

Älyruukku - ePlant

Jaakko Kammonen ja Minna Kuivalainen

Tekijä(t) Jaakko Kammonen, Minna Kuivalainen	
Koulutusohjelma Tietojenkäsittelyn koulutusohjelma	
Raportin/Opinnäytetyön nimi Älyruukku - ePlant	Sivu- ja liitesivumäärä 80+0
<p>Tämän opinnäytetyön tavoitteena on luoda saatavilla olevan kirjallisuuden, komponenttien sekä oman kokemuksen pohjalta automatisoitu kastelujärjestelmä, jota voidaan hallinnoida sille rakennetun applikaation kautta. Automatisoitu kastelujärjestelmä ePlant on tarkoitettu harrastetoimintaan ja pikkutarkkuutta vaativien kasvi-intoilijoiden iloksi. Se soveltuu sisäkäyttöön ja pääsääntöisesti ruokayrttien kasvattamiseen. Ruukku saa virtansa 12V-virtalähteestä ja on suunniteltu ikkunalaudalle sopivaksi. Sovellus on toteutettu React Native -mobiilisovelluskehityksellä ja ruukku Arduinon ohjelmointiympäristöä käyttäen. Sovellus käyttää tietokantana Googlen kehittämää Firebase-alustaa, jonka kautta tapahtuu myös käyttäjän autentikointi. Tietokantaan tallentuu käyttäjän omat kasvit ja ruukut, jota sovelluksessa näytetään. Rekisteröityminen tapahtuu ePlant-sovelluksen kautta, jonka jälkeen käyttäjä pääsee kirjautumaan sisään ja aloittamaan ruukkujen sekä kasvien lisäämisen.</p> <p>Applikaation avulla käyttäjä pystyy lisäämään, nimeämään, muokkaamaan sekä poistamaan kasveja ja ruukkuja. Sovelluksessa on mahdollista valita kasvatustekniikaksi joko multaversio tai vesiviljely, jotka määrittelevät mitä dataa yrtistä mitataan ja näytetään. Sovellus ja ruukku kommunikoivat keskenään MQTT-viestiprotokollan avulla ja dataa voidaan hallita avoimeen lähdekoodiin perustuvan ThingSpeakin avulla. Opinnäytetyö on tehty vuonna 2021 kesän ja syksyn aikana.</p> <p>Opinnäytetyö koostuu teoreettisesta viitekehyksestä, varsinaisesta empiriasta sekä pohdintaosiosta. Teoreettisessa osuudessa käydään läpi Internetin syntyä, digitalisaatiota globaalilla tasolla sekä esineiden internetin elinkaarta käyttötarkoituksineen. Teoreettinen osuus on pohja varsinaiselle empirialle, eli kehittämällemme älyruukulle ja mobiiliapplikaatiolle.</p>	
Asiasanat Arduino, ESP32, IoT, JavaScript, MQTT, React	

Sisällys

1	Johdanto	1
1.1	Tutkimusmenetelmä.....	2
1.2	Tutkimuskysymykset ja rajaukset	2
1.3	Keskeiset käsitteet	3
2	Digitalisaatio ja Esineiden Internet.....	5
2.1	Internet.....	6
2.2	World Wide Web	7
2.3	Sosiaalinen verkko	9
2.4	Internet of Things	10
2.4.1	Historia.....	11
2.4.2	Arkkitehtuuri	14
2.4.3	Käyttötarkoitus	15
2.4.4	Kyberturvallisuus.....	22
2.4.5	Tulevaisuus.....	23
2.5	Järjestelmäsiru	25
2.5.1	Arduino Uno	25
2.5.2	ESP32.....	26
	Arduino IDE.....	27
2.6	3D-mallinnus.....	29
2.7	3D-tulostus.....	31
3	Älyruukku	35
3.1	Ruukun rakentaminen	35
3.1.1	3D-mallinnus ja tulostus	36
3.1.2	Komponentit.....	43
3.2	Integraatio.....	48
3.3	Jatkokehitys	51
4	Mobiiliapplikaatio.....	53
4.1	Ohjelmistoarkkitehtuurin suunnittelu.....	53
4.2	Graafinen ilme.....	53
4.3	Integraatio.....	64
4.4	Jatkokehitys	69
5	Pohdinta.....	70
5.1	Johtopäätökset.....	70
5.2	Oman oppimisen arviointi.....	71
	Lähteet	74

1 Johdanto

Tämän opinnäytetyön tarkoituksena on perehtyä esineiden internetin historiaan, nykyyhteen, erilaisiin käyttötarkoituksiin ja sen kautta varsinaiseen toiminnalliseen puoleen, kehittämäämme automatisoituun älyruukkuun. Vaikka esineiden internetistä uutisoidaan valtamediassa hyvinkin usein, on se edelleen suurimmalle osalle tuntematon käsite. Esineiden internetillä tarkoitetaan laitteita, jonka tietoja voidaan lukea ja ohjata langattoman tai kiinteän verkon yli. Verkkoon yhdistetty esine voi olla muun muassa älykello, kahvinkeitin, yksittäinen anturi, joka mittaa lämpötilaa, veden pH-arvoa, mullan kosteutta tai jokin suurempi kokonaisuus, kuten itseohjautuvat ajoneuvot. Esineiden internetiä voidaan hyödyntää laajasti kaikilla eri osa-alueilla, jotka helpottavat ihmisten arkipäiväistä elämää. Suurin arvo muodostuu kuitenkin laitteen fyysisen ympäristön keräämästä datasta, jota voidaan analysoida ja hyödyntää muun muassa tekoälyn kehittämisessä sekä muilla vaativilla osa-alueilla, kuten terveysteknologiassa ja logistiikkasektorilla.

Opinnäytetyön aiheena on React Native -mobiilisovelluskehityksellä toteutettu ePlant mobiiliapplikaatio, joka kommunikoi ESP32:lla ja Arduino ohjelmointiympäristöllä rakennetun IoT-ruukun kanssa. IoT-ruukku on toteutettu 3D-mallinnuksena sekä tulostettu fyysiseksi ruukuksi, johon on asetettu sille tarvittavat komponentit ja anturit. Olemme käyttäneet avoimeen lähdekoodiin perustuvaa ThingSpeakia, joka tallentaa sekä näyttää käyttäjälleen dataa käyttämällä HTTP- ja MQTT-protokollaa Internetin kautta. ThingSpeakista saatu tieto haetaan mobiiliapplikaation avulla, jolloin käyttäjä pystyy näkemään kasvin arvot reaaliaikaisesti omassa puhelimessaan. Datatallennukseen sekä käyttäjien autentikointiin olemme käyttäneet Googlen Firebase-tietokantaa, jonne käyttäjän omat kasvit, ruukut ja henkilökohtaiset tiedot ovat tallennettu. Mobiiliapplikaation puolella käyttäjä pystyy hallinnoimaan lisäämiään ruukkuja ja kasveja kokonaisvaltaisesti.

Opinnäytetyö johdattelee lukijan Internetin luomisen alkulähteitä aina nykypäivän esineiden internetin automatisoituihin järjestelmiin. Opinnäytetyön toiminnallinen puoli käsittelee suunnittelemaamme IoT-älyruukkua ja mobiiliapplikaatiota, jotka kokonaisuudessaan luovat toimivan automatisoidun kastelujärjestelmän.

Tarkoituksenamme on avata lukijalle ruukun teknillistä, koodillista sekä visuaalista puolta ja muodostaa kokonaisvaltainen kuva rakentamastamme integraatiosta. Sovelluksen kanssa kommunikoiva ruukku on mahdollista 3D-tulostaa sille tehdyn mallinnuksen mukaisesti tai vaihtoehtoisesti käyttää vanhaa jo olemassa olevaa ruukkua.

1.1 Tutkimusmenetelmä

Opinnäytetyön runkona käytimme toiminnallista tutkimusmenetelmää, joka on tieteellisestä näkökulmasta katsottuna verrattavissa konstruktiviseen tutkimusmuotoon. Se sisältää teoreettisen viitekehyksen, joka pohjautuu varsinaiseen toiminnalliseen osuuteen (julkaisemaamme tuotteeseen) sekä opinnäytetyöraportin, jossa dokumentoidaan ja arvioidaan varsinainen prosessi. Toiminnallisen opinnäytetyön tarkoituksena on tutkia ja kehittää tuotetta työn toteuttamiseen tehtyjen valintojen sekä ratkaisujen kautta. Opinnäytetyön teoriaosuuden osalta olemme tutkineet olemassa olevaa aineistoa ja suhtautuneet kriittisesti siellä esitettyihin näkökulmiin. Olemme verranneet eri lähteitä samasta aihealueesta, jotta tieto olisi mahdollisimman luotettavaa ja laadukasta.

Tutkimusmenetelmän toiminnallinen osuus painottuu kehittämäämme älyruukkuun, jossa olemme verranneet erilaisia, omaan työhömmе sopivia komponentteja sekä kirjastoja, jotta kokonaisuudesta tulisi selkeä ja tehokas.

1.2 Tutkimuskysymykset ja rajaukset

Opinnäytetyömme on rajattu teoreettisen puolen osalta esineiden internetin historian, nykypäivän, arkkitehtuurin sekä sen erinäisten käyttötarkoitusten käsittelyyn ja analysointiin. Alustuksena käymme läpi ytimekkään ja tiiviin tietopakettin Internetin synnystä, jonka jälkeen teoreettinen osuus etenee lineaarisesti digitaalisen vallankumouksen kautta esineiden internetin maailmaan. Teoreettisessa osuudessa olemme käyneet läpi Internetistä löytyvää aineistoa, aiheeseen liittyviä artikkeleita ja asiantuntijoiden dokumentaatioita.

Kehittämämme älyruukun osalta toiminnallisuudet ovat rajattu mahdollisimman selkeästi, jotta saisimme ammennettua siitä syntyviä tuloksia tehokkaasti asetettujen aikaraamien sisällä. Älyruukku keskittyy pääsääntöisesti yrttien viljelyyn, jossa vaihtoehtoina ovat vesiviljelmä- ja multatekniikka. Tietokannastamme löytyy rajallinen määrä suosituimpia ja käytännöllisimpiä yrttejä, joiden kasvutarpeet ovat esitetty hakunäkymässä. Älyruukku toimii ainoastaan sisätiloissa ja on mitattu ikkunalaudalle sopivaksi. Se sopii muun muassa harrastelijoille sekä käyttäjäkunnille, jotka ovat kiinnostuneita kasvien hoitamisesta ja haluavat seurata siitä saatua dataa reaaliaikaisesti. Mobiiliapplikaation osalta olemme rajanneet toiminnallisuuksien osalta mahdollisimman tarkasti jättäen sille jatkokehitysmahdollisuuden. Applikaation tarkemmat tekniset tiedot löytyvät empiriaosuudesta.

Olemme asettaneet aikaraamit niin toiminnallisen tutkimustyön tekemiseen kuin kirjalliseen työhön, jotta opinnäytetyö valmistuisi sovittuna aikana. Takarajaksi olemme asettaneet marraskuun, jonka loppuun mennessä opinnäytetyö olisi kokonaisuudessaan valmis.

1.3 Keskeiset käsitteet

Tässä kohdassa avaamme opinnäytetyössä käyttämäämme terminologiaa, jotta lukijan on helppo ymmärtää kontekstissa esiintyviä käsitteitä. Käsitteet ovat aakkosjärjestyksessä.

3D-mallinnus (engl. *3D-modeling*) on tekniikka, jota käytetään erinäisten pintojen tai objektien graafisten esitysten tuottamiseksi manipuloimalla pisteitä, reunoja ja kulmia 3D-virtuaalitulossa. 3D-mallinnusta hyödynnetään muun muassa videopeleissä ja elokuvissa visuaalisten tehosteiden luomiseksi, kaupallisessa mainonnassa sekä arkkitehtuurin suunnittelussa. (Slick 24.9.2020.). Digitaalinen prototyyppi tallennetaan ja tiedosto voidaan tulostaa 3D-mallinnuksille tarkoitetulla printterillä. 3D-mallinnukselle on olemassa useita eri ohjelmistoja, joista yksi tunnetuimpia on avoimeen lähdekoodiin perustuva Blender.

API (lyh. *Application Programming Interface Key*) on ohjelmointirajapinta, joka mahdollistaa useiden sovellusten keskinäisen kommunikoinnin ja tiedonsiirron linkitettyjen ohjelmien välillä, käyttäjän uniikin identifioimisen tai verkkosivun ohjelmointikutsun. (The RapidAPI s.a.). Ohjelmistokehityksessä käyttäjälle luotua uniikkia API-avainta käytetään muun muassa verkkosivujen RESTful API-sovellusliittymissä, jossa voidaan hakea, lähettää, päivittää tai poistaa tietoja HTTP-pyyntöinä.

Arduino on avoimen lähdekoodin alusta, jota käytetään erinäisissä elektroniikan projekteissa. Arduino koostuu fyysisestä, helppokäyttöisestä laitteistosta sekä tietokoneelle ladattavasta ohjelmistosta (IDE), jota käytetään koodin kirjoittamiseen ja lataamiseen fyysiselle alustalle. (Arduino s.a.a). Arduinoa hyödynnetään muun muassa IoT-applikaatioissa, 3D-tulostuksissa ja sulautetuissa ympäristöissä.

ESP32 on Espressif Systemsin kehittämä pienitehoinen, monipuolinen ja edullinen ohjelmoitava piirilevy, jossa on sisäänrakennettu Wi-Fi moduuli ja Bluetooth-yhteys. ESP32 piirilevyä voidaan hyödyntää muun muassa erilaisissa anturipohjaisissa IoT-projekteissa, kotitalousautomaatioissa, lämpötilan mittauksissa ja verkkosovelluksissa. (Boral 25.5.2020.)

IoT (lyh. *Internet of Things*), eli esineiden internetillä tarkoitetaan järjestelmiä, jotka mahdollistavat fyysisten laitteiden tiedonsiirron ja ohjauksen Internet-verkon kautta ilman ihmisen ja tietokoneen välistä vuorovaikutusta. IoT-laite koostuu sensoreista ja laitteista, jotka lähettävät tietoa pilveen. Pilvessä tieto analysoidaan ja lähetetään tarvittaessa käyttöliittymään esimerkiksi ilmoituksen tai sähköpostin muodossa. (McClelland 29.10.2016.)

MQTT (lyh. *Message Queuing Telemetry Transport*) on avoimeen lähdekoodiin pohjautuva viestintäprotokolla, joka perustuu luotettavaan standardiseen laitteiden väliseen kommunikointiin. (Stacey 16.6.2020). MQTT-protokollaa hyödynnetään laajasti muun muassa teollisuusalalla, suurissa yrityksissä sekä erinäisissä IoT- sovelluksissa ja projekteissa, joista yhtenä tunnetuimpana esimerkkinä toimii viestintäpalvelu Facebook Messenger.

OTA (lyh. *Over-the-Air*) tarkoittaa standardisoitua tekniikkaa, joka mahdollistaa ohjelmistojen, sekä laiteohjelmistojen turvallisen päivittämisen etänä. OTA-tekniikan avulla voidaan suorittaa muun muassa virhe- ja tietoturvakorjauksia sekä ominaisuuteen liittyviä päivityksiä, jotka parantavat laitteen toimivuutta ja vähentävät tietoturvaan liittyviä riskejä. (Agarwal 26.6.2020.) OTA-päivitystä käytetään laajasti muun muassa autoteollisuudessa, mobiililaitteiden ohjelmoinnissa sekä erinäisissä IoT-projekteissa.

REST-rajapinta (engl. *Representational State Transfer*) on arkkitehtuurimalli, jota sovelletaan verkkosovellusten ohjelmointirajapinnoissa. REST-mallia hyödynnetään muun muassa tietojen, sisällön ja algoritmien tuottamiseen ja asettamiseen erilaisissa verkko-, mobiili- ja laitesovelluksissa. (Lane 28.9.2020.) REST-rajapinta käyttää HTTP-protokollan mukaisia pyyntöjä, joiden avulla voidaan hakea (GET), luoda (POST), päivittää (PUT), poistaa (DELETE) tai päivittää osittain (PATCH) verkkosovelluksissa käsiteltävää dataa.

Tietokanta (engl. *database*) tarkoittaa järjestelmällistä tiedonkeruuta, joka mahdollistaa datan sähköisen tallennuksen ja käsittelyn. Suosituimpia tietokantatyyppejä ovat muun muassa hajautetut ja keskitetyt tietokannat, relaatiotietokannat, avoimen lähdekoodin tietokannat sekä JSON-tietokannat. Näitä hyödynnetään esimerkiksi asiakastietojen, taulukoiden ja asiakirjojen tallentamiseen. (Guru 99 s.a.)

2 Digitalisaatio ja Esineiden Internet

Kun puhutaan digitalisaatiosta, monet mieltävät sen informaatioteknologioiden käytön lisääntymiseksi. Kyseessä on kuitenkin laajempi, koko yhteiskuntaa koskettava murrostila, joka toimii kokonaisvaltaisena tekijänä niin liiketoiminnassa kuin arkielämässäkin.

Yhteiskunnan rakennemuutoksen myötä vaikuttamisen keinot ja ihmisten arkinen työ kokivat suuren muutoksen, kun digitaaliset työkalut syrjäyttivät kaikki käsin prosessoitavat manuaaliset vaiheet. Digitalisaatiota voidaankin verrata 1700- ja 1800-luvun vaihteessa Iso-Britanniassa tapahtuneeseen teolliseen vallankumoukseen, joka mullisti teolliset tuotantomenetelmät ja muutti yhteiskuntaa sekä ympäristöä enemmän kuin mikään muu koko ihmiskunnan historiassa. Digitalisaatio on ilmiö, joka tarkoittaa tietotekniikan kokonaisvaltaista käyttöönottoa yhteiskuntamme jokaisessa eri osa-alueessa. Se aiheutti pysyvän muutoksen ihmisten ajattelutapaan siitä, miten koemme, toimimme sekä näemme ympäriväätä ja alati muuttuvaa maailmaa.

Tietotekniikan yleistymisen ponnahduslautana pidetään 1980-luvun loppua, jolloin kotitietokoneita alkoi näkyä arkielämän toiminnoissa. Varsinainen murros syntyi kuitenkin vasta 1990-luvun tietämillä, jolloin kehityksen ytimessä olivat digitaaliset matkapuhelimet, tietokoneet ja Internet. Samaan aikaan matkapuhelinteknologia koki mullistavan käänteen, kun toisen sukupolven matkapuhelinverkko 2G syrjäytti perinteiset analogiset järjestelmät. GSM-standardilla toimiva 2G-verkko käynnistettiin kaupallisesti Suomessa Radiolinjan (nykyinen Elisa) toimesta vuonna 1991. GSM-standardista tuli yleisin ja käytetyin 2G-teknologia maailmalla. (What's a G s.a.) Tietokoneiden ja mobiilipuhelimien yleistyessä kotitalouksissa, myös teollisuusallalla koettiin suuria yhteiskunnallisia muutoksia. Informaatioteknologian eksponentiaalisen kasvun myötä syntyi uusia innovatiivisia palveluita, joka lopulta johti digitaaliseen vallankumoukseen ja teollisen internetin syntyyn.

Lisääntynyt tietojenkäsittelyn tehokkuus, nopeat verkkoyhteydet ja anturitekniikan kehitys ovat kokonaisuudessaan luoneet uudenlaiset kilpailukykyisemmät markkinat, jotka mahdollistavat jo olemassa olevien prosessien optimoinnin sekä uusien tuotteiden ja palveluiden tuottamisen edullisempaan hintaan. IoT-teknologiaa hyödynnetään nykypäivänä muun muassa teollisuuden eri valmistusprosessien ohjauksessa ja valvonnassa, terveydenhuollossa, logistiikassa sekä liikenneinfrastruktuurissa. Sen lisäarvo perustuu pääsääntöisesti systemaattisuuteen ja lisääntyneeseen tiedonmäärään. Teknologian merkittävä hinnan lasku johti tietotekniikan käytön yleistymiseen, joka puolestaan kiihdytti jo valloillaan olevaa kehitystahtia. Informaatioteknologia on tullut jäädäkseen osaksi yhteiskuntamme ekosysteemiä ja jatkaa edelleen kehittymistään ihmisten kollektiivisesta toimesta.

2.1 Internet

Internet on yksi niistä harvoista keksinnöistä, jotka ovat maailmanlaajuisesti mullistaneet ennennäkemättömästi tietokone- ja viestintäteknologian. Varsinaisena perustana tälle pidetään telegrammin, puhelimen, radion ja tietokoneen keksimistä, jotka loivat valmiuksia integraatioille. Internet on nykyisin laajalle levinnyt globaali tietoinfrastruktuuri, jonka historia on hyvin monivaiheinen. Erinäisten verkkotyökalujen käytön lisääntyminen mahdollisti sähköisen kaupankäynnin, tiedonhankinnan ja yhteisöllisen toiminnan toteuttamisen, jolla oli pysyviä rakenteellisia muutoksia niin tietoteknisten alojen kuin koko yhteiskunnan näkökulmasta katsottuna.

Internetin historia ajoittuu 1960-luvulle, jolloin MIT-tietotekniikan tutkija JCR Licklider jakoi ajatuksensa maailmanlaajuisesta tietokoneverkosta Yhdysvaltain puolustusministeriön ARPA-tutkimusprojektia kehittävien kollegoiden kanssa. Yhdessä Leonard Kleinrockin ja Thomas Merrillin kanssa Lawrence G. Roberts julkaisi patenttikytkentäisen suunnitelman ARPANET:istä DARPA:n rahoittamalle tietoverkolle, joka lanseerattiin vuonna 1969. (Jefferson Online 22.11.2016.) Verkon perustamisen syinä olivat alun perin pitkät maantieteelliset etäisyydet tutkijoiden ja tietokoneiden välillä. Termi "Internet" luotiin alun perin viittaukseksi kahden verkon yhdistelmään ja sana on edelleen käytössä.

Vuonna 1983 ARPANET jaettiin kahteen osaan: ARPANET:in siviiliversioon sekä MILNET:iin, joka oli puolestaan tarkoitettu sotilas- ja puolustusvirastojen käyttöön. Samoihin aikoihin ARPANET:in tietoliikenneprotokolla NCP poistettiin käytöstä ja tilalle nousi nykypäivänäkin tunnettu TCP/IP-protokolla. Lopullinen harppaus tapahtui kuitenkin vuonna 1985, jolloin se rakennettiin UNIX-käyttöjärjestelmän versioon. (Featherly s.a.) ARPANET suljettiin vuonna 1990 Internetin käyttöönoton myötä vain kaksi vuotta ennen kuin Tim Berners-Lee teki läpimurron ottamalla käyttöön nykypäivänäkin tunnetun World Wide Webin.

ARPA:n patenttikytkentäisen suunnitelman käynnistämisen aikoihin vuonna 1962 maailmaa ravisuttava Kuuban ohjuskriisi ajoi Yhdysvallat ja silloisen Neuvostoliiton sodan partaalle. Yhdysvaltalainen voittoa tavoittelematon kansainvälinen politiikan aivoriihi RAND Corporation pyrki poikkitieteelliseen ongelmanratkaisuun Yhdysvaltojen armeijan hyväksi analysoimalla poliittisia päätöksiä, tukemalla ydinaseiden vastakkainasettelussa ja digitalisaation murrosvaiheissa. Kylmän sodan kynnyksellä RAND:in tutkija Paul Baran tarjosi ratkaisua Yhdysvaltain viranomaisten pohtiessa erinäisiä kommunikointitapoja ydinhyökkäyksien varalle. Viranomaisten huolena oli, ettei keskitetyt kytkentälaitteet selviäisi vihollisen iskuista ja sotilashallinnon perusverkko kaatuisi. Hän esitti oman visionsa hajautetusta viestintäverkosta, joka mahdollistaisi yhteydenpidon, vaikka

kytkentäsolmut olisivat tuhoutuneet hyökkäyksen seurauksena. Tämä hajautettu viestintäverkko mahdollisti myöhemmin maailmanlaajuisen verkon perustamisen. (RAND Corporation s.a.)

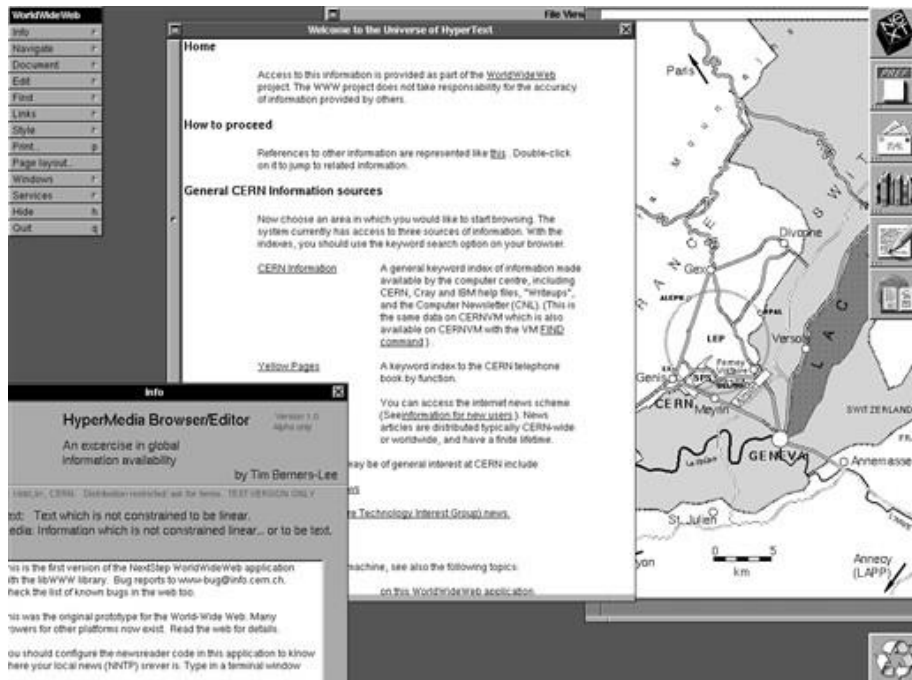
2.2 World Wide Web

Varsinaisena World Wide Webin luojana pidetään brittiläistä tietojenkäsittelytieteen tutkijaa, Tim Berners-Leetä. Hän työskenteli ohjelmistosuunnittelijana CERN-nimisessä hiukkASFysiikan laboratoriossa Sveitsissä lähellä Geneveä valmistuttuaan Oxfordin yliopistosta ja keskittyi selvittämään tutkijoiden tiedon jakamiseen liittyviä ongelmia. Tim keksi kuinka tietoa voitaisiin jakaa Internetin kautta samalla, kun miljoonat tietokoneet olivat kytköksissä toisiinsa. Hän hyödynsi visiossaan uutta hypertekstiä, joka tunnetaan paremmin verkon merkintäkielenä HTML. (World Wide Web Foundation s.a.)

Hyperteksti-termin esitti kuitenkin ensimmäisen kerran tietotekniikan kehittäjä Ted Nelson jo vuonna 1965 Xanadu-projektin myötä. Sen tarkoituksena oli luoda tietokoneverkko yksinkertaisella käyttöliittymällä, mutta rahoituksen puuttuminen, huono ajoitus ja idealistinen mentaliteetti luoda konsepti liike-elämän ulkopuolella johtivat projektin kaatumiseen. Tim Berners-Leen tekemän läpimurron myötä Nelsonin perintö jäi vaille parrasvaloja ja hän koki itsensä väärinymmärretyksi. (Sirk 24.7.2020.)

Tim Berners-Lee esitteli visionsa vuonna 1989 silloiselle CERN:in johtajalle Mike Sendallille, joka näki projektin ainoastaan epämääräisen jännittävänä luomuksena. Verkko ei varsinaisesti ollut koskaan CERN:in virallisia projekteja, mutta Mike antoi Timille vapaat kädet työskennellä sen parissa vuoden 1990 syyskuuhun saakka. Vuoden 1990 lokakuussa Tim Berners-Lee oli kehittänyt kolme tänäkin päivänä tunnettua perustavanlaatuista teknologiaa: hyperteksti (HTML), merkkijono, jolla esitetään uniikin tiedon sijainti (URI) ja siirtoprotokolla, joka mahdollistaa WWW-palvelimien ja selainten tiedonsiirron (HTTP). (World Wide Web Foundation s.a.)

Vuonna 1990 lopulla Tim Berners-Leen kehittämä web-palvelin ja verkkoselain olivat CERN:illä jo toiminnassa NeXT-tietokoneella demonstraationa hänen visiolleen. NeXT-tietokone oli Steve Jobsin varhaisimpia projekteja. Kaikkien aikojen ensimmäinen verkkosivu sisälsi informaatiota World Wide Web-projektista, palvelimen teknisistä ominaisuuksista ja linkkejä muihin tulevaisuudessa saatavana oleviin verkkopalvelimiin. (CERN s.a.a) Tekstipohjainen verkkoselain julkaistiin yleiseen käyttöön tammikuussa vuonna 1992.



Kuva 1. Kuvakaappaus Tim Berners-Leen luomasta NeXT World Wide Web selaimesta (CERN s.a.b)

World Wide Web verkkoselain julkaistiin yleiseen käyttöön tammikuussa vuonna 1992. WWW-verkkoselain sai tuulta alleen vasta, kun ohjelmistosuunnittelija Marc Andreessen kehitti Mosaic -nimisen graafisen selaimen vuonna 1993. Se loi standardin, joka oli perustana myöhemmin julkaistuille selaimille Netscape Navigatorille (1994) ja Internet Explorerille (1995). Vuonna 1995 mennessä Internetissä oli jo miljoonia aktiivista käyttäjä. (Gregersen s.a.)

Vuonna 1995 julkaistu Internet Explorer oli ohjelmistojätti Microsoft Corporationin kehittämä verkkoselain, joka suunniteltiin alun perin Windows 95-käyttöjärjestelmän lisäosaksi. Internet Explorer integroitiin vuonna 1996 osana Windowsin käyttöjärjestelmää, joka antoi merkittävää kilpailullista etumatkaa muiden Internet-selainten valmistajien keskuudessa. Internet Explorerista tuli pian suosituin ja käytetyin verkkoselain koko maailmassa. (Gregersen s.a.) Kiivas kehitystahti mahdollisti myös muiden ohjelmistojättien heräämisen ja pian käyttöön alkoi tulla muita luotettavampia ja stabiilimpia verkkoselaimia.

