

Choregraphen käyttö Pepperin ohjelmoinnissa

Micky Kyei
20.8.2018

Sisälllys

1	MIKÄ ON PEPPER?	2
1.1	Kuka on luonut Pepperin ja miksi?	2
1.2	Pepperin tekniset tiedot	2
1.3	Pepperin kanssa kommunikointi	3
2	PEPPERIN OHJELMOINTITYÖKALUT	4
2.1	Eri tavat ohjelmoida Pepper	4
2.2	Choregraphe	4
2.2.1	Käyttöliittymä	5
2.2.2	Toiminnot eli Boxit	6
3	PEPPERIN OHJELMOINTI	7
3.1	Pepperin käynnistäminen ja siihen Choregraphen yhdistäminen	7
3.2	Ohjelman rakentaminen	8
3.3	Äänitiedostojen toistaminen	9
3.4	Puheen tunnistus	9
3.5	Dialogit	10
3.6	Animation mode ja aikajanat	13
3.7	Solitary ja Interactive Activityt	15
3.8	Iän ja kasvopiirteiden tunnistus	16
3.9	Flow Control -ehtoboxit	18
	LÄHTEET	21

1 MIKÄ ON PEPPER?

1.1 Kuka on luonut Pepperin ja miksi?

Pepper on SoftBank Roboticsin vuonna 2014 julkaisema humanoidirobotti. Pepper on Softbank Roboticsin uusin humanoidirobotti, joka on kehitetty yhteistyössä ranskalaisen yhtiön, Aldebaranin kanssa. Aldebaranin aikaisemmat robotit ovat nimeltään NAO ja ROMEO. Pepper eroaa edeltäjistään siten, että se pystyy tulkitsemaan ihmisten tunteita lukemalla kasvopiirteitä ja äänensävyä, ja pystyy muuttamaan käytöstään sen mukaisesti. (Byford, 2014.)

Pepperiä yleensä hyödynnetään asiakaskokemuksen parantamiseksi. Pepperiä käytetään mm. asiakkaiden opastamiseen, viihdyttämiseen ja erilaisiin vastaanottotoimintoihin. Muun muassa Nescafe on ostanut 1000 Pepperiä yhtiön kodinkonemyymälöihin auttamaan asiakkaitaan etsimään kahvinkeittäimiä. (Reese, 2016.)

Pepper kykenee myös itse lähestymään ihmisiä. Pepperin avulla yritykset voivat houkuttaa paljon mielenkiintoa ja mahdollisesti lisää asiakkaita. Pepperiä voidaan personalisoida lataamalla siihen ohjelmia SoftBank Robotics Storesta. Pepper-robotin Tampereen ammattikorkeakoululle myynyt Loomis Tekniikka tarjoaa myös palvelua, jossa he räätälöivät Pepperin eri käytökset asiakkaan tarpeiden mukaisesti (Loomis Tekniikan Pepper-esite).

1.2 Pepperin tekniset tiedot

Pepperillä on neljä mikrofonia, kaksi HD kameraa ja 3-D syvyys sensori. Näiden avulla Pepper kykenee havaitsemaan ihmisten ympäristössään ja ottamaan heihin katsekontaktin. Pepperiin integroidun tabletin avulla Pepper voi antaa tietoa visuaalisessa muodossa, tai kerätä esimerkiksi asiakaspalautetta. Pepper on 120 senttimetriä pitkä ja 28 kiloa painava. Se on suunniteltu näyttämään mahdollisimman vaarattomalta. Pepper puhuu yli viittätoista eri kieltä, mukaan lukien suomi. Pepper pystyy toimimaan autonomisesti noin kaksitoista tuntia, ennen kuin se pitää ladata. (SoftBank Roboticsin Pepper-esite.)

1.3 Pepperin kanssa kommunikointi

Kun Pepper on autonomisessa tilassa, ihmiset voivat jutella Pepperille ja Pepper ihmisille. Pepper ymmärtää yleisimmät kysymykset, ja osaa vastata niihin. Pepperin huomion saa joko menemällä sen silmien eteen puolentoista metrin sisälle, kutsumalla sitä nimeltä tai pyytämällä sitä tulemaan luokse, tai koskettamalla jotain sen sensoreista. Kun Pepper on kohdistanut huomionsa henkilöön, se pyrkii pitämään katsekontaktin, ja seuraa henkilöä katseellaan. Pepper ei vaihda huomion kohdettaan, jos se prosessoi jotain ärsykettä, jos se juttelee jollekin, tai jos sillä on jokin Activity kesken.

Pepperin silmät kertovat Pepperin tilasta. Kun Pepper kuuntelee katsomaansa henkilöä, pyörii sen silmien ympärillä laivastonsiniset LED-valot. Kun valot muuttuvat vihreiksi, prosessoi Pepper kuulemaansa lausetta. Kun valot ovat vaaleansiniset, Pepper ei ole havainnut ihmistä. Valojen ollessa valkoiset Pepper puhuu itse. Kun silmien kulmissa palavat vaaleanpunaiset valot, on Pepper havainnut ihmisen.

2 PEPPERIN OHJELMOINTITYÖKALUT

2.1 Eri tavat ohjelmoida Pepper

SoftBank Robotics tarjoaa ohjelmistokehitystyökaluja usealle eri ohjelmointikielelle (C++, Python, Java, JavaScript). SoftBank Roboticsilla on myös Choregraphe-niminen graafinen ohjelmointityökalu. Choregraphe on suunniteltu siten, ettei se vaadi aikaisempaa kokemusta ohjelmoinnista. Choregraphen voidaan luoda Pepperille kehonkieltä tarvitsematta ohjelmoida robotin jokaista niveltä yksi kerrallaan. Choregraphella voidaan myös luoda Pepperille erilaisia verbaalisia kanssakäymistapoja. Tähän on käytetty QiChat -nimistä ihmisten ja robottien väliseen dialogiin suunniteltua ohjelmointikieltä. (SoftBank Robotics – Tools.)

2.2 Choregraphe

Choregraphe on sovellus, jonka avulla voidaan luoda animaatioita, dialogeja ja toimintoja SoftBank Roboticsin roboteille. Choregraphen avulla voidaan yhdistää tietokone robottiin, jolloin on yksinkertaista ladata ja testata käyttäjän luomia ohjelmia. Choregraphella voidaan myös testata ohjelmia käyttämällä virtuaalista robottia. Graafisen käyttöliittymän avulla käyttäjä voi rakentaa ohjelmia kirjoittamatta riviäkään koodia. Toiminnot luodaan raahaamalla ne kirjastosta ohjelman kulkukaavioon, jonka jälkeen niitä voidaan räätälöidä muuttamalla niiden parametreja. Choregraphe tarjoaa kirjaston eri toimintoja roboteille, joita yhdistelemällä voidaan luoda monimutkaisia ohjelmia.

