

Received June 5, 2019, accepted July 3, 2019, date of publication July 11, 2019, date of current version August 6, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2927461

ePHoRt: Towards a Reference Architecture for Tele-Rehabilitation Systems

JORGE-LUIS PÉREZ-MEDINA¹, KARINA BEATRIZ JIMENES-VARGAS¹, LOUIS LECONTE², SANTIAGO VILLARREAL³, YVES RYBARCZYK¹, AND JEAN VANDERDONCKT³

¹Intelligent and Interactive Systems Laboratory, Universidad de Las Américas, Quito 170125, Ecuador

²Ecole Normale Supérieure, 94230 Paris-Saclay, France

³Louvain Research Institute in Management and Organizations, Université catholique de Louvain, 1348 Ottignies-Louvain-la-Neuve, Belgium

Corresponding author: Jorge-Luis Pérez-Medina (jorge.perez.medina@udla.edu.ec)

This work was supported by the Universidad de Las Américas, Quito, Ecuador, and by the Secretaría de Educación Superior, Ciencia, Tecnología e Innovación (Senescyt)–INEDITA under Grants: SIS.JPM.18.01 and 20190020CI.

ABSTRACT In recent years, the software applications for medical assistance, including the tele-rehabilitation, have known a high and a continuous presence in the medical area. The ePHoRt is a Web-based platform for the remote home monitoring rehabilitation exercises in patients after hip replacement surgery. It involves a learning phase and a serious game scheme for the execution and evaluation of the exercises as part of a therapeutic program. Modular software architecture is proposed, under the patient perspective, to be used as a reference model for researchers or professionals who wish to carry out tele-rehabilitation platforms, and to guarantee security, flexibility, and scalability. The architecture incorporates two main components. The first one manages the patient's therapeutic programs taking into account two principles: 1) maintain loose coupling between the layers of the framework and 2) Don't Repeat Yourself (DRY). The second one evaluates the performed exercises in real time considering an independent acquisition mechanism for the patient movements and two artificial algorithms. The first algorithm allows evaluating the quality of the movements, while the second one allows assessing the levels of pain intensity by recognizing the patient's emotions when performing the movements. Details of the components and the meta-model of the architecture are presented and discussed considering their advantages and disadvantages.

INDEX TERMS Software architecture, web-based platform, tele-rehabilitation, gesture recognition, emotion recognition, affective computing.

I. INTRODUCTION

The implementation of a therapeutic program via an interactive multimedia web-based platform is required for tele-rehabilitation which is an specialized field of telehealth.

To ensure the success, the flexibility and the scalability of a web-interactive platform a previous design of a software architecture is required to guides the development process of its components. The architecture offers a solid base and overview of the product [1] and incorporates software patterns. Don't repeat yourself (DRY) is a well-used base in software engineering that seeks to avoid the redundancy of using software patterns [2]. This principle includes the use of abstractions, normalization of data and separation of concerns

in simple and unambiguous pieces of software [3]. Patterns provide designers with a vocabulary to express architectural visions, presenting small pieces of software for designers in order to communicate more effectively [4].

Unfortunately, in the context of tele-rehabilitation systems, academic and industrial advances [5]–[7] do not offer sufficient the detailed descriptions of the proposed technological infrastructures or the technological definitions to carry them out. Therefore, it is not possible to re-use the proposed methods.

ePHoRt stands for “e-Platform for Home Rehabilitation” [8] for helping patients after hip replacement surgery. We adopt the *User-Centered agile Development (UCD)* approach [9] to specify and to design the concepts, the meta-model's elements and the components of the software architecture in a flexible and efficient manner.

The associate editor coordinating the review of this manuscript and approving it for publication was Wajahat Ali Khan.

This work presents a modular software architecture to guarantee flexibility and scalability. It is a vital step towards building and maintaining platforms because it is a bridge between the requirements and the design phases [10]. It manages the complexity of a problem by breaking them down to smaller manageable modules [11] and by determining the structures that will shape the software system and the way in which its elements communicate to each other. Our architecture is based on a modular structure to support tele-rehabilitation programs and to allow an easy adaptation of the system for the rehabilitation of other motor disabilities (for example, stroke).

