

Opinnäytetyö (AMK)

Tieto- ja viestintäteknikka

2024

Juho Martti

Langattoman käyttöönoton kehittäminen ilmanvaihtokoneelle



Opinnäytetyö (AMK) | Tiivistelmä

Turun ammattikorkeakoulu

Tieto- ja viestintätekniikka

2024 | 49 sivua

Juho Martti

Langattoman käyttöönoton kehittäminen ilmanvaihtokoneelle

Ilmanvaihtokoneen tehtävä on pitää rakennusten sisäilma raikkaana. Opinnäytetyön tavoitteena oli kehittää Vallox Oy:lle ratkaisu, jonka avulla ilmanvaihtokoneiden asentajat saavat tehtyä ilmanvaihtokoneiden käyttöönoton langattomasti heidän älypuhelimillaan. Ratkaisun piti olla sellainen, että ilmanvaihtokoneisiin ei tarvinnut tehdä muutoksia. Päädyttiin kehittämään verkkosovitin, jonka saa liitettyä ilmanvaihtokoneen Ethernet-porttiin.

Verkkosovittimen toteuttamisessa hyödynnettiin ESP32-mikrokontrolleria ja sille tarkoitettua ohjelmointikehitysympäristö ESP-IDF:ää. Verkkosovittimen toteutuksessa käytettiin seuraavia menetelmiä. DHCP-palvelinta käytettiin laitteiden IP-osoitteiden määrittämiseen. Wi-Fin ja Ethernetin välisen liikenteen mahdollistamiseksi käytettiin Wi-Fi- ja Ethernet-rajapintojen yhdistämistä. Hyödynnettiin Wi-Fi-tukiasemaa, jonka avulla asentajat pystyivät yhdistämään älypuhelimet verkkosovittimeen ja hallitsemaan ilmanvaihtokoneita. Käytettiin verkkopalvelinta, jonka avulla tehtiin verkkosivu, josta asentaja sai avattua ilmanvaihtokoneen käyttöliittymän helposti.

Opinnäytetyön tuloksena tuotettiin verkkosovitin, joka täyttää työn alussa määritetyt vaatimukset. Verkkosovitinta testattiin ja varmistettiin sen toiminta ilmanvaihtokoneiden käyttöönotossa älypuhelimien kanssa. Tämän lisäksi kehitettiin kehityslevylle kotelo, joka suojaa levyä, sekä luotiin yksinkertainen ohje verkkosovittimen käytölle.

Asiasanat:

mikrokontrolleri, Wi-Fi, ilmanvaihtokone, verkkosovitin

Bachelor's Thesis | Abstract

Turku University of Applied Sciences

Information and Communications Technology

2024 | 49 pages

Juho Martti

Developing wireless commissioning for ventilation units

The function of the ventilation machine is to keep the indoor air of buildings fresh. The objective of this thesis was to develop a solution for Vallox Oy that would allow ventilation unit installers to wirelessly commission ventilation units using their smartphones. The solution had to be such that no modifications were needed to the ventilation units. The decision was made to develop a network adapter that can be connected to the Ethernet port of the ventilation unit.

The implementation of the network adapter utilized the ESP32 microcontroller and the corresponding programming environment ESP-IDF. The following methods were used in the implementation of the network adapter: a DHCP server was used to assign IP addresses to the devices; to enable traffic between Wi-Fi and Ethernet, Wi-Fi and Ethernet interfaces were combined; a Wi-Fi access point was utilized to allow installers to connect their smartphones to the network adapter and manage the ventilation units; a web server was used to set up a web page from which the installer could easily access the ventilation unit's user interface.

As a result of the thesis, a network adapter that meets the requirements set at the beginning of the work was produced. The network adapter was tested and its operation in commissioning ventilation machines with a smartphone was confirmed. Also, a case was developed for the development board to protect the board, and a simple guide was created for using the network adapter.

Keywords:

Microcontroller, Wi-Fi, Ventilation unit, Network adapter

Sisältö

Lyhenteet	7
1 Johdanto	9
2 Mikrokontrollerit	11
2.1 Mikrokontrollerin historia	11
2.2 Mikrokontrollerin toiminta	12
2.3 Mikrokontrollerityyppejä	12
2.3.1 Dataväylän leveys	12
2.3.2 Rakenteen arkkitehtuuri	13
2.3.3 Käskykannan arkkitehtuuri	14
2.3.4 Mikrokontrollerin käyttötarkoitukset	15
2.4 Mikrokontrollerin ja mikroprosessorin erot	16
3 Wi-Fi	18
3.1 Wi-Fin historia	18
3.2 Wi-Fin toiminta	19
3.3 Verkkolaitteet	21
3.4 Wi-Fin tietoturva	22
3.4.1 Wi-Fin uhat ja riskit	23
3.4.2 Suojausprotokollat	24
4 Suunnittelu	26
4.1 Vaatimukset	26
4.2 Langattoman teknologian valitseminen	27
4.3 Mikrokontrollerien vertailu ja valinta	28
4.4 Kehitysympäristöt	29
4.4.1 ESP-IDF	30
4.4.2 ESP Arduino Core	30
4.4.3 Kehitysympäristön valitseminen	31
4.5 Toteutuksen suunnitteleminen	31

5 Toteutus	33
5.1 Ympäristön asentaminen	33
5.2 Ohjelmointi	34
5.2.1 Projektin luominen ja rajapintojen määrittäminen	34
5.2.2 IP-osoitteiden määrittäminen	34
5.2.3 Ilmanvaihtokoneen käyttöliitymän avaaminen	36
5.2.4 Verkkosovittimen verkkosivun parantaminen	36
5.3 Lisätoiveiden toteuttaminen	38
5.4 Testaus	40
6 Yhteenveto	43
Lähdeluettelo	44

Kuvat

Kuva 1. ISO/OSI-malli.	20
Kuva 2. Verkkosiltojen esimerkkikuva.	22
Kuva 3. ESP32 Ethernet kit.	29
Kuva 4. Visual Studio code-lisäosa.	33
Kuva 5. Kuva verkkosivusta, jossa näkyy ilmanvaihtokoneen IP-osoite.	36
Kuva 6. Lopullinen toteutus verkkosivusta.	37
Kuva 7. Verkkosivun ulkoasu, kun ilmanvaihtokone ei ole yhdistetty verkkosovittimeen.	38
Kuva 8. Kehityslevylle tehty kotelo.	39
Kuva 9. Käyttöohje Wi-Fi-yhteyden muodostamiseen ja verkkosivun avaamiseen.	39
Kuva 10. Kuvaus testausympäristöstä.	41

Taulukot

Taulukko 1. Assembly-koodi-esimerkit	14
Taulukko 2. MoSCoW-taulukko	26
Taulukko 3. Älypuhelimien eri langattomat teknologiat ja tiedot.	27

Lyhenteet

AES	Advanced Encryption Standard
CCMP	Counter Mode with CBC-MAC Protocol
CISC	Complex Instruction Set Architecture
CPU	Central Processing Unit
CSS	Cascading Style Sheets
DHCP	Dynamic Host Configuration Protocol
DoS	Denial-Of-Service
DSSS/CCK	Direct-Sequence Spread Spectrum/Complementary Code Keying
EAP	Extensible Authentication Protocol
ESP-IDF	Espressif IoT Development Framework
FCC	Federal Communications Commission
GPIO	General-Purpose Input/Output
HTML	Hyper Text Markup Language
I/O	Input / Output
I2C	Inter-Integrated Circuit
IDE	Integrated Development Environment
IEEE	Institute of Electrical and Electronics Engineers
IoT	Internet of Things
IP	Internet Protocol
MITM	Man-In-The-Middle
MOS	Metal-Oxide-Semiconductor
PoE	Power over Ethernet

PSK	Pre-Shared Key
QR	Quick-Response
RAM	Random-Access Memory
RC4	Rivest Cipher 4
RISC	Reduced Instruction Set Architecture
ROM	Read Only Memory
SAE	Simultaneous Authentication of Equals
SPI	Serial Peripheral Interface
SSDP	Simple Service Discovery Protocol
TCP	Transmission Control Protocol
TKIP	Temporal Key Integrity Protocol
UART	Universal Asynchronous Receiver Transmitter
USB	Universal Serial Bus
VPN	Virtual Private Network
WECA	Wireless Ethernet Compatibility Alliance
WEP	Wired Equivalent Privacy
WLAN	Wireless Local Area Network
WPA	Wi-Fi Protected Access

1 Johdanto

Ilmanvaihtokoneen tehtävä on vaihtaa ilmaa rakennuksissa ja tuoda raikasta ulkoilmaa sisälle ja puhaltaa jäteilman ulos. Tämän avulla rakennuksien ilmanlaatu pysyy raikkaana, sekä estetään kosteuden kertyminen sisätiloihin ja vältetään rakennuksen homehtuminen. Yleisesti ilmanvaihtokoneissa on lämmöntalteenotto, joka lämmittää tuloilmaa asunnosta poistettavan ilman avulla. Näin ilmanvaihtokoneet pysyvät energiatehokkaina. [1] [2]

Kun ilmanvaihtokone käynnistetään ensimmäisen kerran, pitää siihen tehdä käyttöönotto. Käyttöönotossa valitaan käyttöliittymän kieli, koneen aika, käyttäjätasot, tulo- ja poistoilmavirtojen asetukset sekä eri ilmanvaihdon käyttötilojen arvot. Tulo- ja poistoilmavirran määrittämisessä tarvitaan ilmavirranmittauslaitteistoa. Tämän takia ammattilaisen on tehtävä käyttöönotto.

Jokaiselle asunnolle ja huoneistolle on oma ilmanvaihtosuunnitelma, jossa on määritetty tilan tulo- ja poistoilmavirrat. Ilmavirranmittausta tehtäessä ensiksi tulee säätää ilmanvaihtokanaviston venttiilit oikeisiin asentoihin. Tämän jälkeen puhallimet säädetään mahdollisimman lähelle ihannearvoja laitevalmistajan puhallinkäyrästä avulla. Ilmanvaihtokoneen kokonaisilmavirran säätämisessä voidaan käyttää apuna ilmanvaihtokoneessa olevia tai sen mukana tulevia mittayhteitä. Ilmavirtojen säätämisessä tulee välttää ilmavirtojen kuristamista venttiileillä, jotta ilmanvaihtokone saadaan toimimaan mahdollisimman vähääänisesti ja energiatehokkaasti. [3]

Opinnäytetyön toimeksiantaja on Vallox Oy, joka valmistaa päätuotteena ilmanvaihtokoneita, mutta myös ilmankäsittelykoneita, huippuimureita ja liesikupuja. Opinnäytetyön tarkoituksena on tehdä ratkaisu, joka mahdollistaisi ilmanvaihtokoneen langattoman käyttöönoton.

Toimeksiantajan ilmanvaihtokoneen käyttöönoton pystyi tekemään vain ilmanvaihtokoneen yhteydessä olevasta ohjainpaneelistä tai tietokoneella Ethernet-kaapelin välityksellä, sillä ilmanvaihtokoneessa ei ole tukea langattomille ratkaisuille, kuten Bluetoothille tai Wi-Fille. Ilmanvaihtokone

voidaan kuitenkin toimittaa ilman ohjauspaneelia eikä kaikilla asentajilla ole käytössä tietokonetta, jonka kautta voisi tehdä käyttöönoton.

Tässä työssä tavoitteena on löytää ratkaisu, jonka avulla käyttöönoton voisi tehdä langattomasti esim. älypuhelimella ilman että ilmanvaihtokoneen ohjelmistoon tarvitsee tehdä muutoksia. Tarkoituksena on kehittää ohjelmisto ulkoiselle mikrokontrollerille, joka muuttaa Ethernet-kaapelin kautta tuotavan tiedon langattomaksi. Ratkaisuna toteutettava mikrokontrolleri olisi erillinen tuote, joka liitettäisiin ilmanvaihtokoneeseen Ethernet-kaapelilla ja sen avulla ilmanvaihtokoneen ohjauskäyttöliittymän voisi avata älypuhelimien selaimiin.

