težave pri uporabi prevodov in lahko občasno javlja težavo z datoteko "migration_lang.php". Težavo odpravimo z zagonom posodobitvene skripte "agendavcli", ki popravi zapise v bazi.

```
./bin/agendavcli dbupdate
```

Ta korak je potreben, če imamo namen v prihodnje uporabiti prevod spletnega vmesnika. Za uporabo znotraj podjetja trenutno prevod še ni potreben, bo pa zagotovo eden od pogojev pri razširitvi rešitve v produkcijo. Prevode lahko enostavno vključujemo v vmesnik s prevajanjem privzetih datotek, ki se nahajajo v datoteki "../agendav/web/lang".

Po namestitvi je potrebno še urediti konfiguracijske datoteke, ki določajo povezavo programa z bazo, izgled spletnega vmesnika in nekaj drugih osnovnih funkcij.

```
<?php if ( ! defined('BASEPATH'))
    exit('No direct script access allowed');
//url strani
$config['base_url'] = 'https://agendav.domena.si/';
//nastavimo katere napake naj se beležijo v dnevnik
$config['show_in_log']= array('ERROR','INFO','AUTHERR','AUTHOK');
//lokacija dnevniških datotek.
//Apache mora imeti pravice za pisanje v te mape
$config['log_path'] = '/var/log/agendav';
//kriptirni ključ
$config['encryption_key'] = 'A*****kdk';
$config['proxy_ips'] ='';
//naslov
$config['site_title'] = 'Arnes spletni koledar';
//Logo
$config['logo'] = 'logotip.png';
//noga
$config['footer'] = 'Arnes';
//Preusmeritev po odjavi
$config['logout_redirect_to'] = 'https://agendav.domena.si';
//prikazovanje URL naslova v javnem koledarju
$config['show_public_caldav_url'] = TRUE;
//prevod, ki ga želimo uporabiti
$config['default_language'] = 'sl';
//želimo 24 urni način dneva
$config['default_time_format'] = '24';
//privzet prikaz datuma
$config['default_date_format'] = 'ymd';
//polni format datuma
$config['format_full_date'] = '%a %e %B %Y';
//določimo prvi dan koledarja 0=nedelja
```

```
$config['default_first_day'] = 1;
//privzeta časovna cona.
Izbor se nahaja na --> http://php.net/timezones
$config['default_timezone'] = 'Europe/Ljubljana';
```

V datoteki "caldav.php" smo morali določiti način povezave in povedati odjemalcu, katero vrsto strežnika uporabljamo. V tem delu moramo tudi določiti način avtentikacije in formo, ki jo odjemalec AgenDAV ob prijavi posreduje strežniku DAViCal, da lahko dostopa do koledarja vsakega posameznega uporabnika. Nastavitvena datoteka mora za delovanje povezave med strežnikom CalDAV in spletnim vmesnikom za upravljanjem s koledarjem izgledati tako:

```
//avtentikacijska metoda, ki zahteva PHP Curl razširitev
$config['caldav_http_auth_method'] = CURLAUTH_BASIC;
//url davical strežnika, kjer podamo ob prijavi uporabniško ime
//za dostop do posameznikovega koledarja
$config['caldav_principal_url'] =
    'https://davical.domena.si/caldav.php/%u/';
//naslov za javne koledarje
$config['public_caldav_url'] =
    'https://davical.domena.si/caldav.php/%s/';
//omogočiti moramo deljenje koledarjev
$config['enable_calendar_sharing'] = TRUE;
// nastavimo pravice, ki jih ima lastnik koledarja
$config['owner_permissions'] =
array('all', 'read', 'unlock', 'read-acl',
    'read-current-user-privilege-set',
    'write-acl', 'C:read-free-busy',
    'write', 'write-properties', 'write-content',
    'bind', 'unbind');
//dovoljenja pri objavljenem koledarju na spletu
$config['read_profile_permissions'] =
```

```
array('C:read-free-busy', 'read');
$config['read_write_profile_permissions'] =
    array('C:read-free-busy',
        'read', 'write');
$config['default_permissions'] = array('C:read-free-busy');
```

V datoteki "caldav.php" smo tako podali najosnovnejše informacije, ki so potrebne za povezavo koledarskega strežnika in aplikacije, prek katere lahko uporabniki uporabljajo svoj koledar.

