Service Provision in the Hospital of the Future

Developers

Final version (presented here) by Daniel Stroud and Peer-Olaf Siebers. Based on initial work from Penny Siebert, Elvira Perez Vallejos, Tommy Nilsson, and Peer-Olaf Siebers.

EABSS Version

1.0

Related Publication(s)

- Stroud D, Wagner C, and Siebers PO (2019) 'Agent-Based Simulation Modelling for Reflecting on Consequences of Digital Mental Health'. Working Paper.
- Siebert P, Siebers PO, Vallejos EP, and Nilsson T (2020) 'Driving complementarity in interdisciplinary research: A reflection', International Journal of Social Research Methodology, DOI: 10.1080/13645579.2020.1743545.

Focus

Initially just for discussion (PhiloLab), as describe in Siebert et al (2020). Afterwards, updated for implementation (EABSS), as described in Stroud et al (2019).

Motivation

Proactive machines are an idea that in the future hospitals will have automated machines as well as Doctors and Nurses that can help and treat patients. This project uses the idea of proactive machines to test the impact of technology on mental state and opinions of hospital patients.

Gathering Knowledge

Discussion with multi-disciplinary team. Revision and implementation by modeller.

Step 1: Define Objectives

Aim

• Reflect on the consequences of digital mental healthcare

Hypotheses

- Trust between machines/humans will affect human-human relationships in a negative way
- Machine-look influences our decisions

Experimental factors

- Number of doctors, proactive machines, and patients
- Number of beds

Responses

- Mental state of patients, e.g. happiness
- Opinions of each patient on the doctors and robots
- The trust of the patients on the robots

Step 2: Define Scope

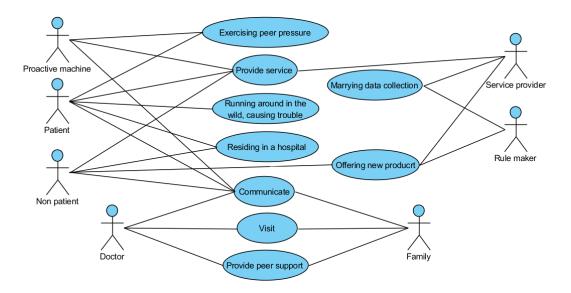
Key driver(s): Focus on fictive operational level

Category		Element	Decision	Justification
Actor	Human	Patient	Include	Patients are the ones who are being tested
		Doctor	Include	Doctors need to be compared against the Robot doctors
		Visitor	Include	Visitors can be used to influence patients
		Nurse	Exclude	Nurse can be viewed as the same as doctors in the simulation
	Proactive machine	Robot doctor	Include	Need to be compared against doctors
	Buildings	Hospital	Include	Location where the model is held
		Home	Include	Location where agents are held when they are being used
Physical	Furniture	Beds	Include	Patients need somewhere to stay
Environment	Structure	Walls	Exclude	Not needed because we are not interested in agents movements or collisions
		Doors	Exclude	Not needed to test the hypotheses
Category				
Calego	ry	Element	Decision	Justification
Catego	ry Robot	Decide how long to treat patient for	Decision Include	Justification Needed for more accurate results
Catego		Decide how long to		
Catego	Robot doctor behaviour	Decide how long to treat patient for Accurate	Include	Needed for more accurate results Movement is not needed to test the
Social and psychological	Robot doctor	Decide how long to treat patient for Accurate movement of robot Decide how long to	Include Exclude	Needed for more accurate results Movement is not needed to test the hypotheses
Social and	Robot doctor behaviour Doctor	Decide how long to treat patient for Accurate movement of robot Decide how long to treat patient for Accurate movement of	Include Exclude Include	Needed for more accurate results Movement is not needed to test the hypotheses Needed for more accurate results Movement is not needed to test the
Social and psychological	Robot doctor behaviour Doctor behaviour Patient	Decide how long to treat patient for Accurate movement of robot Decide how long to treat patient for Accurate movement of doctor Random hospital	Include Exclude Include Exclude	Needed for more accurate results Movement is not needed to test the hypotheses Needed for more accurate results Movement is not needed to test the hypotheses
Social and psychological	Robot doctor behaviour Doctor behaviour Patient behaviour Visitor	Decide how long to treat patient for Accurate movement of robot Decide how long to treat patient for Accurate movement of doctor Random hospital visits Decide whether the visitor should	Include Exclude Include Exclude Include	Needed for more accurate results Movement is not needed to test the hypotheses Needed for more accurate results Movement is not needed to test the hypotheses Needed for more accurate results

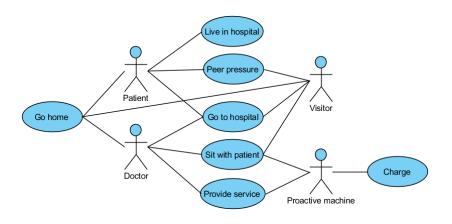
Step 3: Define Key Activities

Actor roles and related use cases

Initial diagram:



Final diagram (generated after completing Step 5):



Step 4: Define Stereotypes

Doctor

Stereotype	Extra time treating patients (Minutes)	
Senior	0	
Junior	10	

Robot

Stereotype	Humanlike variable	Effect on human opinion
Humanlike	0.5-1	Positive effect
Robotlike	0-0.5	Negative effect

