

Hans Hokka

Palvelurobotiikan opetus lukiolaisille - tapaustutkimus

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Tietotekniikka

Insinöörityö

20.9.2018

Tekijä(t)	Hans Hokka
Otsikko	Robotiikan opettaminen lukiolaisille - tapaustutkimus
Sivumäärä	31 sivua + 4 liitettä
Aika	20.9.2018
Tutkinto	Insinööri (AMK)
Tutkinto-ohjelma	Tietotekniikka
Ammatillinen pääaine	Ohjelmistotekniikka
Ohjaaja(t)	Yliopettaja Päivi Haho
<p>Tämän lopputyön aiheena on Metropolia Ammattikorkeakoulussa järjestettävän robotiikka-kurssin suunnittelu ja saatujen kokemusten tarkastelu. Lisäksi lopputyön tarkoituksena on toimia valmiina pohjana uusille robotiikkakursseille. Kursseja järjestettiin kaksi: toinen TET-harjoittelijoille, toinen lukiolaisille. Aihe työlle syntyi Metropolian halukkuudesta tarjota tällainen kurssi toisen asteen opiskelijoille.</p> <p>Työn tarkoituksena oli saada käsitys siitä, millaiseksi robotiikan opetus pitää muodostaa toisen asteen opiskelijoita varten. Työn puitteissa suunnitellaan kurssi alusta alkaen. Käsitellään opetuksessa käytettävät robotit sekä ohjelmointikielien. Tämän jälkeen luodaan tarvittava oppimateriaali sekä muu materiaali, mm. YouTube-mainos. Ensimmäisen kurssin jälkeen kerätään välitulokset, joiden perusteella tehdään kurssiin tarvittavat muutokset toista toteutusta varten.</p> <p>Työn lopputuloksena voidaan saadun palautteen sekä opetuskokemusten perusteella todeta Nao-robotin sekä Choregraphe-ohjelmointikielen sopivan hyvin toisen asteen opetukseen. Kurssien toteutukset olisivat palautteen perusteella voineet olla pidemmät, jolloin olisi voitu mennä robotiikkaan ja ohjelmointiin syvemmälle. Opetettava materiaali oli joka tapauksessa määrältään sopiva kurssien pituuksiin verrattuna.</p>	
Avainsanat	Choregraphe, Nao, robotiikka

Author(s)	Hans Hokka
Title	Teaching service robotics to high schoolers (!)
Number of Pages	31 pages + 4 appendices
Date	20 September 2018
Degree	Bachelor of Engineering
Degree Programme	Computer Engineering
Professional Major	Software Engineering
Instructor(s)	Päivi Haho, Principal Lecturer
<p>The subject of this final thesis is the design of a robotics course at Metropolia Polytechnic and a review of the experiences gained. In addition, the aim of the thesis is to act as a basis for new robotics courses. Two courses were organized, one for TET trainees, one for high school students. The subject matter was the Metropolia's willingness to provide such a course for secondary students.</p> <p>The purpose of the thesis was to get an idea of how robotic teaching should be made for secondary students. The course is designed from the beginning. The robots and the programming languages were surveyed. After that, the necessary material and other materials, e.g. YouTube advertising were created. After the first course, intermediate results are collected to make changes to the course for the second implementation.</p> <p>As a result of the work, based on the feedback received and the teaching experience, the Nao robot and the Choregraphe programming language can be well adapted to the second level of instruction. The implementation of the courses could have been longer in duration based on the feedback. The course could have gone deeper into robotics and programming. The material to be taught was, in any case, an appropriate amount compared to the length of the courses.</p>	
Keywords	Choregraphe, Nao, robotics

Sisällys

1	Johdanto	2
2	Opinnäytetyön toteutus	3
2.1	Lähtötiedot tulevista osallistujista	3
2.2	Robotin valinta	3
2.2.1	Turtlebot 3	3
2.2.2	Arduino	5
2.2.3	Lego Mindstorms	5
2.2.4	Nao	7
2.2.5	Miksi Nao	7
2.3	Ohjelmointikielen valinta	8
2.3.1	Esimerkkiohjelma C++:lla	8
2.3.2	Esimerkkiohjelma Pythonilla	10
2.3.3	Esimerkkiohjelma Choregraphella	10
2.3.4	Choregraphe tarkemmin	12
2.4	Opetusmetodin valinta	12
2.4.1	Behavioristinen oppimiskäsitys	13
2.4.2	Humanistinen oppimiskäsitys	14
2.4.3	Kognitiivinen oppimiskäsitys	15
2.4.4	Konstruktiivinen oppimiskäsitys	16
2.4.5	Ongelmakeskeinen oppiminen (Problem-based learning, PBL)	17
3	Opinnäytetyön tulokset	18
3.1	Robottikurssin kuvaus	18
3.1.1	Osallistujat, materiaalit ja menetelmät	18
3.1.2	Kurssin sisältö	19
3.2	Palaute kurssista	31
4	Yhteenveto	33
	Lähteet	34
	Liitteet	36
	Liite 1. Roboviikon esite	36
	Liite 2. Kurssitodistus	37

Liite 3. Robokurssin YouTube-mainos	38
Liite 4. Luettelo kaikesta käytetystä ja tuotetusta materiaalista.	39

Kiitokset

Lämmin kiitos kaikille projektiin osallistuneille, erityisesti Jouni Heleniukselle.

1 Johdanto

Insinööriyön tavoitteena on suunnitella ja toteuttaa robotiikkakurssi toisen asteen opiskelijoille. Lisäksi tarkoituksena on toteuttaa ja tallettaa kaikki kurssilla käytettävä materiaali, esim. opetuskalvot, esimerkkiohjelmat sekä mainokset.

Robotiikka on oma laaja tieteenalansa. Mahdollisuuksia sen opettamiseen on lukemattomia. Riippuen kohderyhmästä ja käytettävissä olevasta ajasta mahdollisuuksista pitää valita juuri oikeat. Jos haasteita on suhteessa kohderyhmään liikaa tai liian vähän, oppiminen ei ole mielenkiintoista ja motivaatio laskee.

Metropolia Ammattikorkeakoulussa järjestettävässä kurssissa tiedettiin kohderyhmä sekä käytettävissä olevat tuntimäärät. Kohderyhmän osaamistasoa eikä oppimiskoetta vielä kuitenkaan tiedetty. Annettujen tietojen perusteella selvitettiin mahdollisimman tarkasti, millä ohjelmointikielellä ja millä robotilla opetus olisi järkevin toteuttaa.

Tämä oppinäytetyön kirjallinen osuus aloitetaan kartoittamalla kohderyhmä. Seuraavaksi käydään läpi mahdolliset valinnat robotiksi, ohjelmointikieleksi sekä opetusmetodiksi. Tämän jälkeen selvitetään kurssin tekniseen sisältöön vaikuttaneet tekijät sekä opetusmetodin valintaan johtaneet seikat. Oppinäytetyön tulokset esitetään, jonka jälkeen pureudutaan johtopäätöksiin.

Edellä mainittujen seikkojen perusteella käytettäväksi robotiksi valikoitui Aldebaran Roboticsin Nao. Mahdolliseksi ohjelmointikieleksi jäi robotin valinnan perusteella C++, Python sekä Choregraphe. Näistä valittiin visuaalinen vaihtoehto Choregraphe. Kurssin perusta oli näin muodostettu.

Oppimateriaalin luominen aloitettiin Naoon sekä Choregrapheen tutustumalla. Näillä toteutettuja kursseja löytyi haulla vain yksi. Sen oppimateriaalin pohjalta suunniteltiin varsinainen opetussisältö sekä -materiaali. Kurssin alussa oppimateriaali tehtiin uudelleen pääasiassa oppijoiden nopean omaksumisen seurauksena.

Lisäksi kurssille tarvittiin opetusmateriaali (kalvot, esimerkkiohjelmat, Choregraphen lisämoduulit), mainosmateriaali (pdf-tiedosto sekä YouTube-mainos) sekä kurssin todistukset ja muut sekalaiset materiaalit. Nämä luotiin käyttämällä mm. Adobe Illustratoria sekä Adobe Premieriä.

2 Opinnäytetyön toteutus

2.1 Lähtötiedot tulevista osallistujista

Kun aloimme suunnitella kurssia, meillä oli pohjana tietyt vaatimukset. Opiskelijoiden tuli olla 16–19-vuotiaita, PK-seudulta sekä joko lukiolaisia, peruskoulun päättäneitä tai Omnian tai Varian opiskelijoita.

Kurssille haettiin opiskelijoita, joita kiinnostivat robotit ja ohjelmointi. Opiskelijoiden määrä oli rajattu 25:een. Ajankohdaksi oli määrätty 7. – 11.8.2017, paikaksi Metropolian Leppävaaran kampus. Lukiolaisille kurssi hyväksiluettiin soveltaviin kursseihin.

Ikäraja oli kapea, jolloin osattiin ennakoida tarkasti, minkä ikäisiä opiskelijoita kurssille on tulossa. Koulutustason määrittely oli laajahko, joten se ei antanut juurikaan työkaluja kurssin suunnitteluun. Vapaaehtoisuudesta sekä ajankohdasta (kesäloma) voitiin päätellä, että kurssille hakeutuu henkilöitä, joita kurssin sisältö erityisesti kiinnostaa.

2.2 Robotin valinta

Kurssia suunniteltaessa tutkittiin tarkasti eri vaihtoehtoja opetusrobotiksi. Kattava lista löytyy liitteestä 2. Huomioiden kurssin vaatimukset, neljä vaihtoehtoa nousi esiin. Seuraavassa käsitellään jokainen erikseen.

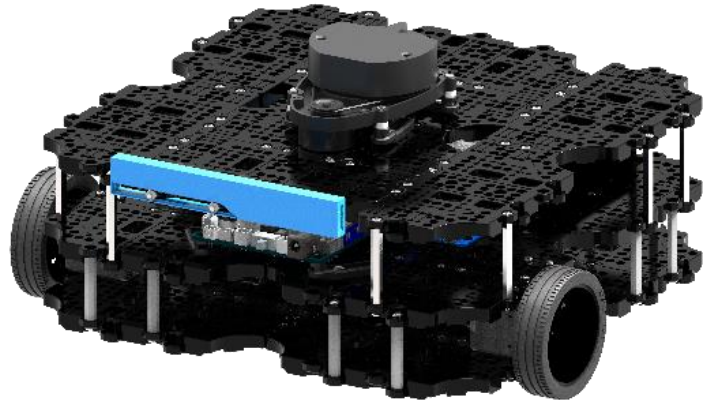
2.2.1 Turtlebot 3

Turtlebot 3 on uusin malli Turtleboteista. Se on pienikokoinen (138 mm x 178 mm x 192 mm) ja kohtuuhintainen (644 €). Turtlebot 3 käyttää ROSia (Robot Operating System). Sensoreita ja moottoreita siinä on kattava määrä. Se on myös laajennettava. Valmistajan mukaan robotti on maailman suosituin ROS:ia käyttävä laite. [1.]

TurtleBot3
Burger ↻



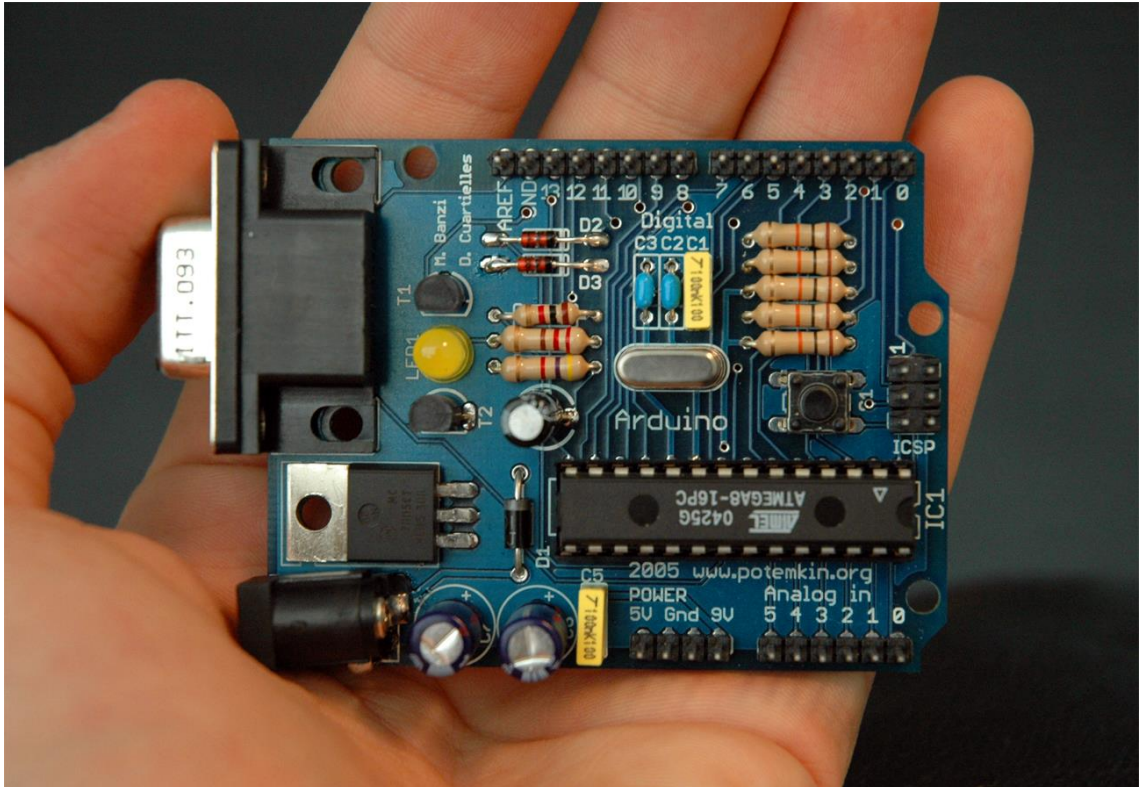
TurtleBot3
Waffle ↻



Kuva 1. Turtlebot 3

2.2.2 Arduino

Arduino on yleisnimi 22 eri laitteelle. Opetukseen näistä sopii parhaiten malli CTC 101. CTC 101 toimitetaan pakettina, joka sisältää yli 700 komponenttia ja osaa. Lisäksi pakettiin kuuluu kattava oppimateriaali, joka sisältää valmiita projekteja, esimerkkiohjelmia ja käyttäjätuen. Jokainen paketti maksaa 2170 €. CTC 101:lle on olemassa valtava määrä open source -dataa [2.]