4.5 Roundcube vtičnik

Roundcube je brezplačen in odprtokoden odjemalec IMAP z vmesnikom, ki je podoben vmesnikom v aplikacijah, nameščenih lokalno na računalniku. Za delovanje potrebuje običajen strežnik LAMP, na katerega z domače strani vmesnika prenesemo namestitveno mapo in sledimo navodilom za namestitev, ki so na voljo na domači strani vmesnika.

Prednosti vmesnika Roundcube se hitro pokažejo, ko ugotovimo, da lahko vanj vključujemo vtičnike, ki nam zelo razširijo osnovno funkcionalnost vmesnika samega. Ena od nalog, ki smo si jih zastavili pred začetkom izdelave te naloge, je bila možnost, da v obstoječo rešitev vključimo odjemalca, ki bi se bil sposoben povezati s strežnikom DAViCal. Tako bi lahko vsak uporabnik, ki vmesnik uporablja, začel uporabljati tudi novo storitev koledar, ki bo le klik stran od osnovnega pogleda v poštnem predalu.

Ob začetku izdelave naloge razširitvi vmesnika ni kazalo dobro, ker nismo našli razvitega vtičnika, ki bi nam omogočal željene funkcije. Kazalo je, da bo storitev na voljo le za interno uporabo zaposlenih v podjetju, dokler sami ne razvijemo aplikacije, ki bo sposobna komunicirati s strežnikom DAViCal. Proti koncu izdelave naloge pa se je vse obrnilo. Majhna skupina razvijalcev, ki delujejo pod imenom MyRoundcube, je razvila vmesnik, iz katerega smo lahko izločili komponente, ki smo jih nato uporabili pri vdelavi v naš vmesnik Roundcube.

Težji del te naloge je bil ravno izločiti komponente, ki jih ne bomo uporabili za našo rešitev, ker je bil vtičnik nov in zelo slabo dokumentiran. Tako smo morali s sistemom postopnega vključevanja in pregledovanja odvisnosti komponent določiti tiste, ki smo jih nato vgradili v naš testni vmesnik.

Samo dodajanje vtičnikov v vmesnik Roundcube ni zapleteno; v konfiguracijski datoteki za spletni vmesnik je potrebno le definirati, katere vtičnike želimo uporabiti. Paziti pa moramo, da so vsi pravilno nastavljeni, saj v nasprotnem primeru delovanje vmesnika postane nepredvidljivo in nestabilno.

Ker je konfiguracijska datoteka precej zajetna, se v podrobnosti ne bomo spuščali preveč, temveč bomo pokazali le poglaviten del, ki vtičniku omogoči, da se poveže s posameznikovim računom na koledarskem strežniku DAVICAL.

```
$rcmail_config['default_caldav_backend'] = array(
    'user' => '%u',
    'url' => 'https://davical.domena.si/caldav.php/%u/calendar');
```


Poglavje 5

Testiranje v realnem okolju

Na tej točki imamo postavljene in povezane vse glavne in podporne servise, ki jih potrebujemo za pravilno delovanje spletnega koledarja. S testi želimo preveriti delovanje sistema v realnem okolju preden lahko z rešitvijo preidemo v produkcijsko fazo. V testih bomo ponazorili običajno uporabo storitve tako s strani skrbnika, kot tudi s strani uporabnika.