We also incorporate two intelligent modules to evaluate the quality of the movements performed by the patient and to get a reference of the levels of pain that the individual suffers. Currently, the capture of the movements is done through a Kinect device; however, the flexibility of the proposed architecture allows us to incorporate others mechanisms of acquisition. For its part, the recognition of the emotions and especially pain ensures that the patient can face some issues and it is necessary that the physiotherapist is informed in order to carry out supplementary studies for the patient's recovery. Also, we can warn that the patient may have some complication, in this case the physiotherapist may carry out supplementary studies that allow knowing the patient's health status.

The remainder of this paper is organized as follows. Next section presents the related work. Third section introduces the main functionalities of the *ePHoRt platform*. Fourth section presents the software architecture. Section five introduces the meta-model. Finally, the paper is discussed and future work is proposed.

II. RELATED WORK

In the domain of tele-rehabilitation systems, KiReS [5], [6] is the closest solution to the *ePHoRt* project. Nevertheless, recent contributions do not present details that facilitate the reproduction of a tele-rehabilitation solution. Our main contribution is to give details of an architecture for designing a tele-rehabilitation systems. It can also be a reference model for future work.

A framework for remote motion and function assessment (Tele-MFAsT) using internet connected Kinect cameras, which stream their data into a 3D virtual environment was presented in [7]. It describes how the architecture allows data streaming (video, depth, skeleton and audio) from Kinect cameras connected to the network. The solution can be used by the therapist to remotely monitor a patient through an ecologic 3D virtual environment that enhances a feeling of telepresence, which facilitates the assessment of kinematical parameters (trajectories of movement and articulation angles) of the patient's performance.

Su *et al.* described a system based on Kinect for permitting home-based rehabilitation (KEHR) using a Dynamic Time Warping (DTW) algorithm and fuzzy logic to ensure the effectiveness and safety of home-based rehabilitation without

the immediate supervision of a physician. Using KEHR, the patient first performs a prescribed exercise in the presence of a health care professional. The exercise is recorded as a baseline for evaluating the patient's rehabilitation exercise at home, and those evaluations are used as a reference for the patients to validate their exercise performance and to prevent adverse events. A summary report of the outcomes may also be automatically uploaded to a cloud setting for physicians to monitor the patient's progress and to adjust the therapeutic program.

The current studies use gesture recognition algorithms as a mechanism for evaluating movements. However, these works do not consider face recognition as a mechanism to automatically detect the emotional status of the user, such as a feeling of pains, when performing the exercises. Virtual rehabilitation environments may afford greater patient personalization if they could harness the patient's affective state, including pain states [12].

Facial expression communicates emotion, intention, physical state, and regulates interpersonal behavior. Automated Face Analysis for detecting, synthesizing, and understanding of facial expression is a vital focus of basic research [13]. The context of our research around the *ePHoRt* platform is aligned with efforts to offer intelligent systems to allow objective measurement and to monitor the pain from the interpretation of facial expression. [14] presents a review of the literature on affective body expression perception and recognition. Emotions, and specifically pain levels, can be assessed through different approaches. Pain expressions are different for one person to another. Certain patients are very expressive regarding their pain, whereas others exhibiting minimal discomfort. Pain is often poorly assessed, which can result in improper treatment [15].

An automatic and reliable assessment of the onset, intensity, and pattern of occurrence of pain would help to ensure a more customized therapeutic program. Given these potential medical implications, increasing efforts are underway to develop intelligent systems to enable objective measurement and to monitor the pain [15]. In [16], authors propose a review of the state-of-the art in automatic, objective assessment of pain from video or camera input. The review discusses current challenges and, among them, pain intensity for research and clinical use.

The possibility of automatically detecting low movement related self-efficacy (MRSE) with the long-term aim of designing affect aware technology for physical therapeutic program coaching was studied in [17]. Pain typically is measured by patient self-report, but self-reported pain is difficult to interpret and may be impaired or in some circumstances not possible to obtain. Reliable and valid measures of pain are essential for conducting research on chronic pain. These measures have been applied again with periods of 2 weeks, 2 months and after the end of treatment. Changes in pain intensity and its stability throughout the treatment were studied. The results obtained suggest that the individual 0-10 ratings of pain intensity have a sufficient psychometric strengths

to be used in the investigation of chronic pain [18]. Currently, the most rapid and reliable means of assessing pain in physical rehabilitation is through the visual analog scale (VAS) [19]. This scale present good reliability to assess disability in patients with chronic musculoskeletal pain [20].