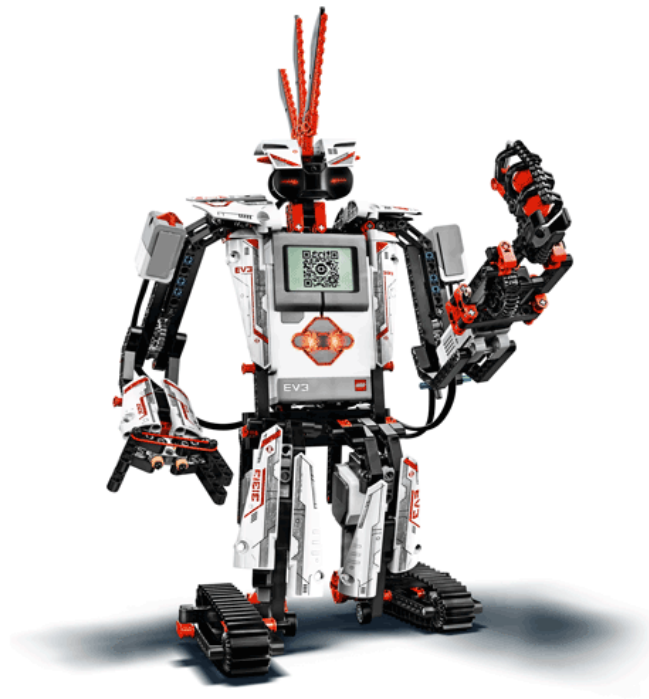


Kuva 2. Arduino

2.2.3 Lego Mindstorms

Lego Mindstorms EV3 on Legon uusin robottisarja. Hinnaltaan se on edullinen, 399,95 €. Laitteessa on kattavat liitännät, mm. USB, Wlan sekä Micro SD -kortinlukija. Pakettiin kuuluu interaktiivisia servomootoreita, kaukosäädin, EV3 Brick ja Lego-palikoita. Palikoilla robotin voi rakentaa haluamallaan tavalla. Valmiit rakennusohjeet löytyvät 17 eri robotille, mm. humanoidirobotti, kävelevä dinosaurus sekä luikertelevä käärme.

Sensoreita, jotka tulevat mukaan, ovat värintunnistin, kosketuksentunnistin ja infrapunasensori. Ohjelmointiohjelmisto löytyy tableteille, PC:lle ja Macille. Älypuhelimeen voi ladata Commander-sovelluksen, jolla voi ohjata robottia. [3, 4.]

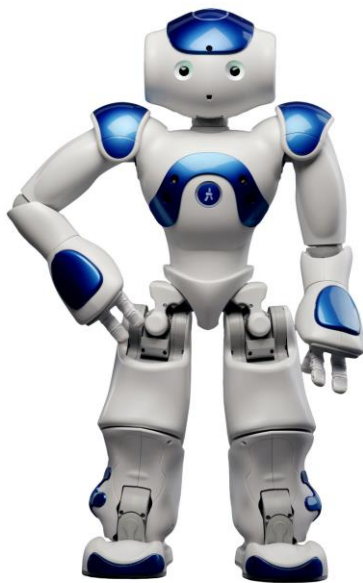


Kuva 3. Lego Mindstorms EV3

2.2.4 Nao

Aldebaranin valmistama Nao on vertailluista roboteista kallein, noin 5000 €. Siinä on yli 50 sensoria, mm. 2 hd-kameraa ja neljä suuntaavaa mikrofonia. Nao osaa esimerkiksi tunnistaa kasvot, puhua ja kuunnella, kävellä ja käsitellä esineitä. Useiden ominaisuuksiensa ansiosta Nao on monipuolisesti kiinnostava, käytännöllinen ja sopii moneen tarkoitukseen.

Nao painaa 5,4 kg, ja se on kestävä, suunniteltu käytettäväksi. Robotin kaatuessa se nousee itsestään ylös. Naoa voi ohjelmoida visuaalisella Choregraphe-ohjelmointikielillä. Tämä mahdollistaa progressiivisen opetuksen, jossa oppimiskäyrä on järkevä. [5.]



Kuva 4. Nao

2.2.5 Miksi Nao

Metropolialle oli hankittu jo valmiiksi useita Nao-robotteja. Kallis hinta ei siis muodostunut esteeksi. Päätökseen vaikutti myös se, että Naolla pystyy tekemään paljon asioita pienellä vaivalla. Näin oppilaat pääsivät tekemään heti alussa konkreettisia sovelluksia.

Vaihtoehtoja robotin valinnalle ei ollut lopulta paljoa. Arduino, Lego sekä TurtleBot ovat hyvin erilaisia. Arduinossa on korkeampi oppimiskynnys, samoin TurtleBotissa. Lego on taas helpompi oppia, mutta se on muita merkittävästi rajoittuneempi.

2.3 Ohjelmointikielen valinta

Naao voi ohjelmoida kolmella eri ohjelmointikielellä: C++:lla, Pythonilla sekä Choreographella. Ennakkotietojen perusteella usealla ei ollut mitään ohjelmointikokemusta. Pythonin opettamiseen olisi tarvittu erillinen kurssi, samoin kuin C++:n. Choreographe visuaalisena ohjelmointikielenä aukeaa nopeiten, jolloin kurssilla voidaan keskittyä itse tarkoitukseen, robotin ohjaamiseen. [6.]

2.3.1 Esimerkkiohjelma C++:lla

C++ soveltuu erinomaisesti tiettyihin käyttötarkoituksiin, mutta se on myös kolmesta vaihtoehdosta työläin opetella. Esimerkkikoodeissa 1 ja 2 näytetään Hello World -esimerkkiohjelma C++:lla Naolle. [7.]

```
-  /**
-   * Copyright (c) 2011 Aldebaran Robotics. All Rights Reserved
-   * \file sayhelloworld.cpp
-   * \brief Make NAO say a short phrase.
-   *
-   * A simple example showing how to make NAO say a short phrase using the
-   * specialized proxy ALTextToSpeechProxy.
-   */
-
-
-
- #include <iostream>
- #include <alerror/alerror.h>
- #include <alproxies/altexttospeechproxy.h>
-
-
- int main(int argc, char* argv[])
- {
-     if(argc != 2)
-     {
-         std::cerr << "Wrong number of arguments!" << std::endl;
```

```

-     std::cerr << "Usage: say NAO_IP" << std::endl;
-     exit(2);
- }
-
- /** The desired phrase to be said. */
- const std::string phraseToSay = "Hei maailma!";
- try
- {
-     /** Create an ALTextToSpeechProxy so that we can call the say method
-     * Arguments for the constructor are:
-     * - IP of the robot
-     * - port on which NAOqi is listening. Default is 9559
-     */
-     AL::ALTextToSpeechProxy tts(argv[1], 9559);
-
-     /** Call the say method */
-     tts.say(phraseToSay);
- }
- catch (const AL::ALError& e)
- {
-     std::cerr << "Caught exception: " << e.what() << std::endl;
-     exit(1);
- }
- exit(0);
- }

```

Esimerkkikoodi 1. Hello world C++:lla, cpp-tiedosto.

```

1. ##
2. # Copyright (c) 2011 Aldebaran Robotics. All Rights Reserved.
3.
4. cmake_minimum_required(VERSION 2.6.4 FATAL_ERROR)
5. # Give a name to the project.
6. project(sayhelloworld)
7. # This include enable you to use qibuild framework
8. find_package(qibuild)
9.
10. # Create an executable named sayhelloworld,
11. # with the source file : sayhelloworld.cpp
12. qi_create_bin(sayhelloworld sayhelloworld.cpp)
13.
14. # Tell CMake that sayhelloworld depends on ALCOMMON.

```

```

15. # This will set the libraries to link sayhelloworld with,
16. # the include paths, and so on
17. qi_use_lib(sayhelloworld ALCOMMON)

```

Esimerkkikoodi 2. Hello World C++:lla, txt-tiedosto.

2.3.2 Esimerkkiohjelma Pythonilla

Pythonin oppimiskäyrä on loivempi kuin C++:lla, mutta se vaatii silti tekstipohjaisen ohjelmoinnin perusteiden oppimisen. Esimerkkikoodi 3 näyttää, miten Hello World ohjelmoidaan Pythonilla Naolle. [8.]

```

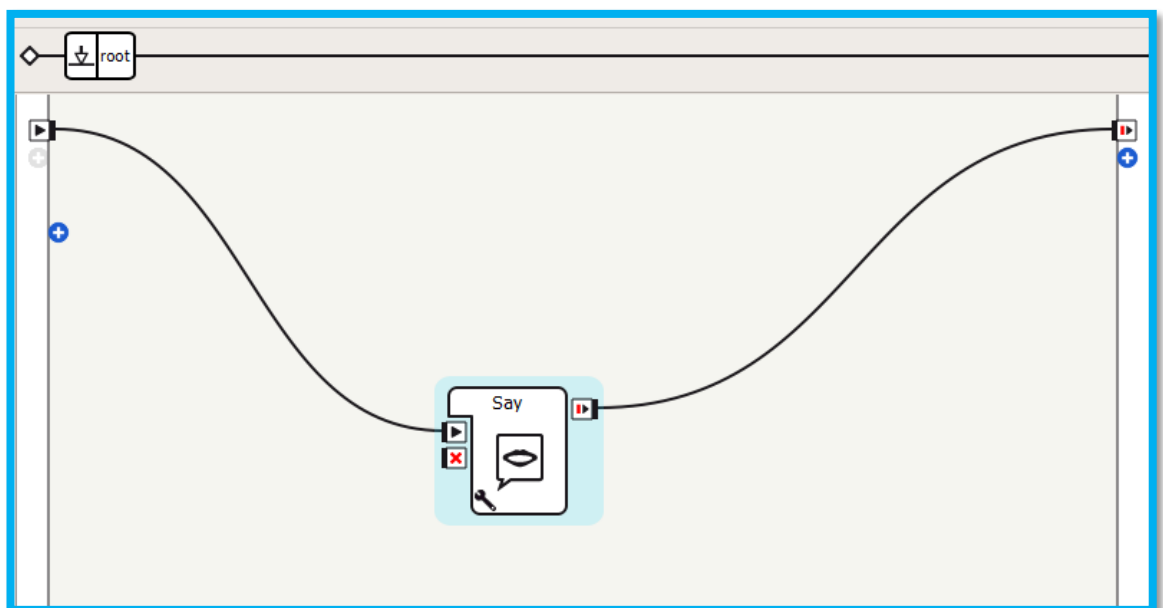
1. from naoqi import ALProxy
2. tts = ALProxy("ALTextToSpeech", "<IP of your robot>", 9559)
3. tts.say("Hei maailma!")

```

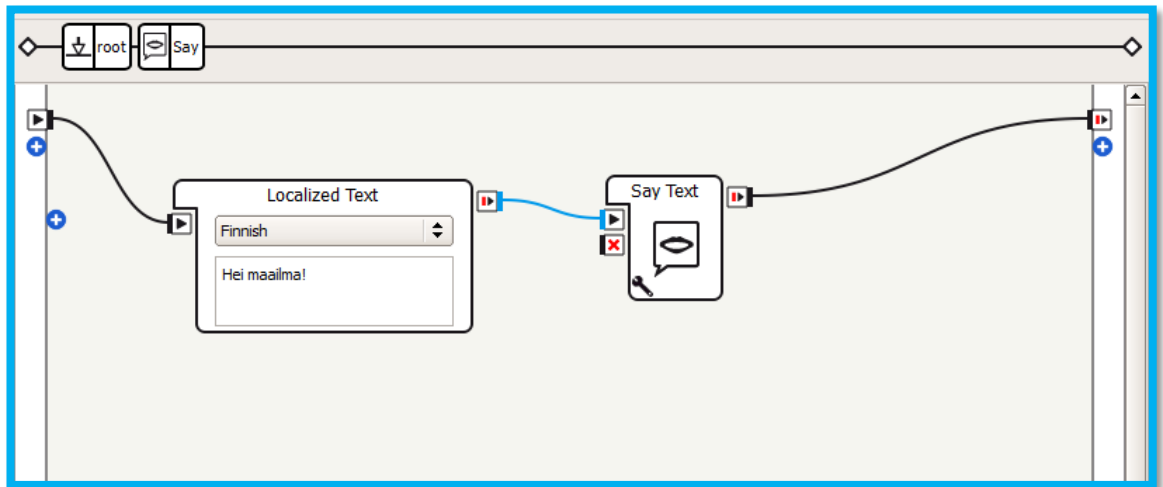
Esimerkkikoodi 3. Hello World Pythonilla, py-tiedosto.