5.1 Načrt testiranja

Osnovne teste delovanja smo izvajali tudi med namestitvijo posameznih delov sistema, kjer do težav ni prihajalo. Ostale teste bomo razdelili v dve skupini. V prvem delu bomo testirali možnosti, ki jih ima skrbnik prek omogočenih vmesnikov za administracijo uporabniških računov. V drugem pa samo uporabo rešitve, kot bo to počel uporabnik. S tem želimo izločiti napake, ki se lahko pojavijo na strani uporabnika ob splavitvi storitve.

Preizkusni scenariji v skrbniškem vmesniku:

- dodajanje in brisanje uporabnikov,
- dodajanje in odvzemanje pravic uporabnikom,
- spreminjanje atributov.

Preizkus delovanja rešitve:

- dostop prek spletnih vmesnikov,
- nastavitve in uporaba odjemalca.

5.2 Administracija uporabnikov

V administratorskem načinu upravljanja z strežnikom DAViCal imamo pravice za dodajanje uporabnikov in za določanje njihovih pravic. Dodajanje in brisanje uporabnikov poteka brez zapletov. Paziti moramo le pri določanju nivojev pravic, ki jih ima uporabnik. V kolikor napačno določimo pravice je možnost, da se bodo pojavile težave na strani uporabnika, ko bo želel dodati nov koledar ali deliti obstoječega.

Uporabnik mora imeti za uporabo vseh funkcij, ki nam jih ponujata Agen-DAV in koledarski vtičnik za Roundcube, pravice za branje in pisanje, branje pri trenutnem dostopu in pravice za zahteveke "prost/zaseden". Te pravice nam določijo funkcionalnost, da lahko koledarski vnos delimo z drugimi uporabniki sistema in da ti lahko dobijo povratno informacijo o naši zasedenosti v predlaganem terminu.

The screenshot shows a web-based user administration interface. At the top, a blue header bar displays "Principal: caltest". Below this, the user's details are shown in a light purple background with various input fields and checkboxes. The "Principal ID" is 1001, "Username" is "user", "Fullname" is "caltest", and "Email Address" is "ime.priimek@domena.si". The "Date Format Style" is set to "European" and "Principal Type" is "Person". Under "Privileges granted to All Users", there are several tabs: ALL, READ/WRITE, READ, FREE/BUSY, SCHEDULE DELIVER, and SCHEDULE SEND. The "ALL" tab is selected, showing a grid of checkboxes for various permissions, most of which are checked. At the bottom, there is a blue button labeled "APPLY CHANGES".