Internet Explorerin asemaa horjutti Netscapen työntekijän kehittämä verkkoselain Mozilla Firefox, joka julkaistiin vuonna 2004. Uuden verkkoselaimen tarkoituksena oli korjata Internet Explorerissa ilmenneitä tietoturvapuutteita sekä tarjota nopeaa ja turvallista verkon käyttöä. Vuonna 2008 Google kehitti oman verkkoselaimen nimeltään Google Chrome. Chrome tarjosi kokonaan uudenlaisen ominaisuuden, joka mahdollisti välilehtien

lisäämisen ja poistamisen ilman selaimen kaatumista. Google Chromesta tuli hallitseva selain vuonna 2013, joka ylitti Internet Explorerin ja Mozilla Firefoxin suosion. Vuonna 2015 Microsoft Corporation lopetti Internet Explorerin käytön ja tilalle tuli verkkoselain Microsoft Edge. (Gregersen s.a.) Edellä mainittujen rinnalle on tullut muita merkittäviä verkkoselaimia, kuten Opera (1995) ja Applen kehittämä Safari (2003).

2.3 Sosiaalinen verkko

Ihmiset ovat tunnetusti sosiaalisia olentoja, jotka ovat kautta aikain kommunikoineet keskenään ympäri maailmaa. Kaupankäynti ja yhteydenpito tapahtuivat kuitenkin vain kasvatusten, joka puolestaan jarrutti globalisoitumista niin kulttuurillisesti kuin taloudellisestikin. Digitalisaation aikakausi mahdollisti kuitenkin uuden ilmiön syntymisen, joka mullisti niin perinteisen liiketoiminnan kuin ihmisten tavan kommunikoida ja ylläpitää ihmissuhteita. (Social Industries 23.5.2018.) Termi sosiaalinen media (lyh. *some*) on tekninen perusta Web 2.0-käsitteelle, jolla viitataan World Wide Webin toimintamallin asiakaskeskeiseen prosessointiin. Sosiaalinen media on verkkoviestintäympäristö, jossa jokainen käyttäjä voi olla aktiivinen viestijä tai sisällöntuottaja. Sosiaalisen median alustoja käytettiin aiemmin vain pöytätietokoneen tai kannettavan tietokoneen kautta ja vasta mobiililaitteiden kehityksen myötä käyttö sekä sisällön tuottaminen kasvoivat eksponentiaalisesti niin kaupallisen kuin yksityisen käyttäjän näkökulmasta katsottuna.

Sosiaalinen media sai ensikosketuksen jo vuonna 1979 perustetun Usenetin myötä. Usenet oli yksinkertainen keskustelualusta, jossa yksityishenkilöt pystyivät julkaisemaan artikkeleita ja uutisia ryhmäkeskusteluihin. Keskustelualustojen edelläkävijä toimi myös inspiraationa XML-kieleen perustuvilla RSS-verkkosyötteille. Usenetin myötä alkoi syntyä uusia keskustelualustoja, joissa ihmiset pystyivät kommunikoimaan reaaliaikaisesti chat-huoneiden kautta. Nämä innovaatiot olivat yhdessä varsinainen ponnahduslauta nykypäivän sosiaaliselle medialle. (Social Industries 23.5.2018.)

Vuonna 2003 julkaistiin sosiaalinen alusta MySpace, josta tuli keskeinen osa uutta nuorisokulttuuria. Samana vuonna perustettu LinkedIn oli ensimmäinen sosiaalinen foorumi, jonka toiminta perustui puhtaasti liiketoiminnalle ja työelämään verkostoitumiselle. Varsinainen sosiaalinen murros tapahtui kuitenkin vuonna 2004, kun yhdysvaltalainen ohjelmistosuunnittelija Mark Zuckerberg julkaisi sovellusalusta Facebookin Harvardin yliopiston käyttöön. Facebookista tuli hyvin suosittu yliopiston opiskelijoiden keskuudessa ja kuukauden kuluessa aktiivisia käyttäjiä oli jo runsaat 10 000. Vuoteen 2006 mennessä Facebookilla oli jo 7 miljoonaa käyttäjää ja Zuckerberg laajensi sovellusalustan myös yliopistoyhteisön ulkopuolelle. Facebook pitää paikkaansa

sosiaalisten alustojen johtajana 2,2 miljardilla aktiivisella käyttäjällään. Vuosina 2005–2011 lanseerattiin myös seuraavat vaikutusvaltaiset sosiaalisen median edelläkävijät: YouTube, Twitter, Instagram, Pinterest ja Snapchat. (Social Industries 23.5.2018.)

Sosiaalisen median lyhyt historia osoittaa, että konsepti on jatkuvan muutoksen alaisena. Uusia palveluita syntyy jatkuvalla syötöllä, eivätkä kaikki onnistu pääsemään samalla mittakaavalla valtaviirran suosioon. Nykypäivän sosiaalinen media on vuorovaikutteinen, itsenäinen ja käyttäjälähtöinen. Se on muuttanut tapaamme kommunikoida, kokea, elää ja tuntea asioita aivan uudella tavalla, johon jokainen on joutunut omalta osaltaan mukautumaan. Sosiaalinen media on pala teknologiaa, sisältöä ja ennen kaikkea ihmislähtöistä kommunikointia, joka on juurtunut perustavanlaatuisesti yhteiskuntamme rakenteisiin.

2.4 Internet of Things

Useat ovat varmasti kuulleet puhuttavan käsitteestä esineiden internet (engl. *Internet of Things*), sillä se on ollut jo pidemmän aikaa polttava puheenaihe globaalissa mittakaavassa. Nopea teknologiakehitys on mahdollistanut uudenlaisen digitaalisen järjestelmän, jossa kaikki esineet kommunikoivat ihmisten sekä toistensa kanssa itsenäisesti. Esineiden internetillä tarkoitetaan miljardeja laitteita ympäri maailmaa, jotka keräävät sensoreidensa avulla tietoa fyysisestä ympäristöstä ja jakavat sekä vastaanottavat niitä Internet-verkon välityksellä puolin ja toisin. Internetiin kytketyt esineet toimivat itsenäisesti osana laajempaa infrastruktuuria mahdollistaen saumattoman ja älykkään ympäristön. Esineiden internetillä viitataan laitteiden lisäksi myös käsitteisiin, kuten kotitalousautomaatioon, kaupunkeihin ja infrastruktuuriin. Se on mahdollistanut yhteiskunnan kokonaisvaltaisen kehittymisen niin terveydenhuollon, logistiikan, liiketoiminnan, turvallisuuden kuin kaupunkien innovatiivisen suunnittelun osalta. Esineiden internetistä on tullut lyhyen aikavälin sisällä osa ihmisten jokapäiväistä elämää.

Esineiden internetiä voidaan kutsua toisinaan myös termillä teollinen internet. Teollisella internetillä ei viitata niinkään itse teknologiaan, vaan tieto- ja viestintäteknologian soveltamiseen teollisuustuotannossa sekä yhteiskunnan uusien innovaatioiden kehittämisessä, kuten koneoppimisessa, langattomassa viestinnässä ja sisäänrakennetuissa järjestelmissä. Teollinen internet on mahdollistanut uudenlaisia liiketoimintamalleja usealla teollisuuden alalla, jonka ansiosta voidaan luoda älykästä automaatiota personoitujen tarpeiden mukaan. Yhtenä suurimpana esimerkkinä pidetään teollisuusautomaatiota, jossa intensiiviset työtehtävät siirtyvät koneen hoidettavaksi mahdollistaen työntekijän keskittymisen olennaisiin asioihin rutiinitoimenpiteiden sijaan. (Tikka 22.4.2015.)

2.4.1 Historia

Käsite liitetystä laitteesta juontaa juurensa jo vuoteen 1832, jolloin keksittiin ensimmäinen sähkömagneettinen lennätin. Sähköisten signaalien avulla toimiva lennätin mahdollisti suoran viestinnän kahden eri laitteen välillä. (Khvoynitskaya 25.11.2019.) Sähköinen lennätin olikin ensimmäinen viestintäväline, jonka avulla pystyttiin luomaan suoria yhteyksiä eri puolille maailmaa. Mullistavan keksinnön ansiosta perinteisten fyysisten sanomalehtien sijaan pystyttiin kertomaan maailman tapahtumista reaaliaikaisesti ja kaupankäynti muutti muotoaan. Kilpailukykyisillä markkinoilla pysyminen vaati sähkölennättimen käyttöä ja uusiin muutoksiin mukautuminen oli elinehto.

Historian ensimmäinen yhdistetty laite oli 1970-luvulla Carnegie Melonin yliopistossa sijainnut Coca-Colan myyntiautomaatti, jota paikalliset ohjelmoijat kuluttivat päivittäin ahkeraan tahtiin. Koska myyntiautomaatti ammotti usein tyhjyyttään ja aikaa kului portaissa ramppaamiseen, he päättivät keksiä ongelmaan ratkaisun.

Tietojenkäsittelytieteen osaston jäsenet integroivat myyntiautomaattiin mikrokytkimen, jonka avulla he pystyivät arvioimaan, kuinka monta tölkkiä laitteessa oli vielä saatavana. Keksinnön avulla he pystyivät myös seuraamaan automaatin lämpötilaa ja arvioimaan pitäisikö jäähdytyslaite juomat tarpeeksi kylminä. Kyseinen keksintö käynnistikin jatkotutkimuksia ja uusia kehittymismahdollisuuksia Internetiin kytketyistä koneista ympäri maailman. (Browning 25.7.2018.)

Varsinainen termi "esineiden internet" keksittiin ensimmäisen kerran vuonna 1999, kun brittiläinen tekniikan edelläkävijä Kevin Ashton esitti näkemyksensä RFDI-järjestelmästä, joka käyttäisi sähkömagneettista kenttää identifioimaan sekä seuraamaan esineisiin liitettyjä tunnisteita. Hän painotti erityisesti termiä "Internet" kiinnittääkseen ihmisten huomion, sillä Internetistä oli tulossa pian yksi mullistavimmista ja puhutuimmista teknologioista. Esineiden internetin suosio kiihtyi kuitenkin vasta vuosina 2010–2011 ja saavutti lopulta massamarkkinoiden tietoisuuden vuonna 2014. (Knud 19.12.2014.) Kevin Ashtonia pidetään varsinaisena IoT-teknologian "isänä". Hänen keksinnöllään oli kiistatta suuri merkitys esineiden internetin ja tietotekniikan kehityksen läpimurrossa.

Esineiden internetin termin keksimisen jälkeen kehitystahti kiihtyi huimaa vauhtia ja suurimmat mullistukset tapahtuivatkin seuraavien vuosikymmenten aikana. Ensimmäisenä Internetiin kytkettynä laitteena pidetään insinöörien John Romkeyn ja Simon Hackettin vuonna 1990 kehittämää Sunbeam-leivänpaahdinta. Sunbeam käytti TCP/IP-verkkojen SNMP-tietoliikenneprotokollaa, jolla ohjailtiin leivänpaahdinten virrat päälle ja pois kytkentää. Keksintö sai kuitenkin kritiikkiä siitä, että ihmisen oli vielä lisättävä leipä

paahtimeen ja vuonna 1991 kaksikko lisäsi keksintöönsä robottivarren, joka suoritti toimenpiteen automaattisesti. Laitteesta tuli täysin päästä päähän (engl. *end-to-end*) toimiva järjestelmä. (Living Internet s.a.)



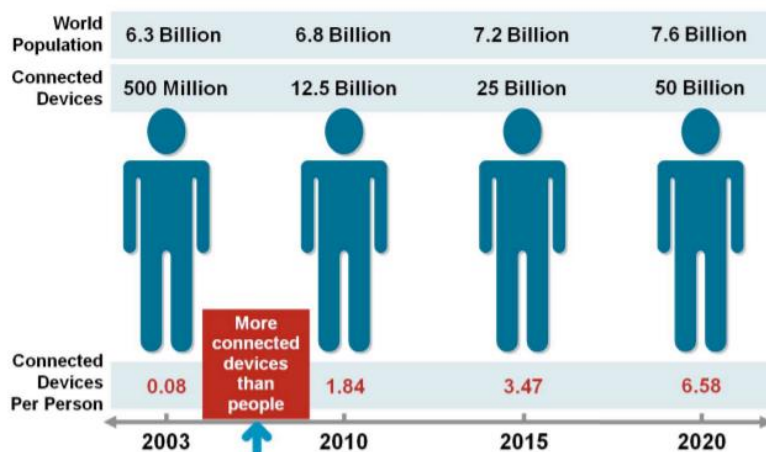
Kuva 2. Hackett demonstroimassa kehittämäänsä leivänpaahtinta (Living Internet s.a.)

2000-luvun ensimmäiset vuosikymmenet olivat esineiden internetin kannalta oleellisia. Käsite alkoi saavuttaa suurta suosiota langattomien anturiverkkojen, perinteiset automaatioalojen ja sulautettujen järjestelmien kehityksen myötä. Suurten teknologiajättien lisääntynyt kiinnostus vauhditti IoT-teollisuuden ennennäkemätöntä kasvua ja Internetiin liitettyjä esineitä alkoi tulla yhä enemmän kuluttajamarkkinoille. Laajalle levinnyt kiinnostus mahdollisti myöhemmin halpojen ja vähän virtaa kuluttavien komponenttien lisääntymisen.

Vuonna 2000 LG Electronics otti käyttöön Internetiin kytketyn jääkaapin, jonka avulla käyttäjät pystyivät tekemään ostoksia verkossa ja soittamaan videopuheluita. (Khvoynitskaya 25.11.2019). Tuotanto lopetettiin kuitenkin pian, sillä jääkaappi ei ollut myyntimenestys korkean hintansa takia, eikä esineiden internetin aika ollut vielä tarpeeksi kypsä saavuttaakseen laajoja kuluttajamarkkinoita. Internetiin kytkettyä jääkaappia pidetään kuitenkin merkittävänä modernina keksintönä esineiden internetin historiassa, joka vauhditti tulevien innovatiivisten teknologioiden syntymistä ympäri maailmaa.

Vuosina 2004–2005 esineiden internet alkoi levitä sanomalehtien kautta ihmisten tietoisuuteen ja uusia innovatiivisia ratkaisuja syntyi tiheään tahtiin. Merkittävimpiin keksintöihin lukeutui myös vuonna 2005 Rafi Haladijan ja Olivier Mévelin kehittämä Internetiin kytketty laite Nabaztag, joka oli pienikokoinen kuluttajakäyttöön tarkoitettu kanin muotoinen robotti. Nabaztag pystyi Wi-Fi-verkon kautta lataamaan muun muassa sääennusteita, osakemarkkinoiden muutoksia sekä lukemaan omistajalleen ääneen uutisia ja sähköposteja. Robottia päivitettiin kolme kertaa koko historiansa aikana, mutta lukuisat tekniset ongelmat, kehitysmahdollisuuksien suppeus ja kilpailun kiristymisen ajoivat Nabaztagin palvelimen ja asiakaspalvelun lakkauttamiseen vuonna 2015. (HQSoftware 12.7.2018.)

Vuonna 2008 Sveitsin Zürichissä järjestettiin ensimmäinen kansainvälinen teollisuuden ja korkeakoulujen IoT-konferenssi, joka kokosi yhteen tutkijat ja harjoittelijat eri puolilta maailmaa. Kolme päivää kestävässä konferenssissa esiteltiin erinäisiä tutkimustuloksia, raportteja sekä demoja nykyisestä teknologiasta. (ETH Zurich 2007). Konferenssi järjestettiin varsin hyvään aikaan, sillä Cisco IBSG:n (2011, 3) mukaan samana vuonna Internetiin kytkettyjen laitteiden määrä (12,5 miljardia) ylitti maapallolla olevien ihmisten määrän (6,8 miljardia). Tämä tekee liitettyjen laitteiden lukumääräksi noin yhden (1,84) henkilöä kohden. Edellä esitettyjen lukujen myötä voidaan katsoa esineiden internetin ”syntyneen” virallisesti vuosien 2008–2009 välisenä aikana.



Kuva 3. Esineiden internet – Kuinka Internetin seuraava kehitysaskel muuttaa kaiken. Cisco White Paper (Cisco IBSG 2011.)

Vuosina 2011–2014 tapahtui merkittäviä käänteitä esineiden internetin historiassa, kun IP-protokollan uusin versio lanseerattiin julkisena uuden sukupolven standardina. IPv6:n julkistaminen toi lisää näkyvyyttä ja kiinnostusta teollisen internetin mahdollisuuksista yritysmaailman ja rahoitusjärjestöjen silmissä. Kansainväliset teknologiajätit Samsung,

Apple ja Google alkoivat keskittyä IoT-antureiden ja laitteiden tuotantoon niin autoteollisuudessa, terveydenhuollossa kuin kuljetus- ja logistiikkapuolellakin. Tämän seurauksena yhdistetyt laitteet alkoivat yleistyä ihmisten jokapäiväisessä elämässä ja uuden sukupolven teollinen vallankumous oli alkanut. (Khvoynitskaya 25.11.2019.)

Samoihin aikoihin kotitalousautomaatioissa alettiin hyödyntää antureita kodinturvajärjestelmissä sekä kodin valaistuksen ja termostaattien tarkassa tunnistamisessa, joita käyttäjät pystyivät ohjaamaan puhelimiensa kautta.

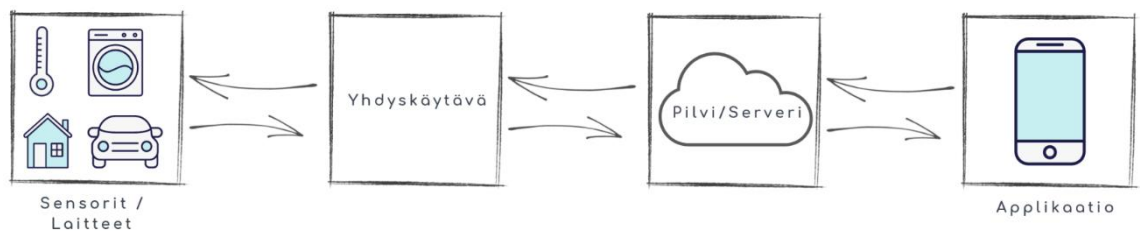
Vuonna 2014 Dublinista tuli maailman ensimmäinen älykäs kaupunki, joka käytti IoT-teknologiaa kaupungin kokonaisvaltaisen toiminnan parantamiseen. (SIMON IoT 2020). Uudenlaisessa kaupunkisuunnittelussa pyrittiin parantamaan ihmisten elämänlaatua sekä keskittymään ympäristön kuormitusta koskeviin ongelmakohtiin informaatio- ja viestintäteknologiaa hyväksikäyttäen. Suuret teknologiayritykset näkivät älykkään kaupungin hankkeet potentiaalisina menestyksen alueina, joka houkutteli lopulta muita yrityksiä ja matkapuhelinoperaattoreita mukaan edistämään turvallisen sekä mukautuvan kaupungin rakentamista tieteen ja teknologian avulla.

Vuosina 2017–2019 esineiden internet jatkoi kasvuaan ja kehittäminen oli entistäkin helpompaa. Komponenttien parempi saatavuus ja hinnan aleneminen synnyttivät uusia innovaatioita niin pienemmillä toimialoilla kuin DIY-harrastajienkin keskuudessa. Lohkoketjuteknologiaa ja tekoälyä alettiin integroida Internetiin kytkettyihin esineihin, itseohjautuvia autoja paranneltiin, älykkäitä laitteita hyödynnettiin eri kaupunkien, sähköverkkojen sekä logistiikan hallitsemisessa ja koneoppimista sekä analytiikkaa hyödynnettiin suuren datan ja pilvipalveluiden tallennettujen tietojen käsittelyssä. (Braun 25.1.2019.) IoT-ratkaisujen myötä voitiin taata kehityksen jatkuvuus myös tulevana vuosina muuttaen samalla myös juurtuneita tapoja elämisen ja työskentelyn suhteen, jotka ovat omalta osaltaan portti parantuneeseen elämänlaatuun.

2.4.2 Arkkitehtuuri

Esineiden internetin arkkitehtuuri koostuu monesta eri liikkuvasta osasta, jotka muodostavat yhtenäisen kokonaisuuden toimiakseen halutulla tavalla. Vaikka IoT-laitteet ja järjestelmät voivat erota toisistaan, esineiden internetin monikerroksisen arkkitehtuurin perusta ja prosessoitava data ovat suurin piirtein samat. Esineiden internet perustuu neljään yksinkertaiseen arkkitehtuurikerroksen rakennuspalikkaan: fyysiset anturit ja toimilaitteet, yhdyskäytävät ja -kehykset, pilvipalvelu sekä mobiiliapplikaatio. *Anturit* mittaavat fyysisestä maailmasta saatua tietoa, kuten huoneen lämpötilaa tai sijaintia ja lähettävät tiedot yhdyskäytävään. *Yhdyskäytävä* käsittelee ja kerää tiedon lähettäen ne

eteenpäin pilvipalveluun tallennettavaksi. *Pilvipalvelussa* vastaanotettu tieto tallennetaan ja käsitellään turvallisesti, jonka jälkeen käyttäjä voi sovelluksen kautta tarkastella tietoa muun muassa kaavioiden muodossa. *Mobiiliapplikaation* kautta käyttäjä voi halutessaan lähettää antureille erilaisia komentoja, jotka muuttavat esimerkiksi huoneessa sijaitsevan ilmastointilaitteen oletusarvoja. Nelikerroksisen arkkitehtuurin lisäksi yksi tärkeimmistä IoT-ekosysteemin ominaisuuksista on skaalautuvuus, jonka ansiosta järjestelmän on mahdollista kasvaa samassa mittasuhteessa organisaation tai projektin tarpeiden kanssa. Hyvin suunnitellun IoT-arkkitehtuurin ansiosta tuottavuus, turvallisuus ja tehokkuus voivat parantua niin yksittäisen kuluttajan kuin teollisuusalojen näkökulmasta.



Kuva 4. Kuinka IoT toimii. Esineiden internetin arkkitehtuuri – 4 rakennuspalikkaa (mukaillen Allurwar 2020.)

2.4.3 Käyttötarkoitus

Esineiden internetin lyhyttä historiaa tarkastellessa voidaan todeta, että eksponentiaalisen teknologiakehityksen myötä Internetiin kytketyt laitteet jatkavat sulautumistaan muihin tekniikoihin, kuten koneoppimiseen, tekoälyyn, big dataan sekä pilvi- ja reunalaskentoihin. Wi-Fi-verkon jatkuva laajentuminen, antureiden ja pilviratkaisujen kustannusten aleneminen, laskentatehon lisääntyminen sekä kasvava älypuhelisten ja etäohjattavien laitteiden käyttö ovat mahdollistaneet älykkäiden esineiden yleistymisen ihmisten arkielämässä. Seuraavaksi tarkastellaan IoT-tekniikan hyödyntämismahdollisuuksista esimerkein niin kotitalousautomaation, teollisuuden, ihmisten arkielämän, logistiikan, terveydenhuollon kuin kaupallisen ympäristön näkökulmasta katsottuna.

Ihminen

Esineiden internet on noussut viime aikoina vahvaan rooliin yksittäisen kuluttajan arkielämässä. Henkilökohtaisten langattomien laitteiden käyttö on lisääntynyt räjähdysmäisesti ja yhä useampi sovellus on integroitunut kuluttajan päivittäisiin rutiineihin. Kuluttajan IoT:ssa (CIoT) langattomat laitteet ovat varustettu yksilöllisillä tunnisteilla (UID) mahdollistaen itsenäisen kokonaisuuden erilaisissa käyttötarkoituksissa. Tämä helpottaa kuluttajan omien tietojen jakamista ilman käyttäjän suoraa vuorovaikutusta toistensa tai laitteiden kanssa. Nopeasti kehittyvät ja kasvavat

tietojenkäsittelyominaisuudet lisäävät IoT-sovellusten tehokkuutta ja helppokäyttöisyyttä, jotka parantavat skaalautuvuutta sekä alentavat tarpeetonta virrankulutusta. (Taylor s.a.)

Esimerkkejä kuluttajan IoT-laitteista ovat muun muassa älykellot, älysormukset sekä älykkäät päälle puettavat laitteet, kuten aktiivisuusrannekkeet, jotka mittaavat käyttäjän kalorien saantia, sykevälivaihtelua, kehonkoostumusta, painoa sekä verenpainetta reaaliaikaisesti. Markkinoille odotetaan tulevaisuudessa uuden sukupolven puettavaa teknologiaa, joka olisi skaalautuvampi sekä vähemmän riippuvainen älypuhelimesta. Puettavan teknologian avulla olisi myös mahdollista tunnistaa tulevaisuudessa käyttäjän tulevia terveysongelmia ennen kuin ihminen osaisi tunnistaa niitä itse.

Koti

”Smart Home” on yksi tunnetuimpia käsitteitä esineiden internetin maailmassa, joka kuvastaa kotitalouden laitteiden hallintaa älykkäällä tavalla. Älykkäät laitteet ovat nykyään jo hyvinkin arkisia asioita ihmisten jokapäiväisessä elämässä muun muassa puhelinten, sormuksien sekä rannekkeiden muodossa. Vaikka älykäs teknologia on yleistynyt ihmisten keskuudessa merkittäväällä tavalla, varsinaiset älykodit eivät ole vielä osa valtavirtaa. Kotitalousautomaation tarkoituksena on luoda toimivia ratkaisuja, jotka vähentävät käyttökustannuksia, lisäävät asumisen turvallisuutta sekä helpottavat ihmisten arkea tehden asumisesta mahdollisimman vaivatonta.

Älykkäässä kodissa laitteet ovat yhdistetty toisiinsa, joita voidaan ohjata älypuhelimien, tietokoneen tai tabletin kautta. Järjestelmien avulla voidaan ohjata muun muassa valaistusta, lämmitystä, turvajärjestelmää, ilmastointia, kodinkoneita, viihdejärjestelmiä, kulunvalvontaa sekä energianmittausta. Vaikka älykkäiden kotien IoT-ratkaisut ovat edelleen kalliita, eikä kaikilla ole varaa niihin, jatkuva kysyntä ja lisääntynyt esiintyvyys ovat pienentäneet kustannuksia tasaisen hitaasti. Esineiden internetin kiivas kehitystahti mahdollistaa tulevaisuudessa yhtenäisen organisoidun ja harkitun älylaitteiden ekosysteemin, jotka kommunikoivat keskenään parantaen sekä yksinkertaistaen ihmisten jokapäiväisiä rutiineja. (Puzhevich 25.7.2019).

Kotitalousautomaatio luo uuden käsityksen älykkäästä ympäristöstä, joka ennustaa asukkaiden tarpeita, parantaa kodin mukavuutta sekä turvallisuutta tarjoten saumattoman käyttäjäkokemuksen. IoT-teknologiaan investoidaan nyt enemmän kuin koskaan aiemmin, joka antaa mahdollisuuden keskittyä parantamaan älykodin ekologista elämäntapaa tukevia ominaisuuksia, kuten rakentamista ja suunnittelua kestävän kehityksen mukaisesti.

Kaupallinen ympäristö

Esineiden internetin jatkuvan kasvun myötä myös kaupallisen ympäristön ekosysteemi on kokenut suuria mullistuksia monilla eri osa-alueilla. Fyysiset laitteet, digitaaliset yhteydet ja massiivinen tiedonkeruu ovat mahdollistaneet vähittäiskauppoja vakauttamaan liiketoimintaansa, pienentämään kustannuksia sekä kasvattamaan toivottua liikevoittoa.

Esineiden internet on parantanut huomattavasti vähittäiskaupoissa käytettävien CRM-järjestelmien ominaisuuksia, kuten kognitiivista tietojenkäsittelyä, massadatan analytiikkaa ja liiketoiminnan funktionaalisia virtauksia. CRM-järjestelmään kerätyt tiedot tarjoavat asiakkaan tarpeiden, sijainnin ja käyttäytymisen perusteella mahdollisuuden parantaa asiakaskokemusta sekä luoda uusia asiakassuhteita parantaen samalla yrityksen tulovirtoja. Asiakaskokemusten personoinnin avulla markkinointipäälliköt pystyvät tekemään entistä tietoisempia valintoja kohdentamalla markkinointia kustannustehokkaammin huomioiden yksilön tarpeet ja mielenkiinnon kohteet. Asiakasmarkkinoinnin lisäksi IoT-teknologiaa on hyödynnetty myymälänhallinnassa sekä toimitusketjujen ja logistiikan osa-alueilla. Älykkäät seurantajärjestelmät tuottavat arvokasta tietoa toimitusketjun vaiheista parantaen laadunhallintaa, jäljittävyttä ja turvallisuutta. IoT-ratkaisujen myötä yritykset ovat kasvattaneet liikevoittoa, vähentäneet hävikkiä, parantaneet jäljitettävyyttä sekä lisänneet toimintatehokkuutta tarjoten laadukkaampia palveluja asiakkailleen. Myymälänhallinnassa automatisoidaan tilauksia, optimoidaan tuotesijoittelua ja inventointia analysoimalla yksittäisen tuotteen tietoja parantaen samalla varastonhallintaa sekä minimoiden mahdollisia myymälävarkauksia. (Digiteum Team 2021.)

Yksi vähittäiskaupan mullistavimmista muutoksista on myymälähallinnan tehokkuutta nostavat itsepalvelukassat. Uuden teknologian ansiosta myymälähallintaan kohdistuva työvoiman määrä on voitu siirtää muihin tehtäviin, joissa ollaan enemmän riippuvaisia ihmisen vuorovaikutuksesta, myyntimäärät ovat kasvaneet sekä jonot ovat lyhentyneet parantaen samalla asiakkaan kokemusta mutkattomasta asioinnista. Esineiden internetin käyttöönotto kaupallisessa ympäristössä on suuri avaintekijä nykyisen liiketoiminnan maailmassa.

Toimisto

Älykkään toimistoteknologian hyödyntäminen on ollut viime vuosina merkittävässä roolissa yritysmaailman näkökulmasta katsottuna. Yritykset pyrkivät esineiden internetin avulla takaamaan toimiston jatkuvan toimivuuden ilman ihmisen välitöntä vuorovaikutusta. Tehokkuuden lisääntyminen, kustannusten väheneminen ja optimaalisen työympäristön luominen ovat olleet merkittäviä etuja yritystoiminnalle. Pienimuotoisten tehtävien

automatisointi on vapauttanut henkilöstöä keskittämään aikansa varsinaisten projektien toteuttamiseen, jolla on puolestaan ollut välittömiä vaikutuksia yhtenäisen toimistoarkkitehtuurin kehittämisessä. Anturiteknologian avulla on pystytty minimoimaan energiankulutusta automatisoimalla muun muassa virrankatkaisun ohjausta ja lisäämällä älytermostaatteja, joka mittaa kosteutta sekä lämpötilaa luodakseen optimaalisen työympäristön jokaiselle työntekijälle. Älykkään toimistoteknologian avulla voidaan myös hallinnoida varastojen inventointia ja logistiikkaa liiketoiminnan sujuvoittamiseksi. (Starostin 15.4.2021.)