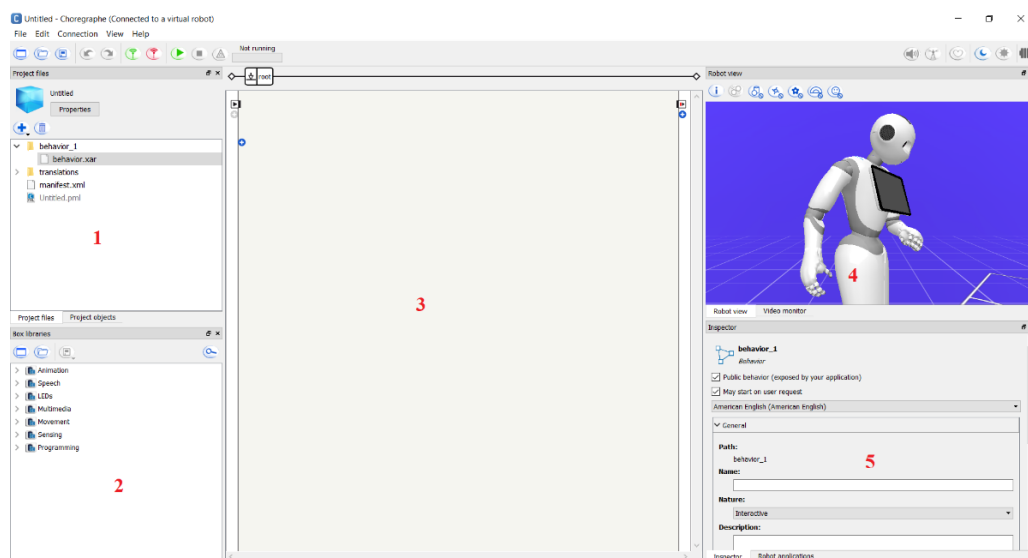
Kun luodaan ohjelma, sitä kutsutaan Behavior-ohjelmaksi. Kun Behavior-ohjelmalle annetaan jokin laukaisin, kutsutaan sitä Activity-ohjelmaksi. Activity-ohjelmia on kahdenlaisia: Inteactive ja Solitary. Interactive Activityt vaativat kanssakäymistä ihmisen kanssa. Solitary Activityt voidaan suorittaa ilman ihmistä. Solitary Activityt voidaan keskeyttää, jos jonkin Interactive Activityn laukaisin on toteutettu. Interactive Activityjä ei keskeytetä muiden Activity-ohjelmien takia.

2.2.1 Käyttöliittymä

Choregraphessa on viisi pääpaneelia ja työkalupalkki. Työkalupalkista vasemmasta laidasta yhdistetään ja katkaistaan yhteys robotteihin, luodaan ja avataan projekteja, ja ajetaan luotuja ohjelmia. Työkalupalkin oikeasta laidasta säädetään äänenvoimakkuutta, hallitaan autonomista tilaa, sekä laitetaan robotti lepäämään tai herätetään se. Työkalupalkki näyttää myös robotin akun tilan.

Numero	Nimi	Tehtävä
1.	Project files	Hallinnoi projektin tiedostoja.
2.	Box libraries	Sisältää toiminnot, jotka raahataan ohjelman kulkukaavioon.
3.	Kulkukaavio	Kaavio, jossa luodaan ohjelmat.
4.	Robot view	3D representaatio robotin tilasta, joka päivittyy reaaliajassa robotin liikkeessa. Video monitor -välilehdellä nähdään robotin kameran näkymä.
5.	Inspector	Hallinnoi ohjelman tietoja, esim nimeä ja kuvausta. Toimintoa klikkaamalla tässä voidaan asettaa toiminnolle parametreja. Robot applications -välilehdestä voidaan paketoita ja asentaa ohjelmia Pepperiin, tai poistaa siitä ohjelmia.

TAULUKKO 1. Choregraphen pääpaneelit ja niiden tehtävät (Kuva 1).

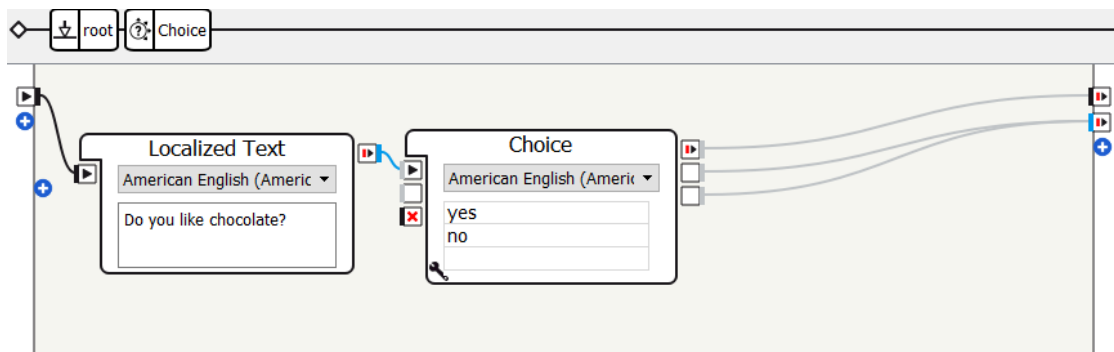


KUVA 1. Choregraphen pääpaneelit. Kuvankaappaus Choregraphesta.

2.2.2 Toiminnot eli Boxit

Choregraphessa toimintoja kutsutaan Boxeiksi. Boxeja yhdistelemällä ja muokkaamalla luodaan ohjelmat. Boxeja voidaan luoda itse, tai ne voidaan raahata Choregraphen kirjastosta suoraan kulkukaavioon, jonka voi myös mieltää Boxina. Kuten muissa Boxeissa, on kulkukaaviossa sisään- ja ulostulokohtat. Näiden sisään- ja ulostulokohtien avulla voivat Boxit käynnistyä ja pysähtyä, sekä kommunikoida toistensa kesken. Boxit yhdistetään linkkien avulla.

Boxeja on kolmen tyyppisiä: Script, Timeline ja Flow Diagram. Script Box on Box joka sisältää vain ohjelmakoodia. Tämä on Box-tyypeistä yksinkertaisin. Timeline Box sisältää ohjelmakoodia sekä oman aikajanan. Aikajanojen avulla voidaan ajoittaa toimintoja toistensa kanssa. Flow Diagram Box sisältää ohjelmakoodia sekä oman kulkukaavion. Flow Diagram Boxit voivat sisältää muita Boxeja. Näiden avulla voidaan visuaalisesti selkeyttää koko ohjelman kulkukaaviota. Flow Diagram Boxeille voidaan lisätä tai poistaa sisään- ja ulostulokohtia. (Kuva 2.)



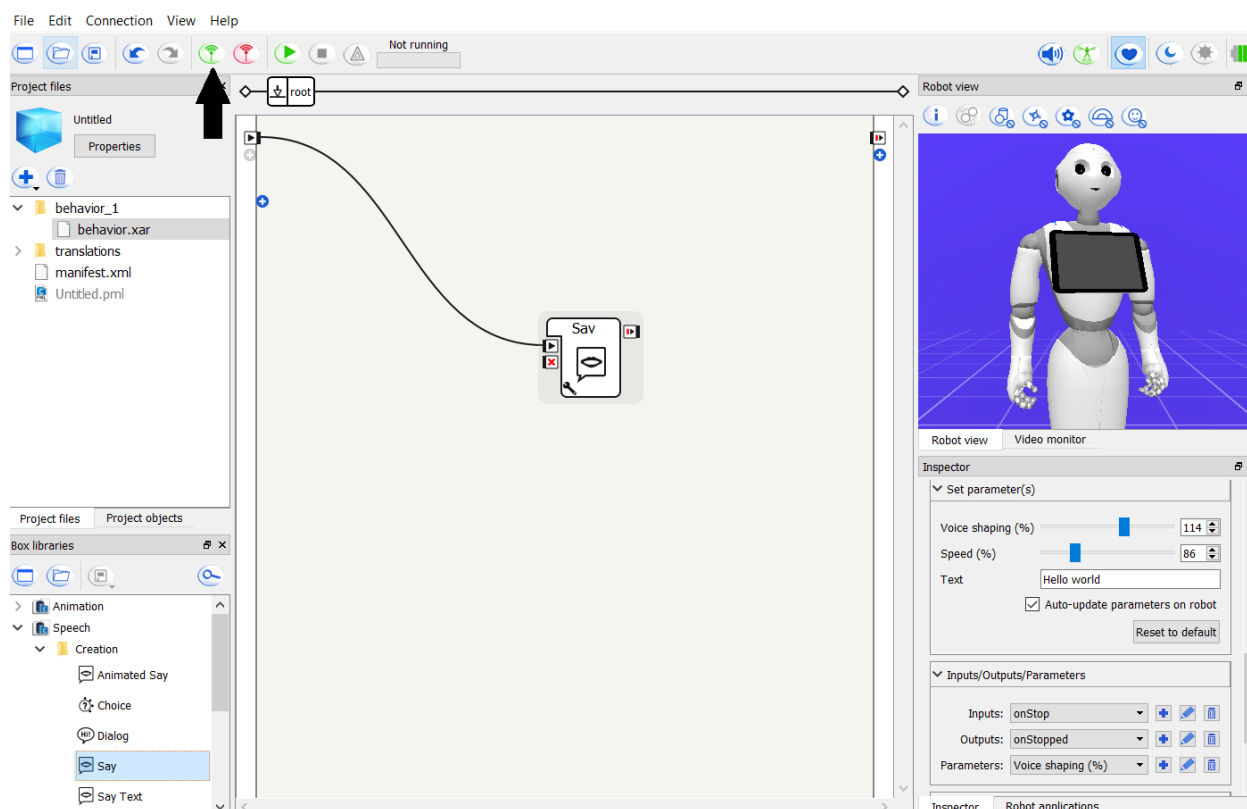
KUVA 2. Flow Diagram Box. Kuvankaappaus Choregraphesta.