Principal ID:	1001																																				
Username:	user																																				
Change Password:																																				
Confirm Password:																																				
Fullname:	caltest																																				
Email Address:	ime.priimek@domena.si																																				
Locale:	*** Default Locale ***																																				
Date Format Style:	European																																				
Principal Type:	Person																																				
Administrator:	<input type="checkbox"/>																																				
Active:	<input checked="" type="checkbox"/>																																				
Privileges granted to All Users:	<table border="1"> <thead> <tr> <th>ALL</th> <th>READ/WRITE</th> <th>READ</th> <th>FREE/BUSY</th> <th>SCHEDULE DELIVER</th> <th>SCHEDULE SEND</th> </tr> </thead> <tbody> <tr> <td><input checked="" type="checkbox"/> Read</td> <td><input checked="" type="checkbox"/> Write Metadata</td> <td><input checked="" type="checkbox"/> Write Data</td> <td><input type="checkbox"/> Override a Lock</td> <td><input type="checkbox"/> Read Access Controls</td> <td></td> </tr> <tr> <td><input checked="" type="checkbox"/> Read Current User's Access</td> <td><input checked="" type="checkbox"/> Create Events/Collections</td> <td><input checked="" type="checkbox"/> Delete Events/Collections</td> <td></td> <td></td> <td></td> </tr> <tr> <td><input type="checkbox"/> Write Access Controls</td> <td><input checked="" type="checkbox"/> Read Free/Busy Information</td> <td><input checked="" type="checkbox"/> Scheduling: Deliver an Invitation</td> <td></td> <td></td> <td></td> </tr> <tr> <td><input checked="" type="checkbox"/> Scheduling: Deliver a Reply</td> <td><input checked="" type="checkbox"/> Scheduling: Query free/busy</td> <td><input checked="" type="checkbox"/> Scheduling: Send an Invitation</td> <td></td> <td></td> <td></td> </tr> <tr> <td><input checked="" type="checkbox"/> Scheduling: Send a Reply</td> <td><input checked="" type="checkbox"/> Scheduling: Send free/busy</td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	ALL	READ/WRITE	READ	FREE/BUSY	SCHEDULE DELIVER	SCHEDULE SEND	<input checked="" type="checkbox"/> Read	<input checked="" type="checkbox"/> Write Metadata	<input checked="" type="checkbox"/> Write Data	<input type="checkbox"/> Override a Lock	<input type="checkbox"/> Read Access Controls		<input checked="" type="checkbox"/> Read Current User's Access	<input checked="" type="checkbox"/> Create Events/Collections	<input checked="" type="checkbox"/> Delete Events/Collections				<input type="checkbox"/> Write Access Controls	<input checked="" type="checkbox"/> Read Free/Busy Information	<input checked="" type="checkbox"/> Scheduling: Deliver an Invitation				<input checked="" type="checkbox"/> Scheduling: Deliver a Reply	<input checked="" type="checkbox"/> Scheduling: Query free/busy	<input checked="" type="checkbox"/> Scheduling: Send an Invitation				<input checked="" type="checkbox"/> Scheduling: Send a Reply	<input checked="" type="checkbox"/> Scheduling: Send free/busy				
ALL	READ/WRITE	READ	FREE/BUSY	SCHEDULE DELIVER	SCHEDULE SEND																																
<input checked="" type="checkbox"/> Read	<input checked="" type="checkbox"/> Write Metadata	<input checked="" type="checkbox"/> Write Data	<input type="checkbox"/> Override a Lock	<input type="checkbox"/> Read Access Controls																																	
<input checked="" type="checkbox"/> Read Current User's Access	<input checked="" type="checkbox"/> Create Events/Collections	<input checked="" type="checkbox"/> Delete Events/Collections																																			
<input type="checkbox"/> Write Access Controls	<input checked="" type="checkbox"/> Read Free/Busy Information	<input checked="" type="checkbox"/> Scheduling: Deliver an Invitation																																			
<input checked="" type="checkbox"/> Scheduling: Deliver a Reply	<input checked="" type="checkbox"/> Scheduling: Query free/busy	<input checked="" type="checkbox"/> Scheduling: Send an Invitation																																			
<input checked="" type="checkbox"/> Scheduling: Send a Reply	<input checked="" type="checkbox"/> Scheduling: Send free/busy																																				
APPLY CHANGES																																					

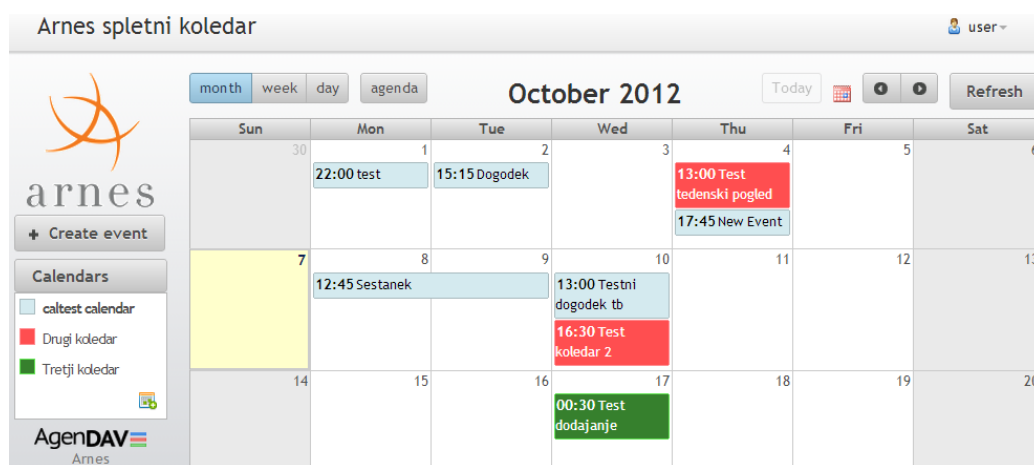
Slika 5.1: Administracija posameznega uporabnika

