

**KAUNO TECHNOLOGIJOS UNIVERSITETAS**  
**INFORMATIKOS FAKULTETAS**  
**KOMPIUTERIŲ KATEDRA**

**VALDEMARAS JASAITIS**

**AUTOMATINĖ BEVIELIO TINKLO PRIEIGOS TAŠKŲ**  
**PAIEŠKOS SISTEMA**

**MAGISTRO DARBAS**

**Darbo vadovas prof. dr. R. Jasinevičius**

**KAUNAS, 2008**

**KAUNO TECHNOLOGIJOS UNIVERSITETAS**  
**INFORMATIKOS FAKULTETAS**  
**KOMPIUTERIŲ KATEDRA**

**AUTOMATINĖ BEVIELIO TINKLO PRIEIGOS TAŠKŲ**  
**PAIEŠKOS SISTEMA**

**MAGISTRO DARBAS**

**Recenzentas**

**dr. Arūnas Žvironas**

**2008-01-**

**Vadovas**

**prof. dr. Raimundas Jasinevičius**

**2008-01-**

**Magistrantas**

**Valdemaras Jasaitis, IFM-2/4 gr.**

**2008-01-**

**KAUNAS, 2008**

# TURINYS

IVADAS.....	7
1. BEVIELIO INTERNETO PAIEŠKOS SISTEMŲ ANALIZĖ .....	9
1.1 TYRIMO SRITIS, OBJEKTAS IR PROBLEMA .....	9
1.2 PASAULINIO LYGIO ANALOGIŠKŲ SISTEMŲ LYGINAMOJI ANALIZĖ .....	11
1.2.1 Hotspot Finder.....	11
1.2.2 Hitchhiker.....	12
1.2.3 FON WiFi.....	12
1.2.4 Nokia 6301 .....	13
1.3 LITERATŪROS ŠALTINIuose PATEIKTŲ SPRENDIMŲ PROBLEMAI SPREŠTI LYGINAMOJI ANALIZĖ .....	14
1.3.1 Objektų ar procesų valdymas .....	14
1.3.2 Miglotasis valdymas.....	16
1.3.2.1 Miglotosios valdymo sistemos .....	18
1.3.2.2 Miglotosios pateikties priemonės ekspertui .....	20
1.3.2.3 Miglotųjų sistemų valdymo privalumai: .....	21
1.4 ANALIZĖS METODŲ IR PRIEMONIŲ PARINKIMAS BEI PROGRAMINĖS ĮRANGOS ANALIZĖ.....	22
1.5 PROJEKTAVIMO METODŲ IR PRIEMONIŲ PARINKIMAS .....	26
1.6 RIZIKOS FAKTORIŲ ANALIZĖ.....	27
1.7 REZULTATO KOKYBĖS KRITERIJAI.....	27
1.8 ANALIZĖS IŠVADOS .....	28
2. PROJEKTAVIMAS .....	29
2.1 PROJEKTO TIKSLAS .....	29
2.2 REIKALAVIMO MODELIS .....	29
2.2.1 Programos grafinės sąsajos kūrimas .....	30
2.2.2 Prisijungimo bei paieškos valdymo biblioteka.....	32
2.2.3 Duomenų talpyklos biblioteka .....	34
2.2.4 Miglotosios logikos biblioteka .....	35
2.2.5 Sistemos saugumas ir suderinamumas su kitomis sistemomis.....	41
2.3 SISTEMOS PROJEKTAS .....	42
2.4 TESTAVIMAS.....	46
2.5 SISTEMOS NAUDOJIMO INSTRUKCIJA .....	47
2.6 SISTEMOS DIEGIMAS .....	50
2.7 PROJEKTO IŠVADOS.....	50
3. EKSPERIMENTINIS TYRIMAS.....	51
3.1 SUKURTOS SISTEMOS KOKYBĖS TYRIMAS.....	51
3.2 SISTEMOS APRIBOJIMAI IR GALIMYBĖS .....	54
3.3 TOLIMESNĖS SISTEMOS TOBULINIMO PLĖTOJIMO GALIMYBĖS .....	55
IŠVADOS.....	56
LITERATŪRA.....	58
SUMMARY .....	60
TERMINŲ, SANTRUMPŲ ŽODYNAS .....	61

## PAVEIKSLŲ SĄRAŠAS

<b>1 pav.</b> Prieigos taškų diagrama .....	10
<b>2 pav.</b> Programos Hotspot Finder pagrindiniai langai .....	11
<b>3 pav.</b> Programos Hitchhiker pagrindiniai langai .....	12
<b>4 pav.</b> Programos FON WiFi pagrindiniai langai .....	12
<b>5 pav.</b> Programos jungimosi langai.....	13
<b>6 pav.</b> Objekto valdymo sistema.....	15
<b>7 pav.</b> Priklausomybės aibei A tikrumo funkcija .....	17
<b>8 pav.</b> Valdymo sistema su grįžtamuju ryšiu.....	18
<b>9 pav.</b> Tiesioginio valdymo sistema .....	19
<b>10 pav.</b> Proceso valdymo sistema ir miglotosios logikos reguliatoriaus architektūros modelis.....	20
<b>11 pav.</b> Veiklos panaudojimo atvejų diagrama.....	22
<b>12 pav.</b> Supaprastinta klasių analizės diagrama.....	23
<b>13 pav.</b> Nokia Series 60 platformos telefonai.....	24
<b>14 pav.</b> FON maršrutizatorius.....	24
<b>15 pav.</b> Tinklo modelis .....	25
<b>16 pav.</b> Komponentinė sistemos architektūra .....	27
<b>17 pav.</b> Sistemos sluoksninė struktūra.....	30
<b>18 pav.</b> Grafinės sąsajos klasių diagrama .....	31
<b>19 pav.</b> Pagrindinis sistemos langas .....	32
<b>20 pav.</b> Pagrindinis sistemos meniu.....	32
<b>21 pav.</b> Prisijungimo komponento klasių diagrama.....	33
<b>22 pav.</b> Duomenų talpyklos komponento klasių diagrama.....	35
<b>23 pav.</b> Miglotosios logikos struktūra.....	36
<b>24 pav.</b> Įvertis „Greitis“.....	37
<b>25 pav.</b> Įvertis „Judėjimo kryptis“ .....	38
<b>26 pav.</b> Įvertis „Signalio stiprumas“ .....	38
<b>27 pav.</b> Išėjimų rezultatų termai .....	39
<b>28 pav.</b> Rezultato pavyzdys .....	40
<b>29 pav.</b> Migloto komponento klasių diagrama .....	41
<b>30 pav.</b> Sistemos sekų diagrama .....	42
<b>31 pav.</b> Veiklos diagrama apjungianti automatinę paiešką.....	43

<b>32 pav.</b> Komponentų sekų diagrama.....	44
<b>33 pav.</b> Sistemos duomenų bazė .....	45
<b>34 pav.</b> Pilna sistemos struktūra .....	45
<b>35 pav.</b> Prieigos taškų paieška .....	47
<b>36 pav.</b> HandyWi meniu .....	47
<b>37 pav.</b> Paieškos režimo langas .....	48
<b>38 pav.</b> HandyWi vartotojo paskyros langai.....	48
<b>39 pav.</b> HandyWi prisijungimas prie maršrutizatoriaus .....	49
<b>40 pav.</b> BateryInfo programos langas .....	52
<b>41 pav.</b> Elemento stiprumas, po įvairių bandymų .....	53

## LENTELIŲ SĄRAŠAS

1 lentelė. <b>Programų analizė</b> .....	14
2 lentelė. <b>FON charakteristikos</b> .....	24
3 lentelė. <b>Funkciniai reikalavimai</b> .....	26
4 lentelė. <b>Nefunkciniai reikalavimai</b> .....	26
5. lentelė. <b>Iėjimai</b> .....	37
6. lentelė. <b>Duomenų bazės laukai</b> .....	45
7 lentelė. <b>Elemento stiprumas, kai įrenginiai nejuda</b> .....	51
8 lentelė. <b>Elemento stiprumas, kai įrenginiai juda chaotiškai</b> .....	52
9 lentelė. <b>Elemento stiprumas, kai įrenginiai judėjo pastoviu greičiu</b> .....	53

## IVADAS

XXI – imame amžiuje sunku įsivaizduoti civilizuoto žmogaus gyvenimą be interneto. Internetas tampa būtinybe ir kuo toliau, tuo labiau mes nuo jo esame priklausomi. Jau dabar turime šaldytuvus, kurie automatiškai sugeba užsakyti prekes internetu. Viskas yra gerai, jei mūsų įrenginiai yra stacionarūs ir jie nejudą kartu su mumis, tačiau žmogaus poreikiams augant reikalingas visada geras, saugus ir patikimas interneto ryšys. Tuo tikslu žmonės sugalvojo bevielį interneto ryšį.

Bevielis internetas (WiFi) tampa kasdienybe mūsų gyvenime. Kuo toliau tuo daugiau išmaniųjų telefonų turi integruotus WiFi adapterius ir vis dažniau siūloma naudotis bevielio interneto ryšiu. Viso šito privalumas akivaizdus - mes turime pigų greitą ir saugų ryšį.

**Temos aktualumas** pasireiškia tuo, kad dabartiniu momentu yra sukurta daug prieigos taškų paieškos sistemų, tačiau šios sistemos „nemoka“ automatiškai susirasti bei prisijungti prie tinkamiausio prieigos taško. Pagrindinė egzistuojančių sistemų problema ta, kad prisijungimas prie prieigos taškų buvo labai skirtingas, dėl šios priežasties ir nebuvo galimybių kurti „protingą“ prisijungimo sistemą. Tačiau rinkoje sparčiai daugėja prieigos taškų, prie kurių galima prisijungti naudojant tik vienintelį vartotojo vardą ir slaptažodį, todėl atsirado patraukli galimybė kurti automatinę prieigos taškų paieškos sistemą.

Kita egzistuojančių sistemų problema ta, kad nėra aišku, koku būdu kursime automatinę paieškos sistemą. Vienas paprasčiausių ir lengviausių būdų būtų periodiškai ieškoti prieigos taškų (AP), ir prisijungti prie stipriausią ryšį turinčio AP. Bet Symbian SDK (Software Development Kit) pateiktoje dokumentacijoje parašyta, kad pastovus prieigos taškų ieškojimas reikalauja didelių energetinių telefono resursų, todėl maitinimo elementas nepakrautas budės labai trumpai. Tai yra labai didelis trūkumas, kadangi norėdami turėti pastovų internetą greit iškrautume savo telefoną ir išvis prarastume ryšį.

Taigi, kuriant automatinę AP paieškos sistemą, reikia skirti didelį dėmesį telefono energetiniams resursams išsaugoti. Dėl šios priežasties, reikės suprojektuoti protingą prieigos taškų valdymo sistemą, kuri būtų tiek „protinga“ ir ieškotų AP tada, kai to tikrai reikia.

**Tyrimo objektas** – tarp WiFi adapterių ir išmaniųjų telefonų pastovaus ryšio išlaikymas.

**Tyrimo tikslas** – sukurti programinę įrangą, kuri turėtų galimybę automatiškai surasti aplinkoje esančius prieigos taškus, naudotų kiek įmanoma mažiau energetinių telefono resursų bei teiktų prisijungimo galimybę.

**Tyrimo problema** – pastovaus bevielio interneto ryšio judančiuose objektuose stoka ir greitai išsenkantys telefono energetiniai resursai.

**Tyrimo uždaviniai:**

1. Apžvelgti veikiančias programinės įrangos sistemas, kurios sugeba prisijungti prie prieigos taškų, identifikuoti jų problemas.

2. Padaryti „protingą“ automatinio prisijungimo sistemą, kuri automatiškai nieko nežinant vartotojui susirastų ir prisijungtų prie vis naujų prieigos taškų.

3. Pademonstruoti programos veikimo galimybes.

Darbas sudarytas iš keturių dalių.

**Bevielio interneto paieškos sistemų analizė.** Aprašoma tyrimo sritis, objektas bei remiantis literatūros analize atskleidžiamas, per esamas bevielio interneto prieigos taškų paieškos sistemų problemas, temos aktualumas.

**Projektavimas.** Ši dalis apima komponentų projektavimą bei programavimą. Projektavimui naudojama UML, Visio ir FuzzyTech įrankiai. Šioje dalyje yra pavaizduotos komponentų klasių, įvykių bei sekų diagramos, pateikta duomenų bazės schema bei miglotųjų įverčių diagramos.

**Eksperimentinis tyrimas.** Naujai sukurtos „HandyWi“ sistemos kokybinių tyrimo duomenų palyginimas su analogiškais rinkoje funkcionuojančiomis sistemomis.

**Išvados.** Šioje dalyje pateikta apibendrinta informacija apie sukurta bevielio interneto prieigos taškų paieškos sistemos funkcionalumą ir išskirtinumą bei tolimesnes jos plėtojimo galimybes.

**Tyrimo metodika:**

- mokslinės literatūros analizė;
- publikacijų ir pranešimų analizė.

**Darbo rezultatas** – sukurta bevielio interneto prieigos taškų valdymo sistemos reikšmė jos vartotojams. Rezultatų palyginimas.



# 1. BEVELIO INTERNETO PAIEŠKOS SISTEMŲ ANALIZĖ

## 1.1 TYRIMO SRITIS, OBJEKTAS IR PROBLEMA

Pagrindinis *darbo tikslas* – *sukurti veiksmingą programinę įrangą, kurios pagalba būtų galima automatiškai prisijunginėti prie aplinkoje išsidėsčiusių bevielio interneto prieigos taškų ir pademonstruoti programos veikimą Symbian 9.x OS telefonams*. Taigi, šio darbo *tyrimo sritis* apims *bevielio interneto prieigos taškų paieškos sistemos kūrimo galimybes*.

WiFi – bevielio ryšio technologija, skirta įrenginiams naudojantiems plačiajuostį radijo ryšį. Per prieigos taškus (angl. *hotspot*) WiFi naudojamas mobilių ir stacionarių įrenginių prieigai prie interneto, internetinei balso telefonijai, žaidimams, taip pat duomenų tarp įvairių elektroninių prietaisų (DVD grotuvų, televizorių, vaizdo kamerų) mainams. Ši technologija remiasi IEEE 802.11 standartu.

Nuo technologijos sukūrimo 1997 m. WiFi palaikymas įdiegtas populiariausiose operacinėse sistemose, bevielio ryšio įranga komplektuojama daugelyje šiuolaikinių nešiojamų bei delninių kompiuterių, taip pat mobiliuose telefonuose. WiFi prieigos taškai dažnai įrengiami degalinėse, restoranuose ir kitose viešose vietose[1].

Kiekvieną dieną vis daugėja vartotojų, turinčių galimybę prisijungti prie prieigos taškų. Augant rinkai vis labiau domimasi, kaip ir kur būtų galima pakeisti korinius tinklus, kadangi korinių tinklų, kurie kartais vadinami 2,5G tinklais, duomenų perdavimo sparta skaičiuojama dešimtimis kb/s, o WLAN - Mb/s.

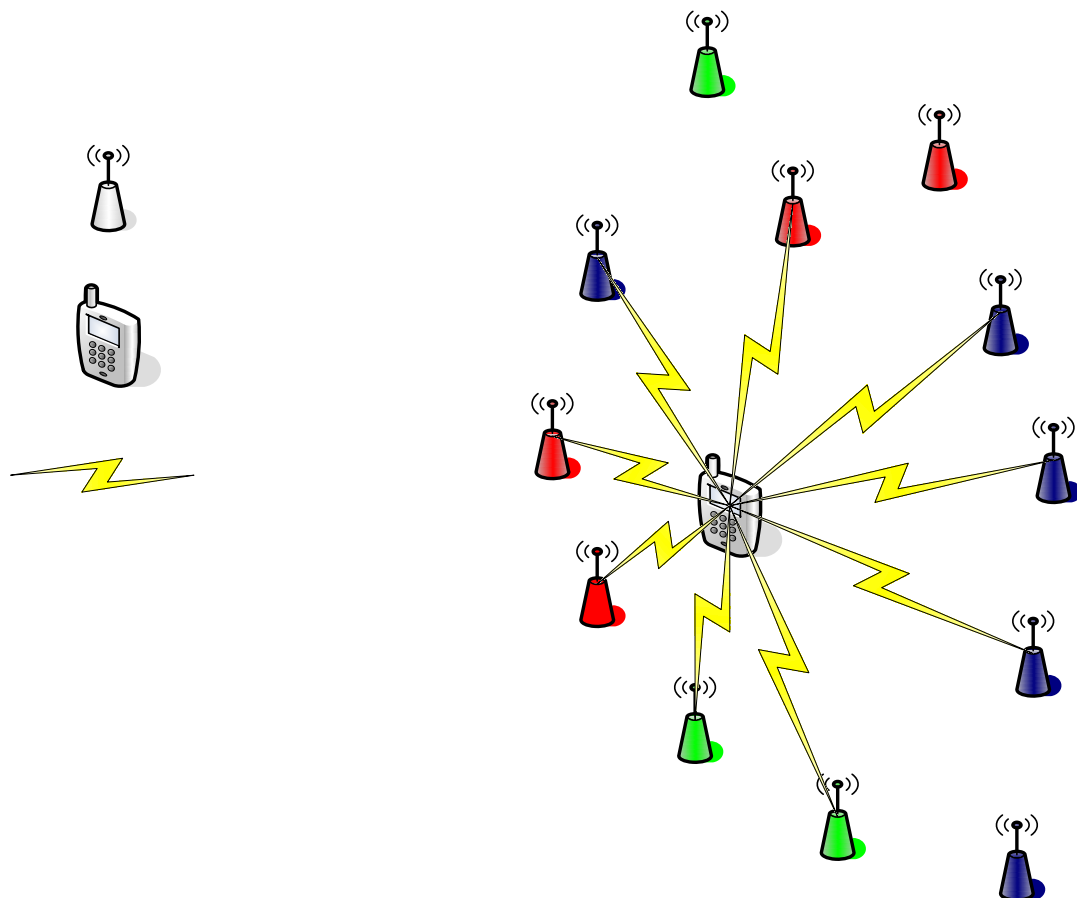
Su WiFi - bevielio greitaigio kompiuterinio tinklo technologija, galima greitai sukurti infrastruktūrą mobilaus interneto ryšiui teikti. Teritorija, kur yra įdiegta WiFi infrastruktūra (ryšio zona) vadinama hotspot. HotSpot'e galima naudotis greitaigiu interneto ryšiu bevieliu būdu - nenaudojant jokių laidų, telefono linijų ar modemo. Dėl paprasto ir pigaus bevielio tinklo technologijos vis daugiau garsių firmų galvoja apie tai, kaip apjungti WiFi tinklus. Pavyzdžiui, kompanija „Truphone“, kuri nukreipia mobiliųjų telefonų skambučius per internetą, ženkliai sumažindama jų kainą, įkūrėjas James Tagg tvirtina, kad Didžiojoje Britanijoje šiuo metu egzistuoja daugiau nei 5 mln. WiFi prieigos taškų, taigi ši belaidė technologija tampa vis labiau prieinama vartotojams[2]. Labai patrauklus „Truphone“ kompanijos siūlomos skambučių sistemos aspektas yra tas, kad skambučiai yra sąlyginai „nemokami“ paslaugos tiekėjams, kuriems nereikia mokėti nacionaliniams tinklams už skambučio perdavimą.

Mobiliojo telefono naudojimas „VoIP“ skambučiams populiarus ne tik Europoje, bet ir JAV, kur WiFi plėtojamas ne mažiau. Vienintelė visuotinio tokio tipo paslaugų taikymo kliūtimi yra laikomas sudėtingas įdiegimas, dėl ko šios paslaugos dar nėra prieinamos visiems vartotojams[2]. James Tagg

paminėjo vieną iš šiame darbe sprendžiamų problemų tai, kad yra *sudėtinga įdiegti tokią sistemą, kuri gebėtų prisijungti vis prie reikiamo prieigos taško*. O problema iškyla, todėl, kad WiFi stotelės yra pasiekiamos maždaug tik 100 metrų atstumu. Dėl šios priežasties *sistema turi dažnai persijunginėti prie kitų prieigos taškų, o dažnas persijunginėjimas reikalauja didelių energetinių mobilaus įrenginio resursų*.

Taigi pagrindinė *darbo problema – išlaikyti pastovų bevielio interneto ryšį judančiuose išmaniuosiuose telefonuose ir išspręsti šių telefonų energetinių resursų problemą*. Tai yra ne tik užtikrinti pastovų internetą kiek tai yra įmanoma, bet ir prailginti telefono elemento budėjimo laiką. Problemos sprendimai bus pademonstruoti ant Symbian trečiosios kartos (platformos Series 60) operacinės sistemos telefonų.

P. Sharmos grupė pasiūlė į mobiliuosius prietaisus įdiegti programinę įrangą, kuri „jaustų“ šalia (pvz., vieno traukinio vagono) esančius „savus“ prietaisus ir iš jų sukurtų WiFi tinklą [3]. Kuriamos *sistemos tikslas ir yra atpažinti „savus“ tai yra FON prieigos taškus, bei prie jų sugebėti prisijungti automatiškai*. Pirmame paveiksle pavaizduota supaprastinta situacija.



1 pav. Prieigos taškų diagrama

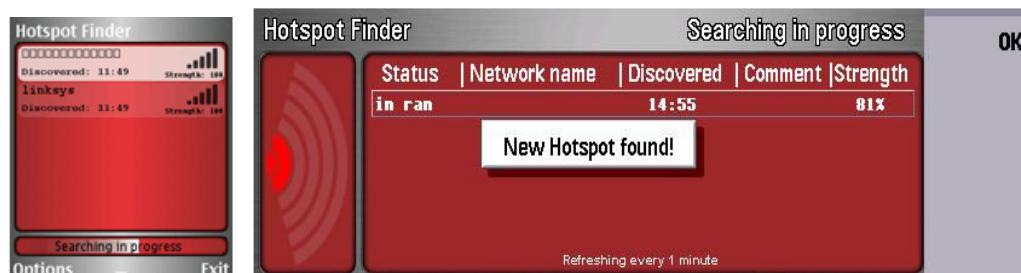
Skirtingomis spalvomis pateikti prieigos taškai demonstruoja jų įvairovę pagal prisijungimo galimybes. Dėl didelės prieigos taškų gausos yra keblu sukurti automatinę prieigos taškų paieškos ir prisijungimo sistemą, kadangi kiekvienas prieigos taškas turi tik jam būdingą prisijungimo strategiją. Tačiau ispanų kompanija „FON Community“ ([www.fon.com](http://www.fon.com)), pasitelkusi į pagalbą tokius gigantus, kaip „Google“ ([www.google.com](http://www.google.com)) ir „Skype“ ([www.skype.com](http://www.skype.com)), nusprendė sukurti ir išleisti į rinką labai daug vienos rūšies prieigos taškų. Šie prieigos taškai vadinami FON vardu. Tarkime, kad, 1 paveiksle mėlyni prieigos taškai yra FON rūšies prieigos taškai, raudoni TEO Zebra prieigos taškai, o žali visi kiti likę prieigos taškai. Kuriama programinė įranga „mokės“ automatiškai prisijunginėti tik prie FON tipo prieigos taškų.