2 Mikrokontrollerit

Mikrokontrollerit (engl. microcontrollers uCs tai microcontroller units MCUs) ovat erittäin pieniä yhden sirun itsenäisiä mikrotietokoneita. Mikrokontrollerit koostuvat prosessorista (CPU), muistista (RAM, ROM), sisään-/ulostuloporteista (I/O ports) ja ajastimista. Mikrokontrollerit sisältävät erilaisia kommunikaatorajapintoja tiedonsiirtoa varten, kuten SPI, I2C, UART ja USB. [4]

Mikrokontrollereita löytyy esimerkiksi autoista, jääkaapeista, pesukoneista, tulostimista, juoma-automaateista, maksupäätteistä, kännyköistä, kaukosäätimistä sekä monista muista erilaisista laitteista. Tyypillisesti mikrokontrollereiden tehtävänä on suorittaa valmiiksi ohjelmoituja tehtäviä toistuvasti tai ajastetuvin aikavälein. [5]

2.1 Mikrokontrollerin historia

Ensimmäisen mikrokontrollerin kehitti Texas Instrumentsin insinööri Gary Boone apunaan MOS-osaston markkinointipäällikkö Daniel Baudouin sekä insinööri Michael Cochran. Ensimmäinen mikrokontrolleri kehitettiin laskimia varten ja se hyödynsi RAM- ja ROM-muistien käyttöä, minkä ansiosta laskimiin tarvittavien komponenttien määrää vähentyi 50-kertaisesti. [6]

Texas Instruments julkaisi ensimmäiset yksisiruiset laskinmikropiirit TMS1802NC:n vuonna 1971 ja TMS0100:n vuonna 1972. TMS1802NC- ja TMS0100-laskinmikropiirien rakenne muistutti mikrokontrollerin rakennetta. Mikrokontrollereiksi niitä ei voinut kuitenkaan kutsua, koska ne olivat vain tarkoitettu laskinkäyttöön ja käyttäjä ei pystynyt muokkaamaan ohjelmistokoodia. [6]

Vuonna 1974 Texas Instruments julkaisi yleiskäyttöön tarkoitetun TMS1000-mikrokontrollerisarjan. TMS1000-sarjan mikrokontrollerit täyttivät kriteerit, joiden mukaan ne voitiin luokitella mikrokontrolleriksi. [7]

2.2 Mikrokontrollerin toiminta

Mikrokontrolleri koostuu prosessorista, muisteista, I/O-porteista ja ajastimista. Prosessori (CPU) on kuin mikrokontrollerin aivot. Se hallitsee ja valvoo kaikkea, mitä mikrokontrollerissa tapahtuu. Prosessorin tehtäviin kuuluu toimintaohjeiden lukeminen muistilta ja niiden käsitteleminen. Prosessori suorittaa muistista saatujen ohjeiden mukaiset toiminnot. [5]

Mikrokontrolleri sisältää yleensä kahta erilaista muistia RAM ja ROM. RAM-muistia käytetään väliaikaisen tiedon tallentamiseen, johon mikrokontrollerin on päästävä käsiksi nopeasti. RAM-muistiin ei voi tallentaa tietoa pysyvästi, koska se tyhjenee joka kerta, kun virta katkaistaan mikrokontrollerista. ROM-muistia käytetään pysyvien tietojen tallentamiseen. ROM-muistiin tallennetaan mikrokontrollereiden käyttämä ohjelmistokoodi eli toimintaohjeet. [5]

Mikrokontrollereissa on I/O-portit, joiden avulla se kommunikoi muiden komponenttien kanssa. Ajastin toimii mikrokontrollerin kellona. Ajastimen avulla mikrokontrolleri pysyy tietoisena siitä, kuinka kauan eri toiminnoista on ja se osaa suorittaa ne oikein aikaväleihin. [5]

2.3 Mikrokontrollerityyppejä

Mikrokontrolleja on paljon erilaisia. Mikrokontrollerit voidaan lajitella dataväylän laajuuden, rakenteen arkkitehtuurin, komentosarja-arkkitehtuurin ja käyttötarkoituksen perusteella. [8]

2.3.1 Dataväylän leveys

Yleisin ja yksinkertaisin mikrokontrollereiden luokittelutapa on dataväylien leveyden mukaan. Yleisimmät dataväyläleveydet ovat 8 b, 16 b ja 32 b. Bittien määrä kertoo, kuinka paljon dataa pystytään käsittelemään yhdellä laskentakierroksella. Esimerkiksi 8-bittinen mikrokontrolleri pystyy käsittelemään lukuja väliltä 0 – 255 ($2^8 = 256$ eri lukua), kun taas 32-bittinen

pystyy käsittelemään lukuja väliltä 0 – 4 294 967 295 ($2^{32} = 4\,294\,967\,296$ eri lukua). Dataväylän laajuus kertoo mikrokontrollerin tehokkuudesta. Mitä laajempi dataväylä on, sitä tehokkaampi mikrokontrolleri on. [8]

2.3.2 Rakenteen arkkitehtuuri

Mikrokontrollereita voidaan luokitella niiden rakennearkkitehtuurin mukaan. Kaksi yleisimmin käytettyä arkkitehtuuria mikrokontrollereissa ovat Harvard- ja Von Neumann-arkkitehtuurit.

Harvard-tyyppin mikrokontrollereissa on kaksi eri muistia, toinen ohjelmiston koodille ja toinen tiedon käsittelylle. Erilliset muistialueet vähentävät tiedon vioittumisen riskiä, mutta tuovat rajoituksia muistin varaukselle. Mikäli toisen muistialueen muisti loppuu, käyttäjä ei pysty varaamaan toiselta muistilta tilaa muuhun tarkoitukseen, vaikka siellä olisi vapaata tilaa. Muistit ovat käytettävissä vain niille tarkoitetuille tiedoille. Molemmilla muisteilla on omat tietoväylät, mikä mahdollistaa ohjelmakäskyjen ja tietojen käsittelyn yhtäaikaisesti ja tämä mahdollistaa nopeamman suorittamisen. Kaksi tietoväylää johtaa kuitenkin monimutkaisempiin arkkitehtuureihin ja nostaa mikrokontrollerin hintaa. [8] [9]

Von Neumann-tyyppin mikrokontrollereissa on yksi muisti, johon tallennetaan ohjelmistokoodi ja käsiteltävät tiedot. Yhden muistin ansiosta prosessori voi käyttää ohjelmakäskyjä ja käsiteltävää tietoa peräkkäin samalta muistilta, mikä yksinkertaistaa ohjelmistokehitystä ja arkkitehtuuria. Yhden muistin ansiosta Von Neumann-arkkitehtuurin muistin käyttö on joustavampaa kuin Harvard-arkkitehtuurin ja mahdollistaa dynaamisen muistitilan hallinnan. Yhden muistin arkkitehtuuri voi hidastaa ohjelmistokäskyjen suorittamista, sillä prosessori voi joutua odottamaan käskyjen hakemista, jos tietojen käsittely on käynnissä. Koska kaikki tallennetaan yhdelle muistille, lisääntyy riski tiedon päällekkäin kirjoittamiselle, mikä johtaa tiedon vioittumiseen. [8] [9]

Harvard-arkkitehtuuria käytetään yleisemmin mikrokontrollereissa sen nopean ja luotettavan ohjelmiston suorittamisen takia. Von Neumann-arkkitehtuuria käytetään enemmän tietokoneissa sen dynaamisen muistin takia. [9]

2.3.3 Käskykannan arkkitehtuuri

Mikrokontrollereita voidaan luokitella myös prosessorien komentosarjan arkkitehtuurin mukaan. Komentosarjan arkkitehtuureja on kahta eri tyyppiä CISC (Complex Instruction Set Architecture) ja RISC (Reduced Instruction Set Architecture). [8]

CISC-prosessorit pystyvät käsittelemään monimutkaisi komentoja, joka mahdollistaa usean toiminnon suorittamisen yhdellä komennolla. CISC-prosessorit käyttävät vähemmän muistia, sillä haluttu toiminnallisuus voidaan toteuttaa pienemmällä koodin määrällä. Monimutkaisten komentojen suoritus vaatii kuitenkin prosessorilta lisää suoritusaikaa, mikä lisää prosessorin virrankulutusta. [10]

RISC-prosessorit käsittelevät yksinkertaisempia komentoja kuin CISC-prosessorit, minkä takia ohjelmakoodia voidaan tarvita huomattavasti enemmän yksinkertaisiin tehtäviin, mutta yksinkertaisten komentojen ansiosta suorittaminen on nopeaa. Nopean suorituskyvyn ansiosta RISC-prosessoreiden virrankulutus on pienempi kuin CISC-prosessoreiden. [10]

Taulukossa 1 on kuvattu, kuinka CISC- ja RISC-arkkitehtuurit eroavat toisistaan. Taulukossa on kahden eri muistipaikkoihin tallennettujen lukujen yhteenlaskutoiminto CISC ja RISC tarvitsemilla komennoilla Assembly-koodina. CISC suorittaa tämän yhteenlaskun yhdellä komennolla, kun taas RISC tarvitsee neljä eri komentoa. [10]

Taulukko 1. Assembly-koodi-esimerkit

CISC-esimerkki	RISC-esimerkki
ADD 0A, 1A	LOAD A, A0
	LOAD B, A1
	ADD A, B
	STORE A0, A

Taulukossa 1 näkyvä CISC-esimerkki tarkoittaa sitä, että CISC-prosessori tarvitsee vain yhden komennon ADD. Koska CISC-prosessori pystyy käsittelemään arvoja suoraan muistista, se voi käyttää muistiosoitetta suoraan komennossa. [10]

Taulukossa 1 näkyvä RISC-esimerkki tarkoittaa sitä, että RISC-prosessori ei pysty käsittelemään tietoja suoraan muistista. Muistissa oleva arvo pitää ladata ensin rekisteriin muistiosoitteen avulla, minkä jälkeen ne voidaan vasta laskea yhteen. Lopuksi yhteenlaskettu arvo tallennetaan takaisin muistiin. [10]

2.3.4 Mikrokontrollerin käyttötarkoitukset

Mikrokontrollereita voidaan luokitella niiden käyttötarkoitusten mukaan. Käyttötarkoitukset voidaan jakaa neljään eri ryhmään niiden tarpeiden mukaan: teollisuus-, ajoneuvo-, terveys- ja kuluttajaelektroniikka. [8]

Jotta mikrokontrolleri voidaan luokitella teollisuuskäyttöön, sen on oltava kestävä, luotettava ja sen pitää toimia vaativissa olosuhteissa. Vaativia olosuhteita ovat vaihtelevat lämpötilat (-40°C - $+125^{\circ}\text{C}$), äärimmäinen kosteus, mekaaniset iskut tai värinät, sähkömagneettiset häiriöt ja pitkäaikaiset käyttöajat. Näitä mikrokontrollereita käytetään esimerkiksi robottien valmistukseen ja teollisuuden prosessien ja moottorien ohjaamiseen, koska mikrokontrollerit on optimoitu toistuviin ja ympäristöön reagoiviin tehtäviin. [8]

Ajoneuvomikrokontrollerit ohjaavat moottorin, turvallisuuslaitteiden ja kuljettajan hallintalaitteiden toimintaa. Ajoneuvoissa käytettävien mikrokontrollereiden tulee olla kestäviä, ja niiden on pystyttävä toimimaan vikatilanteissa luotettavasti. Niiden tulee myös toimia vaihtelevissa lämpötiloissa. [8]

Terveydenhoitoon tarkoitetuilla mikrokontrollereilla on vaativammat turvallisuuskriteerit kuin edellä mainituilla ryhmillä. Näiden on täytettävä tiettyjä standardeja, kuten ISO 13485. Terveystieteissä on tärkeää, että

mikrokontrollerit ovat luotettavia, tarkkoja ja pystyvät toimimaan reaaliaikaisesti. [8]

Kuluttajaelektronikkaan suunnattuja mikrokontrollereita käytetään monissa kodinkoneissa ja lähes kaikissa kodinelektronikkalaitteissa, kuten älytelevisioissa ja äänijärjestelmissä. Jotta mikrokontrolleri voidaan luokitella kuluttajaelektronikkamikrokontrolleriksi, sen tulee tukea seuraavia toimintoja: käyttöliittymän ohjausta, alhaista virrankäyttötilaa, multimedian käsittelyä, IoT-yhteyksiä (Wi-Fi, Bluetooth), äänikäsittelyä ja näyttöjä. [8]

2.4 Mikrokontrollerin ja mikroprosessorin erot

Mikrokontrollerin ja mikroprosessorin ero on siinä, että mikrokontrolleri sisältää kaikki sen toimintaan tarvittavat komponentit yhdellä sirulla, kun taas mikroprosessori tarvitsee ulkoisia komponentteja toimiakseen. Käytännössä mikrokontrolleri sisältää pienen ja vähätehoisen mikroprosessorin. Muita merkittäviä eroja mikrokontrollerin ja mikroprosessorin välillä ovat hankintahinta, laskentanopeus ja virrankulutus. [11]

Mikrokontrollerin hinta on halvempi kuin mikroprosessorin, koska siinä on kaikki tarvittava mukana yhdellä sirulla. Mikroprosessori tarvitsee paljon ulkoisia komponentteja toimiakseen, mikä nostaa sen hankintahintaa.

