Developing Solutions to Identify and Treat Nocturnal Disorientation in Dementia Patients

Juan Luis Pérez Díez
Department of Software Engineering and Artificial Intelligence
Complutense University of Madrid, Madrid, Spain
jlpdiez@ucm.es

ColoSAAL

(TIN2014-57028-R)

UNIVERSIDAD COMPLUTENSE MADRID

GRASIA

Index

Initial state

The tools for the job

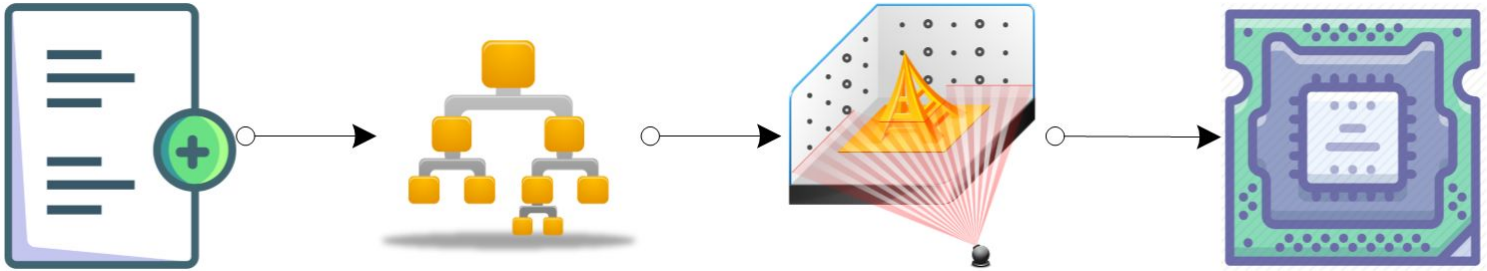
- **Twenty interviews** to dementia patients:
  - Alzheimer's disease, vascular dementia, lewy bodies…
  - Various degrees of disease evolution.
  - Multiple ages and socioeconomic statuses.
- **AIDE** - **A**mbient **I**ntelligence **D**evelopment **E**nvironment:

  * **Virtual Living Lab** *

  - **INGENME ⇒ SociAALML**: Graphical editor to model agents, environments and their behaviour.
  - **PHAT** Framework: Transforms the models of SociAALML into 3D simulations

Workflow

From interviews to solutions



Interview analysis

Build models

Scenario simulations

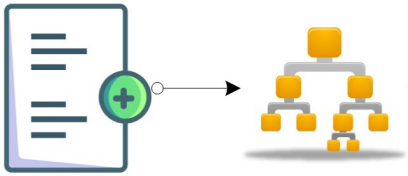
Experiments / Hardware

Interview Analysis

- Lots of reading, underlining and note taking.

- Key aspects of focus:
  - **Sociodemographic** data
  - Everyday **problems** and **difficulties**
  - **Technological** discussion
  - Day to day **activities** ⇒ Simulations

Build models
SociAALML Editor

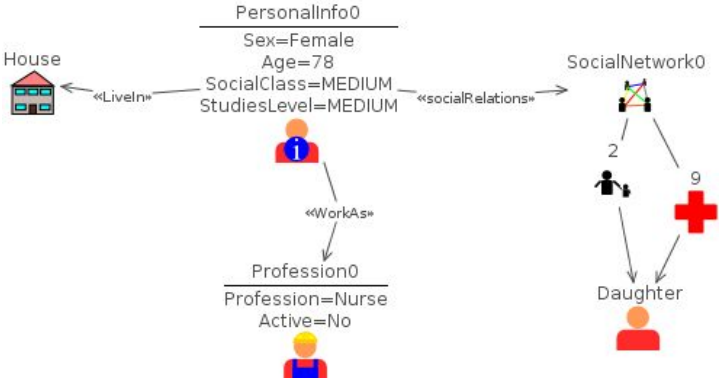


First step: Replicate the information given in the interview through the use of the SociAALML Editor's entity-relationship diagrams

- **Social** and **Human** diagrams model age, profession and social relationships for all agents involved
- **ADL**, **Activity** and **Sequential Tasks** diagrams constitute the behaviour of agents. These can be associated to specific times of the day and can have different conditions applied to them
- **Disease Spec**, **Symptom Evolution** and **Filter** diagrams model specific problems, their evolution throughout the day and effects on the agents
- **Building**, **Floor**, **Room**, **Devices**, etc... make up the diagrams of the scenario in which the simulations will take place
- **Simulation** diagrams tie everything together