## 1.2 PASAULINIO LYGIO ANALOGIŠKŲ SISTEMŲ LYGINAMOJI ANALIZĖ

Analogiškų sistemų lyginamoji analizė leidžia suprasti kas jau padaryta, ko trūksta esamosiose sistemose, kokie šių sistemų privalumai bei trūkumai. Analizė taip pat padės nuspręsti, kaip turėtų atrodyti kuriamos sistemos vartotojo sąsaja.

### 1.2.1 Hotspot Finder

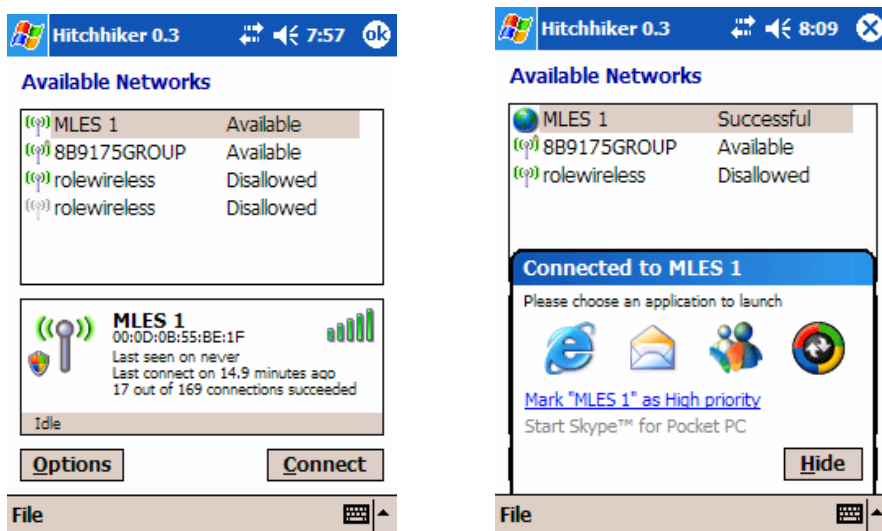
„HotSpot Finder“ tai programa, kuri leidžia vartotojui, bet kada susirasti šalia jo esančius prieigos taškus ir prisijungti prie jų. Programos interfeisas gražiai padarytas, be to programa gali dirbti ir paslėptuoju režimu. „HotSpot Finder“ vartotojui pranešinėja, kada jis priartės prie kokio nors prieigos taško. Didžiausias šios programinės įrangos *trūkumas tas, kad prisijungimas prie prieigos taškų yra rankinis*. Be to, prieigos taškų paieška atliekama pastoviai, arba tam tikru periodu. Vartotojui prisijungus prie prieigos taško, sistema toliau atlikinėja paiešką nustatytu periodu. Ši sistema veikia trečiosios kartos Symbian operacinėje sistemoje[12]. Taigi ši programa turi automatinę prieigos taškų paieškos sistemą, tačiau *programa veikia periodiškai*, neatsižvelgdama į tai, ar tikrai yra būtina pakartotinai ieškoti prieigos taškų; tai didžiausias trūkumas, kadangi *nėra taupomi išmaniojo telefono resursai*. Be to, *periodinis prieigos taškų ieškojimas neužtikrina pastovaus interneto*.



2 pav. Programos Hotspot Finder pagrindiniai langai

### 1.2.2 Hitchhiker

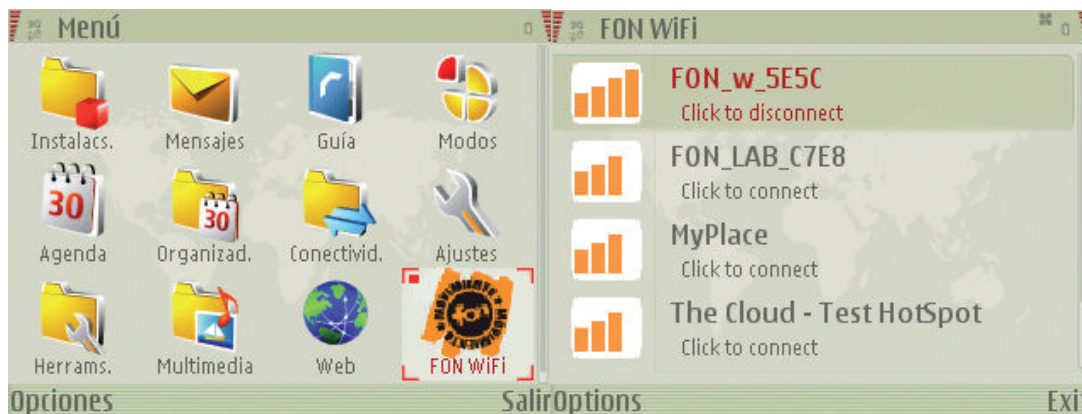
“Hitchhiker“ tai programa, kuri leidžia vartotojui prisijungti prie „Pocket PC“ bevielio interneto. Ši programa labai paprasta ir *nenaudoja jokio automatizavimo*; tiesiog visas valdymas atliekamas paties programinės įrangos vartotojo. Tai, labai didelis nepatogumas ir trūkumas. Be to, ši sistema *dirba tik „Windows CE“ operacinėje sistemoje*[4].



3 pav. Programos Hitchhiker pagrindiniai langai

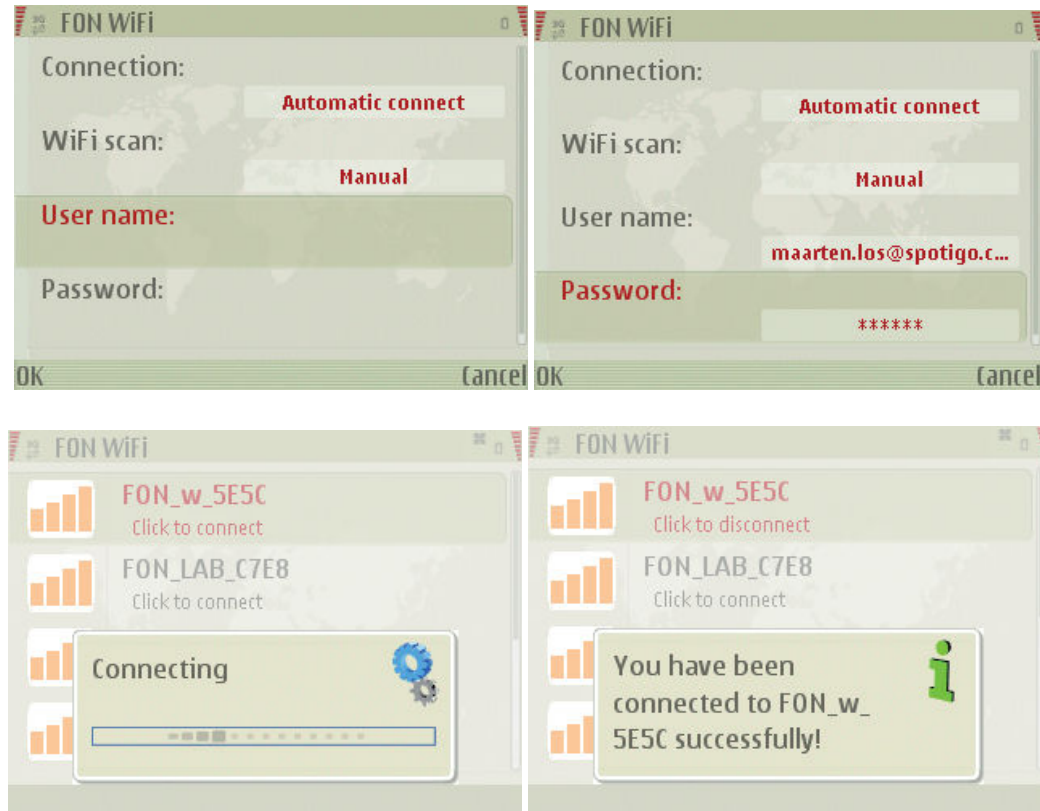
### 1.2.3 FON WiFi

FON WiFi - tai programa, kuri nutaikyta į FON prieigos taškus. Projektas labai rimtas ir konkurencingas. Sistema gali prisijungti sėkmingai ir prie kitų (tai yra ne FON) prieigos taškų, tačiau automatinis prisijungimas palaikomas tik su FON AP. Be to, programinė įranga taip pat palaiko trečios kartos Symbian OS.



4 pav. Programos FON WiFi pagrindiniai langai

Be visų teigiamų savybių, ši programinė įranga turi taip pat galimybę prisijunginėti prie prieigos taškų automatiškai būdu. Tačiau, bandant šią sistemą pastebėta, kad *prieigos taškų paieška sistemoje vyksta periodiškai*. Tai yra trūkumas, kadangi, jei vartotojas yra prisijungęs prie „labai stipraus“ prieigos taško, jis visai nesitiki, kad programinė įranga pradės ieškoti kitų prieigos taškų ir tokiu būdu naudos veltui telefono ir elemento resursus. [5] Šiame darbe tai bus pagrindinė vieta, į kurią bus atkreipiamas didžiausias dėmesys. Kuriamai programinei įrangai reikia įdiegti „protingą“ bevielių prieigos taškų sistemą, kadangi, kaip buvo minėta AP paieška reikalauja didžiausių elemento išteklių.



5 pav. Programos jungimosi langai

#### 1.2.4 Nokia 6301

Didžiausia mobiliųjų telefonų gamintoja pasaulyje suomių kompanija "Nokia" pristatė naują įrenginį, kurio pagalba skambinti galima bus ne tik naudojantis GSM tinklu, bet ir internetu.

Mobiliųjų telefonų gamintoja savo pranešime teigia, kad modelis 6301 sukurtas taikant UMA (unlicensed mobile access) technologiją. Ji aparato vartotojui leidžia naudotis ir tradiciniu GSM tinklu, ir prisijungti prie bevielio interneto. Pasak gamintojų, tokiu būdu sujungiami antžeminio ir mobiliojo ryšio privalumai, o vartotojai galės išvengti mobiliojo ryšio operatorių mokesčių. [6]

Pasak Bloomberg naujienų agentūros, naujasis "Nokia" gaminys pasiekti turėtų Europos vartotojus dar iki metų pabaigos, todėl dabar nuosekliau įvertinti šią technologiją dar sudėtinga.

### 1.2.5 Programų lyginamoji analizė

Pirmoje lentelėje pateikta prieigos taškų paieškos sistemų lyginamoji analizė, iš kurios matosi, kad rinkoje trūksta automatinio prieigos taškų paieškos sistemų. Esamos nepasižymi pilnu valdymo patogumu, neturi „išmanumo“ galimybių, tai yra, taupant mobiliojo telefono energetinius resursus, susirasti prieigos taškus tik tada, kai tai būtina. Programos buvo vertinamos kelėtos inžinierių: Valdemaro Jasaičio; Lino Maldučio ir Mindaugo Stonio. Kiekvienas inžinierius užpildė pirmąją lentelę. Šioje lentelėje (1 lentelė) pateikta bendra inžinierių nuomonė.

1 lentelė. Programų analizė

<b>Programos Kriterijai</b>	<b>Hotspot Finder</b>	<b>Hitchhiker</b>	<b>FON WiFi</b>
Valdymo patogumas. (Dešimties balų vertinimo sistemoje)	<b>8</b>	<b>9</b>	<b>10</b>
Automatinis prisijungimas	<b>Nėra</b>	<b>Nėra</b>	<b>Yra</b>
Darbas foniniame režime	<b>Yra</b>	<b>Yra</b>	<b>Yra</b>
„Protinga“ paieška	<b>Nėra</b>	<b>Nėra</b>	<b>Nėra</b>

Toliau bus apžvelgiami įvairūs, literatūroje minimi, metodai, kurių pagalba galima sukurti „protingą“ programinę įrangą, kuri sugebėtų išlaikyti pastovų interneto ryšį judančiuose objektuose ir taupyti telefono resursus.

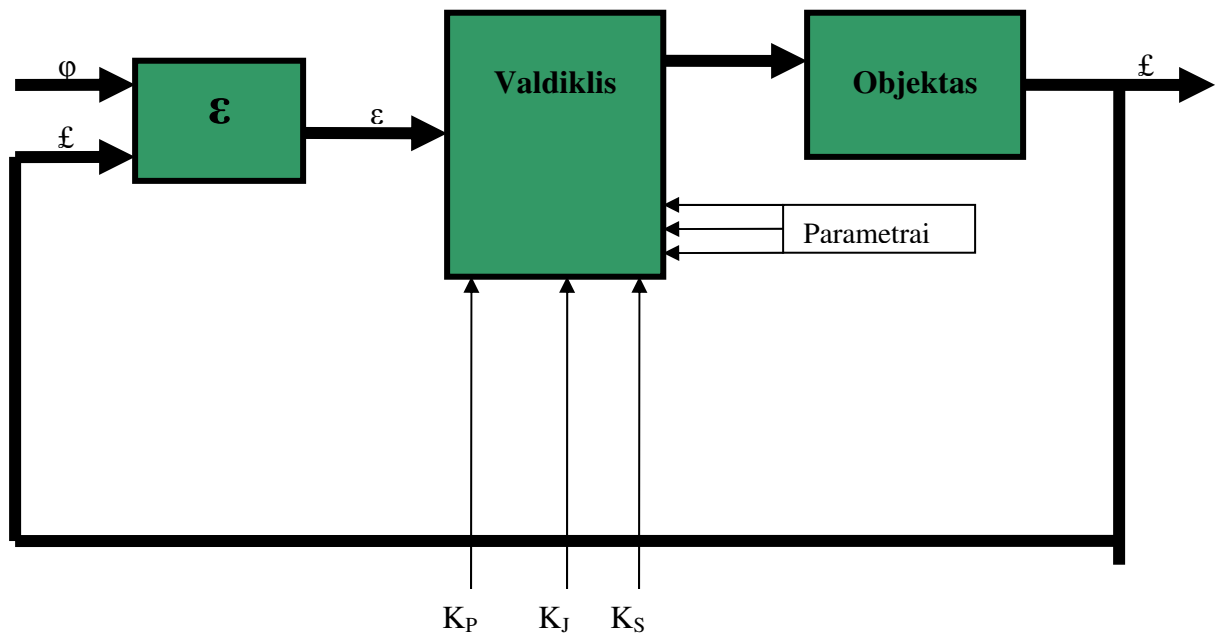
## 1.3 LITERATŪROS ŠALTINIUOSE PATEIKTŲ SPRENDIMŲ PROBLEMAI SPREŠTI LYGINAMOJI ANALIZĖ

### 1.3.1 Objektų ar procesų valdymas

Valdymo sistema yra vadinama informacijos apdorojimo algoritmų ir priemonių jiems realizuoti visuma, skirta užduotam valdymo tikslui pasiekti. Valdymu suprantame tikslinį poveikį, dėl kurio objektas pereina į reikalaujamą būseną. Valdomu objektu gali būti įvairios staklės, kontrolės matavimo aparatūra, technologiniai procesai, lanksčios gamybinės sistemos [7]. Valdymo sistemos struktūra grafiškai pavaizduota blokine schema, ją galima detalizuoti skirtingais lygmenimis. Labiausiai apibendrinta automatinio valdymo sistemos struktūrinė schema pavaizduota 6 paveiksle.



$\varphi$  – Laukiama/pageidaujama objekto reakcija  
 $\xi$  – reakcija  
 $\varepsilon$  – realios reakcijos nesutapimai  
 Kad valdomasis objektas elgtųsi taip, kaip norime, reikėtų, kad  $\varepsilon$  kitimo prasmė artėtų prie nulio.



**6 pav.** Objekto valdymo sistema

Teorija siūlo[8], kad valdant objektą ir formuojant poveikį  $\varepsilon$  (6 paveikslas), reiktų išryškinti valdymo paklaidą

$u_1(t) = K_p \varepsilon(t)$  integruoti pagal laiką  $u_2(t) = K_i \int \varepsilon(t) dt$  ir pridėti kitimo tendencijos išraišką

$$u_3(t) = K_s \frac{d\varepsilon(t)}{dt} \quad (1)$$

Valdiklyje gali būti realizuoti labai įvairūs valdymo algoritmai. Vienas iš jų yra PID valdymo algoritmas. Tokio reguliatoriaus įėjimo signalas  $u(t)$  pagal paklaidos signalą  $\varepsilon(t)$  apskaičiuojamas taip[8]:  $u(t) = u_1(t) + u_2(t) + u_3(t)$ , tai yra

$$u(t) = K_p \varepsilon(t) + K_i \int \varepsilon(t) dt + K_s \frac{d\varepsilon(t)}{dt}; \quad (2)$$

Čia  $K_p$ ,  $K_i$  ir  $K_s$  tam tikro didumo proporcingumo koeficientai. Šių koeficientų reikšmės priklauso nuo objekto savybių bei nuo valdymo tikslų. Tačiau šis siūlomas valdymo modelis netinka, kadangi šiame darbe nekuriame konkretaus valdiklio, o tiesiog naudojames IEEE 802.11g standarto bevielio ryšio

įrenginiu. Darbo uždavinys yra, kiek įmanoma sumažinti prieigos taškų paiešką ir, žinoma, užtikrinti pastovų ryšį, nekeičiant IEEE 80211g įrenginio charakteristikų.

### 1.3.2 Miglotasis valdymas

Kalbėdami apie miglotąsias sistemas, pirmiausia susiduriame su terminais *tradicinė aibė* ir *miglotoji aibė*. Miglotoji aibių teorija yra apibendrinta klasikine loginių aibių teorija, tačiau čia objekto priklausomybės laipsnis aibėje nėra apibrėžtas sveikaisiais skaičiais 0 (objektas aibei nepriklauso) ir 1 (objektas aibei priklauso), bet gali būti, bet kuris skaičius iš intervalo [0,1]. Miglotųjų aibių teorijos pradininku laikomas L. A. Zadehas, kuris pirmasis 1965 m. aprašė tos teorijos pagrindus.[9] 1974 m. E. Mamdani ir jo vadovaujama tyrinėtojų grupė pirmieji pritaikė miglotąją logiką procesams valdyti, o 1979 m. tai buvo panaudota energetikos problemoms spręsti. Dabar miglotosios valdymo sistemos plačiai naudojamos įvairiems sudėtingiems procesams valdyti rizikos valdyme, elektros energetikoje, laivų navigacijoje, cemento gamyboje ir daugelyje kitų sričių.

Miglotojo valdymo metodai paremti miglotomis aibėmis bei veiksmams su jomis. Dabar gana plačiai taikomos dvi miglotojo valdymo sistemos: Mamdani ir Sugeno. Miglotosios logikos sistemos (FLS) apibūdina miglotosios aibės (angl. Fuzzy sets) ir miglotieji ryšiai (angl. Fuzzy relations). FLS yra taisyklėmis pagrįstos sistemos, naudojamos netiesiniam ryšiui tarp sistemos įėjimo ir išėjimo erdvių sudaryti. Vienas svarbiausių miglotosios logikos elementų yra lingvistiniai kintamieji, kurie sudaromi iš vienas kitą dengiančių miglotų aibių, fizikine prasme apibūdinančių tą kintamąjį [10].

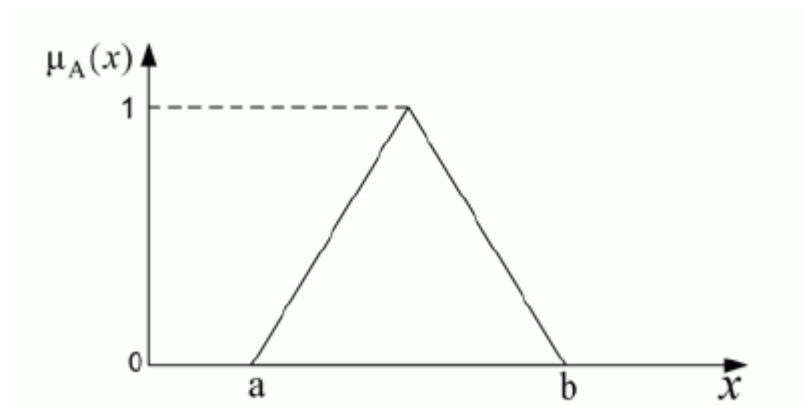
Tarp įėjimo ir išėjimo yra „Taisyklių rinkinys“, kuris ir sprendžia problemą. „Taisyklių rinkinyje“ gali būti ir miglotosios logikos, tiesinės, ekspertinės sistemos, neuroniniai tinklai, diferencialinės lygtys, daugiamatės lentelės ir t. t. [11].

Miglotoji aibė – tai išplėstoji klasikinė aibė. Jei  $X$  yra įėjimo erdvė, o  $x$  – tos erdvės elementai, tai miglotoji aibė  $A$  erdvėje  $X$ :

$$A = \{x, \mu_A(x) | x \in X\}; \quad (3)$$

čia  $\mu_A$  –  $x$  priklausomybės aibei  $A$  tikrumo funkcija (7 pav.).





7 pav. Priklausomybės aibei A tikrumo funkcija

Įėjimo erdvės intervale  $[a, b]$  priklausomybės funkcija įgauna nenulinę vertę. 7 pav. pateikta priklausomybės funkcija yra trikampio formos. Tačiau ji gali būti ir trapecijos, Gauso skirstinio, varpo ar kitos formos. Lingvistinės IF..., THAN... taisyklės yra naudojamos sąlyginėms formuluotėms sudaryti (pvz., jei  $x$  yra A, tuomet  $y$  yra B), kur A ir B yra lingvistiniai įverčiai, o  $x$  priklauso miglotai aibei X,  $y$  – miglotajai aibei Y. Pirmoji taisyklės „jei  $x$  yra A, tuomet  $y$  yra B“ dalis vadinama prielaida, antroji – pasekmė arba išvada. [11]. Taisyklės yra sudaromos naudojantis ekspertų patirtimi ir žiniomis. Taisyklėse gali būti naudojamos taip pat loginės operacijos *arba* ir *ne*. Operacija *ir* (aibių sąjunga) realizuojama naudojant *min* (minimizavimo) funkciją, operacija *arba* (aibių sankirta) – naudojant *max* (maksimizavimo) funkciją, o operacija *ne a* (aibės paneigimas) – naudojant  $1-a$ .

Mamdani tipo sistemose išvados miglotos, ir jų miglotąsias aibes daro iš įėjimų miglotųjų aibių *agregavimo* operatorius, t. y. visų taisyklių loginių teiginių darinys į Mamdani miglotąją išvadą, o pastaroji yra determinuojama į konkretų dydį. Šį išvadų darymo metodą 1975 m. pasiūlė E. H. Mamdani [9].