Mikroprosessoreissa laskentanopeudet ovat suurempia kuin mikrokontrollereissa. Mikroprosessoreiden kellotaajuudet ovat yleensä gigahertseissä ja mikrokontrollereiden megahertseissä. Laskentanopeus kertoo kuinka paljon tietoa prosessori voi käsitellä sekunnin aikana. Mikrokontrollerit on optimoitu suorittamaan tiettyjä tehtäviä juuri tarvittavalla nopeudella ja virrankulutuksella. Pienen kokonsa ansiosta mikrokontrolleri voidaan asentaa lähemmäs ohjattavaa komponenttia, mikä joissain tapauksissa voi tarkoittaa nopeampaa toimintaa verrattuna mikroprosessoriin. Mikroprosessorit ovat tarkoitettu tehtäviin, jotka ovat monimutkaisia ja tarvitsevat paljon ennalta arvaamatonta laskentatehoa. [11]

Mikrokontrollereilla on selvä etu virrankulutuksessa. Koska mikrokontrollerit on optimoitu määritettyjen tehtävien suorittamiseen, vaativat ne vähemmän virtaa. Mikroprosessorit ovat tehokkaita ja nopeita, joka luo tarpeen enemmälle virralle ja voi vaatia kytkemistä ulkoiseen virtalähteeseen, kun taas mikrokontrolleri voi toimia pitkään pienellä akulla. [11]

3 Wi-Fi

Wi-Fi on langaton lähiverkkoteknologia, joka mahdollistaa eri laitteiden välisen kommunikaation ja mahdollistaa laitteiden yhdistämisen internetiin ilman fyysisiä kaapeleita. [12]

Wi-Fi on tuotemerkki IEEE 802.11-standardille. Se kehitettiin, koska standardille haluttiin käyttäjäystävällisempi nimi. Nykyisin nimellä Wi-Fi Alliance tunnettu yhdistys perustettiin ylläpitämään Wi-Fi-tuotemerkkiä ja valvomaan Wi-Fi nimen alla myytävien tuotteiden toimintaa ja turvallisuutta [13]. He palkkasivat Interbrand markkinointifirman kehittämään sille paremman nimen. Interbrand kehitti termin Wi-Fi, jonka uskotaan olevan sanaleikki Hi-Fi-termistä. [14] Hi-Fi-termiä käytetään kuvaamaan korkealaatuisia äänilaitteita ja tarkoittaa "high fidelity" korkeaa luotettavuutta. Wi-Fi ei ole lyhenne mistään, mutta myöhemmin Wi-Fi-termille on keksitty tarkoitus ja sen ajatellaan olevan lyhenne "wireless fidelity", joka tarkoittaa langatonta luotettavuutta. [15]

3.1 Wi-Fin historia

Ensimmäinen langatonverkko oli ALOHAnet. Se kehitettiin Havaijin yliopistossa vuonna 1971. ALOHAnet todisti langattomien yhteyksien mahdollisuuden ja toimi esimerkkinä nykyiselle Wi-Fi-teknologialle. [16]

Wi-Fin historia alkaa vuodesta 1985, kun FCC (Federal Communications Commission) vapautti taajuuden, jota voi käyttää ei-rekisteröityyn käyttöön. [17] Vapautettu taajuus oli 2,4 GHz, mikä on sama taajuus, jota mikroaaltouunit käyttävät. [14]

Vuonna 1990 perustettiin IEEE 802.11 työryhmä langattomalle paikalliselle verkolle. Yksi tärkeä tekijä ryhmän perustamisessa oli Victor Hayes. Häntä kutsutaan Wi-Fin isäksi. [18] Ensimmäinen IEEE 802.11-standardi julkaistiin vuonna 1997, mikä mahdollisti 2 Mbit/s langattoman tiedonsiirron. [19]

Vuonna 1999 Apple julkaisi AirPort langattoman reitittimen ja iBook kannettavan tietokoneen, jotka käyttivät samaan aikaan julkaistua IEEE 802.11-teknologiaa. Applen ansiosta IEEE 802.11-teknologian suosio kasvoi ja aloitti langattoman vallankumouksen. [19]

Vuonna 1999 useat yritykset kokoontuivat ja perustivat voittoa tavoittelemattoman yhdistyksen nimeltä WECA. WECA-yhdistyksen tarkoituksena on taata paras käyttäjäkokemus laitteesta huolimatta, joka käyttää IEEE 802.11-teknologiaa. [20] Vuonna 2000 yhdistys palkkasi Interbrand-yrityksen keksimään paremman nimen IEEE 802.11-teknologialle. He keksivät Wi-Fi-termin ja IEEE 802.11 uudelleen brändättiin. Yhdistys vaihtoi myös nimensä Wi-Fi Allianceksi. [21]

3.2 Wi-Fin toiminta

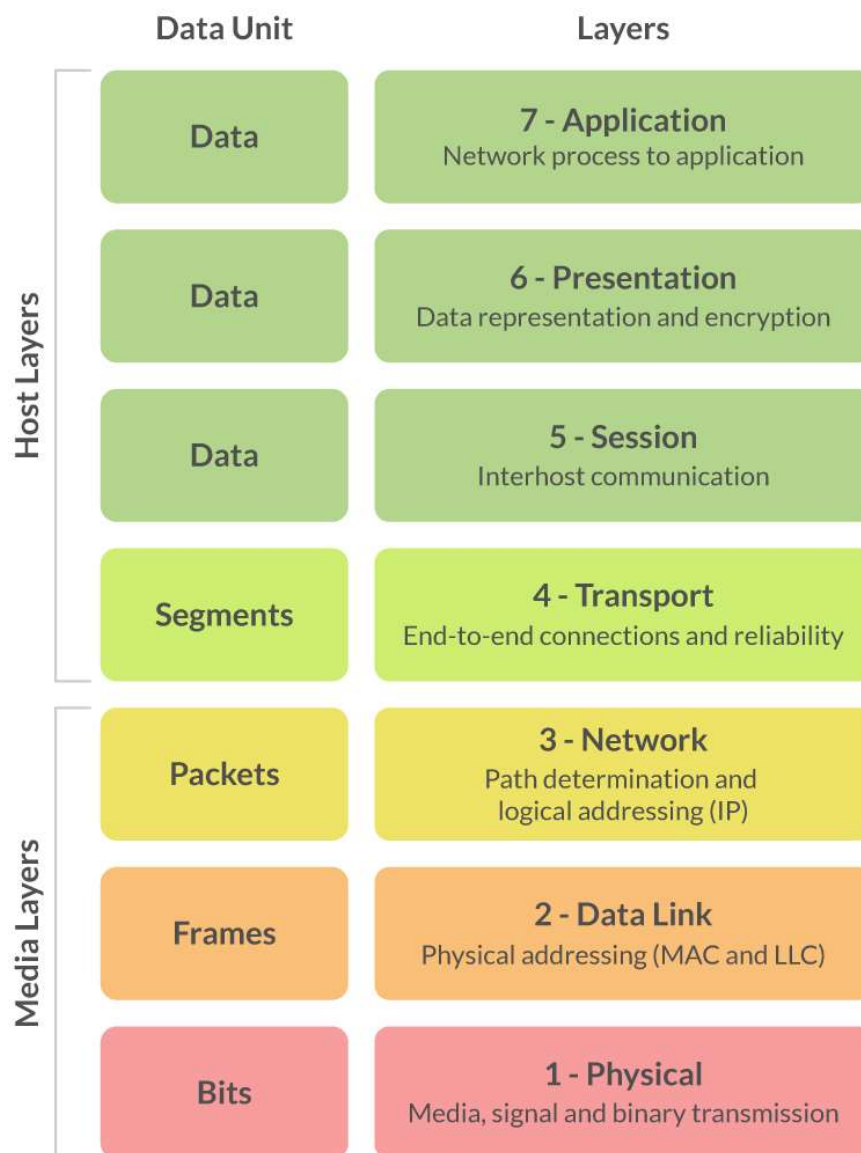
Wi-Fi lähettää dataa radiotaajuuksien avulla. Aluksi se käytti vain 2.4 GHz:n taajuutta, mutta nykyään se toimii monella eri taajuudella. Käytetyimmät taajuudet ovat 2.4 GHz ja 5 GHz. [21] Laitteiden Wi-Fi-kortit muuttavat tiedon radioaalloiksi ja lähettävät tiedot antennin avulla toiselle laitteelle. [22]

IEEE 802.11b tai Wi-Fi 1 implementoi modulaatiojärjestelmiä, joita kutsutaan DSSS/CCK (direct-sequence spread spectrum/complementary code keying). DSSS/CCK mahdollisti Wi-Fin nopeamman tiedonsiirron ja vähensi häiriöitä, jotka johtuivat samaa aallonpituutta käyttävistä laitteista, kuten mikroaaltouunit ja itkuhälyttimet. Koska Wi-Fi käyttää samaa aallonpituutta tiedonsiirtoon kuin mikroaaltouunit ja itkuhälyttimet, niin DSSS/CCK:n käyttö vähensi myös näistä johtuvia häiriöitä. [19]

Wi-Fin toimintakantavuus sisätiloissa on 10–35 m, mutta avoimissa tiloissa se voi olla jopa yli 100 m. Kantavuserot johtuvat esteistä, kuten seinistä, huonekaluista ja muista fyysisistä esteistä. Jotkut materiaalit, kuten paksut betoni- tai metalliseinät saattavat estää Wi-Fi-signaalin kokonaan. Kantavuus riippuu myös lähetystehosta ja käytettävästä taajuudesta. Korkeammilla taajuuksilla, esimerkiksi 5 GHz:n taajuudella, on lyhyempi lähetyskantavuus

kuin 2,4 GHz:n taajuudella, mutta korkeammat taajuudet tarjoavat nopeamman tiedonsiirron. [16] [23]

Kahden laitteen välisen tiedon siirtämiseen Wi-Fi käyttää TCP/IP-protokollaa. TCP/IP koostuu kahdesta eri protokollasta TCP (Transmission Control Protocol) ja IP (Internet Protocol). IP-protokolla kuuluu kuvassa 1 kuvatun ISO/OSI-mallin tasolle 3 (verkko) ja TCP-protokolla tasolle 4 (siirto) [24].