Step 5: Define Agent and Object Templates

Relevant classes

Patient -doctorTreatment : double -robotTreatment : double -opinion : double onCheckupList : boole an -checkupTime : double satsfaction : double request : boolean -discharged : boole an -patience : double -inBedBool : boolean hasFamily : boolean -bed : Bed -condition : double -talkative : double -currentRobot : Robot -currentDoctor : Doctor +moveToAtHome() +moveToInHospital() +connect() +disconnect() +disconnectFromAll() +findFreeBed() +addToCheckupList() +removeFromCheckupList() +moveToDoctorTreatment() +moveToRobotTreatment() +moveToMixTreatment() +moveToNoTreatment() +goingHome()

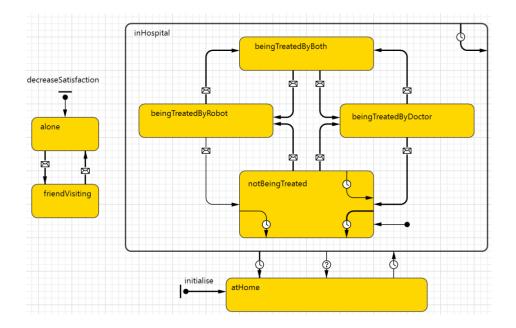
-level : double -waitTime : double -fatigute : double -patientGoneHome : boolean -initiallyAtWork : double -timeToStartWork : double -currentPatient : Patient -needsBreak : boolean -requestsWaiting : boolean +moveToAtHome() +moveToAtHome() +moveToAtWork() +moveToWaiting() +moveToWithPatient() +checkCheckupList()

Doctor

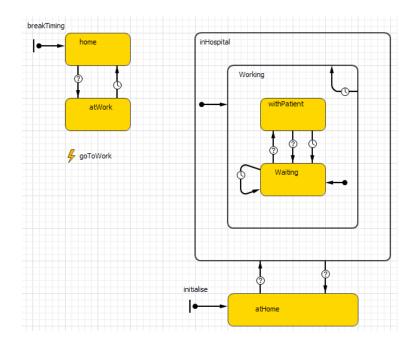
Robot -waitTime : double -price : double -humanLikeVar : double -patientGoneHome : boolean -charge : double -startX : double -startX : double -startY : double -currentPatient : Patient +moveToCharging() +moveToWaitingToBeCharged() +moveToWaiting() +moveToWithPatient() +checkCheckupList() Visitor -visiting : boolean -patient : Patient -avgVisiTime : double -visitRate : double -counter : double -recentlyVisited : double -recentlyVisited : double -waitTime : double +moveToAtHome() +moveToAtHome() +contemplate() +moveToPatient()



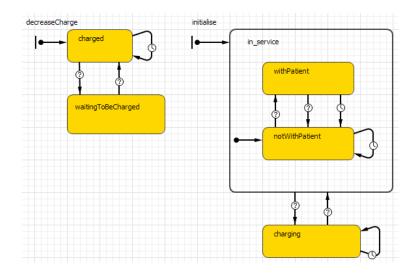
Patient state chart



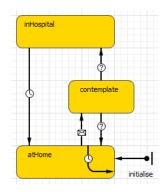
Doctor state chart



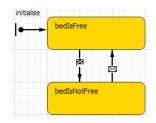
Robot state chart



Visitor state chart

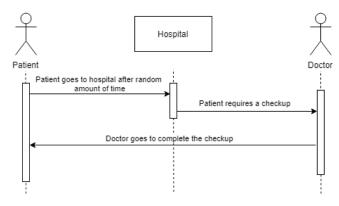


Bed state chart

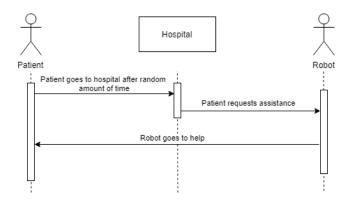


Step 6: Define Interactions

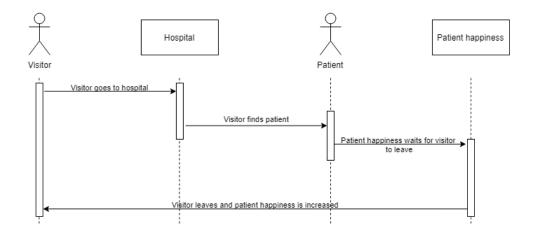
Sequence diagram for the use case "patient getting treated by doctor"



Sequence diagram for the use case "patient getting treated by robot"



Sequence diagram for the use case "visitor visits patient"



Step 7: Define Artificial Lab

Artificial Lab class definition

ArtificialLab -visitors : Visitor[] -robots : Robots[] -patients: Patients[] -patientsToSee : Patient* -patientCheckups : Patient* -freeBeds : Bed* -doctor1stShift : Doctor[] -doctor2ndShift : Doctor[] -beds : Bed[] -bed1 : int -rand : double -patientWithFamilies : Patient[] +initializeAgents() +visitorsEnter()

Implementation

Implementation in AnyLogic