Sugeno išvada taip pat yra miglota, joje kiekvienos taisyklės teiginys yra visų įėjimų tiesinis darinys, o išėjimai - visų taisyklių teiginių tiesinis darinys. Šį metodą 1985 m. pasiūlė M. Sugenas. Pagrindinis Sugeno ir Mamdani išvadų skirtumas tas, kad Sugeno išvados priklausomumo funkcijos yra arba tiesinės įėjimų priklausomumo funkcijų atžvilgiu arba pastovios. Sugeno miglotoji sistema yra kompaktiškesnė ir skaičiavimo prasme efektyvesnė už Mamdani sistemą, ji gerai tinka adaptyviems modeliams sudaryti ir netiesinėms sistemoms modeliuoti. Mamdani sistema turi kitų privalumų. Ji yra labiau intuityvi, geriau tinka žmogaus teikiamiems duomenims, plačiai taikoma praktikoje [9].

Sprendinys arba išvada gaunama išėjimų lingvistinių kintamųjų priklausomumo funkcijų pavidalu. Konkretiems išėjimų dydžiams gauti Mamdani miglotoji sistema paprastai naudoja svorio centro (centroid) arba maksimumo vidurio metodus. Svorio centro metodas yra labiausiai paplitęs. Čia

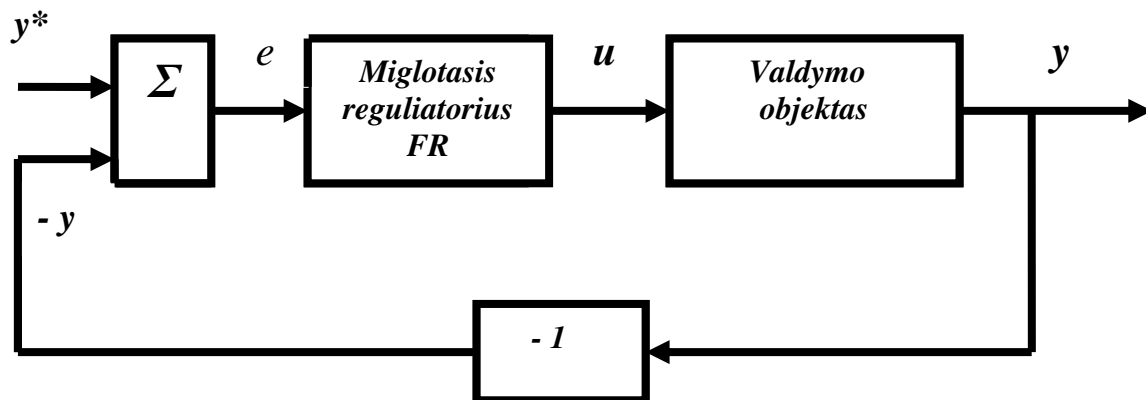
sprendinys yra išėjimų miglotosios aibės priklausomumo funkcijos svorio centras. Maksimumo vidurio metodas taip pat dažnai taikomas praktikoje dėl jų paprastesnio skaičiavimo. Sugėnas miglotosiose sistemose naudoja pasvertų vidurkių ir pasvertų sumų metodus. Miglotieji valdikliai iš esmės priklauso aiškioms valdymo sistemų klasėms ta prasme, kad konkrečius įėjimų kintamuosius dydžius atitinkamai transformuoja į konkrečius išėjimus. Šio tipo miglotoji sistema bendru atveju yra netiesinė beinercinė sistema.

### 1.3.2.1 Miglotosios valdymo sistemos

Miglotojo valdymo sistemos, labiausiai tinka tada, kai miglotumas yra pačiame valdomajame įtaise. Todėl miglotosios sistemos gali būti tada kai, net neaiškus pats valdymo objektas. Miglotosios valdymo sistemos, panašiai, kaip ir klasikinės, skirstomos į šias:

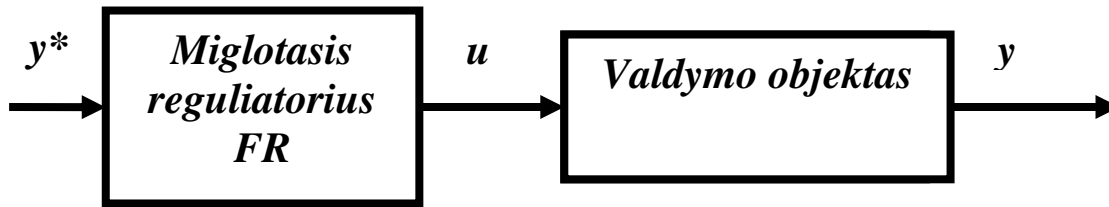
1. Grįžtamojo ryšio valdymo sistemas, kai valdymo poveikis  $u$  valdomajam objektui apskaičiuojamas, atsižvelgiant į valdymo paklaidą  $e$ , kuri yra valdomo objekto išėjimų ir užduoties dydžių skirtumas (8 pav.).

2. Tiesioginio valdymo sistemos, kai valdomo objekto išėjimų dydis  $y$  neturi jokios įtakos apskaičiuojant valdymo poveikį  $u$  (9 pav.)[12].



8 pav. Valdymo sistema su grįžtamojo ryšiu

Čia  $\Sigma$  – sumatorius, FR – miglotasis (fuzzy) reguliatorius,  $e = y^* - y$  – valdymo paklaida,  $y^*$  – užduotis,  $y$  – valdymo objekto išėjimų dydis, t. y. reguliuojamas arba valdomas dydis,  $u$  – valdymo dydis arba valdymo signalas. [9]

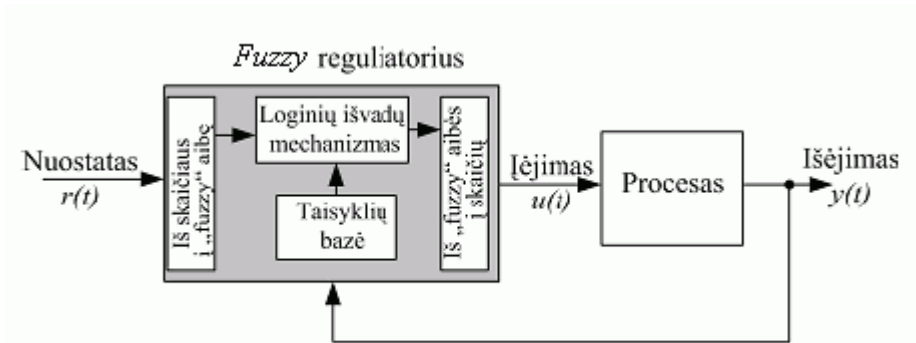


**9 pav.** Tiesioginio valdymo sistema

Čia:  $y^*$  – užduotis,  $y$  – valdomo objekto išėjimo dydis,  $u$  – valdymo dydis

Bendru atveju valdomas objektas gali turėti daug įėjimų ir išėjimų, todėl dydžiai  $e$ ,  $u$  ir  $y$  yra vektoriai. Čia 8 ir 9 paveiksluose keturkampiai vaizduoja sistemos atskiras dalis, pvz. reguliatorių, valdymo objektą ir t. t., o rodyklės rodo kintamųjų dydžių arba signalų veikimo kryptį. Tiesioginio valdymo sistemos naudojamos tada, kai sąryšis tarp valdomo objekto įėjimų  $u$  ir išėjimų  $y$  yra gerai žinomas ir kai objekto išėjimus neveikia vidiniai ir išoriniai trikdžiai. Jų privalumas tas, kad jos yra stabilesnės už grįžtamojo ryšio valdymo sistemas, t. y. jų išėjimų dydis neturi polinkio švytuoti arba stipriai nukrypti į kurią nors pusę. O grįžtamojo ryšio valdymo sistemų privalumas tas, kad grįžtamasis ryšys sumažina visos valdymo sistemos jautrumą išoriniams bei vidiniams trikdžiams, jeigu reguliatoriaus struktūra ir parametrai tinkamai parinkti. Sudėtingose valdymo sistemose grįžtamasis ryšys kai kada naudojamas kartu su tiesioginiu valdymu. Yra daug galimų šių dviejų valdymo principų panaudojimo būdų, tačiau dažniausiai naudojamas valdymo signalų sumavimas. Tai gana paprastas ir efektyvus dviejų valdymo principų panaudojimo būdas.

Pagrindinė valdymo, taikant miglotąją logiką, teigiamybė – galimybė valdyti sudėtingas sistemas, panaudojant žmogiškąją patirtį ir protą, nedalyvaujant žmogui. Žinios panaudotos sukurti tokiam reguliatoriui, kuris valdo sistemą taip, kaip pats geriausias operatorius. Miglotosios logikos reguliatoriaus modelis yra panašus į kitus tradicinių reguliatorių modelius, tačiau išsiskiria iš visų kitų savo įėjimo ir išėjimo parametrų aprašymo struktūra, taip pat ir pačiu įėjimų transformavimo į išėjimus (sprendinio priėmimo) procesu. 10 pav. pateikta proceso valdymo sistema ir miglotosios logikos reguliatoriaus architektūros modelis.



**10 pav.** Proceso valdymo sistema ir miglotosios logikos reguliatoriaus architektūros modelis

Reguliatorių sudaro:

**Taisyklių bazė** (angl. rule base) yra „Jei..., Tada...“ (angl. If...– Then...) taisyklių rinkinys. Taisyklės pagrįstos ekspertų patarimais, kaip tinkamai valdyti sistemą. Loginių išvadų mechanizmas (angl. inference mechanism) išrenka, kurios taisyklės yra tinkamiausios, ir jas pritaiko.

**Miglotuvo sąsaja** (angl. fuzzification interface) reguliatoriaus įėjimo signalus pakeičia taip, kad loginių išvadų mechanizmas galėtų suaktyvinti ir pritaikyti logines taisykles.

**Determinatoriaus sąsaja** (angl. defuzzification interface) keičia logines išvadas į valdomojo proceso įėjimo signalą.

### 1.3.2.2 Miglotosios pateikties priemonės ekspertui

Sprendžiant įvairius projektavimo, planavimo, pirkimo, pardavimo ir valdymo uždavinius, tenka spręsti daugiakriterinius analizės ir vertinimo uždavinius. Šiam tikslui dažnai naudojami trys ekspertiniai metodai: porinių palyginimų, Pareto ir miglotųjų aibių metodai. Porinių palyginimų metodas vartotojui lengviausiai suprantamas. Taikydamas šį metodą, ekspertas privalo paeiliui palyginti dvi alternatyvas ir nuspręsti, kiek viena alternatyva yra geresnė už kitą. Aišku, galimas ir atsakymas „Nežinau“. Pavyzdys, V. Šaltenio ir G. Dzemydos [9] sprendimų priėmimo paramos programinėje priemonėje realizuotas paprastųjų palyginimų metodas, kai ekspertas turi pasakyti, kuri alternatyva geresnė arba „nežinau“. Yra pasiūlyta nemažai metodų, skirtų Pareto sprendinių aibės arba jos dalies radimui. Taikant daugumą metodų, uždavinys yra optimizuoti vieną funkciją. Minėta programinė priemonė padaryta taip, kad ekspertas, keisdamas kriterijų svorius  $w_j$  ( $j=1,k$ ), čia  $k$  – kriterijų skaičius, gali stebėti ne tik maksimizavimo rezultatą  $r_j$ , bet ir kitų alternatyvų, priklausančių Pareto sprendinių aibe, rikiuotę.

Porinių palyginimų ir Pareto paramos priemonės remiasi tiksliais metodais. Miglotųjų dydžių metodas yra artimas Pareto metodui, tačiau leidžia ekspertui papildomai įvertinti savo abejones. Šis metodas pagrįstas miglotųjų skaičių, esant trikampei priklausomumo funkcijai savybėmis. Kiekvienas

normuotas kriterijus  $U_j(X_i)=U_{ji}$  išreiškiamas miglotuoju skaičiumi  $L_j(X_i)=L_{ji}$  esant trikampiai priklausomumo funkcijai. Miglotųjų dydžių būdingas vertes  $L_{ji}^m$  (mažiausia),  $L_{ji}^v$  (vidutinė),  $L_{ji}^d$  (didžiausia) parenka ekspertai, atsižvelgdami į savo abejones. Reikia pažymėti, kad visi normuoti kriterijai  $U_{ji}$  yra intervale  $[0,1]$ , o juos atitinkantys miglotieji dydžiai  $L_{ji}$  yra intervale  $[0,10]$ . Programinė priemonė apskaičiuoja alternatyvų dominavimo virš visų kitų alternatyvų matus  $d_i$ . Ši programinė priemonė pagelbsti ekspertui ir operuoja su miglotaisiais dydžiais, aprašančiais eksperto nuomonę apie tiriamo objekto variantus arba alternatyvas. Pateikties priemonė pateikia sprendimą – atsakymą ekspertui, remdamasi jo paties pateikta miglotąja informacija apie vertinimo kriterijų vertes nagrinėjamoms alternatyvoms ir veiksmams su miglotaisiais skaičiais. Dažnai būna keli ekspertai, todėl ekspertų nuomonės gali skirtis ir tada pateikties priemonė apskaičiuoja vidutinę vertinant dalyvavusių ekspertų nuomonę.

Rengiant įvairius projektus, ekspertinė pateikties priemonė padeda ekspertams įvertinti ruošiamo projekto variantus pagal daugelį kiekybinių ir kokybinių kriterijų. Pavyzdžiui, pagal projekto kainą, įgyvendinimo kainą, eksploatacijos kainą, patikimumą, saugumą, savistovumą ir kita [9].

### *1.3.2.3 Miglotųjų sistemų valdymo privalumai:*

1. Miglotąją logiką lengva suprasti, nes formuluojamos valdymo taisyklės yra lingvistinės.
2. Galima operuoti netiksliais duomenimis.
3. Ji suteikia daugiau alternatyvų pasirenkant valdymo tikslus.
4. Leidžia geriau įvertinti diagnostikos simptomų ir priežasčių dviprasmiškumus.
5. Miglotieji modeliai gali remtis ekspertų sukauptomis žiniomis apie sistemą, jo patirtimi ir netgi intuicija.
6. Daugeliu atvejų miglotieji valdymo modeliai papildo įprastinius.

Žinoma, miglotosios valdymo sistemos turi ir trūkumų – nėra teoriškai apibrėžtas optimalus taisyklių skaičius, nėra aiškos metodikos, kaip pateikti turimas žinias ir patirtį apie objektą ir valdymo sistemą, miglotosios valdymo sistemos suderinimas užima nemažai laiko.

***Miglotojo valdymo sprendimo būdas***, kaip tik *bus tinkamas siekiant užtikrinti nuolatinį interneto ryšį išmaniajame telefone ir norint taupyti šio telefono energetinius resursus*, kadangi kuriamoje sistemoje neturime konkrečių įverčių. Darbo tikslas nėra keisti ar perdaryti bevielio įrenginio įtaisą. Sistemos tikslas yra valdyti procesą, nesigilinant į bevielio įrenginio charakteristikas. Naudojant miglotąjį valdymą bus galima apjungti žmogiškąją patirtį, atsižvelgiant į ekspertų siūlymus ir sukaupytų bandymų rezultatus.

## 1.4 ANALIZĖS METODŲ IR PRIEMONIŲ PARINKIMAS BEI PROGRAMINĖS ĮRANGOS ANALIZĖ

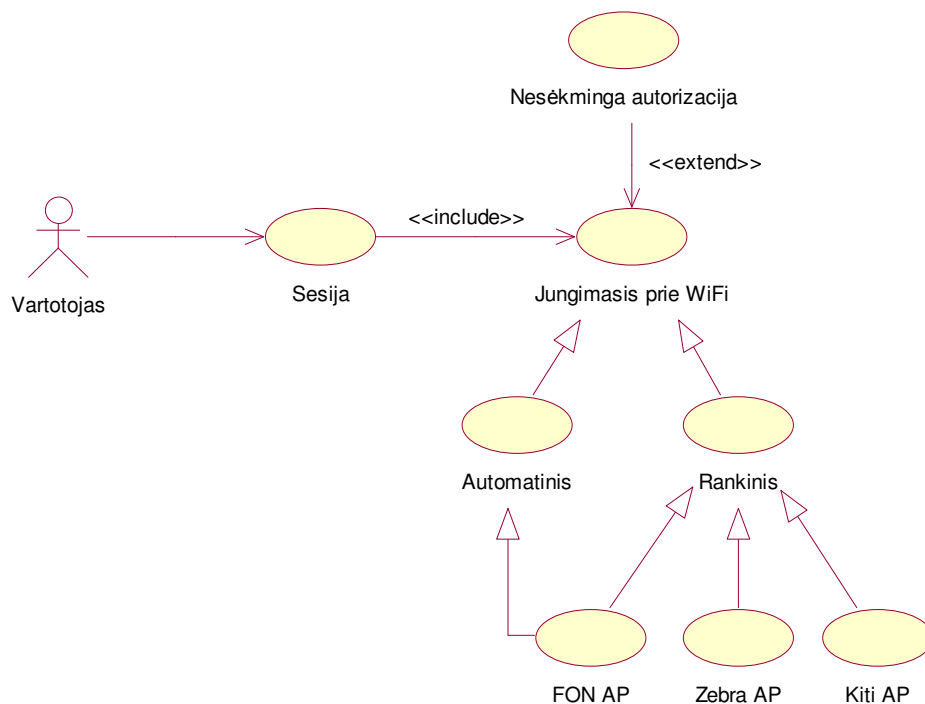
Išnagrinėjus esamas sistemas bei literatūroje siūlomus sprendimus toliau bus pateikiami kuriamos sistemos UML modeliai:

1. Veiklos panaudojimo atvejų modelis.
2. Supaprastinta klasių analizė.

UML modeliavimas naudojamas, kad geriau įsigilinti ir suvokti vartotojų reikalavimus. Struktūrinis projektavimas taikytinas, kai vykstantys procesai bus griežčiau apibrėžti.

- **Veiklos panaudojimo atvejų modelis**

Atsižvelgus į egzistuojančias sistemas bei į Symbian programų kūrimo standartus nubraižomas veiklos panaudojimo atvejų modelis.



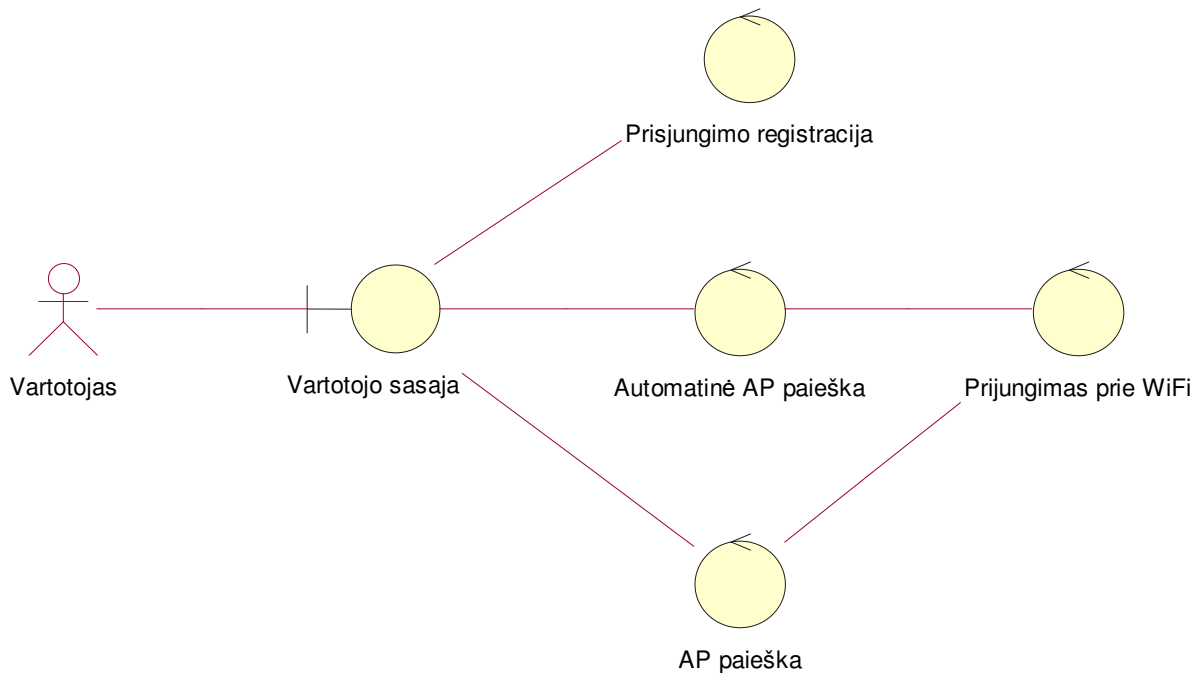
11 pav. Veiklos panaudojimo atvejų diagrama

Diagramoje (11 pav.) aktoriumi pavaizduotas programinės įrangos vartotojas. Kuriama programinė įranga neturi daugiau vartotų grupių. Tai buvo nuspręsta išnagrinėjus egzistuojančias sistemas (jose visose buvo tik viena vartotojų grupė). Be to, „Nokia“ įdiegia į telefonus programinę įrangą, kuri

nėra skirstoma į vartotojų grupes. Todėl šiame darbe kuriama sistema taip pat turės tik vieną - bendrą vartotojų grupę.

- **Supaprastinta klasių analizės diagrama**

Iš veiklos panaudojimų atvejų diagramos išskiriamos viena *boundary* ir keturis *control* tipo klasės. Diagramoje (12 pav.) vaizduojamos šios klasės. Tai supaprastintas modelis, kad paprasčiau suprasti, kaip modeliuoti programinę įrangą, kokios dalys turi didžiausią reikšmę. 12 pav. pavaizduota *boundary* tipo klasę vaizduoja vartotojo sąsajos komponentą, kuris atskirs vartotoją nuo viso valdymo mechanizmo. Kitos *control* klasės demonstruoja svarbiausias kuriamos sistemos dalis, tai yra prieigos taškų suradimas automatinio ir paprasto būdu bei WLAN tinklo sukūrimas. Visas šis modelis rodo, kad vartotojas prisijunginės prie prieigos taškų per vartotojo sąsajos komponentą. Ši diagrama padės nustatyti kuriamos sistemos komponentus, bei sudaryti pilną sistemos struktūrą.



**12 pav.** Supaprastinta klasių analizės diagrama

Toliau pateikiamos įrenginių charakteristikos (telefonų bei bevielio tinklo maršrutizatoriaus savybės), pademonstruotas sistemos sujungimas ir paruošimas darbui.

Dauguma Symbian 9.x telefonų turi integruotus WiFi 802.11b/g adapterius. Šie adapteriai palaiko WEP kodavimą (64 ir 128 bitų), WPA, 802.1x, SIP ir VoIP. Kuriama programinė įranga turi ryšį

su *prieigos taškais* ir *išmaniaisiais telefonais*. Žemiau (13 pav.) pavaizduota keletas išmaniųjų telefonų. Visi telefonai turi Symbian operacinę sistemą su Series 60 platforma.



13 pav. Nokia Series 60 platformos telefonai

Toliau, 14 paveiksle parodytas FON maršrutizatorius. Keletas techninių duomenų apie FON maršrutizatorių pateikti antroje lentelėje.