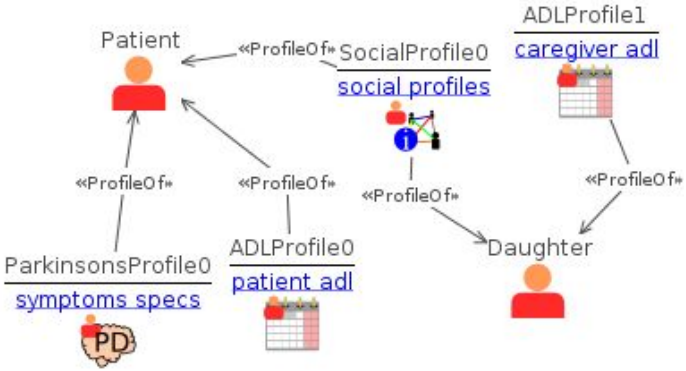
SociAALML Editor

Agent info and social profile



SocialSpecDiagrams:

- Personal and professional info.
- Social relationships.



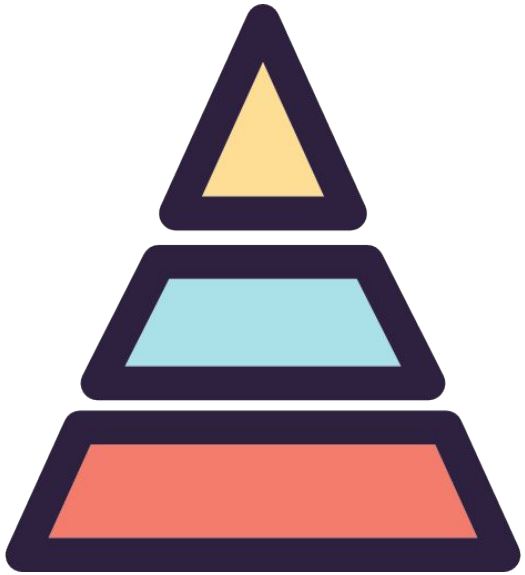
HumanProfileSpecDiagrams:

- Associate each agent to their daily activities.
- Specify symptom diagrams.

SociAALML Editor

Agent planning and environment interaction

From the more general planning to the specifics of each task.
Conditional planning can be performed.



ADLSpecDiagram

↓

ActivityDiagram

↓

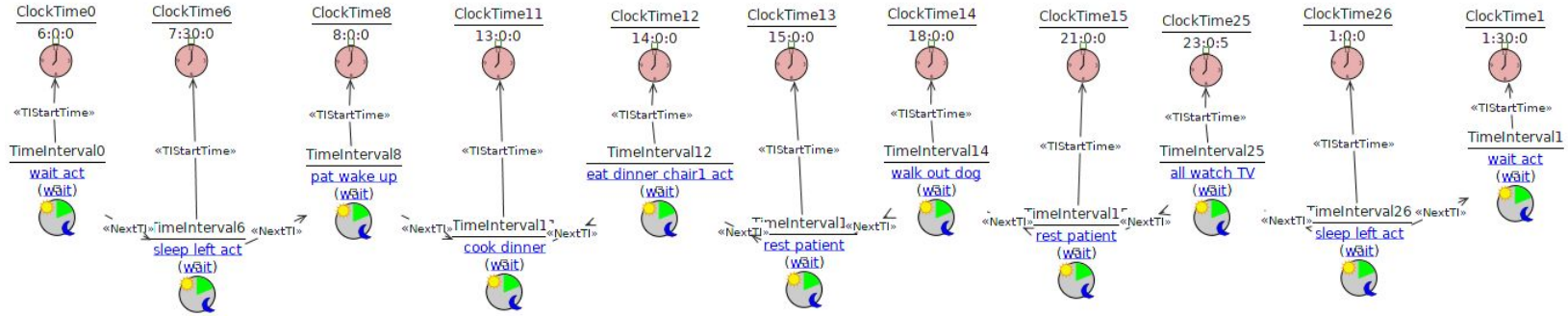
SequentialTaskDiagram

SociAALML Editor

Agent planning and environment interaction

ADLSpecDiagram:

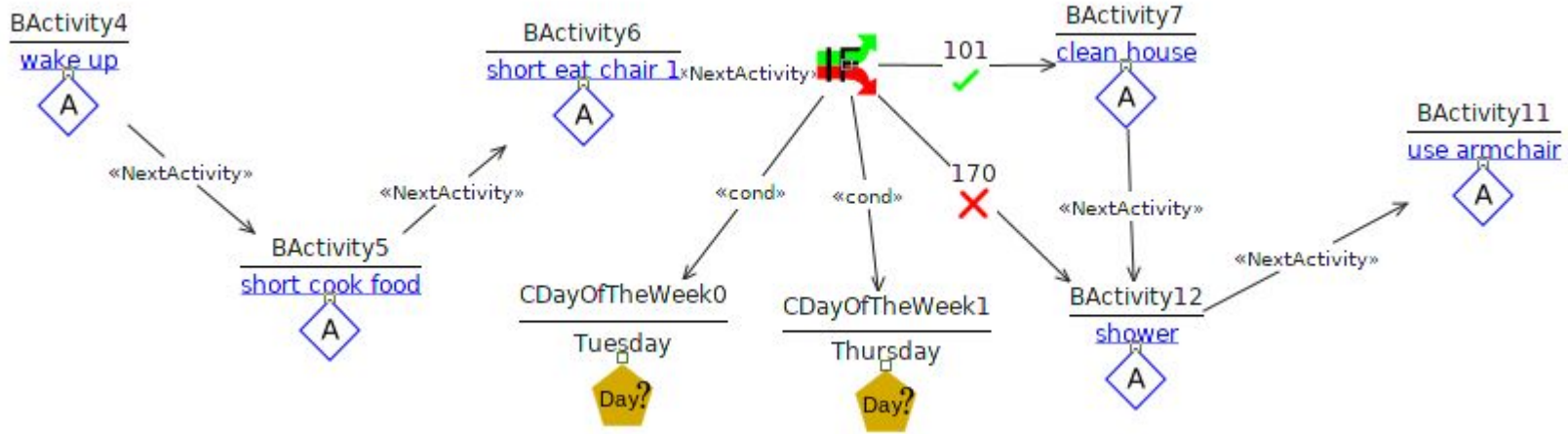
Assign **activity sequences** to **times of the day**.



SociAALML Editor

Agent planning and environment interaction

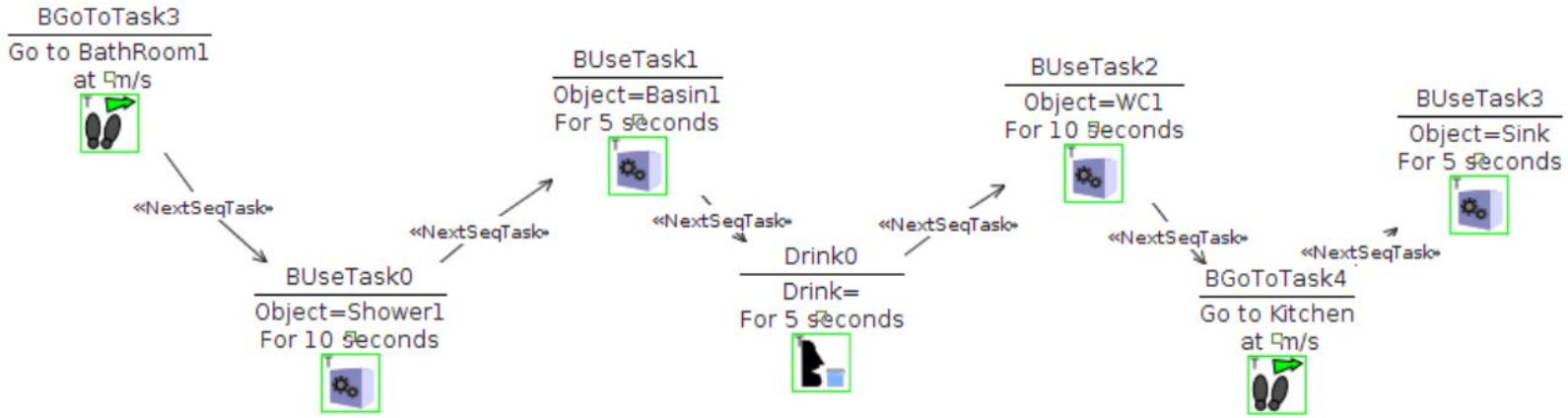
ActivityDiagram:
They contain **sequences of tasks** which can be **conditionally executed**.