3 PEPPERIN OHJELMOINTI

3.1 Pepperin käynnistäminen ja siihen Choregraphen yhdistäminen

Pepper käynnistetään painamalla sen rinnassa olevaa nappia. Nappi sijaitsee tabletin takana. Käynnistysprosessin aikana Pepperin korvien LED-valot kertovat käynnistykseen edistymisestä. Käynnistyminen voi kestää useita minuutteja.

Jotta Pepperiin voidaan ajaa ohjelmia, on siihen ensin luotava yhteys Choregraphessa. Tämä tapahtuu painamalla Choregraphen ylälaudassa sijaitsevaa ”Connect to” -nappia.



KUVA 3. Kuvankaappaus Choregraphesta.

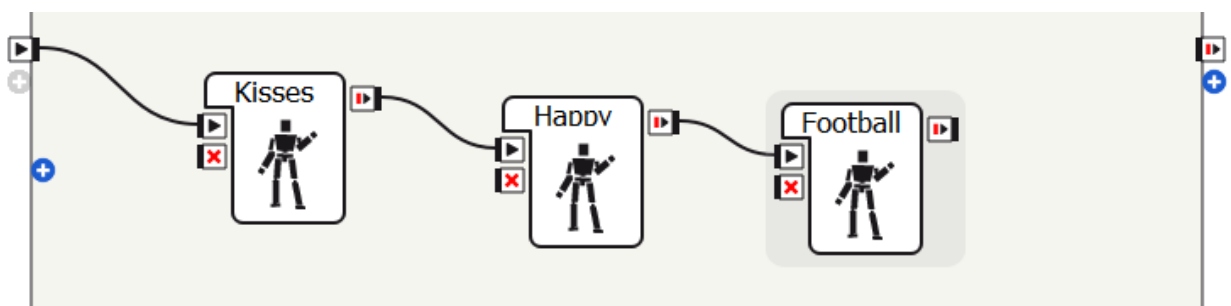
Tämän jälkeen ohjelma pyytää valitsemaan robotin. Valitse kohta ”Use fixed IP/hostname” ja lisää siihen Pepperin IP-osoite. Tämän jälkeen olet luonut yhteyden Pepperiin. Pepper kertoo IP-osoitteensa, kun sen rinnassa olevaa nappia painetaan kerran.

3.2 Ohjelman rakentaminen

Choregraphen ”Box libraries” -palkissa sijaitsee erilaisia toimintoja, joita Pepperille voi antaa. Valitse sieltä toiminto ”Say” ja raahaa se Choregraphen keskelle sijaitsevalle alueelle, joka toimii ohjelman kulkukaaviona, kuvan 1 mukaisesti. Yhdistä toiminnon onStart -kohta koko ohjelman onStart -kohtaan klikkaamalla jompaakumpaa kohtaa ja raahaamalla viiva niiden välille. Klikkaamalla luomaasi Boxia voit asettaa sille eri parametreja. Boxien nimiä ja kuvauksia voidaan myös muuttaa. Raahaamalla viivan ulostulokohdasta onStopped koko ohjelman oikeassa reunassa sijaitsevaan onStopped -kohtaan varmistetaan se, ettei Pepper jää pyörittämään ohjelmaa sen suoritettuaan. Nyt voit ladata ja pyörittää ohjelman painamalla ylälaudassa olevaa vihreää play-nappia.

Pepperillä on myös ”Animated say” -toiminto, jolloin hän liikuttaa raajojaan sanoessaan parametriksi annetun lauseen. Toiminnon parametriksi annetaan myös haluttu animaatio (lista animaatioista: http://doc.aldebaran.com/2-1/naoqi/audio/animatedspeech_advanced.html#juju-robot-list-of-tags-available-by-default).

Choregraphen käyttöliittymällä on helppo määrittää, missä järjestyksessä toiminnot suoritetaan. Yleisin Boxien sisääntulokohta on onStart. onStart käynnistää Boxin toiminnan. Yleisimmän ulostulokohdan, onStopped-kohdan avulla voidaan määrittää, mitä tehdään sen jälkeen, kun Boxin toiminta loppuu.



KUVA 4. Kolme perättäistä toimintoa, jotka suoritetaan erikseen. Kuvankaappaus Choregraphesta.

Boxien sisään- ja ulostulokohtia on neljää eri tyyppiä: Bang, Number, String, Dynamic.

Bang-tyyppiset sisään- ja ulostulokohdat ovat yleisimpiä; ne eivät vie tai vastaanota mitään dataa. Number-tyyppiset sisään- ja ulostulokohdat vievät ja vastaanottavat numeron tai

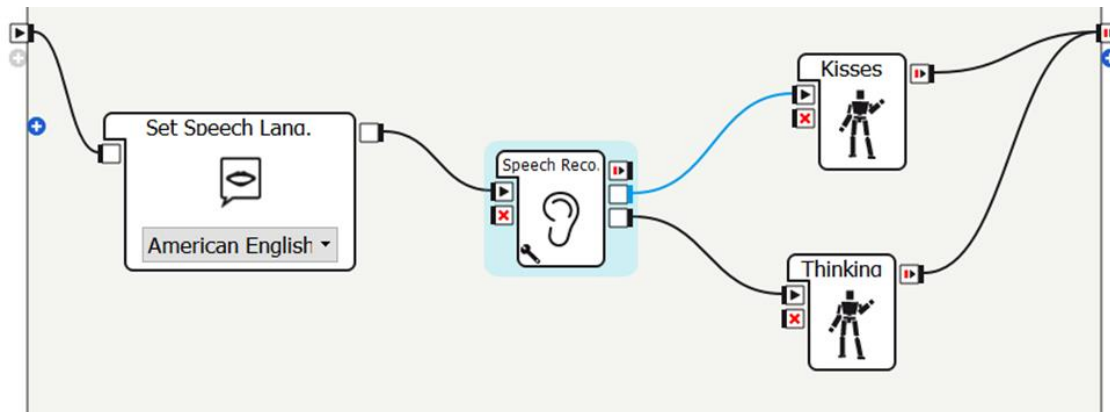
taulukon numeroita, kun taas String-tyyppiset vievät ja vastaanottavat dataa tekstin muodossa. Dynamic-tyypin sisään- ja ulostulokohdat ovat yhdistelmä kaikkia muita tyyppejä. Ne voivat viedä ja vastaanottaa numeroita tai tekstiä. Ne voivat myös olla viemättä tai vastaanottamatta minkäänlaista dataa.

3.3 Äänitiedostojen toistaminen

Äänitiedostojen toistamiseen on Choregraphessa Play Sound -Boxi. Toistettava tiedosto on tuotava projektiin käyttämällä ”Import files” -toimintoa, joka löytyy projektitiedostojen yläpuolella sijaitsevan sinistä +-merkkiä klikkaamalla. Ohjelmaan lisätyn Play Sound -toiminnon parametreistä voi määrittää, minkä tiedoston haluaa toistaa. Parametreistä voi myös määrittää äänitiedoston alkamiskohdan ja äänenvoimakkuuden.