2 lentelė. FON charakteristikos

<b>Modelis</b>	FON2100
<b>Standartas</b>	IEEE 802.3, IEEE 802.11B, IEEE 802.11G (54Mbit/s)
<b>Kanalų skaičius</b>	11
<b>Portai</b>	Interneto: One 10/100 RJ-45-Ethernet Port & one Power Port
<b>Bevielis</b>	WiFi Protected Access (WPA), WEP, WPA2
<b>Atestacijos</b>	FCC, IC-03, CE, WIFI (802.11b, 802.11g) WPA, WPA2



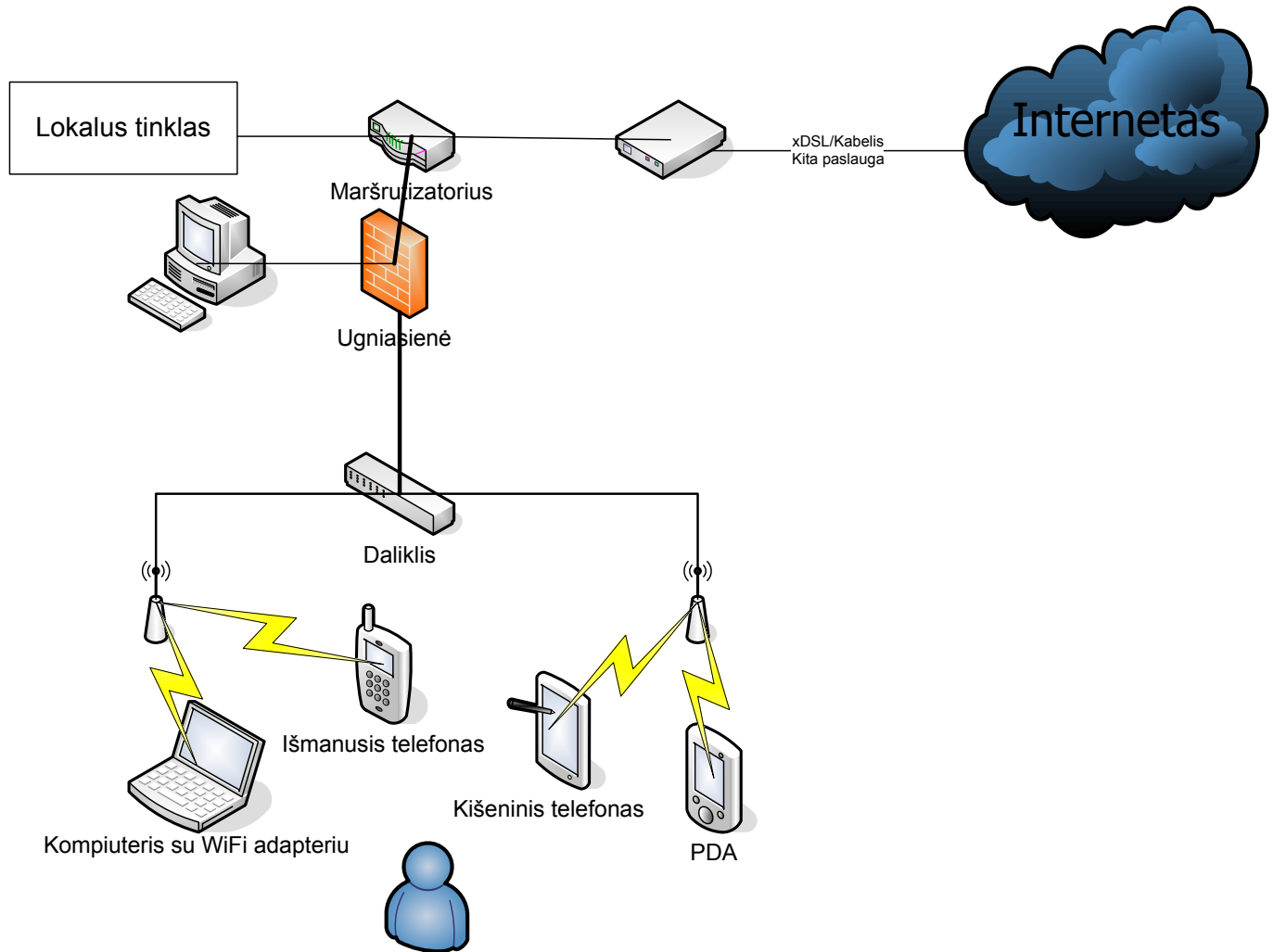
14 pav. FON maršrutizatorius

FON kompanija, savo maršrutizatoriui papildomai sugalvojo keletą „strategijų“. Kai vartotojas užregistruoja savo maršrutizatorių, jis turi galimybę pasirinkti dvi strategijas: „*Linus*“ arba „*Bills*“. Pasirinkus „*Linus*“ reiškia, kad norime dalintis internetu už dyką, bet už tai įgyjama galimybė jungtis už



dyką prie, bet kurio pasaulyje esančio FON maršrutizatoriaus. Pasirinkus *Bills* strategiją galime uždarbiauti, dalinant savo internetu.

Pateiktame 15 paveiksle sumodeliuota pilna tinklo struktūra. Esmė labai paprasta – interneto gavėjas dalinasi savo internetu. FON tipo prieigos taškai yra pajungti prie vartotojo tinklo. Taigi, tokiu būdu yra labai paprasta pasidaryti namų vartotojams potinklius ir šiomis dienomis tokių prijungtų prieigos taškų sparčiai daugėja. *Vartotojai*, dalindamiesi savo internetu per FON maršrutizatorius, **turi galimybę naudotis** kitose vietose **nemokamu internetu**. *Dėl šios priežasties ir atsirado noras sukurti automatinę prieigos taškų „gaudymo“ sistemą.*



15 pav. Tinklo modelis

Sumodeliavus veiklos panaudojimo (11 pav.) diagramą ir atsižvelgus į prieš tai nagrinėtas egzistuojančias sistemas, bei į Symbian OS siūlomus programinės įrangos kūrimo aspektus iš čia yra

suformuojami sistemos vartotojo funkciniai reikalavimai (3 lentelė) ir informacijos sistemos nefunkciniai reikalavimai (4 lentelė).

3 lentelė. **Funkciniai reikalavimai**

Vartotojas	Reikalavimai
<i>Sistemos vartotojas</i>	<ol style="list-style-type: none"> <li>1. Galimybė susikurti prisijungimui prie prieigos taško reikalingą informaciją.</li> <li>2. Galimybė pasirinkti jungimosi strategiją.</li> <li>3. Išsaugomi ir vėliau pasiekiami visi nustatymai ir visa suvesta informacija.</li> </ol>

4 lentelė. **Nefunkciniai reikalavimai**

Kam skirtas reikalavimas	Reikalavimo tipas	Reikalavimas
Techninė, bei programinė įranga	<p>Sudėtis</p> <p>Darbingumas</p>	<p>Numatoma, kad sistema dirbs Symbian trečios kartos operacinėje sistemoje (Series 60). Reikalinga vietos - 1MB. Būtina, kad išmanusis telefonas turėtų WiFi adapterį. Rekomenduojami telefonai yra: Nokia E60, E61, N80, N91, N92, N95.</p> <p>Sistema turi būti parengta dirbti visą parą.</p>
Vartotojo interfeisas (aplinka)	Naudojimas	Sistemos interfeisas turi būti paprastas naudoti, komponentai išdėstyti pagal galiojančias vartotojo interfeiso kūrimo rekomendacijas ir standartus. Vartotojo interfeiso kalba, pagal nutylėjimą – anglų.
Programinės įrangos tobulinimas	Išplečiamumas	Programa turi būti objektiškai suprojektuota ir būti lengvai išplečiama naujomis savybėmis per trumpą laiką.
Duomenys	Saugumas	Sistema turi užtikrinti saugų prisijungimą. Duomenų kriptografija yra reikalinga.

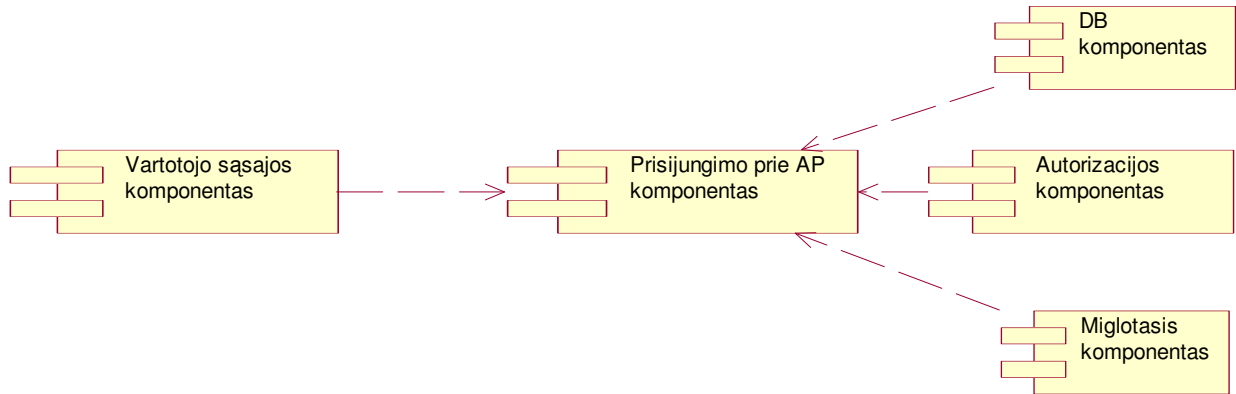
## 1.5 PROJEKTAVIMO METODŲ IR PRIEMONIŲ PARINKIMAS

Šiai sistemai projektuoti pasirinktas metodas yra analogiškas analizės metodams. Sistema bus projektuojama UML modeliais, panaudojant paketą „Rational Rose 2002“, „Microsoft Visio 2003“, bei migloto modelio modeliavimui „FuzzyTech“ programą.

- **Bendroji architektūra**

Žemiau (16 paveiksle) yra pateikta bendroji supaprastinta visos sistemos komponentinė architektūra. Visa sistema dirba Symbian operacinėje sistemoje. Ši sistema savyje turi integruotą prieigos taškų duomenų bazę, iš kurios paimami prieigos taškų vardai. Taip prisijungimo valdiklis paruošia

operacinę sistemą prisijungti prie konkrečių prieigos taškų. Visa sistema bus sudaryta iš penkių komponentų: *Vartotojo sąsajos komponentas (GUI component)*, *Prisijungimo komponentas (FonyEngine component)*, *DB komponentas (DBEngine component)*, *Autorizacijos komponentas (WLANLogin component)*, ir *Miglotuvo komponentas (Fuzzy component)*.



**16 pav.** Komponentinė sistemos architektūra [13]

## 1.6 RIZIKOS FAKTORIŲ ANALIZĖ

Kuriant programinę įrangą gali pasitaikyti, kad visos funkcijos nebus sėkmingai įgyvendintos. Didžiausia rizika yra susijusi su automatinio prieigos taškų paieškos ir prisijungimo mechanizmu, kadangi tokio tipo sistemose nebuvo bandoma sukurti pilnai automatizuoto prisijungimo bei bevielio tinklo paieškos. Tačiau pasitelkus miglotosios logikos sprendimų teoriją, ši problema turi būti paprastai išsprendžiama, kadangi nagrinėjamoji sistema turi tik tris miglotuosius įverčius (Apie juos bus kalbama vėliau). Kitas rizikos faktorius tas, kad sistema yra labai priklausoma nuo ją supančių prieigos taškų. Jei sistemos vartotojas atsidurs tokiaime taške, kuriame nėra pasiekiamų prieigos taškų, aišku, kad sistema negalės pilnai funkcionuoti.

## 1.7 REZULTATO KOKYBĖS KRITERIJAI

Visada yra svarbus galutinis rezultatas. Šiame darbe pagrindiniai kriterijai – pastovaus ryšio kokybės užtikrinimas, išmintingojo telefono resursų mažinimas. Programinė įranga turi užtikrinti, kad kuo mažiau būtų naudojama prieigos taškų paieška ir prisijungimas, kadangi visi šie veiksmai reikalauja nemažų energetinių telefono resursų (elemento tarnavimo laikas mažėja). Dėl šitos priežasties ir yra pasirinktas miglotasis uždavinio sprendimas (tai buvo nuspręsta atlikus literatūros analizę), kuris turi

užtikrinti, kad sistema perieško ir prisijungia prie vis naujo prieigos taško tik tada, kai tikrai to reikia. Telefono elemento „stiprumui“ pamatuoti naudosime savo sukurtą pagalbinę programą. Taipogi, vartotojo sąsaja ir visas programos veikimas turi tenkinti „Symbian Signed“ kriterijus, kadangi visos Symbian programos turi praeiti šios organizacijos patikrinimą.

## **1.8 ANALIZĖS IŠVADOS**

Siekiant Symbian 9.x OS telefonams sukurti ir pademonstruoti veiksmingą programinę įrangą, kurios pagalba būtų galima automatiškai prisijunginėti prie aplinkoje išsidėsčiusių bevielio interneto prieigos taškų, buvo atlikta panašių sistemų analizė. Rinkoje yra naudojamos kelios bevielio interneto prieigos taškų paieškos sistemos. Atlikus jų analizę, pastebėta, kad jos yra nepatogios vartotojams, nesugeba užtikrinti pastovaus interneto ryšio išmaniuosiuose telefonuose ir, nuolat atlikdamos prieigos taškų paiešką, netaupo šių telefonų energetinių resursų. Taip pat pastebėta, kad automatiniai prieigos taškų suradimo ir prisijungimo uždaviniai yra nedaug kur panaudoti. Dauguma žinomų tokio tipo programinės įrangos siūlo tik rankinį ir periodinį valdymą.

Taigi, siekiant sukurti bevielio interneto prieigos taškų paieškos sistemą, tenkinančią šiandieninius išrankaus vartotojo poreikius tokius kaip: patogus valdymas, nuolatinis internetinio ryšio užtikrinimas, telefono energetinių resursų taupymas, buvo apžvelgta literatūroje siūlomi ir galimi pritaikyti minėtos problemos sprendimo būdai. Buvo nagrinėjama objekto ir proceso valdymo sistema, kai objektui valdyti yra naudojamas valdiklis. Kitas būdas netiesinėms sistemoms valdyti šiuo metu vis plačiau taikoma miglotoji logika.

Apžvelgus minėtos problemos sprendimo būdus, buvo pastebėta, kad tinkamiausias būdas, neturint konkrečių įverčių, siekiant užtikrinti nuolatinį interneto ryšį išmaniajame telefone ir norint taupyti šio telefono energetinius resursus, yra miglotojo valdymo taikymas. Darbo tikslas nėra keisti ar perdaryti bevielio įrenginio įtaisą. Sistemos tikslas yra valdyti procesą, nesigilinant į bevielio įrenginio charakteristikas. Naudojant miglotąjį valdymą galima apjungti žmogiškąją patirtį, atsižvelgiant į ekspertų siūlymus ir sukaupų bandymų rezultatus.

Šioje analizės dalyje taip pat nuspręsta, kad projektavimui bus naudojamas UML ir struktūrinis projektavimas, kurių pagalba buvo aprašyti vartotojų reikalavimai ir apibrėžti procesai. Taip pat pateikta bendra visos sistemos veikimo schema, supažindinta su charakteristinėmis numatomų naudoti įrengimų savybėmis bei identifikuoti pagrindiniai naujos sistemos funkciniai reikalavimai, kurie projektiškai bus pateikiami kitoje šio darbo dalyje.

## **2. PROJEKTAVIMAS**

### **2.1 PROJEKTO TIKSLAS**

Projektavimas yra bene vienas sunkiausių sistemos kūrimo etapų. Nuo projektavimo kokybės priklauso tai, kaip lengva bus sistemą realizuoti, kaip sudėtinga bus sistemą pakeisti pasikeitus kliento norams bei kiek sudėtinga sistemą bus prižiūrėti.

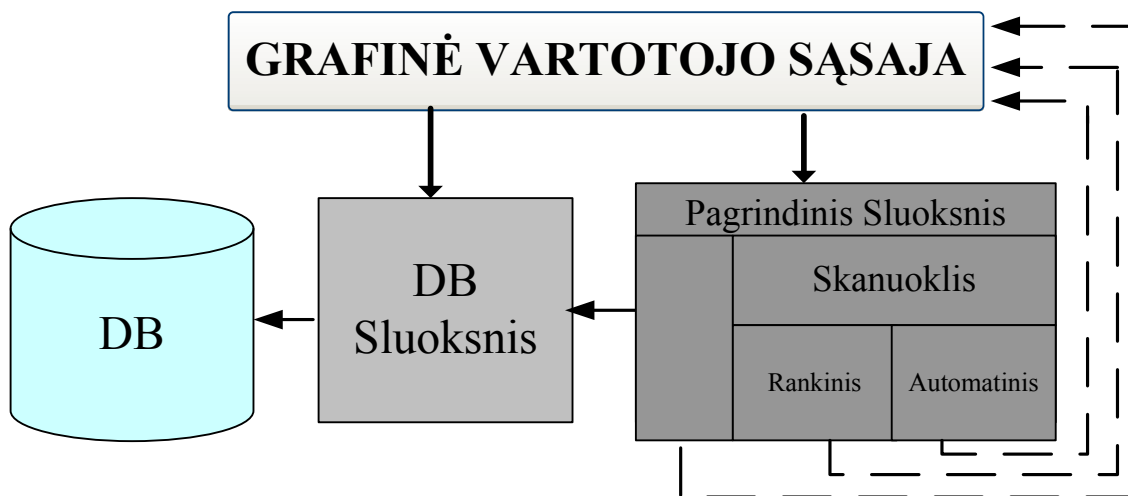
Igyvendinant pagrindinį šio darbo tikslą, t. y. kuriant bevielio interneto prieigos taškų paieškos sistemą (pavadinimu „HandyWi“), sugebančią užtikrinti pastovų internetą mobiliuosiuose telefonuose ir naudojančią minimaliai telefono energetinių resursų, bus sudaryta loginė programos architektūra, nubraižyta sekų, klasių bei veiksmų diagramos komponentams, suprojektuotas miglotasis komponentas ir pateiktas šio komponento veikimo pavyzdys. Bus apžvelgtos galimos realizacijos alternatyvos, taip pat bus suprojektuotas sistemos vartotojo interfeisas ir parengtas testavimo modelis. Visa tai užtikrins kuriamos sistemos kokybę, garantuos valdymo patogumą ir paprastumą.

### **2.2 REIKALAVIMO MODELIS**

Reikalavimų modelyje bus pasiūlyti bei suprojektuoti atskiri komponentai kuriamai programinei įrangai. Komponentai tai programos fragmentai, kurie išskaidyti į atskiras dalis ir yra nepriklausomi vienas nuo kito. Analizės metu buvo nustatyta, kad sistema bus sudaryta iš keturių komponentų. Taigi, atitinkamai sistemos realizacija bus išskaidyta į:

- Programinės įrangos grafinės sąsajos kūrimą.
- Prisijungimo bei paieškos valdymo biblioteką.
- Duomenų talpyklos biblioteką.
- Miglotosios logikos elementų biblioteką.

Žemiau pateiktame paveiksle yra pavaizduotas supaprastintas šių komponentų apjungimas į vientisą pilnai funkcionuojančią sistemą (17 pav.). Diagrama sudaryta iš dviejų sluoksnių: vartotojo sąsajos ir pagalbinių komponentų: miglotosios logikos, duomenų talpyklos ir prisijungimo bei paieškos valdymo komponentų. Tai yra paprastas klasikinis sprendimas, kai vartotojo sąsaja yra atskiriama nuo valdymo logikos. Šiuo principu bus kuriama bevielio interneto prieigos taškų paieškos sistema, kuri galės būti lengvai papildoma naujomis savybėmis, neatliekant didelių pokyčių sistemos struktūroje.



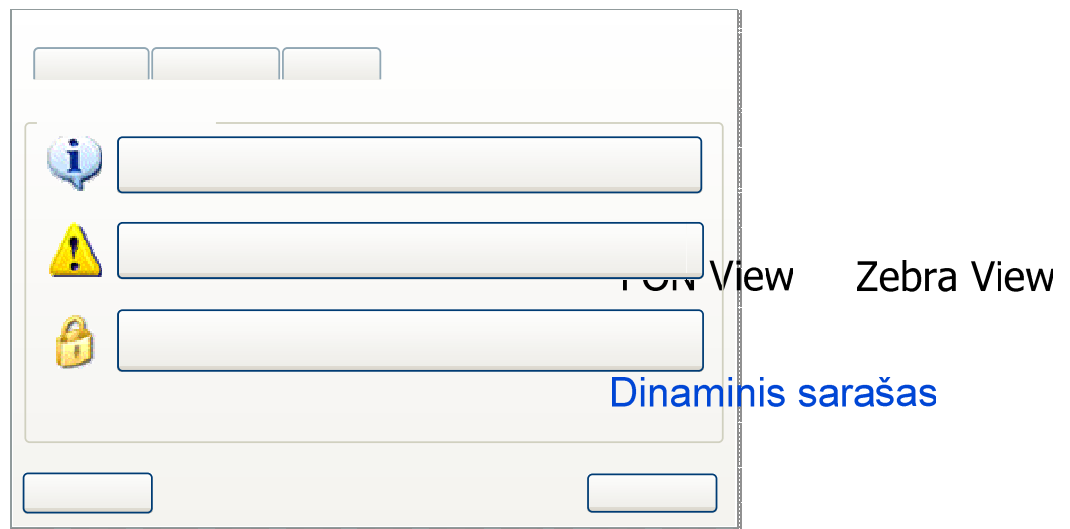
17 pav. Sistemos sluoksninė struktūra

Remiantis projektuojamos sistemos sluoksnine struktūra, toliau bus projektuojami komponentai, nubraižomos jų siūlomos klasių ir įvykių bei sekų diagramos.

### 2.2.1 Programos grafinės sąsajos kūrimas

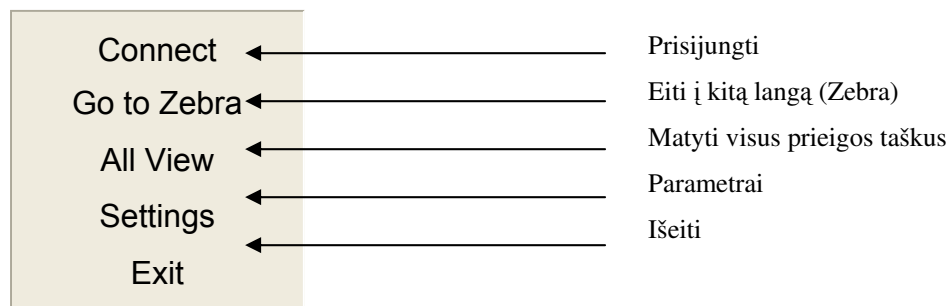
Kuriama sistema turės du pagrindinius langus: rastiems FON prieigos taškams skirtas langas, ir kitas langas apims visus likusius prieigos taškus. Visi šitie langai *CFonyAppView*, *CZebraAppView* bei *CAIAppView* (žr. 18 pav.) yra paveldėti iš bazinės virtualios klasės *CBaseAppView*. *CBaseAppView* klasė turi ryšį su klase *CFONyContainer*. Konteineris tai tarsi panelis, kuriame atliekami visi braižymo darbai, tai yra sąrašo ir jo elementų braižymas. Visos grafinės sąsajos pagrindinė klasė yra *CFONyAppUi*. Šios klasės pagalba yra valdomi visi HandyWi aplikacijoje esantys langai, taip pat ši klasė sukuria *CFonyEngine* klasę, o toliau visiems esantiems langams yra paduodama rodyklė į šią klasę. Papildomi langai yra parametrų langas (*CFONySettingListView*), čia yra nustatomi programos parametrai ir autorizavimo langas (*CFONyAccountView*), kur yra laikoma visa prisijungimui prie prieigos taško reikalinga informacija. Visa programos parametrų ir prisijungimo informacija yra saugoma duomenų bazėje. Paveiksle tai atvaizduota ryšiu į klasę *CDBEngine*.