Älykkään toimistoteknologian merkittävyys on korostunut erityisesti vallitsevan globaalin pandemian aikana, jolloin työntekijöiden turvallisuuteen on haluttu panostaa entistäkin enemmän. Monet yritykset ovat alkaneet käyttää toimistotiloissa LED-valaistusta auttaakseen ihmisiä pitämään etäisyyttä toisiinsa ja näin ollen takaamaan turvalliset työskentelyolosuhteet. Turvallisuusautomaatioon panostaminen on myös säästänyt yrityksiltä rahaa ja vähentänyt fyysisten varkauksien uhkaa. Älykkäät tietoturvaratkaisut ovat mullistaneet kameroiden perinteisen etäohjauksen mahdollistamalla toimistolle menevien ja lähtevien henkilöiden seurannan ääni- ja kasvojen tunnistustekniikan avulla. Samalla voidaan valvoa ja estää ulkopuolisten ihmisten luvattoman pääsyn rajoitetuille alueille, joka puolestaan lisää työpaikan turvallisuustasoa kokonaisvaltaisesti. Älykkään toimistoautomaation arkkitehtuurin kehittäminen vaatii tarkkaa suunnittelua sekä ymmärrystä siitä, millaisia antureita kannattaisi hyödyntää ja mitä eri osa-alueita järjestelmässä tulisi hallinnoida. (Starostin 15.4.2021.)

Tehtaat

Älykkään ja turvallisen tulevaisuuden parantamiseksi valmistajat ovat yhä enemmän alkaneet hyödyntää esineiden internetiä tehdastuotannossa sekä tuotekehityksessä. IoT-teknologian avulla voidaan parantaa tuotantoteollisuuden laitekokonaisuuksia ja ohjausjärjestelmiä, jotka pystyvät havaitsemaan sekä välttämään prosessissa esiintyviä ongelmia etäyhteyden avulla. IoT-teknologiaa sovelletaan niin varastonhallinnassa, ennakoivissa huoltotoimenpiteissä kuin omaisuuden seurannassa.

Älykkäät varastohallintaratkaisut helpottavat valmistajia automatisoimaan varaston seurantaa ja raportointia, joiden avulla pystytään varmistamaan jatkuva läpinäkyvyys saatavilla olevista varastotuotteista sekä optimoimaan varastotilauksien ja toimittamisen välistä aikaa. Tämä perustuu IoT- ja RFID-teknikkaan, jossa varaston tuotteilla on yksilöllinen tunnus eli niin kutsuttu tagi. Varastotuotteet merkitään RFID-tunnisteella, jonka avulla voidaan lukea tuotetiedot ja välittää ne pilveen tallennuksen käsittelyä varten. IoT-teknologia tarjoaa valmistajalleen reaaliaikaisen kuvan jokaisen tuotteen sijainnista,

statuksesta sekä kunnosta, mikä heijastuu automaattisesti parempaan asiakaskokemukseen sekä tuotteiden tehokkaaseen markkinoille pääsyyn. IoT-teknologian käyttöönoton myötä teollisuusyksiköt ovat pystyneet parantamaan tuotannon laatua ja valmistusprosessien tuottavuutta. (Shiklo 25.12.2018.)

Kustannusten vähenemisen ja prosessien nopeutumisen myötä teollisuusyksiköiden johtajat ovat voineet kohdentaa resursseja paremmin muun muassa työntekijöiden turvallisuutta ja koulutusta koskevissa asioissa. Työntekijöitä voidaan seurata RFID-tunnisteiden keräämien sijaintitietojen sekä puettujen antureiden avulla, jotka keräävät puolestaan tietoja sykkeestä, ihon lämpötilasta ja muista oleellisista parametreista. Anturin antamat tiedot välitetään pilveen, jossa analysoidaan saatua dataa kontekstittietojen kanssa. Teollinen internet on kompakti työkalu yritysten tuottavuuden maksimoimiseksi, tuotannon käyttöajan ylläpitämiseksi sekä kustannusten vähentämiseksi. IoT-teknologian käyttöönotto tehdastuotannossa vaatii kuitenkin täsmällistä suunnittelua toimivan kokonaisuuden toteuttamiseksi. (Shiklo 25.12.2018.)

Autoteollisuus

Esineiden internetin yleistymisen on näkynyt vahvasti myös autoteollisuudessa, jossa teknologiakehitys on ollut huimaa. Vaikka IoT-teknologiaa on hyödynnetty autoissa jo 90-luvulta lähtien muun muassa diagnostiikkajärjestelmän, Bluetoothin ja USB-protokollan muodossa, lopullinen läpimurto saavutettiin vasta paljon myöhemmin. Nykyään autot ovat varusteltu lukuisilla antureilla, palveluilla ja laitteilla, jotka toimivat Internet-yhteydellä parantaen käyttäjän turvallisuutta sekä ajomukavuutta.

Hyvänä esimerkkinä IoT-teknologian hyödyntämisestä nykyisessä autoteollisuudessa on Tesla Motorsin itseohjautuvat autot, jotka käyttävät tekoälyä sekä koneoppimisalgoritmeja ennustamaan autojen ja jalankulkijoiden käyttäytymistä eri tilanteissa. Teslan toimitusjohtajan Elon Muskin mukaan kaikki Tesla-autot toimivat yhtenä verkostona, sillä yhden auton keräämien anturitietojen avulla opittu tieto siirtyy ohjelmistopäivitysten mukana myös muihin Tesla-autoihin. Näin ollen jokainen auto pystyy korjaamaan itsensä ja Teslan omistajan ei tarvitse huolehtia auton viemisestä korjaamolle. (Strongbytes 2019.)

Esineiden internetin ansiosta ajoneuvot integroituvat entistäkin paremmin, jolla on puolestaan suotuisia vaikutuksia esimerkiksi liikenneturvallisuudessa, saasteiden ja energiankulutuksen vähentämisessä sekä suurien kaupunkien liikenneuhkien minimoinnissa. Näiden lisäksi yksittäisen kuluttajan suurimpia hyötyjä IoT-teknologiassa ovat ennaltaehkäisevä huolto, jossa anturit reagoivat reaaliaikaisesti auton kaikkiin

muutoksiin sekä erilaiset kamerat ja anturit, jotka vähentävät esimerkiksi inhimillisiä virheitä parkkipaikalla. IoT-teknologian kehityksen ansiosta ajaminen on entistäkin helpompaa ja turvallisempaa.

Terveydenhuolto

Esineiden internet on muuttanut terveydenhuoltosektoria ennennäkemättömällä tavalla. Ennen terveysteknologian mullistusta potilaiden terveydentilan seuranta sekä ihmisten välinen vuorovaikutus rajoittui ainoastaan käynteihin sekä puhelin- ja tekstiviestikontakteihin, eivätkä lääkärit tai sairaalat pystyneet seuraamaan potilaan terveyttä reaaliaikaisesti ja antamaan suosituksia tarpeen mukaisesti. Esineiden internetin yleistymisen on mahdollistanut etävalvonnan terveydenhuollossa, joka on puolestaan vapauttanut huomattavan määrän resursseja parantaen samalla potilaiden hoitotuloksia. Käytettävät IoT-laitteet tarjoavat terveydenhuololle digitalisoidun ja yhdistetyn ekosysteemin, jossa ihmiset työskentelevät sekä tekevät yhteistyötä ilman fyysisiä rajoituksia. Älykäs organisointi ja hallintatyökalujen kokonaisvaltainen hyödyntäminen ovat tuottaneet tulosta myös liiketoiminnan näkökulmasta katsottuna. Esineiden internet on epäilemättä mullistanut terveydenhuollon alaa, joka hyödyttää niin potilaita, lääkäreitä, sairaaloita kuin vakuutusyhtiöitäkin.

Potilaille tarkoitetut älykkäät puettavat laitteet mahdollistavat muun muassa verenpaineen, sykkeen ja verensokerin mittaamista, joiden avulla käyttäjä voi seurata omia tuloksia, yksilöidä kalorinkulutusta, määritellä harjoituksia sekä oman terveydentilan seurantaan liittyviä lääkäritapaamisia. IoT-laitteita voidaan hyödyntää myös unenlaadun seurannassa, syövän hoidossa, Alzheimerin havaitsemisessa jo taudin varhaisessa vaiheessa, hyytymien ja astman seurannassa, liikehäiriöissä sekä sydänkohtausten ennaltaehkäisemisessä. (Kruglyak 9.12.2020.) Suurimman päivittäisen hyödyn ovat saaneet iäkkäämmät potilaat, jotka tarvitsevat säännöllistä terveydentilan seuranta. Seurantalaitteiden hälytysmekanismin ansiosta voidaan lähettää signaali perheenjäsenille, asianomaisille sekä terveydenhuollon työntekijöille, mikäli potilaan tila muuttuu huolestuttavampaan suuntaan tai hän kaatuu tapaturmaisesti. Tämä on ollut erityinen apu muun muassa yksineläville vanhuksille, jotka asuvat vielä kotonaan, eivätkä tarvitse säännöllistä apua päivittäisissä askareissa.

Potilaiden terveyden seurannan lisäksi IoT-teknologiaa hyödynnetään muilla terveydenhuoltosektorin osa-alueilla, kuten sairaalan sisäisessä seurannassa. Lääketieteelliset laitteet on merkitty antureilla, joiden avulla voidaan seurata niiden sijaintia sairaalassa reaaliaikaisesti. Näitä laitteita ovat muun muassa pyörätuolit, sumuttimet, defibrillaattorit, happipullot sekä välttämättömät lääkintä- ja valvontalaitteet.

(Kruglyak 9.12.2020.) IoT-laitteiden yleistyminen terveydenhuoltoalalla on avannut valtavia mahdollisuuksia sairauksien ennaltaehkäisemiseen ja parantuneeseen elämänlaatuun. Laitteiden tuottamat valtavat datamäärät muuttavat terveydenhuollon ekosysteemiä mahdollistaen sairaaloiden kantokyvyn parantaen samalla liiketoimintaprosessien ja työnkulun tehokkuutta.

Kaupungit

”Smart City” on laajalle levinnyt käsite, joka kuvastaa taloudellista ja teknologista kaupunkisuunnittelua älykkään hankkeen muodossa. Älykkään kaupungit koostuvat uudentlaisesta ajattelutavasta hyödyntää resursseja olemassa olevien teknologioiden sekä monimuotoisen data-analytiikan avulla. Parantuneet laskentatehot ja anturiteknologian laaja käyttöönotto ovat mahdollistaneet kokonaisinfrastruktuurin järjestelmien älykkään suunnittelun. Älykkään kaupungin hankkeita on lähdetty toteuttamaan erilaisin strategiain tarpeiden mukaan. Liikkuminen, puhtaus ja jätehuolto, hyvinvointi ja terveys, energia ja ympäristö sekä ekologinen, sosiaalinen ja taloudellinen kestävyys ovat olleet keskeisiä kehityksen kohteita. Pyrkimys on integroida ja kehittää älyteknologioita ihmisiä paremmin palvelevaksi kokonaisuudeksi. Älykkään kaupungin toimintamallissa korostetaan erityisesti teknologista ja liiketaloudellista puolta, jossa minimoidaan energian hukkakäyttöä, parannetaan taloutta liikenteessä kuluvan ajan vähentämisellä sekä vahvistetaan julkisen/kansanterveyden resursseja älykkäiden järjestelmien avulla. (Koponen, Mustonen & Spilling 2014, 26.)

Kuljetus ja logistiikka

IoT-teknologia on parantanut huomattavasti kuljetus- ja logistiikka-alan monimutkaisten järjestelmien sekä varastojen hallintaa. IoT-anturitietoja hyödynnetään muun muassa autojen, kuorma-autojen, junien ja laivastojen reitityksissä sääolosuhteiden, ajoneuvojen saatavuuden sekä kuljettajan saatavuuden mukaan. Samalla voidaan seurata antureiden avulla elintarvikkeiden, lääkkeiden ja muiden lämpötilaherkkien lähetysten lämpötilan valvontaa rakennetun seurantasovelluksen avulla. Sovellus lähettää hälytyksiä, kun lämpötila nousee tai laskee kriittiselle tasolle tuotteen säilyvyyden kannalta ja näin ollen lisää tuotteiden turvallisuutta samalla ennaltaehkäisten suuria hävikkejä. (Oracle s.a.) Toiminnan tehokkuuden lisäksi IoT-teknologian hyödyntäminen vähentää kustannuksia, päästöjä, ajoneuvojen kulumista sekä parantaa mahdollisuuksia saavuttaa korkeammat kestävä kehityksen ja yritys vastuun vaatimukset.

2.4.4 Kyberturvallisuus

Kyberturvallisuuden rooli on kasvanut IoT-teknologian eri osa-alueilla räjähdysmäisesti lisääntyneiden kyberhyökkäysten ja digitalisaation kehityksen myötä. Laaja käyttöönotto on tehnyt IoT-turvallisuudesta kriittisen ongelman, sillä viime aikoina laitteet ovat olleet merkittävässä määrin alttiina verkkohyökkäyksille, jotka voivat pahimmassa tapauksessa johtaa tietojen päätyminen väärin käsiin ja edetä lopulta liiketoiminnallisiin tappioihin. Yleisimmät haavoittuvuudet johtuvat useimmiten monimutkaisesta arkkitehtuurista, epäammattimaisista suojaukseen kohdistuvista määrityksistä, fyysisestä turvallisuudesta tai epävakaista laiteohjelmistoista. Voittoa tavoittelematon ja avoimeen lähdekoodiin pohjautuva OWASP (Open Web Application Security Project) on esitellyt kattavan määritelmän kymmenestä suurimmasta IoT-arkkitehtuuriin kohdistuvista haavoittuvuuksista, joiden katsotaan olleen syynä kyberhyökkäyksien onnistumisille. OWASP-yhteisön tarkoituksena on kasvattaa ihmisten tietoisuutta turvallisen ohjelmistokehityksen eri menetelmistä ja kehittää uusia ratkaisuja alan haasteisiin.

Kriittisimmät haavoittuvuudet ovat luokiteltu *injektioihin* (taustajärjestelmäkyselyn, kuten SQL-kyselyiden muokkaaminen niin, että hyökkääjällä on mahdollista vaikuttaa mitä informaatiota tietokannasta palautetaan), *autentikointi- ja istuntohaavoittuvuuksiin* (tunnistautumisen ohittaminen tai toisen käyttäjän istunnon valtuuksien väärinkäyttö kohdepalvelussa), *arkaluontoisten tietojen paljastumisiin* (arkaluontoisten tietojen vuotaminen väärin käsiin tietomurron aikana), *XSS-hyökkäyksiin* (hyökkääjällä on mahdollista lukea käyttäjän valtuuttamaa sisältöä tai muokata sivuston rakennetta haluamallaan tavalla), *turvattomiin objektiviittauksiin* (tietokantamuokkaukset, kuten id:n arvon muuttaminen ilman oikeuksia), *turvattomiin konfiguraatioihin* (sovellusallustaa ei ole päivitetty tai asennettu asianmukaisesti), *pyyntövääreennöksiin* (palvelu ei varmenna pyynnön alkuperää, jolloin hyökkääjä voi suorittaa komentoja käyttäjän tahdonvastaisesti), *validoimattomiin uudelleenohjauksiin* (käyttäjä uudelleenohjautuu sovelluksessa olevan linkin kautta kohdepalveluun, jolloin hyökkääjällä on mahdollisuus ohjata vahingolliselle kolmannen osapuolen sivustolle), *haavoittuvien komponenttien käyttöön* (sovelluksen kirjastoja ja komponentteja ei päivitetä asianmukaisesti, jonka seurauksena ulkopuolinen henkilö pääsee murtautumaan palveluun) sekä *puuttuvien funktiotason pääsynhallintaan* (palvelussa käyttäjäprofiilia ei autentikoida valtuuksien määrittämiseksi, jolloin hyökkääjä pääsee muun muassa käsiksi arkaluontoisiin tietoihin tai pahimmassa tapauksessa kaappaamaan koko palvelun itselleen). (2NS s.a. 5–13)

Tunnetuimpana tietoturvaloukkauksen esimerkkitapauksena pidetään 21. lokakuuta 2016 tapahtunutta Mirai-bottiverkon hajautettua palvelunestohyökkäystä (DDoS), joka aiheutti massiiviset palvelukatkokset verkkosivuille ympäri maailmaa. Asiantuntijoiden mukaan

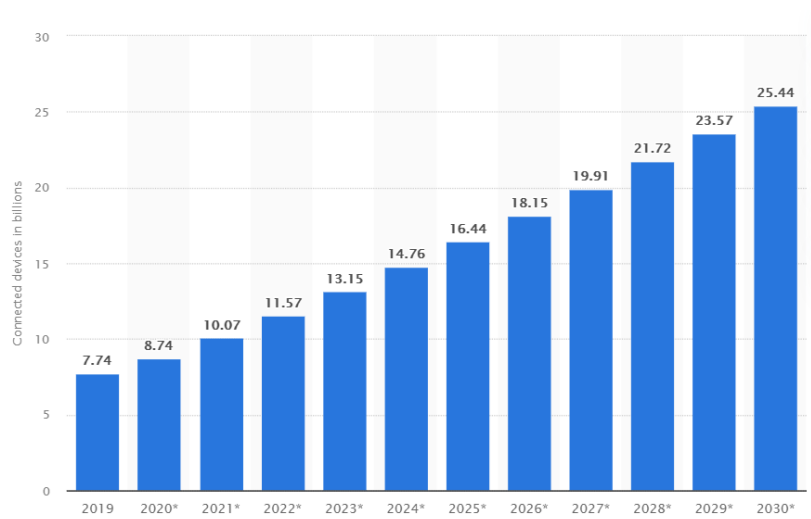
kyseessä oli laajamittaisin hyökkäys koko Internetin sekä IoT:n historiassa ja tapaus antoi laajamittaisesti perspektiiviä siihen, miten paljon potentiaalia bottiverkon kaltaisilla haittaohjelmilla on vahingoittaa esineiden internetin ekosysteemiä. Mirai on haittaohjelma, joka muuttaa Linuxia käyttävät verkkolaitteet etäohjattaviksi boteiksi. Sen toimintaperiaate jakautuu haittaohjelmien levitykseen sekä verkkohyökkäyksiin. Kohteena olivat Internetiin kytketyt laitteet, kuten kotireitittimet, valvontakamerat sekä muut kodin älylaitteet. (Cloudflare s.a.) Tietoturvaloukkauksilla voi olla merkittäviä vahinkoja niin yhteiskunnan, organisaation kuin yksittäisen käyttäjän osalta, joka vaatii ehdotonta valppautta sekä kattavaa suunnitelmaa. Ennaltaehkäisevät toimenpiteet auttavat rajoittamaan hyökkäyksen aiheuttamia vahinkoja, jotta voidaan keskittyä suojelemaan palvelun arkaluontoisimpia asioita.

2.4.5 Tulevaisuus

Esineiden internetistä on tullut yksi vallitsevista teknologioista. Se on käynyt läpi suuria muutoksia ja kasvattanut suosiota varsin lyhyessä ajassa. Vuosikymmen sitten Internetin välityksellä toimivat älykkäät esineet kuulostivat lähinnä utopistiselta hyperkytkeytyneeltä maailmankuvalta, jossa tulevaisuuteen liittyvät jännitteet muovaisivat yhteiskuntamme tapoja, toimintamalleja ja arvoja merkityksellisellä tavalla. Nykypäivänä päälle puettavat laitteet, kotitalousautomaatio sekä muut ihmisten elämää helpottavat älylaitteet tuntuvat jo arkisilta ja normaaleilta asioilta.

Vallankumouksellisen IoT-kehityksen myötä maailma tulee vielä muuttumaan seuraavan vuosikymmenten aikana muovaten yhteiskuntamme jokaista osa-aluetta. Sähkötuotanto, maatalous, rakentaminen, autoteollisuus sekä terveydenhuolto kokevat dramaattisia mullistuksia, kun esineiden internet mahdollistaa miljardien laitteiden samanaikaisen käyttöönoton sekä massiivisen datamäärään hyödyntämisen automatisoiden erilaisia liiketoimintaprosesseja koneoppimisen ja tekoälyn avulla. Tämä edellyttää toimivaa infrastruktuuria, kapasiteettien kasvattamista, tekoälyn optimaalista hyödyntämistä sekä tiivistä synergiaa IoT-laitteiden ja verkkoalustojen välillä.

Vuonna 2020 Internetiin yhdistettyjä laitteita oli kaiken kaikkiaan noin 8.74 miljardia kappaletta. Statistan mukaan IoT-laitteiden määrä tulee kolminkertaistumaan maailmanlaajuisesti 25.4 miljardiin vuoteen 2030 mennessä. (Holst 2021.)



Kuva 5. IoT-laitteiden määrä maailmanlaajuisesti 2019–2030. Statista (Holst 2021).

2.5 Järjestelmäsiru

Viimeisen vuosikymmenen aikana integroitavien piirien muuttuessa yhä monimutkaisemmaksi ja kalliimmaksi kokonaisuudeksi, puolijohdeyritykset alkoivat ylläpitää suuria tutkimus- ja kehitysbudjetteja, jotta tuotantoprosessi voisi tuottaa yhä ajantasaisempia, tehokkaampia ja halvempia sirupohjaisia elektroniikkatuotteita. Puolijohteita kutsutaan siruiksi tai elektronisiksi piirikomponenteiksi, joita löytyy monista tuhansista tuotteista, kuten tietokoneista, älypuhelimista, aurinkokennoista sekä muista elektronisista laitteista. Puolijohteiden avulla voidaan upottaa pienempiä ja yksinkertaisempia järjestelmiä yhteen siruun, joka mahdollistaa tehokkuuden lisäämisen vähentäen järjestelmäkustannuksia.

Järjestelmäsiru eli SoC (*System of a chip*), on piiri, joka integroi yhden tai useamman tietokoneen tai muun järjestelmän elektroniset osat. Nämä komponentit sisältävät useimmiten tietokonegrafiikan käsittelyyn tarkoitetun mikroprosessorin (GPU), keskusyksikön (CPU) sekä tietokonejärjestelmän keskusmuistin (RAM). Järjestelmäsirun tarkoituksena on vähentää energiahukkaa, virrankulutusta sekä massiivisten järjestelmien käyttämää tilaa. SoC on mahdollistanut lukuisten kannettavien laitteiden käytön paikasta riippumatta tinkimättä kuitenkaan ominaisuuksista tai toiminnasta. Sirua hyödynnetään muun muassa älypuhelimissa, tietokoneissa, kameroissa ja muissa langattomissa tekniikoissa, jonka vuoksi se on ollut korvaamaton osa alati kehittyvässä tietotekniikan maailmassa. (AnySilicon s.a.)

2.5.1 Arduino Uno

Arduino Uno on avoimeen lähdekoodiin perustuva elektroniikka-alusta, joka kommunikoi Arduinolle yksinkertaistetun C++-ohjelmointikielen ja IDE-ohjelmointiympäristön kautta. Arduino Unosta on tullut varsin suosittu ihmisten keskuudessa sen helppokäyttöisyyden vuoksi. Se sopii erityisesti harrastetoimintaan, aloitteleville ohjelmoijille, kouluprojekteihin sekä suunnittelijoiden ja taiteilijoiden prototyyppien rakentamiseen, sillä alustan mikroprosessorin muisti ja nopeus ovat hyvin rajalliset, eikä sen puitteissa ole mahdollista toteuttaa suurempia projekteja. Arduino Uno soveltuu esimerkiksi lämpötilan seurantaan ja monitorointiin, LED-valojen kytkemiseen sekä yksinkertaisten toistuvien tehtävien hoitamiseen.

Arduino Uno on optimoitu kaiken tasoisille käyttäjille, jonka ansiosta se soveltuu kokeneiden käyttäjien lisäksi myös aloitteleville harrastelijoille, joilla ei ole ohjelmoinnista aikaisempaa kokemusta. Yksinkertaistetun C++-kielen sekä verkossa löytyvän laajan opetusohjelmien ja projektivalikoiman ansiosta Arduino Uno on matalan kynnyksen

mikrokontrolleri, jonka avulla voi oppia ja rakentaa haluttuja projekteja kustannustehokkaasti.

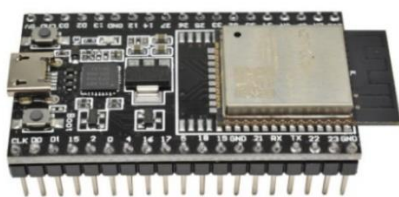


Kuva 6. Arduino Uno Rev3-mikrokontrolleri (Arduino Store s.a.)

2.5.2 ESP32

ESP32 on Espressif Systemsin kehittämä edullinen, monipuolinen ja pienitehoinen 32-bittinen järjestelmäsiru (MCU), johon on integroitu Wi-Fi ja Bluetooth-moduulit. Järjestelmäsirun Wi-Fin nopeudet voivat olla jopa 150Mbps, mikä tekee siitä erinomaisen komponentin monenlaisiin IoT-projekteihin, jotka edellyttävät Internet-yhteyttä. ESP32 tarvitsee vain vähän ulkoisia komponentteja, joka tekee laitteiston ja sovelluskokonaisuuksien suunnittelusta helppoa ja halpaa. Se integroituu helposti Arduino IDE-kehitysympäristön kanssa, joka laajentaa mahdollisuuksia erilaisten anturipohjaisten IoT-laitteiden kehittämiseen esimerkiksi kotitalousautomaatiossa.

ESP32 soveltuu moniin eri ohjelmointiympäristöihin ja on ohjelmoitavissa monella eri ohjelmointikielellä mukaan lukien C++ ja Python. Sitä hyödynnetään muun muassa älypuhelimissa, puettavissa elektronisissa laitteissa sekä harrastajien DIY-projekteissa. ESP32 integroituu perinteisiin verkkosovelluksiin ja pystyy kommunikoimaan muun muassa sähköpostien ja tekstiviestien välityksellä. Tärkeimmät ominaisuudet kuitenkin painottuvat anturiteknikkaan, jossa voidaan mitata sääolosuhteita, lämpötiloja sekä muiden tarkkuutta vaativia arvoja. (Boral 25.5.2020.)



Kuva 7. ESP32-moduuli (diymore s.a.)

Arduino IDE

Arduino IDE (integroitu kehitysympäristö) on avoimeen lähdekoodiin perustuva monialustainen ohjelmisto, joka mahdollistaa koodin kirjoittamisen, kokoamisen ja lataamisen kommunikoidakseen yhteensopivan laitteiston kanssa. Arduino tukee C/C++ -kieltä ja on saatavana Windows, Linux ja macOS-käyttöjärjestelmille. (Arduino s.a.b)

Luonnokset

Ohjelmistoalustalla kirjoitettua koodia kutsutaan luonnoksiksi, jotka kirjoitetaan tekstieditorilla ja tallennetaan tiedostotunnisteella .ino. Tallentamisen jälkeen Arduinossa on mahdollista tarkistaa koodi mahdollisista virheistä, jotka ohjelmisto näyttää sarjamonitorissa. Useimmiten tämä aiheuttaa monille päänvaivaa, sillä Arduinolla on tiukat syntaksivaatimukset ja koodin tarkistaminen voi kaatua pelkästään käyttäjän lisäämiin ylimääräisiin välimerkkeihin. Prosessin mennessä läpi odotustenmukaisesti voidaan siirtyä koodin varsinaiseen lataamiseen. Arduino kokoaa ja lähettää koodin järjestelmäsiruun, joka mahdollistaa yhteyden haluttuun laitteen toimintoon.

Sarjamonitori

Sarjamonitori on Arduinon yksi hyödyllisimmistä työkaluista, joka helpottaa luonnoksien virheiden tunnistamista. Se on ponnahdusikkuna, joka toimii erillisenä päätelaitteena sarjatietojen lähettämisen ja vastaanottamisen. Kun yhteys tietokoneen ja laitteen välillä on muodostettu, sarjamonitorissa näkyy haluttu vuorovaikutus tekstiriveinä. Sarjamonitorin ansiosta käyttäjän on helppo seurata jokaisen komponentin toimintaa reaaliajassa ja korjata prosessissa tapahtuvia virheitä.

Kirjastot

Arduinon yksi parhaimpia lisäominaisuuksia ovat sisäänrakennetut kirjastot, jotka tarjoavat perustoimintojen lisäksi ominaisuuksia, joiden avulla järjestelmäsirun toimintoja voidaan laajentaa ja parantaa. Kirjastot nimetään koodin yläpuolelle #include-lausekkeella, joka mahdollistaa ulkoisten kirjastojen sisällyttämisen luonnokseen. Kirjastotyyppejä on kahdenlaista, joista toinen toimii vuorovaikutuksessa tietyn komponentin osan kanssa ja toinen toteuttaa uusia toimintoja. IDE:n mukana on asennettu useita kirjastoja, mutta käyttäjä voi halutessaan luoda myös itse omia kirjastoja ja sisällyttää ne luonnokseen. (Arduino s.a.c)

Muuttujat

Arduinossa muuttujat ovat tapa nimetä ja tallentaa ohjelmassa haluttu arvo myöhempää käyttöä varten. Kaikki muuttujat nimetään ennen niiden käyttämistä, sillä ohjelma lukee koodia ylhäältä alas rivi kerrallaan. Vaikka globaalien muuttujien määrittäminen, nimeäminen sekä alkuarvon asettaminen eivät ole välttämätöntä, olisi ne hyvä esittää, jotta arvojen hyödyntäminen olisi tulevaisuudessa helpompaa ja koodillisesti myös paremmin ymmärrettävissä. (Arduino s.a.d) Komennolla `#define` voidaan määrittää vakio, joka ei muutu ohjelman aikana. Asetettua arvoa voidaan myöhemmin hyödyntää luonnoksessa viittaamalla tähän vakioon, joka helpottaa muistamista ja parantaa koodirakennetta.