2.3.3 Esimerkkiohjelma Choregraphella

Choregraphessa ohjelmoinnin perusteiden opettelu jää pois, jolloin voidaan keskittyä oleelliseen. Kuvissa 5 ja 6 demonstroidaan Hello World Choregraphella. [9.]



Kuva 5. Say-moduuli Choregraphessa



Kuva 6. Say-moduulin sisältö

2.3.4 Choregraphe tarkemmin

Choregraphe toimii lähes kaikilla alustoilla, Windowsilla, Macilla ja Linuxilla. Kuten todettu, se on visuaalinen. Se on työpöytäsovellus, jolloin erillisiä ohjelmointiympäristöjä ei tarvitse pystyttää.

Choregraphella voi luoda helposti animaatioita Naolle. Choregraphessa voi simuloida robotin toimintaa ennen robotin käskyttämistä. Kaiken oleellisen voi tehdä visuaalisesti. Jos tarvitaan lisää ominaisuuksia, voidaan valmiita moduuleita muokata Pythonilla. Muokkaaminen tapahtuu valmista koodia muuttamalla, joten tekstikoodausta ei tarvitse erikseen opetella. [10.]

2.4 Opetusmetodin valinta

Mitä oppimisella ylipäätään tarkoitetaan? Hietikon [11] mukaan oppimisella tarkoitetaan muun muassa oppijan informaation määrän kasvamista, sen painamista mieleen, palauttamista muistista ja prosessointia niin, että oppijan maailmankuva muuttuu ja oppija kehittyy yksilönä. Salmisen ja Suhosen [12] mukaan oppimista on se, että oppija koostaa oman käsityksensä aiheena olevista asioista. Tämän tuloksena oppija harjaantuu vanhojen ja opittujen taitojen käytössä. Oppimisen saavutukset eivät ole samat kaikilla, vaan joskus oppimistulokseksi jää hyvinkin pinnallinen muistijälki, joskus taas päästään syvään ymmärrykseen ja tiedon soveltamisen mahdollistamiseen. Opetus, paraskaan mahdollinen, ei siis takaa automaattisesti hyvää oppimistulosta. [12.]

Nykyään katsotaan, että oppiminen on aktiivista toimintaa. Samoin nykyisissä oppimiskäsityksissä painotetaan oppijan aktiivisuutta. Aktiivisuutta saadaan oppimiskäsitysten mukaan lisättyä painottamalla opetuksessa sekä yhteisöllisyyttä sekä yksilöllisyyttä. [11.]

Aktiivinen oppija prosessoi uutta tietoa vanhojen tietojen avulla, ja aktiivista oppijaa rohkaistaan kertomaan, miten itse käsittää asiat. Oppijoita ylipäätään kannustetaan keskustelemaan asioista ja kertomaan mielipiteensä niistä. Aktiivisessa oppimisessa opettajan tarkoituksena on oppimisen helpottaminen virittämällä keskusteluita ja yhteistoimintaa. [11.]

Passiivinen oppija taas saa tiedon valmiina opettajalta, ja painaa sen mieleensä, jonka jälkeen hän soveltaa sitä. Jos oppiminen on passiivista, dialogia ei ole, vaan tieto siirtyy vain opettajalta oppijalle, ei toiseen suuntaan. [11.]

Oppijan omaksuma tieto antaa hänelle mahdollisuuden luoda jollekin toiselle arvoa. Koulututkinnon suorittaminen antaa usein ammatin. Tutkinnon avulla myös oletettavasti menestyy paremmin työmarkkinoilla. [11.]

Opettajalla on teoriaan ja käytäntöön perustuva kuva opetuksesta ja oppimisesta. Tätä sanotaan oppimiskäsitykseksi [13]. Oppimiskäsityksiä on useita ja niistä konstruktivinen oppimiskäsitys on nykyään vallalla [12].

Oppimiskäsitykset antavat opettajalle pohjan käytännön työhön ja sen suunnittelemiseen [12]. Riippuu paljon myös opetettavasta asiasta ja opiskelijaryhmästä, millaiseen oppimiskäsitykseen liittyvä opetusmenetelmä on kulloinkin tarkoituksenmukainen. [14.]

2.4.1 Behavioristinen oppimiskäsitys

Behaviorismi oppimissuuntauksena on ollut vallalla pitkän ajan. Se alkoi 1. maailmansodasta ja päättyi 1960-luvulle. Behaviorismi rakentuu suoraan luonnontieteellisen käyttäytymisen tutkimuksen päälle. [5]. Behavioristisen käsityksen mukaan eläimet ja ihmiset oppivat samalla lailla [11]. Behavioristisessa oppimiskäsityksessä ihminen saa tietoa ympäristöstään empiirisesti. Käsityksen mukaan ihmisen tietoisuuteen ei voida nähdä, joten ainoa tapa saada tietoa ihmisestä on ulkoisen käyttäytymisen tutkiminen. Samoin on oppimisen kanssa: se toteutetaan myös oppijan ulkopuolelta. Lisäksi behavioristisessa oppimiskäsityksessä nähdään ihmisen olevan vailla vapaata tahtoa, jolloin häntä voi ohjailla ulkopuolelta tulevien ärsykkeiden avulla. [12.]

Oppimista voidaan siis säädellä ulkoisesti. Tämä määrää sen, miten opetusjärjestelyt tehdään. Koska oppiminen on käyttäytymisen muutosta, tavoitteet opetusjärjestelyissä määritellään juuri käyttäytymisen muutoksen kautta. [15.]

Behavioristisessa oppimiskäsityksessä oppijan roolina on olla passiivinen tiedon vastaanottaja. Opettajan tehtävä taas on jakaa tietoa, annostella sitä ja olla tiedon lähde.

Opettaja toimii oppimisprosessin ohjaajana ja hallitsijana. Tieto on ositettu mahdollisimman pieniin osiin, behaviorismiin oleellisena osana kuuluvan mekaanisen järjestelmällisyyden mukaan. Ärsyke-reaktio-kytkentää soveltaen opettaja antaa oppijalle ärsykeen, esimerkiksi tehtävän tai tentin. Jokaisen osan jälkeen annetaan palautetta, joka annetaan heti. Tämän oppija mieltää palkkioksi. Oppijan motivaatio oppimiseen syntyy tästä palkitsemisesta. Oppimisprosessi on silloin onnistunut, kun oppija reagoi ärsykkeeseen opettajan suunnitteleamalla tavalla. On tärkeää, että oppijat näkevät suorituksensa myös jälkeinpäin, mikä vahvistaa oppimista. B. F. Skinner (1904–1990) ajatteli, että ihmisen käyttäytymistä voidaan ohjata positiivisen palautteen perusteella. Heikoistakin suorituksista löytyy asioita, joista voi antaa positiivista palautetta. [11, 12, 13.]

Behavioristisessa oppimiskäsityksessä oletetaan, että liiallinen sosiaalinen vuorovaikutus oppijoiden kesken pahasta, koska opettajan auktoriteetti tällöin heikkenee. Luokkaopetus taas mahdollistaa opettajan auktoriteetin säilymisen. Tämä muistuttaa arkikäytystämme oppimisesta.

Vaikka behaviorismi onkin jo nähnyt aikaa, sillä on vielä käyttöä. Esimerkkinä voi toimia vaikka kielten opetuksessa käytettävä tietokoneohjelma. Ohjelma näyttää oppijalle kysymyksen (ärsyke). Oppija vastaa kysymykseen (reaktio). Kun oppilas saa vastauksen oikein, annetaan positiivinen palaute ("Oikein meni, hienoa!"). Oppija mieltää positiivisen palautteen palkinnoksi. Oppijan motivaatio kasvaa palkinnon saamisen myötä. Prosessia toistetaan, kunnes saadaan opettajan suunnittelema tulos, tässä sanavaraston laajeneminen. [15]

2.4.2 Humanistinen oppimiskäsitys

Humanistinen oppimiskäsitys juontaa juurensa humanistiseksi psykologiaksi kutsutusta teoriasta. Suuntauksen keulahahmoja olivat esimerkiksi Abraham Maslow (Maslow'n tarvehierarkia) sekä Carl Rogers (yksi psykoterapian tutkimuksen käynnistäjistä). [16]

Humanistisen psykologian mukaan minän/itsen käsite on keskeisessä asemassa. Teorian mukaan minä/itse löytyy vain ihmisiltä, kun taas behaviorismin mukaan ihmistä ei voi erottaa eläimestä. [11, 15.]

Humanistisen oppimiskäsityksen mukaan oppiminen rakentuu vaiheittain. Ensin toimitaan ja kerätään kokemuksia. Seuraavaksi reflektoidaan näitä kokemuksia. Kolmannessa vaiheessa pyritään käsitteellistämään reflektoinnin tulokset. Tulosten pohjalta pyritään luomaan uusi teoria. Viimeiseksi kokeillaan uutta teoriaa käytäntöön. [15.]

Humanistinen oppimiskäsitys painottaa yksilön omaa käyttäytymisen säätelyä. Vastavasti opetuksessa painotetaan yksilön omia henkilökohtaisia kokemuksia. Oppimisessa on tärkeää, että oppijan omaa kasvua ja itseohjautuvuutta tuetaan. Tällä tarkoitetaan sitä, että oppija itse kokee, kuinka hyvin hän on saavuttanut asetetut tavoitteet ja mitä osa-alueita hänen tulee kehittää. [15.]

2.4.3 Kognitiivinen oppimiskäsitys

Kognitiivinen oppimiskäsitys on jatkoa behavioristiselle oppimiskäsitykselle. Kognitivismissa tieto katsotaan koostuvan oppijan itse tekemistään tulkinnoista, skeemoista (aiempaan kokemukseen pohjautuvat tiedot ja taidot) sekä konstruoinneista (muodostetut tietorakenteet). Oppija mielletään tiedon prosessoijaksi. Oppija voi asettaa itselleen tavoitteita ja siten vaikuttaa omaan oppimiseensa. [15.]

Kognitivismissa oleellista on yksilön oppiminen. Oppija kerää ja rakentaa tietoa omana henkisenä prosessinaan. Reflektio on oleellinen osa oppimista. [12.]

Kognitivismissa oppiminen koostuu 1) oppijan muodostamasta kuvasta ulkoisesta maailmasta, 2) oppijan vertaamisesta tästä kuvasta aikaisempaan empiriaan sekä 3) oppijan pyrkimyksestä löytää selityksiä opittavalle ilmiölle. [15.]

Kognitivismissa painotetaan oppilaan omaa tavoitehakuista toimintaa. Oppija saa informaatiota, jota hän käsittelee itse määräämiensä tavoitteidensa mukaan. Behavioristiseen oppimiskäsitykseen verrattuna kognitivismissa nähdään oppija ”ymmärtävänä, ajattelevana ja ympäristöä jäsentävänä yksilönä”. [11.]

Oppijan itselleen asettamansa tavoitteet syntyvät oppijan aiemmasta empiriasta. Uusi informaatio lisätään aiempaan. Uuden informaation käsittely ei ala alusta, koska aiempi tieto vaikuttaa siihen. Oppijan oma toiminta saa aikaan oppimisen. [11.]

2.4.4 Konstruktiivinen oppimiskäsitys

Konstruktivismissa on eri suuntauksia, joiden ominaisuudet vaihtelevat. Niitä yhdistää näkemys siitä, että informaatio on yksilön itse rakentamaa ja se, että oppija on aktiivinen toimija, empirian ja informaation tulkitsija aiemmin oppimansa pohjalta. [15.] Konstruktiivismi ei siis ole yksi tarkkaan rajattu oppimiskäsitys, vaan enemmänkin sateenvarjoparadigma, joka sisältää useita näkemyksiä. [14.]

Konstruktivismissa oppimisen käsitetään olevan oppijan itsenäistä tiedon käsittelyä, konstruointia. [14.] Konstruktiivismissa tieto käsitellään oppijan omien informaationkäsittelymekanismien kautta. [13.]

Oppija nähdään aktiivisena tiedon prosessoijana. Tietoa ei voi suoraan siirtää oppijalle, vaan oppija itse konstruoi informaation. Oppijan oma aktiivisuus, informaation prosessointikyvyt sekä metakognitiiviset taidot korostuvat. [17.] Pohjana ovat oppijan aiemmin opittu sekä aikaisempi empiria. Nämä määrittelevät perustan, jonka mukaan hän yrittää löytää, painottaa ja tulkita informaatiota. [14.] Oppiminen perustuu aiemmalle tiedolle. Oppija lisää kokemukseensa uutta tietoa ja esimerkiksi mielikuvia sekä jäsentää ajatuksensa uudelleen. Koska kenelläkään oppijalla ei ole samaa oppimishistoriaa, samat asiat tulkitaan todennäköisesti eri tavoilla. [14.] Opettajan rooli on ohjaaja, enemmän kuin tiedon siirtäjä. Tavoitteet määritellään yhdessä oppijan kanssa. [13.]

Opetustilanteessa (saman) tiedon erilaiset tulkinnat muodostavat haasteen, joka on ratkaistava. Opettajan on siis valikoitava tähän sopivat opetusmenetelmät. Konstruktiivismissa opettajan tehtäväkuva on lähinnä oppijan kannustaminen hankkimaan itsenäisesti tietoa sekä hänen tukeminen tämän prosessin aikana. [14.]

Konstruktivismissa nähdään, että oppiminen on sidoksissa tilanteeseen. Lisäksi sosiaalisen vuorovaikutuksen katsotaan olevan oleellinen osa oppimista. Sosiaalisen vuorovaikutuksen menetelmät ovatkin yleisesti käytettyjä. Yhdessä pohtiminen on opetuksen ydintä. [14.]