19 pav. Pagrindinis sistemos langas

Toliau (20 pav.) yra pavaizduotas siūlomas pagrindinis programos meniu.



20 pav. Pagrindinis sistemos meniu

Option

### 2.2.2 Prisijungimo bei paieškos valdymo biblioteka

Prisijungimo ir paieškos biblioteka yra viena iš esminių ir pagrindinių šio darbo tikslų. Šis komponentas apjungs ir naudos kelis kitus komponentus, kadangi šis komponentas turi mokėti ieškoti prieigos taškus prisijungti prie AP bei sukurti WLAN tinklą, tai pat turi gebėti valdyti procesą, tai yra išlaikyti pastovų interneto ryšį. Tam, kad komponentas netaptų labai didelis ir gremėzdiškas jis turės bendrauti su kitais komponentais, autorizacijos, duomenų bazės bei miglotosios logikos komponentais. Žemiau pateiktame paveiksle (žr. 21 pav.) yra pavaizduota prieigos taškų valdymo ir prisijungimo biblioteka. Pagrindinė klasė yra *CFONyEngine*. Šios klasės pagalba yra valdoma visa biblioteka. *CFONyEngine* klasė apjungia klases *CConnector*, *CManualDiscovery*, *CAutoDiscovery*, *CWLANLoginEngine*, *CDbEngine*, *CFONyCertEngine*, *CMd5Factory*.





- *CManualDiscovery* – Ši klasė yra atsakinga už prieigos taškų suradimą. „Manual“ reiškia, kad klasė atlieka paiešką tik tada, kai yra užduodama užklausa iš vartotojo sąsajos, kitaip tariant, paieška atliekama rankiniu būdu. Ši klasė taip pat yra aktyvi, paveldėta iš *CActive* klasės.

- *CAutoDiscovery* – Tai klasė atsakinga už prieigos taškų suradimą. „Auto“ reiškia, kad klasė atlieka paiešką automatiškai. Dėl šios priežasties ir yra sprendžiamas miglotasis uždavinys, kad paieška būtų atliekama tik tada, kai to reikia. Ši klasė taip pat yra aktyvi, paveldėta iš *CActive* klasės.

- *CWLANLoginEngine* – Klasė atlieka autentifikaciją prie FON prieigos taško.

- *CDbEngine* – Ši klasė reprezentuoja duomenų bazės komponentą. Jos pagalba bus galima paimti ir padėti duomenis iš/iš duomenų bazės/ę. Pavyzdžiui, atliekant autentifikaciją yra reikalingas vartotojo vardas ir slaptažodis. Naudodami *CDbEngine* ši informacija yra lengvai prieinama.

- *CFONyCertEngine* – Ši klasė yra atsakinga už saugumą. Norėdami autentifikuotis prie prieigos taško, yra reikalinga perduoti vartotojo vardą bei slaptažodį. Tam, kad nebūtų „pakištas“ apgavikiškas web. puslapis, kuris paimtų įvestą vartotojo vardą ir slaptažodį, yra naudojamas sertifikatas, kurį reikia patvirtinti.

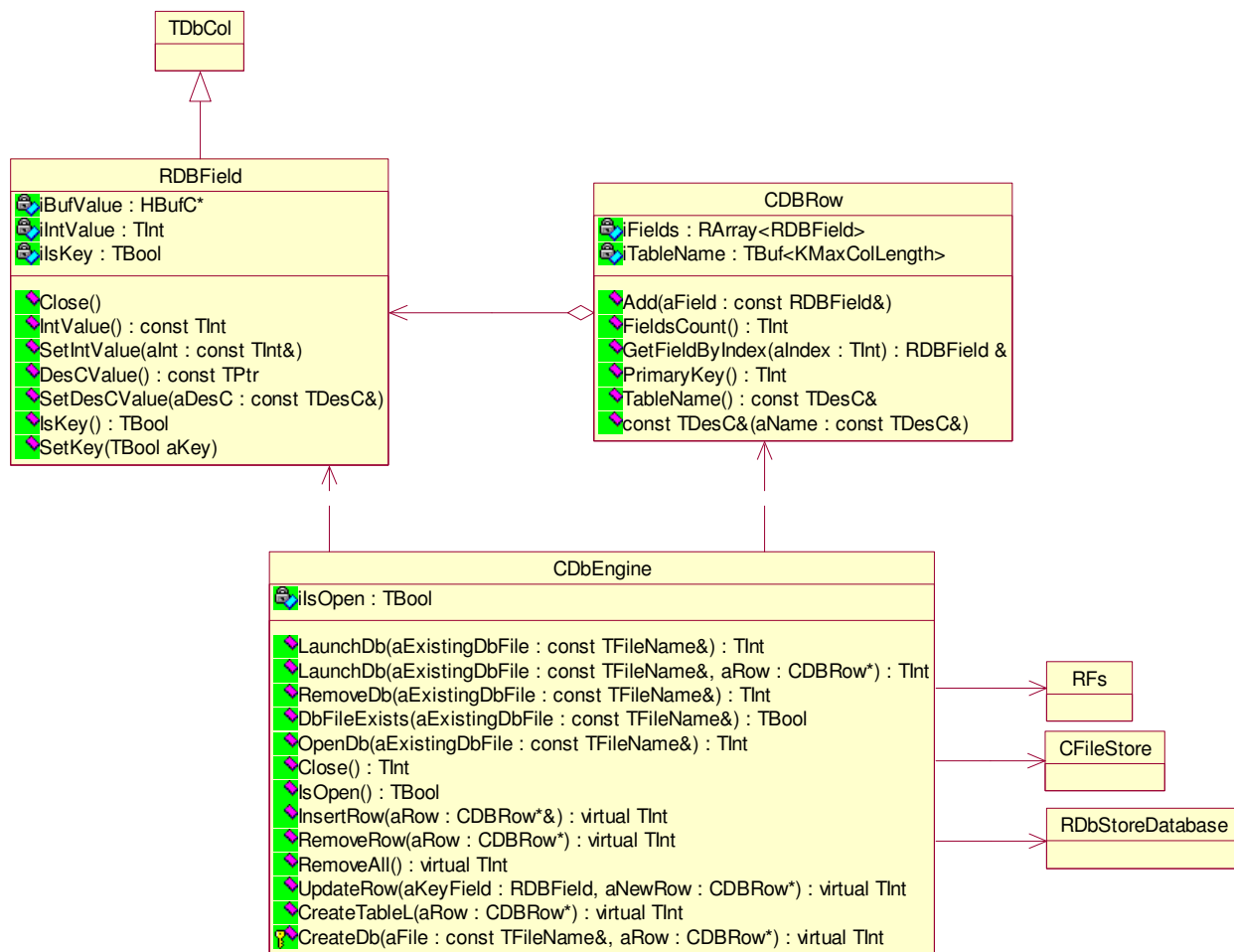
- *CMd5Factory* – Klasė atsakinga už kriptografiją, pavyzdžiui, vartotojo slaptažodis yra užkoduojamas ir laikomas užkoduotas duomenų bazėje. Taip pat MD5 yra naudojamas prieigos taškų sąrašo palyginimui.

### 2.2.3 Duomenų talpyklos biblioteka

Duomenų biblioteka skirta duomenims valdyti. Duomenys turi būti laikomi ir lengvai pasiekiami iš duomenų bazės (žr. 22 pav.). *CDbEngine* klasės pagalba visa tai lengvai galima padaryti. *CDbEngine* klasė turi ryšį su klasėmis *CDBRow* ir *RDBField*.

- *CDBRow* – Atitinka duomenų bazės eilutę. Ši klasė laiko savyje laukų masyvą. Kiekvienas laukelis atitinka tradicinės duomenų bazės stulpelį.

- *RDBField* – Tai klasė, kuri reprezentuoja duomenų bazės stulpelį. Šis stulpelis „geba“ laikyti tik dviejų tipų informaciją: tai skaičius ir teksto eilutės.



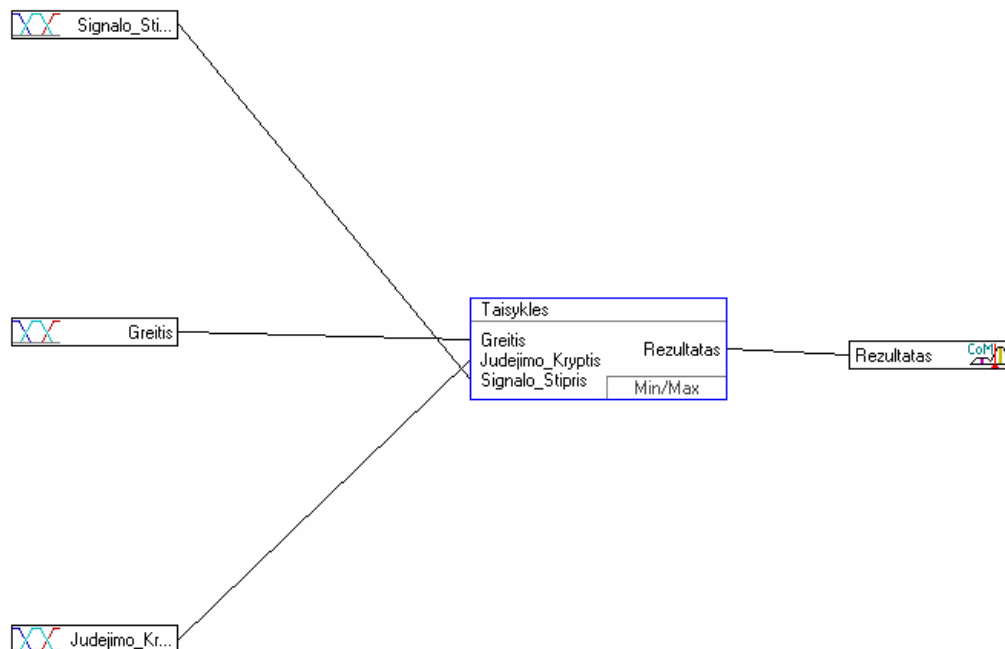
22 pav. Duomenų talpyklos komponento klasių diagrama

## 2.2.4 Miglotosios logikos biblioteka

### Automatinės paieškos sistemos struktūra

Analizės metu buvo išnagrinėta keletas objekto valdymo klausimų sprendimo būdų ir šiam darbui labiausiai tinkančiu buvo pasirinktas miglotasis valdymas. Naudojantis FuzzyTech programa buvo sukurta sistemos struktūra (žr. 23 pav.). Šis valdymas pagrįstas miglotosios logikos principu, kai rezultatas tam tikru būdu formuojamas iš įėjimo reikšmių. Miglotuvas perverčia konkrečius įėjimus į miglotuosius įverčius. Miglotasis išvadų įrenginys sustato reikšmes į taisyklių bloką. Rezultatas yra gaunamas iš taisyklių bloko. Gautosios miglotosios reikšmės yra perverčiamos į konkrečius atsakus.

Žemiau esančiame paveiksle pavaizduota dalis miglotosios bevielio paieškos sistemos struktūros. Ši struktūra apima įėjimų interfeisą, taisyklių bloką ir išėjimų interfeisą.



23 pav. Miglotosios logikos struktūra

Analizės dalyje buvo nustatyta, kad, norit sėkmingai pritaikyti miglotąjį valdymą, reikia surinkti ekspertų komandą, kuri galėtų sudaryti taisyklių bazę. Tai vienas svarbiausių uždavinių, kadangi sėkmingam objekto valdymui toji bazė turi didžiausią įtaką. Kad geriau suprasti WLAN įrenginio elgseną buvo išnagrinėtos esamos sistemos, atlikta daug testavimo bandymų, kurių metu buvo nustatyta, kokie kriterijai svarbiausi įrenginio veikimui. Iš sukaupėtų žinių sistemos ekspertai (Valdemaras Jasaitis, Mindaugas Stonys, Linas Maldutis, Modestas Čepaitis, Vidas Mikulskis ir Algirdas Stonys) nustatė ir apibrėžė tris įverčius, kurie turi įtakos pastovaus interneto išlaikymo uždavinyje. Ketvirtoje lentelėje pateikti šie įverčiai, taip pat ekspertų nustatyti miglotosios logikos termų vardai:

1. „Signalo stiprumas“ - šis įvertis bus žymimas  $S$  raide. Signalo stiprumo svarba akivaizdi: jei bevielio tinklo adapteris yra prisijungęs prie prieigos taško, kurio signalo stiprumas yra „stiprus“, tai sistema bus aprūpinta geru internetu, ir priešingai, jei signalas yra „silpnas“, sistema turi ieškoti naujos alternatyvos, tai yra naujo prieigos taško. Kad paprasčiau būtų skaičiuoti, signalo stiprumo vienetai sunumeruojami intervale  $[0; 100]$ . Nulis reiškia, kad signalo stiprumas yra prasčiausias, o šimtas rodo patį didžiausią signalo stiprumą.

2. „Greitis“ - šis įvertis bus žymimas  $\Delta v$  raide. Ekspertai nustatė, kad kitimo greitis yra ne kas kita, kaip signalo stiprumų skirtumas dabartinio ir prieš tai buvusio laiko momentu  $\Delta v = |S_t - S_{t-1}|$  ir matuojamas taip pat signalo stiprumo vienetais nuo 0 iki 100.

3. „Judėjimo kryptis“ - šis įvertis bus žymimas „ $\Delta k$ “. Aplinkoje yra stebimi dažniausiai ne vieno, o kelių prieigos taškų signalų stiprumai. Tarkime,  $A$  yra jautrumo zonoje esančių prieigos taškų

aibė. Jei bent vienas šios aibės elemento signalo stiprumas pradeda didėti, tai reiškia, kad objektas juda artyn prie kažkokio prieigos taško. Judėjimo krypties vienetai taip pat išreikšti per signalo stiprumą. Judėjimo kryptis parodo signalo stiprumo kitimą dabartiniu ir prieš tai buvusių laiko momentu  $\Delta k = S_i - S_{i-1}$ . Matuojami signalo stiprumo išvestiniais vienetais nuo  $[-60; 60]$ .

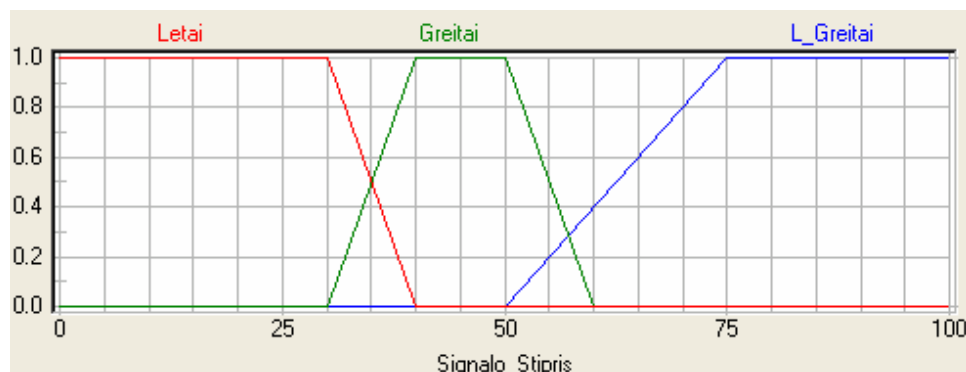
Sistema turi užtikrinti pastovų internetą realiu laiku. Dėl šios priežasties įvedamas taip pat ir laiko žymėjimas. Laikas žymimas raide „t“.

5. lentelė. Įėjimai

#	Kintamųjų Vardai	Matavimo vienetai	Min	Max	Termų Vardai
1	Greitis	Signalų Stiprumas	0	100	Lėtai Greitai Labai Greitai
2	Judėjimo Kryptis	Signalų Stiprumas	-60	60	Nuo Nejuda Tolyn
3	Signalų Stiprumas	Signalų Stiprumas	0	100	Silpnas Vidutinis Geras L_Geras

#### ➤ Greičio termai

Greičio aibė yra padalinta į tris terminus. Ši skaičių nustatė ekspertai, atlikdami bei tyrinėdami signalo stiprumo kitimus, judant nuo maršrutizatoriaus ir link maršrutizatoriaus. Pirmasis terminas - „Lėtai“. Tikrumo funkcijos reikšmė yra lygi vienetui intervale nuo  $S_i \in [0;30]$  (24 pav.), nuliui tampa, kai  $S_i \geq 40$ . Antrasis terminas vadinasi „Greitai“. Tikrumo funkcijos reikšmė yra lygi vienetui intervale nuo  $S_i \in [40;50]$ ; nuliui tampa, kai  $S_i \notin (30;60)$ . Trečiasis terminas vardas „L\_Greitai“. Tikrumo funkcijos reikšmė yra lygi vienetui intervale nuo  $S_i \in [75;100]$ ; nuliui tampa, kai  $S_i \leq 50$ .

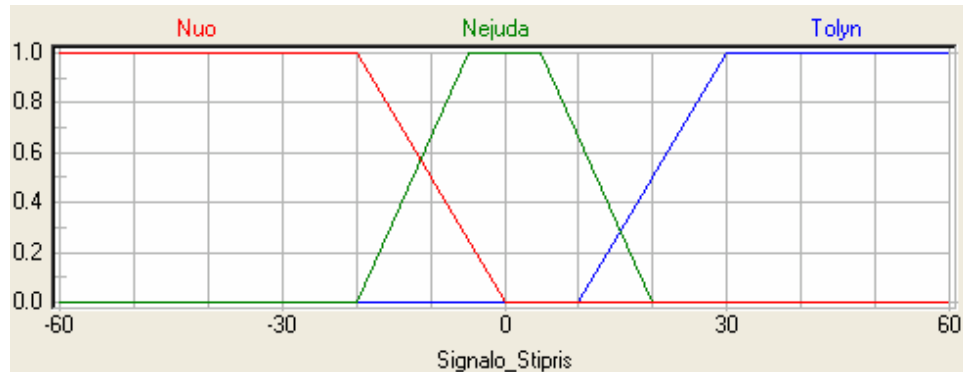


24 pav. Įvertis „Greitis“

#### ➤ Judėjimo krypties termai

Judėjimo krypties aibė yra padalinta į tris terminus. Ši skaičių nustatė ekspertai, atlikdami bei tyrinėdami signalo stiprumo kitimus, judant nuo maršrutizatoriaus ir link maršrutizatoriaus. Pirmasis

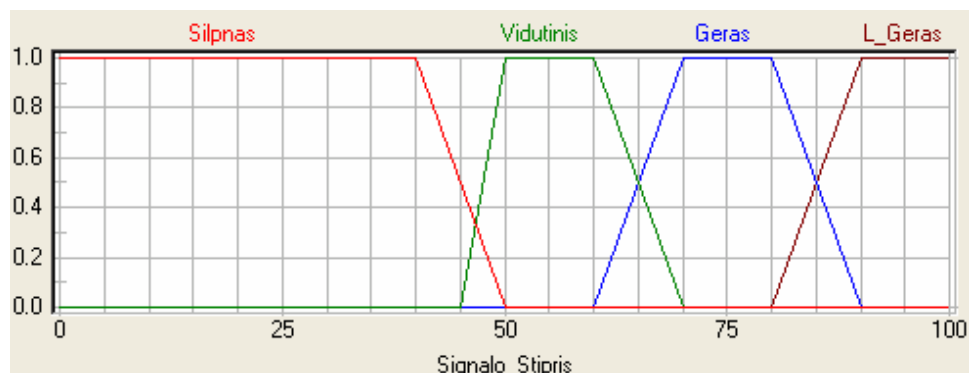
termas pavadintas „Nuo“ vardu (25 pav.). Tikrumo funkcijos reikšmė yra lygi vienetui intervale nuo  $S_i \in [-60; -20]$ ; nuliui tampa, kai  $S_i \geq 0$ . Antrasis terminas pavadintas „Nejuda“ vardu. Tikrumo funkcijos reikšmė yra lygi vienetui intervale nuo  $S_i \in [-5; 5]$ , nuliui tampa, kai  $S_i \notin (-20; 20)$ . Trečiasis terminas pavadintas „Tolyn“ vardu. Tikrumo funkcijos reikšmė yra lygi vienetui intervale nuo  $S_i \in [30; 60]$ ; nuliui tampa, kai  $S_i \leq 10$ .



25 pav. Įvertis „Judėjimo kryptis“

#### ➤ *Signalų stiprumo termų*

Signalų stiprumo aibė yra padalinta į keturis terminus. Terminų skaičių ir reikšmes nustatė ekspertai; taip pat buvo atsižvelgta į esančių sistemų signalų stiprumo traktavimo skalę kitose sistemose. Pirmo termino vardas yra „Silpnas“. Tikrumo funkcijos reikšmė tampa lygi vienetui intervale nuo  $S_i \in [0; 40]$ ; nuliui tampa, kai  $S_i \geq 50$ . Antrasis terminas pavadintas „Vidutinis“ vardu. Tikrumo funkcijos reikšmė yra lygi vienetui intervale nuo  $S_i \in [50; 60]$ ; nuliui tampa, kai  $S_i \notin (45; 70)$ . Trečiasis terminas pavadintas „Geras“ vardu. Tikrumo funkcijos reikšmė yra lygi vienetui intervale nuo  $S_i \in [70; 80]$ ; nuliui tampa, kai  $S_i \notin (60; 90)$ . Ketvirtąjį terminą vadiname „L\_Geras“. Tikrumo funkcijos reikšmė yra lygi vienetui intervale nuo  $S_i \in [90; 100]$ ; nuliui tampa, kai  $S_i \leq 80$ .