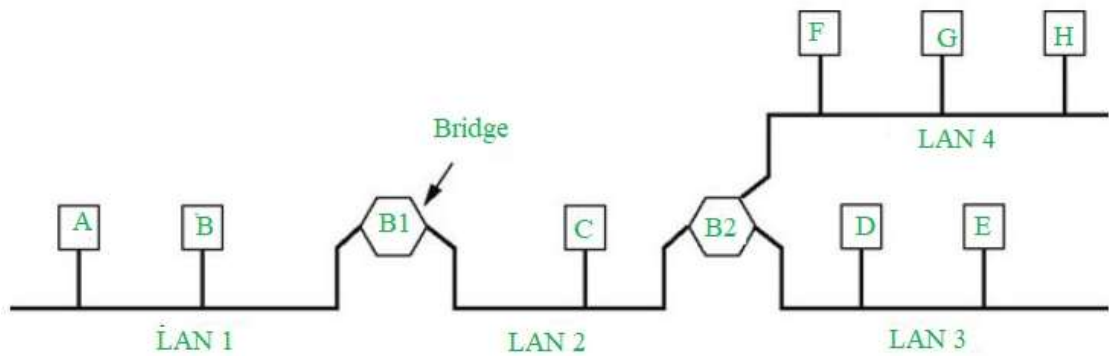
Kuva 1. ISO/OSI-malli.

IP:n tehtävänä on löytää laite, johon tieto lähetetään ja TCP:n tehtävänä on tiedon lähettäminen. TCP/IP-protokolla purkaa tiedon pienempiin osioihin ja järjestellee osiot niin, että vastaanottava laite osaa koota ne oikeassa järjestyksessä takaisin tiedon alkuperäiseen muotoon. Tiedon purkamisen ansiosta koko tietoa ei tarvitse lähettää uudestaan, vaikka osa osioista katoaisi tai vioittuisi. [25] [26]

3.3 Verkkolaitteet

Verkkolaitteita on monenlaisia moneen eri tarkoitukseen. Niitä tarvitaan toimivan verkon luomiseen. Verkkolaitteita ovat muun muassa tukiasema, verkkosilta, yhdyskäytävä, keskitin (engl. hub), modeemi, toistin, reititin ja kytkin. Yritys käytössä voidaan tarvita satoja, ellei jopa tuhansia eri laitteita toimivan verkon luomiseksi, jotta saadaan yrityksen tarpeita vastaava verkko. Yksinkertaisessa kotiverkossa saattaa tarvita vain kahta eri laitetta. [27] Tässä opinnäytetyössä keskitytään verkkosiltoihin ja tukiasemiin.

Verkkosilta on laite, jota käytetään usean verkon yhdistämiseen yhdeksi verkoksi. Verkkosillan avulla verkkoon yhdistetyt laitteet voivat kommunikoida toistensa kanssa. Verkkosilloja on kolmea erilaista ja ne ovat läpinäkyvä-, lähdereitti- ja translaatioverkkosilta. Kuvassa 1. on esimerkki läpinäkyvästä- (B1) ja lähdereittiverkkosillasta (B2) [28]. Läpinäkyviä verkkosilloja käytetään useiden verkkojen yhdistämiseksi toisiinsa. Lähdereittiverkkosilloja käytetään reitin valitsemisessa kahden isäntäkoneen välillä. Translaatioverkkosilloja käytetään verkon muuttamiseksi toisenlaiseksi verkoksi, kuten Ethernet-verkko Wi-Fi-verkoksi. [29]



Kuva 2. Verkkosiltojen esimerkkikuva.

Tukiasema on laite, joka mahdollistaa langattomien laitteiden yhdistyä Wi-Fi-yhteyden avulla lähiverkkoon. Tukiasema luo langattoman lähiverkon WLAN:iin. Yleensä tukiasemaan tuodaan internetyhteys Ethernet-kaapelin avulla, joka mahdollistaa lähiverkkoon yhdistyneiden laitteiden pääsyn internetiin. [30]

3.4 Wi-Fin tietoturva

Tietoturva on tärkeä osa Wi-Fi-verkkoja, koska sen tarkoituksena on suojata verkkoja luvattomalta käytöltä, salakuuntelulta, häirinnältä ja muilta mahdollisilta uhilta. [31] Wi-Fi-verkon tulisi käyttää suojausprotokollaa, mieluiten WPS3:a tai WPS2:ta. Ne salaavat tiedonsiirron. Pitää myös tarkistaa, että palomuuuri on käytössä. Palomuuuri estää tuntemattomien yhteydenottoyritykset. Tämän lisäksi on hyvä rajata sallittujen yhteyksien määrä. Lisätoimena voi käyttää VPN:ää, joka salaa myös tiedonsiirron. [32]

On myös välttämätöntä vaihtaa suojausprotokollan käyttämä oletussalasana vahvaan salasanaan, joka sisältää yhdistelmän pieniä ja isoja kirjaimia sekä numeroita ja erikoismerkkejä. Salasanassa pitää välttää yleisten sanojen tai termien käyttämistä, kuten "salasana", sekä henkilökohtaisten tietojen käyttämistä, kuten nimiä tai osoitteita. [32]

3.4.1 Wi-Fin uhat ja riskit

Koko ajan lisääntyneen Wi-Fi-verkon käytön myötä, myös uhkien ja riskien määrää on kasvanut. Yleisin riski Wi-Fi-verkossa on laitteen yhdistäminen julkiseen tai niin sanottuun avoimeen verkkoon. Avoimet verkot ovat suojaamattomia ja ne altistavat laitteet haitta- ja vakoiluohjelmille sekä salakuuntelulle ja muille uhille. Avoin verkko on nimensä mukaan avoin, johon kuka tahansa voi yhdistyä ja nähdä sen kautta tapahtuvan tietoliikenteen. Tämä ei tarkoita, että kodin Wi-Fi-verkko olisi täysin turvallinen, jollei tarvittavia suojoitoksia ole tehty. Näitä ovat suojausprotokollien ja palomuurin käyttäminen. [31]

Wi-Fin tietoturvaan liittyviä uhkia ovat muun muassa salasanahyökkäykset, väliintulohyökkäykset (MITM), "rogue access point" -hyökkäykset, salakuuntelut ja palvelunestohyökkäykset (DoS). "Rogue access points" ovat valvomattomia Wi-Fi-tukiasemia, jotka ovat tunkeilijan hallinnassa. [32]

Salasanahyökkäyksessä tunkeilija koittaa arvata tai murtaa Wi-Fi-verkon salasanan. Väliintulohyökkäyksessä tunkeilija tunkeutuu kahden laitteen yhteyden väliin, minkä jälkeen tunkeilija pystyy varastamaan arkaluonteista tietoa tai asentamaan haitta- tai vakoiluohjelmia. "Rogue access point" -hyökkäyksessä tunkeilija luo luotettavan näköisen tukiaseman. Tukiasema on tunkeilijan hallinnassa ja tunkeilija näkee kaiken tietoliikenteen mitä tukiaseman kautta tehdään. [32]

Salakuuntelu Wi-Fi-yhteydessä tunkeilija kuuntelee kahden laitteen kommunikointia Wi-Fi-verkossa. Tässäkin hyökkäyksessä tunkeilija näkee laitteiden välillä kulkevan tiedon ja saa varastettua arkaluonteisia tietoja. Palvelunestohyökkäys tapahtuu, kun tunkeilija ylikuormittaa verkkoa, jolloin verkosta tulee käyttökelvoton. [32]

3.4.2 Suojausprotokollat

Langattomille yhteyksille on neljä pääasiallista suojausprotokollaa. Ne ovat WEP, WPA, WPA2 ja WPA3. Nämä protokollat on kehittänyt Wi-Fi Alliance. Protokollat mahdollistavat yksityisyyden ja luottamuksellisuuden langattomilla yhteyksillä, mikä toteutetaan hyödyntäen turvallisuustoimintoja, kuten salaus ja todennus. [33]

WEP oli ensimmäinen protokolla Wi-Fi-verkon turvaamiseen. WEP julkaistiin vuonna 1997 ja Wi-Fi Alliance lopetti sen tukemisen vuonna 2004. WEP salaa Wi-Fi-verkon datan 64- tai 128-bittisen heksadesimaaliavaimen avulla. Ongelma WEP:n kanssa oli se, että WEP salasi kaikkien laiteiden yhteydet samalla avaimella. Saman avaimen käyttäminen esti verkon ulkopuolella olevia käyttäjiä näkemästä tietoja, mutta jos tunkeilijalla oli pääsy Wi-Fi-verkkoon, se saattoi nähdä kaikkien verkossa olevien laiteiden tietoliikenteen. Ajan myötä, kun tietokoneiden laskentatehot kasvoivat, tunkeilijat löysivät heikkouksia salauksessa ja pystyivät helposti murtamaan salauksen. Nykyään WEP-protokollaa pidetään turvattomana ja sen käyttöä pitäisi välttää. [34]

WPA julkaistiin vuonna 2003, ja se korvasi WEP:n. WPA käyttää temporaalisen avaimen eheysprotokollaa (TKIP), joka luo uuden salausavaimen jokaiselle tietopakettille, joka lähetetään siinä verkossa. TKIP esti tunkeilijoiden oman avaimen luomista. WPA käyttää 128-bittistä salausavainta. WPA myös tarkistaa onko joku katsonut tai muuttanut tietopakettia lähetyksen aikana. [34]

WPA2 on päivitetty versio WPA:sta ja se julkaistiin vuonna 2004. WPA2 korvasi TKIP:n salauslohkoketjutusviestien todennuskoodiprotokollalla (CCMP). CCMP käyttää AES:sää salauksessa, joka on parempi kuin TKIP:n käyttämä RC4. WPA2 käyttää myös vahvaa todennusta, joka toimii kahdella eri asetuksella. Ne ovat henkilökohtainen asetus (PSK) ja yritysasetus (EAP). PSK-asetus käyttää jaettua salasanaa, jonka vain tukiasema ja yhdistettävä laite tietävät. WPA2-protokollaa käytetään yleensä kotiverkoissa. EAP-asetus käyttää vahvistuspalvelinta laitteiden ja käyttäjien vahvistamiseen. [34]

WPA3 julkaistiin vuonna 2018. Se lisäsi turvallisuustoimintoja molempiin WPA2:n käyttämiin asetuksiin sekä korjasi KRACK-heikkouden, joka huomattiin vuonna 2017. WPA3 käyttää 256-bittistä salausavainta ja AES-salausta, joka toteutetaan SAE-protokollaa käyttäen. SAE-protokolla parantaa suojausta offline-hyökkäyksiä vastaan ja suojaa salasana-arvausyrityksiltä, sillä SAE-protokolla käyttää vahvempia kryptografia-algoritmeja ja turvallisempaa avaimenvaihtometodia. [34]

4 Suunnittelu

4.1 Vaatimukset

Opinnäytetyön tarkoituksena on luoda ratkaisu, jonka avulla Ethernetin kautta kulkeva tieto muutetaan käyttämään langatonta verkkoa. Tärkein vaatimus toteutettavalle ratkaisulle on, ettei ilmanvaihtokoneisiin tarvitse tehdä rakenteellisia tai ohjelmallisia muutoksia. Muita vaatimuksia ovat ilmanvaihtokoneen käyttöönoton helppous ja että käyttöönoton saa tehtyä langattomasti älypuhelimella.

Toteutettavien vaatimusten valitsemiseen käytettiin MoSCoW-menetelmää, joka auttaa määrittämään tärkeimmät toiminnot ja vaatimukset. MoSCoW-menetelmässä vaatimukset luokitellaan neljään eri luokkaan: pakko olla, pitäisi olla, voisi olla ja ei tarvitse.

Vaatimukset käytiin läpi MoSCoW-periaatteiden avulla toimeksiantajan kanssa ja määriteltiin mitä opinnäytetoteutus pitää sisällään ja kirjattiin vaatimukset (taulukko 2).