Koska oppijan oma aktiivisuus on kriittistä hyvän oppimistuloksen kannalta, pyritään sitä lisäämään. Aktiivisuutta kasvattaa erityisesti motivaatio, joka taas on edellytys oppimiselle. [14, 17.] Motivaatio lisääntyy silloin, kun oppija ymmärtää opittavan asian tärke-

den. [13] Aktiivisuutta lisäävät myös riittävän haastavat tehtävät sekä oppijan omat tavoitteet oppimisprosessissa. Lisäksi tärkeitä ovat myös metakognitiiviset taidot. Kun oppija pystyy käyttämään näitä taitoja, hän voi reflektoida ja muokata omaa toimintaansa. Tämä kohentaa oppimisen laatua. [17.]

Konstruktivisessa oppimisenäkemyksessä ei painoteta niinkään oppimisen tulosta, vaan enemmänkin oppijan läpi käymää prosessia. Ulkoa opettelu ei ole niin tärkeää kuin ymmärtäminen. Oppijan omaa reflektointia korostetaan. Koska esim. behaviorismissa arvioidaan lopputulosta ja konstruktivismissa taas oppimisprosessia, oppimisen arvioiminen on haastavampaa. Lisäksi konstruktivismissa arviointia tekevät myös oppijat, ei pelkästään opettaja. [13.] Oleellista on, että tiedon jäsentyneisyyden laatu sekä tietojen monipuolisuus määrittävät, kuinka oppija voi soveltaa tietoa tulevaisuudessa. [17.]

2.4.5 Ongelmakeskeinen oppiminen (Problem-based learning, PBL)

Ongelmakeskeinen oppiminen tarkoittaa sitä, että opiskelijat yrittävät itse määrittellä ongelman, analysoimaan sitä sekä ratkaista sen. Tässä korostuu opiskelijoiden oma aktiivisuus. Tarkoitus on myös, että opiskelijat toimivat itsenäisesti. [18.]

Ongelmakeskeistä oppimista käytettäessä opettajan rooli on eri kuin perinteisessä opetuksessa. PBL-menetelmässä opettaja toimii enemmänkin fasilitaattorina, vähemmän auktoriteettina. [18.]

Tämä mahdollistaa käytännönläheisen oppimisen, jossa korostuvat tosielämää vastaavat ongelmat. Pelkän tiedollisen kehittymisen sijaan Problem-based learning-metodissa oppija kehitty myös muun muassa ongelmanratkaisutaidoiltaan. Lisäksi esimerkiksi oppimisasenteet muuttuvat väistämättä. [18.]

PBL:n arvioitiin sopivan kurssin opetusmetodiksi, koska tarkoituksena oli juuri saada opiskelijat itse tekemään. Ideana oli antaa opiskelijoille tietty tavoite, jonka suorittamiseen heidän itsensä piti löytää keinot. Tavoitteen tarkka määrittely jäi myös opiskelijoiden työksi. Opettajan rooliksi päätettiin fasilitaattori, joka auttaa tarvittaessa kädestä pitämällä ja jakaa välillä myös uutta tietoa.

3 Opinnäytetyön tulokset

3.1 Robottikurssin kuvaus

Robottikurssi järjestettiin 7. - 11.8.2017. Paikaksi valittiin Metropolian Leppävaaran kampus (Vanha maantie 6, 02650 Espoo). Kurssin kestoksi määriteltiin 24 tuntia, joka jakaantui seuraavasti:

- maanantai 7.8 klo: 9.00–15.00
- tiistai 8.8. klo 9:00 –15:00
- keskiviikko 9.8 9:00–15:00
- torstai 10.8 klo 14:00–18:00
- perjantai 11.8 klo 14:00–18:00.

Päivittäin pidettiin yksi tauko, kestoaltaan 45–60 minuuttia. Kurssin opettajana toimi Hans Hokka, vastuuopettajana yliopettaja Päivi Haho. Hansin vastuulle kuuluivat robottien, läppärien ym. roudaus, tilojen varaaminen, robottien koodaamisen opetus, ryhmien tukeminen ja ohjaus sekä oppimisprosessin valvonta.

3.1.1 Osallistujat, materiaalit ja metodit

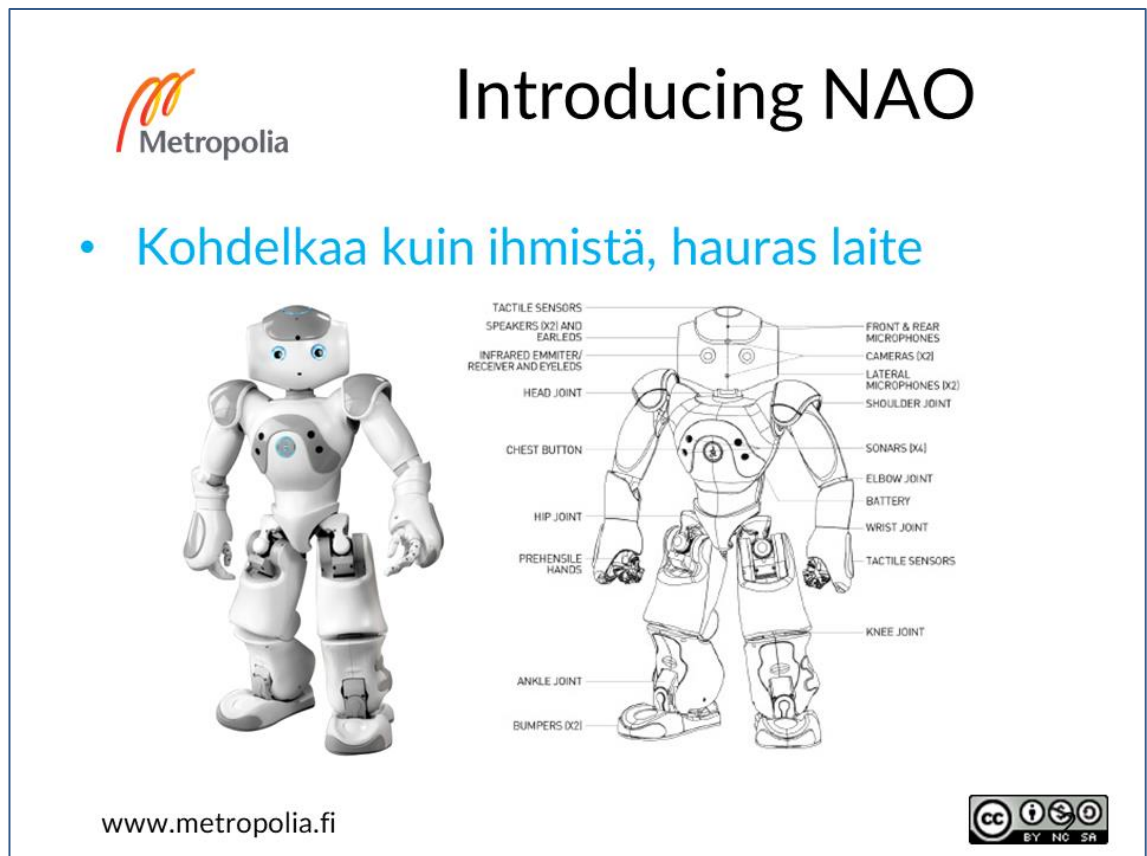
Kurssille ilmoittautuneita oli 12. Saapuneita oli seitsemän, joista yksi tyttö, loput poikia. Kaikki olivat lukiolaisia, iältään 17–19. Kolmella oli ohjelmointikokemusta Pythonista, yhdellä C++:sta. Ohjelmointikokemus ei näyttänyt vaikuttavan oppimisen vaikeusasteeseen.

Jokaisella ryhmällä oli käytössään kannettava tietokone, Nao sekä tarvittavat lisätarvikkeet. Tietokoneelle oli valmiiksi asennettu Choregraphe.

Opetusmateriaalina käytettiin valmista Choregraphella tehtyä esimerkkiohjelmaa, PowerPoint-slidejä, valmiita moduuleita Choregrapheen, Boston Dynamicsin videoita robottien tämänhetkisestä kehitystasesta sekä Naon käytöstä aiemmin kuvattuja videoita.

3.1.2 Kurssin sisältö

Kurssi aloitettiin esittäytymisellä, joka jälkeen tutustuttiin Naoon. Tutustumisessa käytiin läpi Naon rakenne, sensorit, moottorit, kamerat, äänet, nivelet sekä varoimenpiteet (kuva 8).



Kuva 8. Naon tutustuminen.

Fyysisten elementtien jälkeen keskityttiin Choregraphen käyttöön. Ensin esiteltiin Choregraphe ja sen perusosa, Box (kuva 9). Tämän jälkeen voitiin siirtyä tutkimaan Choregraphen alkunäkymää. Näkymän elementit käytiin läpi, kaksi paneelia vielä tarkemmin (kuva 10).



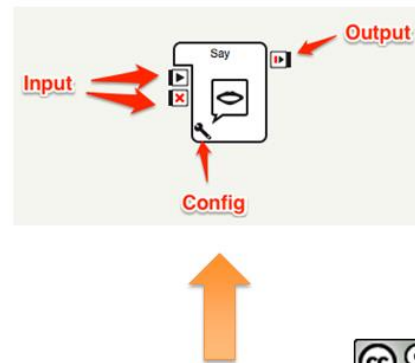
Choregraphe Softwa



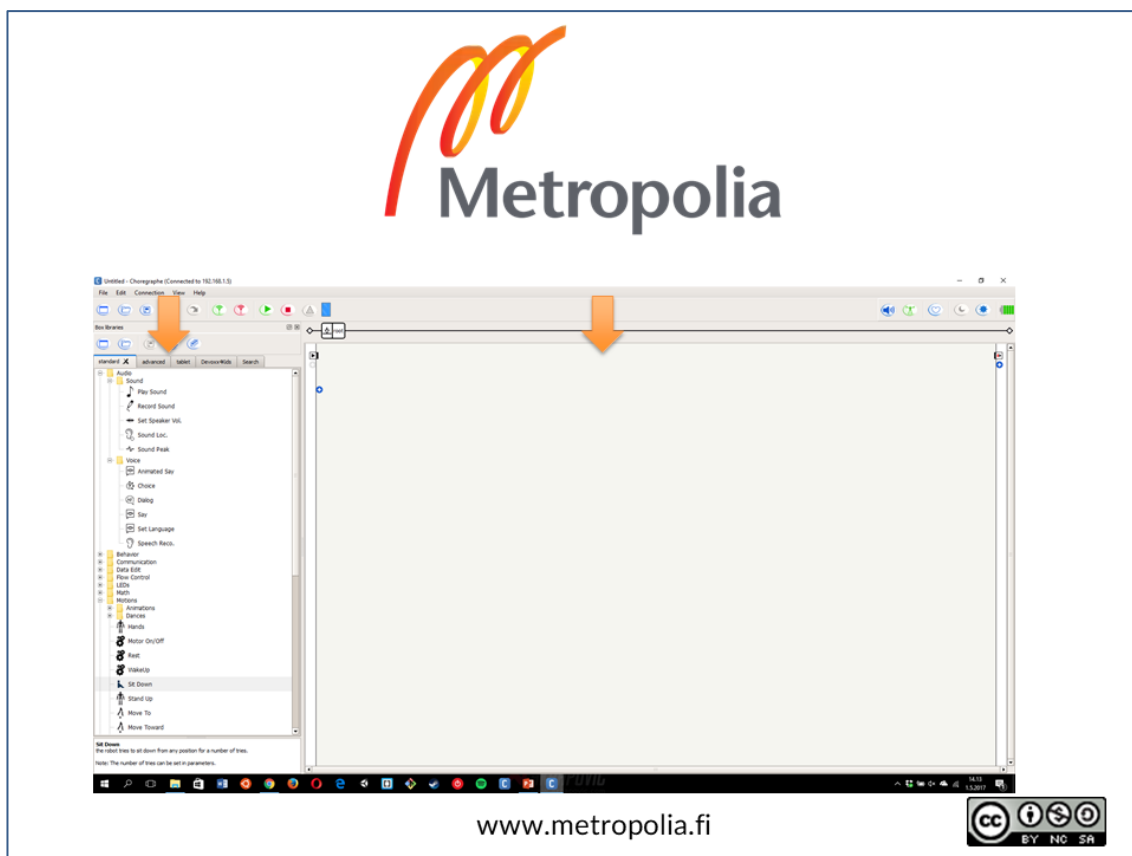
- NAO Visual Programming
 - Allows users of NAO to create and edit simple movements and interactive behaviors.
- Demonstration