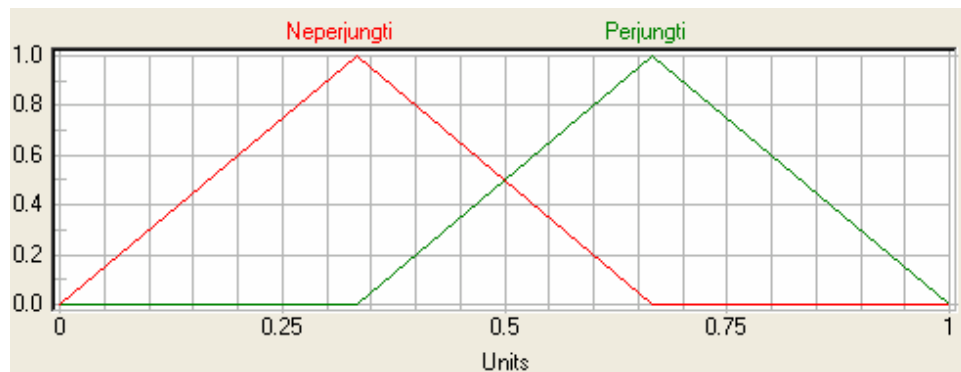


26 pav. Įvertis „Signalų stiprumas“

### ✍ Išėjimo rezultatas

Išėjimų rezultatai yra formuojami iš taisyklių bazės rinkinio. Gauti rezultatai padengia kiekvieno termo reikšmę pagal tikrumo funkcijos rezultatą. Kaip buvo rašyta analizės dalyje termų padengimas priklauso nuo pasirinkto metodo, analizės metu buvo aprašyti du metodai: Mamdani ir Sugeno. Termų plotui padengti (išvados daryti) naudojamas Mamdani metodas. Svorio centro išėjimų termai pavaizduoti 28 pav. Svorio centrui rasti panaudojama masės centro formulė [14].

$$x_s = \frac{\int_a^b xQ(x)dx}{\int_a^b Q(x)dx} \quad (6)$$



**27 pav.** Išėjimų rezultatų termai

Toliau pateikiamos miglotojo komponento veikimo pavyzdys. Tarkime, kad ekspertai užrašė šias taisykles į taisyklių bazę:

IF (<S<sub>L\_Geras</sub>> & <Δv<sub>Lėtai</sub>> & <Δk<sub>Nejuda</sub>>) | (<S<sub>L\_Geras</sub>> & <Δv<sub>Greitai</sub>> & <Δk<sub>Artėja</sub>>)

THEN <Neperjungti>

IF <S<sub>Vidutinis</sub>> & <Δv<sub>Greitai</sub>> & <Δk<sub>Tolėja</sub>>

THEN <Perjungti>

Tarkime, kad į sistemą pateko tokie įverčiai:

S<sub>L\_Geras</sub> = 0.7; S<sub>Vidutinis</sub> = 0;

Δv<sub>Lėtai</sub> = 0; Δv<sub>Greitai</sub> = 1;

Δk<sub>Nejuda</sub> = 0.2; Δk<sub>Artėja</sub> = 0.8; Δk<sub>Tolėja</sub> = 0;

Pirmoji sąlyga perrašoma taip: IF  $a \mid b$  THEN  $\langle \text{Neperjungti} \rangle$ , kur  $a = \langle S_{L\_Geras} \rangle \& \langle \Delta v_{Lėtai} \rangle \& \langle \Delta k_{Nejuda} \rangle$  ir  $b = \langle S_{L\_Geras} \rangle \& \langle \Delta v_{Greitai} \rangle \& \langle \Delta k_{Artėja} \rangle$ . Tada įstačius reikšmes gaunama:

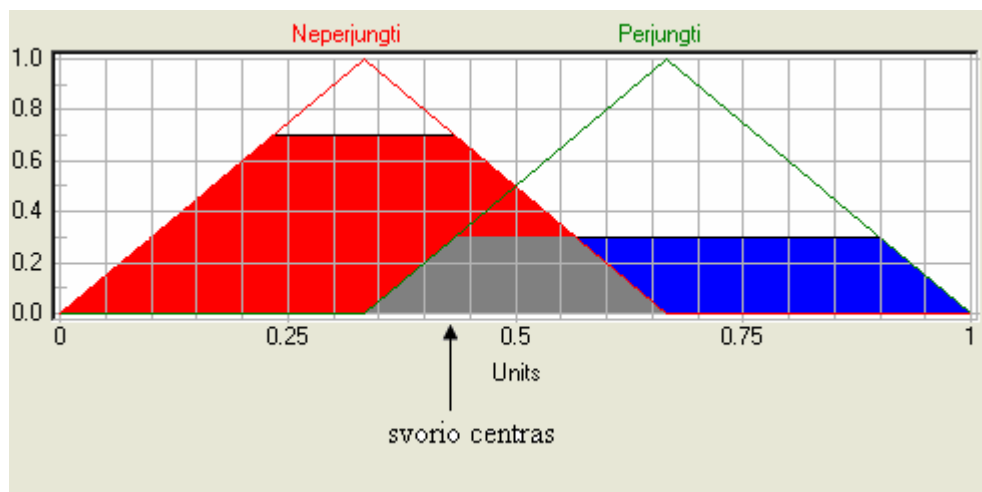
$$a = \text{Min}(0.7, 0, 0.2) = 0;$$

$$b = \text{Min}(0.7, 1, 0.8) = 0.7;$$

$$\text{Max}(0, 0.7) = 0.7$$

Tikrumo funkcijos gavimas pavaizduotas 28 grafike. Rezultate yra tik du termai: „neperjungti“ ir „perjungti“. Raudona spalva padengtas plotas rodo „neperjungti“ termo svorį, gautas iš nagrinėjamo pavyzdžio. Mėlyna spalva žymi – „perjungti“ termo svorį, o pilka – bendrą persidengiantį plotą. Naudojantis svorio centro formule (6) surandamas svorio centras, kuris ir yra tikrumo funkcijos galutinis atsakymas.

Taigi, kad nereikia atlikti paieškos, tikrumo funkcija yra lygi 0.7, tai daugiau už 0.5, todėl sistema šiuo atveju neatliktų paieškos. Kaip jau buvo minėta pavyzdys patvirtina, kad taisyklių bazė atlieka svarbiausią vaidmenį šiame uždavinio sprendime, tai yra šio valdiklio pagrindas.



28 pav. Rezultato pavyzdys

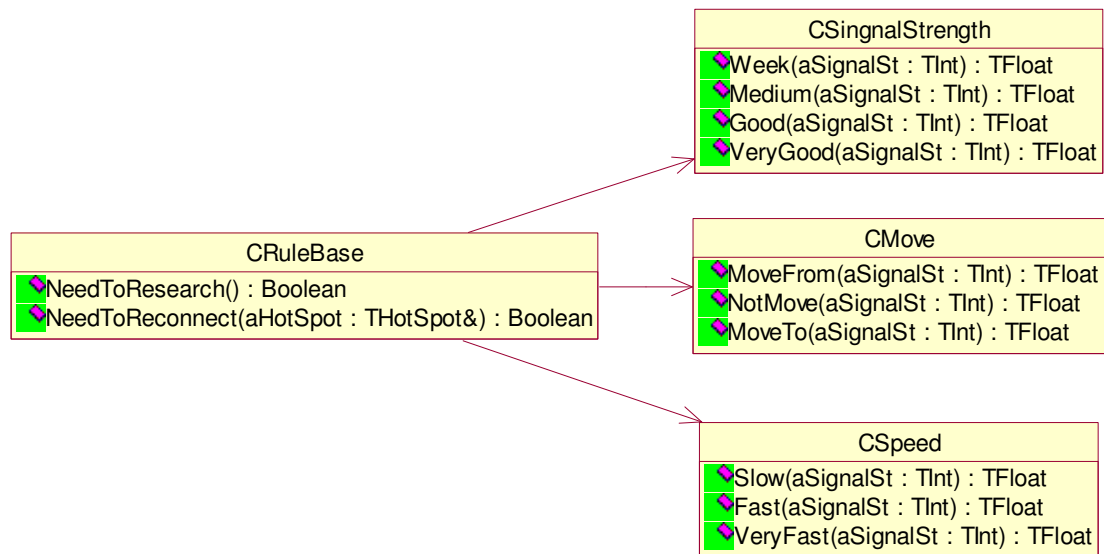
✎ *Migloto komponento klasių diagrama.*

Šis komponentas atsako į klausimą ar reikia naujai ieškoti prieigos taškus, ar tai nėra būtina. Pagrindinės klasės vardas yra *CRuleBase* (žr.29 pav.), kuri turi tris pagalbines klases *CSignalStrength*, *CMove* ir *CSpeed*.

- *CSignalStrength* – Gražina visų termų („Silpnas“, „Vidutinis“, „Geras“ ir „L\_Geras“) tikrumo funkcijų  $\mu(x)$  įverčius.
- *CMove* – Gražina visų termų („Nuo“, „Nejuda“ ir „Tolyn“) tikrumo funkcijų  $\mu(x)$  įverčius.



- *CSpeed* – Grąžina visų termų („Letai“, „Greitai“ ir „L\_Greitai“) tikrumo funkcijų  $\mu(x)$  įverčius.



29 pav. Migloto komponento klasių diagrama

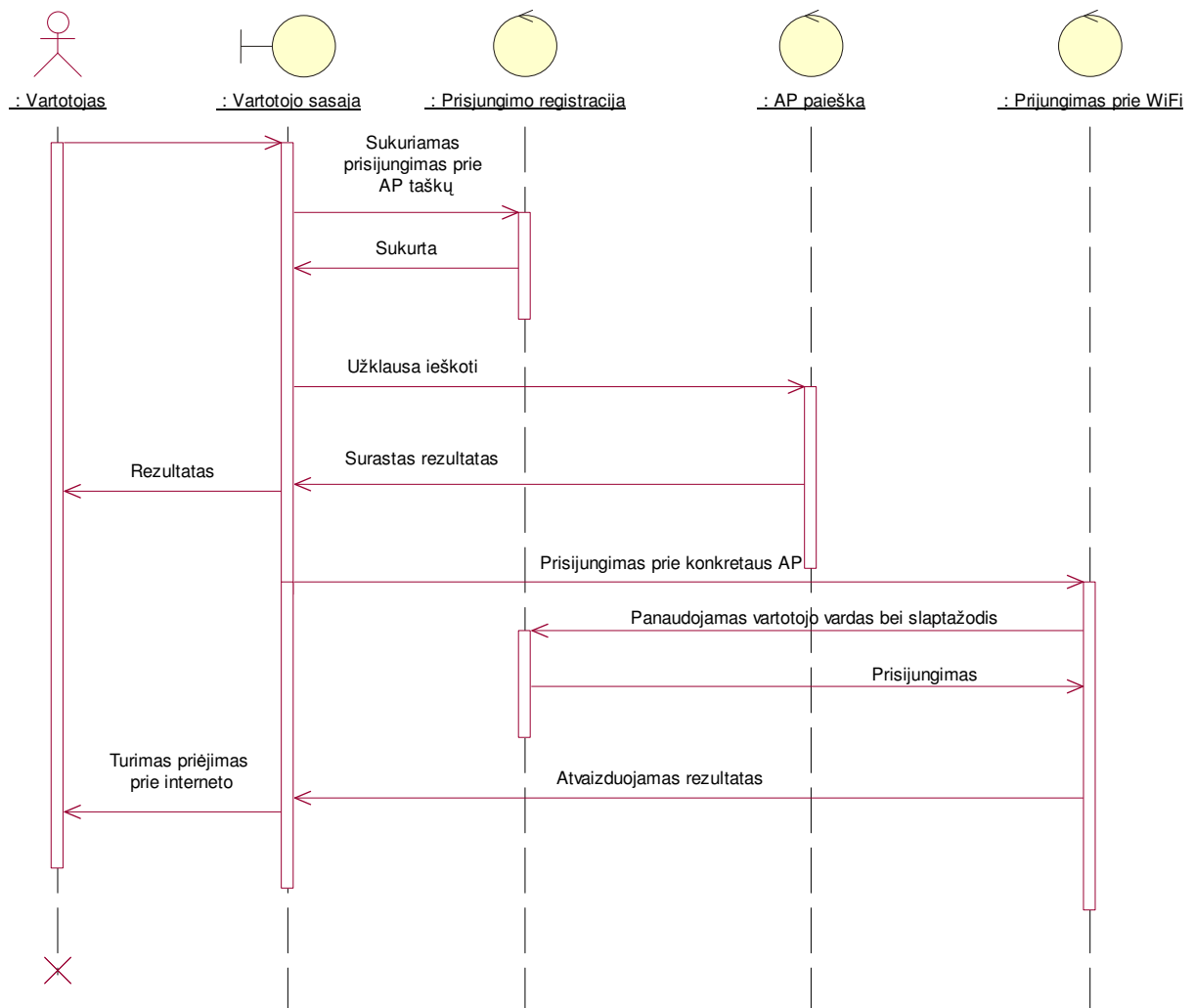
### 2.2.5 Sistemos saugumas ir suderinamumas su kitomis sistemomis

Projektuojant sistemą ir jos komponentus, reikia būtinai atsižvelgti ir į sistemos saugumą bei suderinamumą su kitomis sistemomis. Potencialios saugumo spragos galėtų būti negalėjimas naudoti MD5 hešavimo algoritmo vartotojo slaptažodžiui saugoti, bei vartotojo vardo bei slaptažodžio siuntimas maršrutizatoriui. Priežastis paprasta: užkodavę su MD5 nebežinomas originalus vartotojo slaptažodis. Dėl šios priežasties pasitelkiamas Base64 kodavimo/dekodavimo algoritmas. Instaliavus programą kartu su sistema bus įdegtas ir FON sertifikatas. Sistemai prisijungus prie maršrutizatoriaus, jei bus atsiųstas tas pats FON sertifikatas, taip sistema žinos, kad ši informacija atkeliavo tikrai iš maršrutizatoriaus, ir tai užtikrina, kad vartotojo vardas bei slaptažodis bus tikrai nusiusti į reikimą vietą, tai yra į maršrutizatorių. Duomenų siuntimas vyksta per HTTPS protokolą. Duomenų siuntimo saugumą per WiFi užtikrina IEEE 802.11g standartas.

Suderinamumo problemos gali kilti dėl WiFi adapterio, kadangi kitos programos gali norėti lygiagrečiai naudotis adapteriu taip pat. Tačiau Symbian OS yra gerai apgalvota dėl stabilumo, todėl visi „sunkūs“ veiksmai tokie, kaip prieigos taškų paieška, atliekami atskirame procese. Kad sistema stabili ir patikimai dirba Symbian 9.x operacinėje sistemoje, rodo atliktas testas „Symbian Signed“ organizacijoje.

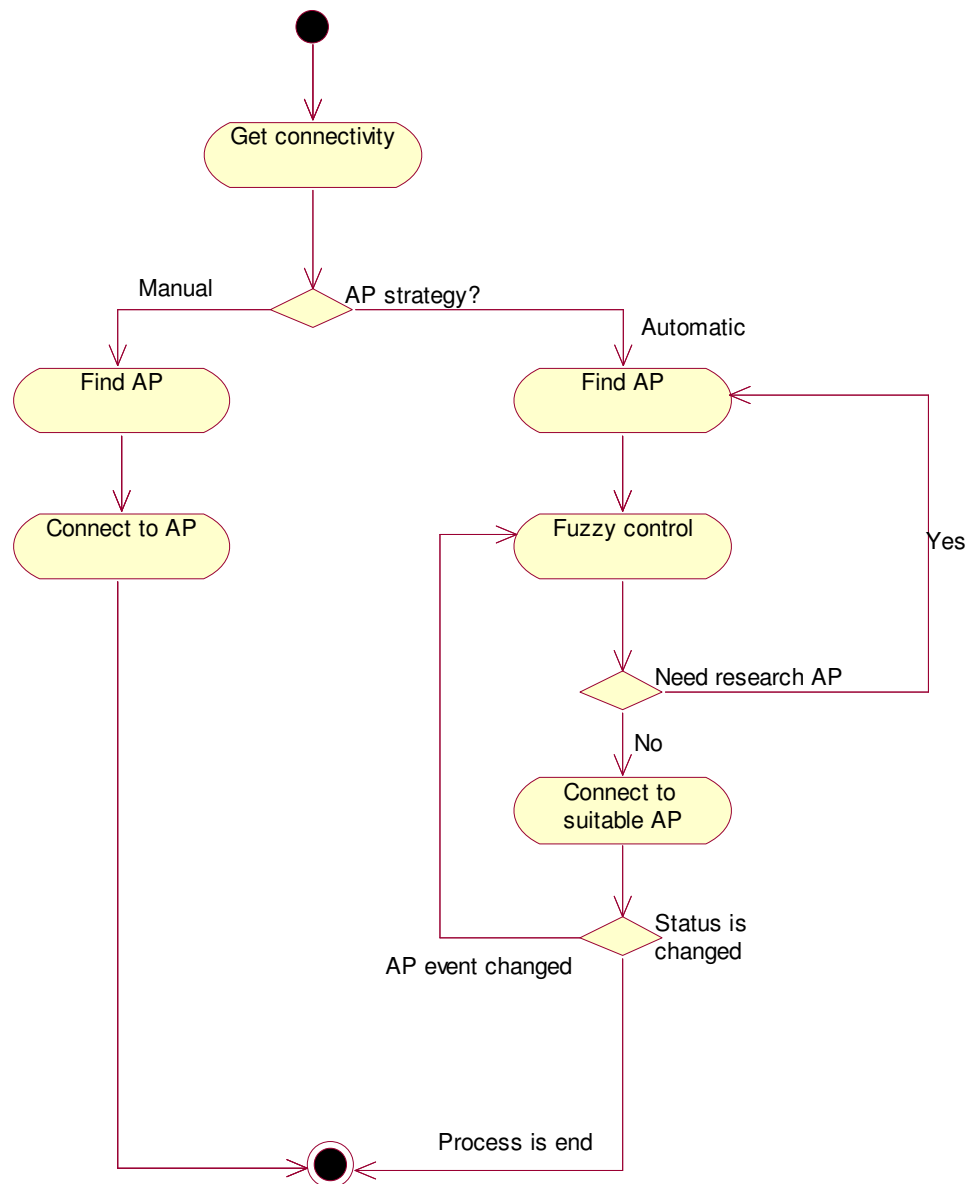
## 2.3 SISTEMOS PROJEKTAS

Praeitoje dalyje buvo projektuojami atskiri komponentai. Šio poskyrio tikslas yra sujungti komponentus į vientisai veikiančią sistemą, kad geriau būtų galima perprasti kuriamos sistemos struktūrą, jos veikimą bei suvokti komponentų tarpusavio priklausomybę. Tuo tikslu buvo panaudota sekų diagrama (30 pav.), kuri yra sudaryta iš abstrakčių klasių diagramos (12 pav.). Visos šios abstrakčios klasės tapo atskirais programos komponentais, kurie padeda geriau suprasti bei reikalui esant lengviau papildyti kuriamą sistemą naujomis savybėmis. Iš 30 paveikslo matyti, kad vartotojas bendrauja tik per vartotojo sąsają, o toliau vartotojo sąsaja naudoja pagalbinius komponentus, kurie suranda prieigos taškus bei teikia prisijungimą prie jų.



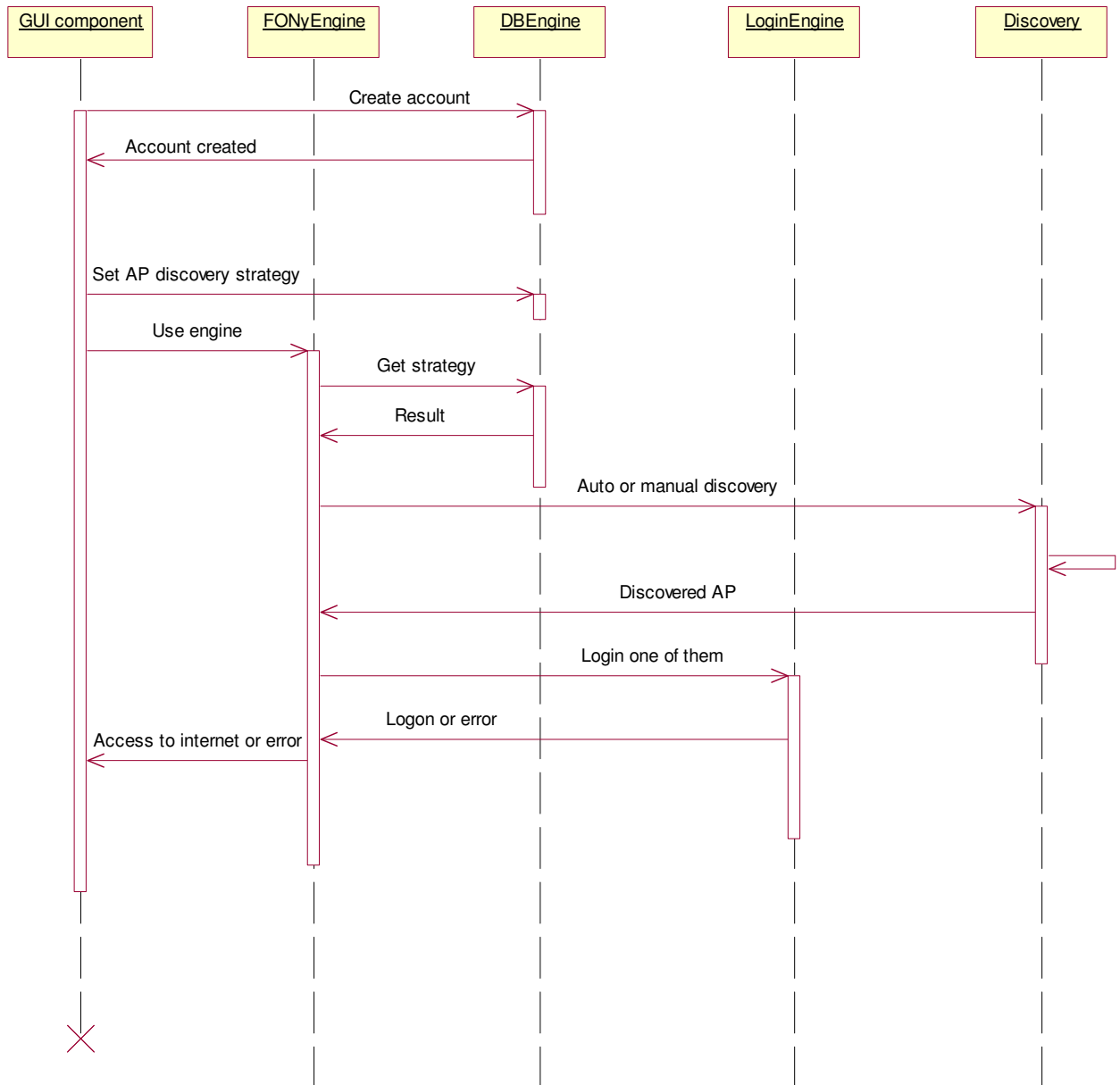
30 pav. Sistemos sekų diagrama

31 paveiksle pavaizduota veiklos diagrama, kuri vaizduoja prieigos taškų paieškas ir prisijungimus prie rastų AP. Taip pat, ši diagrama vaizduoja konkrečius veiksmus bei veiklas, kai sistema ieško prieigos taškų. Prieigos taškai gali būti ieškomi rankiniu arba automatinio būdu. Vykdamas paiešką rankiniu būdu, sistemos vartotojas pats sprendžia prie kurio prieigos taško prisijungti, o automatinio būdu yra valdomas procesas t.y. sistema pati ieško prieigos taškų ir prisijunginėja prie jų. Ši veiklos diagrama apima *CDiscovery*, *CManualDiscovery* bei *CAutoDiscovery* klases (žiūrėti į 21 pav.). Diagrama, parodė, kad miglotasis komponentas turės ryšį su *CAutoDiscovery* klase.