Pin-määritelmät

Järjestelmäsiruissa voi olla sekä analogiset että digitaaliset nastat, jotka määrittelevät tulon (INPUT) ja lähdön (OUTPUT). Arduino IDE:ssä digitaaliset ”pinoutit” merkitään syntaksina `pinMode(pin, mode)`, `digitalRead(pin)` ja `digitalWrite(pin, value)`-komennoilla, kun taas analogiset merkitään `analogRead(pin)` ja `analogWrite(pin, value)`-komennoilla. Analogisia nastoja voidaan käyttää digitaalisten nastojen mukaisesti määrittelemällä analogiatulot esimerkiksi A0, A1 jne. Jokaisessa nastassa on sisäinen vetovastus, joka voidaan kytkeä päälle ja pois arvoilla HIGH tai LOW. (Arduino s.a.e)

Funktiot

Arduinossa on kaksi vaadittua funktiota, `setup()` ja `loop()`. Setup-funktiolla aloitetaan luonnoksen kokoaminen sekä kirjastojen käyttö alustamalla ja asettamalla luonnoksessa esiintyvät muuttujat. Tätä funktiota käytetään luonnoksessa vain kerran jokaisen järjestelmäsirun käynnistyksen ja nollauksen jälkeen. Setup-funktion määrittämisen jälkeen asetetaan toinen vaadittu funktio, `loop`. Loop, eli silmukka suorittaa nimensä mukaisesti toiminnon aina uudelleen, kunnes asetettu ehto täyttyy.

```

#include <AutoConnect.h>
#include <WiFi.h>           // ESP32 WiFi-yhteys
#include <WebServer.h>      // ESP32 WebServer

const char* user_api_key = "          "; // User API Key
const char* channel_id = "          ";   // Channel ID
const String write_api_key = "          "; // Write API Key
const String read_api_key = "          "; // Read API Key

WebServer Server;
AutoConnect Portal(Server);
AutoConnectConfig config;
WiFiClient wifiClient;

int vesiPumppu = 33; // PIN
int vesiTasoAnturiVirta = 5; // PIN
int kosteusAnturiVirta = 18; // PIN

void setup() {

    pinMode(vesiPumppu, OUTPUT);
    pinMode(vesiTasoAnturiVirta, OUTPUT);
    pinMode(kosteusAnturiVirta, OUTPUT);

    delay(1000);

```

Kuva 8. Esimerkki koodirakenteesta

2.6 3D-mallinnus

3D-mallinnus on niin kutsuttu tietokonegrafiikan prosessi, jossa kolmiulotteinen objekti luodaan mallinnusohjelman avulla. 3D-mallinnuksella on merkittävä rooli teknologiakehityksen näkökulmasta katsottuna. Mallinnustekniikkaa hyödynnetään niin *rakennusteollisuudessa* (koneohjaus ja rakennusten mallinnus), *arkkitehtuurisuunnittelussa* (rakennusten ja rakenteiden renderöinti), *lääketieteessä* (laitteiden osat, proteesit ja implantit), *kaupallisessa mainonnassa* (kuluttajamarkkinointi ja yritysmaailma), kuin *elokuva- ja videopeliteollisuudessa* (animointi ja visuaalisten komponenttien, kuten hahmojen tai ympäristöjen luonti). 3D-objekteja voidaan luoda automaattisesti tai manuaalisesti luomalla jonkinlaisen perusmuodon, kuten kuution tai pallon, jonka jälkeen manipuloidaan virtuaalisen tilan pisteitä erinäisillä mallinnustyökaluilla. Nämä pisteet muodostavat yhtenäisen verkon ja lopulta varsinaisen objektin. (Petty s.a.)

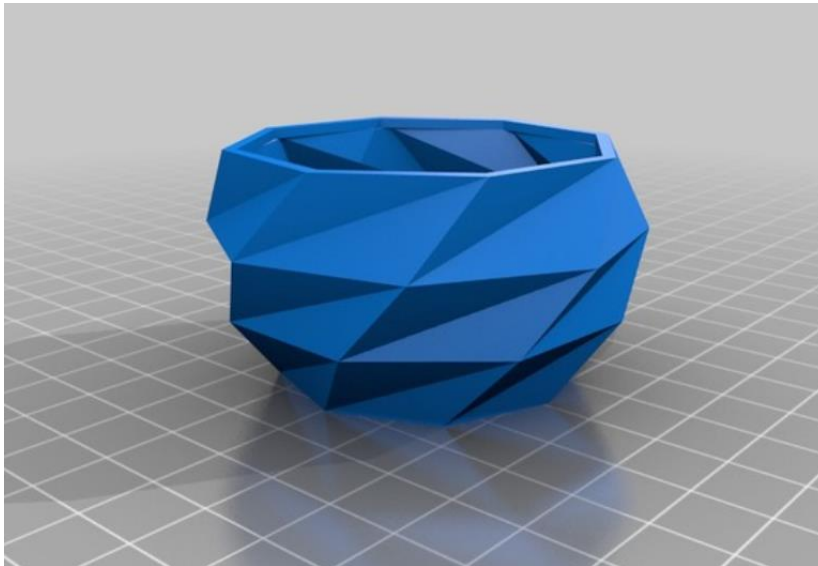
Oli kyse sitten rakennusalan suunnittelutyöstä, arkkitehtuurista, tuotesuunnittelusta tai elokuva- ja videopelisuunnittelusta, valittavana on monia tarpeisiin soveltuvia 3D-mallinnusohjelmia. Jokaisella mallinnusohjelmalla on ominaisuuksia, jotka palvelevat paremmin käyttötarkoituksia kuin toiset. Siksi käyttäjän on hyvä tietää, millaista kolmiulotteista objektia olisi tarkoitus luoda ja mikä ohjelma palvelisi parhaiten omaa tarkoitusta. Yhtenä esimerkkinä on tietokoneavusteiset suunnittelu- ja valmistusohjelmat (CAD/CAM), joita käytetään lähinnä prototyyppien valmistamiseen ja valmiiden osien suunnitteluun. Tähän edellä mainittuun käyttötarkoitukseen soveltuisi parhaiten

pilvipohjainen 3D-mallinnuksen ohjelmistoalusta Fusion 360. Käyttäjällä on valittavanaan laaja kirjo erilaisia mallinnusohjelmia, joihin tutustumalla pääsee parhaiten alkuun omassa suunnittelutyössä saaden siitä parhaan mahdollisen hyödyn.

Yksi suosituimmista mallinnusohjelmista on ilmainen ja avoimeen lähdekoodiin perustuva Blender. Blender on C/C++-ohjelmointikielipohjainen 3D-mallinnusohjelma, joka sisältää ainutlaatuisia työkaluja sekä alan standardien mukaisia ominaisuuksia, joita monet ammattilaiset, tutkijat, studiot, taiteilijat ja animaattorit hyödyntävät tuodakseen mielikuvituksensa visuaaliseen maailmaan. Blenderillä on kattava mallinnustyökaluvalikoima, joka mahdollistaa muun muassa animoinnin, teksturoinnin, renderöinnin sekä kolmiulotteisten mallien muuntamisen ja muokkaamisen. (Blender Foundation 2021.)

Mallinnusohjelmassa on myös sisäänrakennettu Python-tulkki, joka latautuu käynnistyksen yhteydessä pysyen aktivoituneena koko käyttäjäistunnon ajan. Blenderin sulautettu tulkki tarjoaa pääsyn ohjelman omiin tietoihin, luokkiin ja toimintoihin, joita hyödyntämällä pystytään päivittämään sekä määrittelemään käyttöliittymän eri ominaisuuksia. Osa 3D-mallinnusohjelmien vaihtoehtoista ovat ilmaisia ja osa vaatii kuukausittaista tai kertaluontoista summaa maksettavaksi. Ilmaisten ohjelmien kohdalla käyttäjältä vaaditaan usein teknistä tietämystä sekä oma-aloitteisuutta tutustua eri mallinnustyökaluihin, sillä osa mallinnusohjelmien käyttöliittymistä saattavat olla aloittelijalle lähtökohtaisesti vaikeita ja sekavia. Ammattikäyttöön soveltuvat ohjelmat ovat usein maksullisia, mutta tarjoavat asiakaspalvelutukea ongelmatilanteiden sattuessa. Ilmaisisissa ohjelmissa on harvoin asiakaspalvelutuen mahdollisuutta, mutta esimerkiksi Blenderillä on laaja yhteisö ja foorumi, josta voi saada apua muilta 3D-suunnittelijoilta kasvattaen samalla perusteellisen tukiverkoston yhteisöllisyyttä.

3D-teknologia on mahdollistanut muun muassa rakennusten, pihojen sekä sisustuksen suunnittelun kustannustehokkaasti helpottaen samalla hankkeista koituvaa työtaakkaa. Se ollut korvaamaton apuväline lääketieteessä, jossa potilaan vahingoittuneista kehon osista on voitu mallintaa ja muotoilla tarkkuutta vaativia uniikkeja implantteja. 3D-tekniikan avulla voidaan luoda myös monimutkaisia ja yksityiskohtaisia osia esimerkiksi ajoneuvoihin sekä lentokoneisiin tai luoda uuden osan hajonneen esineen tilalle.



Kuva 9. 3D-mallinnus – esimerkkinä mallinnus kukkaruukusta (Riganti 16.7.2018.)

2.7 3D-tulostus

3D-tulostus on lisävalmistusprosessi, jonka avulla luodaan kolmiulotteisia kiinteitä objekteja digitaalisen mallinnuksen mukaan. Valmistusprosessi mahdollistetaan tulostamalla ohuita materiaalikerroksia peräkkäin ja sulattamalla ne lopulta yhtenäiseksi 3D-objektiksi. Kolmiulotteisten objektien tulostus on vaihtoehto perinteiselle tuotteen valmistusmenetelmälle, jossa halutut esineet on suunniteltu modifioimalla raaka-aineita ja käyttämällä niille tarkoitettua muotteja prosessin loppuunsaattamiseksi. (Techopedia 2018.)

3D-tulostuksen avulla voidaan tuottaa monimutkaisia muotoja käyttämällä materiaalia huomattavasti vähemmän verrattuna perinteisiin valmistusmenetelmiin. 3D-tulostustekniikka on lisännyt myös valmistuksen tuottavuutta, jolla on pidemmällä aikavälillä potentiaalia myös sellaisiin massatuotantoprosesseihin, jotka vaativat suurempia tulostusnopeuksia. Tekniikkaa on hyödyntänyt nykyisten tulostusnopeuksien puitteissa muun muassa pienvalmistajat, jotka kehittävät markkinoille prototyyppien osia ja laitteita pienemmissä erissä. 3D-tulostuksen ansiosta prosessista aiheutuvat materiaalikustannukset vähenevät ja tuote saadaan tehokkaan suunnittelutyön tuloksena nopeammin markkinoille.

Kolmiulotteista tulostustekniikkaa käytetään lähes kaikilla toimialoilla. Näistä esimerkkinä ovat muun muassa kuluttajalle lanseeraamat tuotteet, teollisuudessa käytettävät varaosat ja työkalut, terveydenhuoltoon suunnitellut proteesit, arkkitehtuurin mukaiset pienoismallit sekä elokuvissa käytettävät lavasteet. 3D-tulostus on suuressa suosiossa myös DIY-harrastajien keskuudessa, sillä tulostimet ja siihen tarvittavat materiaalit ovat suhteellisen

halpoja. Tämä antaa mahdollisuuden toteuttaa mitä monimutkaisimpia projekteja, jossa rajana on vain käyttäjän oma mielikuvitus.

3D-tulostukseen soveltuvia laitteita löytyy useita eri malleja, jotka sopivat niin kotikäyttöön kuin yritystoimintaan. 3D-tulostimien ominaisuudet vaihtelevat ja siksi on kiinnitettävä huomiota muun muassa tulostimen resoluutioon, tulostusnopeuteen sekä tulostuskokoon. Yleisin vaihtoehto on *FFF-tulostin*, joka on edullinen ja helppokäyttöinen, eikä sen käyttö vaadi sen suurempaa erityisosaamista. Tulostusprosessissa kelaan asetettu lankamateriaali syötetään tulostimeen, jonka jälkeen lanka lämmitetään ja materiaalikerrokset sulautuvat kokonaiseksi massaksi muodostaen tulostusalustalle 3D-objektin. Lankatulostimissa tulostuksen laatu ei ole kovin tarkkaa ja useimmiten pinta jää rosoiseksi.

Paremmen tulostustarkkuuden saamiseksi vaihtoehtona on suosittu, usein ammattikäyttöön soveltuva *SLA-tulostin*, joka tulostaa pienemmätkin yksityiskohdat tarkasti. SLA-tekniikalla 3D-malli rakentuu nestepohjaisesta hartsista, joka kovettuu UV- ja LED-valolla. (Scandinavian Photo s.a.) SLA-tulostin on ideaali prototyyppien kehittämiseen, lääketieteellisiin malleihin, miniatyyrien, pelinappuloiden ja koteloiden tulostamiseen, jotka sisältävät monimutkaisempia yksityiskohtia. Vaikka SLA-tekniikka on yksi markkinoiden tarkimmista tulostustekniikoista ja sillä voi luoda melkein mitä vaan, tulostaminen vie yleensä kauan aikaa ja kustannukset ovat suhteellisen korkeat.

FFF-tulostimessa tulostusmateriaalina käytetään yleisimmin perinteistä *PLA*:ta. Se on ympäristöystävällinen materiaali, joka tuottaa kiinteän ja napakan tulosteen. Materiaali ei ole kovin joustava, jonka vuoksi se katkeaa tarpeeksi kovaa väännettäessä. PLA sopii erityisesti 3D-tulostuksesta kiinnostuneille aloittelijoille, sillä se on halpa ja helppotulosteinen. Toisena suosittuna tulostusmateriaalina käytetään *PETG*:iä. PETG on materiaaliltaan kestävämpi ja joustavampi kuin PLA, mutta pinnaltaan hieman pehmeämpi. Materiaali on alttiimpi naarmuille ja pintavaurioille sileän pintansa takia, mutta kestää voimakkaampia iskuja yllättävän hyvin. PETG on hintaluokaltaan kuitenkin hieman kalliimpi kuin muut materiaalit. Joustavuus, lämpötila ja iskunkestävyys tekevät materiaalista ihanteellisen esimerkiksi mekaanisille osille, jotka kokevat jatkuvaa tai äkillistä kuormitusta. PETG on elintarviketurvallinen ja siksi erityisen suuressa suosiossa muun muassa juomapullojen valmistuksessa. (Slump 27.5.2021.)

SLA-tulostimessa käytetään puolestaan nestepohjaista materiaalia, eli hartsia. Hartsia on erittäin joustavaa ja useimmiten läpinäkyvää, joka jäljittelee kumin kaltaista materiaalia. Hartsia käytetään etenkin tarkkuutta vaativissa prototyypeissä, jotka kestävät kulutusta ja

taivutusta säilyttäen samalla muotonsa. Materiaali on tasaista ja mahdollistaa pientenkin yksityiskohtien tulostamisen. Hartsia käytetään lääketieteessä muun muassa laitteiden ja proteesien mittatilaustyössä sekä miniatyyrien, korujen ja elokuvalavasteiden valmistuksessa. Hartsitulostus ei ole kuitenkaan parhain vaihtoehto suurikokoisten objektien tulostamiseen, sillä se on merkittävästi kalliimpaa kuin muut materiaalit. Hartsitulostus vaatii myös kolmivaiheisen jälkikäsittelyn ja monet tukirakenteet, jotka tuovat huomattavan paljon lisätyötä halutun objektin loppuunsaattamiseksi.

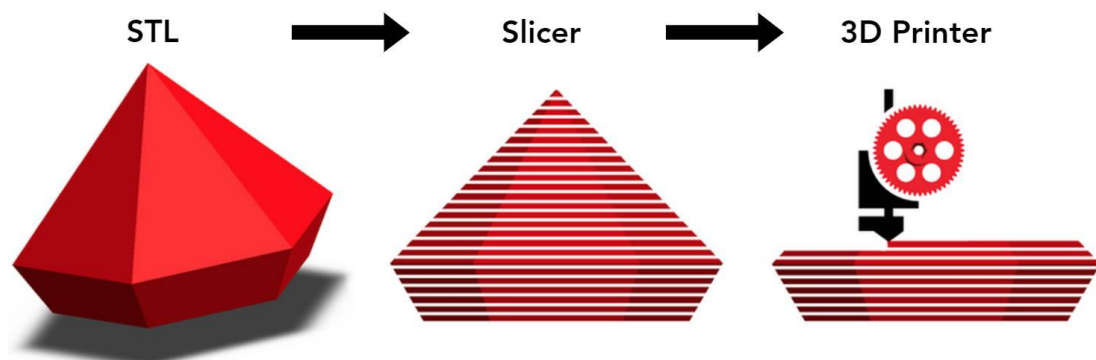


Kuva 10. 3D-tulostus – esimerkkinä valmis kukkaruukku (Riganti 16.7.2018.)

Viipalointi

3D-mallien tulostuksessa käytetään sliceria, eli niin kutsuttua viipalointia, jonka ansiosta voidaan määritellä tulostimelle tietyt ohjeet, minkä pohjalta haluttu objekti muodostetaan. Viipaloinnilla tarkoitetaan nimensä mukaisesti 3D-mallinnuskuvan leikkaamista satoihin eri vaakasuoriin 2D-kerroksiin, jotka tulostetaan aina yksi kerros kerrallaan. Koska 3D-mallin viipalointi on suhteellisen vaikeaa, olisi hyvä valita helppokäyttöinen viipalointiohjelma. Yksi markkinoiden parhaimmista ja helppokäyttöisimmistä viipalointiohjelmista on avoimeen lähdekoodiin perustuva Ultimaker Cura. Sen lisäksi, että Cura on ilmainen, se sisältää vain vähän teknisiä asetuksia, joka on etenkin aloittelevien 3D-suunnittelijoiden mieleen.

Digitaalisten 3D-mallinnusten tulostamiseksi käytetään standardisoitua tiedostomuotoa STL (.stl), sillä kaikki mallinnusohjelmat eivät kata jokaista olemassa olevaa formaattia. Siksi ongelmana on usein käyttäjän luoman mallinnuksen tiedostomuodon yhteensopivuus viipaloinnin ja tulostuksen kanssa. Kun objekti on muutettu ymmärrettävään muotoon, määritetään tulostukseen liittyvät asetukset. Käyttäjän syöttämät tulostusohjeet sisältävät 3D-tulostusparametreja, kuten esimerkiksi objektin kerroksen korkeuden, tulostusnopeuden sekä muiden osien sijoittelun. Tässä vaiheessa tulostettavan objektin kokonaismittoja voidaan vielä tarvittaessa muuttaa, joka mahdollistaa tukirakenteiden lisäämisen, jotta malli pysyy pystyssä koko tulostusprosessin ajan. Kun objekti on tulostettu, voidaan tukirakenteet poistaa. (Carolo 13.4.2020.)



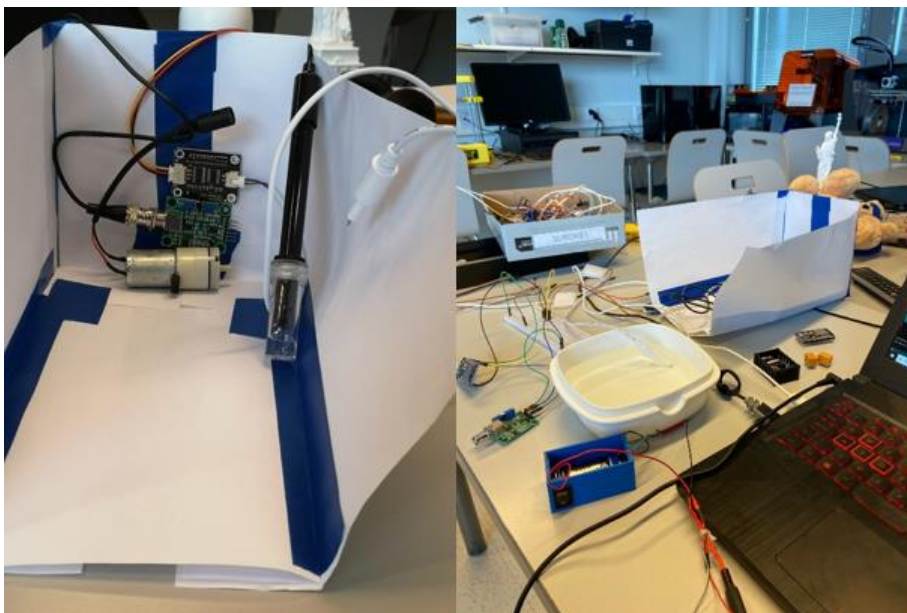
Kuva 11. 3D-mallin viipalointi tulostusta varten (3DMaker Engineering s.a.)

3 Älyruukku

Opinnäytetyönä olemme rakentaneet automatisoidun kastelujärjestelmän, jota voidaan hallinnoida sille rakennetun applikaation kautta. Ruukku on mitoitettu ikkunalaudalle sopivaksi ja on harrastetoimintaan soveltuva. Älyruukussa on vaihtoehtoina kaksi kasvatustekniikkaa: multaversio ja vesiviljely. Vesiviljelyssä ruukkuun on asetettu pH- ja TDS-anturit, jotka mittaavat veden pH- ja ppm-arvoja sekä näyttävät mittaustulokset reaaliaikaisena käyttäjälleen. Multaversiossa ruukkuun asetetaan multa- ja vesitasoanturit, jotka puolestaan mittaavat mullan kosteutta sekä vesisäiliön veden tasoa. Multaversiossa käyttäjällä on mahdollisuus kastella kasvi joko automaattisesti tai manuaalisesti. Opinnäytetyössä keskityimme vesiviljelytekniikan rakentamiseen ja suunnitteluun.

3.1 Ruukun rakentaminen

Suunnittelutyön ensimmäisessä vaiheessa pohdimme millainen ruukusta voisi mahdollisesti tulla. Mietimme mitä kaikkia ominaisuuksia se voisi sisältää ja mitä olemassa olevia koulun komponentteja voisimme hyödyntää. Katsoimme harrastelijoiden YouTube-videoita, luimme artikkeleita sekä selasimme Internetistä ihmisten tekemiä 3D-ratkaisuja. Tilasimme loput tarvittavat komponentit eBaysta ja aloitimme ruukun suunnittelun. Demonstroimme ideaamme rakentamalla A4-paperista realistisen kokoisen kehyksen, jonne asetimme tarvittavat komponentit, jotta meidän olisi helpompi lähteä mallintamaan varsinaista ruukkuja. Testasimme komponenttien toimivuutta Arduinon ohjelmointiympäristössä valmiiden sketsien avulla ja kalibroimme mittareita, jotta saisimme sarjamonitorissa näkyviin oikeat arvot.



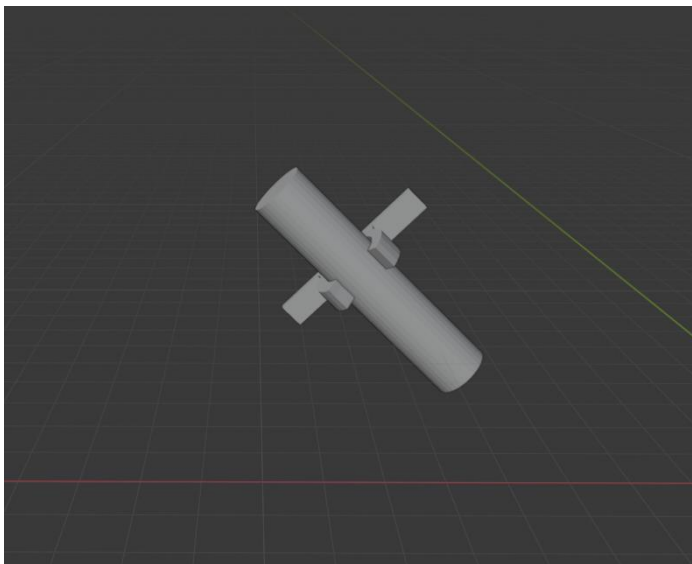
Kuva 12. Raakile – Ruukun ensimmäinen suunnitteluvaihe

3.1.1 3D-mallinnus ja tulostus

Suunnittelutyön toisessa vaiheessa aloimme hahmottelemaan ruukkua 3D-grafiikan mallinnusohjelma Blenderillä. Pohdimme millaisia rajoitteita 3D-tulostuksessa tulisi olemaan ja millainen ruukku olisi realistisesti mahdollista toteuttaa. Ensimmäisenä otimme selvää, minkä kokoisia esineitä koulun 3D-tulostimet pystyivät tulostamaan ja oliko meillä tarvittava määrä tulostuslankaa. Saimme selville, että koulun Colido Diy-tulostimen tulostuspinta-ala oli 200x200x170mm. Vaihtoehtoina meillä oli tulostaa ruukku parissa eri osassa ja liimata osat yhteen tai tulostaa isommat osat toisella 3D-tulostimella.

Päädymme lopulta Jaakon omistamaan Ender 5+ -tulostimeen, joka mahdollisti maksimissaan 350x350x400mm kokoisten esineiden tulostamisen. Suunnittelimme ruukun 320x150x100mm kokoiseksi, jotta tulostusaika lyhenisi noin 22 h pituiseksi. Päätimme yhteistuumin, että ruukussa olisi mahdollista kasvattaa kuutta yrttiä kerrallaan ja aloimme etsimään ihmisten tekemiä 3D-ratkaisuja Internetistä. Keräsimme ideoita ylös muiden mallintamista tuotteista ja sovelsimme niiden parhaimpia ominaisuuksia työssämme.

Suunnittelutyön kolmannessa vaiheessa aloimme hahmottelemaan komponenttien sijoittamista ruukun sisäosaan ja mietimme, miten pH- ja TDS-anturit saataisiin kiinnitettyä ruukkuun kiinni mahdollisimman esteettisesti. Ensimmäinen 3D-mallinnettava osa oli anturin kiinnike, jotta pH-mittari pystyttiin asettamaan kiinni ruukun seinämään.



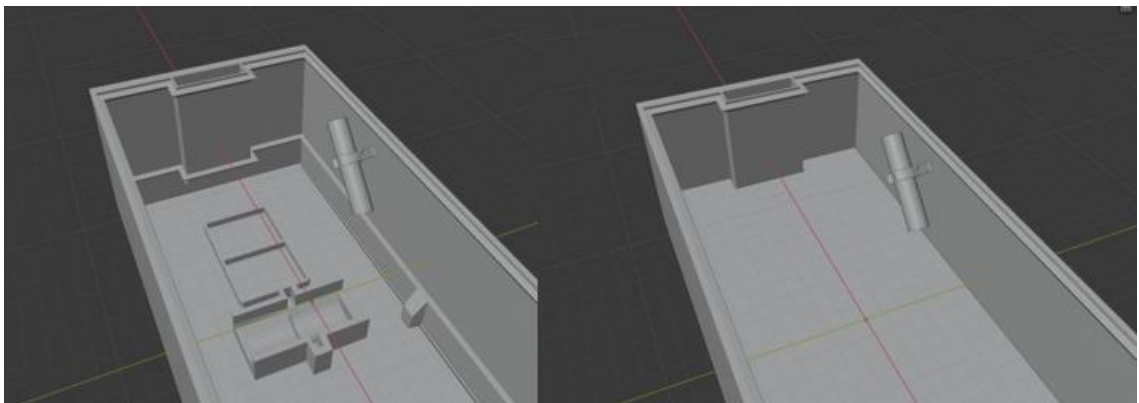
Kuva 13. 3D-mallinnus – Anturin kiinnike

Tulostuksen ensimmäinen versio oli kuitenkin liian leveä tarkoitukseemme, sillä anturia ei saanut kätevästi painettua telineeseen kiinni ja jouduimme tekemään muutoksia mallinnukseen. Päädyimme kaventamaan kiinnikettä, jotta tulostusmateriaali antaisi periksi ja se olisi tarpeeksi tukeva anturin kiinnittämiseen sekä irrottamiseen. Yhdellä kiinnikkeellä anturit olivat kuitenkin liian heikosti kiinni, jonka vuoksi teimme kaksi kiinnikettä yhtä anturia kohden.



Kuva 14. 3D-tulostus – Anturin kiinnike

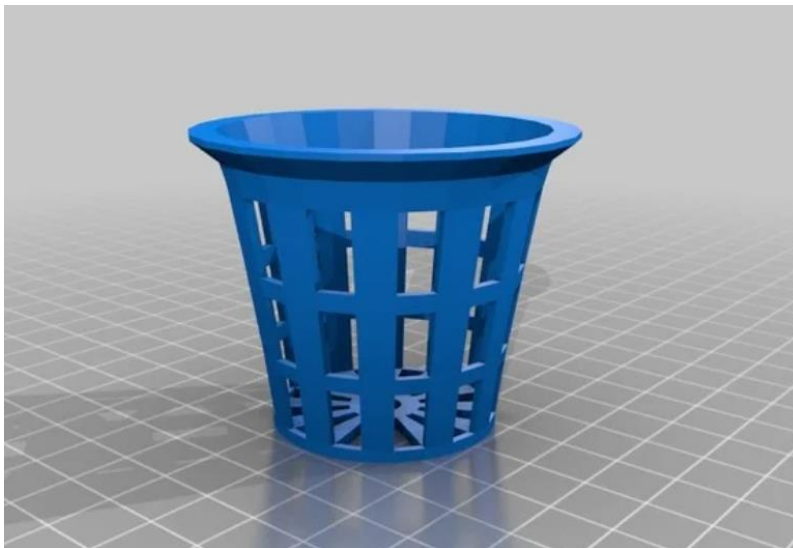
Anturin kiinnikkeen suunnittelun ja tulostuksen jälkeen pohdimme, miten tarvittavat komponentit saatiin asetettua paikalleen ja kuinka sähköjohdot saataisiin asetettua antureille sekä vesitilassa sijaitsevalle ilmaletkulle. Ratkaisuna tähän oli erillinen tekniikalle tarkoitettu pohja ja ruukun seinämään sijoitettava kapea kuilu, jota pitkin johdot saataisiin kätevästi tuotua. Tavoitteenamme oli saada mahdollisimman esteettinen ratkaisu valtavalle johtomäärälle. Tekniikkapohjan tulostusajaksi muodostui 10 h 40 min.



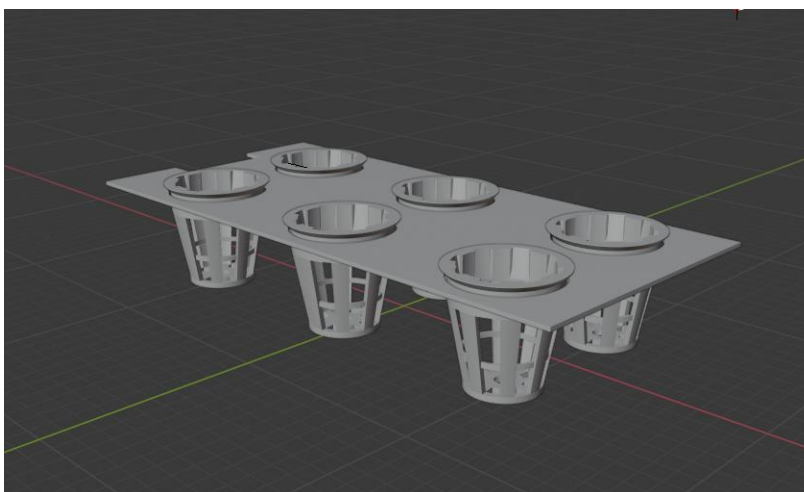
Kuva 15. 3D-mallinnus – Tekniikkapohja

Tekniikalle varattuun pohjaan suunnittelimme telineen ilmapumpulle, releelle sekä anturien piirikorteille. Varsinaiseen ruukkuun suunnittelimme pohjan, jotta ylempänä oleva vesi ei pääsisi piirikorttien kanssa kosketuksiin. Pohjan pysyvyyden takaamiseksi teimme ruukkuun niin kutsutun kauluksen, joka pitäisi pohjaa paikallaan.