5.3 Spletna vmesnika AgenDAV in Roundcube

Z uporabo obeh spletnih vmesnikov smo pridobili možnost urejanja koledarjev preko spleta. Prvi je samostojna rešitev, je direktno dosegljiv prek povezave `https` in ne zahteva uporabniškega računa za elektronsko pošto, kot je to potrebno za dostop prek vmesnika za pregledovanje spletne pošte Roundcube. Pri uporabi vmesnika Roundcube moramo paziti, da ima uporabnik enako uporabniško ime in geslo za dostop do pošte, kot ga ima nastavljenega na poštnem strežniku; v nasprotnem primeru prijava v spletni koledar ne bo delovala in bo potreben vpis pravega gesla za koledarski strežnik v nastavitve vtičnika, ki se povezuje s koledarskim strežnikom. Preizkusili smo dodajanje, brisanje, uvoz, izvoz in distribucijo koledarjev. V vseh primerih sta se oba odjemalca izkazala kot zanesljiva. Koledarski strežnik je v ozadju deloval brez težav in je omogočal takojšnjo sinhronizacijo med obema vmesnikoma.

Roundcube vmesnik pa nam ponuja nekaj funkcionalnosti več, kot nam jih lahko samo vmesnik AgenDAV. Ker je vmesnik Roundcube prvotno namenjen za pregledovanje elektronske pošte, nam tako tudi omogoča pošiljanje elektronskih sporočil, ki jih lahko uporabljamo kot opomnik.

Preizkusili smo tudi skupno uporabo koledarjev z drugimi uporabniki. Z uporabo vmesnika AgenDAV poteka delitev koledarjev prek uporabniškega imena, kar je za interno rabo v podjetju zadovoljivo, ne pride pa v poštev v produkciji, kjer bi uporabniki za delitev koledarja morali poznati uporabniško ime osebe, s katero želijo deliti koledar.



Slika 5.2: Prikaz vnosov v spletnem vmesniku AgenDAV

Calendar ::: user

Provider

Username

Password

CalDAV Url
f.e.: <https://www.google.com/calendar/dav/john.doe@gmail.com/ever>

Authentication Method

Reminders

Replicate CalDAV database every minutes

Replicate CalDAV for the years

Default view

Timeslots per hour

Event duration (default)