Taulukko 2. MoSCoW-taulukko

Pakko olla	<ul style="list-style-type: none"> - Ilmanvaihtokoneen Ethernet-yhteyden muuttaminen Wi-Fi-yhteydeksi. - Ilmanvaihtokoneen käyttöliittymä tulee saada avattua älypuhelimien selaimessa. - Ilmanvaihtokoneen käyttöliittymän avaaminen selaimen helposti ja yksinkertaisesti, ilman että käyttäjällä on erikoisosaamista
Pitäisi olla	<ul style="list-style-type: none"> - Wi-Fi-tukiasemaan yhdistäminen QR-koodin avulla.
Voisi olla	<ul style="list-style-type: none"> - Kehityslevylle kotelon teko esittelyä varten. - Yksinkertainen ohje, joka sisältää QR-koodin.
Ei tarvitse olla	<ul style="list-style-type: none"> - Internet-yhteyden luominen.

Taulukossa 2. olevien vaatimusten mukaan lähdettiin valitsemaan tarvittavia teknologioita ja työkaluja sekä suunnittelemaan toteutusta.

4.2 Langattoman teknologian valitseminen

Älypuhelimissa käytetyt langattomat teknologiat ovat NFC, Bluetooth, Wi-Fi, LTE ja GPS. Nämä teknologiat eroavat toisistaan etäisyyden ja yleisimmän käyttötarkoituksen mukaan. Taulukossa 3. on esitetty eri langattomien teknologioiden eroavaisuudet toisistaan. Paikallisiin yhteyksiin käytetään pääasiassa Bluetoothia ja Wi-Fiä. [35]

Taulukko 3. Älypuhelimien eri langattomat teknologiat ja tiedot.

Tyyppi	Esimerkki teknologia	Kantavuus*	Nopeus*	Yleisimmät käyttötarkoitukset
Välitön läheisyys	NFC	10 cm	200 Kbps	Lähimaksaminen
Lähialue	Bluetooth	10 m	2 Mbps	Lisälaitteiden yhdistäminen
Paikallinen alue	Wi-Fi	100 m	200 Mbps	Internettiin yhdistäminen paikallisesti
Laaja alue	LTE	20 km	20 Mbps	Internettiin yhdistäminen mobiiliverkolla
Maailmanlaajuinen	GPS	20 000 km	N/A	Paikannus palveluihin

Langattoman teknologian valinta tehtiin Wi-Fin ja Bluetoothin välillä, sillä tarkoituksena on luoda langaton ratkaisu, joka toimii älypuhelimien kanssa. Ilmanvaihtokone avaa ohjauskäyttöliittymän selaimeen ja käyttää siinä TCP/IP-

protokollaa, kuten Wi-Fi. Ratkaisuun valittiin Wi-Fi, sillä Bluetooth ei tue TCP/IP-protokollaa.

Wi-Fi on myös turvallisempi ja monipuolisempi, joka mahdollistaa laajemmat jatkokehitys mahdollisuudet verrattuna Bluetoothiin. Bluetoothin merkittävin etu Wi-Fiin verrattuna on sen vähäinen virrankulutus. [36] Ratkaisussa virrankulutusta ei kuitenkaan tarvitse ottaa huomioon, koska laite on kytkettynä jatkuvaan verkkovirtaan.

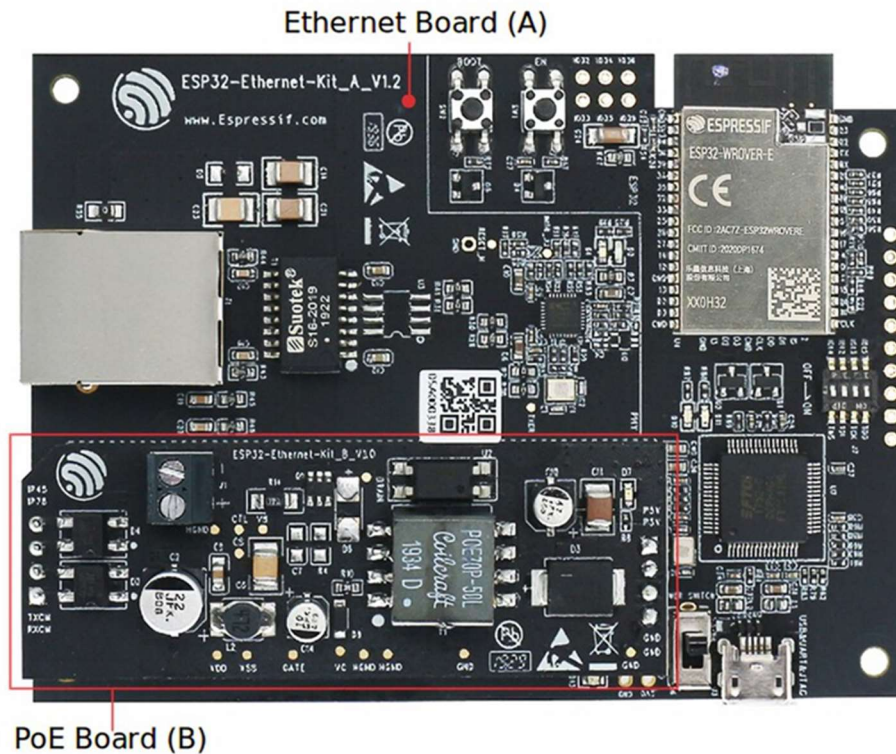
4.3 Mikrokontrollerien vertailu ja valinta

Mikrokontrollerin valintaan vaikuttavia tekijöitä ovat muun muassa projektin tarpeet, mikrokontrollerin saatavuus ja hankintahinta. Opinnäytetyön toteutuksessa mikrokontrollerin vaatimuksena oli, että sen tulee tukea Wi-Fiä ja siinä pitää olla Ethernet-kaapeliliitäntämahdollisuus. ESP32-mikrokontrolleri vastasi parhaiten vaatimuksia, sillä kaikissa ESP32-mikrokontrollereissa on Wi-Fi-toiminnallisuudet sisäänrakennettuna.

ESP32 on edullinen ja vähävirtainen mikrokontrolleriperhe, jota valmistaa kiinalainen Espressif-yritys. Kaikki ESP32-mikrokontrollerit sisältävät Wi-Fi- ja Bluetooth-toiminnallisuudet. ESP32-mikrokontrollerit käyttävät Tensilica Xtensa LX6-mikroprosessoreita, jotka ovat RISC-komentoarkkitehtuuripohjaisia ja käyttävät Harvard-rakennearkkitehtuuria. [37]

Toteutukseen valitsin ESP32 Ethernet-kehityspaketin (kuva 3.) Kehityspaketissa tulee mukana kaksi piirilevyä [38]. A-piirilevyllä on ESP32-WROVER-E-mikrokontrolleri, I/O-portit, Ethernet-liitin ja muut tarvittavat komponentit. A-piirilevyssä on myös microUSB-liitin, jonka kautta

mikrokontrollerille saa ladattua ja testattua ohjelmistokoodin, sekä saa monitoroitua mikrokontrollerin toimintaa.



Kuva 3. ESP32 Ethernet kit.

B-piirilevy on PoE-levy (Power over Ethernet), joka mahdollistaa virran tuomisen A-piirilevylle Ethernet-liittimen kautta. A-piirilevy toimii myös ilman B-piirilevyä. A-piirilevyn voi virroittaa microUSB:n kautta tai piirilevyn voi kytkeä 5 voltin virtalähteeseen.

4.4 Kehitysympäristöt

ESP32-mikrokontrollereiden ohjelmointiin on useita erilaisia ohjelmointivaihtoehtoja, kuten Espruino, MicroPython, ESP-IDF ja ESP Arduino Core. Kaikki mainitut vaihtoehdot ovat avointa lähdekoodia ja ovat vapaasti käytettävissä. [39]

Espruino on JavaScript-ohjelmistokehityspaketti ja laiteohjelmisto, joka simuloi Node.js-toimintaa. Tämä mahdollistaa mikrokontrollereille ohjelmiston kehityksen JavaScript-ohjelmointikielen kanssa. [39] [40]

MicroPython on toteutus Python 3 -ohjelmistokielestä, joka on kirjoitettu C-ohjelmistokielellä. MicroPython sisältää pienen osan Python 3 -peruskirjastosta ja on optimoitu toimimaan mikrokontrolleissa. MicroPython mahdollistaa matalan tason ohjelmistojen luomisen Python-ohjelmistokielen kanssa. [41]

Tutkimisen tuloksena selvisi, että kaksi ohjelmistokehitysvaihtoehtoa olivat suosittumia kuin muut. Ne olivat ESP-IDF ja ESP Arduino Core, joten päätettiin tehdä valinta näiden väliltä.

4.4.1 ESP-IDF

ESP-IDF on ESP32-mikrokontrollerien virallinen ohjelmistokehityspaketti ja on vapaasti käytettävissä. ESP-IDF sisältää erilaisia kirjastoja, työkaluja ja ohjelmointirajapintoja ESP32 -mikrokontrollerin ohjelmistokehitykseen. Koska ESP-IDF on ESP32-mikrokontrollereiden virallinen ohjelmistokehityspaketti ja Espressifin ylläpitämä, ESP-IDF saa kaikki uusimmat toiminnot ensimmäisenä käyttöön. ESP-IDF tarjoaa myös laajan valikoiman toiminnallisuuksia varsinkin esineiden internetin (IoT) tarpeisiin. ESP-IDF käyttää standardi C- ja C++ -ohjelmistokieliä, mikä mahdollistaa näiden ohjelmistokielen toimintojen ja ominaisuuksien hyödyntämisen. ESP-IDF:lla on lisäosat Eclipselle ja Visual Studio Codelle, mikä takaa ketterän ja helppokäyttöisen ohjelmistokehityksen kokemuksen. [42] [43]

4.4.2 ESP Arduino Core

ESP Arduino Core on ohjelmistokehityspaketti, joka on suunniteltu yksinkertaistamaan ESP32-mikrokontrollereiden ohjelmistokehitystä ja on vapaasti käytettävissä. ESP Arduino Core käyttää Arduino-ympäristöä ja ohjelmistokieltä, joka on yksinkertaistettu toteutus C ja C++ -ohjelmistokielistä.

Arduino-ohjelmistokieli käyttää yksinkertaisempia syntakseja kuin standardi C ja C++, ja se on suunniteltu vastaamaan mikrokontrollereiden ohjelmointitarpeisiin. Arduino-ohjelmointikieli ei käytä kaikkia C ja C++ ominaisuuksia ja toiminnallisuuksia, mikä voi rajoittaa kehitystyötä. Arduino Core käyttää Arduino -ympäristöä, joka mahdollistaa Arduino IDE:n käyttämisen. Arduino IDE on Arduino-ympäristölle kehitetty oma yksinkertainen ja käyttäjäystävällinen ohjelmointiympäristö. [42]

4.4.3 Kehitysympäristön valitseminen

Itselläni oli hyvin vähän kokemusta ESP-IDF- ja ESP Arduino Core-kehitysympäristöistä. Ensin harkittiin ESP Arduino Coren valitsemista, koska ESP Arduino Coren etuna oli sen aloittelijaystävällisyys ja kattava dokumentaatio ja muuta apua löytyi myös paljon. Lisätutkimuksen jälkeen huomattiin, että Espressif on dokumentoinut ESP-IDF:n toiminnot hyvin ja internetistä löytyy myös paljon tietoa sen kehitysmahdollisuuksista.

Tutkimusten myötä valittiin kehitysympäristöksi ESP-IDF:än, koska siinä on enemmän toiminnallisuuksia kuin Arduino Coressa. Lisäksi ESP32 virallisena ohjelmistokehityspakettina se saa kaikki uusimmat toiminnot ensimmäisenä käyttöön. ESP-IDF käyttää standardi C- ja C++-ohjelmointikieliä, mikä mahdollistaa tehokkaan koodin kirjoittamisen eikä aiheuta rajoituksia koodia kirjoittaessa.