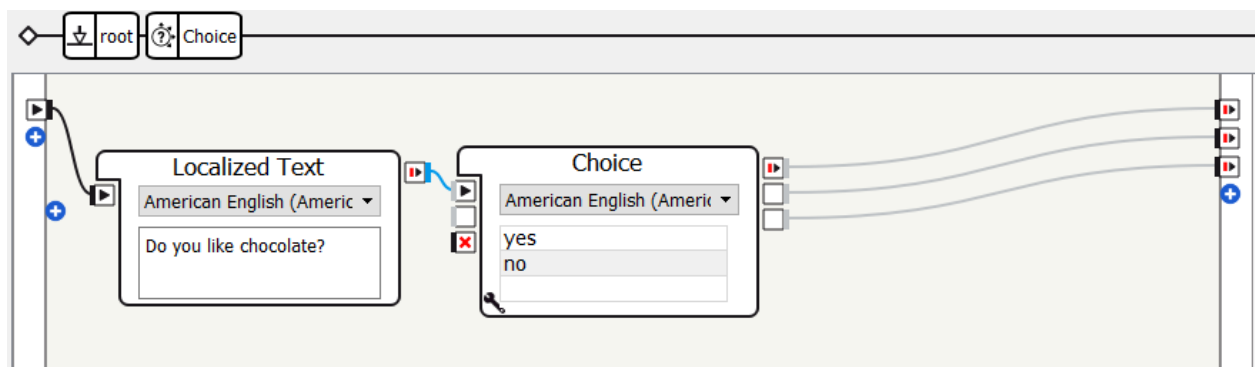
3.4 Puheen tunnistus

Pepper kykenee kuuntelemaan ja reagoimaan puheeseen ja sanoihin, jotka hän ymmärtää. Raahaa Speech-kansioista Set Speech Language -Box ja liitä sen sisääntulokohta ohjelman sisääntulokohtaan. Aseta kieleksi American English. Tämän jälkeen raahaa Speech-kansioista Speech Recognition -toiminto ja liitä sen sisääntulokohta Set Speech Language -toiminnon ulostulokohtaan. Speech Reco -toiminnolla on parametrinaan lista ymmärretyistä sanoista. Tässä tapauksessa oletussanoina ovat ”yes” ja ”no”. Liitä Speech Reco -toiminnon ulostulokohtiin wordRecognized ja onNothing eri toiminnot. Ohjelmaa pyörittäessä Pepper tekee eri toiminnon riippuen siitä, kuuleeko hän yhden ymmärretyistä sanoista vai ei.



KUVA 5. Esimerkki puheen tunnistamisesta. Kuvankaappaus Choregraphesta.

Speech-kansion Choice-toiminnon avulla Pepper voi reagoida eri tavalla riippuen siitä, mikä vastauksen hän kuulee. Korvaa edellisen esimerkin Speech Reco -toiminto Choiceella. Tuplaklikkaamalla Choicea pääset sen sisäiseen toimintaan käsiksi. Lisää toiminnolle ulostulokohta. Laita ulostulokohdan tyypiksi ”Bang” ja sen Nature -kohtaan valitse onStopped. Tällöin ohjelma jatkaa eteenpäin vasta kun Choice -toiminto on loppunut. Nyt tekemäsi Choice -toiminto jatkaa eteenpäin eri ulostulokohdasta riippuen siitä, mikä vastauksen Pepper kuulee.

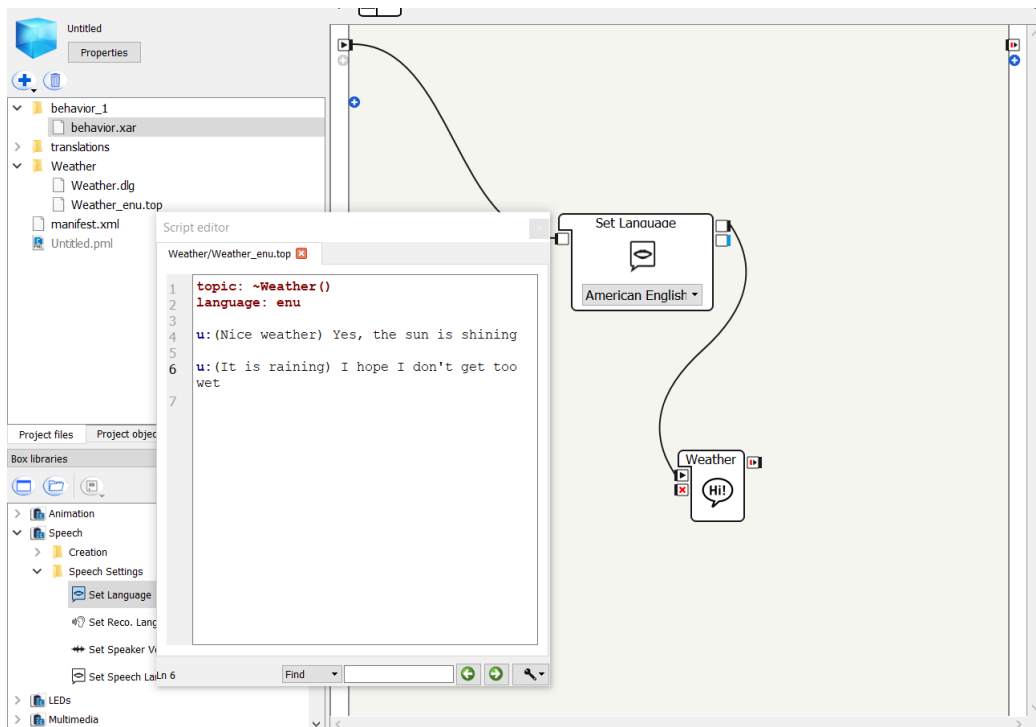


KUVA 6. Choice -Box. Kuvankaappaus Choregraphesta.

3.5 Dialogit

Dialogien avulla voidaan luoda Pepperille sovitut vastaukset hänen kuulemiin lauseisiin. Raahaa aluksi Speech-kansiosta Set Language -toiminto, ja aseta kieleksi American English. Klikkaa ohjelman kulkukaavion harmaata aluetta ja valitse Create a new box > Dialog. Lisää dialogille aihe ja paina ok-nappia. Nyt projektin tiedostoihin on lisätty uusi kansio, joka on nimetty nimeämäsi aiheen mukaisesti. Valitse kansioista _enu.top -päätteinen tiedosto ja

tuplaklikkaa sitä. Ruudulle avautuu editori, jossa voit muokata dialogia. Yhdistä tekemäsi dialogitoiminto Set Language -toimintoon.

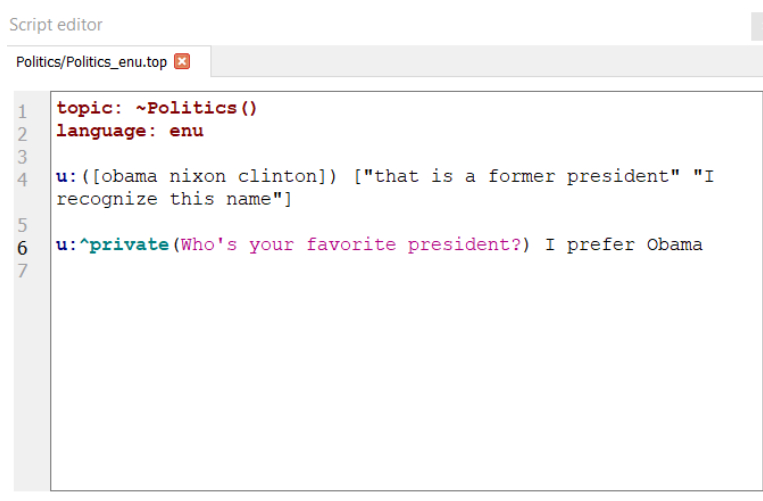


KUVA 7. Esimerkki dialogista. Kuvankaappaus Choregraphesta.

_enu.top -tiedostojen rivit ovat sääntöjä. Sääntöjen syntaksi on yksinkertainen:

- u: on avainsana käyttäjän omalle säännölle.
- Sulkujen sisällä oleva teksti on ihmissyötteitä, eli se mitä Pepper kuulee.
- Sulkujen jälkeinen teksti on se, mitä Pepper vastaa kuulemaansa syötteeseen.