First weekday

Workdays

Slika 5.3: Nastavitve v vmesniku Roundcube

5.4 Odjemalci

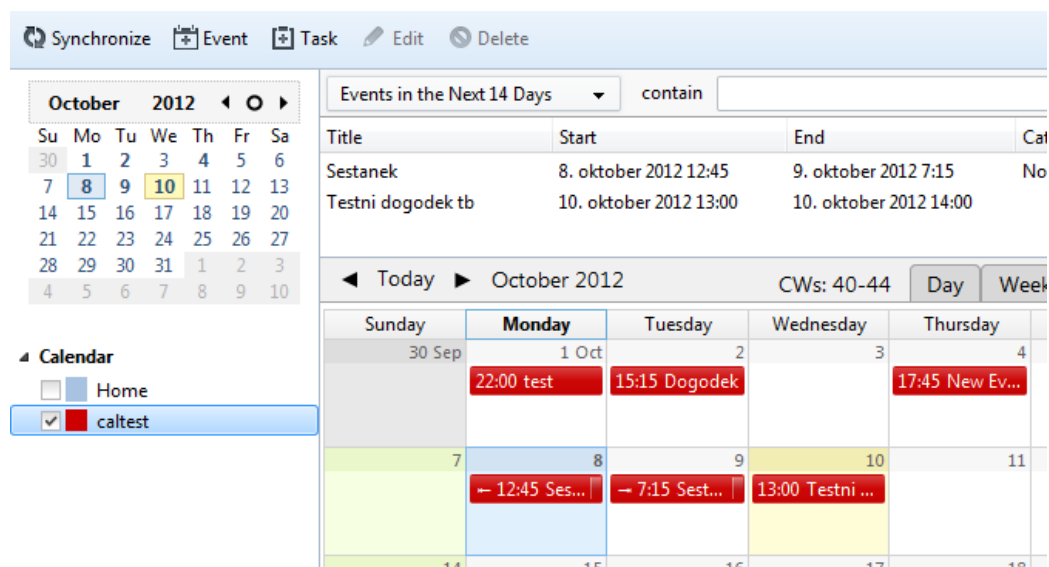
Preizkusili smo tudi dodajanje koledarjev v nekaj najbolj znanih odjemalcev, ki podpirajo protokol CalDAV. V vseh odjemalcih je bilo mogoče brez težav dodati koledar. Za dodajanje povezave s koledarskim strežnikom moramo vedeti samo svoje uporabniško ime ter geslo in naslov, na katerem se naš koledarski strežnik nahaja. Večina odjemalcev nam pri dodajanju spletnega koledarja ponudi možnost, da izberemo protokol CalDAV, in od nas zahteva le naslov koledarja. Pri tem nam uporabniškega imena ni potrebno vpisovati, ker je razvidno že iz vpisane povezave, in moramo vpisati le geslo.

Edini odjemalec, ki je povzročal težave, je bil Microsoft Outlook, ki že ima vključen koledar in ni odvisen od vtičnikov, kot je na primer Lightning za Thunderbird. Težavo smo rešili z uvozom vtičnika CalDAV za Outlook, ki je programu omogočil način povezave s strežnikom. Ko smo to storili, se je tudi Outlook normalno sinhroniziral s strežnikom DAViCal. Tu moramo poudariti, da so spremembe vidne prek vseh treh načinov, ki so nam na voljo za vpogled v koledar. Kar je tudi logično, saj se morajo vsi odjemalci sinhronizirati z istim računom. V kolikor sprememba ni vidna takoj, to pomeni, da se sinhronizacija, ki poteka na določene cikle, še ni izvedla. Sinhronizacijo lahko sprožimo v vseh odjemalcih s klikom na gumb "Sinchroniziraj".

5.5 Rezultati testiranja

Izvedeni testi so pokazali, da je izbrana programska oprema stabilna in jo je moč uporabiti tako v poslovnem okolju, kot tudi za postavitev lastnega sinhronizacijskega strežnika, v kolikor kot uporabnik ne zaupamo rešitvam, ki so trenutno dosegljive na tržišču.

Čeprav med testiranjem na večje težave nismo naleteli, lahko v bodoče pričakujemo napake, ki se pojavijo kot posledica površne uporabe storitve s strani uporabnika. Primer, kjer se takšna težava lahko pojavi, je pri povezovanju nekaterih aplikacij, namenjenih mobilnim napravam z operacijskim sistemom Android.



Slika 5.4: Razširitev Lightning za Mozilla Thunderbird

Poglavje 6

Sklepne ugotovitve

Skozi diplomsko nalogo smo spoznali storitev koledarskega strežnika in tehnologije, ki stojijo v ozadju. Nalogo bi si lahko poenostavili z uporabo plačljive storitve, vendar smo se omejili na odprtokodne rešitve, ki so nam prinesle precej prikritega dela. Preskusili smo več možnih rešitev in se na koncu odločili za rešitev, ki je potrebovala najmanj dodatnega razvoja. Za takšno se je izkazal strežnik DAViCal, ki nam je že ob namestitvi nudil funkcionalnosti, zahtevane v začetku naloge.