SociAALML Editor

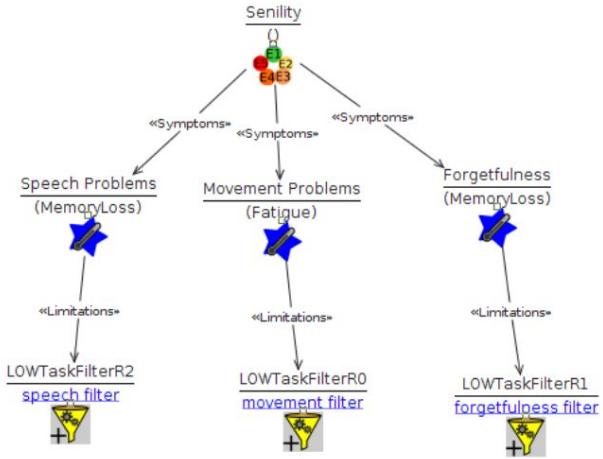
Agent planning and environment interaction

SequentialTaskDiagram:
Specifics of every activity. They contain info on **duration** and **object** of interaction.

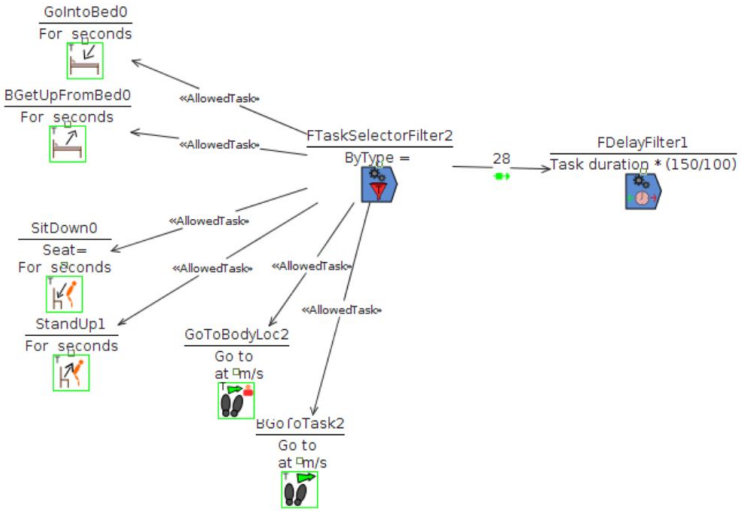


SociAALML Editor

Illness specifics



DiseaseSpecDiagram:
Tree of **symptoms** with **filters** applied to them.
Can also contain the **symptom evolution** across time or while taking/not taking medication.