4.5 Toteutuksen suunnitteleminen

Ennen toteutusvaihetta tutustuttiin paremmin ESP32-mikrokontrolleriin ja ESP-IDF-kehitysympäristöön sekä mietittiin, kuinka toteutuksessa tehtävän verkkosovittimen tulee toimia. Listattiin vaiheet, joita käyttäjän pitää tehdä ennen kuin hän pääsee tekemään ilmanvaihtokoneen käyttöönoton.

1. Verkkosovittimen käynnistäminen.
2. Ilmanvaihtokoneen yhdistäminen verkkosovittimeen Ethernet-kaapelin kanssa.
3. Älypuhelimien yhdistäminen verkkosovittimen tukiasemaan QR-koodin avulla.
4. Ilmanvaihtokoneen käyttöliittymän avaaminen selaimen.
5. Ilmanvaihtokoneen käyttöönoton tekeminen.

Ensimmäiseksi käyttäjän pitää kytkeä verkkosovitin virtalähteeseen. Tämän jälkeen sovitin käynnistää itsensä ja alustaa tarvittavat rajapinnat ja toiminnot.

Seuraavaksi yhdistetään ilmanvaihtokone verkkosovittimeen Ethernet-kaapelin kanssa. Kun ilmanvaihtokone on yhdistetty, verkkosovittimen tulee antaa ilmanvaihtokoneelle IP-osoite. IP-osoite määritetään DHCP-palvelimen avulla, koska ilmanvaihtokone odottaa vastausta DHCP-palvelimelta.

Seuraavaksi käyttäjän pitää yhdistää verkkosovittimeen älypuhelin tai muu laite, jossa on verkkoselain ja joka pystyy yhdistymään Wi-Fiin. Yhdistetylle laiteelle annetaan IP-osoite DHCP-palvelimelta.

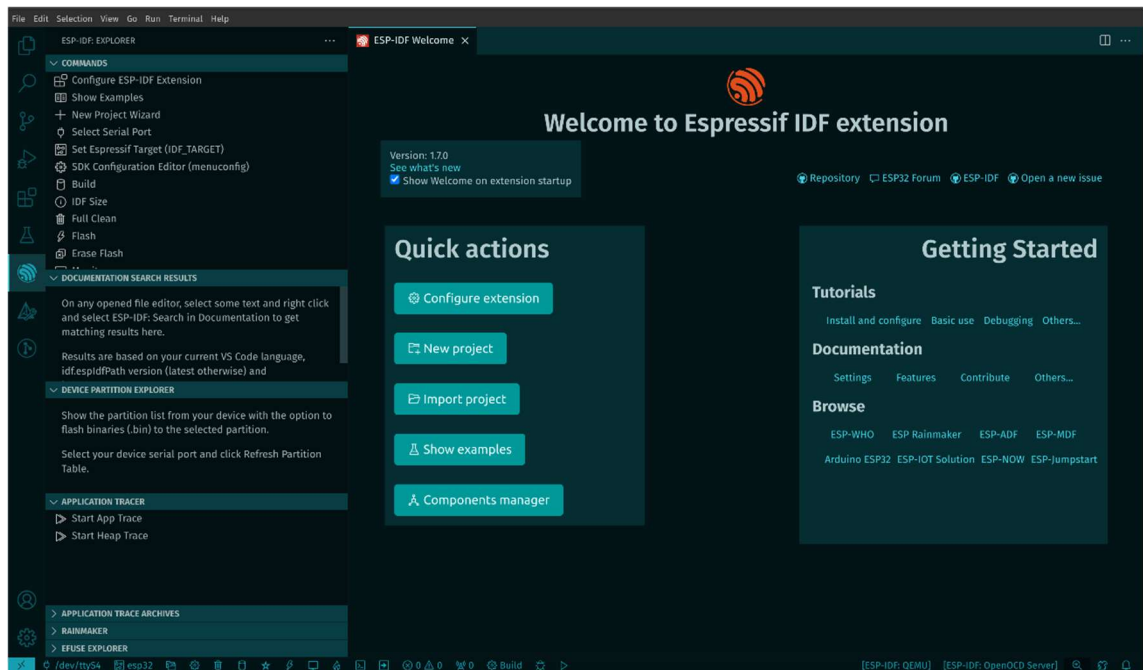
Viimeisessä vaiheessa käyttäjä avaa selaimessa ilmanvaihtokoneen käyttöliittymän käyttäen ilmanvaihtokoneelle määritettyä IP-osoitetta. Kun käyttäjä on saanut ilmanvaihtokoneen käyttöliittymän auki selaimen, hän voi aloittaa ilmanvaihtokoneen käyttöönoton.

5 Toteutus

5.1 Ympäristön asentaminen

ESP-IDF:ä asentaminen onnistui helposti Espressifin verkkosivulta löytyneiden ohjeiden avulla. Verkkosivulta löytyi asennusohjeet kaikille yleisimmille käyttöjärjestelmille (Windows, macOS ja Linux), sekä asennusohjeet Espressifin Eclipseen ja Visual Studio codeen tarjoamien lisäosien asentamiseen. ESP-IDF:ää pystyy käyttämään myös ilman lisäosia. Verkkosivulta löytyi käyttöohjeet ESP-IDF:n käyttöön ilman lisäosia sekä lisäosien kanssa. Tähän toteutukseen valittiin Ubuntu pohjainen Linux-käyttöjärjestelmä ja Visual Studio code ja siihen sopiva lisäosa.

Kuvassa 4. on Visual Studio-lisäosa, jonka ansiosta ei tarvinnut muistaa erilaisia konsolikomentoja ja se nopeutti kehitystyötä pikanäppäimien ansiosta. Lisäosa mahdollisti myös helpon koodin virheenkorjaamisen Visual Studio codessa.



Kuva 4. Visual Studio code-lisäosa.

5.2 Ohjelmointi

5.2.1 Projektin luominen ja rajapintojen määrittäminen

Toteutuksen ohjelmointivaiheessa ensimmäisenä luotiin uusi ESP32-projekti, joka onnistui helposti ESP-IDF Visual Studio code-lisäosan avulla. Kun projekti oli luotu, valittiin kohdelaite. Laitteen valinta onnistui helposti lisäosan avulla ja se loi määrittelytiedoston. Tässä tiedostossa määritetään eri GPIO:den arvot ja muut laiteen tarvitsemat arvot, sekä määrittellään IDE:lle laitteesta tarvittavat tiedot.

Ensimmäisenä määritettiin rajapinnat. Verkkosovittimessa tarvittavat rajapinnat olivat Ethernet ja Wi-Fi. Seuraavaksi määriteltiin Ethernet- ja Wi-Fi-rajapinnat ja niiden toiminnot. Wi-Fi määritettiin toimimaan tukiasemana, mikä mahdollistaa toisten Wi-Fi-laitteiden yhdistämisen tukiasemaan. Kun molemmat rajapinnat oli määritetty, mietittiin, miten rajapintojen välinen kommunikaatio tapahtuisi.

DHCP-palvelinta tarvittiin älypuhelimien ja ilmanvaihtokoneen IP-osoitteiden määrittämiseen. ESP-IDF:ssä oli mahdollista yhdistää Ethernet ja Wi-Fi-rajapinnat yhdeksi rajapinnaksi, mikä mahdollistaa rajapintojen välisen kommunikaation sekä yhden DHCP-palvelimen käytön.

5.2.2 IP-osoitteiden määrittäminen

DHCP-palvelin tuotti myös ongelmia. DHCP-palvelin määrittää yhdistyneille laitteille kasvavan IP-osoitteen yhdistymisjärjestyksessä. Eli ensimmäiselle yhdistetylle laitteelle annetaan IP-osoite 192.168.4.2 ja seuraavalle 192.168.4.3 ja niin edelleen. Mikäli ilmanvaihtokonetta ei yhdistetä ensimmäisenä verkkosovittimeen, käyttäjä ei voi olla varma ilmanvaihtokoneen IP-osoitteesta.

Suunnitteluvaiheessa ajateltiin, että ilmanvaihtokoneelle olisi voinut antaa muuttumattoman IP-osoitteen, joka olisi annettu Ethernet-rajapintaan liitettylle laitteelle. Tämä kuitenkin ei onnistu, koska Ethernet-rajapinta on yhdistetty Wi-Fi-rajapinnan kanssa yhdeksi rajapinnaksi. Koska rajapinnat toimivat

yhtenäisenä rajapintana ja käyttävät yhteistä DHCP-palvelinta, eri rajapintoihin liitettyjen laitteiden tunnistaminen ei onnistu. Tutkimisen ja testaamisen jälkeen keksittiin kaksi eri ratkaisua ilmanvaihtokoneen IP-osoitteen selvittämiseksi.

Ensimmäinen vaihtoehto oli katsoa, milloin Ethernet-porttiin liitetään ilmanvaihtokone ja tallentaa seuraava DHCP:n määrittämä IP-osoite Ethernet käyttöön. Ethernet-ajureiden avulla saatiin tietoon, koska Ethernet-porttiin liitetään ilmanvaihtolaite. Ethernet-ajuri määrittää silloin Ethernet-portille MAC-osoitteen, jolloin tiedetään, että seuraava IP-osoite on tarkoitettu Ethernet-porttiin liitetulle laitteelle. Huono puoli tässä vaihtoehdossa on se, että verkkosovitin tallentaa muistiin minkä tahansa laitteen IP-osoitteen, joka liitetään Ethernet-porttiin, vaikka laite ei olisi ilmanvaihtokone. Tämä saa näyttämään, että ilmanvaihtokone olisi liitetty, vaikka oikeasti se ei olekaan.

Toinen vaihtoehto oli käyttää SSDP (Simple Service Discovery Protocol) -protokollaa. SSDP lähettää kaikkiin verkkoon yhdistettyihin laitteisiin "M-SEARCH" viestin, mikä ilmoittaa muille laitteille, että se etsii toisia verkkoon liitettyjä laitteita. Laitteet vastaavat lähetettyyn viestiin, jos ne täyttävät viestissä lähetetyt vaatimukset, kuten laitteen palvelun tyyppi, laitteen tyyppi ja UUID (Universally Unique Identifier) täsmäävät viestin vaatimuksiin. Laite vastaa viestiin niiden omilla tiedoilla, kuten IP-osoitteella ja palvelun tyyppillä. [44] SSDP:n avulla saadaan poistettua ensimmäisen vaihtoehdon ongelma. Näin pystytään varmistamaan, että liitetty laite on Vallox-ilmanvaihtokone. Huono puoli on se että, SSDP on huomattavasti monimutkaisempi toteuttaa verrattuna ensimmäiseen vaihtoehtoon, sekä ei voida olla täysin varmoja, että tämä toimisi tässä toteutuksessa.

Molempien vaihtoehtojen hyviä ja huonoja puolia vertailtiin ja päätettiin valita ensimmäinen vaihtoehto, koska se oli varmempi ja yksinkertaisempi toteuttaa. Valittu ratkaisu näyttää minkä tahansa Ethernet-porttiin liitetyn laitteen ilmanvaihtokoneena, mutta tätä ei nähty suurena ongelmana, koska verkkosovitinta käyttävät ilmanvaihtokoneiden asentajat. Asentajat käyttävät verkkosovitinta pääasiassa kohteissa, joissa ei ole muita liitettäviä laitteita. Jos tulevaisuudessa ongelmia ilmenee, niin voidaan toista ratkaisua tutkia.

5.2.3 Ilmanvaihtokoneen käyttöliittymän avaaminen

Seuraavaksi piti ratkaista ongelma, miten käyttäjä saa ilmanvaihtokoneen IP-osoitteen tietoonsa. Koska verkkosovittimessa ei ole näyttöä, missä IP-osoitteen olisi voinut näyttää, niin piti keksiä joku toinen keino. Tähän ongelmaan keksittiin ratkaisuksi verkkosivun luominen tukiasemaan. Verkkosivun saa auki, kun laite on yhdistetty tukiasemaan ja kirjoitetaan 192.168.4.1 IP-osoite selaimen. Verkkosivulle tulee näkyviin Ethernet-porttiin liitetyn laitteen IP-osoite, kuten kuvassa 5. näkyy. Tämän avulla käyttäjä saa avattua ilmanvaihtokoneen käyttöliittymän.