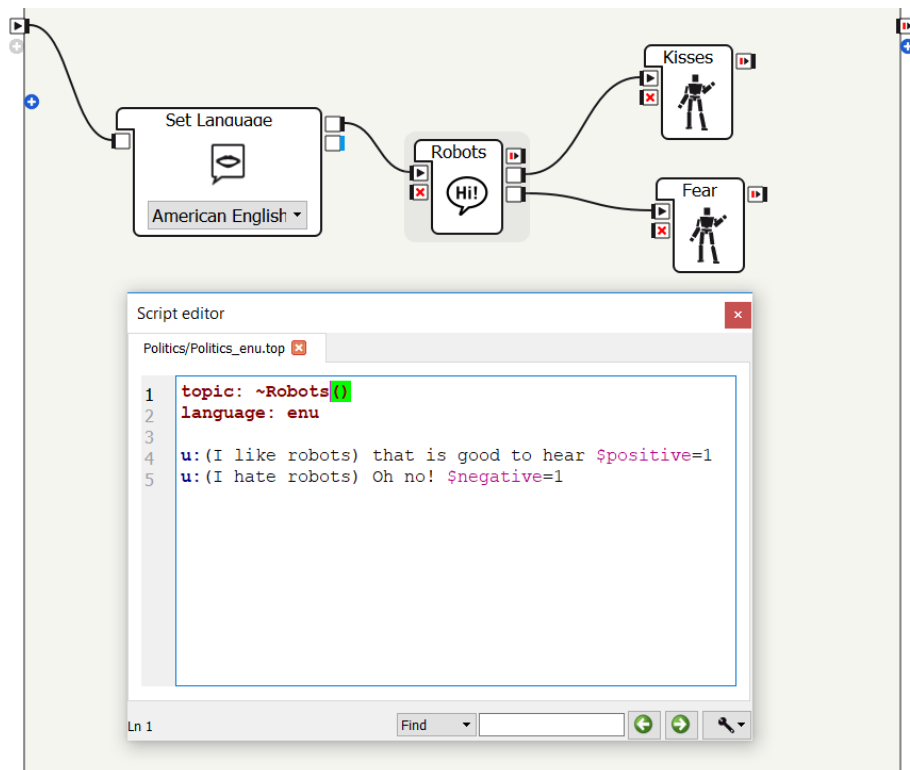
Dialogien säännöille voidaan asettaa useita ihmissyötteitä tai vastauksia (Kuva 8).



KUVA 8. Yksityiset säännöt dialogeissa. Kuvankaappaus Choregraphesta.

Kuvan 8 esimerkin mukaisesti ihmisyytteen hakasulkujen sisällä olevat sanat ovat kaikki vaihtoehtoja, jotka laukaisevat sen jälkeisen vastauksen. Pepper vastaa käyttäen sen vaihtoehtoja, jotka sijaitsevat hakasulkujen sisällä, järjestyksessä. Tällöin dialogeista ei tule yksitoikkoisia. Huomioi että lausevaihtoehdot ovat laitettava lainausmerkkien sisälle.

Kuva 8 antaa myös esimerkin yksityisistä säännöistä. Yksityiset säännöt tulevat käyttöön, vasta kun dialogi saa Pepperin huomion, eli vasta kun jokin dialogin julkisista säännöistä on toteutettu. Dialogien säännöissä voidaan myös määrittää, mistä ulostulokohdasta ohjelma jatkuu (Kuva 9).

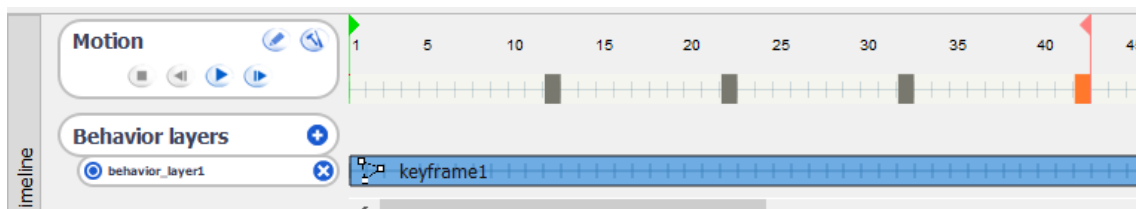


KUVA 9. Esimerkki ulostulokohdan muuttamisesta. Kuvankaappaus Choregraphesta.

Kuvassa 7 on Robots -nimiseen Dialog Boxiin lisätty kaksi ulostulokohtaa, positive ja negative. Sen _enu.top -tiedoston säännöissä voidaan määrittää, kummasta ulostulokohdasta ohjelma jatkuu, riippuen laukaistusta säännöstä. Tällöin voidaan luoda ohjelmia, joissa Pepper toimii eri tavalla riippuen ihmisyyttestä.

3.6 Animation mode ja aikajanat

Choregraphen avulla voidaan luoda Pepperille omia liiketoimintoja, joita kutsutaan animaatioiksi. Animaatiot tehdään tallentamalla Pepperille halutut asennot Timeline Boxiin. Boxin käynnistyessä Pepper käy läpi aikajanan ja toteuttaa siihen tallennetut asennot. Jotta animaation luominen voidaan aloittaa, on Pepperin autonominen toiminta lopetettava. Tämän voi tehdä painamalla Pepperin rintanappia kahdesti, tai Choregraphen kautta. Raahaa Animation-kirjastosta Timeline Box kulkukaaviolle. Tuplaklikkaa luomaasi boxia nähdäksesi aikajanan kulkukaavion yläpuolella.



KUVA 10. Aikajana, johon on tallennettu neljä asentoa. Kuvankaappaus Choregraphesta.

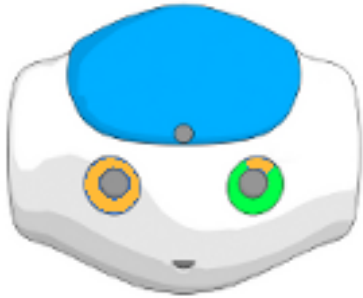
Aikajanan yllä olevat numerot ovat Frame-yksikköjä. Tummanharmaat palkit ovat tallennettuja asentoja. Aikajanalla oleva vihreä indikaattori on Start Frame, ja punainen indikaattori on End Frame. Nämä indikaattorit nimiensä mukaan kuvaavat animaation ensimmäistä ja viimeistä Framea. Näiden indikaattorien ulkopuolisia asentoja ei toteuteta. Timelinen luomisen jälkeen on varmistettava, että Pepper on hereillä. Pepperin voi herättää Choregraphen yläpalkin oikeassa laidassa olevasta napista.



KUVA 11. Choregraphen yläpalkin oikea laita. Vihreä nappi käynnistää Animation Moden. Oikeanpuolimmainen nappi herättää Pepperin. Kuvankaappaus Choregraphesta.

Kun Timeline on luotu ja Pepper on hereillä, voidaan käynnistää Animation Mode. Animation Modessa Pepper voidaan liikuttaa haluttuun asentoon, jonka jälkeen asento tallennetaan aikajanalle. Pepperin käsiä ja päätä voidaan liikuttaa vain, jos niiden nivelten jäykkyys on pois päältä. Tämä tapahtuu käsien tapauksessa pitämällä kämmenselässä olevaa kosketussensoria pohjassa, ja pään tapauksessa sen keskimmäistä kosketussensoria napauttamalla.