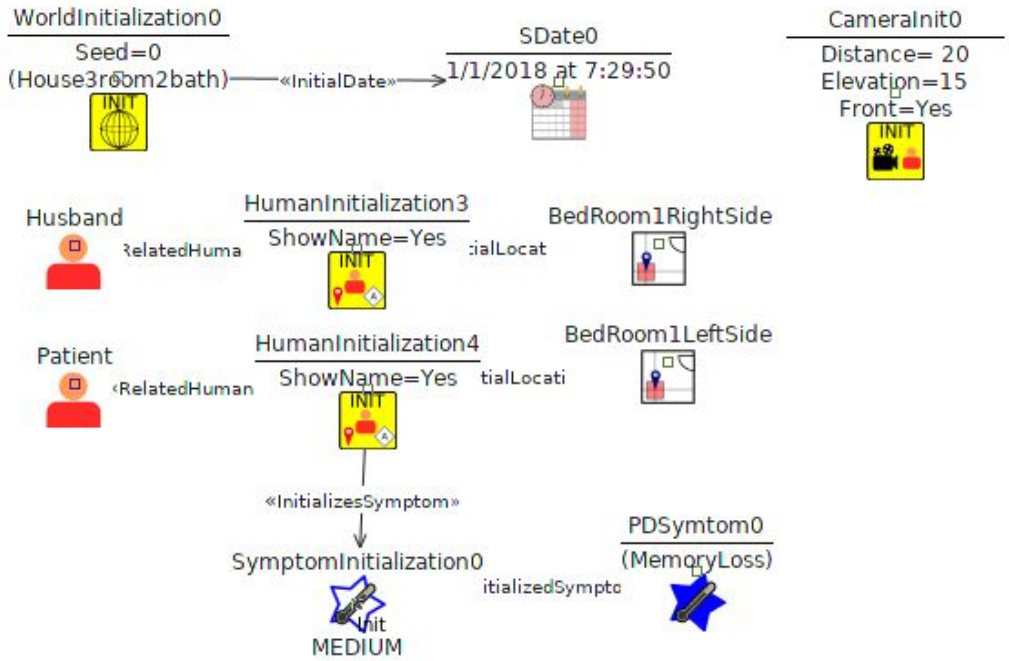


FilterDiagram:
Model the way in which **planning is modified** when under a symptom. Can also contain **conditionals.**

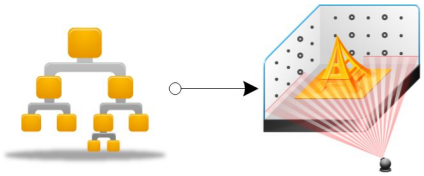
SociAALML Editor

Preparing simulations

For each interview **various simulations** are prepared.
Symptoms can be isolated or combined, **environment** can be changed. **Starting time** and date can vary along with **camera position** and other various features.



Simulate Scenarios
PHAT Simulator

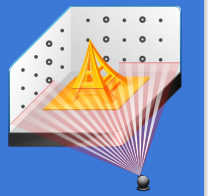


Once we have modeled the interview within the tool, we can **run simulations:**

- Enable, disable or mix **symptoms**
- Attach **virtual sensors** to gather data

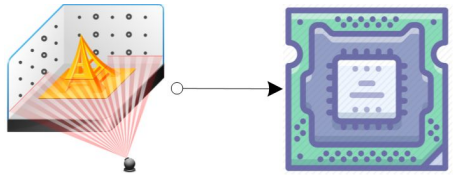
For this project we made complete **24-hour simulations** of a patient's day with varying symptoms.

PHAT Simulator



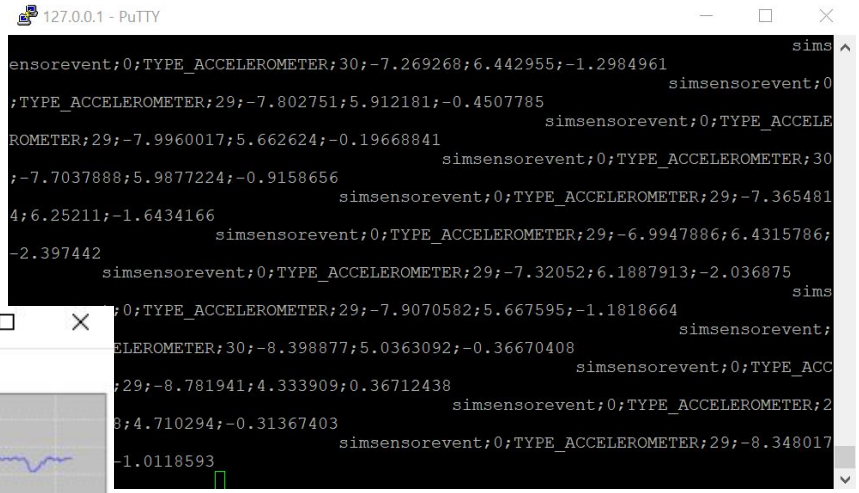
Experimenting

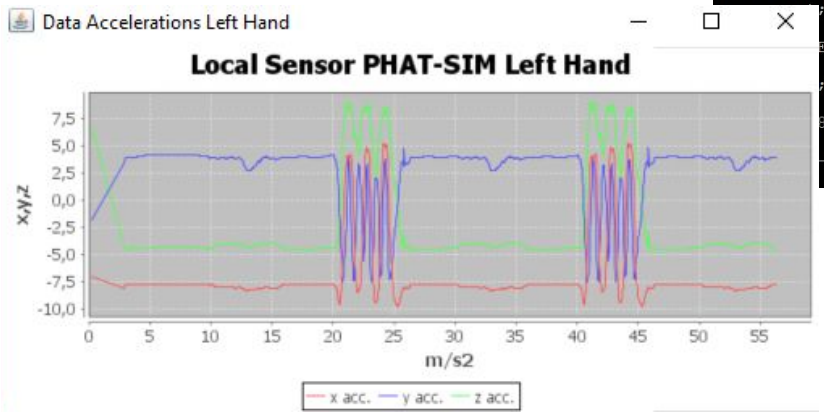
Gathering data from virtual sensors



We can attach **virtual hardware** to the scene, **view** and **collect** its **data**.