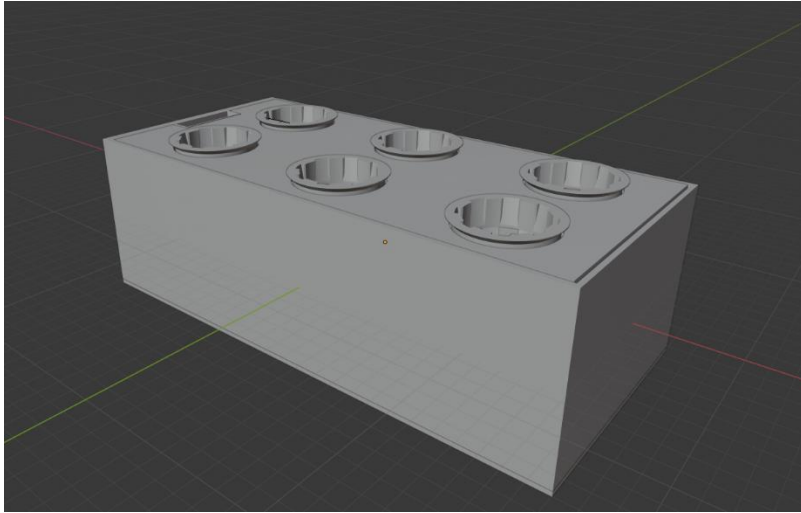
Seuraavaksi aloimme suunnittelemaan kasveille omia kasvatusruukkuja. Aihetta tutkiessamme löysimme valmiin mallin, joka oli juuri sopiva tarpeisiimme. 3D-mallinnuksia suunnitteleva ja työstävä Tom Schönlau oli julkaissut Thingiverse-palvelussa itse tekemän kasvatusruukun, jota lähdimme soveltamaan omaa käyttötarkoitustamme varten. Lähdimme muokkaamaan ruukun kokoa sopivaksi ja suunnittelimme siihen muotin, jotta kasvatusruukut saataisiin upotettua telineeseen. Parin tulostustestin jälkeen saimme sopivan kokoisen muotin, josta tuli ruukullemme myös kansi.



Kuva 16. 3D-mallinnus – Yksittäinen kasvatusruukku (Schönlau 4.4.2018.)

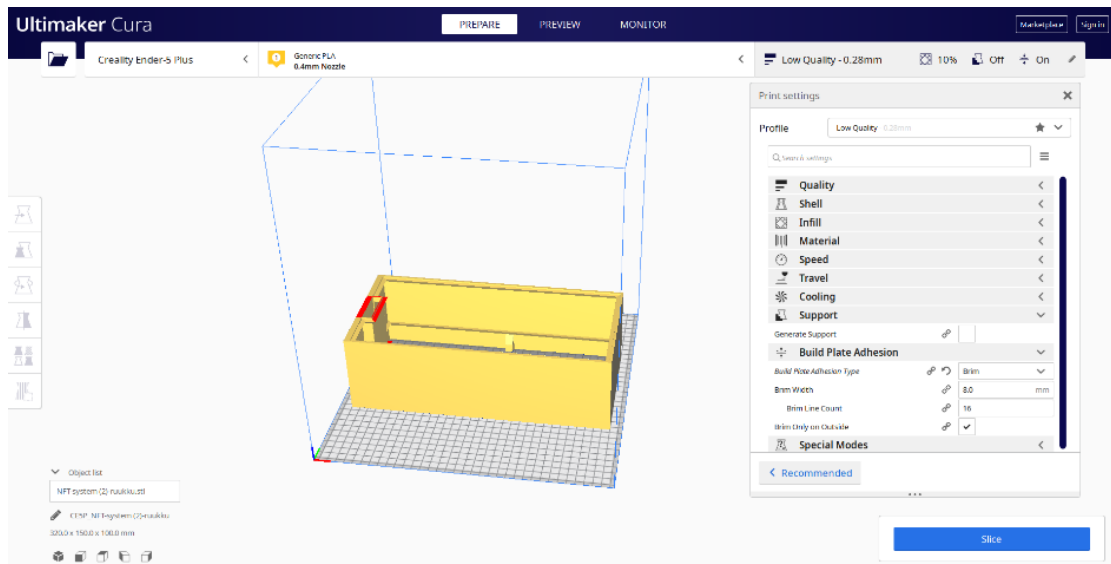


Kuva 17. 3D-mallinnus – Valmis ruukkuteline kasvatusruukkuineen



Kuva 18. 3D-mallinnus – Valmis ruukku

Seuraavana toimenpiteenämme oli tulostaa ruukun osat. Tutustuimme viipalointitekniikkaan, joka mahdollisti tiedostojen muuntamisen tulostimelle sopivaan muotoon. Käytimme työssämme Ultimaker Cura Sliceria, joka on avoimeen lähdekoodiin perustuva tehokas viipalointityökalu. Pystyimme tuomaan Blenderillä suunnittelemamme mallinnukset suoraan Slicer-ohjelmistoon, jossa stl.-tyyppiset tiedostot muunnettiin .gcodeksi. Tulostusvaiheessa valitsimme 3D-tulostimen mallin, jonka ansiosta ohjelma pystyi selvittämään mitat sekä muut tulostusta koskevat oleelliset tiedot.



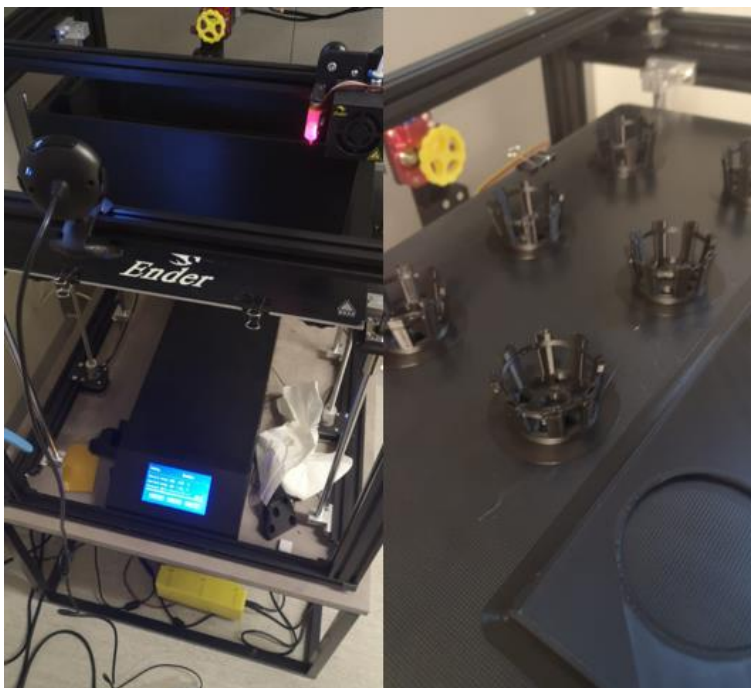
Kuva 19. Ultimaker Cura – tulostusnäky

Tulostusvaiheet

Kasvatusruukkujen tulostamisen ensimmäinen vaihe epäonnistui, sillä tulostus keskeytyi langan loppumisen vuoksi. Toisella yrityskerralla tulostus onnistui, mutta kasvatusruukkujen ympärille muodostui ylimääräistä lankaa, joka johtui todennäköisemmin väärästä lämpötilasta sekä tulostuksen nopeudesta. Asetusten muuttamisella olisi ollut mahdollista korjata ja parantaa tulostuksen laatua, mutta lopputulos oli mielestämme tarpeeksi hyvä. Saimme siistittyä osaan kiinnittyneet ylimääräiset langat puukolla, jonka ansiosta uutta tulostuskertaa ei tarvittu.

Ruukun kehyksen tulostamisen ensimmäisessä vaiheessa kohtasimme meille tuntemattoman ongelman, jonka vuoksi jouduimme tulostamaan kehyksen uudestaan. Toinen yrityskerta tuotti haluamamme lopputuloksen ja saimme tulostettua ruukun ongelmitta. Onnistuneen version tulostusajaksi kertyi yhteensä 22 tuntia.

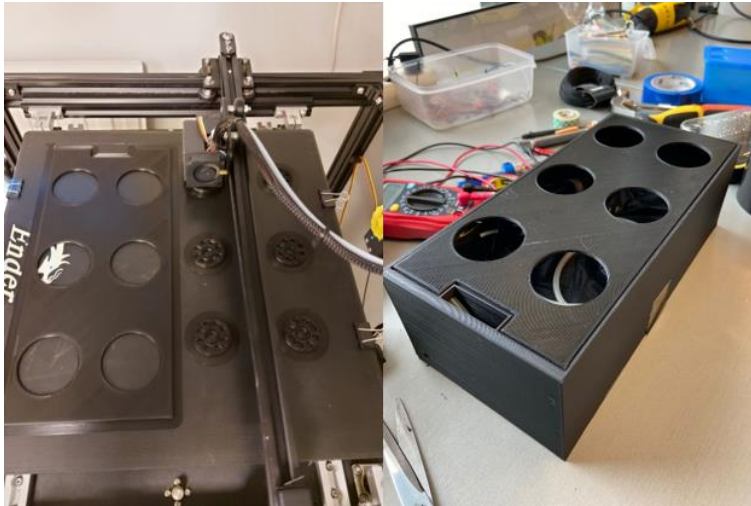
Muiden osien tulostukset onnistuivat ensimmäisellä kerralla. Kaikki osat laitettiin yön aikana tulostukseen, jotta pystyimme jatkamaan päivällä ruukun tekoa eteenpäin. Koska tulostusajat olivat sen verran pitkiä ja itse tulostin sijaitsi varastohuoneessa, valvoimme prosessin etenemistä kameran avulla. Tämä antoi meille mahdollisuuden tarvittaessa reagoida nopeasti esimerkiksi langan loppumiseen tai muihin teknisiin ongelmiin.



Kuva 20. 3D-tulostus – Kasvatusruukkujen tulostusprosessi

Tulostusajat

Osa	Tulostusaika
Ruukun kehys	n. 22 tuntia
Ruukun välipohja	6 tuntia 30 minuuttia
Tekniikkapohja	10 tuntia 40 minuuttia
Ruukkuteline	n. 5 tuntia
Kasvatusruukut 6 kpl	yhteensä n. 11 tuntia
Anturitelineet 4 kpl	yhteensä n. 48 minuuttia



Kuva 21. 3D-tulostus – Valmis ruukkuteline

Ruukun haastavimmaksi ongelmakohtaksi muodostui vesitiiviys, joka on 3D-tulostettujen esineiden tyypillisin päänsä. Tähän ongelmaan löysimme kuitenkin ratkaisun valuepoksihartsista. Aloitimme varsinaisen prosessin tiivistemassalla, jota levitimme pohjan reunoihin. Käytimme tiivistemassaa myös välipohjan liimaamiseen, jonka jälkeen siirryimme testaamaan valuepoksihartsin toimivuutta. Aloitimme ruukkutelineestä, sillä halusimme varmistaa kyseisen aineen tehokkuuden ennen varsinaiseen ruukkuun siirtymistä. Jouduimme odottamaan hartsin kuivumista pari päivää ennen kuin saimme lisättyä siihen uuden kerroksen. Hartsia levitettiin lopulta ruukun pohjaan ja seinämiin muutama otteeseen varmistaen, ettei ruukkuun jäisi mahdollisia veden mentäviä rakoja. Valuepoksihartsikerroksen jälkeen teimme testin ruukun vesitiivyydestä. Ruukusta tuli odotuksenmukainen ja tekniikkapohjassa sijaitsevat komponentit säilyivät kuivina.



Kuva 22. Ruukkuteline – valuepoksihartsin testaus

Kun vedenpitävyys oli varmistettu ja tarvittavat osat tulostettu, saimme koottua valmiin ruukun yhdeksi kokonaisuudeksi. Alla olevassa kuvassa ruukku on sijoitettu hyllylle, jotta lukijalle hahmottuisi sen todelliset mittasuhteet kodin ympäristössä.



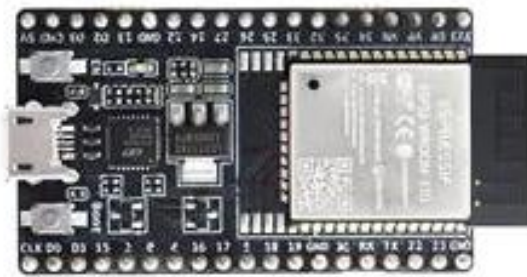
Kuva 23. 3D-tulostus – Valmis ruukku hyllyllä

3.1.2 Komponentit

3D-tulostetun ruukun valmistuttua aloimme kartoittamaan olemassa olevia komponentteja, joiden avulla olisi mahdollista toteuttaa halutut ruukun toiminnallisuudet oman budjettimme rajoissa. Tärkeimpänä tavoitteenamme oli löytää mahdollisimman monikäyttöinen mikrokontrolleri, johon olisi integroitu Bluetooth ja Wi-Fi. Päädyimme lyhyen harkinnan jälkeen pienitehoiseen ja edulliseen ESP32-järjestelmäsiruun, sillä kyseinen siru oli meille jo ennestään tuttu sekä olimme todenneet sen sopivaksi käyttötarkoituksiamme varten. Antureiden osalta meillä ei ollut juurikaan vaihtoehtoja, sillä ammattikäyttöön tarkoitetut anturit maksoivat sen verran paljon, ettei budjettimme antanut siihen myöten. Päätöksemme jälkeen aloimme pohtimaan, miten saisimme komponentit toimimaan yhtenä kokonaisuutena, jotta haluttu lopputulos olisi mahdollista toteuttaa.

ESP32-DevKitC-32D

Käytimme älyruukussa ESP32-järjestelmäsirua antureiden lukemiseen sekä datan lähettämiseen ja vastaanottamiseen. Siru toimii niin sanotusti pääkomponenttina kokonaisuudelle, joka kommunikoi ThingSpeak-palvelimen kanssa. ESP32-järjestelmäsiru on varustettu yhdellä sisäänrakennetulla ydinprosessorilla, joka pystyy suoriutumaan yhdestä tehtävästä aina kerrallaan. Järjestelmäsiru prosessoi koodia ja funktioita rivi riviltä, minkä vuoksi se ei pysty lukemaan kahden anturin lukuja samanaikaisesti. Edellä mainittu asia ei kuitenkaan estänyt meitä käyttämästä sirua, mutta antoi arvokasta tietoa muun muassa koodien suunnittelun osalta.



Kuva 24. ESP32-DevKitC-32D (Mouser s.a.)

Ilmapumppu

Älyruukussa käyttämämme ilmapumppu toimii suhteellisen yksinkertaisesti, mutta on toiminnaltaan juuri sopiva tarpeisiimme. Ilmapumpulle syötetään jännitteellä virtaa, jonka ansiosta pumpun sisällä oleva propelli lähtee pyörimään tiivistäen ilman letkuun ja luoden samalla painetta. Ilma johdetaan letkulla älyruukun vesitilaan, jonka vuoksi veden happiarvot pysyvät halutussa arvossa. Virransyötön ohjaus tapahtuu releen kautta.



Kuva 25. Ilmapumppu ja letku (electric_studio s.a.)

Rele

Releellä on useimpia toimintaperiaatteita, mutta yleisimpiä näistä ovat NC ja NO (*normally open* ja *normally closed*). NO-rele toimii syöttämällä ohjausvirtaa käämiin. Virta luo puolestaan magneettikentän, joka yhdistää A- ja B-piirit ja näin ollen virtapiiri sulkeutuu antaen virran kulkea läpi. NC-rele puolestaan toimii käänteisesti. A- ja B-piiri ovat yhdistettyinä niin kauan, kunnes käämille syötetään ohjausvirtaa vetäen kärjet irti toisistaan. Älyruukun kohdalla ESP32:ltä syötetään ohjausvirtaa NC-releelle. Rele toimii tässä tapauksessa ilmapumpun varsinaisena katkaisijana. Kun ilmapumppu halutaan pois päältä, syötetään käämiin ohjausvirtaa, jolloin A- ja B-kärjet irtoavat toisistaan ja virtapiiri katkeaa.



Kuva 26. NO- ja NC-relekaavio (AspenCore s.a.)

Älyruukkua varten tilatun releen kohdalla olimme ottaneet huomioon kastelujärjestelmän käyttötarpeet, sillä kasveille tarkoitettu vesi tarvitsee ilmaa jatkuvasti. Mikäli jostain syystä rele tai ESP32 hajoaisi, NC-rele varmistaisi sen, että pumppu olisi siitä huolimatta päällä.

ESP32-järjestelmäsirun avulla voidaan asettaa 3.3 voltin ohjausjännite, joka mahdollistaa ilmapumpulle tarvittavan ohjausvirransyötön. *digitalWrite*-komennolla syötetään ohjauspinnin numero ja arvo. HIGH-arvo aktivoi virransyötön ja LOW-arvo lopettaa sen. Ohjauspinni on määritelty aikaisemmin luodulla muuttujalla.



Kuva 29. pH-anturi (DFRobot s.a.)

```
static int PHTasoAnturiArvo() {
    int measurements=0;
    float x = analogRead(pHSense);
    for (int i = 0; i < samples; i++)
    {
        measurements += analogRead(pHSense);
        delay(10);
    }
    float voltage = 4.96 / adc_resolution * (measurements/5.75)/samples;
    Serial.print("pH= ");
    Serial.println(ph(voltage));
    delay(3000);

    float returnValue = ph(voltage);

    return returnValue;
};
```

Kuva 30. Koodirakenne – pH-anturin arvojen lukeminen ja palautus

TDS-anturi

TDS-anturi mittaa veteen liuenneiden aineiden kokonaismäärää. Vesi sisältää tyypillisesti mineraaleja, jotka ajan myötä voivat johtaa kalkin kertymiseen eri komponentteihin ja putkiin lyhentäen samalla niiden käyttöikää sekä tarkkuutta. TDS-mittauksen ansiosta käyttäjä pystyy tarvittaessa reagoimaan ja torjumaan edellä mainitun ongelman. Pitoisuudet mitataan ja ilmoitetaan ppm-arvona (parts per million), joka kertoo kuinka paljon litrassa, on milligrammoja. TDS-anturilla ei voida kuitenkaan mitata veden epäpuhtautta, minkä vuoksi se ei sovellu veden laadun määrittämiseen. (Boubel s.a.)

Anturin käyttöönotto oli varsin vaivatonta, sillä valmistaja oli luonut erilaisia testikoodeja, joita oli mahdollista soveltaa omia käyttötarkoituksia varten. Testausvaiheessa huomasimme, miten epätarkka anturi oli, sillä mittaustulokset heittelivät noin 10–20 %. Kasvien kohdalla ppm-arvojen heitto ei ole kuitenkaan haittaava, sillä veden epäpuhtaudet eivät vaikuta merkittäväällä tavalla kasvuolosuhteisiin. Tarkemman mittaustuloksen

saamiseksi on mahdollista käyttää laadukkaampaa ja siltä osin myös kalliimpaa mittaria, joka ei oman budjettimme raameissa ollut kuitenkaan mahdollista. Sovelsimme valmistajan tarjoamia testikoodeja omaa käyttötarkoitustamme varten. Luimme useiden keskustelupalstojen kommentteja, kuinka vedessä toimivat anturit korrosoituvat, mikäli niille syötetään jatkuvasti jännitettä. Muokkasimme edellä mainittujen käyttäjäkommenttien perusteella ppm-arvoja lukevaa funktiota siten, että anturille syötetään virta vain mittauksen ajaksi. Tarkoituksenamme on säilyttää älyruukussa olevat komponentit mahdollisimman pitkäikäisinä.



Kuva 31. TDS-mittari (Keyestudio Wiki 2021.)

```
static int TDSAnturiArvo() {  
  
    digitalWrite(TDSVirta, HIGH); // Virrat päälle -> odotus -> luetaan -> tallennetaan muuttujaan tieto -> virrat pois  
    delay(10000);  
  
    analogBuffer[analogBufferIndex] = analogRead(TdsSensorPin); //read the analog value and store into the buffer  
    analogBufferIndex++;  
    if(analogBufferIndex == SCOUNT) {  
        analogBufferIndex = 0;  
    }  
  
    for(copyIndex=0;copyIndex<SCOUNT;copyIndex++){  
        analogBufferTemp[copyIndex]= analogBuffer[copyIndex];  
        averageVoltage = getMedianNum(analogBufferTemp,SCOUNT) * (float)VREF/ 1024.0;  
        // read the analog value more stable by the median filtering algorithm, and convert to voltage value  
        float compensationCoefficient=1.0+0.02*(temperature-25.0);  
        //temperature compensation formula: fFinalResult(25°C) = fFinalResult(current)/(1.0+0.02*(fTP-25.0));  
        float compensationVolatge=averageVoltage/compensationCoefficient; //temperature compensation  
        tdsValue=(133.42*compensationVolatge*compensationVolatge*compensationVolatge  
        - 255.86*compensationVolatge*compensationVolatge + 857.39*compensationVolatge)*0.5;  
        //convert voltage value to tds value  
    }  
    //Serial.print("voltage:");  
    //Serial.print(averageVoltage,2);  
    //Serial.print("V ");  
    Serial.print("TDS Value:");  
    Serial.print(tdsValue,0);  
    Serial.println("ppm");  
    delay(5000);  
    digitalWrite(TDSVirta, LOW);  
    float returnValue = tdsValue;  
    return returnValue;  
};
```

Kuva 32. Koodirakenne – ppm-arvojen mittaaminen ja palautus

Ruukun hinta

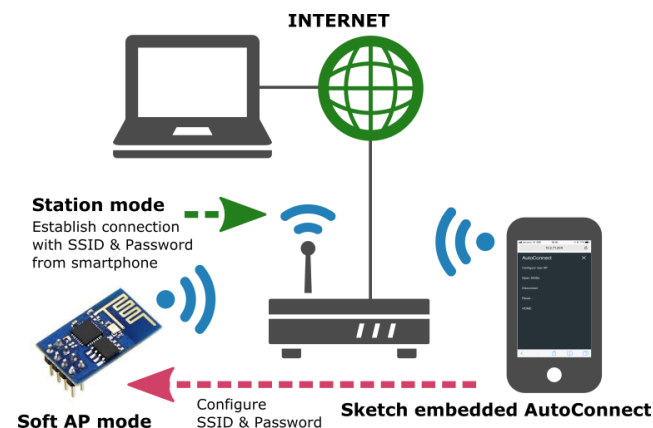
Älyruukkuun tarvittavat komponentit ja tulostusmateriaalikustannukset tulivat maksamaan meille yhteensä 106 euroa. Osan 3D-tulostuslangoista kustansi Haaga-Helian ammattikorkeakoulun 3D-laboratorio. Jouduimme tulostamaan muutaman osan uudestaan epäonnistuneen tulostuksen takia, minkä vuoksi kustannukset nousivat hieman odotettu korkeammiksi. Suurimmat kustannukset tulivat 3D-tulostuslangasta ja 3D-epoksivaluhartsista.

Kustannukset

Osa	Hinta
ESP32-järjestelmäsiru	8 euroa
Jännitteenmuunnin	1 euro
Virtalähde	3 euroa
Rele	1 euro
Ilmapumppu	10 euroa
pH-anturi	15 euroa
TDS-anturi	7 euroa
3D-tulostuslanka	n. 40 euroa
3D-epoksivaluharts	21 euroa

3.2 Integraatio

Älyruukun integroimiseksi loimme ESP32-järjestelmäsiruun toimivan Wi-Fi-yhteyden, jotta pystyimme käyttämään koodissamme ThingSpeakin omaa Arduino-kirjastoa HTTP-pyyntöjen lähettämiseksi ja vastaanottamiseksi. Käytimme Hieromon Iksamoon luomaa AutoConnect-kirjastoa, joka luo käyttäjälle verkkokäyttöliittymänäkymän, mahdollistaa Internet-yhteyden sekä hotspotin luomisen ilman kovakoodattua SSID:tä ja salasanaa.



Kuva 33. AutoConnect-kirjaston toiminnallisuuden dynaaminen rakenne (Ikasamo s.a.)

AutoConnect-kirjasto tarjoaa myös Over the Air (OTA)-ohjelmoinnin, joka mahdollistaa uuden ohjelman päivittämisen tai lataamisen langattomasti. OTA on erittäin hyödyllinen ominaisuus, mikäli ESP32-moduuliin ei pääsisi jostain syystä fyysisesti käsiksi. Tämän ansiosta myös huoltotoimenpiteisiin menee huomattavasti vähemmän aikaa.

AutoConnect

Anycast-6ce52e

36% Ch.10

Game33794

16% Ch.4

AirPort33794

12% Ch.4

B0EB57720745-2G

10% Ch.3

Total:6 Hidden:2

SSID

SSID

Passphrase

Passphrase

Enable DHCP ☒

Apply

Kuva 34. Wi-Fi-yhteyden ja hotspotin luominen verkkokäyttöliittymän kautta

Yhteyden luomisen aikana kirjasto luo ESP32-moduulille oman hotspotpisteen, jonka avulla puhelin, tabletti tai tietokone voidaan yhdistää Wi-Fi-verkkoon. Käyttäjälle avautuu valmiiksi luotu tai itse muokattu käyttöliittymänäkymä, josta löytyy kaikki käytettävissä olevat langattomat yhteydet. Käyttäjä valitsee listalta oman verkon ja syöttää Wi-Fi-verkon salasanan. Kun yhteys on luotu, se tallentuu ESP32-moduuliin ja on valmis käytettäväksi.

Kun Wi-Fi yhteys on luotu, siirrytään Arduinon luonnokseen tekemään tarvittavat muutokset. Aluksi luodaan konfiguraatioon vaadittava muuttuja (engl. *config*), johon täytetään hotspotin nimi, salasana sekä haluttu otsikko. Kirjasto mahdollistaa käyttöliittymänäkymän kokonaisvaltaisen muokkaamisen, mutta päätimme ajan säästämiseksi hyödyntää kirjaston mukana tullutta valmista sivupohjaa. Käyttöliittymänäkymän viimeistelyn jälkeen lisäsimme OTA-päivitysominaisuuden ja määrittelimme mihin komponenttiin sivu avautuu kirjaututtaessa. Config-muuttuja syötetään lopulta funktiolle, joka luo portaalin ja avaa halutun hotspotin.

```

delay(1000);
config.apid = "ePlantVol2";
config.psk = "salasana";
config.portalTimeout = 60000;
config.title = "ePlant";
config.retainPortal = true;
config.ota = AC_OTA_BUILTIN;
config.menuItems = AC_MENUITEM_CONFIGNEW;

Portal.config(config);
Serial.begin(115200);
Serial.println();

if (Portal.begin()) {
  Serial.println("Wifi connected: " + WiFi.SSID());
  Serial.println("Wifi connected: " + WiFi.localIP().toString());
}

WiFi.setAutoReconnect(true); // Automaattinen Wifin yhdistys, jos yhteys katkeaa
// Hotspot-nimi
// Salasana
// 60 sec Hotspot
// Avattavan Hotspot-nettisivu title
// Hotspots koko ajan paalla
// Over-The-Air -päivitys
// Nettisivun komponentit

// Wifi avaa web-portaalin
// Serial monitor

```

Kuva 35. Koodirakenne – Hotspotin lisääminen älyruukun luonnokseen

Kun Internet-yhteys oli luotu, tehtävänämme oli saada anturien arvot näkymään mobiiliapplikaatiossa. Löysimme Arduinon ladattavista kirjastoista ThingSpeakille määritellyn uniikin kirjaston, jonka avulla pystyimme lähettämään tarvittavan datan ThingSpeakin analytiikka-alustalle, josta haluttu data oli mahdollista hakea mobiiliapplikaation avulla. Kirjasto mahdollisti myös sovelluksen rakentamisen ilman palvelinpuolta, eli "back-endiä".

Ensimmäisenä luodaan yhteys ThingSpeakin palvelimeen, jonka jälkeen asetetaan datan lähettämiseen vaadittu muuttuja. Muuttuja sisältää autentikointiin tarvittavan API-avaimen (engl. *API-key*) sekä kanavan (engl. *channel*), joihin syötetään älyruukun anturiarvot. Lopuksi asetimme ylätunnistetiedot (engl. *header*), joka kertoo, missä muodossa data lähetetään ja miten tietoja tulee käsitellä. Kun prosessi on suoritettu onnistuneesti, halutut anturiarvot tulostuvat Arduinon sarjamonitoriin.

```

if (wifiClient.connect(server, 80)) // api.thingspeak.com
{ // Thingspeak tunneli avataan
  // API-key
  String data_to_send = write_api_key;
  data_to_send += "sfield1=";
  data_to_send += PHarvo;
  data_to_send += "sfield2=";
  data_to_send += PPMarvo;
  data_to_send += "";

  wifiClient.print("POST /update HTTP/1.1\n");
  wifiClient.print("Host: api.thingspeak.com\n");
  wifiClient.print("Connection: close\n");
  wifiClient.print("X-THINGSPEAKAPIKEY: " + write_api_key + "\n");
  wifiClient.print("Content-Type: application/x-www-form-urlencoded\n");
  wifiClient.print("Content-Length: ");
  wifiClient.print(data_to_send.length());
  wifiClient.print("\n\n");
  wifiClient.print(data_to_send);
  delay(1000);
  Serial.println("SensorData");
  Serial.println( );
  Serial.println("PH-Value:");
  Serial.println( PHarvo );
  Serial.println( );
  Serial.println("PPM-value:");
  Serial.println( PPMarvo );
  Serial.println( );
}

```

Kuva 36. Koodirakenne – ThingSpeakin integrointi älyruukkuun

3.3 Jatkokehitys

Lopputuloksena muodostui automatisoitu kastelujärjestelmä, joka on hallinnoitavissa sille rakennetun applikaation kautta. Vaikka älyruukku on täysin toimiva ja omaa silmää miellyttävä, tuotteen jatkokehittäminen on tärkeää, jotta lopputuotteesta tulisi entistä parempi, tehokkaampi ja edullisempi. Jatkokehittämisen avulla voidaan parantaa käyttäjäkokemusta ja pitää tuote uusimmat vaatimukset täyttävänä kokonaisuutena.