Furthermore, the use of VAS compared with other scales that measure physical function and hip pain like WOMAC [21] and SF-36 [22], has been proved to be equal or even more efficient and sensitive to the above mentioned scales [23]. Likewise, a recent study shows that there is no a clinically relevant difference between the traditional paper-based VAS assessment and VAS scores obtained from laptop computer and mobile phone-based platforms [24].

Rybarczyk *et al.* presented a characterization of the existing studies and approaches to build a physical tele-rehabilitation system [25]. These works do not describe the technical details of their architectural solutions which difficult the realization of new implementation of physical rehabilitation. Likewise, the architectures of these systems do not consider an artificial intelligent module that allows for a rigorous assessment of the rehabilitation exercises. In addition, any contribution is evidenced regarding the recognition of facial gestures and pain assessment. ePHoRt includes both an assessment module of the quality of the movement and the emotional state of the patient. Our work also, proposes a more advanced approach, since we developed and implemented an intelligent module to capture and evaluate the intensities of pain that a patient suffers when performing the experiments. Thus, physiotherapists will be informed of pain activity throughout the therapeutic program.

III. THE EPHORT PLATFORM

Considering the patient's tasks, the main features of the platform are concentrated around a core and a real time evaluation component. The syntax of the task tree model was inspired by the *Concur Task Trees (CTT)* notation described in [26]. In our case, icons may vary due to the toolkit used. We have adopted this nomenclature because it supports the design of interactive applications for user interface model-based design. It generates a tree graphical representation of the hierarchical decomposition of the tasks provided by the platform.

Three tasks must be carried out by the patient with the applications: learning general recommendations, performing exercise and communicating. Learning general recommendation implies the patient reading divers resources in order to understand the therapeutic program. Performing exercises implies following a set of instructions to perform the exercises. Communicating is just a way of sending messages between the patient and the therapist.

A. THE THERAPY RESOURCES

The therapy resources allows for empowering the patient [27], who can perform autonomously a set of rehabilitation activities from a program elaborated by the health professionals (e.g. physiotherapists). Many functionalities are implemented in the platform in order to support the patients in their

self-recovery process. The hierarchical links between these functionalities are illustrated in Figure 1.

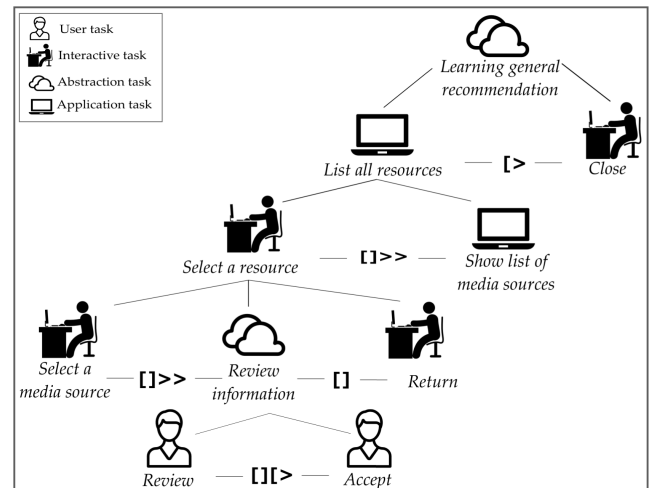


FIGURE 1. Learning task diagram.

B. LEARN GENERAL RECOMMENDATIONS

For each rehabilitation process, patients are provided with a set of multi-modal information in form of text, audio and video [28]. This information allows patients to be trained to use the platform and to understand the procedure to complete the therapeutic program.

The task model of Figure 1 shows the activities that must be carried out by the patient for learning general conditions. The patient can initially view a list of learning resources, select one of them and display an additional list of associated multimedia files. Then, patients can choose the modality (vocal, auditory and visual) of any learning resource according to their preferences, review the information or return to the general list of learning resources.

C. PERFORM REHABILITATION EXERCISES

A rehabilitation session is planned by the physiotherapist and it is composed of a set of exercises. An exercise is based on the composition pattern [29] as shown in the top of the Figure 2. This means complex exercises can be constructed from simple exercises.