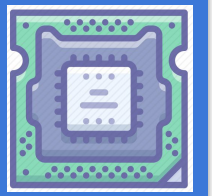
- Door sensors (Hall-effect)
- Presence sensors
- Accelerometers
- Cameras
- Even Android phones!





PoC Hardware Development

Helping caregivers rest at night



Problem:

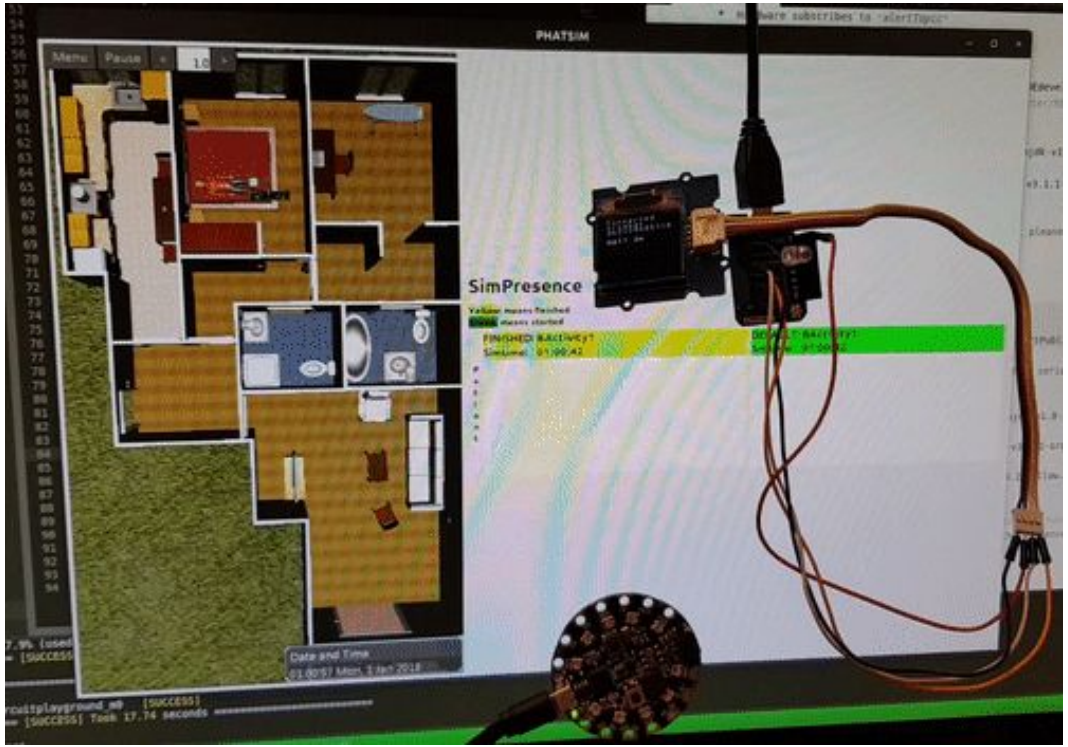
- It is reported by many caregivers that they have **trouble resting at night** because they have to be **aware of the patients** waking up to **go to the bathroom**.
- These patients **disorient** rather frequently and may be **unable to turn on the light** or **find their destination**.
- Adding to these they are usually elderly people with **mobility problems** who are **prone to falling**.

This could be **solved** with various **technological solutions** depending on the **level of the symptoms.**

- Presence-activated lights.
- Presence detection which lights the path to the bathroom.
- Caregiver warning system ⇒

PoC Hardware Development

Caregiver warning system



The system **detects** when the patient wakes up at night and **alerts** the caregiver.

- One **movement sensor** is located in the patient's room. In this case it's a virtual sensor.
- One device is used to warn the caregiver by the use of **lights** and a **speaking voice**.
- Other device is located outside the house to warn of an emergency which the caregiver did not attend.

Thank you!

Main project page

http://grasiagroup.fdi.ucm.es/aidendd

Github projects

https://github.com/Melkoroth/AIDEdevelopment

https://github.com/Melkoroth/AIDEdevelopmentHardware

jlpdiez,@ucm.es