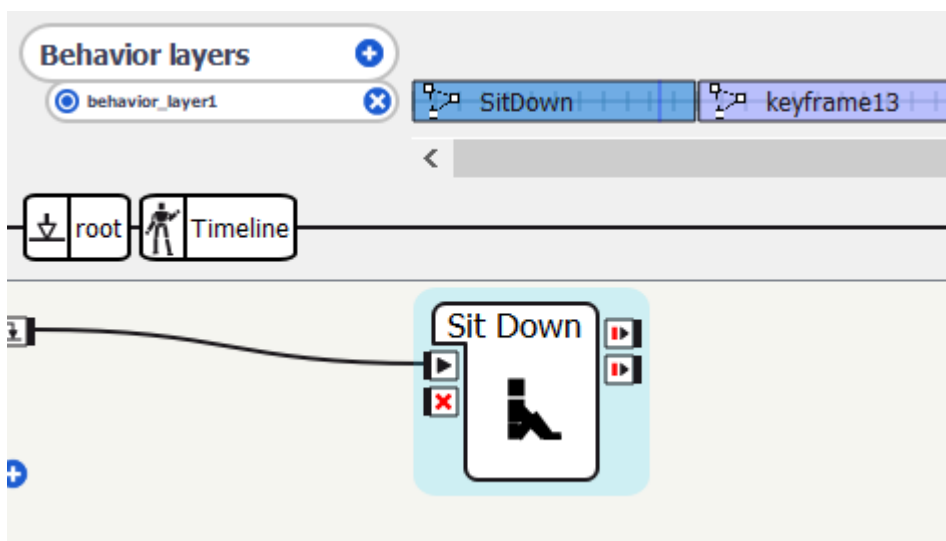
Nivelten jäykkyyden voi tarkistaa Pepperin silmistä. Oranssi valo kertoo kehonosan olevan jäykkänä, kun vihreä valo kertoo nivelen jäykkyyden olevan pois päältä. Kun silmien ylemmät osat ovat vihreitä, on pään jäykkyys pois päältä. Jos silmän loppuosa on vihreä, on sen puolen käden jäykkyys pois päältä.



KUVA 12. Pää ja oikea käsi on jäykkänä. Kuvankaappaus Choregraphesta.

Asentoja tallennetaan painamalla Pepperin kaikkia kosketussensoreita samanaikaisesti. Asennot voidaan myös tallentaa klikkaamalla aikajanaa hiiren oikealla näppäimellä ja valitsemalla ”Store joints in keyframe” ja valitsemalla mikä osa kehon asennosta tallennetaan. Asentoja voidaan esikatsella klikkaamalla niiden palkkeja aikajanalta. Asentojen ajoitusta voidaan muuttaa siirtämällä niiden palkkeja aikajanalla. Aikajanalan valikosta voidaan myös kopioida toisen puolen arvot toiselle puolelle, tai vaihtaa puolien arvoja keskenään.

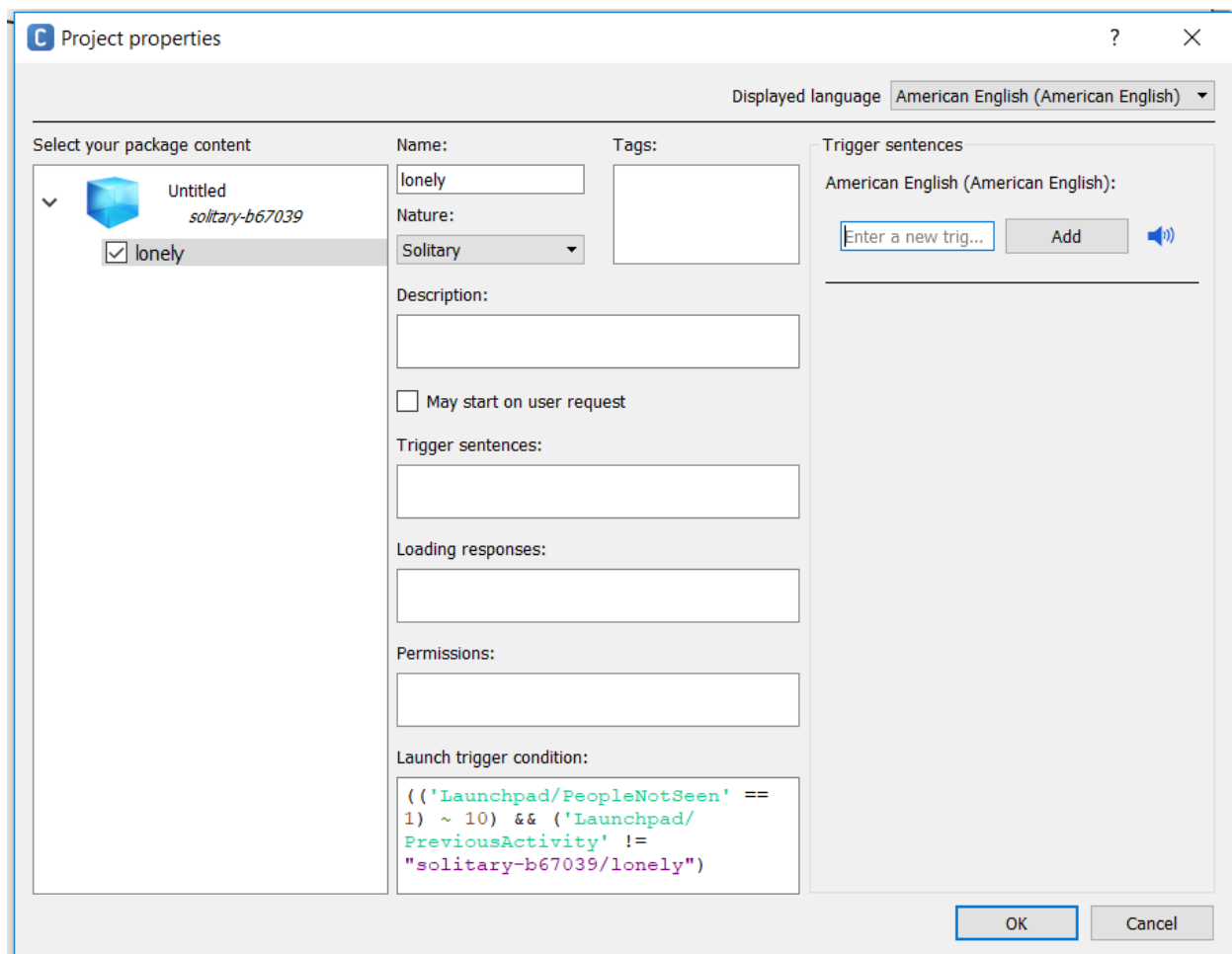
Behavior layers -toiminnon avulla voidaan lisätä toimintoja toteutettavaksi samanaikaisesti Timelinen kanssa. Uusi Behavior Layer lisätään klikkaamalla aikajanaa halutun Framen



kohdalta hiiren oikealla näppäimellä ja valitsemalla Insert Keyframe. Luotua Keyframea voidaan muokata klikkaamalla sitä.

3.7 Solitary ja Interactive Activityt

Pepper kykenee myös tekemään asioita ilman, että joku juttelee hänelle. Näitä kutsutaan Solitary Activityiksi. Tällaisten toimintojen kehittämisessä on käytettävä Trigger Conditions - lausekkeita. Trigger Conditions -lausekkeet ovat ohjelmoinnissa usein käytettäviä ehtolauseita. Kun toiminnon kaikki ehtolauseet toteutuvat, toteutetaan toiminto. Toiminnosta saadaan Solitary -tyyppinen muuttamalla sen tyyppiä Project Properties -asetuksista. Valitsemalla haluamasi toiminto Project Properties -asetuksista voidaan sille asettaa myös uusi nimi, kuvaus ja Trigger Conditions -ehtolauseet.



KUVA 14. Solitary Activity. Kuvankaappaus Choregraphesta.

Tutkitaan kuvan 12 ehtolauseita:

```
(('Launchpad/PeopleNotSeen' == 1) ~ 10) && ('Launchpad/PreviousActivity' != "solitary-b67039/lonely")
```

Ehtolauseen osa	Selitys
('Launchpad/PeopleNotSeen' == 1) ~ 10)	Launchpad/PeopleNotSeen on aina arvoltaan joko 1 tai 0. Kun arvo on 1, silloin Pepper ei ole havainnut ihmisiä. Tässä tapauksessa PeopleNotSeen -muuttujan arvon on oltava 1 kymmenen sekunnin ajan.
&&	AND -ehto. AND -ehto vaatii, että kyseisen toiminnon molempien ehtojen on toteuduttava, ennen kuin toiminto voidaan aloittaa.
('Launchpad/PreviousActivity' != "solitary-b67039/lonely")	PreviousActivity kertoo, mikä on viimeisin Pepperin toteuttama toiminto. Se sisältää toiminnon ID:n (löytyy Project Properties -asetuksista) ja toiminnon kansion nimen. Tässä tapauksessa PreviousActivity on oltava eri kuin kyseessä oleva toiminto. Tällöin toimintoa ei toisteta useita kertoja peräkkäin.