Ruukku

Jotta saataisiin markkinoille soveltuva ja silmiä hivelevä lopputulos, vaaditaan tekijältä tietynlaista tarkkuutta sekä osaamista 3D-mallintamisesta. 3D-mallinnuksia työkseen tekevät ammattilaiset saattavat suunnitella jopa kuukausia riippuen, kuinka yksityiskohtainen ja monimutkainen lopputuote olisi tarkoitus olla. Koska lähdimme liikenteeseen täysin nollasta sekä osaamisemme koostui Haaga-Helia ammattikorkeakoulun järjestämään ja Heikki Hietalan vetämään 3D-mallinnuskurssin perusteisiin, tietotaidot olivat lähinnä aloittelijan tasolla. Ruukun fyysinen ilme on ylivoimaisesti jatkokehityksen suurin muuttuja. Ruukun muotoa, materiaalia, kokoa ja väriä voidaan kehittää loputtomasti, sillä jokaisella on oma mielipiteensä siitä, millaiselta täydellinen ruukku tulisi näyttää. Rakentamamme älyruukku oli omasta mielestämme tarpeeksi hyvä ja kompakti kokonaisuus opinnäytetyön loppuunsaattamiseksi. Ruukku on esteettinen ja minimalistinen, joka täyttää sille asetetut vaatimukset.

Koodi

Ohjelmointia voidaan tehdä monella eri tapaa. Ensimmäinen tapa on rakentaa niin kutsuttu luurankomalli, jota kehitetään ja täydennetään pala palalta. Toinen tapa on tehdä kokonaisuus, jossa yritetään saada kaikki palaset lokahtamaan paikalleen varmistaen samalla, että kaikki toimii. Jälkimmäisessä tavassa koodirakenne saattaa olla usein sekava, eikä tehokkuudesta ole otettu maksimaalista hyötyä. Koska aikamme oli rajallista, päädyimme rakentamaan toimivan kokonaisuuden, jota oli mahdollista jatkokehittää opinnäytetyön valmistumisen jälkeenkin. Tämä vaihtoehto antoi meille mahdollisuuden saattaa opinnäytetyö päätökseen sille asetettujen aikaraamien sisällä.

Komponentit

Jälkikäteen huomasimme, että ESP32-järjestelmäsirussa on syvä lepotila (engl. *Deep Sleep*), jota emme päässeet hyödyntämään. Syvässä lepotilassa ESP32:n virrankulutus tippuu lähes nolnaan, kunnes siru herätetään sille asetetun komennon avulla. Tämä toimintatila pidentää akkujen ja paristojen käyttöikää ja optimoi virrankulutusta. Koska

meidän käyttötarkoituksessamme sirun ei tarvitsisi olla jatkuvasti päällä, tämä olisi ollut loistava ominaisuus.

Älyruukussa käyttämämme komponentit palvelevat käyttötarkoitustamme hyvin. Mikäli laitteesta haluaisi mahdollisimman tehokkaan ja tarkan kokonaisuuden, olisi mahdollista ostaa laadukkaampia, mutta samalla myös kalliimpia komponentteja. Ammattikäyttöön soveltuvat komponentit ovat pidempi-ikäisiä sekä mittaukset ovat tarkempia ja säännöllisiä. Laadukkaammilla komponenteilla on omat kiinnityspisteet ja kokokin on huomattavasti kompaktimpi. Tämä mahdollistaisi tekniikan supistamisen pienempään ja esteettisempään tilaan.

4 Mobiiliapplikaatio

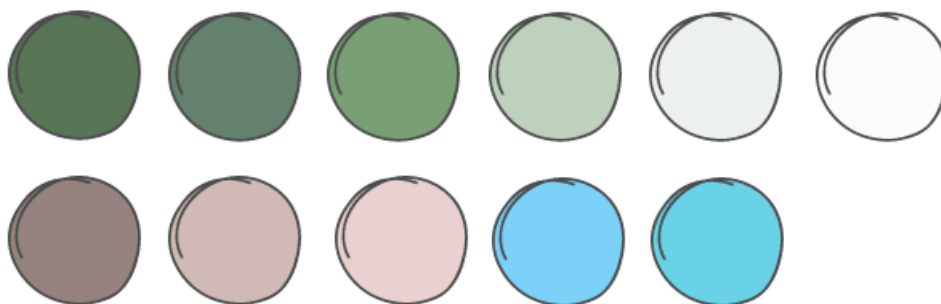
4.1 Ohjelmistoarkkitehtuurin suunnittelu

Mobiiliapplikaation kehittäminen alkoi tietokannan etsinnällä. Tarkoituksenamme oli löytää sopiva tietokanta, joka palvelisi käyttötarkoituksiamme ja olisi mahdollisimman helppokäyttöinen. Useimmat tietokannat olivat maksullisia, kuten esimerkiksi Microsoft Azure-pilvipalvelu, joka vaatii jo muutamien käyttökertojen jälkeen sitoutumista maksulliseen jäsenyyteen. Monissa tietokantapalveluissa laskutetaan joko käytön mukaan tai kuukausittaisin laskutettavan käyttömaksun muodossa. Halusimme ottaa käyttöön ilmaisen tietokannan, sillä budjettimme oli rajallinen, eikä käyttämämme datamäärä tarvinnut maksullisten palvelujen massiivista tallennustilaa. Tutustuimme Google Firebase-tietokantaan, joka vastasi kaikkia tarpeitamme.

Firestore mahdollisti käyttäjien autentikoinnin ja tietojen tallentamisen kuukaudeksi tiettyyn datamäärään saakka ilman minkäänlaista käyttömaksua. Tietokannan löydyttyä aloimme suunnittelemaan luurankomallia siitä, mitä kaikkea tulisimme tarvitsemaan ja miten integrointi voitaisiin toteuttaa. Suunnittelimme kaiken perusteellisesti, sillä emme halunneet ongelmien ilmenevän siinä vaiheessa, kun prosessi olisi jo edennyt loppuvaiheeseen. Käytimme hyödyksemme koululla olevaa fläppitaulua, jossa suunnittelimme vaihe vaiheelta ohjelmistoarkkitehtuurin mahdolliset rakennuspalikat.

4.2 Graafinen ilme

Mobiiliapplikaation graafinen ilme on toteutettu yhtenevällä tyyllillä jokaisen sovellusnäytön kohdalla. Kuvat ja komponenttien sijoittelu noudattavat samaa tyyliä, jotta kokonaisuus olisi mahdollisimman yhdenmukainen. Värimaailma on toteutettu käyttämällä vihreän eri sävyjä, ruskean eri sävyjä, harmaata, valkoista ja sinistä. Tarkoituksenamme on luoda mahdollisimman selkeä, tyylikäs ja esteettinen käyttöliittymä, joka palvelee mutkattomasti antaen samalla miellyttävän käyttäjäkokemuksen.



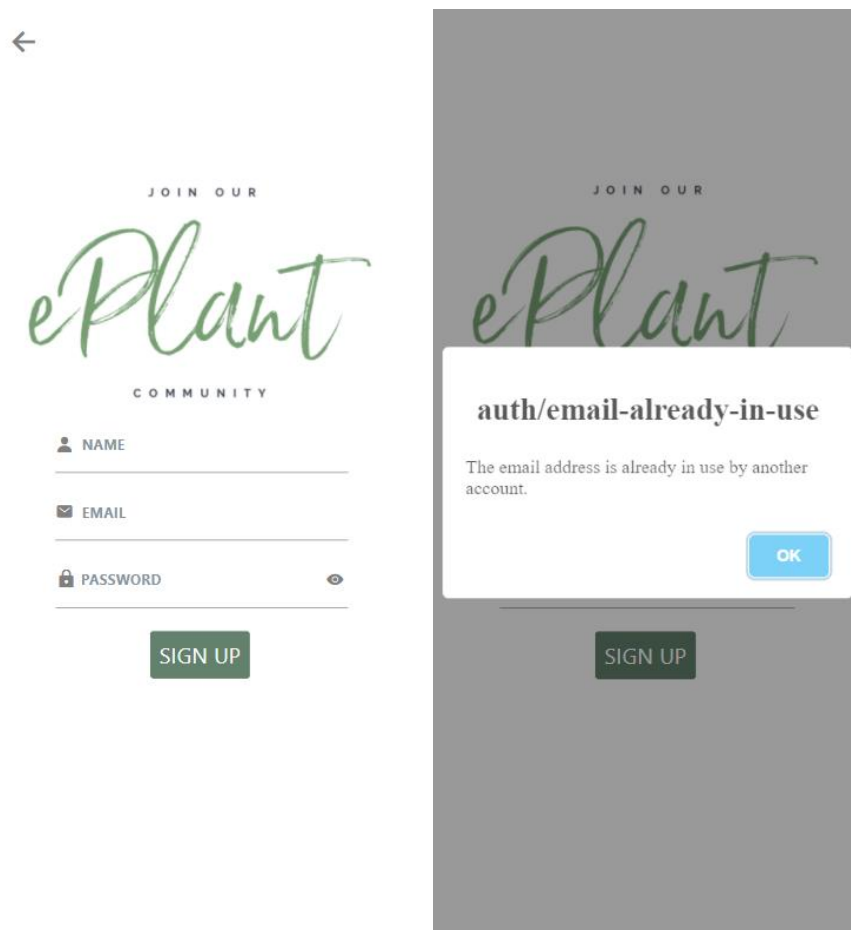
Kuva 37. Väripaletti mobiiliapplikaatiossa käytettävistä väreistä

Käyttäjätunnuksen luominen

Sovelluksen käynnistyessä avautuu kirjautumista varten tarkoitettu ikkuna (kts. kuva 39). Kirjautumisikkunassa sijaitsee teksti: ”*Ei jäsen?*” ja sen alapuolella: ”*Rekisteröidy nyt*”-linkki, jossa käyttäjätunnuksen luominen tapahtuu vaivattomasti.

Päästyään rekisteröitymisikkunaan, käyttäjä syöttää nimensä, sähköpostinsa ja valitseman salasan. Salasan oikealla puolella sijaitsevasta silmäkuvakkeesta käyttäjä voi tarkistaa, onko syötetty salasana oikein kirjoitettu. Tämän jälkeen käyttäjä painaa ”*Rekisteröidy*”-painiketta, joka ohjaa käyttäjän kirjautuneena kotinäkömään, mikäli rekisteröityminen on suoritettu onnistuneesti.

Rekisteröitymisen onnistumiseksi sovellus tarkistaa, onko samalla sähköpostilla jo luotu käyttäjätunnusta. Mikäli rekisteröinti epäonnistuu, ilmaantuu virheikkuna, jossa käyttäjälle kerrotaan, että kyseinen sähköpostiosoite on jo liitetty toiseen olemassa olevaan käyttäjätunnukseen. Tässä vaiheessa käyttäjän on syötettävä joku muu sähköpostiosoite.



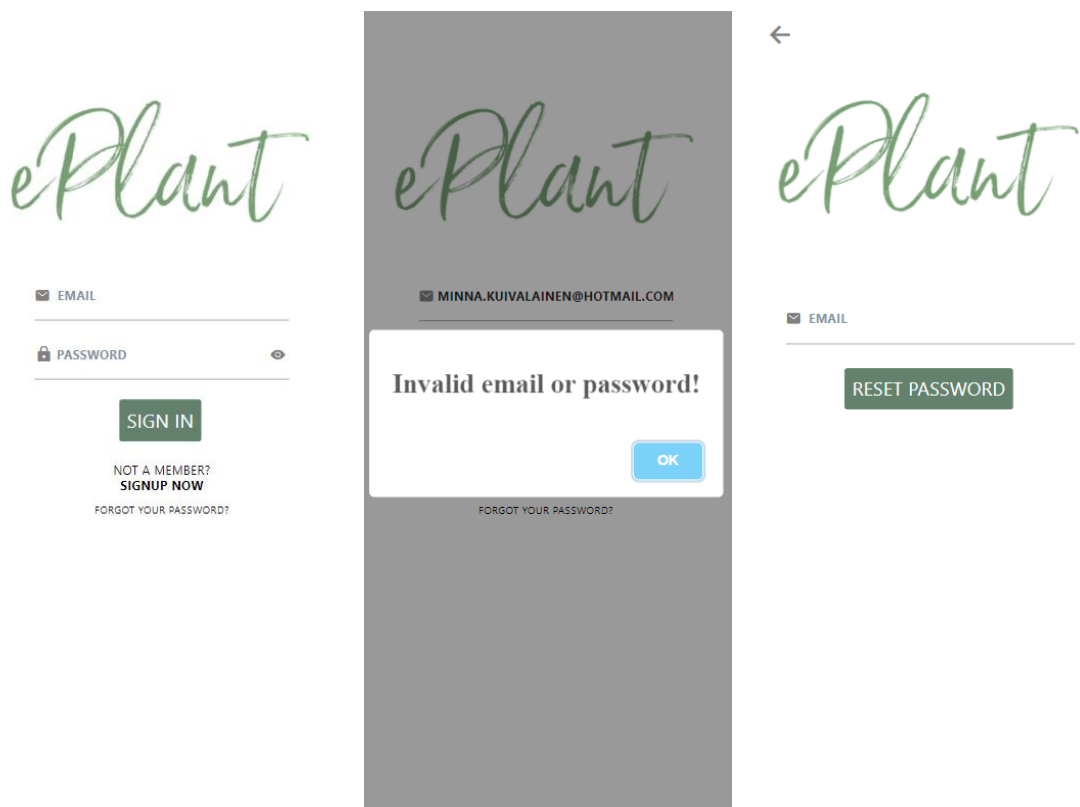
Kuva 38. Sovellusnäky – rekisteröityminen ja virheikkuna, kun käyttäjän syöttämä sähköposti on jo käytössä

Kirjautuminen

Jos käyttäjä on kirjautunut ulos, sovellus ohjaa käynnistytksen aikana automaattisesti kirjautumisikkunaan. Sovellukseen kirjautuminen tapahtuu sähköpostin ja salasanan syöttämisellä. Mikäli käyttäjätunnus on oikein, sovellus ohjaa käyttäjän kotinäkymään, jossa omien kasvien tarkastelu, lisäys ja poisto tapahtuvat. Jos käyttäjätunnus on syötetty väärin, ilmaantuu virheikkuna, joka kertoo sähköpostiosoitteen tai salasanan olevan väärä.

Unohtunut käyttäjätunnus

Jos käyttäjä on unohtanut salasansansa, on se mahdollista palauttaa siihen sähköpostiosoitteeseen, joka oli ilmoitettu rekisteröinnin yhteydessä. Kirjautumisikkunan alalaidassa on: ”*Oletko unohtanut salasiasi?*”-linkki, jossa salasanan vaihtaminen onnistuu vaivattomasti. Kun käyttäjä syöttää sähköpostiosoitteensa ja painaa ”*Vaihda salasan*”-painiketta, ilmaantuu syötettyyn sähköpostiosoitteen salasanan vaihtamiseen liittyvät ohjeet.

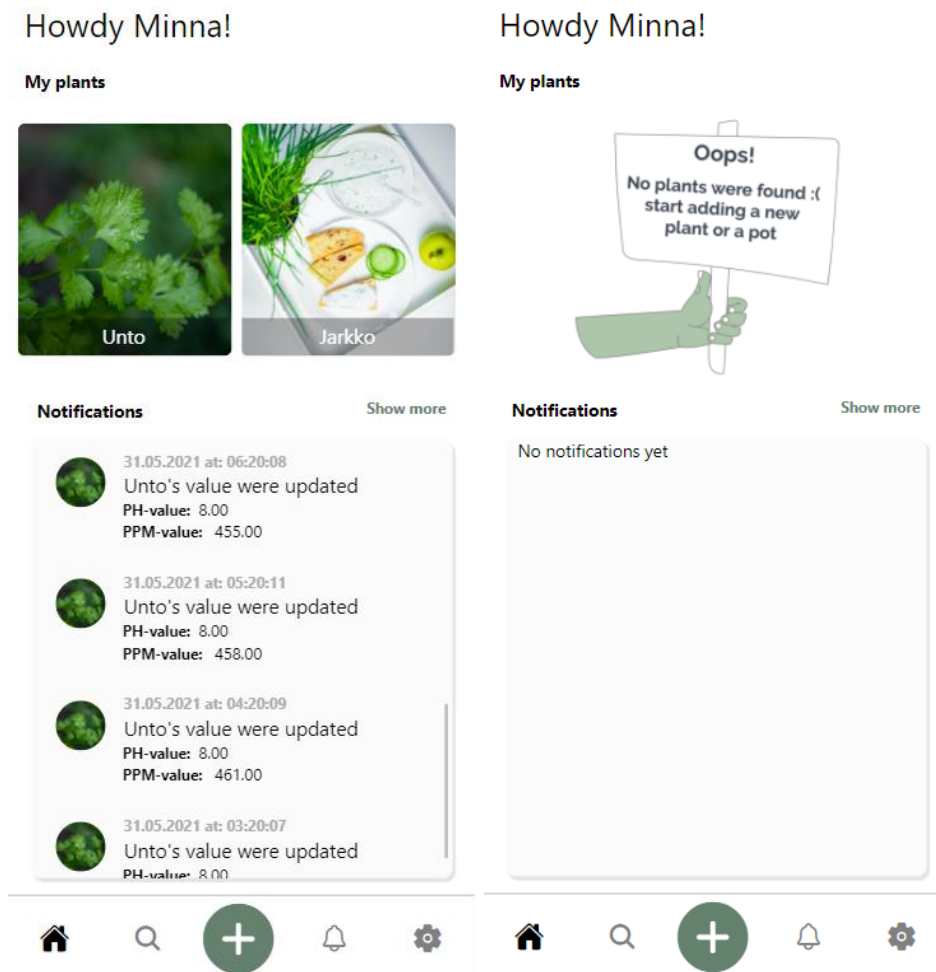


Kuva 39. Sovellusnäky – kirjautumisikkuna, virheikkuna, kun käyttäjä on syöttänyt virheellisen sähköpostin tai salasanan ja käyttäjän salasanan nollaus

Kotinäkymä

Kun kirjautuminen on suoritettu onnistuneesti, sovellus ohjaa käyttäjän kotinäytölle, johon ilmaantuu teksti: *"Hei käyttäjänimi!"*. Kotinäyttö on sovelluksen keskittymä, jossa käyttäjä voi seurata omia lisättyjä kasveja sekä anturin antamia arvoja alla olevasta ilmoitukset-listasta. Mikäli käyttäjä haluaa tarkastella arvoja tarkemmin, ilmoitusten oikealla yläkulmassa sijaitsee *"Näytä lisää"*-linkki, jota painamalla pääsee erilliseen sovellusnäkymään, jossa päivitettyt anturiarvot näkyvät kokonaisena listana. Näytön yläosassa näkyy kaikki käyttäjän lisäämät kasvit horisontaalisena listana. Jos käyttäjätunnukselle ei ole lisätty yhtäkään kasvia, sovellusnäkymään ilmestyy kuva, joka kertoo: *"Voi ei! Kasveja ei löytynyt :(aloita lisäämällä kasvi tai ruukku"*.

Käyttäjä voi alhaalla sijaitsevasta navigointipalkista siirtyä eri sovellusnäkymiin. *Mökkikuvake* ohjaa käyttäjän kotinäkymään, *suurennuslasikuvake* ohjaa käyttäjän hakusivustolle, *pluskuvake* puolestaan mahdollistaa kasvien ja ruukkujen lisäämisen, *hälytyskellokuvake* ohjaa käyttäjän ilmoitukset-sivulle ja *rataskuvakkeesta* käyttäjä voi halutessaan muokata asetuksia.

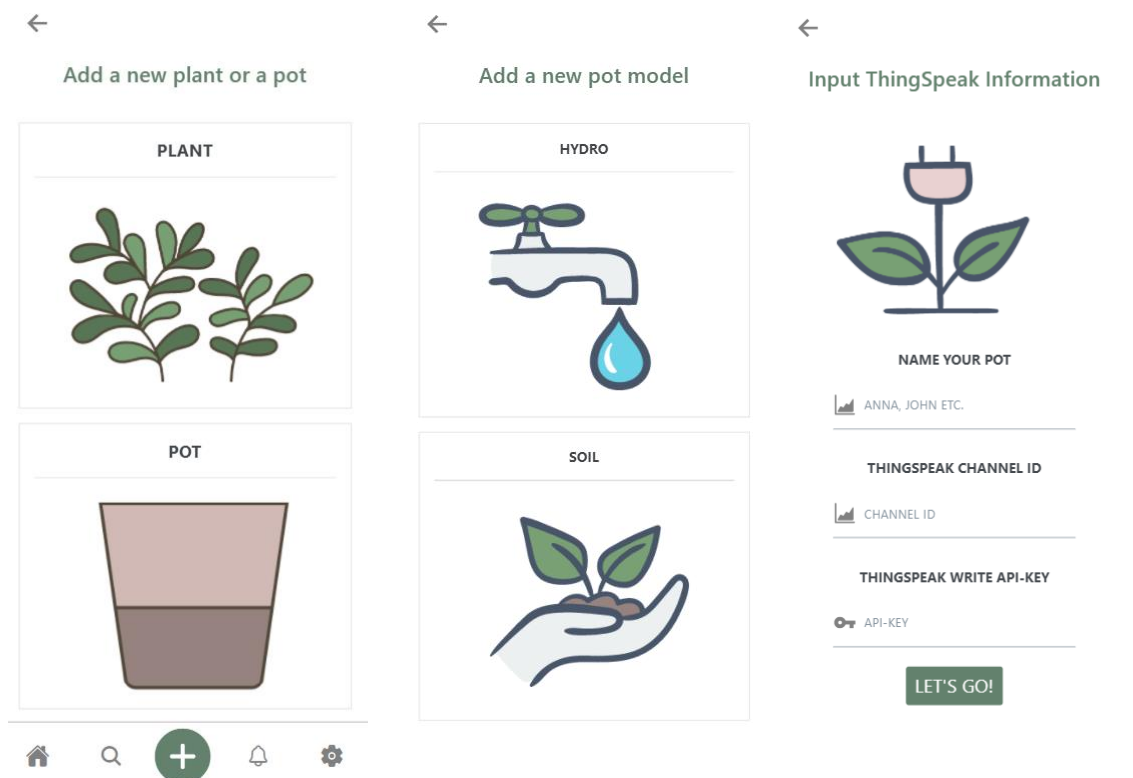


Kuva 40. Sovellusnäkymä – Kotinäkymä, kun kasveja on lisätty ja kotinäkymä ilman lisättyjä kasveja

Ruukun lisääminen

Sovelluksen käytön aloittamiseksi käyttäjän tulee lisätä ruukku haluamalleen kasville. Ruukun lisääminen tapahtuu navigaatiopalkissa sijaitsevan pluskuvakkeen kautta. Käyttäjä siirtyy näkymään, jossa valittavana joko kasvin tai ruukun lisäys. Käyttäjän painaessa ruukun kuvaa, avautuu kaksi eri ruukun mallia: vesiviljelytekniikka tai multaversio.

Riippumatta mallista, käyttäjä nimeää ruukun sekä syöttää ThingSpeakista saadun kanavan uniikin ID:n ja API-tunnukset. Kun tarvittavat tiedot ovat syötetty, painetaan "Nyt mennään!"-painiketta, jolloin käyttäjä siirtyy päänäkymään. Sovellus ilmoittaa ilmoitusikkunassa: "Uusi ePlant on lisätty!", jos ruukku on lisätty käyttäjälle onnistuneesti. Mikäli ruukun lisääminen epäonnistuu, käyttäjälle ponnahtaa ilmoitus: "Sinulla ei voi olla kahta ePlant-ruukkua samalla kanavan ID:llä ja API-avaimella!". Käyttäjän on tässä tapauksessa syötettävä tiedot uudelleen sekä valittava toinen kanava ja API-avain uuden ruukun lisäämiseksi. Sovelluksessa ei ole mahdollista yhdistää kahta ruukkua samalle ThingSpeak-kanavalle, mikä helpottaa käyttäjää hahmottamaan, onko kyseisillä tiedoilla jo yhdistetty toinen ePlant. Tämä ehkäisee käyttäjää tekemästä peruuttamattomia muutoksia, joka voisi aiheuttaa muun muassa anturiarvojen sekoittumisen keskenään.



Kuva 41. Sovellusnäky – Ruukun tai kasvin lisääminen, kasvatustekniikan valitseminen ja ThingSpeak-tietojen lisääminen

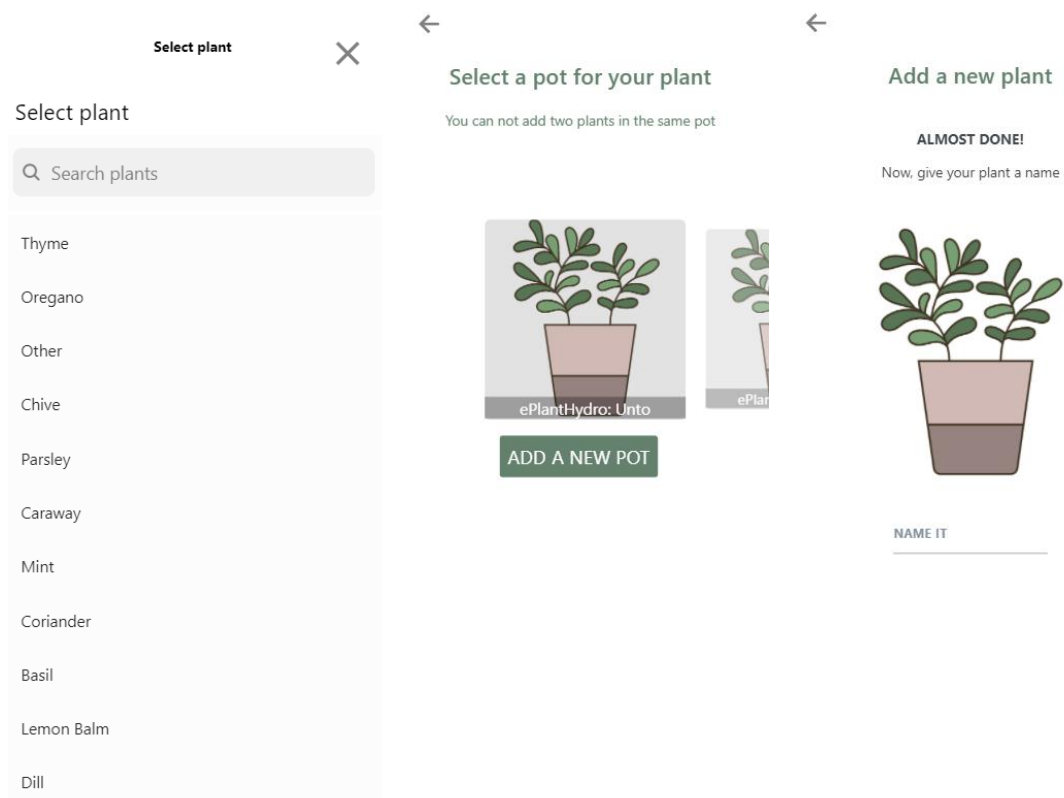
Kasvin lisääminen

Sovelluksessa kasvin lisääminen tapahtuu navigaatiopalkissa sijaitsevan pluskuvakkeen kautta. Käyttäjä siirtyy näkymään, jossa valittavana joko kasvin tai ruukun lisäys.

Käyttäjän painaessa kasvin kuvaa, avautuu lista sovelluksen tietokantaan tallennetuista yrteistä. Mikäli käyttäjä ei löydä sopivaa vaihtoehtoa, listalta löytyy kohta "Muu", joka ei määrittele lisätyn kasvin lajia. Tämä on tärkeä ominaisuus siinä vaiheessa, kun pH- ja ppm-arvot optimoidaan tietyn kasvin kasvuolosuhteiden mukaan.

Hakutoiminnon avulla käyttäjä pystyy etsimään listalta tiettyä yrttiä. Tämä osoittautuu tärkeäksi ominaisuudeksi, kun tietokantaan on tallennettu useita eri vaihtoehtoja ja lista kasvaa pitkäksi.

Kun käyttäjä on valinnut kasvin, siirrytään näkymään, jossa valitaan sille haluttu ruukku. Mikäli käyttäjällä ei ole yhtään ruukkua lisättynä, voidaan tässä kohtaa lisätä uusi ruukku "Lisää uusi ruukku"-painikkeen kautta. Kun käyttäjä on valinnut sopivan ruukun, annetaan kasville vielä nimi ja prosessi on valmis. Mikäli kasvi on lisätty onnistuneesti, käyttäjä ohjataan kotinäkymään, johon ponnahtaa ilmoitusikkuna: "Uusi Unto on lisätty!".

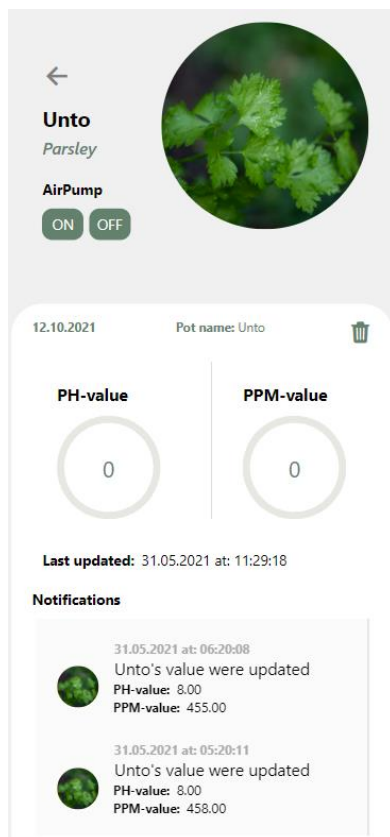


Kuva 42. Sovellusnäkö – Kasvilajin valinta, ruukun valitseminen ja nimeäminen

Oman kasvin tiedot

Kun käyttäjä haluaa tarkastella lisätyn kasvin tietoja, painetaan päänäkylässä halutun kasvin kuvaa. Kuva ohjaa kyseisen kasvin omalle sivulle, josta käyttäjä voi seurata anturin antamia arvoja tarkemmin. Sovellusnäkyvän yläosassa näkyy kasvin nimi sekä kasvilaji kuvineen. Kasvilajin alapuolella sijaitsee ilmapumpun manuaalinen toiminto, josta käyttäjä voi hallinnoida pumpun laittamista päälle ja pois. Sovellusnäkyvän keskiosassa sijaitsee päivämäärä, milloin kasvi on lisätty käyttäjän tietokantaan, ruukun nimi sekä roskakorikuvake, jota painamalla kyseinen kasvi voidaan poistaa ruukusta. Kun käyttäjä painaa roskakorikuvaketta, ponnahtaa ilmoitusikkuna: *"Poista kasvi, oletko varma?"*. Kun käyttäjä painaa "OK"-painiketta, ilmestyy toinen ilmoitusikkuna: *"Puff! Kasvi poistettu!"*, joka kertoo kasvin poiston onnistuneen odotetun mukaisesti.