For instance, “hip abduction”, “slow flexion of hip and knee”, “hip extension” and “frontward, sideways, backward” steps sequence represent instances of the leaf class as shown in Figure 2. This means that they are simple exercises. Then, a first complex exercises can be defined from “hip abduction” and “slow flexion of hip and knee”. Also, another complex exercise may include the first composition and also the exercises: “hip extension” and “frontward, sideways, backward” steps sequence.

The task model of Figure 3 shows the activities that must be carried out by the patient in performing an exercise. The patients must answer a preliminary questionnaire regarding their health status. It has to be processed by the platform

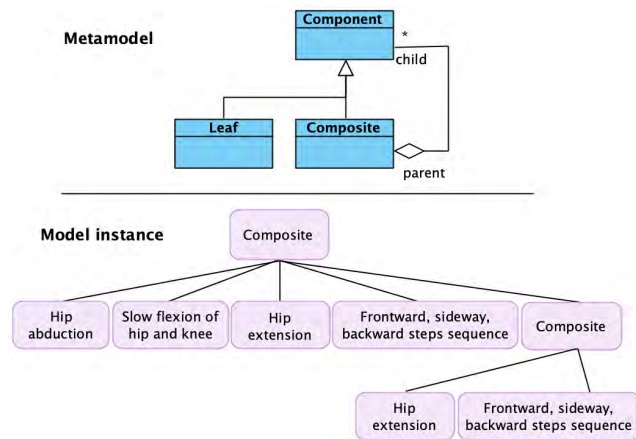


FIGURE 2. The composition pattern meta-model and an instance model of example.

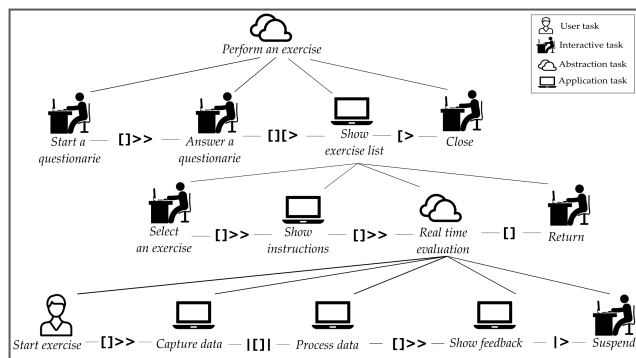


FIGURE 3. Perform a rehabilitation exercise task diagram.

to authorize or disavow the patient to start a rehabilitation session. The preliminary questionnaire allows the physiotherapist getting relevant information from the patients like recent surgeries, levels of pain, other health problems.

The answers will help the physiotherapists to determine if the patients can perform the exercises autonomously or if they need a face-to-face consultation. First, if the patient is authorized to start a rehabilitation session, the system activates the execution interface of the exercises. An exercise consists of a set of series with several repetitions according to the reeducation plan predefined. There is a short break between each repetition to provide the patient with a feedback about the quality of the movement. The trials are assessed in terms of range of motion (ROM) and compensatory movements. Once a trial is completed, the patients receive a graphic-based feedback regarding their performance.

For each exercise, the patient can see a set of instructions to perform the movements correctly. Once the patient starts an exercise, an avatar is displayed in a virtual environment. The interface invokes an acquisition module and a game associated with the activity that must be performed. The patient's movements are mapped in real time with the avatar's motion. Also, a progress bar provides the patient with the number of trials performed, while the score indicates the number of

trials correctly performed (e.g. each time the patient's leg gets the right range of motion and shoots a virtual ball). The movement quality is assessed in real time by artificial intelligence algorithms and the results are graphically presented to the user. The graphical-based feedback includes green and red color to describe how well the patient performed a movement. The red color represents a bad amplitude of the limb movement, while the green color indicates a correct one movement. Textual descriptions are included to give the patient a better understanding of the performed movement and the current information regarding the serial number and the repetitions. Figure 4 (right side) shows an example of the graphical feedback displayed to the patient. In this representation red color show a bad amplitude of the patient limb movement.

A progress bar and a score is added in order to observe the progress of the exercise and to know how well the exercise was performed. In addition, the patients can suspend a rehabilitation session at any time. In this case, they have to provide the physiotherapist with a reason to abort the session.