www.metropolia.fi

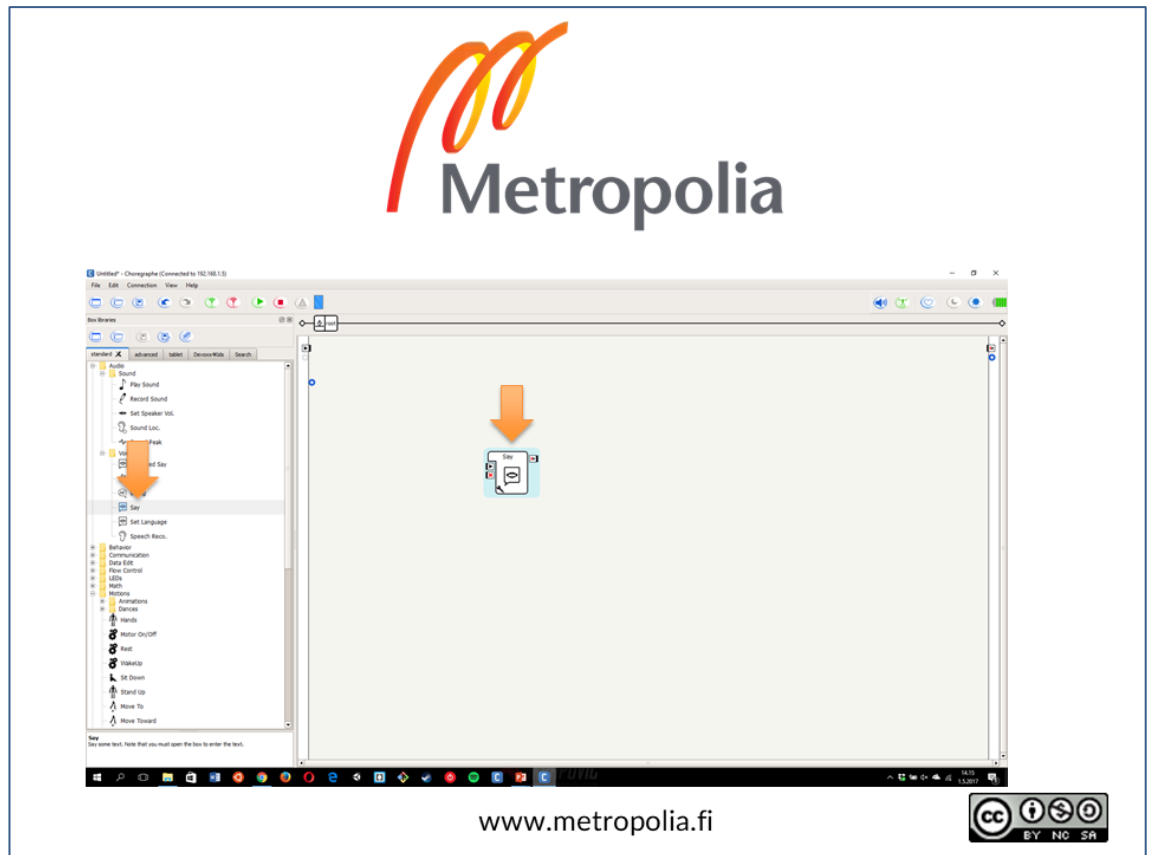


Kuva 9. Choregraphen Moduulin esittely.

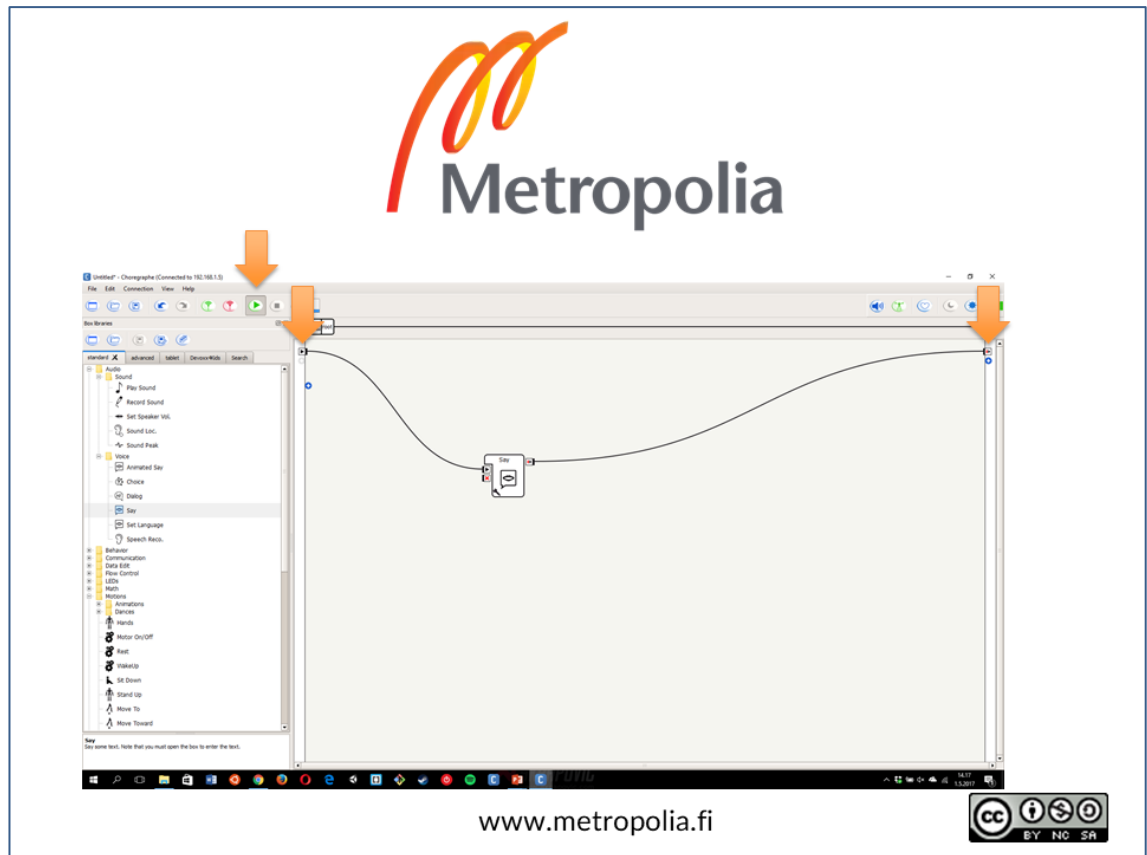


Kuva 10. Choregraphen työtilan elementit.

Kun työtila sekä tärkeimmät paneelit olivat tulleet tutuksi, siirryttiin ensimmäisen Boxin luomiseen (kuva 11). Kun Boxi oli luotu, se kytkettiin aloitukseen ja lopetukseen. Ohjelma käynnistettiin ensimmäistä kertaa (kuva 12).

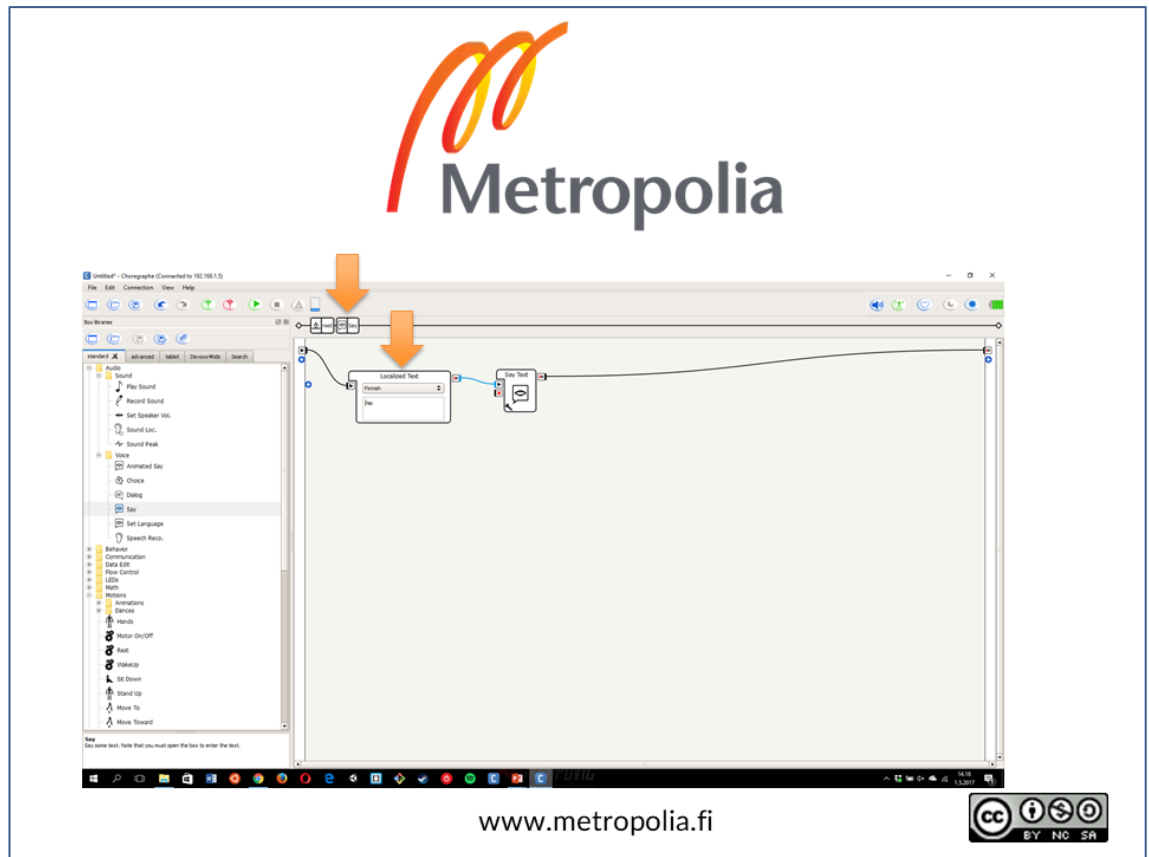


Kuva 11. Moduulin luominen työtilaan.

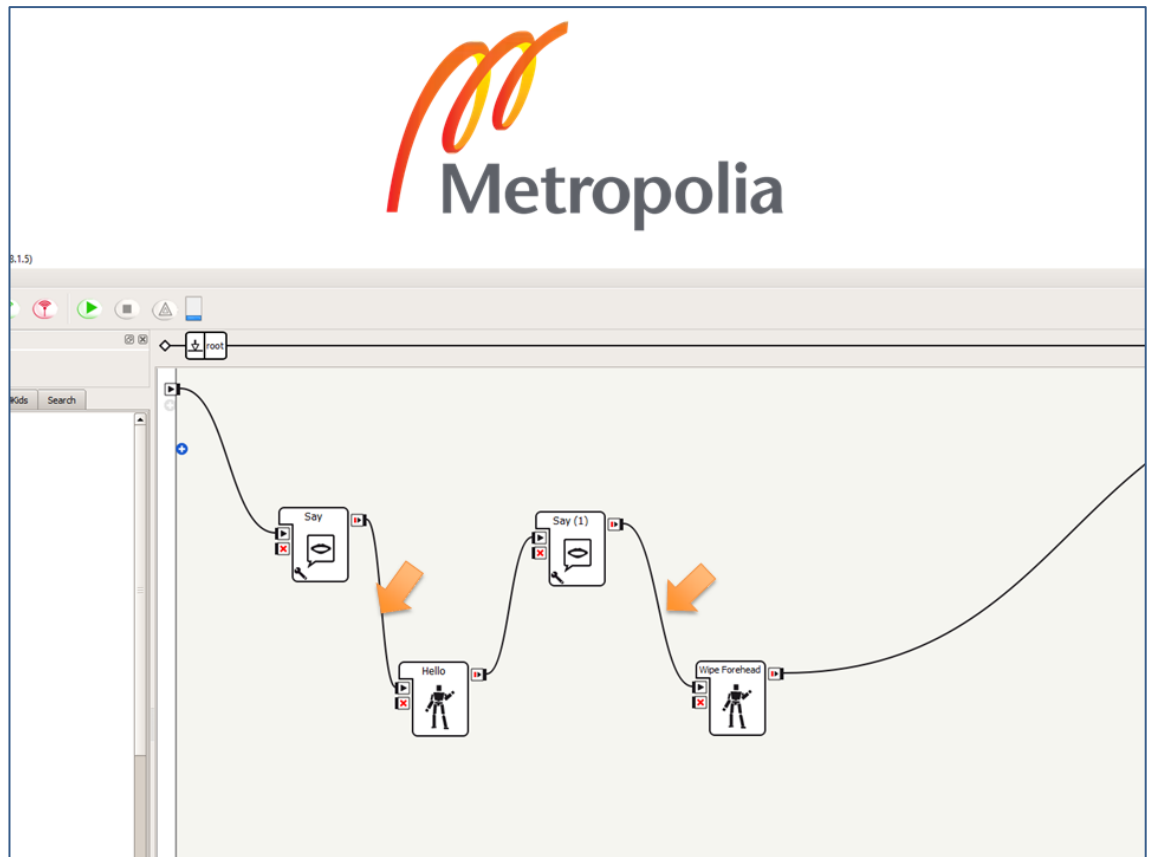


Kuva 12. Moduulin kytkeminen aloitukseen ja lopetukseen. Ohjelman käynnistyminen.

Kuvassa 13 näytettiin, kuinka Boxiin zoomataan. Layerit tulivat tässä kohdassa ensimmäistä kertaa mukaan. Viimeiseksi koottiin kaikki yhteen ja nähtiin, miten valmis ohjelma muodostuu Boxeja ketjuttamalla (kuva 14).