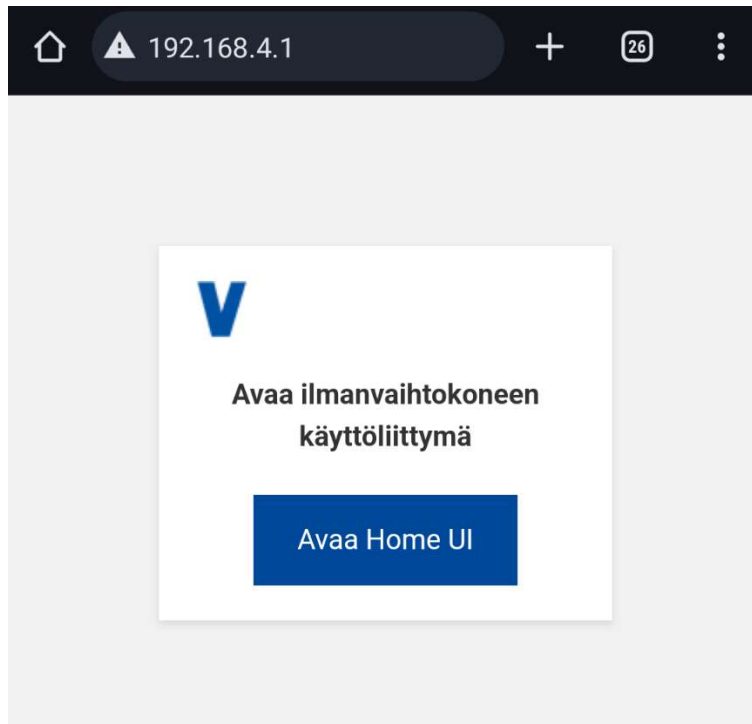


Kuva 5. Kuva verkkosivusta, jossa näkyy ilmanvaihtokoneen IP-osoite.

5.2.4 Verkkosovittimen verkkosivun parantaminen

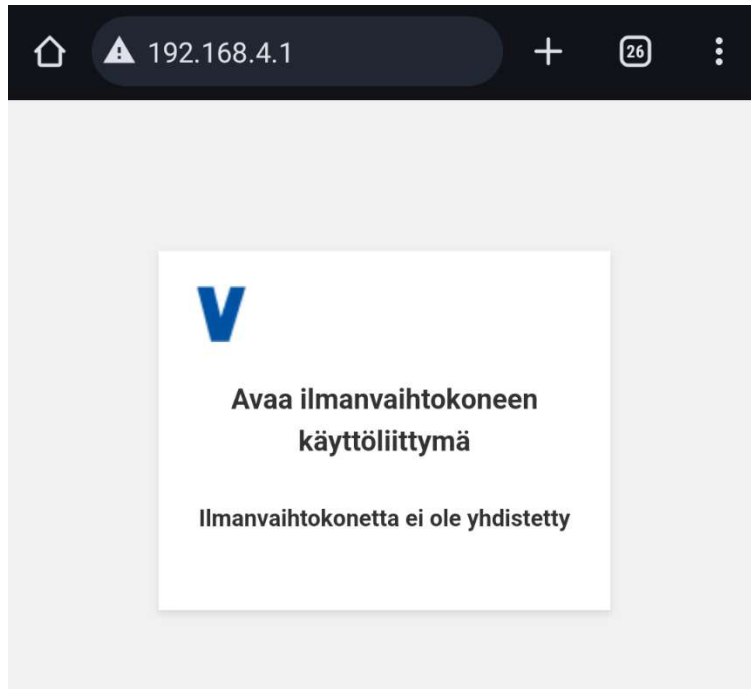
Luodulle verkkosivulle lisättiin HTML- (HyperText Markup Language) ja CSS (Cascading Style Sheets) -koodia, jotta sivu olisi selkeämpi ja helppokäyttöisempi, sekä samalla parannettiin verkkosivun ulkoasua. Ilmanvaihtokoneen IP-osoitteesta tehtiin linkki, jonka ansiosta käyttäjän ei enää tarvitse kirjoittaa IP-osoitetta selaimen ja tukee vaatimuksena ollutta helppokäyttöisyyttä. Kun käyttäjä painaa kuvassa 6. näkyvää "Avaa Home UI" -painike niin ilmanvaihtokoneen käyttöliittymä aukeaa automaattisesti

selaimeen. Tämä painike tulee näkyviin, jos verkkosovittimeen on liitetty ilmanvaihtokone Ethernet-porttiin.



Kuva 6. Lopullinen toteutus verkkosivusta.

Mikäli Ethernet-porttiin ei ole liitetty ilmanvaihtokonetta, kun verkkosivu avataan, silloin "Avaa Home UI" -painiketta ei ole näkyvässä ja annetaan virheilmoitus. Kuvassa 7 on esitetty verkkosivuston ulkoasu siinä tapauksessa, kun ilmanvaihtokonetta ei ole yhdistetty.



Kuva 7. Verkkosivun ulkoasu, kun ilmanvaihtokone ei ole yhdistetty verkkosovittimeen.

5.3 Lisätoiveiden toteuttaminen

Ohjelmointiosion jälkeen suunniteltiin kotelo kehityslevylle ja yksinkertaista käyttöohjetta verkkosovitteelle. Kehityslevylle tehtiin yhteistyössä Vallox Oy:n kanssa 3D tulostettu kotelo (kuva 8.), mikä sekä suojaa levyä, että antaa tyylikkäämmän ulkoasun.



Kuva 8. Kehityslevylle tehty kotelo.

Verkkosovittimelle suunniteltiin käyttöohje. Kuvassa 9 on kuvattu suunniteltu käyttöohje Wi-Fi-yhteyden muodostamiseen ja verkkosivun avaamiseen.

VALLOX
HOME of FRESH AIR

Wi-Fi-yhteyden määrittäminen

Saat määritettyä Wi-Fi:n
QR-koodin tai alla olevien tietojen avulla
SSID nimi: Vallox AP
Salasana: Vallox12345

Wi-Fi QR-koodi



Nettisivun avaminen

Wi-Fi-yhteyden määrittämisen jälkeen
saat avattua nettisivun
skannaamalla QR-koodin
tai kirjoittamalla selaimen: 192.168.4.1

Nettisivu QR-koodi



Kuva 9. Käyttöohje Wi-Fi-yhteyden muodostamiseen ja verkkosivun avaamiseen.

Vaatimuksena oli, että ohjeen tuli olla yksinkertainen, helposti ymmärrettävä ja mahdollisimman kompakti, jotta se olisi mahdollista esimerkiksi kiinnittää laitteeseen tarralla. Ohjeisiin tuli myös sisällyttää Wi-Fi-yhteyden määrittäminen QR-koodilla sekä tukiasemassa olevan verkkosivun IP-osoite QR-koodina. QR-koodien tarkoituksena oli tehdä verkkosovittimen käytöstä mahdollisimman nopea ja helppo käyttäjälle. Näiden lisäksi ohjeeseen lisättiin tarvittavat tiedot Wi-Fi-yhteyden muodostamiseen ja verkkosivun avaamiseen, jos QR-koodien lukeminen ei ole mahdollista. Ohjeen tekemiseen käytettiin GIMP-kuvankäsittelyohjelmaa ja QR-koodien luomiseen käytettiin qrcodemonkey-verkkosivua.

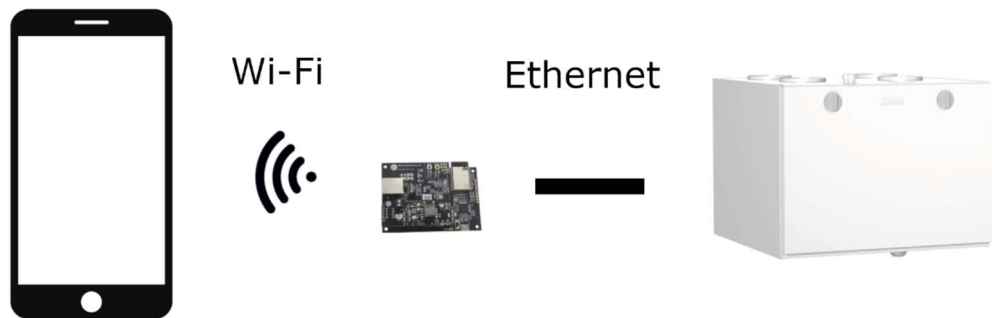
5.4 Testaus

Verkkosovittimen testaukseen tarvittiin ilmanvaihtokone, ESP32 Ethernet-kehityslevy, Ethernet-kaapeli ja älypuhelin. Niiden avulla saatiin testattua verkkosovittimen toiminta ja käytettävyys. Verkkosovittimesta tehtiin seuraavat testit:

1. Ilmanvaihtokoneen ja älypuhelimien liittäminen verkkosovittimeen ja ilmanvaihtokoneen käyttöliittymän avaaminen älypuhelimien selaimessa.
2. Älypuhelimien liittäminen verkkosovittimeen ennen ilmanvaihtokonetta ja käyttöliittymän avaaminen älypuhelimien selaimiin.
3. Verkkosovittimen testaaminen usean eri ilmanvaihtokoneen kanssa.
4. Verkkosovittimen verkkosivun avaaminen usean eri selaimen kanssa. Testattavat selaimet Firefox, Chrome, Safari ja Edge.
5. Verkkosovittimen testaaminen yhdistämällä useita eri laitteita verkkosovittimen tukiasemaan ja verkkosivun päivittäminen samanaikaisesti.

Ensimmäiseksi pystytettiin testausympäristö, josta on kuvaus kuvassa 10. Aluksi ilmanvaihtokoneeseen kytkettiin virta. Tämän jälkeen verkkosovittimeen kytkettiin virta ja liitettiin se ilmanvaihtokoneeseen Ethernet-kaapelin välityksellä. Tämän jälkeen yhdistettiin älypuhelin verkkosovittimen luomaan

tukiasemaan QR-koodin avulla ja avattiin verkkosivu älypuhelimien selaimiin toisen QR-koodin avulla. Verkkosivu aukesi selaimiin, josta saatiin avattua ilmanvaihtokoneen käyttöliittymä, jonka jälkeen saatiin tehtyä ilmanvaihtokoneen käyttöönotto.



Kuva 10. Kuvaus testausympäristöstä.

Testattiin myös älypuhelimien yhdistämistä ennen ilmanvaihtokonetta, jotta saatiin varmistettua, että verkkosivulla olevaan painikkeeseen on osoitettu oikea IP-osoite. Painiketta painamalla saatiin avattua ilmanvaihtokoneen käyttöliittymä, vaikka ilmanvaihtokoneen IP-osoite oli vaihtunut.

Edellä mainitut testit suoritettiin myös useammalla eri ilmanvaihtokoneella, jotta pystytiin varmistamaan, että verkkosovittin toimii kaikkien Vallox-ilmanvaihtokoneiden kanssa. Verkkosovittimen verkkosivua testattiin vielä useamman eri selaimen, kuten Firefoxilla, Chromella, Safarilla ja Edgellä kanssa, jotta nähtiin, toimiiko verkkosivu oikein ja näyttääkö sen ulkoasu asianmukaiselta. Verkkosivu avautui selaimiin oikein ja verkkosivun ulkoasu näytti siltä kuin pitikin.

Ohjelmointivaiheessa oli määritetty, että samanaikaisesti tukiasemaan voi olla yhdistettynä maksimissaan 4 laitetta. Verkkosovittimen toimintaa testattiin niin, että sen luomaan tukiasemaan oli yhdistettynä monta eri laitetta. Samalla testattiin verkkosivun päivittämistä yhtäaikaaisesti kaikilla laitteilla, jotta nähtäisiin, pystyykö verkkosovitin käsittelemään useita käskyjä yhtäaikaisesti. Näin voitiin varmistaa, että verkkosovitin pystyy käsittelemään käskyt oikein ilman verkkosovittimen ohjelmiston kaatumista.