D. COMMUNICATING BETWEEN THE STAKEHOLDERS

Patients and physiotherapist can communicate to each other through a message exchange module. The task of communicating involves reading and sending messages between physiotherapist and patient (Figure 5). The message can be text-based and/or audio-based.

IV. SOFTWARE ARCHITECTURE

A. THE CORE

Considering the modular software architecture described in Figure 6, the application is built in *Django 2.0.5*. This web framework uses the Model-View-Controller (MVC) design pattern and works under the two central principles: (1) maintain loose coupling between the layers of the framework and (2) the Don't Repeat Yourself (DRY) [30]. The MVC pattern separates application functionality into three kinds of components. The *model* represents the knowledge and has logic to update controller if its data changes. The *view* represents the visualization of the data. The *controller* determines the data flow into model object and updates the view whenever data changes. It keeps *view* and *model* separate and acts on both.

B. REAL TIME EVALUATION

Two additional modules are created for the real time evaluation. The first one for processing data with a RESTful API and the second one for acquiring movement data with a client application. They communicate to each other for processing the movement data and for giving the results using JSON data format. The RESTful API is developed using *Django Rest Framework 3.8.2*.¹ It is a powerful and flexible toolkit of Django to build RESTful Web Services [31].

The client application is built with *Kinectron 1.4.2* and *Processing 1.4.8*. Kinectron is an open source tool to capture the

¹Django Rest Framework - is a powerful and flexible toolkit for building Web APIs. <https://www.django-rest-framework.org/>

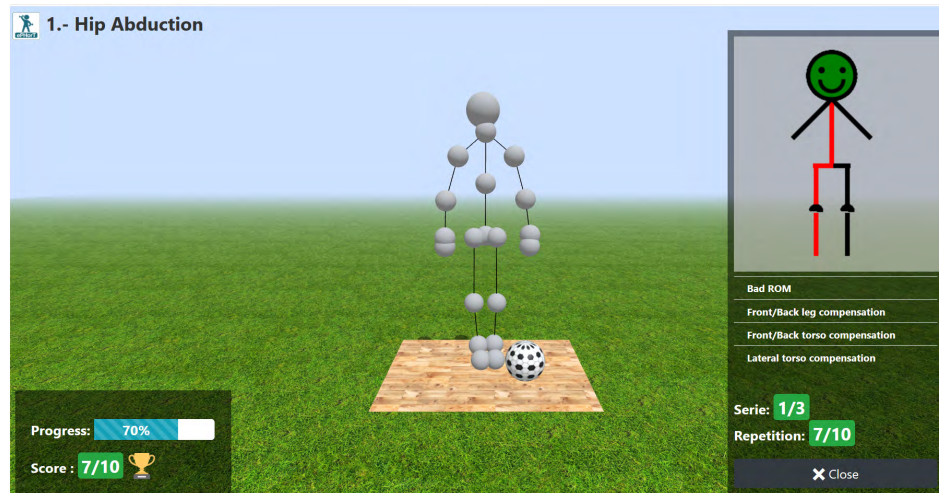


FIGURE 4. The patient's avatar user interface.

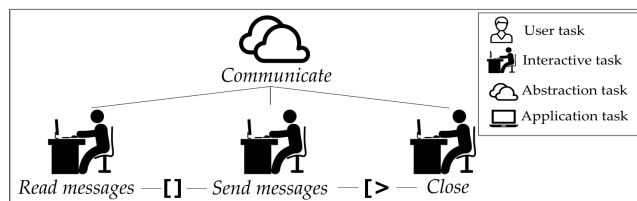


FIGURE 5. Communicate task.

movement data. It has two components: a server that broadcasts Kinect data, and an API that receives Kinect data [32]. A Kinect v2 camera was used to capture the movements made by the patient. It facilitates the extraction of the relevant characteristics for the recognition of the movements made by the patient at the time of performing the rehabilitation exercises. A cross-browser JavaScript Library, called *Three.js* ², is incorporated to create and to display an animated 3D computer graphics avatar in a web browser. A serious game makes use of the avatar to assist the patients in the development of their tele-rehabilitation movements (see Figure 4).

Eventually, the data of the skeleton and the video are captured in order to be processed by the AI algorithms, through a Restful API. An HTTP POST request is used to save the data and to process it after a repetition. In order to have a knowledge about the pain of the patient, an HTTP PUT request is used to inspect the face of the patient each second during the movement. Then, after a repetition, an HTTP POST request is used to save a string with those results.