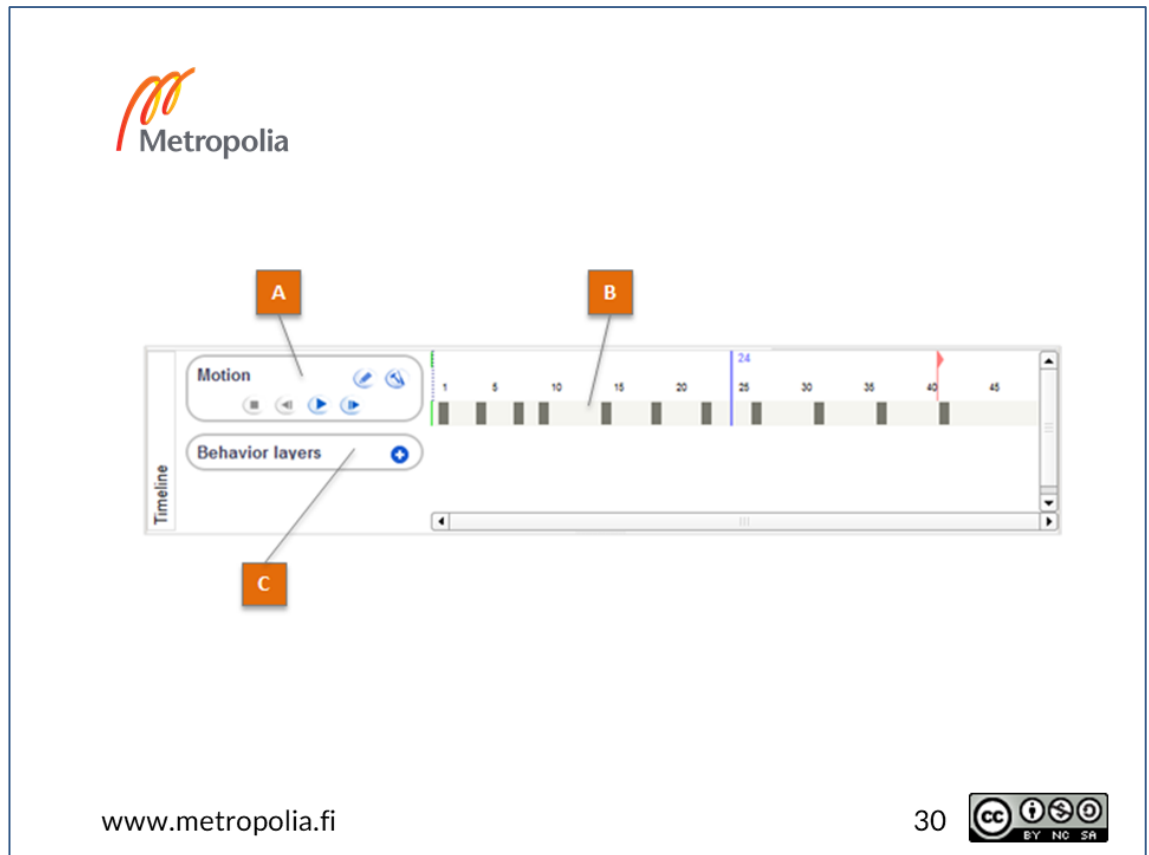


Kuva 13. Moduulin zoomaaminen, layereiden esittely.

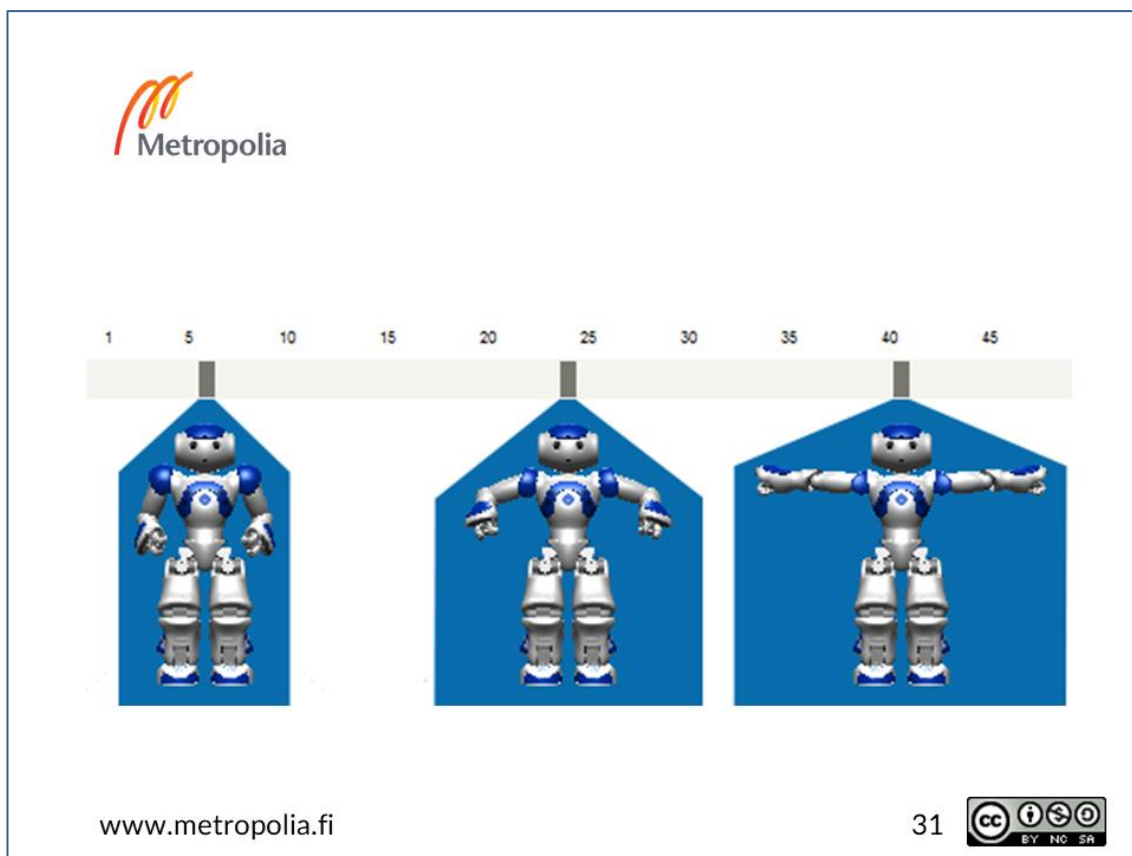


Kuva 14. Moduulien ketjuttaminen.

Ensimmäisen päivän palautteen perusteella toisena päivänä siirryttiin animaatioiden luomiseen Choregraphella (kuva 15 ja 16). Animoinnin omaksumisen jälkeen oppilailla oli hyvä ymmärrys Naon ohjelmoinnista sekä kyky selvittää asioita itsenäisesti Naon ohjekirjaa käyttäen.



Kuva 15. Johdatus animointiin, Timelinen käyttö.

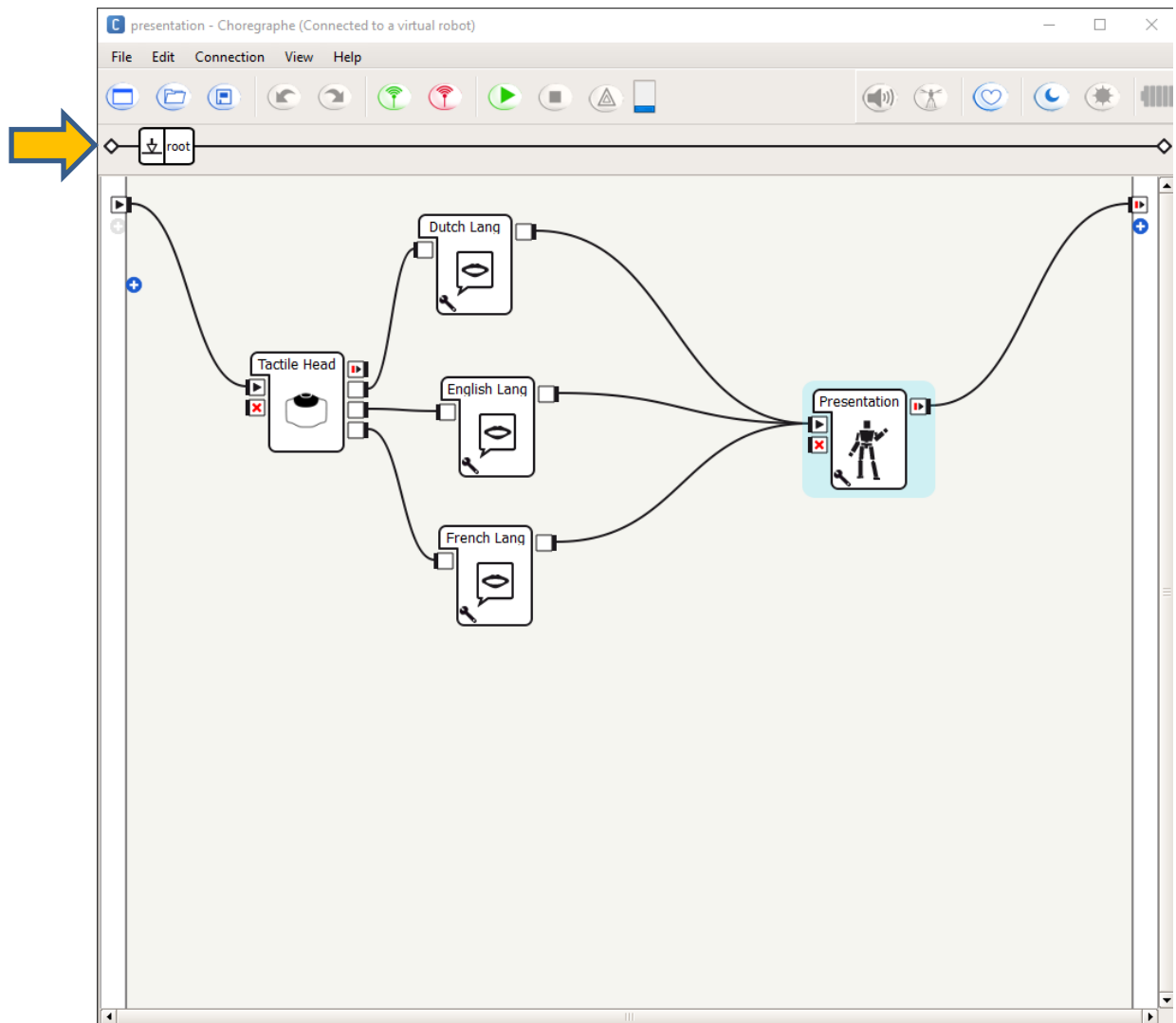


Kuva 16. Timelinen käyttö.

Kun perusasiat olivat hallussa, voitiin aloittaa omien prototyyppien suunnittelu. Päivi Haho veti oppilaille puolen päivän mittaisen suunnitteluosion käyttäen palvelumuotoilua. Jokainen ryhmä keskittyi tästä eteenpäin Naon ohjelmointiin oman prototyypin kehittämiseksi.

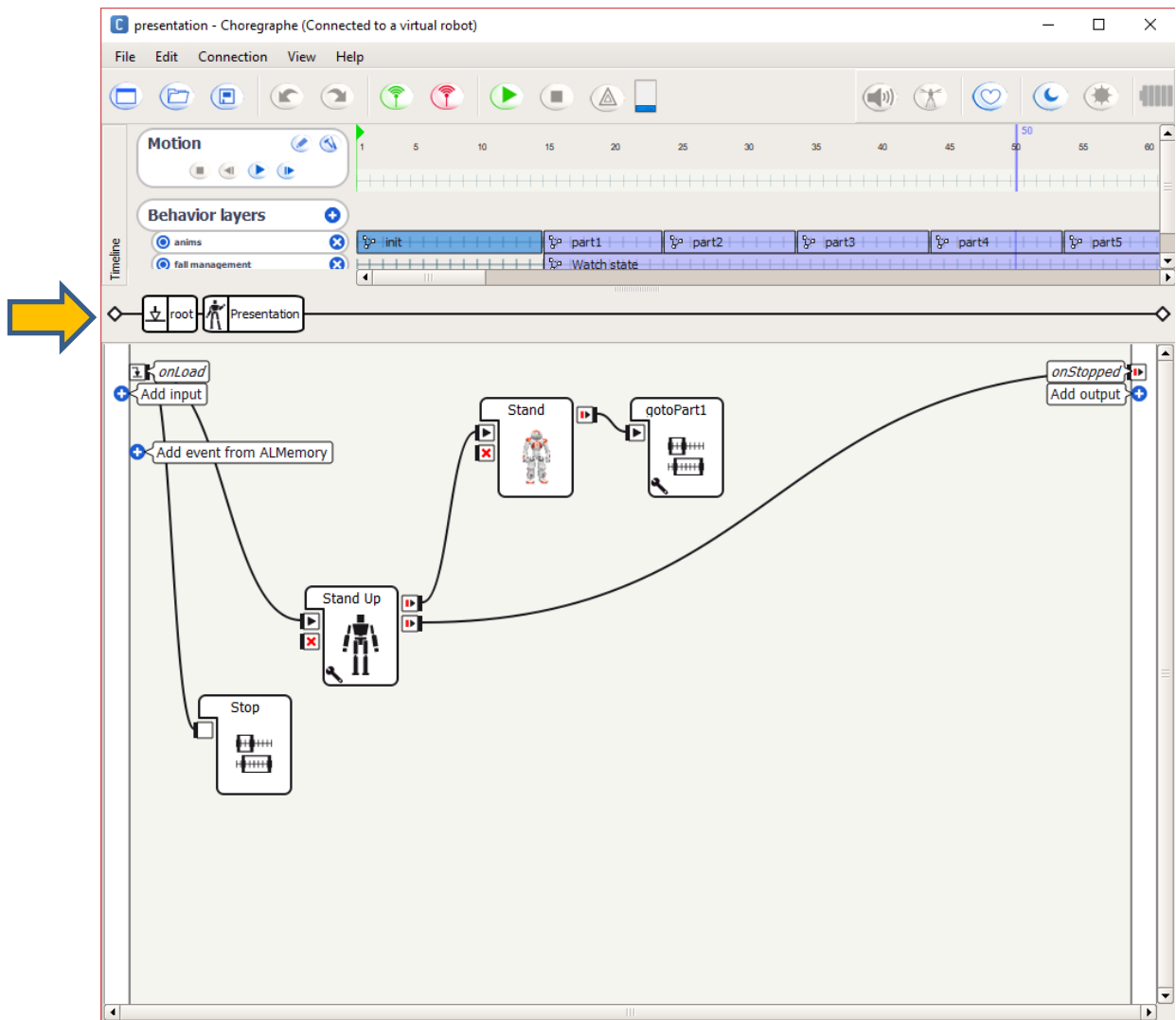
Kun prototyypit olivat valmiit, ryhmät lähtivät esittelemään niitä kentälle, ulkopuolisille ihmisille. Kerätyn palautteen perusteella ryhmät tekivät esityksen, jonka esittivät kaikille yhteisesti.