Solitary Activityn testaaminen tapahtuu asentamalla toiminto Pepperiin. Tämä tapahtuu painamalla Package and Install -nappia Robot Applications -paneelissa.

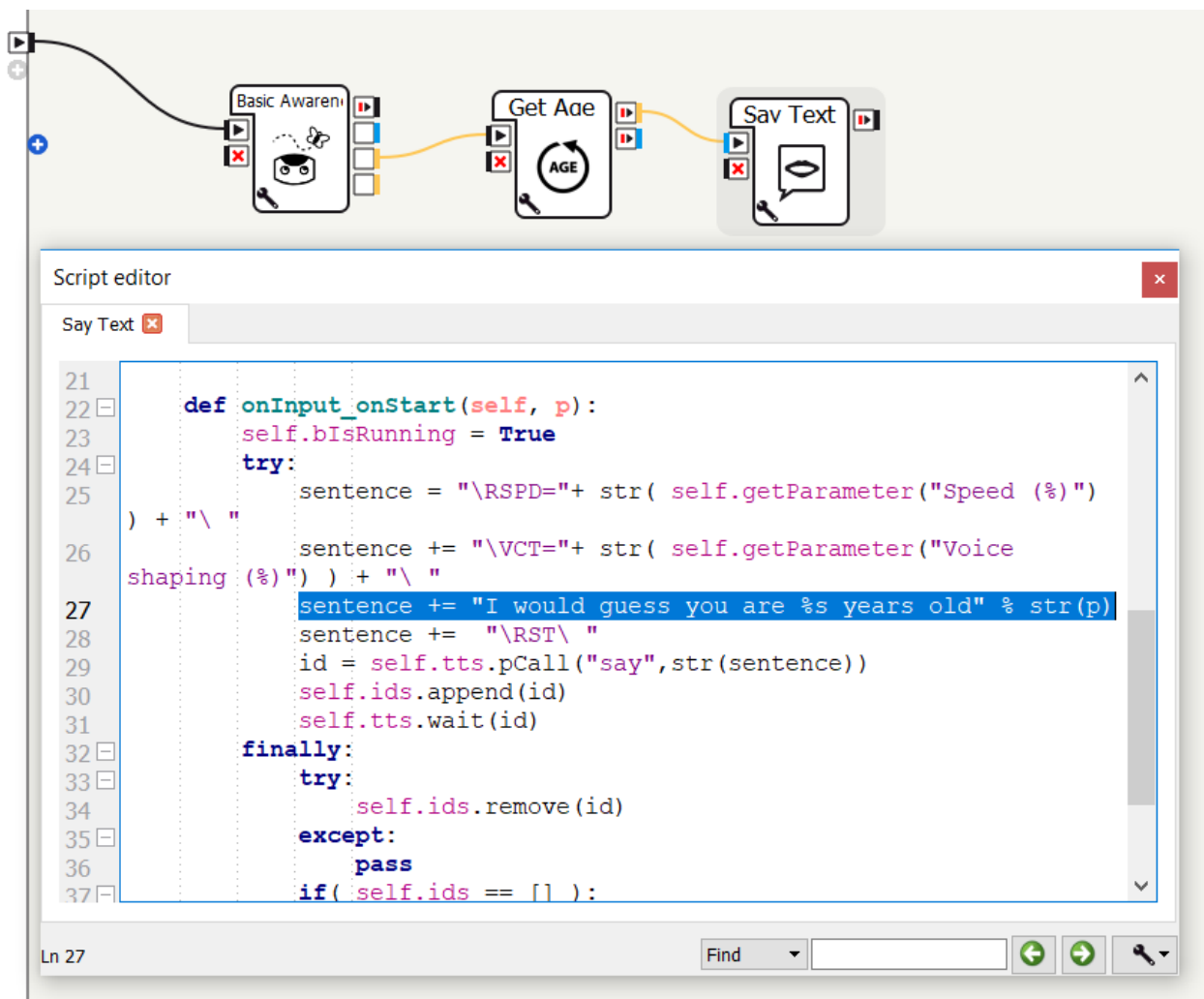
Interactive -tyyppiset toiminnot toteutetaan samalla tavalla kuin Solitary -tyyppin toiminnot. Interactive -tyyppisille toiminnoille voidaan lisätä Trigger Sentence -lauseita, jotka Pepper kuullessaan käynnistää toiminnon, jos Trigger Condition -ehtolauseet toteutuvat.

3.8 Iän ja kasvopiirteiden tunnistus

Pepper kykenee arvioimaan henkilöiden iän ja tunnistamaan eri ilmeitä. Näitä molempia varten on saatava Pepper tilaan, jossa hän havaitsee vain ihmisiä, ei kosketusta, ääntä tai liikettä. Tätä varten on raahattava Sensing -kirjastosta Basic Awareness -toiminto ohjelman

kulkukaavioon. Sen parametreista voidaan asettaa ne ärsykkeet, joihin Pepper reagoi. Sensing-kirjaston Get Expression ja Get Age -toiminnot tunnistavat kasvopiirteet ja iän. Kun ikä tai kasvopiirre on tunnistettu, toiminto lähettää saamansa arvon seuraavalle toiminnolle.

Jotta Pepper voi sanoa sen tunnistaman iän tai kasvopiirteen, on lisättävä Say Text -toiminto. Say Text -toiminto sanoo automaattisesti sen vastaanottaman syötteen. Say Text -toiminto yleensä lausuu vain sen vastaanottaman syötteen, mutta kuvassa 13 rivillä 27 on näytetty esimerkki siitä, miten Say Text -toimintoa voi muokata.

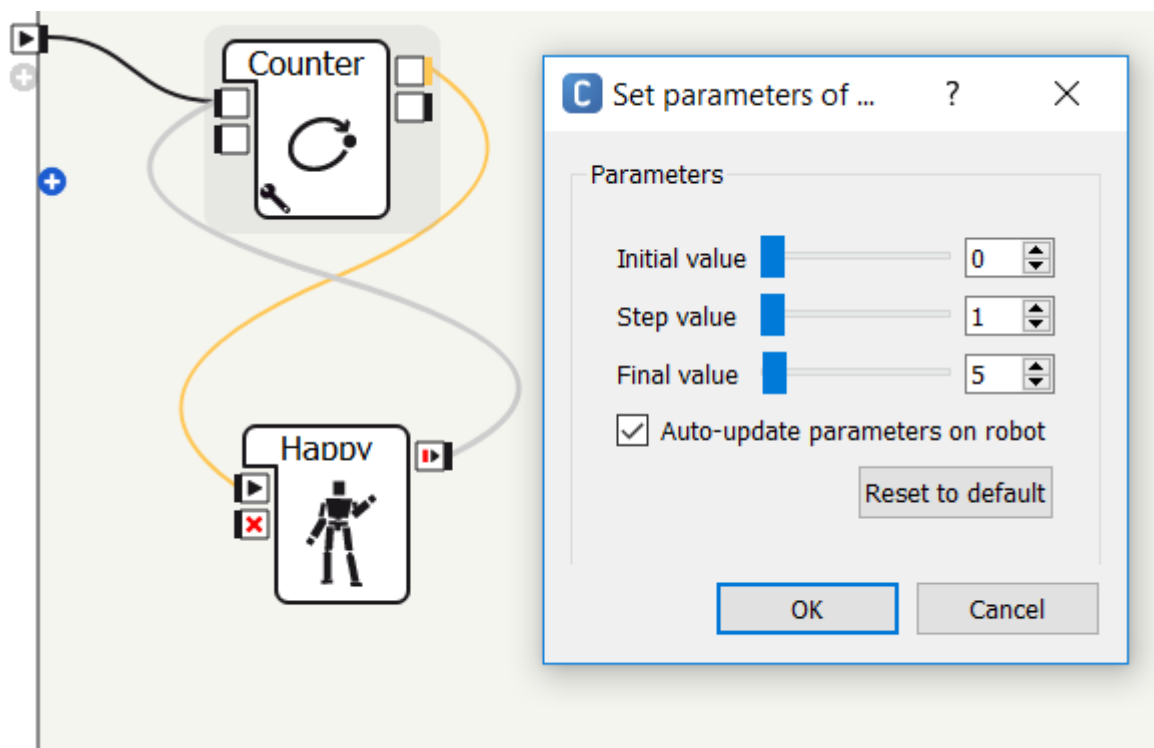


KUVA 15. Iän tunnistaminen ja sen sanominen. Kuvankaappaus Choregraphesta.