Alapuolella sijaitsee kasvin sen hetkiset pH- ja ppm-arvot sekä päivämäärä, milloin viimeisimmät tiedot ovat haettu ThingSpeakista. Anturiarvojen alapuolelta löytyy ilmoitukset-lista, josta löytyy kymmenen viimeisintä arvoa. Mikäli käyttäjä haluaa tarkastella siitä vanhempia arvoja, löytyvät ne ilmoitukset-sovellusnäkyvästä. Jos käyttäjä on syöttänyt ThingSpeakin kanavan tiedot väärin, sovellus ilmoittaa: *"Virhe! Jokin meni vikaan kasvin tietoja ladattaessa. Tarkista, että kanavan ID on oikein!"*. Käyttäjä voi muokata ThingSpeakin kanavatietoja jälkikäteen käyttäjäasetuksista.



Kuva 43. Sovellusnäkyvä – Oman kasvin tiedot

Hakunäkymä

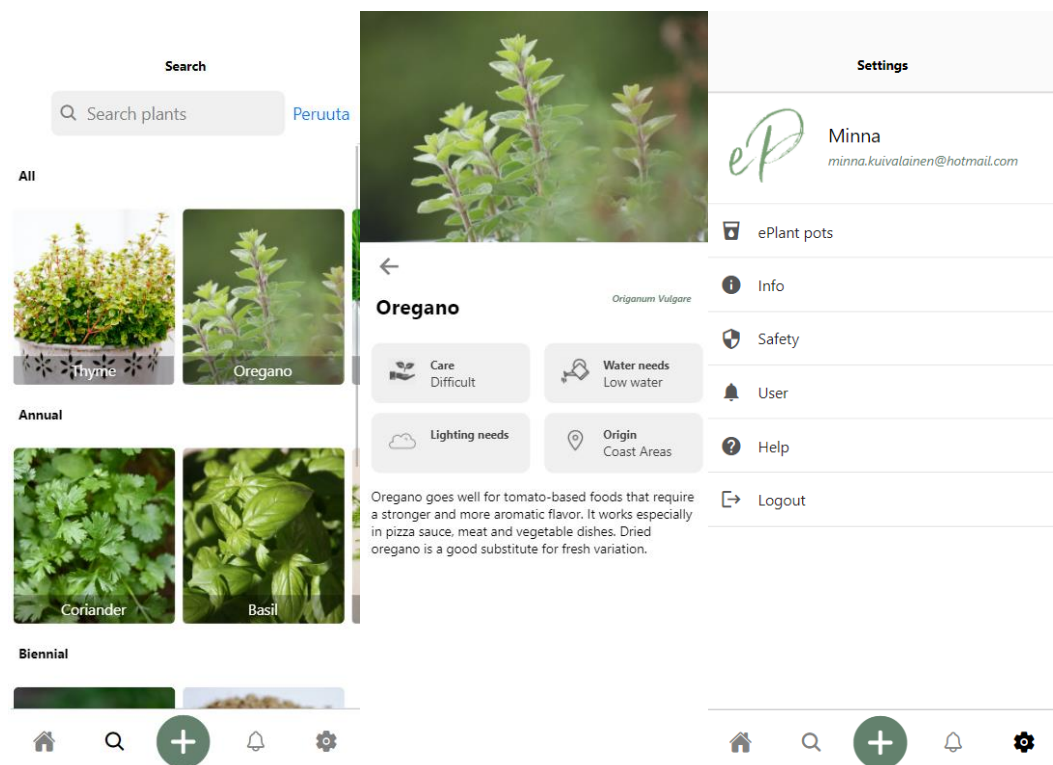
Sovelluksen hakunäkymästä käyttäjä voi tarkastella tietokannasta löytyviä kasveja horisontaalisesti. Kasvit ovat jaettu eri kolmeen eri kasvukategoriaan (*jokavuotinen*, *kaksivuotinen* ja *monivuotinen*), jotta niiden löytäminen olisi mahdollisimman helppoa. Käyttäjä voi halutessaan myös selata listaa, josta löytyvät kaikki tietokantaan lisätyt kasvit. Hakutoiminnon avulla voidaan hakea kasvia suoraan nimen perusteella, joka säästää huomattavasti aikaa sekä vaivaa. Painamalla halutun kasvin kuvaa tai nimeä, käyttäjä siirtyy sovellusnäkymään, josta voidaan tarkastella yksittäisen kasvin tietoja tarkemmin.

Yksittäisen kasvin tiedot

Käyttäjän siirryttyä yksittäisen kasvin sovellusnäkymään, avautuu tarkempia tietoja kasvutarpeista, helppohoitoisuudesta, veden- ja valontarpeesta, alkuperästä sekä yleistietoa muun muassa siitä, millaiseen ruokaan kyseinen yrtti soveltuisi parhaiten.

Asetukset

Rataskuvaketta painamalla käyttäjä siirtyy asetuksiin, josta voidaan hallinnoida lisättyjä ruukkuja, hakea tietoa sovelluksen toiminnasta, tarkastella tietoturvallisuuden liittyviä asioita, muokata käyttäjätietoja, etsiä tietoa sovelluksen käyttöönottamisesta tai kirjautua ulos. Asetuksissa näkyy myös, mihin sähköpostiosoitteeseen käyttäjätunnus on liitetty.



Kuva 44. Sovellusnäkymä – Tietokannan hakutoiminto, yksittäisen kasvin tiedot ja käyttäjäasetukset

Ruukut

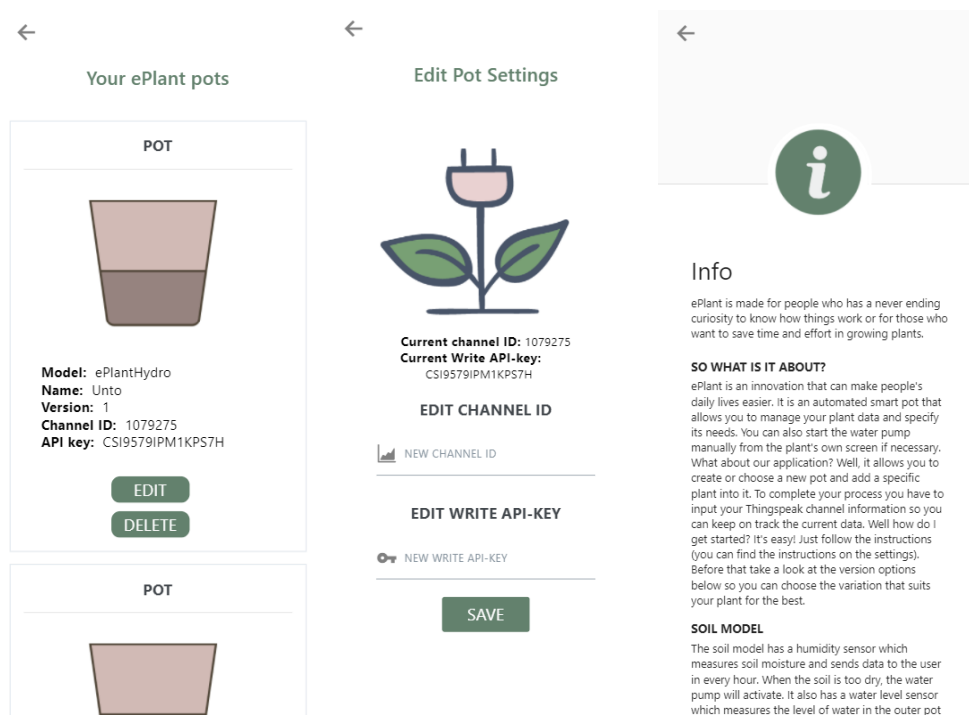
Kun käyttäjä haluaa tarkastella lisäämiään ruukkuja tai muokata esimerkiksi ThingSpeak-kanavan ID:tä tai API-avainta, painetaan rataskuvaketta, joka ohjaa asetukset-sovellusnäkymään. Asetusten ensimmäisenä linkkinä löytyy ”ePlant-ruuku”, joka ohjaa käyttäjän hallinnointisivulle. Hallinnointisivulla on listattu kaikki käyttäjän tietokantaan lisäämät ruukut, jonka kautta tietojen muokkaaminen ja ruukun poistaminen tapahtuu vaivattomasti. Sovellukseen ponnahtaa ilmoitusikkuna, jossa varmistetaan, haluaako käyttäjä varmasti poistaa kyseisen ruukun tietokannasta. Käyttäjän painaessa ”Tee se!”-painiketta, ilmestyy teksti ”ePlant on poistettu!”, kun ruuku sekä kasvi ovat poistettu tietokannasta onnistuneesti.

Muokkaa ruukkua

Kun käyttäjä haluaa muokata yksittäisen ruukun tietoja, painetaan ”Muokkaa”-painiketta, joka ohjaa uuteen sovellusnäkymään. Ruukun muokkaaminen tapahtuu uuden ThingSpeak-kanavan ID:n ja API-avaimen syöttämisellä. Uudet tiedot tallennetaan painamalla ”Tallenna”-painiketta. Sovellus ilmoittaa myös nykyisen kanavan ID:n ja API-avaimen tiedot, jotta käyttäjä voi tarkistaa ovatko tiedot oikein vai väärin.

Info

Informaatio-sovellusnäkymästä löytyy tietoa älyruukun toiminnallisuudesta sekä sovelluksen kasvatustekniikoista. Tämä tietopaketti helpottaa käyttäjää valitsemaan oikean kasvatusmenetelmän oman kasvin käyttötarpeita ajatellen.



Kuva 45. Sovellusnäkö – Käyttäjän ruukut, ruukun muokkaaminen ja tekniset tiedot

Tietosuojakäytäntö

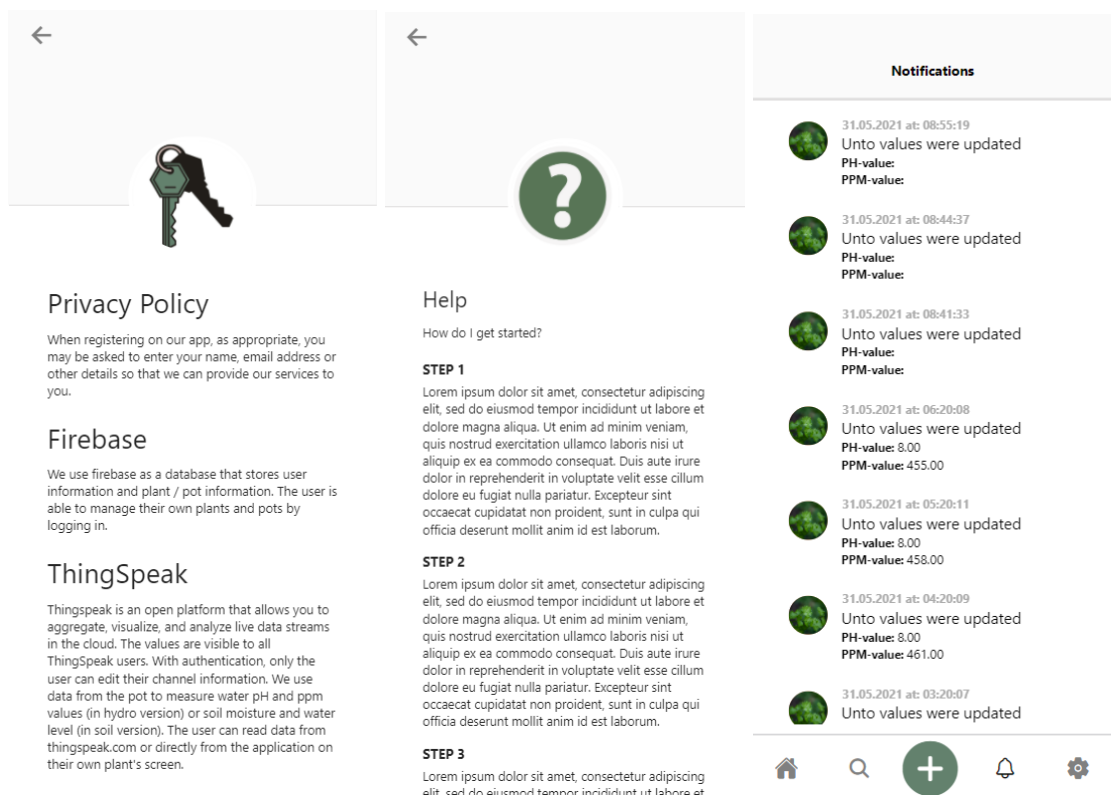
Käyttäjä pystyy tarkastelemaan sovelluksen tietosuojakäytäntöjä koskevia asioita turvallisuus-sovellusnäkyssä. Turvallisuutta koskevissa asioissa kerrotaan, millaisia tietoja tarvitaan, jotta käyttäjä pystyy hyödyntämään sovelluksen tarjoamia palveluja. Tekstissä on mainittu erikseen, mitä tietoja Firebase tarvitsee, jotta käyttäjä voisi aloittaa kasvien sekä ruukkujen lisäämisen oman käyttäjätunnuksensa alle. ThingSpeakin osalta on mainittu käyttäjän lisäämien kanavien olevan julkisia, mutta autentikoinnin ansiosta vain oikeudet omaava käyttäjä voi muokata tietoja omalla alustallaan.

Apua

Käyttäjälle on luotu kattavat, vaiheittain etenevät ohjeet älyruukun osalta, jotta käyttöönotto olisi mahdollisimman helppoa ja vaivatonta. Ohjeita seuraamalla käyttäjä saa kokonaisvaltaisen kuvan sovellusta koskevista toiminnallisuuksista.

Ilmoitukset

Käyttäjä pystyy halutessaan tarkastelemaan lisäämiensä kasvien anturilukemia yksityiskohtaisemmin ilmoitukset-sovellusnäkyästä. Listalta löytyy myös jokaisen kasvin vanhimmat historiatiedot, jotka eivät näy oman kasvin näkyssä. Mikäli käyttäjällä ei ole lisättyä ruukkuja tietokantaan, sovellus ilmoittaa: *"Ei lisättyjä ePlant-ruukkuja! Lisää uusi kasvi nähdäksesi ilmoitukset!"*. Ilmoitukset ovat lajiteltu uusimmasta vanhimpaan.



Kuva 46. Sovellusnäky – Yleiset tiedot sovelluksesta ja kasvikohtaiset ilmoitukset

Käyttäjä

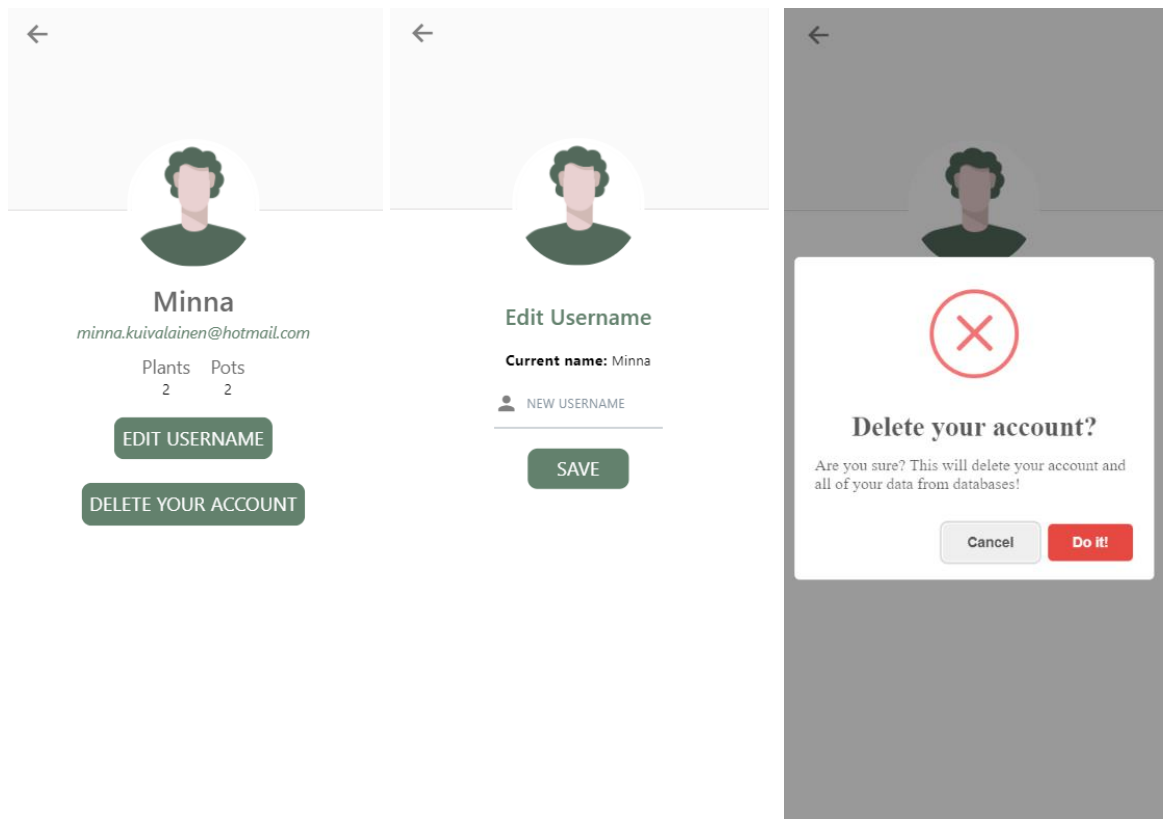
Käyttäjä voi halutessaan tarkastella omia tietojaan, muokata käyttäjänimeä tai poistaa kaikki tietonsa sovelluksen tietokannasta käyttäjä-sovellusnäkymän kautta. Tiedoissa näkyy käyttäjän nimi, sähköpostiosoite sekä kuinka monta kasvia ja ruukkua kyseisen käyttäjätunnuksen alle on lisätty.

Muokkaa nimimerkkiä

Käyttäjän on mahdollista vaihtaa käyttäjänimensä painamalla ”*Muokkaa käyttäjänimeä*”-painikkeesta. Käyttäjä siirtyy uuteen sovellusnäkymään, jossa näkyy käyttäjän nykyinen nimi. Käyttäjänimen muutokset viimeistellään painamalla ”*Tallenna*”-painiketta. Sovellus ilmoittaa: ”*Käyttäjä on päivitetty!*” ja siirtyy tallennuksen jälkeen päänäkymään.

Poista käyttäjä

Mikäli käyttäjä halua poistaa kaikki tiedot sovelluksen tietokannasta, painetaan ”*Poista käyttäjätunnus*”-painiketta, jolloin ponnahtaa ilmoitusikkuna: ”*Poista käyttäjätunnus? Oletko varma? Tämä poistaa tunnuksesi ja kaiken datan tietokannasta!*”. Käyttäjä voi vielä tässä vaiheessa perua komennon painamalla ”*Peruuta*”-painiketta. Poistettua käyttäjätunnusta tai siihen liitettyjä kasveja ja ruukkuja ei voi saada enää takaisin, kun se on jo kerran poistettu. Sovellus ohjaa käyttäjän kirjautumisnäkymään poiston jälkeen.



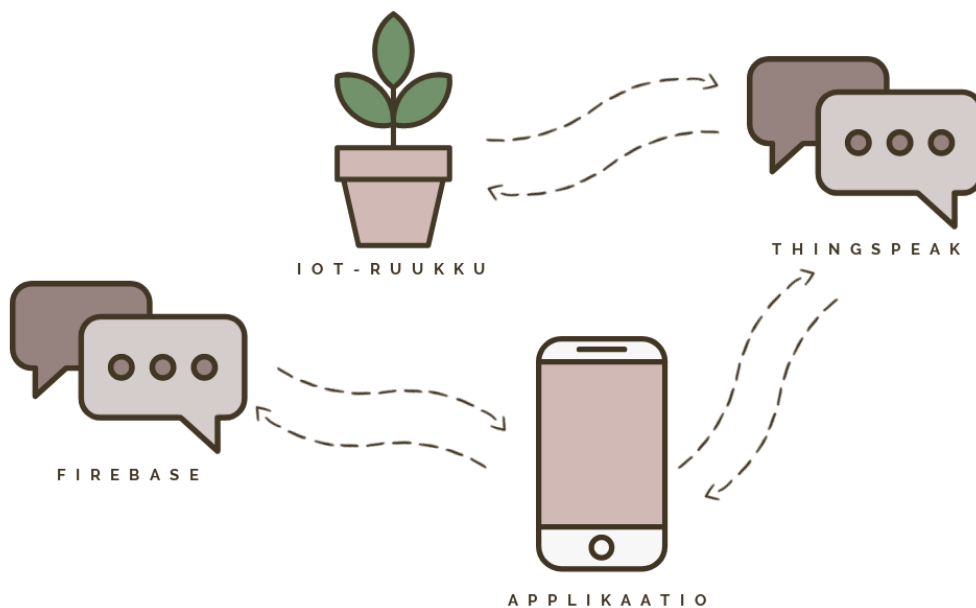
Kuva 47. Sovellusnäkymä – käyttäjän profiili, käyttäjätietojen muokkaaminen ja poistaminen

4.3 Integraatio

Järjestelmäintegraatio on pähkinäkuoressa sovellusten, tietojen ja laitteiden yhdistämistä yhdeksi tehokkaaksi kokonaisuudeksi. Yksityiskohtaisen integraation tarve on perustavanlaatuinen liiketoiminnan kehittämisessä markkinoiden jatkuvassa muutosvirrassa. Uuden järjestelmän integrointi tuo merkittävää lisäarvoa, joka mahdollistaa haluttujen toiminnallisuuksien ja ominaisuuksien yhdistämisen. Integraation avulla voidaan parantaa palvelutehokkuutta, automatisoida manuaalisia työvaiheita kustannustehokkaasti, vähentää tietovirheiden määrää ja yksinkertaistaa ohjelmistoarkkitehtuurin rakenteita. (Haglund 9.1.2018).

Integraatiorakenne

Integraatio fyysisen älyruukun ja mobiiliapplikaation välillä tapahtuu ThingSpeak-analytiikka-alustan ja Google Firebase-tietokannan avulla. Alla oleva luurankomalli havainnollistaa lukijaa hahmottamaan integraation kokonaisrakenteen ja datan kulkusuunnan eri teknologioiden välillä.

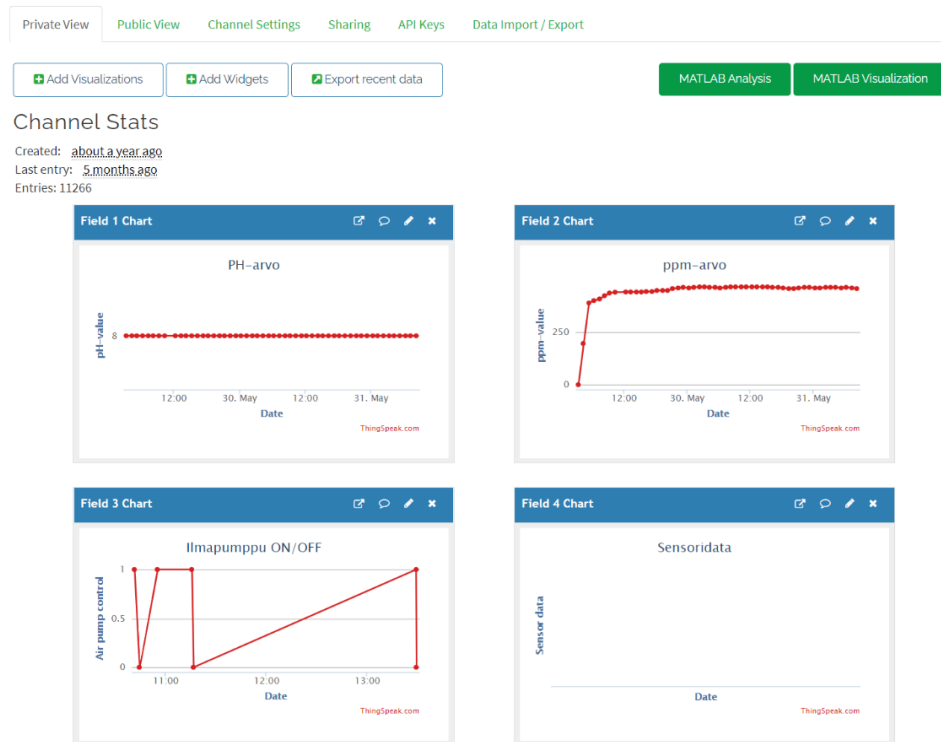


Kuva 48. Integraation luurankomalli

ThingSpeak

Älyruukussa käytettävä ThingSpeak on TheMathWorksin suunnittelema IoT-analytiikka-alusta, jonka avulla voidaan muun muassa lähettää laitteen anturitietoja pilveen ja visualisoida saatua dataa reaaliaikaisesti. ThingSpeakia hyödynnetään useimmiten erinäisten prototyyppien luomiseen sekä analyttisten IoT-järjestelmien testaamiseen. Helppokäyttöisen hallintapaneelin kautta käyttäjä pystyy analysoimaan ja tulkitsemaan antureista saatua dataa selkeämmin. (TheMathWorks, Inc. 2021.) ThingSpeak tarjoaa

käyttäjälleen kanavia, jonne haluttu data voidaan tallentaa ja visualisoida. Kanavien avulla voidaan seurata anturiarvoja päivän sekä kellonajan mukaan, joka auttaa käyttäjää reagoimaan, mikäli saadut arvot eivät ole odotusten mukaisia. Kanavat ovat oletuksena julkisia, mutta käyttäjä voi halutessaan määritellä kanavan yksityiseksi, jolloin hallinnointi tapahtuu yksilöllisen generoidun API-avaimen avulla. API-avaimen ansiosta käyttäjä voi halutessaan lukea tai päivittää kanavansa tietoja.



Kuva 49. ThingSpeakin hallintapaneeli

Älyruukulle rakennetussa mobiiliapplikaatiossa ThingSpeakia käytetään niin kutsuttuna välityspalvelimenä. Mobiiliapplikaatio hakee ja lähettää dataa ThingSpeakin analytiikka-alustalle, jonka ansiosta käyttäjä pystyy seuraamaan yksittäisen kasvin anturitietoja myös omalta puhelimeltaan. ThingSpeak mahdollistaa ilmapumpun hallinnoinnin, joka on ruukun yksi tärkeimpiä ominaisuuksia. Käyttäjä pystyy seuraamaan ilmapumpun päälle ja pois -toimintoa omasta hallintapaneelistaan. Kun käyttäjä painaa mobiiliapplikaatiossa sijaitsevan ilmapumpun ON/OFF-painiketta, ThingSpeakin hallintapaneelin kanavakenttään ilmestyy 1/0, joka kertoo pumpun reagoineen kutsuun onnistuneesti.

```

{
  - channel: {
    id: 1079275,
    name: "ePlant-oppari",
    description: "School-project",
    latitude: "0.0",
    longitude: "0.0",
    field1: "pH-value",
    field2: "ppm-value",
    field3: "Air pump control",
    field4: "Sensor data",
    created_at: "2020-06-10T14:13:16Z",
    updated_at: "2021-11-11T13:16:05Z",
    last_entry_id: 11266
  },
  - feeds: [
    - {
      created_at: "2021-05-31T11:29:03Z",
      entry_id: 11265,
      field1: null,
      field2: null,
      field3: "1",
      field4: null
    },
    - {
      created_at: "2021-05-31T11:29:18Z",
      entry_id: 11266,
      field1: null,
      field2: null,
      field3: "0",
      field4: null
    }
  ]
}

```

Kuva 50. ThingSpeak-kanavan ja anturien data JSON-muodossa

Alla olevassa kuvakaappauksessa näkyy mobiiliapplikaatiossa luotu funktio, joka hallinnoi ilmapumpun toiminnallisuutta ThingSpeakin kautta.

```

const AirPumpON = (write_apikey, fieldNumber) => {
  const url = "https://api.thingspeak.com/update?api_key=" + write_apikey + "&field" + fieldNumber + "=1";
  fetch(url)
    .then((response) => response.json())
    .then((responseJson) => {
      while (true) {
        if (parseInt(responseJson) == 0) {
          AirPumpON(write_apikey, fieldNumber)
          // console.log(responseJson);
        }

        swal(
          {
            title: "AirPump is turning on!",
            button: false,
            timer: 8000
          }
        );

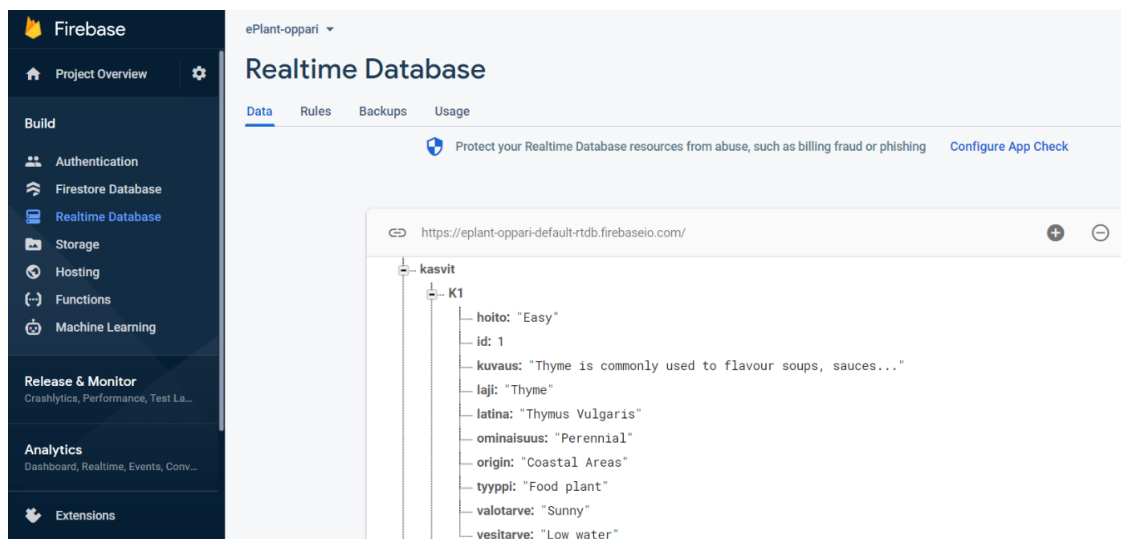
        break;
      }
    });
  .catch((error) => {
    swal(
      {
        title: "Something went wrong!",
        button: true,
      }
    );
  });
}

```

Kuva 51. Koodirakenne – Ilmapumpun hallinnointi ThingSpeakin kautta

Firestore

Älyruukussa käytettävä Firebase Realtime Database on Googlen kehittämä pilvipalvelussa toimiva NoSQL-tietokanta, jota käytetään pääsääntöisesti verkkosovelluksien, Androidin ja iOSin kehittämiseen. Firebase tarjoaa toimivan reaaliaikaisen tietokannan, käyttäjähallinnan sekä palvelintilaa, jotka helpottavat huomattavasti sovelluskehittäjien työtaakkaa. Käyttäjän tiedot tallentuvat tietokantaan JSON-muodossa, joka mahdollistaa tietojen ja objektien välisen esteettömyyden. Reaaliaikaisen synkronoinnin ansiosta käyttäjät pääsevät käsiksi tietoihinsa mutkattomasti laitteesta riippumatta. Firebase säilyttää käyttäjän tallentamia tietoja levyllä, joka mahdollistaa tietokannan käytön myös offline-tilassa. Kun yhteys on muodostettu uudelleen, käyttäjä vastaanottaa tulleet muutokset omalle alustalleen. Helppokäyttöisen hallintapaneelin ansiosta käyttäjältä ei vaadita sen suurempaa ohjelmoinnin osaamista. (Javatpoint s.a.)