Additional technologies, such as *Bootstrap 4.1* ³ and *JQuery 3.1* ⁴ are used to build the interface. *Bootstrap* is a CSS framework that provides a characteristic of responsive

web design. It allows developing interfaces compatible with different kinds of browsers and terminal platforms [33]. *JQuery* is a Javascript library which has CSS selector, flexible animation system and event system, rich plugins and the solutions on browser compatibility issues.

V. THE EPHORT META-MODEL

This section describes the details of the ePHoRt meta-model. It presents a conceptual guide that allows understanding the details of the architecture. A compliant UML class diagram is elaborated to describe the main concepts. The class diagram is the most common in modeling object-oriented platforms and it is used to model the static design view of the platform [34]. The meta-model is divided into seven packages for a better comprehension and usage (Figure 7).

The *Accounts* package contains the user's information. A user can have three roles: administrator, patient and physiotherapist. Patients can send messages, can stick a previously predefined therapeutic program, and can only have one physiotherapist with their assigned therapeutic program. Physiotherapists can manage patient accounts, can send messages and can define customized rehabilitation plans.

The *Therapeutic programs* package allows the physiotherapists planning a therapeutic program considering a set of stages. A stage can have several surveys, resources and exercises.

The stage state machine is illustrated in Figure 8. It shows the transitions between states and presents the point where the sequence of states start or end [35]. A stage object has three states (active, inactive, completed). It is *inactive* when it is created for the first time. It is *active* if the stage is the next one to be performed or if the patient did not complete it. It is *completed* if the stage was fully performed by the patient.

The *Surveys* package allows retrieving the patient's answers from a static survey. Questions were established by the physiotherapists considering their expert knowledge as a

²ThreeJS - JavaScript 3D library <https://threejs.org/>

³Bootstrap - The most popular HTML, CSS, and JS library in the world. <http://getbootstrap.com/>

⁴JQuery - is a fast, small, and feature-rich JavaScript library. <https://jquery.com/>

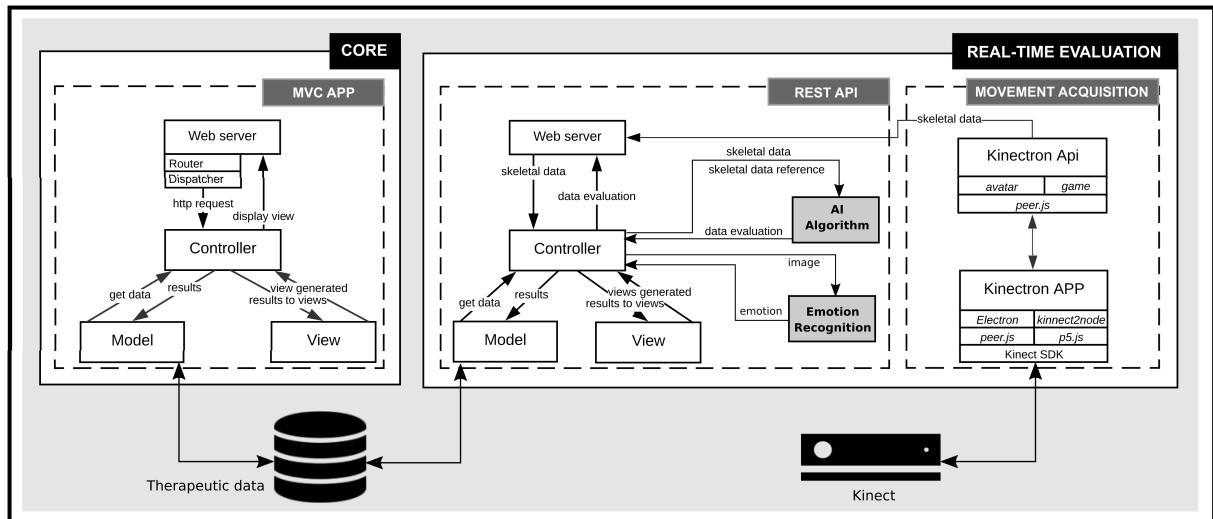


FIGURE 6. ePHoRt software architecture.