Kun Naon sekä Choregraphen perustoimintoihin oli perehdytty, oppijat saivat tutkittavakseen valmiin malliohjelman (kuva 17).



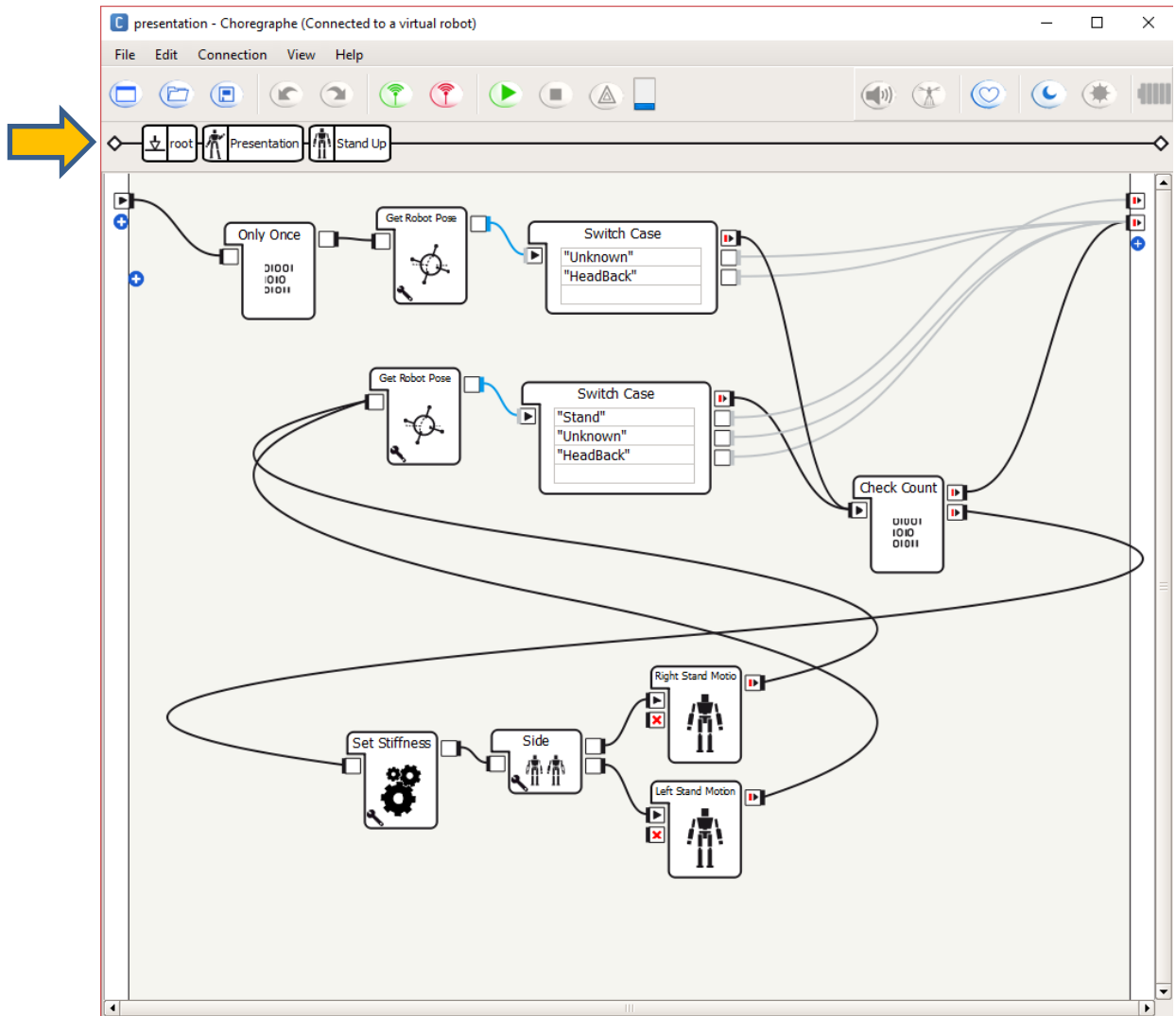
Kuva 17. Malliohjelma tutkittavaksi.

Malliohjelman "Presentation"-Boxista päästiin alemmalle tasolle eli layerille (kuva 18).



Kuva 18. Malliohjelman layereiden tutkiminen.

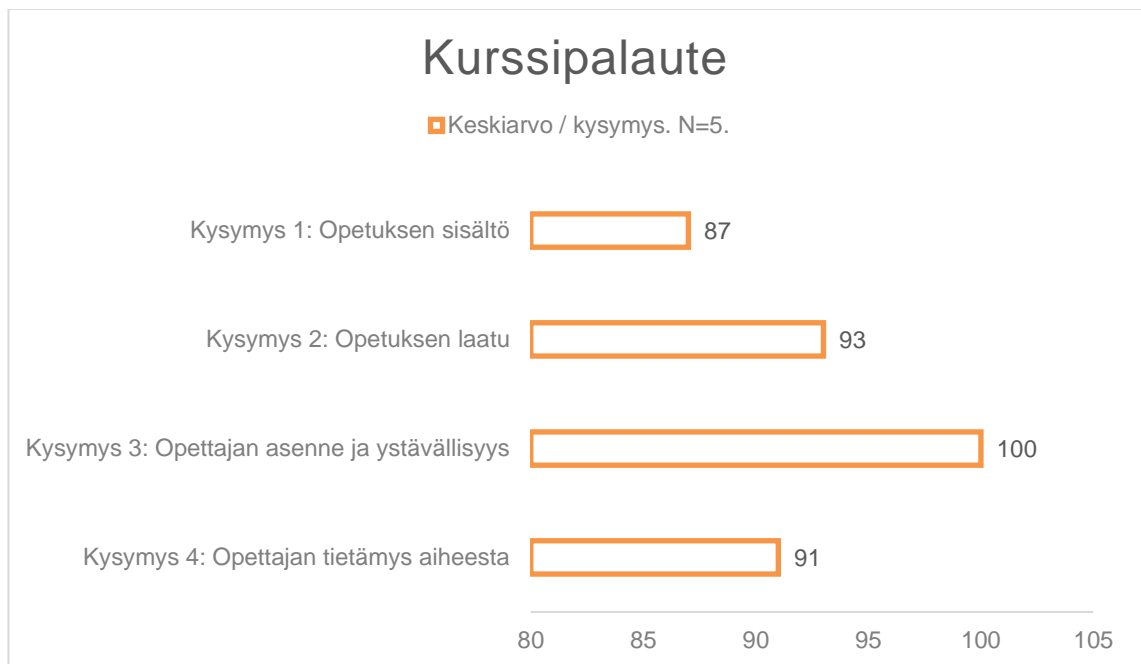
”Stand Up”-Boxiin zoomaamalla päästään vielä yksi layeri alemmas (kuva 19).



Kuva 19. Layereihin zoomaaminen.

3.2 Palaute kurssista

Kurssipalaute kerättiin SurveyMonkey-palvelussa. Jokainen osallistuja voi vastata kysymyksiin nimettömänä. Ensimmäiset neljä kysymystä keräsivät palautteen opetuksen osa-alueiden laadusta. Vastaus voitiin antaa asteikolla 0...100 (**Virhe. Viitteen lähde ei löytynyt.**). Kysymykset viisi ja kuusi olivat avoimia, jotta kurssilaisilta saatiin kerättyä myös hieman kvalitatiivista palautetta (kuva 20).



Kuva 20. Kurssipalautteen kvantitatiivinen osuus.

Kysymys 5: Mikä kurssissa oli parasta?

- onnistuminen aina välillä
- Nao-robotti
- Kun sai robotin tekemään haastaviakin asioita
- Ohjelmointi ja robotilla "leikkiminen"
- Ohjelmointi

Kysymys 6: Mitä toivoisit, että kurssissa muutettaisiin tai kehitettäisiin?

- Enemmän aikaa esim. 2vk.
- Olisi ollut kiva myös tutustua robotin rakenteeseen, toimintaperiaatteeseen ja mekaniikkaan
- Kestävämpi robotti, ettei pelota tuhota 5000€ arvonen juttu
- Kestävämpi robootti

Kuva 21. Kurssipalautteen kvalitatiivinen osa

4 Yhteenveto

Tässä insinööriyössä perehdyttiin robotiikan opetukseen 16–19-vuotiaille. Työn tarkoituksena oli suunnitella ja toteuttaa kurssi, joka mahdollistaa oppijoiden tutustumisen tiettyyn robottiin sekä ohjelmointikieleen. Kurssia varten valittiin robotti, ohjelmointikieli sekä opetusmetodi. Kurssi pidettiin, siitä kerättiin palaute sekä kaikki tuotettu materiaali uutta vastaavaa kurssia varten.

Työn tavoitteet saavutettiin halutulla tavalla, niin omalta, työn tilaajan sekä oppijoiden kannalta. Kurssin konsepti ja luodut materiaalit jäivät työn tilaajan käyttöön, ja niitä on jo hyödynnetty. Suurimmaksi haasteeksi muodostui robottien toimivuus. Onneksi robotteja oli ylimääräisiä, jolloin useimmiten ylikuumentunut tai muuten häiriöinen laite saatiin suoraan vaihdettua. Opiskelijapalautteesta voitiin päätellä, että he haluaisivat mennä robotiikassa ja ohjelmoinnissa syvemmälle. Tämä ei valitettavasti ollut annetun ajan puitteissa mahdollista.

Insinööriyö oli kokonaisuudessaan erittäin mielenkiintoinen projekti. Varsinkin opettaminen oli täysin uutta. Kurssin toteuttamiseen tarvittiin tietoja laajalta alalta, tiedon omaksuminen oli mielenkiintoista. Jännitysmomenttia aiheutti se, ettei vastaavaa kurssia ollut aiemmin järjestetty.

Lähteet

- [1] J. Hietikko, "Robottiikan työelämälähtöisen opintojakson suunnittelu Vamiaan," Vaasa, 2017.
- [2] L. Salminen ja R. Suhonen, "Oppiminen ja oppimismenetelmät ja niiden ammatillisen kehittymisen tukena," Hämeenlinna, 2008.
- [3] Jyväskylän ammattikorkeakoulu, "Oppimiskäsitykset," Jyväskylä.
- [4] S. Pitkänen, "Tieto- ja viestintäteknikan käytön oppaita ja oppimateriaaleja," 2010.
- [5] H. Jelonon, "Pedagogisen ajattelun kehittyminen ammatillisen opettajankoulutuksen aikana," Jyväskylä, 2008.
- [6] P. Kaipainen, "Oppimiskäsityksistä sopiviin opetusmenetelmiin," Jyväskylä, 2008.
- [7] Jyväskylän yliopisto, "Pedagogiset mallit," [Online]. Available: <https://koppa.jyu.fi/avoimet/mit/tietotekniikan-opetuksen-perusteet/Opetusmenetelmista-ja-lahestymistavoista/pedagoginen-malli>. [Haettu 2 Huhtikuu 2018].
- [8] Suomen virtuaaliyliopisto, "Oppimisen teoriasta tukea tieto- ja viestintäteknikan pedagogiseen käyttöön," 11 Huhtikuu 2018. [Online]. Available: http://tievie oulu.fi/verkkopedagogiikka/luku_3/konstruktivismi.htm.
- [9] "Yvistä energiaa yrittäjyyskasvatukseen," [Online]. Available: <http://www.yvi.fi/sanakirja/262-ongelmakeskeinen-oppiminen-problem-based-learning>. [Haettu 13 Elokuu 2018].
- [10] "Robots e-Manual," 10 Tammikuu 2018. [Online]. Available: <http://turtlebot3.robotis.com/en/latest/features.html#worlds-most-popular-ros-platform>.
- [11] "Arduino Store," 11 Tammikuu 2018. [Online]. Available: <https://store.arduino.cc/arduino-ctc-kit-101>.
- [12] "Lego Shop," 11 Tammikuu 2018. [Online]. Available: <https://shop.lego.com/en-FI/LEGO-MINDSTORMS-EV3-31313?p=31313&track=checkprice#shopxlink>.

- [13] "Lego Mindstorms EV3," 12 Elokuu 2018. [Online]. Available: <https://www.lego.com/fi-fi/mindstorms/products/mindstorms-ev3-31313>.
- [14] "Aldebaran Python SDK," 10 Maaliskuu 2018. [Online]. Available: <http://doc.aldebaran.com/2-1/dev/python/index.html>.
- [15] "Aldebaran Choregraphe User Guide," 2 Huhtikuu 2018. [Online]. Available: <http://doc.aldebaran.com/2-1/software/choregraphe/index.html>.
- [16] "Aldebaran Programming Documentation," 13 Elokuu 2018. [Online]. Available: http://doc.aldebaran.com/2-1/dev/programming_index.html#dev-index.
- [17] "Aldebaran Programming Example," 12 Helmikuu 2018. [Online]. Available: <http://doc.aldebaran.com/2-1/dev/cpp/examples/core/helloworld/example.html#cpp-examples-helloworld>.
- [18] "Aldebaran Nao Documentation," 13 Elokuu 2018. [Online]. Available: http://doc.aldebaran.com/2-1/family/robots/index_robots.html.
- [19] "Aldebaran Using Choregraphe," 10 Maaliskuu 2018. [Online]. Available: http://doc.aldebaran.com/2-1/software/choregraphe/choregraphe_first_steps.html.