Z uporabo spletnega odjemalca AgenDAV smo prišli do točke, kjer smo zadostili osnovnim zahtevam podjetja in tako omogočili delovanje aplikacije, namenjene zaposlenim v podjetju. Tako ima sedaj vsak, ki pridobi uporabniško ime in geslo, s katerima se lahko predstavi strežniku, možnost dostopati do koledarskega strežnika in ustvarjati ter deliti svoje koledarje. Z uporabo spletnega vmesnika smo tudi zagotovili dostop do koledarskega strežnika od kjerkoli.

Ko smo prešli ta mejnik, smo se lahko osredotočili na možnost integracije koledarskega strežnika v obstoječo ponudbo. To nam je deloma uspelo z vtičnikom za Roundcube, ki smo ga lahko prilagodili za delovanje z našim strežnikom CalDAV. Vendar bo za uporabo v produkciji potreben še dodaten razvoj vtičnika, ki bo uporabniku skrnil nekaj možnosti, ki lahko vplivajo na delovanje koledarja.

Preskusili smo tudi delovanje večine odjemalcev, ki so trenutno na voljo in dokazali, da smo dosegli visok nivo interoperabilnosti rešitve. Na vseh odjemalcih imamo lahko v nekaj korakih nastavljeno povezavo z strežnikom DAViCal. Tako so vse spremembe, ki jih izvajamo preko ene od naprav, s katero dostopamo do svojega računa, tudi takoj vidne na drugi napravi. Projekt bi osebno označil za uspešen, ker nam je uspelo uresničiti zastavljene cilje, istočasno pa smo raziskali možnosti, ki jih bomo lahko v bodoče uporabili pri razvoju.

Slike

1.1	Pogled s strani uporabnika	2
2.1	Zastavljen nivo interoperabilnosti rešitve	5
3.1	Primer integracije Funambol strežnika v SOGo	9
4.1	DAViCal - namestitev	20
5.1	Administracija posameznega uporabnika	28
5.2	Prikaz vnosov v spletnem vmesniku AgenDAV	30
5.3	Nastavitve v vmesniku Roundcube	30
5.4	Razširitev Lightning za Mozilla Thunderbird	32

Literatura

- [1] (2005) Jim Whitehead, *Open Calendar Sharing and Scheduling with Cal-DAV*. Dostopno na:
<http://users.soe.ucsc.edu/~ejw/papers/dusseault2005CalDAV.pdf>
- [2] (2012) *DAViCal*. Dostopno na:
<http://www.davical.org/>
- [3] (2012) *WebDAV*. Dostopno na:
<http://en.wikipedia.org/wiki/WebDAV>
- [4] (2007) RFC *Calendaring Extensions to WebDAV (CalDAV)*. Dostopno na:
<http://www.ietf.org/rfc/rfc4791.txt>
- [5] (2012) *AgendaDAV*. Dostopno na:
<http://agendav.org/>
- [6] (2012) *SOGO*. Dostopno na:
<http://www.sogo.nu/>
- [7] (2012) *Blog of Andrew McMillan*. Dostopno na:
<http://andrew.mcmillan.net.nz/>
- [8] (2012) *Radicale*. Dostopno na:
<http://radicale.org/>
- [9] (2012) *Roundcube*. Dostopno na:
<http://roundcube.net/>

- [10] (2012) *myRoundcube*. Dostopno na:
<http://myroundcube.com/>
- [11] (2012) *DAViCal*. Dostopno na:
<http://www.davical.org/>
- [12] (2012) *iCalendar*. Dostopno na:
<http://en.wikipedia.org/wiki/ICalendar>
- [13] (2012) *CalConnect Applauds Publication of IETF CalDAV Proposed Standard*. Dostopno na:
<http://calconnect.org/publicity/070320calconnectcaldavrelease.pdf>
- [14] (2012) *The Calendaring and Scheduling Consortium*. Dostopno na:
<http://www.calconnect.org/>