Kuva 52. Firebasen hallintapaneeli

Firebasen reaaliaikaista tietokantaa on hyödynnetty älyruukulle rakennetussa mobiiliapplikaatiossa tallennustilana sekä käyttäjähallinnassa. Firebasen hallintapaneelin kautta voidaan muokata tai poistaa sovelluksen kautta tallennettuja käyttäjätietoja sekä hallinnoida käyttäjälle lisättyjä ruukkuja ja kasveja. Tietokantaan on tallennettu valmiiksi yleisimpiä yrttejä ja hoito-ohjeita kasvuolosuhteiden optimoimisen helpottamiseksi. Käyttäjä pystyy selailemaan olemassa olevia kasveja mobiiliapplikaation kautta, joka helpottaa yksilöimään ruukussa sijaitsevia kasveja.

```

import firebase from 'firebase/app';
import 'firebase/auth';
import 'firebase/database';

// initializing firebase object by giving database and project reference
var firebaseConfig = {
  apiKey: "xx",
  authDomain: "xx",
  databaseURL: "xx",
  projectId: "xx",
  storageBucket: "xx",
  messagingSenderId: "xx",
  appId: "xx",
};

firebase.initializeApp(firebaseConfig);

export default !firebase.apps.length
  ? firebase.initializeApp(config)
  : firebase.app();

```

Kuva 51. Koodirakenne – Firebasen konfigurointi mobiiliapplikaatioon

Alla olevassa kuvakaappauksessa näkyy mobiiliapplikaatiossa luotu funktio, joka lisää käyttäjän kasvin tarvittavine parametreineen. Tiedot tallentuvat Firebasen tietokantaan.

```

function AddPlantToUser(userId, species, plantName, ePlant, navigate) {

  //console.log(ePlant)
  firebase.database().ref('users/' + userId + "/myPlants").push(
    {
      species: species,
      plantName: plantName,
      date: Date(),
      ePlantPot: {
        ePlantID: ePlant[0],
        ePlantName: ePlant[1].ePlantName,
        channel_id: ePlant[1].channel_id,
        write_apikey: ePlant[1].write_apikey,
        ePlantModel: ePlant[1].ePlantModel
      }
    }
  )
  navigate('Home', {showSnackBar: true, plantName: "New " + plantName + " was added!"})
}

```

Kuva 52. Koodirakenne – Kasvin lisääminen Firebase-tietokantaan

4.4 Jatkokehitys

Visuaalinen ilme

Lopputuloksena muodostui käyttäjäystävällinen, grafiikaltaan viimeistelty ja esteettinen käyttöliittymä, joka palvelee kokonaisvaltaisesti tarkoitustaan. Vaikka sovellus on visuaaliselta ilmeeltään viimeistelty, löytyy siitä aina jotakin kehitettävää. Komponenttien ja kuvien päivittämisellä on mahdollista hioa sovelluksen ulkonäköä entistäkin enemmän. Hillityllä efektien lisäämisellä voidaan tehdä komponenteista vieläkin näyttävämpiä ja massasta erottuvampia. Jatkokehittämisen avulla voidaan tarjota sovelluksen käyttäjälle miellyttävämpää käyttäjäkokemusta ja korjata mahdollisia esteettisiä virheitä.

Toiminnallisuus

Sovelluksen tehokkuus ja ominaisuudet ovat hiottu sovelluksen osalta, mutta näiltäkin osin parannettavaa varmasti löytyisi. Sovelluksessa on tällä hetkellä käytössä kasvatustekniikkana multa- ja vesiviljely. Molemmista on tehty yksi versio, joka palvelee tällä hetkellä opinnäytetyön tarkoitusta. Mikäli toiminnallisuutta halutaan laajentaa ja tehdä ruukkuja useampi eri versio, on sovellusta päivitettävä. Sovelluksen tehokkuutta on mahdollista parantaa koodin jäsentelyllä ja tehokkaampien komponenttien käyttöönottamisella. Selkeämpi ja tehokkaampi koodi tekee niin ohjelmoijalle kuin sovelluksen käyttäjällekin miellyttävämmän sekä luotettavamman tuntuksen kokemuksen. Laadukkaat ja aktiivisesti päivittyvät komponentit mahdollistavat omalta osaltaan sovelluksen vakaan toiminnan minimoiden kaatumisriskejä.

Jokaisella kasvilla on ihanteelliset pH- ja ppm-arvot, jotta kasvuolosuhteet olisivat mahdollisimman optimaaliset. Nykyisessä sovelluksessa ei ole kuitenkaan määritelty erikseen kasvien ominaisuuksia, joiden mukaan kasvutarpeet voitaisiin huomioida lajikohtaisesti. Jatkokehitysideana olisi määritellä kasveille uniikit arvot, jotta edellä mainittu asia olisi mahdollista. Tietokantaan lisätystä kasvivalikoimasta olisi mahdollista myös tehdä kattavampi, sillä sovellus sisältää tällä hetkellä vain yleisimmät ja käytetyimmät yrtit.

ThingSpeakin nopeutta sekä käyttäjäystävällisyyttä on mahdollista parantaa ostamalla maksullisen version, jonka ansiosta API-pyyntöjä ei ole rajattu ajallisesti ollenkaan. Käyttämässämme ilmaisessa versiossa pyyntöjä voidaan laittaa yhden kerran puolella minuutissa, joka aiheuttaa sovelluksen käytössä pienen viiveen. Maksullisen version ansiosta käyttäjä voi lähettää API-pyyntöjä joka sekunti ilman jatkuvaa odottelua.

5 Pohdinta

5.1 Johtopäätökset

Opinnäytetyön tarkoituksena oli toteuttaa kunnianhimoisen idean ja kattavan suunnittelun kautta valmis prototyyppi automatisoidusta kastelujärjestelmästä, joka palvelisi niin harrastelijoita kuin pikkutarkkuutta vaativia kasvi-intoilijoitakin hallitsemaan omia kasvejaan kokonaisvaltaisesti. Tietoperustassa johdattelimme lukijan Internetin historiasta nykypäivään, esineiden internetin syntyyn, kehityskaareen ja sen eri käyttötarkoituksiin sekä lopuksi käsitelimme empiirisessä osuudessa käytettyjä teknologioita.

Lähtötilanteena meillä oli suhteellisen pitkälle viety mobiiliapplikaation rakenne, joka vaati graafisen ulkonäön sekä toiminnallisuuksien osalta kattavia muutoksia halutun lopputuloksen saavuttamiseksi. Olimme jo työstäneet multaruukkua sekä siihen integroitua mobiiliapplikaatiota ohjelmointikurssin aikana, mutta kokonaisuus jäi kesken aikataulujen vuoksi. Halusimme saada aloitetun työn päätökseen ja päätimme jatkaa sitä opinnäytetyön muodossa.

Pohdimme suunnitteluvaiheessa, mitä halusimme poistaa keskeneräisestä mobiiliapplikaatiosta ja mitä toiminnallisuuksia tulisi puolestaan niiden tilalle. Mobiiliapplikaatioon tuli uutena ominaisuutena käyttäjähallinta, jonka ansiosta ruukkuja sekä kasveja pystyttiin hallinnoimaan käyttäjäkohtaisesti. Mobiiliapplikaatioon tuli runsaasti uusia toiminnallisuuksia ja käyttöliittymä koki uuden visuaalisen ilmeen. Päivitimme tietokannat, tekstit ja kuvat perusteellisesti sekä lisäsimme vesiviljelyn toiseksi kasvatustekniikaksi multaversion rinnalle. Ruukun uuden kasvatustekniikan myötä lisäsimme toiminnallisuuksia, joiden avulla pystyttiin mittaamaan veden pH- ja ppm-arvoja sekä hallinnoimaan ilmapumpun toiminnallisuutta kasvikohtaisesti.

Kohtasimme fyysisen ruukun rakentamisessa ongelmia, jotka hidastivat etenemisprosessia. Koska koulu ei pystynyt tarjoamaan kaikkia komponentteja mitä olisimme tarvinneet ruukun rakentamiseen, lähdimme tilaamaan puuttuvat komponentit ulkomailta. Komponenttien toimituksessa kesti noin kuukausi, jonka aikana rakensimme mobiiliapplikaatiota eteenpäin. Kun tarvittavat anturit olivat lopulta saapuneet, haimme tietoa siitä, miten anturit ylipäänsä toimivat ja kuinka rakentaa toimiva kokonaisuus ESP32:ella. Seuraavana ongelmaksi muodostui jännitteiden kalibrointi, sillä prototyyppiin hankitut komponentit olivat budjettihintaisia ja siksi myös epätarkkoja suhteessa laadukkaampiin osiin. Eksimme ratkaisun ongelmaan jännitteensäätimestä, jonka avulla muunsimme 12 voltin haluttuun 5.5 volttiin. Testausvaiheessa anturi tuotti kuitenkin vain

oletusarvoja. Luulimme, että kalibroinnissa oli sattunut virhe ja siksi arvot eivät olleet todelliset. Syyksi paljastui kuitenkin se, ettei anturi ollut ”tottunut” veteen. Tunnin mittailujen jälkeen anturit tuottivat lopulta haluttuja arvoja ja pääsimme taas askeleen eteenpäin.

Suunnittelimme fyysisen ruukun osalta kaiken alusta loppuun. Ruukun ja sen osien mallintaminen sekä tulostaminen vaati paljon tutkimustyötä niin ohjelmiston, laitteiden kuin vesitiiviyydenkin osalta. Integrointi ja eri teknologioiden hyödyntäminen olivat opinnäytetyön osalta erittäin opettavaisia vaiheita valmiin prototyypin loppuunsaattamiseksi.

Perustavanlaatuinen suunnittelutyö mahdollisti onnistuneen lopputuloksen toteutumisen asetettujen aikaraamien sisällä. Opinnäytetyön empiirisessä osuudessa valmistui fyysinen ruukku ja mobiiliapplikaatio, jotka kommunikoivat keskenään eri teknologioiden avulla.

5.2 Oman oppimisen arviointi

Opinnäytetyö kasvatti oppimiskokemustamme monella eri tapaa. Alla olevassa taulukossa olemme luetelleet ne osa-alueet, joissa kehitys on ollut suurinta:

Aihealue	Mitä opimme?
Suunnittelutyö	Arkkitehtuurirakenne
3D-mallinnus	Blender-ohjelman hallinta, mallinnustyö ja tiedostotyyppien muuntaminen
3D-tulostus	Tulostimien hallinta ja ruukun osien tulostus
Ohjelmointi	JavaScript ja C++ sekä React Nativen ja Arduinon eri kirjastojen hyödyntäminen
Integrointi	Eri teknologioiden yhdistäminen yhdeksi kokonaisuudeksi
Kirjallinen osuus	Tiedonhaku, lähdekriittisyys, kielioppi, terminologia, tekstirakenteet ja asiatekstin kirjoittaminen
Tiedonhaku	Tarvittavan informaation löytäminen ja parhaiden teknologioiden hyödyntäminen
Kommunikaatio	Mutkatonta ja säännöllistä yhteydenpitoa, yhteistyön sujuvuus ja henkilökemiat
Tiimityöskentely	Taitojen kehittyminen parikoodauksessa ja toisen auttamista ongelmatilanteissa
Tehtävien jako	Töiden jakaminen tasapuolisesti omien vahvuusalueiden mukaan suunnitellun aikataulun saavuttamiseksi
Ongelmanratkaisu	Ratkaisun hakeminen ja löytäminen ongelmatilanteissa

Projektinhallinta	Säännölliset palaverit sekä tuntiaikakirjanpidon ja backlogin ylläpito
Aikataulut	Aikataulujen kiinnittäminen ja prosessin keston arviointi
Päätöksenteko	Vaikeiden päätösten tekeminen (vanhan teknologian korvaamista uuteen)

3D-mallinnuksessa osaamistaitomme pohjautuivat pitkälti Haaga-Heliassa opettavan Heikki Hietalan ylläpitämään 3D-mallinnuskurssiin. Kurssilla raapaistiin lähinnä pintaa siitä, mitä mallinnus oikeasti on ja miten mallinnustyökaluja voidaan hyödyntää. Näiden tietotaitojen pohjalta lähdimme etenemään teoriasta käytäntöön. Suunnittelutyössä otimme huomioon ruukun mitat sekä anturien halkaisijat, jotta saimme tehtyä tarvittavia kiinnikkeitä halutun lopputuloksen saavuttamiseksi.

3D-tulostuksen osalta lähdimme opettelemaan prosessia niin sanotusti kantapään kautta. Tulostuksessa tuli monenlaista ongelmaa vastaan (muun muassa lanka loppui kesken sekä tulostin lakkasi tulostamasta meille tuntemattomasta syystä) ja jouduimme opettelemaan missä asennossa osia pystytään tulostamaan sekä miten lanka käyttäytyy tulostuksen aikana (lämpötila ja tulostusnopeus). Ongelmana meillä oli tulostuksen aikana kertynyt sulanut muovi, joka jähmettyi ja tarttui tulostettuihin osiin muodostaen hämähäkin seittejä muistuttavia muovisäikeitä. Muovisäikeet leikattiin veitsellä pois, sillä se ei vaikuttanut kasvatusruukkujen toimivuuteen.

Graafisen käyttöliittymän suunnittelussa silmä harjaantui esteettisyyden osalta. Käyttämämme komponentit, värit ja kuvat noudattavat samaa linjaa ja muodostavat yhtenevän rakenteen. Jokainen kohta on mietitty loppuun saakka, jotta saisimme mahdollisimman käyttäjäystävällisen, selkeän ja toimivan sovelluskokonaisuuden.

Opinnäytetyön kirjoittamisen osalta olemme kokeneet myös huimaa kehitystä. Oman alan terminologia on tullut tutuksi sekä tiedonhaku ja lähteiden hyödyntäminen ovat parantuneet. Olemme kehittyneet muun muassa lähdekriittisyydessä, asiatekstin kirjoittamisessa sekä opinnäytetyön kokonaisrakenteen hahmottamisessa (lineaarisesti etenevä, tekstin jäsentely, kappaleenvaihdot, tyylien muotoilu, kuvien asettelu, tekstirakenteen keventäminen ja viimeistelytyö).

Fyysisen ruukun osalta saimme kokemusta muun muassa vesitiiviyyden osalta. Aikaisemmassa projektissa ongelmaksi muodostui se, ettei 3D-tulostetut osat olleet vedenpitäviä, eikä käyttämämme gorillateippi ja liima ratkaisseet ongelmaamme. Ankara

tiedonhaku tuotti kuitenkin tulosta ja löysimme valoopoksihartsin sekä tiivistemassan, joiden ansiosta saimme tilkittyä ruukun vesitiiviiksi. Tämä oli esteettisesti myös parempi ratkaisu.

Oman oppimisen lisäksi myös muut sidosryhmät ovat hyötäneet työn tuloksesta. Opinnäytetyötämme on mahdollista käyttää esimerkkinä Haaga-Helian 3D-laboratorion IoT-projektikurssilla. Tämän lisäksi Heikki Hietala voi hyödyntää opinnäytetyöstä tullutta tietoa muun muassa ThingSpeakin osalta. Älyruukusta hyötyvät myös ne harrastelijat, jotka haluavat valmistaa omista materiaaleista valmiin ruukku- ja sovelluskokonaisuuden.

Koemme, että opinnäytetyö oli kokonaisuudessaan erittäin opettava prosessi, josta on varmasti hyötyä tulevaisuuden työelämässä niin työn haun kuin osaamisenkin osalta.

Lähteet

Agarwal, A. 26.6.2020. Understanding Automotive OTA (Over-the-Air Update). PathPartner blogi. Luettavissa: <https://www.pathpartnertech.com/understanding-automotive-ota-over-the-air-update/>. Luettu: 31.3.2021.

Allurwar, N. 11. 6.2020. How does IoT work? – explanation of IoT Architecture & layers. IoTDunia blogi. Luettavissa: <https://iotdunia.com/iot-architecture/>. Luettu: 15.8.2021.

AnySilicon s.a. What is a System on Chip? Luettavissa: <https://anysilicon.com/what-is-a-system-on-chip-soc/>. Luettu: 7.9.2021.

Arduino s.a.a. What is Arduino. Luettavissa: <https://www.arduino.cc/en/guide/introduction>. Luettu: 29.3.2021.

Arduino s.a.b. Arduino Software (IDE). Luettavissa: <https://www.arduino.cc/en/guide/environment>. Luettu: 9.9.2021.

Arduino s.a.c. Libraries. Luettavissa: <https://www.arduino.cc/reference/en/libraries/>. Luettu: 10.9.2021.

Arduino s.a.d. Variables. Luettavissa: <https://www.arduino.cc/en/Reference/VariableDeclaration>. Luettu: 10.9.2021.

Arduino s.a.e. Introduction to the Arduino Board. Luettavissa: <https://www.arduino.cc/en/reference/board>. Luettu: 10.9.2021.

Arduino Store s.a. Arduino Uno Rev3. Luettavissa: <https://store.arduino.cc/products/arduino-uno-rev3/>. Luettu: 15.9.2021.

AspenCore s.a. Electrical Relay. Electronics Tutorials. Luettavissa: https://www.electronics-tutorials.ws/io/io_5.html. Luettu: 17.9.2021.

Blender Foundation 2021. Python API Overview. Python API Documentation. Luettavissa: https://docs.blender.org/api/current/info_overview.html. Luettu: 28.9.2021.

Boral, S. 25.5.2020. What is ESP32 and Why Is It Best for IoT Projects. IoT Tech Trends blogi. Luettavissa: <https://www.iottechrends.com/what-is-esp32/>. Luettu: 29.3.2021.

Boubel, G. s.a. What is a TDS meter and do you need one? Aquasana. Luettavissa: <https://www.aquasana.com/info/tds-meter-what-is-it-and-do-you-need-it-pd.html>. Luettu: 2.10.2021.

Braun, A. 25.1.2019. History of IoT: A Timeline of Development. IoT Tech Trends. Luettavissa: <https://www.iottechrends.com/history-of-iot/>. Luettu: 14.7.2021.

Browning, D. 25.7.2018. IoT Started with a Vending Machine. Machine Design. Luettavissa: <https://www.machinedesign.com/automation-iiot/article/21836968/iot-started-with-a-vending-machine>. Luettu: 1.7.2021.

Carolo, L. 13.4.2020. What Is a 3D Slicer? – Simply Explained. All3DP. Luettavissa: <https://all3dp.com/2/what-is-a-3d-slicer-simply-explained/>. Luettu: 2.10.2021.

CERN s.a.a. A short history of the Web. Luettavissa: <https://home.cern/science/computing/birth-web/short-history-web>. Luettu: 10.5.2021.

CERN s.a.b. A screenshot showing the NeXT world wide web browser created by Tim Berners-Lee. Luettavissa: <https://cds.cern.ch/images/CERN-IT-9001001-01>. Luettu: 10.5.2021.

Cisco IBSG 2011. The Internet of Things – How The Next Evolution of the Internet Is Changing Everything. Cisco White Paper. Luettavissa: https://www.cisco.com/c/dam/en_us/about/ac79/docs/innov/IoT_IBSG_0411FINAL.pdf. Luettu: 12.7.2021.

Cloudflare s.a. What is the Mirai Botnet. Luettavissa: <https://www.cloudflare.com/learning/ddos/glossary/mirai-botnet/>. Luettu: 2.9.2021.

DFRobot s.a. Gravity: Analog pH Sensor / Meter Kit For Arduino. Luettavissa: <https://www.dfrobot.com/product-1025.html>. Luettu: 21.9.2021.

Digiteum Team 1.5.2021. How to Use Internet of Things (IoT) in the Retail Industry. Digiteum blogi. Luettavissa: <https://www.digiteum.com/internet-of-things-retail-industry/>. Luettu: 23.8.2021.

diymore s.a. ESP32-DevKitC V4 ESP32 Based Development Board Module ESP32 WROOM-32D. Luettavissa: <https://www.diymore.cc/products/esp32-devkitc-v4-esp32-based-development-board-module-esp32-wroom-32d>. Luettu: 21.9.2021.

electric_studio s.a. DC Small Mini Air Pump + Tube. eBay. Luettavissa: <https://www.ebay.com.sg/itm/DC5V-12V-6V-9V-Small-Mini-370-Motor-Air-Pump-Oxygen-Pump-Tube-Aquarium-Fish-Tank/383803953991>. Luettu: 15.9.2021.

ETH Zurich. 2009. Internet of Things 2008. International Conference For Industry And Academia March 26-28, 2008 / Zurich. Luettavissa: <https://iot-conference.org/iot2008/>. Luettu: 12.7.2021.

Featherly, K. s.a. ARPANET. Britannica. Luettavissa: <https://www.britannica.com/topic/ARPANET>. Luettu: 18.6.2021.

Great Stuff Hydroponics s.a. Tracking and controlling pH levels in hydroponics. Luettavissa: <https://www.hydroponics-hydroponics.com/tracking-and-controlling-ph-levels-in-hydroponics-i28>. Luettu: 23.9.2021.

Gregersen, E. s.a. World Wide Web information network. Britannica. Luettavissa: <https://www.britannica.com/topic/World-Wide-Web>. Luettu: 27.6.2021.

Guru 99 s.a. What is a Database? Definition, Meaning, Types, Example. Luettavissa: <https://www.guru99.com/introduction-to-database-sql.html>. Luettu: 31.3.2021.

Haglund, J. 9.1.2018. Järjestelmäintegraatio, mitä se on selkokielellä? Alfame blogi. Luettavissa: <https://www.alfame.com/blog/jarjestelmaintegraatio-mita-se-on-selkokielella>. Luettu: 14.10.2021.

Holst 2021. Number of IoT connected devices worldwide 2019-2030. Statista. Luettavissa: <https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/>. Luettu: 6.9.2021.

HQSoftware 12.7.2017. The History of IoT: a Comprehensive Timeline of Major Events, Infographic. HQSoftware blogi. Luettavissa: <https://hqsoftwarelab.com/blog/the-history-of-iot-a-comprehensive-timeline-of-major-events-infographic/>. Luettu: 7.7.2021.

Ikasamo s.a. AutoConnect for ESP8266/ESP32. GitHub, Inc. Luettavissa:
<https://github.com/Hieromon/AutoConnect>. Luettu: 4.10.2021.

Javatpoint s.a. Firebase: Realtime Database. Luettavissa:
<https://www.javatpoint.com/firebase-realtime-database>. Luettu: 12.11.2021.

Jefferson Online 22.11.2016. From ARPANET to World Wide Web: An Internet History Timeline. Thomas Jefferson University blogi. Luettavissa:
<https://online.jefferson.edu/business/internet-history-timeline/>. Luettu: 11.6.2021.

Keyestudio Wiki 2021. KSo429 keyestudio TDS Meter V1.0. Luettavissa:
https://wiki.keyestudio.com/KS0429_keyestudio_TDS_Meter_V1.0. Luettu: 23.9.2021.

Khvoynitskaya, S. 25.11.2019. The IoT history and future. Itransition blogi. Luettavissa:
<https://www.itransition.com/blog/iot-history>. Luettu: 30.6.2021.

Knud, L. 19.12.2014. Why the Internet of Things is called Internet of Things: Definition, history, disambiguation. IoT Analytics blogi. Luettavissa: <https://iot-analytics.com/internet-of-things-definition/>. Luettu: 30.6.2021.

Koponen, J., Mustonen, V. & Spilling, K. 2014. Älykäs kaupunki – Smart City. Liikenne- ja viestintäministeriö. Helsinki. Luettavissa:
http://julkaisut.valtioneuvosto.fi/bitstream/handle/10024/161292/TEM_3_2019_Matkailun_toimialaraportti.pdf. Luettu: 14.7.2021.

Kruglyak, I. 9.12.2020. 20 Examples of Wearables and IoT Disrupting Healthcare. Avenga. Luettavissa: <https://www.avenga.com/magazine/wearables-iot-healthcare/>. Luettu: 23.8.2021.

Lane, K. 28.9.2020. API 101: What Is a REST API. Postman blogi. Luettavissa:
<https://blog.postman.com/rest-api-definition/>. Luettu: 31.3.2021.

Living Internet s.a. The Internet Toaster. Luettavissa:
https://www.livinginternet.com/i/ia_myths_toast.htm. Luettu: 6.7.2021.

McClelland, C. 29.10.2016. IoT Explained – How Does an IoT System Actually Work. Leverage blogi. Luettavissa: <https://www.leverage.com/blogpost/iot-explained-how-does-an-iot-system-actually-work>. Luettu: 29.3.2021.

Mouser s.a. ESP32-DevKitC-32D. Luettavissa:

<https://eu.mouser.com/ProductDetail/Esspressif-Systems/ESP32-DevKitC-32D?qs=%252BEew9%252B0nqrDsObWEpDx6YQ==>. Luettu: 15.9.2021.

Oracle Corporation s.a. What is IoT. Luettavissa: <https://www.oracle.com/internet-of-things/what-is-iot/>. Luettu 14.7.2021.

Petty, J. s.a. What is 3D Modeling & What's It Used For? Concept Art Empire. Luettavissa: <https://conceptartempire.com/what-is-3d-modeling/>. Luettu: 23.9.2021.

Puzhevich, V. 25.7.2019. Internet of Things In Smart Home. Scand blogi. Luettavissa: <https://scand.com/company/blog/internet-of-things-in-smart-home/>. Luettu: 17.8.2021.

RAND Corporation s.a. Paul Baran and the Origins of the Internet. Luettavissa: <https://www.rand.org/about/history/baran.html>. Luettu: 15.6.2021.

Riganti, M. 16.7.2018. Low Poly Planter – Wide Top. Thingiverse. Luettavissa: <https://www.thingiverse.com/thing:2999036>. Luettu: 21.9.2021.

Scandinavian Photo s.a. 3D-tulostus: Valitse juuri sinulle sopivan tulostimen! Luettavissa: <https://www.scandinavianphoto.fi/inspiroidu/teemasivut/3d-tulostus-valitse-juuri-sinulle-sopivan-tulostimen>. Luettu: 30.9.2021.

Schönlau, T. 4.4.2018. Netcup. Thingiverse. Luettavissa: <https://www.thingiverse.com/thing:2850442>. Luettu: 1.8.2021.

Shiklo, B. 25.12.2018. IoT in Manufacturing: The Ultimate Guide. ScienceSoft blogi. Luettavissa: <https://www.scnsoft.com/blog/iot-in-manufacturing>. Luettu: 19.7.2021.

SIMON IoT 2020. The Rise of IoT: The History of the Internet of Things. Luettavissa: <https://www.simoniot.com/history-of-iot/>. Luettu: 8.7.2021.

Sirk, C. 24.7.2020. Ted Nelson, Hypertext, and Hippie Modernism. CRM.ORG. Luettavissa: <https://crm.org/articles/ted-nelson-hypertext-and-hippie-modernism>. Luettu: 18.6.2021.

Slick, J. 24.9.2020. What Is 3D-modeling. Lifewire. Luettavissa: <https://www.lifewire.com/what-is-3d-modeling-2164>. Luettu: 29.3.2021.

Slump, G. 27.5.2021. PLA vs ABS vs PETG: The Differences. All3DP. Luettavissa: <https://all3dp.com/2/pla-vs-abs-vs-petg-differences-compared/>. Luettu: 1.10.2021.

Social Industries 23.5.2018. Did it Really Start with MySpace? A Brief History of Social Media. Social Industries blogi. Luettavissa: <https://social.industries/2018/05/23/did-it-really-start-with-myspace-a-brief-history-of-social-media/>. Luettu: 27.6.2021.

Stacey, L. 16.6.2020. MQTT beginner's guide. ublox blogi. Luettavissa: <https://www.ublox.com/en/blogs/insights/mqtt-beginners-guide>. Luettu 31.3.2021.

Starostin, O. 15.4.2021. IoT Matters in 2021: IoT Enabled Smart Office Automation to Work and Collect Data Human-Free. Hackernoon. Luettavissa: <https://hackernoon.com/iot-matters-in-2021-iot-enabled-smart-office-automation-to-work-and-collect-data-human-free-je1b33j1>. Luettu: 27.7.2021.

Strongbytes 2019. The Perfect Match: Artificial Intelligence and The Internet of Things. Luettavissa: <https://strongbytes.ai/artificial-intelligence-and-the-internet-of-things-a-perfect-match/>. Luettu: 30.8.2021.

survy2014 s.a. 2PCS One 1 Channel Isolated 5V Relay Module Coupling For Arduino optocoupler. eBay. Luettavissa: <https://www.ebay.com/itm/201414873910>. Luettu: 17.9.2021.

Taylor, K. s.a. Everything You Need To Know About Consumer IoT (CIoT). HiTechNectar blogi. Luettavissa: <https://www.hitechnectar.com/blogs/consumer-iot-ciot/>. Luettu: 15.8.2021.

Techopedia 2018. 3D Printing. Luettavissa: <https://www.techopedia.com/definition/17475/additive-manufacturing>. Luettu: 29.9.2021.

TheMathWorks, Inc. 2021. Learn More About ThingSpeak. Luettavissa: https://thingspeak.com/pages/learn_more. Luettu: 11.11.2021.

The RapidAPI s.a. API Key – What is an API Key. The RapidAPI blogi. Luettavissa: <https://rapidapi.com/blog/api-glossary/api/>. Luettu 29.3.2021.

Tikka, T. 22.4.2015. Teollinen internet – mikä se on. Tivi blogi. Luettavissa:
<https://www.tivi.fi/kumppaniblogit/tieto/teollinen-internet-mika-se-on/7527cb6f-715b-314e-b5ed-420e57214b54>. Luettu: 30.6.2021.

World Wide Web Foundation s.a. History of the Web. Luettavissa:
<https://webfoundation.org/about/vision/history-of-the-web/>. Luettu: 5.5.2021.

2NS s.a. Kriittisimmät haavoittuvuudet web-sovelluksissa. Tietoturvaopas 2. Luettavissa:
<https://www.2ns.fi/wp-content/uploads/2018/06/2ns-Pikaopas-kriittisimmista-haavoittuvuuksista-web-sovelluksissa.pdf>. Luettu: 1.9.2021.

3DMaker Engineering s.a. How 3D Printing Works. Luettavissa:
<https://www.3dmakerengineering.com/blogs/3d-printing/how-3d-printing-works>. Luettu: 2.10.2021.