3.9 Flow Control -ehtoboxit

Flow Control Boxien avulla voidaan ehdollistaa jotain toimintoja ohjelman sisällä. Näiden boxien avulla voidaan ohjelmista moniulotteisimpia ja mahdollisesti parempia vuorovaikutuksen kannalta.

Counter -toiminto toistaa siihen liitetyn toiminnon niin monta kertaa, kuin on sen parametreissa määriteltä. Initial value määrittää ensimmäisen arvon, ja Final value määrittää viimeisen arvon. Joka kerta kun Counteriin liitetty toiminto suoritetaan, kasvatetaan Counterin nykyistä arvoa Step valuen verran. Counter on liitettävä suoritettavaan toimintoon siten, että Counterin ulostulo on yhdistetty suoritettavan toiminnon sisääntuloon, ja suoritettavan toiminnon ulostulo on yhdistettävä Counterin onNext -sisääntuloon.

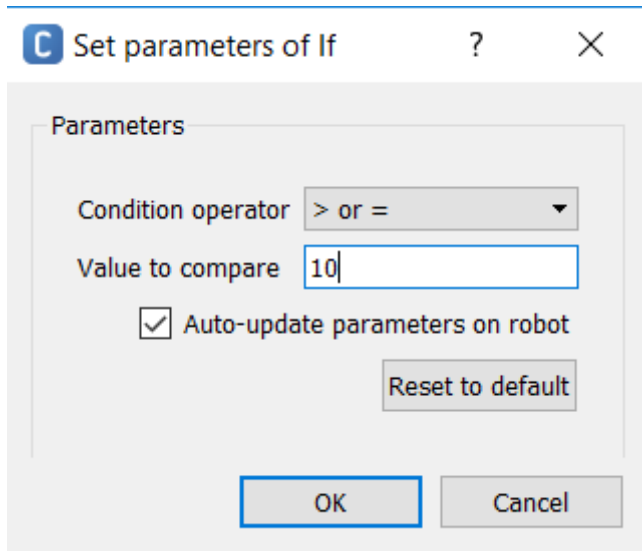


KUVA 16. Counter, joka toistaa Happy -toiminnon kuusi kertaa. Kuvankaappaus Choregraphesta.

Wait for signals -toiminnon avulla voi nimensä mukaisesti odottaa signaaleja useasta eri kohdasta ohjelmaa. Sillä on oletuksena kaksi sisääntuloa, ja se odottaa, että molemmista tulee signaalit, ennen kuin se lähettää signaalin eteenpäin ulostulosta. Tämän avulla voidaan

esimerkiksi odottaa, että kaksi eri toimintoa on suoritettu loppuun ennen eteenpäin siirtymistä.

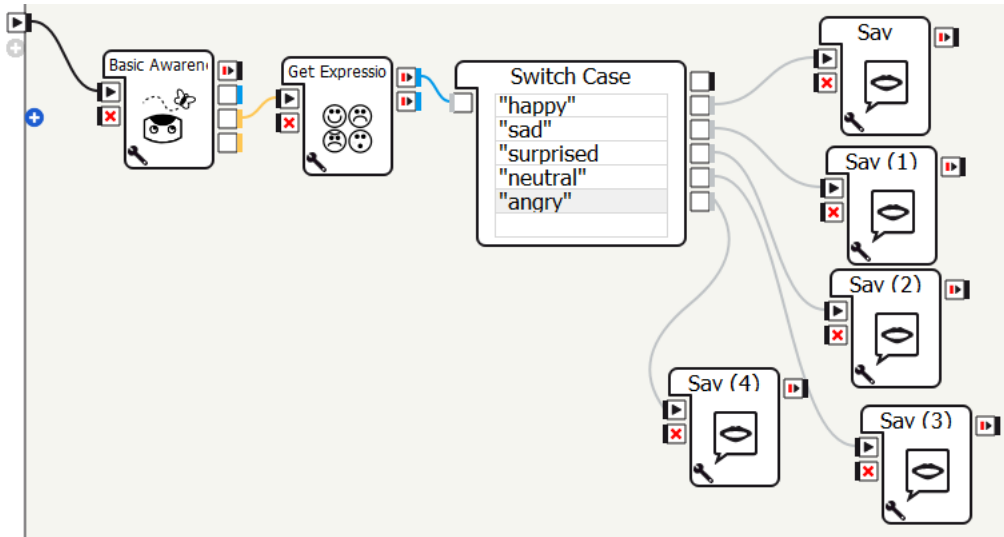
If -toiminnolla on yksi sisääntulo ja kaksi ulostulokohtaa. Kun If -toiminto vastaanottaa syötteen, se tarkastaa, toteutuuko sen parametrina oleva ehto. Jos se toteutuu, se lähettää signaalin output_then -ulostulokohdasta. Jos vastaanotettu syöte ei toteuta ehtoa, se lähettää signaalin output_else -ulostulokohdasta.



KUVA 17. Ehto toteutuu, jos vastaanotettu syöte on suurempi tai yhtäsuuri kuin 10. Kuvankaappaus Choregraphesta.

Only once -toiminnolla varmistetaan, että jokin asia toistetaan vain kerran ohjelman ollessa aktiivinen. Se lähettää signaalin ulostulokohdastaan vain ensimmäisellä kerralla, kun se saa signaalin.

Switch case vastaanottaa syötteen ja vertaa sitä sille asetettuihin arvoihin. Signaali lähetetään eri ulostulosta sen mukaan, mihin arvoa syöte vastasi. Jos syöte ei vastaa mitään arvoa, signaali lähetetään onDefault -ulostulokohdasta.



KUVA 18. Switch case, jossa Pepper sanoo eri asian riippuen tunnistetusta kasvonpiirteestä. Kuvankaappaus Choregraphesta.

LÄHTEET

Oppaan tekemisessä on käytetty apuna Aldebaranin tarjoamaa dokumentaatiota. Dokumentaatio tarjoaa laajemman näkemyksen siitä mitä Choregraphella ja Pepperillä voi tehdä.

<http://doc.aldebaran.com/2-5/index.html>

Byford, S. 2014. SoftBank announces emotional robots to staff its stores and watch your baby. Luettu 10.9.2018.

<https://www.theverge.com/2014/6/5/5781628/softbank-announces-pepper-robot>

Loomis Tekniikan Pepper-esite. Luettu 11.9.2018.

<https://www.loomistekniikka.fi/tuotteet/itse-ilmoittautuminen/pepper-uuden-ajan-asiakaspalvelua/>

Reese, H. 2016. Pepper the robot: a smart person's guide. Luettu 11.9.2018.

<https://www.techrepublic.com/article/pepper-the-robot-the-smart-persons-guide/>

SoftBank Roboticsin Pepper-esite. Luettu 11.9.2018.

<https://www.softbankrobotics.com/emea/en/robots/pepper/find-out-more-about-pepper>

SoftBank Robotics – Tools. Luettu 11.9.2018.

<https://www.softbankrobotics.com/emea/en/robots/tools>

SoftBank Robotics. Who is Pepper? Luettu 10.9.2018.

<https://www.softbankrobotics.com/emea/en/robots/pepper>