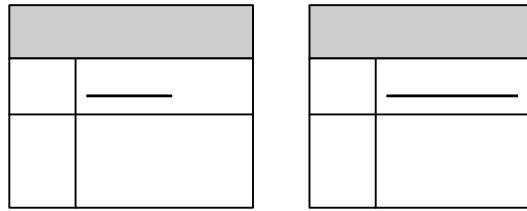
**31 pav.** Veiklos diagrama apjungianti automatinę paiešką

Nubraižius abstrakčią visos sistemos sekų diagramą bei pagrindinės veiklos diagramą, toliau sujungiami komponentai ir nubraižoma jiems sekų diagrama (32 pav.), kuri atvaizduoja vykdomos programos eiliškumą bei komponentų priklausomybę vienas nuo kito.



**32 pav.** Komponentų sekų diagrama

Duomenų bazė yra labai paprasta ir sudaryta tik iš dviejų lentelių, kurios neturi jokio ryšio viena su kita. Lentelėje *USERS* yra saugoma vartotojo prisijungimo informacija prie prieigos taško (33 pav.).



33 pav. Sistemos duomenų bazė

USERS

PK UserID

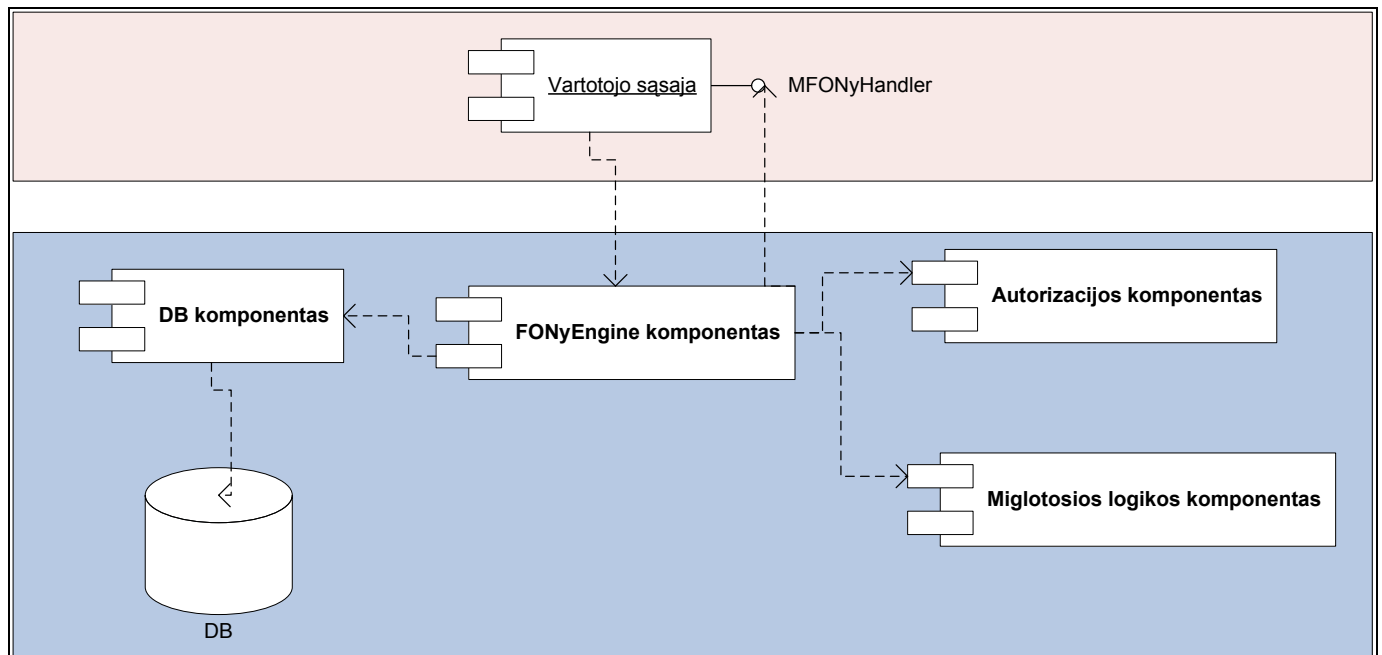
User\_Name

Password

Lentelėje SETTINGS - saugomos programos būsenos, tai yra vartotojo nustatyti programos parametrai (pavyzdžiui, ar sistema turi dirbti automatinio paieškos režimu, ar tiesiog paprastu - tai yra rankiniu). Kaip buvo pavaizduota ankstesnėse diagramose, duomenų bazės komponentas turi tiesioginį priėjimą prie pačios duomenų bazės, o kiti komponentai, kaip ir vartotojo sąsajos komponentas, turi tik duomenų bazės komponento teikiamais metodais - gauti ir patalpinti duomenis į duomenų bazę. Duomenų lentelių laukai išvardinti 6 lentelėje.

6. lentelė. Duomenų bazės laukai

Duomenų lentelės vardas	Stulpelių vardai	Paaiškinimas
USERS	UserID User_Name Password	Pirminis raktas Prisijungimo vardas Slaptažodis
SETTINGS	SettingsID Strategy Timer	Pirminis raktas Prisijungimo tipas: rankinis (reikšmė 0), automatinis (reikšmė 1). Taimeris skirtas, kad sistema neradusi tinkamo AP padarytu pertraukus ir nuolat neatliktu pakartotinių paieškų.



34 pav. Pilna sistemos struktūra

34 paveiksle pavaizduotas pilnas sistemos modelis. Diagramoje išryškinti du sluoksniai, vienas yra vartotojo sąsaja, o kitas - taip vadinamas „variklis“. Per vartotojo sąsaja yra užduodamos komandos visai sistemai. Sistemos atsakymai ir būsenų pasikeitimai vartotojo sąsajai yra atsakomi per *MFONyHandler* interfeisą.

## 2.4 TESTAVIMAS

Įsitikinti, kad programa veikia teisingai, programuotojas ar testavimo komanda gali nagrinėdami programos kodą, taip iš įeinančių duomenų sprendami kokius išėinančius duomenis programos kodas sugeneruos. Taigi, įrodymas, kad programos kodas veikia gali net nereikalausti programos paleidimo. Tačiau tuo klausimu vartotojai turi kitą požiūrį, kad įrodyti vartotojui, jog komponentas veikia korektiškai, reikia pademonstruoti aibę eksperimentų, kurie parodytų sistemos funkcionavimą bei teisingą veikimą esant įvairioms sąlygoms.

Norint pilnai ištestuoti prieigos taškų paieškos sistemą, testavimą reikia suskirstyti į dvi dalis:

- komponentų testavimas;
- vartotojo sąsajos testavimas;

Komponentų testavimas parodys, kaip veikia projektavimo dalyje aprašyti bei pavaizduoti skirtingi sistemos komponentai. Projektuojant ir programuojant komponentus, kiekvienam komponento metodui yra parašomas analogiškas testavimo metodas, kurio paskirtis yra patikrinti metodo veikimą. Toks testavimas literatūroje yra vadinamas „baltos dėžės“ principu. „Baltos dėžės“ principas sako: „Testiniai atvejai gaunami iš programos struktūros. Žinios apie programą naudojamos nustatyti papildomus testinius atvejus. Tikslas yra išbandyti visus programos operatorius ir galimus skaičiavimo kelius.“ [15]. Būtent todėl ir yra rašomi testiniai metodai, kurie turi testuoti ir patikrinti egzistuojančių metodų veikimą. Kai sistemoje yra parašyta daug testinių metodų, jų veikimas ir modeliavimas turi būti centralizuotas ir valdomas specialios platformos, tai yra visi testavimo metodai turi būti iškviečiami reikiama tvarka, tam tikslui yra naudojamas SymbianOSUnit platforma. Ši platforma perdaryta ir pritaikyta iš gerai žinomos CppUnit platformos.

Vartotojo sąsajos testavimas yra atliekamas ne automatizuotai, tai yra visa testavimą atlieka žmogus. Testuotojo darbas yra aplankyti visus dialogus bei stebėti sistemos darbą. Tuo tikslu, yra parašyti „use case“ modeliai, kuriuos reikia patikrinti, kadangi sistema turės praeiti Symbian Signed testavimą.

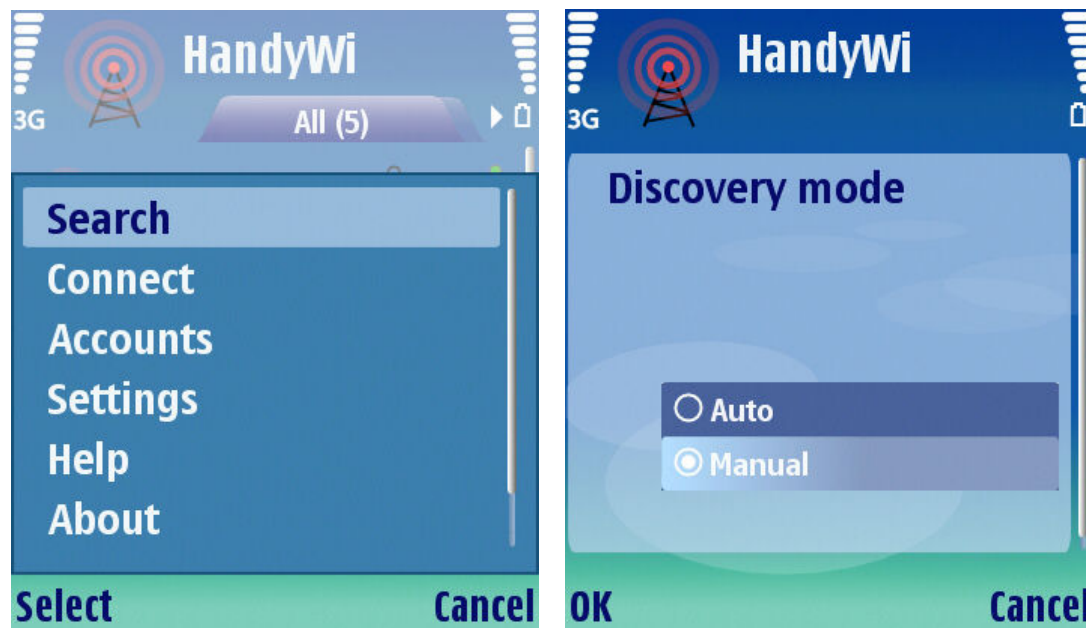
## 2.5 SISTEMOS NAUDOJIMO INSTRUKCIJA

Įdiegus ir paleidus prieigos taškų paieškos sistemą „HandyWi“, ji automatiškai atliks prieigos taškų paiešką (35 pav.). Pasibaigus paieškai yra išrikiuojami rasti aplinkoje prieigos taškai. Pagrindiniame programos lange yra matomi du valdymo mygtukai, tai yra vienas mygtukas „Hide“ kuris paslepia sistemos vartotojo sąsają, tai yra sistema nenustoja dirbusi ji toliau dirba nematomame režime.



35 pav. Prieigos taškų paieška

Paspaudus ant „Options“ mygtuko, pasirodo pagrindinis sistemos komandos (36 pav.).

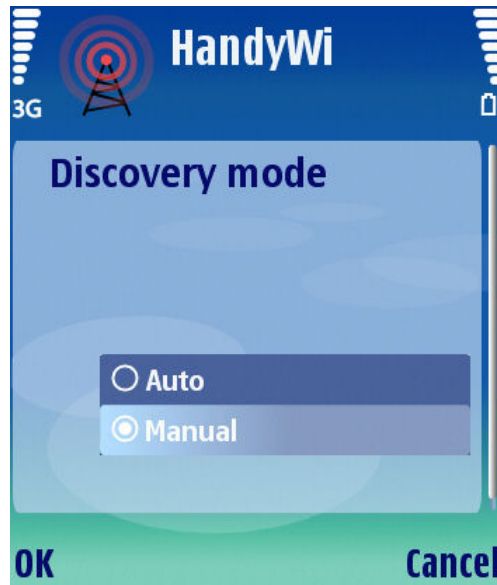


36 pav. HandyWi meniu

**Sistemos veikimo galimybės.** Pagrindinis kuriamos sistemos tikslas buvo sukurti sistemą gebančią automatiškai susirasti bei prisijunti prie prieigos taškų. Taigi sistema teikia dvi funkcijas:

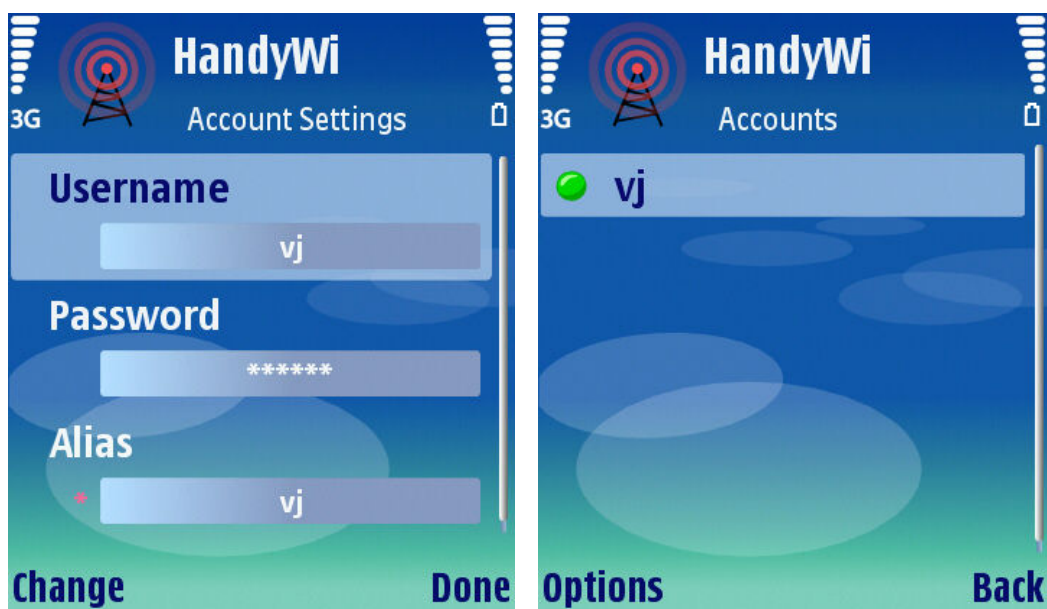
1. Rankinis valdymas (angl. Manual);
2. Automatinis valdymas (angl. Auto);

Sistemos vartotojas pagrindiniame programos lange paspaudęs Options->Settings->Discovery mode išvys 37 paveiksle pavaizduotą langą.



37 pav. Paieškos režimo langas

Šis sistemos langas yra vienas pagrindinių. Sistemos vartotojas čia turi galimybę keisi paieškos režimus iš rankinio režimo į automatinį režimą.



38 pav. HandyWi vartotojo paskyros langai



38 paveiksle pavaizduotas vartotojo paskyros informacinis langas. Šiame lange vartotojas susiveda savo vartotojo vardą bei slaptažodį. Paskyrų sąrašas yra vaizduojamas visos paskyros. Paspaudęs ant „Options“ mygtuko vartotojas aktyvuoja paskyrą, tai reiškia, kad sistema prisijungs prie maršrutizatoriaus, naudodama nurodytą aktyvią paskyrą.

Pilna prisijungimo seka prie pasirinkto maršrutizatoriaus pateikta 39 paveiksle. Pirmiausia sistema bando prisijungti prie maršrutizatoriaus, toliau jei sėkmingai susijungė, sistema siunčia tam tikrą informaciją maršrutizatoriui. Prieigos taškas gavęs informaciją iš telefono atsiunčia jam savo atitinkamo formato informaciją. Toliau HandyWi nusiunčia prisijungimui reikiamą informaciją, kur jei informacija, buvo gera, prieigos taškas duoda priėjimą prie interneto, kitu atveju atsiunčiama klaida, kuri yra parodoma vartotojui.



39 pav. HandyWi prisijungimas prie maršrutizatoriaus

## 2.6 SISTEMOS DIEGIMAS

Sistemos diegimas yra labai paprastas ir nieko nesiskiria nuo bet kuriuos Symbian operacinėje sistemoje diegiamos sistemos. Reikia atkreipti dėmesį tik į tai, kad sistema dirba įrenginiuose, kuriuose yra įmontuotas WiFi adapteris.

## 2.7 PROJEKTO IŠVADOS

Judant kryptingai link pagrindinio šio darbo tikslo, t. y. sukurti bevielio interneto prieigos taškų paieškos sistemą, sugebančią užtikrinti pastovų internetą mobiliuosiuose telefonuose ir naudojančią minimaliai telefono energetinių resursų, šioje darbo dalyje buvo atlikti minėtosios sistemos projektavimo ir programavimo darbai, kurių pasėkoje buvo sukurta „HandyWi“ programa.

Sudarant loginę programos architektūrą nuspręsta, kad kuriama sistema bus išskaidyta į keturis komponentus, kurie atitinka keturis kūrimo etapus:

- Vartotojo sąsajos komponentas - atsakingas už programos valdymo patogumą bei jos grafiką.
- Miglotosios logikos komponentas - skirtas nustatyti ar reikia atlikti pakartotinę prieigos taškų paiešką.
- Duomenų bazės komponentas - teikia priėjimą prie duomenų bazės, tokiu būdu galima lengvai padėti ir gauti duomenis iš duomenų bazės.
- Autentifikacijos komponentas – skirtas vartotojo autorizavimui ir prijungimui prie prieigos taško.

Toks nepriklausomas sistemos kūrimas etapais užtikrina sėkmingą sistemos sukūrimą. Taip pat, tai leidžia paprasčiau sukurti testavimo modelį nepriklausomoms programos dalims. Kiekvienam komponentui buvo nubraižyta UML klasių diagrama, kurios pagalba aprašyti vartotojų reikalavimai ir apibrėžti procesai. Projektuojant miglotosios logikos komponentą buvo panaudota FuzzyTech programa.

Programos veikimo stabilumui užtikrinti buvo parengtas ir pritaikytas testavimo modelis. Siekiant supaprastinti programos naudojimą vartotojui buvo parašyta programos įdiegimo ir naudojimo instrukcijos. Visa tai leidžia užtikrinti kuriamos sistemos kokybę, garantuoja valdymo patogumą ir paprastumą.

Sukurtos sistemos atitikimas iškeltiems funkciniais ir kokybės reikalavimams toliau bus apžvelgiami trečioje šio darbo dalyje. Taip pat bus bandoma įvardinti stipriąsias ir silpnąsias programinės įrangos savybes, o remiantis gautais rezultatais bus siūlomos tolimesnės programos tobulinimo bei plėtojimo galimybės.

### 3. EKSPERIMENTINIS TYRIMAS

#### 3.1 SUKURTOS SISTEMOS KOKYBĖS TYRIMAS

Pasaulyje plačiai paplitusios kompiuterinės sistemos, tačiau informacinėje visuomenėje, nuo kompiuterių pažangos mažai atsilieka ir mobilieji telefonai. Todėl antroje darbo dalyje buvo aprašytas, išmaniesiems telefonams skirtos sistemos kūrimas, kurios pavadinimas yra HandyWi.

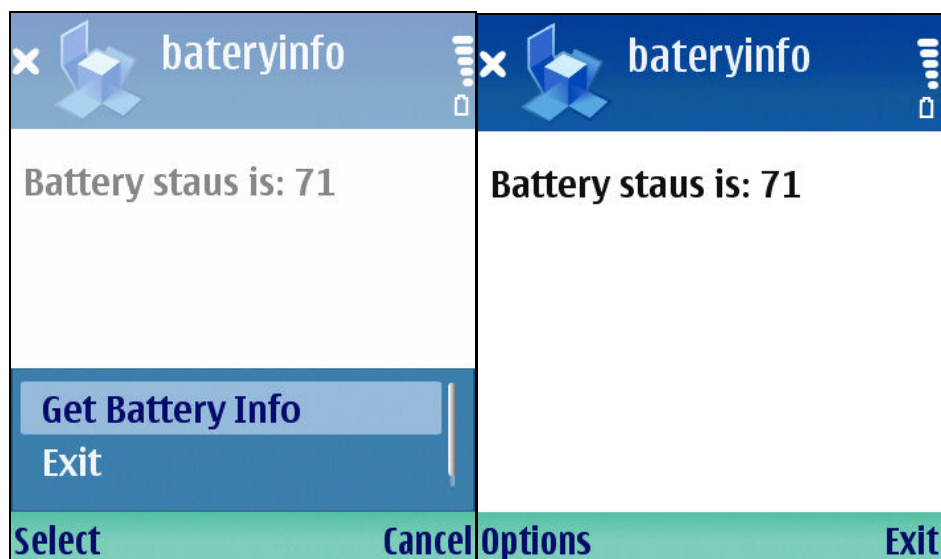
Kuriamai programai buvo iškelti du pagrindiniai uždaviniai:

1. Sukurti automatiškai sugebančią susirasti prieigos taškus programą, kai to reikia, bei gebėti prisijungti prie prieigos taško.
2. Prailginti telefono akumuliatoriaus budėjimo laiką.

Pagrindiniai programos (HandyWi) langai pavaizduoti 35-39 paveiksluose. Sistemos valdymas yra labai paprastas ir aiškus, tai užtikrina „Symbian Signed“ organizacijos suteiktas sertifikatas. 38 pav. pateiktas ypatybių valdymo langas, kuriame programos vartotojas susikuria prisijungimo paskyrą. Prisijungimo paskyra yra reikalinga autorizuojant vartotoją prie FON prieigos taško. Vartotojas turi galimybę nuspręsti ar pats nori susirasti prieigos taškus ir prisijungti prie norimo AP, ar tai paliks spręsti sistemai. Jei yra pasirinktas „Manual“ tipas, tada visą valdymą atlieka sistemos vartotojas. Pasirinkus „Auto“ tipą sistema pati automatiškai pradeda ieškoti bei prisijunginėti prie prieigos taškų. 7 lentelėje yra pateikti skirtingų telefonų markių elementų tarnavimo laikai naudojant „HandyWi“, „Hotspot Finder“ ir „FON WiFi“ programas. Šie laikai gauti panaudojant papildomai sukurtą programą „BaterInfo“ (žr. 40 pav.). Visi įrenginiai buvo pilnai įkrauti prieš pradedant tikrinti baterijos budėjimo laiką. Su kiekvienu telefonu buvo išbandomos šios trys programos. Bandymo trukmė visuose bandymuose buvo aštuonios valandos. Taigi, 7 lentelėje yra pateikti rezultatai, kai įrenginiai niekur nejudėjo ir juos supo trys FON maršrutizatoriai.