Liitteet

Liite 1. Roboviikon esite



KAUNIAINEN



Helsingin kaupunki

Roboviikko

7. - 11.8.2017

HEI SINÄ 16 -19 v. NUORI, KIINNOSTAAKO ROBOTIT JA OHJELMOINTI?

PK-seudun koulut järjestävät yhdessä Roboviikon 7. - 11.8.2017 Metropolian Leppävaaran Campuksella, Vanha Maantie 6. Kurssin aikana kehität Ideasta Innovaatioksi -palvelumuotoilun keinon ratkaisuja, joissa hyödynnetään Nao-robotteja. Viikon aikana tutustut Nao-robotin visuaaliseen ohjelmointikieleen, ohjelmoinnin erusteisiin, kuten liikkeiden tuottamiseen, puheen tunnistamiseen ja hahmontunnistukseen. Kurssilla kehität yhdessä ryhmän kanssa palveluinnovaatioita ja ohjelmoit Nao-robottia Choreographe-ohjelmointikielellä.

Kohderyhmä:

Kurssi on tarkoitettu PK-seudun lukiolaisille, peruskoulun päättäneille ja Omnian ja Varian opiskelijoille.

Arviointi ja kurssisuoritus:

Lukiolaiselle 1 lukiokurssi, hyväksiluetaan lukion soveltaviin kursseihin 8S/H). Kurssitodistus edellyttää säännöllistä osallistumista sekä yhdessä sovittujen tehtävien suorittamista.

Ilmoittaudu 1.8. mennessä tässä: <https://elomake.metropolia.fi/lomakkeet/19459/lomake.html>
Kurssille mahtuu 25 ensimmäiseksi ilmoittautunutta.

Aikataulu: Maanantai 7.8 klo: 9.00 - 15.00, tiistai 8.8. klo 9:00 - 15:00, keskiviikko 9.8 9:00 - 15:00, torstai 10.8 klo 14:00 - 18:00 sekä perjantai 11.8 klo 14:00 - 18:00

Yht. 26 tuntia



Liite 2. Kurssitodistus



ROBOKURSSI

Etunimi Sukunimi

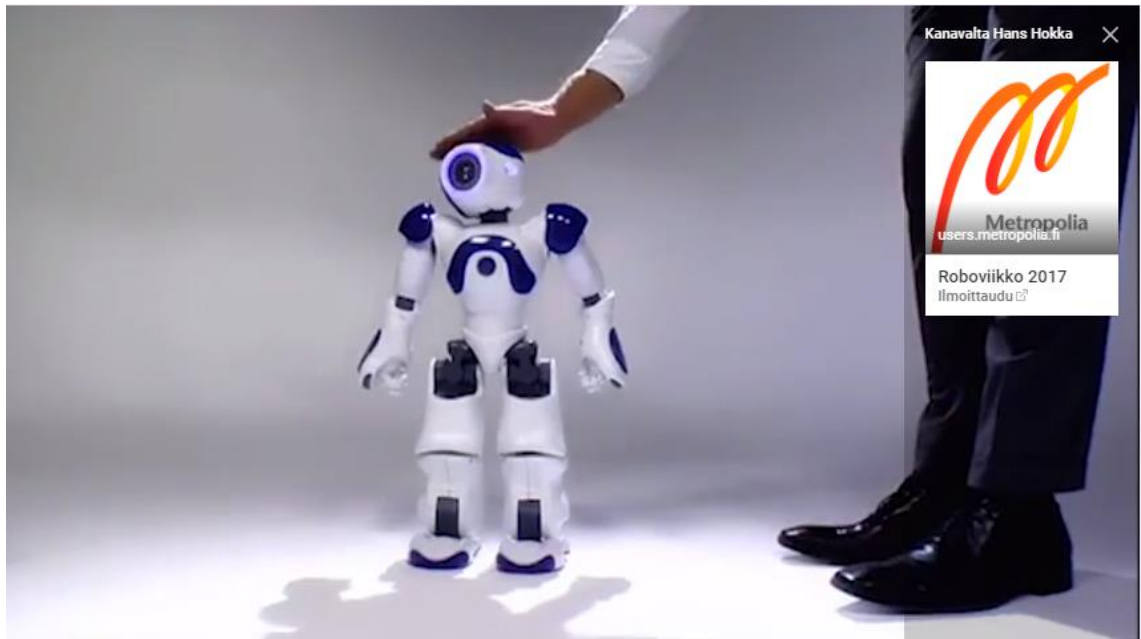
on suorittanut 2. – 5.5.2017 TET-harjoittelun Nao-robotin ohjelmoimisesta Choreography-ohjelmointikielellä. Harjoittelu sisälsi myös Ideasta Innovaatioksi -palvelumuotoilun perusteet. Kurssin kesto oli 16 tuntia.

Espoossa 5. toukokuuta 2017

Päivi Haho

Hans Hokka

Liite 3. Robokurssin YouTube-mainos



Liite 4. Luettelo kaikesta käytetystä ja tuotetusta materiaalista.

Kaikki käytetty materiaali löytyy nimellä robokurssi.zip osoitteesta bit.ly/robokurssi.

Nao\640px-Metropolia_Ammattikorkeakoulu_logo.svg.png

Nao\D4K-2013-NAO_Workshop-EN.pdf

Nao\D4K-2013-NAO_Workshop-EN.pptx

Nao\Devoxx4Kids.cbl

Nao\Devoxx4Kids_2013.zip

Nao\Devoxx4Kids_NAO_2_X.zip

Nao\esite.ai

Nao\esite.pdf

Nao\esite_sahkoinen.pdf

Nao\esite_sahkoinen_2.pdf

Nao\esite_sahkoinen_linkilla.pdf

Nao\esite_s„hk”inen.ai

Nao\filename.txt

Nao\files.txt

Nao\granilogo.jpg

Nao\granilogo.psd

Nao\granilogo_cmyk.png

Nao\helsingin logo cmyk.ai

Nao\helsingin logo cmyk.png

Nao\helsingin logo rgb.jpg

Nao\helsingin logo rgb.psd

Nao\lataus.jpg

Nao\Metropolia_Ammattikorkeakoulu_logo.svg

Nao\Metropolia_cmyk.png

Nao\metropolia_logo.jpg

Nao\metropolia_logo.png

Nao\Metropolia_RGB_B_neg.png

Nao\Metropolia_RGB_C.jpg

NaoNao-selallaan-style.png

NaoNaomark.zip

NaoNao_cmyk.png

Nao\P.O.S.T-Posteljoonirobo-esitelm„.gslides

Nao\P.O.S.T-robotti.gslides
Nao\presentation.crg
Nao\robokurssi.cbl
Nao\Robokurssin todistus.docx
Nao\Robokurssi_slides.pptx
Nao\robotti esitelm,,.gslides
Nao\Roboviikkojuttu.gslides
Nao\TET-prese.pptx
Nao\Turbiini_sininen_logo_cmyk.png
Nao\Turbiini_sininen_logo_jepgi.png
Nao\vantaan logo cmyk.png
Nao\vantaan logo rgb.png
Nao\vantaan logo rgb.psd
Nao\Devoxx4Kids_2013\Devoxx4Kids_2013\.DS_Store
Nao\Devoxx4Kids_2013\Devoxx4Kids_2013\.metadata
Nao\Devoxx4Kids_2013\Devoxx4Kids_2013\behavior.xar
Nao\Devoxx4Kids_2013\Devoxx4Kids_2013\Devoxx4Kids_2013.pml
Nao\Devoxx4Kids_2013\Devoxx4Kids_2013\manifest.xml
Nao\Devoxx4Kids_2013\Devoxx4Kids_2013\Aldebaran\choice_sentences.xml
Nao\Devoxx4Kids_2013\Devoxx4Kids_2013\behavior_1\behavior.xar
Nao\Devoxx4Kids_2013\Devoxx4Kids_2013\behavior_1\Aldebaran\choice_sentences.xml
Nao\Devoxx4Kids_NAO_2_X\devoxx4kids_workshop_1.DS_Store
Nao\Devoxx4Kids_NAO_2_X\devoxx4kids_workshop_1\devoxx4kids.DS_Store
Nao\Devoxx4Kids_NAO_2_X\devoxx4kids_workshop_1\devoxx4kids.metadata
Nao\Devoxx4Kids_NAO_2_X\devoxx4kids_workshop_1\devoxx4kids\devoxxTools.pml
Nao\Devoxx4Kids_NAO_2_X\devoxx4kids_workshop_1\devoxx4kids\manifest.xml
Nao\Devoxx4Kids_NAO_2_X\devoxx4kids_workshop_1\devoxx4kids\behavior_1\behavior.xar
Nao\Devoxx4Kids_NAO_2_X\devoxx4kids_workshop_1\devoxx4kids\behavior_1\Aldebaran\choice_sentences.xml
Nao\Devoxx4Kids_NAO_2_X\devoxx4kids_workshop_1\Devoxx4Kids_2013.DS_Store
Nao\Devoxx4Kids_NAO_2_X\devoxx4kids_workshop_1\Devoxx4Kids_2013.metadata
Nao\Devoxx4Kids_NAO_2_X\devoxx4kids_workshop_1\Devoxx4Kids_2013\behavior.xar

Nao\Devoxx4Kids_NAO_2_X\devoxx4kids_workshop_1\Devoxx4Kids_2013\Devoxx4Kids_2013.pml
Nao\Devoxx4Kids_NAO_2_X\devoxx4kids_workshop_1\Devoxx4Kids_2013\manifest.xml
Nao\Devoxx4Kids_NAO_2_X\devoxx4kids_workshop_1\Devoxx4Kids_2013\Aldebaran\choice_sentences.xml
Nao\Devoxx4Kids_NAO_2_X__MACOSX\devoxx4kids_workshop_1\._DS_Store
Nao\Devoxx4Kids_NAO_2_X__MACOSX\devoxx4kids_workshop_1\._devoxx4kids
Nao\Devoxx4Kids_NAO_2_X__MACOSX\devoxx4kids_workshop_1\devoxx4kids\._DS_Store
Nao\Devoxx4Kids_NAO_2_X__MACOSX\devoxx4kids_workshop_1\devoxx4kids\._metadata
Nao\Devoxx4Kids_NAO_2_X__MACOSX\devoxx4kids_workshop_1\devoxx4kids\._behavior_1
Nao\Devoxx4Kids_NAO_2_X__MACOSX\devoxx4kids_workshop_1\devoxx4kids\._manifest.xml
Nao\Devoxx4Kids_NAO_2_X__MACOSX\devoxx4kids_workshop_1\devoxx4kids\behavior_1\._Aldebaran
Nao\Devoxx4Kids_NAO_2_X__MACOSX\devoxx4kids_workshop_1\devoxx4kids\behavior_1\._behavior.xar
Nao\Devoxx4Kids_NAO_2_X__MACOSX\devoxx4kids_workshop_1\devoxx4kids\behavior_1\Aldebaran\._choice_sentences.xml
Nao\Kurssilta\Cheek - Sokka Irti (Official Music Video).mp3
Nao\Kurssilta\Migos Look At My Dab (WSHH Exclusive - Official Music Video).mp3
Nao\Kurssilta\Rick Astley - Never Gonna Give You Up.mp3
Nao\Kurssilta\robokurssi.cbl
Nao\Kurssilta\tsouni.dab\metadata
Nao\Kurssilta\tsouni.dab\manifest.xml
Nao\Kurssilta\tsouni.dab\Migos Look At My Dab (WSHH Exclusive - Official Music Video).mp3
Nao\Kurssilta\tsouni.dab\tsouni.dab.pml
Nao\Kurssilta\tsouni.dab\behavior_1\behavior.xar
Nao\Kurssilta\tsouni.dab\behavior_1\ExampleDialog\ExampleDialog.dlg
Nao\Kurssilta\tsouni.dab\behavior_1\ExampleDialog\ExampleDialog_enu.top
Nao\Video\1366228_518766964862265_45857_n.mp4
Nao\Video\1366228_518766964862265_45857_n_1.mp4

Nao\Videot\desktop.ini
Nao\Videot\IMG_9073.mp4
Nao\Videot\IMG_9073_1.mp4
Nao\Videot\Nao_7 senses for natural interaction.mp4
Nao\Videot\Nao_teaser2.mp4
Nao\Videot\robokurssi_youtube.mp4
Nao\Videot\Video henkil"lt,, Hans Hokka 2.mp4
Nao\Videot\Video henkil"lt,, Hans Hokka 3.mp4
Nao\Videot\Video henkil"lt,, Hans Hokka 4 (1).mp4
Nao\Videot\Video henkil"lt,, Hans Hokka 4.mp4
Nao\Videot\Video henkil"lt,, Hans Hokka 5.mp4
Nao\Videot\Video henkil"lt,, Hans Hokka 6.mp4
Nao\Videot\Video henkil"lt,, Hans Hokka 7.mp4
Nao\Videot\Video henkil"lt,, Hans Hokka 8.mp4
Nao\Videot\Video henkil"lt,, Hans Hokka.mp4
Nao\Web Pages\googlebc658e2abf1283ac.html
Nao\Web Pages\index.html