Testauksessa havaittiin, että "Avaa Home UI" -painike ei ilmesty automaattisesti verkkosivulle, jos verkkosivu oli avattu ennen kuin ilmanvaihtokone liitettiin verkkosovittimen Ethernet-porttiin. Verkkosivu piti päivittää uudestaan, jotta verkkosivulle ilmesty "Avaa Home UI" -painike, jonka avulla ilmanvaihtokoneen käyttöliittymän sai avattua. Tätä ei lähdetty muokkaamaan vaan jätettiin jatkokehitykseen. Verkkosovitin kuitenkin toimi tarkoitetulla tavalla.

6 Yhteenveto

Opinnäytetyön tavoitteena oli tehdä ratkaisu, jonka avulla ilmanvaihtokoneen asentaja saisi tehtyä ilmanvaihtokoneen käyttöönoton älypuhelimella. Toteutuksen tuli olla helposti käytettävä eikä ilmanvaihtokoneeseen tarvitsisi tehdä muutoksia.

Opinnäytetyön alussa käytiin toimeksiantajan kanssa läpi toteutuksen vaatimukset ja rajattiin toteutuksen laajuus. Vaatimusten kirjaamisen ja toteutuksen rajauksen jälkeen tutkittiin tärkeimpiä teknologioita, joita olivat mikrokontrollerit ja Wi-Fi.

Kun teoria ja vaatimukset oli määritelty, valittiin mikrokontrolleri, jota käytetään toteutettavassa verkkosovittimessa. Seuraavaksi selvitettiin sopivimmat kehitysympäristöt ja työkalut verkkosovittimen tekemiseen.

Opinnäytetyön lopputuloksena kehitettiin verkkosovitin, joka täytti kaikki pakolliset vaatimukset ja osan lisätoiveista, kuten kehityslevylle tehty kotelo ja ohje. Verkkosovitinta tullaan jatkokehittämään opinnäytetyön jälkeenkin. Verkkosovittimelle on mietitty varsinaista järjestelmäpiiriä, sillä tällä hetkellä verkkosovitinta on käytetty kehityslevyssä, joka on liian iso ja jossa on turhia ominaisuuksia, joita käyttäjä ei tarvitse. Verkkosovittimeen on myös mietitty mahdollisuutta, että se voidaan liittää toiseen tukiasemaan. Se mahdollistaisi ilmanvaihtokoneen liittämisen kodin Wi-Fi-verkkoon ja internetiin. Näiden lisäksi myös asentajilta tulleiden palauteiden ja toiveiden perusteella kehitetään uusia toimintoja.

Lähdeluettelo

- [1] Suomen ilmanvaihtosaneeraus, "ilmanvaihtosaneeraus.fi," [Online]. Saatavilla: <https://ilmanvaihtosaneeraus.fi/sisailma/ilmanvaihdon-merkitys/>. [Haettu 8.4.2024].
- [2] Ilmanvaihtokone, "Wikipedia," [Online]. Saatavilla: <https://fi.wikipedia.org/wiki/Ilmanvaihtokone>. [Haettu 25.2.2024].
- [3] Vallox, "Vallox techmanuals," [Online]. Saatavilla: <https://vallox.techmanuals.info/ValloxMV/FIN/onlinehelp/webhelp/kayttoonotto.html>. [Haettu 25.2.2024].
- [4] R. Keim, "All About Circuits," 25.3.2019. [Online]. Saatavilla: <https://www.allaboutcircuits.com/technical-articles/what-is-a-microcontroller-introduction-component-characteristics-component/>. [Haettu 25.1.2024].
- [5] RS Online, "RS Online," [Online]. Saatavilla: <https://ie.rs-online.com/web/generalDisplay.html?id=ideas-and-advice/microcontrollers-guide> [Haettu 25.1.2024].
- [6] S. Leibson, "Electronic Engineering Journal," 14.11.2022. [Online]. Saatavilla: <https://www.eejournal.com/article/a-history-of-early-microcontrollers-part-1-calculator-chips-came-first/>. [Haettu 25.1.2024].
- [7] S. Leibson, "Electronic Engineering Journal," 21.11.2022. [Online]. Saatavilla: <https://www.eejournal.com/article/a-history-of-early-microcontrollers-part-2-the-texas-instruments-tms1000/>. [Haettu 25.1.2024].

- [8] V. K, "Embedded Hardware Design," 12.10.2023. [Online]. Saatavilla: <https://embeddedhardwaredesign.com/microcontroller-types-explained-with-examples/>. [Haettu 2.2.2024].
- [9] Teach Computer Science, "Teach Computer Science," [Online]. Saatavilla: <https://teachcomputerscience.com/von-neumann-harvard-architecture>. [Haettu 10.2.2024].
- [10] V. Kanade, "Spiceworks," 14.2.2023. [Online]. Saatavilla: <https://www.spiceworks.com/tech/tech-general/articles/risc-vs-cisc/>. [Haettu 2.2.2024].
- [11] Total Phase, "Total Phase," 3.12.2019. [Online]. Saatavilla: <https://www.totalphase.com/blog/2019/12/microcontroller-vs-microprocessor-what-are-the-differences/>. [Haettu 2.2.2024].
- [12] J. Sharma, "Shiksha Online," 11.11.2023. [Online]. Saatavilla: <https://www.shiksha.com/online-courses/articles/wifi-types-working-and-applications/>. [Haettu 7.2.2024].
- [13] Wi-Fi Alliance, "wi-fi.org," [Online]. Saatavilla: <https://www.wi-fi.org/who-we-are>. [Haettu 5.5.2024].
- [14] Cable Free, "Cable Free," 18.5.2017. [Online]. Saatavilla: <https://www.cablefree.net/wireless-technology/history-of-wifi-technology/>. [Haettu 7.2.2024].
- [15] P. Kirvan, "TectTarget," 11 2023. [Online]. Saatavilla: <https://www.techtarget.com/searchmobilecomputing/definition/Wi-Fi>. [Haettu 7.2.2024].
- [16] J. Martindale, "Digital Trends," 28.3.2021. [Online]. Saatavilla: <https://www.digitaltrends.com/computing/what-is-wi-fi/>. [Haettu 7.2.2024].

- [17] S. Alberts, "LinkedIn," 1.5.2023. [Online]. Saatavilla: <https://www.linkedin.com/pulse/80211-wireless-history-maturity-wpa-shanneece-alberts>. [Haettu 10.2.2024].
- [18] IEEE Computer Society, "computer.org," [Online]. Saatavilla: <https://www.computer.org/profiles/victor-hayes>. [Haettu 10.2.2024].
- [19] IEEE SA, "IEEE SA," 16.5.2023. [Online]. Saatavilla: <https://standards.ieee.org/beyond-standards/the-evolution-of-wi-fi-technology-and-standards/>. [Haettu 10.2.2024].
- [20] Wi-Fi Alliance, "wi-fi.org," [Online]. Saatavilla: <https://www.wi-fi.org/who-we-are/history>. [Haettu 9.2.2024].
- [21] J. Fruhlinger, "Network World," 3.1.2023. [Online]. Saatavilla: <https://www.networkworld.com/article/968819/what-is-wi-fi-and-why-is-it-so-important.html>. [Haettu 9.2.2024].
- [22] Optimum, "Optimum," 26.4.2022. [Online]. Saatavilla: <https://www.optimum.com/articles/internet/what-is-wifi>. [Haettu 8.2.2024].
- [23] B. Mitchell, "Lifewire," 5.11.2020. [Online]. Saatavilla: <https://www.lifewire.com/range-of-typical-wifi-network-816564>. [Haettu 10.2.2024].
- [24] L. Roy, "n0c," [Online]. Saatavilla: <https://kb.n0c.com/en/glossaire/osi/>. [Haettu 20.3.2024].
- [25] T. G. Mesevage, "Datto," 30.4.2019. [Online]. Saatavilla: <https://www.datto.com/blog/what-is-wi-fi-and-how-does-it-work>. [Haettu 22.2.2024].
- [26] GeeksforGeeks, "GeeksforGeeks," 21.7.2023. [Online]. Saatavilla: <https://www.geeksforgeeks.org/tcp-ip-model/>. [Haettu 22.2.2024].

- [27] J. English, "TechTarget," 14.4.2022. [Online]. Saatavilla: <https://www.techtarget.com/searchnetworking/tip/An-introduction-to-8-types-of-network-devices>. [Haettu 13.2.2024].
- [28] Geeks for Geeks, "Geeks for Geeks," 27.10.2022. [Online]. Saatavilla: <https://www.geeksforgeeks.org/what-is-bridge-in-computer-network-types-uses-functions-differences/>. [Haettu 20.3.2024].
- [29] J. Sharma, "Shiksha Online," 31.12.2023. [Online]. Saatavilla: <https://www.shiksha.com/online-courses/articles/bridge-in-computer-network/>. [Haettu 12.2.2024].
- [30] LigoWave, "LigoWave," [Online]. Saatavilla: <https://www.ligowave.com/difference-between-access-point-and-router>. [Haettu 13.2.2024].
- [31] Cisco, "Cisco.com," [Online]. Saatavilla: <https://www.cisco.com/c/en/us/products/wireless/what-is-wi-fi-security.html#~protocols>. [Haettu 13.2.2024].
- [32] IronWifi, "IronWifi," [Online]. Saatavilla: <https://www.ironwifi.com/blog/the-most-common-wifi-security-threats-and-how-to-prevent-them>. [Haettu 13.2.2024].
- [33] NetSpot, "netspotapp.com," [Online]. Saatavilla: <https://www.netspotapp.com/blog/wifi-security/wifi-encryption-and-security.html>. [Haettu 13.2.2024].
- [34] I. Šlekýté, "NordVPN," 5.6.2023. [Online]. Saatavilla: <https://nordvpn.com/fi/blog/wep-vs-wpa-vs-wpa2-vs-wpa3/>. [Haettu 13.2.2024].

- [35] P. O'Farrell, "Forbes," 14.5.2018. [Online]. Saatavilla: <https://www.forbes.com/sites/riverbed/2018/05/14/5-wireless-technologies-in-your-pocket/>. [Haettu 29.2.2024].
- [36] A. Kranz, "Moko Blue," 21.7.2022. [Online]. Saatavilla: <https://www.mokoblue.com/bluetooth-vs-wifi-which-is-better/>. [Haettu 12.2.2024].
- [37] ESP32, "Wikipedia," [Online]. Saatavilla: <https://en.wikipedia.org/wiki/ESP32>. [Haettu 28.2.2024].
- [38] Espressif, "Espressif," [Online]. Saatavilla: <https://docs.espressif.com/projects/esp-idf/en/stable/esp32/hw-reference/esp32/get-started-ethernet-kit.html>. [Haettu 14.3.2024].
- [39] Last Minute Engineers, "Last Minute Engineers," [Online]. Saatavilla: <https://lastminuteengineers.com/getting-started-with-esp32/>. [Haettu 28.2.2024].
- [40] Espruino, "Espruino," [Online]. Saatavilla: <https://www.espruino.com/>. [Haettu 28.2.2024].
- [41] MicroPython, "MicroPython," [Online]. Saatavilla: <https://micropython.org/>. [Haettu 28.2.2024].
- [42] ESPBoards, "ESPBoards," 18.3.2023. [Online]. Saatavilla: <https://www.espboards.dev/blog/esp-idf-vs-arduino-core/>. [Haettu 28.2.2024].
- [43] Espressif, "Espressif," [Online]. Saatavilla: <https://www.espressif.com/en/products/sdks/esp-idf>. [Haettu 28.2.2024].

- [44] Pelco, "Pelco," 5.1.2022. [Online]. Saatavilla:
https://support.pelco.com/s/article/How-a-UPnP-Search-works-1538586696122?language=en_US. [Haettu 2.4.2024].