7 lentelė. Elemento stiprumas, kai įrenginiai nejudė

#	Telefono modelis	Programos pavadinimas	Elemento stiprumas % (po aštuonių valandų darbo)
1	Nokia N80	„HandyWi“	97
		„Hotspot Finder“	82
		„FON WiFi“	82
2	Nokia N95	„HandyWi“	95
		„Hotspot Finder“	81
		„FON WiFi“	81
3	Nokia E70	„HandyWi“	97
		„Hotspot Finder“	82
		„FON WiFi“	82



40 pav. BatteryInfo programos langas

Tačiau tuo atveju jeigu aplinkoje nebus pasiekiamų prieigos taškų, tai „Hotspot Finder“ ir „FON WiFi“ programos dirbs periodiniu režimu. Šiame darbe sistema buvo taip projektuota, kad jei aplinkoje nebus rasta pasiekimų prieigos taškų, tada „HandyWi“ taip pat pradeda veikti periodiniu režimu, tol kol, suranda bent vieną prieigos tašką. Taigi jei aplinkoje nebus prieigos taškų tai bandymo rezultatai visų testuojamų programų bus labai panašūs.

8 lentelėje yra vaizduojami skirtingų telefonų markių elementų tarnavimo laikai naudojant „HandyWi“, „Hotspot Finder“ ir „FON WiFi“ programas. Visi įrenginiai buvo pilnai įkrauti prieš pradedant skaičiavimus. Su kiekvienu telefonu buvo išbandomos šios trys programos. Bandymo trukmė keturios valandos. Bandymo rezultatai gauti, kai įrenginiai judėjo chaotiškai 74m<sup>2</sup> plote ir juos supo dvylika FON maršrutizatorių.

8 lentelė. Elemento stiprumas, kai įrenginiai juda chaotiškai

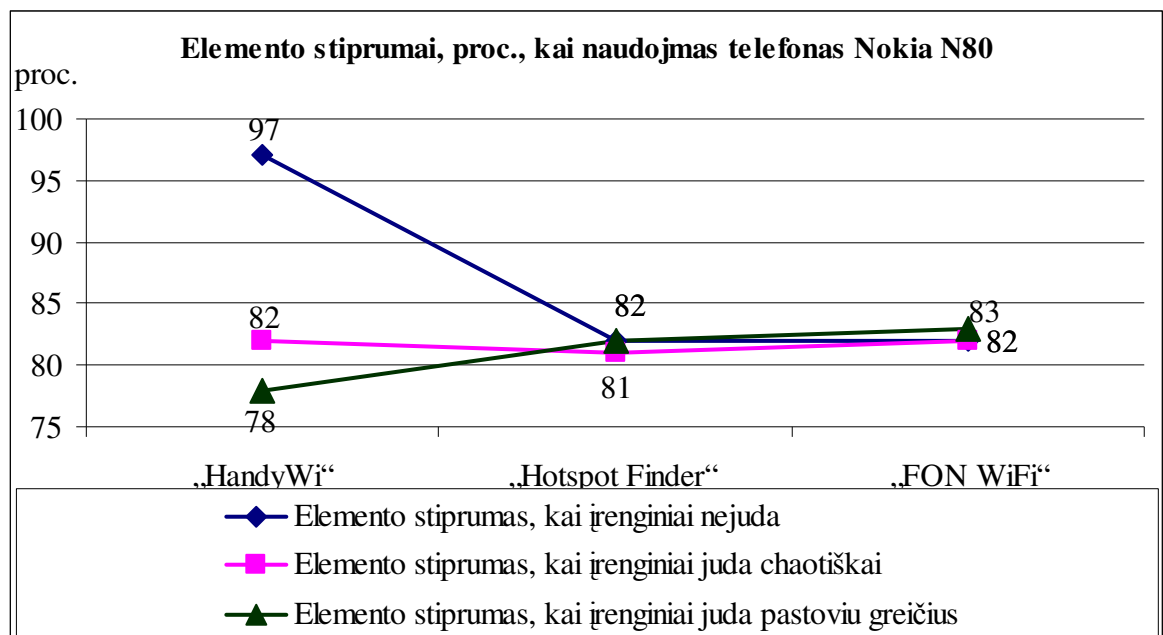
#	Telefono modelis	Programos pavadinimas	Elemento stiprumas % (po keturių valandų darbo)
1	Nokia N80	„HandyWi“	82
		„Hotspot Finder“	81
		„FON WiFi“	82
2	Nokia N95	„HandyWi“	80
		„Hotspot Finder“	83
		„FON WiFi“	81
3	Nokia E70	„HandyWi“	82
		„Hotspot Finder“	82
		„FON WiFi“	79

Pateiktoje 9 lentelėje yra vaizduojami skirtingų telefonų markių elementų tarnavimo laikai naudojant „HandyWi“, „Hotspot Finder“ ir „FON WiFi“ programas. Visi įrenginiai buvo pilnai įkrauti prieš pradedant bandymus. Su kiekvienu telefonu buvo išbandomos šios trys programos. Bandymas truko keturios valandos. Bandymo rezultatai gauti, kai įrenginiai judėjo 2 km ilgio gatvėje Ciuriche (Šveicarija), objektai judėjo apie 3-4 km/h greičiu ir juos supo apie šimtas FON maršrutizatorių.

9 lentelė. Elemento stiprumas, kai įrenginiai judėjo pastoviu greičiu

#	Telefono modelis	Programos pavadinimas	Elemento stiprumas % (po keturių valandų darbo)
1	Nokia N80	„HandyWi“	78
		„Hotspot Finder“	82
		„FON WiFi“	83
2	Nokia N95	„HandyWi“	74
		„Hotspot Finder“	80
		„FON WiFi“	79
3	Nokia E70	„HandyWi“	77
		„Hotspot Finder“	84
		„FON WiFi“	82

Taigi, atlikti bandymai parodė sukurtos sistemos pranašumus. Beveik visose situacijose HandyWi sistema aplenkė savo konkurentes.



41 pav. Elemento stiprumas, po įvairių bandymų

Pirmojo bandymo rezultatai buvo tikėtini iš anksto, kadangi yra akivaizdu: jei sistema kas minutę bando persijunginėti, kai to visai nereikia, tai tokia programa išseikvos elemento resursus daug greičiau, nei tokia sistema, kuri susirado prieigos tašką vieną kartą ir daugiau jokių papildomų paieškų neatliko.

Antrojo bandymo rezultatai yra labai panašūs, kadangi HandyWi programa turėjo dažnai perieškoti prieigos taškus bei iš naujo prie jų prisijungti; kartais nutikdavo taip, kad paieška būdavo atlikinėjama kelis kartus per minutę. Todėl jau čia HandyWi sistema pareikalavo didesnių energetinių resursų. Tačiau reikia atkreipti dėmesį į tai, kad sistema turi ne tik naudoti, kiek įmanoma mažiau energetinių resursų, bet ir užtikrinti pastovų interneto ryšį. Šiuo atveju HandyWi susitvarkė daug geriau už savo „konkurentus“, kadangi pvz. „Hotspot Finder“ pakartotinai ieškodamas prieigos taškų kas minutę, dažnai prarasdavo interneto ryšį.

Trečiojo bandymo metu HandyWi programos rezultatai buvo prasčiausi, kadangi HandyWi turėjo dažniau perieškoti prieigos taškus bei iš naujo prie jų prisijungti, beveik visada nutikdavo taip, kad paieška būdavo atliekama keletą kartų per minutę. Todėl jau čia HandyWi sistema pareikalavo didesnių energetinių resursų. Tačiau, kaip jau minėta anksčiau, reikia atkreipti dėmesį į tai, kad sistema turi ne tik naudoti kiek įmanoma mažiau energetinių resursų, bet ir užtikrinti pastovų interneto ryšį. Šiuo atveju HandyWi susitvarkė daug geriau už savo „konkurentus“, bet ir išekvojo labiausiai telefono elemento resursus. Rezultatas yra akivaizdus, kadangi kitos programos kaip „Hotspot Finder“ ir „FON WiFi“ užtikrino interneto ryšį tik periodiškai, o ne išlaikė jį pastovų.

### 3.2 SISTEMOS APRIBOJIMAI IR GALIMYBĖS

Kiekvienai kuriamai sistemai yra numatyti apribojimai tai yra sistema yra „įstatoma“ į tam tikrus režius. Sukurta „HandyWi“ sistema yra neišimtis. Sistemos vartotojui įjungus automatinį valdymą reikėtų nepamiršti, kad sistema ne visada gali užtikrinti pastovų interneto ryšį. Pastovaus internetinio ryšio užtikrinimą lemia keli faktoriai:

- Kiek aplinkoje yra pasiekiamų prieigos taškų. Čia yra akivaizdu jei aplinkoje sistema neranda prieigos taškų, tai internetinio ryšio šiuo atveju sistema tikrai neužtikrins.
- Jei sistema yra tokioje aplinkoje, kurioje yra užtekstinai prieigos taškų, tai yra bet kokiame taške atsidūrusi sistema pasieks bent vieną prieigos tašką. Tačiau ir tai neužtikrins pastovaus interneto ryšio, kadangi miglotoji biblioteka darydama sprendimą atsižvelgs į tris įverčius:

1. Signalo stiprumą, žymimą  $S$  raide.
2. Judėjimo kryptį. Judėjimo krypties išraiška  $\Delta k = S_t - S_{t-1}$ .
3. Greitį. Greičio išraiška  $\Delta v = |S_t - S_{t-1}|$ .

Abi išraiškos „judėjimo krypties“ bei „greičio“ rodo signalo stiprumo kitimą dabartiniu ir prieš tai buvusiui laiko momentu. Kuo skirtumas didesnis, tuo judėjimo greitis yra didesnis. Sistemos stabilumui

labai didelę reikšmę turi prieigos taškų antenos. Paprasti standartiniai aparatai gali transliuoti ir priimti signalą iki 100 metrų tiesioginio matomumo zonoje. Tačiau tai yra tik teorinis skaičius, kadangi signalą riboja pastatų konstrukcijos. Kitas dalykas tas, kad prieigos taškų paieška, bei prisijungimas prie maršrutizatoriaus trunka nuo kelių iki keliasdešimt sekundžių. Taigi, galime teoriškai paskaičiuoti, kokių maksimaliu vidutiniu greičiu galima judėti, kad sistema gebėtų užtikrinti pastovų interneto ryšį. Šitam uždaviniui išspręsti priėmėme prielaidą, kad aplinkoje yra išdėstyti prieigos taškai taip, kad sistema galės prisijungti prie maršrutizatoriaus atsidūrusi, bet kokiam taške. Taigi, jeigu priimame prielaidą, kad vidutiniškai prieigos taškų paieška trunka 10 sekundžių, o maršrutizatoriai transliuoja 50 metrų spinduliu signalą, tada nesunkiai galime surasti vidutinį greitį

$$v = \frac{L}{t};$$

Istatę davinius gauname, kad vidutinis greitis yra lygus 5 m/s tai yra 18 km/h. Tai yra tik teoriniai paskaičiavimai, kadangi sumodeliuoti tokią situaciją ir atlikti matavimus reikia, tam tikrų prietaisų bei tankiai išdėstytų maršrutizatorių.

### **3.3 TOLIMESNĖS SISTEMOS TOBULINIMO PLĖTOJIMO GALIMYBĖS**

Tolimesnis programos plėtojimo galimybės yra kelios:

Pirma sistema automatiškai geba prisijunginėti tik prie FON prieigos taškų. Lietuvoje FON prieigos taškai nėra dar populiari. Kovo mėnesį Lietuvoje FON registruotos „foneros“ sudarė apie 10% TEO bevielio tinklo zonos skaičiaus. Kad sistema būtų populiareesnė reikėtų padaryti palaikymą, pavyzdžiui, Zebra prieigos taškams.

Antra, reikėtų pagalvoti, kaip praplėsti sistemos galimybes, tai yra, kad sistema gerai veiktų, judant didesniu nei 18 km/h greičiu. Vienas iš sprendimų pritaikyti sistemą prie WinMax technologijos tai pakeltų sistemos tobulumą, kadangi WinMax technologija užtikrina apie 5 km atstumu signalo transliavimą. Bet ši technologija yra dar mažiau populiari šiai dienai.

## IŠVADOS

Bevielis tinklas šiandiena nebe egzotika, o tiesiog būtinybė. Vos ne kiekvienoje didesnėje užėjoje galima pasinaudoti nemokama prieigos zona ir peržiūrėti elektroninį paštą, nudurbti vieną kitą darbėlį ar tiesiog pasiskaityti naujienas. Tik, deja, bevielio tinklo prieinamumas labai ribotas. Paprasti standartiniai aparatai gali transliuoti ir priimti signalą iki 100 metrų tiesioginio matomumo zonoje. Paprastai ją riboja pastatų konstrukcijos. [16] Kita problema - aplinkoje yra įvairiausių prieigos taškų, todėl atsiranda noras, kad sistema gebėtų automatiškai rasti bei prisijungti prie prieigos taškų. Iš to ir gimė siekiamos sistemos tikslas: tai yra sukurti automatiškai gebančia aptikti bei prisijungti prie prieigos taškų sistemą, kuri išlaikytų pastovų interneto ryšį bei kiek įmanoma sumažinti energetinių telefono resursų eikvojimą.

Taigi, siekiant sukurti bevielio interneto prieigos taškų paieškos sistemą, tenkinančią šiandieninius išrankaus vartotojo poreikius tokius, kaip patogus valdymas, nuolatinis internetinio ryšio užtikrinimas, telefono energetinių resursų taupymas, buvo apžvelgta literatūroje siūlomi ir galimi pritaikyti minėtos problemos sprendimo būdai. Po atliktos analizės buvo nuspręsta, sistemos valdymui panaudoti miglotąją logiką, kadangi naudojant miglotąją valdymą galima apjungti žmogiškąją patirtį, atsižvelgiant į ekspertų siūlymus ir sukaupytų bandymų rezultatus.

Projektuojama sistema buvo išskaidyta į keturis etapus: vartotojo sąsajos komponentą, miglotosios logikos komponentą, duomenų bazės komponentą ir autentifikacijos komponentą. Toks nepriklausomas sistemos kūrimas etapais užtikrina sėkmingą sistemos sukūrimą.

Suprojektuotuos ir sukurtos programos veikimo stabilumui užtikrinti buvo parengtas ir pritaikytas baltos dėžės testavimo modelis. Naujai sukurtos sistemos kokybei ir valdymo patogumui bei paprastumui užtikrinti buvo parašytos programos įdiegimo ir naudojimo instrukcijos.

Sukūrus sistemą, buvo atliktas eksperimentinis jos tyrimas, palyginant sukurta sistemą su kitomis sistemomis. Rezultatuose buvo kreipiamas dėmesys į elemento naudojimą, bei interneto užtikrinamumą. Bandymo rezultatai išryškino „HandyWi“ sistemos stipriąsias savybes:

- sistemai nejudant arba judant labai nedideliame plote, „HandyWi“ sugeba ne tik automatiškai rasti ir prisijungti prie prieigos taškų, bet ir išnaudoja mažiausiai elemento resursų. Tai paprasta paaiškinti: kitos sistemos naudoja periodinę paiešką ir dažnai atsitikdavo taip, kad sistemai visai nereikėjo pakartotinai ieškoti prieigos taškų.

- sistemai judant dideliame plote, „HandyWi“ sugeba užtikrinti pastovų internetą (pastovaus interneto užtikrinimui reikia tankaus prieigos taškų tinklo). Kitos sistemos neužtikrina pastovaus interneto ir tai yra akivaizdu, nes jos pakartotinai ieškodavo prieigos taškų atėjus nustatytam laikui ir



neanalizuodavo konkrečios esamos situacijos, tačiau šios sistemos išseikvojo mažiau elemento resursų, kadangi buvo atliktas mažesnis skaičius prieigos taškų paieškų.

Taigi sukurta sistema yra pranašesnė už kitas sistemas tuo, kad pakartotinai ieško prieigos taškų tada kai to reikia. Dėl šios priežasties yra taupomi telefono elemento resursai bei daug geriau užtikrinamas pastovus internetas, net tuo atveju, kai prieigos taškų paieška yra atliekama periodiškai.

Siekiant pagrįsti sistemos funkcionavimo tikrumą buvo paskaičiuotas teorinis sistemos judėjimo greitis, t.y. koku greičiu gali judėti sistema, kad ji dar būtų veiksminga ir užtikrintų interneto ryšį. Gauti rezultatai parodė, kad greitis gali siekti iki 18 km/h.

Skiriant daugiau investicijų tiek laiko, tiek finansų atžvilgiu, būtų galima toliau plėtoti sukurta sistemą patobulinant ją sugebėjimu prisijungti ne vien tik prie „Fon“ prieigos taškų, bet ir, pavyzdžiui, prie „Zebra“ prieigos taškų, taip pat pritaikyti sistemą prie WinMax technologijos.

## LITERATŪRA

1. Wikipedija – laisvoji enciklopedija. Wi-Fi. [žiūrėta 2007-10-05]. Prieiga per internetą < <http://lt.wikipedia.org/wiki/Wi-Fi> >.
2. Petrovas P. „Wi-Fi“ gali pakeisti mobiliuosius telefonus. [žiūrėta 2007-05-25]. Prieiga per internetą < [www.elektronika.lt/news/communication/8007](http://www.elektronika.lt/news/communication/8007) >.
3. Savaiminiai belaidžio ryšio tinklai spartina duomenų perdavimą. Ryšių technikos naujienos Nr.1. 2004. [žiūrėta 2007-05-25]. Prieiga per internetą < <http://rtn.elektronika.lt/rtn/0401/savaiminiai.html> >.
4. Hitchhiker. [cited 2006-12-05]. Available from Internet < <http://hitchhiker.en.softonic.com/pocket> >.
5. FON WiFi Connection Manager description. [cited 2006-11-11]. Available from Internet < <http://handheld.softpedia.com/get/System-Utilities/Communications> >.
6. Naujasis "Nokia" telefonas skirtas ir internetui. [žiūrėta 2007-09-26]. Prieiga per internetą < <http://vz.lt/Default2.aspx?ArticleID=71105581-bac6-46ea-8c22-c9989b1982a6> >.
7. Kontrolės ir valdymo sistemos. [žiūrėta 2006-12-17]. Prieiga per internetą < <http://www.ifko.ktu.lt/~vypet/T120B005/> >.
8. Petrauskas V. Nuolatinės srovės variklio valdymas kompiuteriu. Kaunas. 2000. [žiūrėta 2007.10.05]. Prieiga per internetą: < <http://ifko.ktu.lt/~vypet/KITI/nsvvaldym.htm> >.
9. Nemura A. Neraiškių sistemų panaudojimas elektros energetikoje. Mokslas ir technika, Nr.7-8. 2007.
10. Bannatyne R. Development of Fuzzy Logic in Embedded Control. 1994.
11. Passino K. M., Yurkovich S. Fuzzy Control. –Addison Wesley, 1998. –502 p.
12. HotSpot finder. [cited 2007-10-05]. Available from Internet < <http://my-symbian.com/s60v3/software/applications.php?faq=4&fldAuto=26> >.
13. WiFi APIs?. [cited 2006-12-17] Available from Internet < <http://forum.newlc.com/index.php/topic,5885.0.html> >.
14. Wikipedija – the free encyclopedia. Center of mass. [cited 2007-06-15] Available from Internet < [http://en.wikipedia.org/wiki/Centre\\_of\\_gravity](http://en.wikipedia.org/wiki/Centre_of_gravity) >.
15. Richard Piggin and David Brandt. Wireless ethernet for industrial applications 2006.
16. Pakrijauskas S. Fon dalybos bevieliu internetu. Naujoji komunikacija, Nr. 16. 2007.
17. 3 L žodynas. [žiūrėta 2007-12-25]. Prieiga per internetą

< <http://www.3-l.lt/content/view/15/32/> >.

18. Kutscher D., Jörg. Ott. Deploying VoIP over Public Wireless Networks Challenges and Perspectives 2005-07-05 IQPC VoWLAN Conference, Berlin. P.36.
19. Wikipedija – the free encyclopedia S60 platform. [cited 2007-12-15] Available from Internet  
< [http://en.wikipedia.org/wiki/S60\\_platform](http://en.wikipedia.org/wiki/S60_platform) >.
20. Baršauskienė V. Studijų darbų parengimo tvarka/ V. Baršauskienė, I. Mačrinskienė. Kaunas, 2005.  
78 p. ISBN 9955-09-064-2.

Jasaitis V. Automatic access point discovery system for wireless network: Master's work in informatics engineering / supervisor prof. dr. R. Jasinevičius; Kaunas university of technology, Faculty of Informatics, Department of Information systems – Kaunas, 2008. – 61 pages.

## **SUMMARY**

The object of this work is to keep stable communication between smart phone and wireless adapter. There are many created access points discovery systems at the moment, but those systems “don't know” how to discovery and connect to suitable access point. Also existing systems have lost a lot of smart phone battery energy during discovering access points. The main existing systems issues and issues solving abilities by using fuzzy logic are presented at this work.

The subject of research, object and subject relevance were detectable through literature analysis and through existing access point discovery systems issues are presented in the first part.

The subject of research and object are presented in the first part of this work. Also, subject relevance is presented pursuant to literature analyze of wireless internet access point discovery system.

Object components design and programming process are presented in the second part. UML, Visio and Fuzzytech programs are used for designing components. Events, sequence and class diagrams are viewed in this part, also this part contain database schema and fuzzy sets diagrams. The main wireless internet discovery system is created in this part. The system is called “HandyWi”.

The “HandyWi” system is comparison with existing system and quality data results are stored in the third part.

The conclusions of the work, automatic access point discovery system functionality, oneness and further development abilities are presented in last part.

The work consists of 41 pictures, 9 tables and conceptual dictionary. There have been 20 bibliographical sources used.

## TERMINŲ, SANTRUMPŲ ŽODYNAS

### WiFi -

Wireless Fidelity. - protokolas ir standartas įrenginiams naudojantiems plačiajuostę radijo ryšį, skirtas organizuoti vietinius belaidžius ryšius. Sukurtas Wi-Fi Alliance konsorciumo, **IEEE 802.11** (a,b,c, g) standarto pagrindu. Mobilūs įrenginiai turintys Wi-Fi klientinius gavimo-priėmimo įrenginius gali pasijungti į lokalius tinklus ir gauti interneto prieigą per taip vadinamus ryšio taškus („hotspots“, access point). [17]

### Prieigos taškas

#### *(access point; hotspot) -*

jungimosi taškas - įrenginys, sujungiantis bevieliu komunikacijos įrenginius į bevielį tinklą (wireless network). Paprastai jungiamas prie vielinio kompiuterinio tinklo ir perduoda duomenis tarp vielinio ir bevielio tinklo įrenginių. Keli sujungti AP įrenginiai gali suformuoti didesnę tinklą, leidžiantį daryti roaming'ą. AP įrenginiai turi savo IP adresą konfigūravimui.[17]

### Symbian OS –

populiari operacinė sistema išmaniems telefonams.

### Series 60 –

kompanijos Nokia vartotojo sąsajos platforma padaryta ant Symbian OS.

### Išmanusis telefonas –

mažo kompiuterio procesoriaus galia ir atmintis, integruota į standartinį mobiliojo telefono korpusą, t.y. tai mobilusis telefonas turintis operacinę sistemą, pavyzdžiui Symbian OS, Windows CE ir t.t.

### GUI –

vartotojo sąsaja.

### MB –

megabaitas.