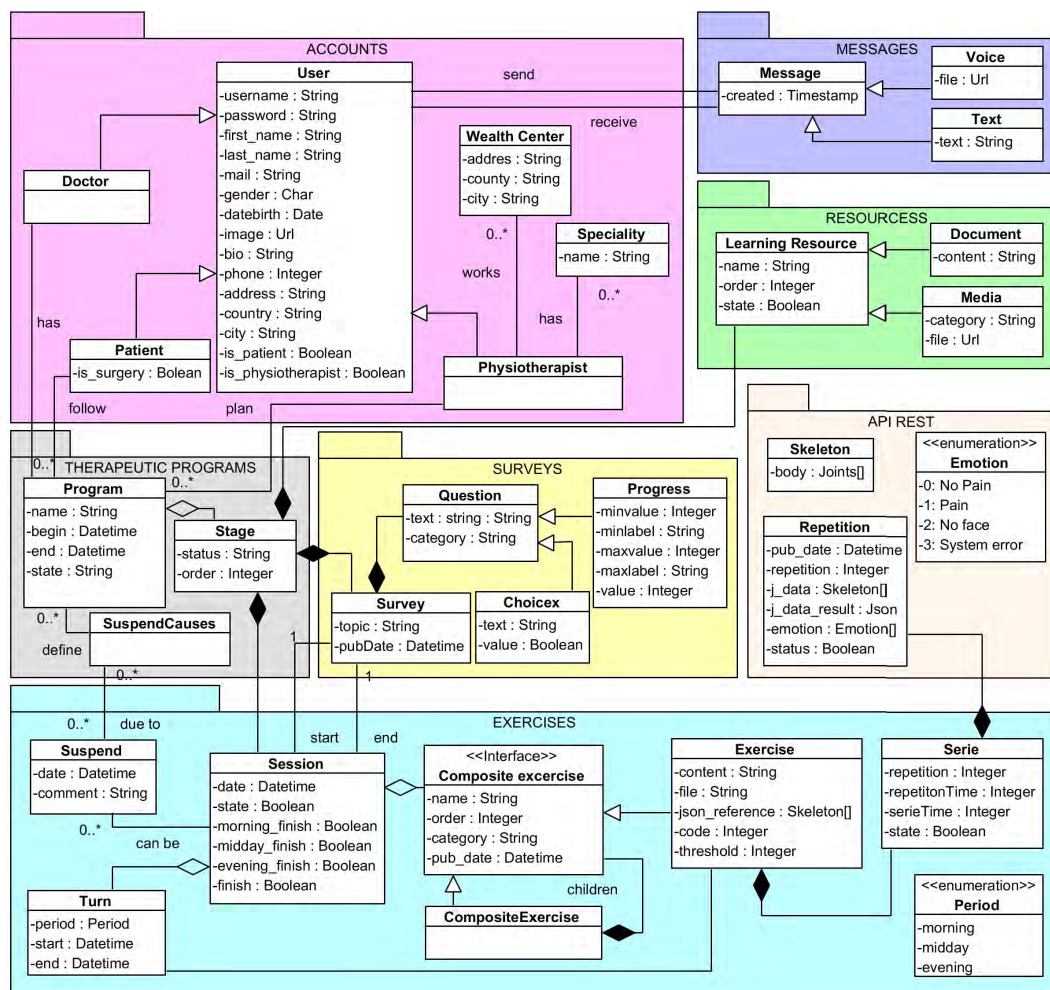


FIGURE 7. ePHoRt class diagram.

way to get the patient's physical condition to suspend or to continue their patient's plan. The answers are validated before performing the exercises through a model to allow or deny the continuation of a therapeutic program.

The static survey is shown in Figure 9. Three questions about the pain, the skin and the edema are required to be answered by the patient. The self-assessment of pain levels is based on the patient's subjective perception. The answers

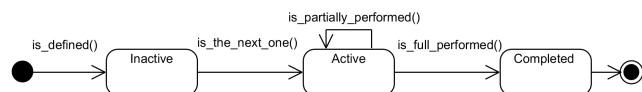


FIGURE 8. State machine for stage object.

can have several combinations. The pain question is measured from 1 to 10 and is evaluated considering three options: 1) less than or equal to six, 2) equals to seven or 3) greater than or equal to eight. The other two questions about the skin and the edema are evaluated considering the three options. Also, for the first question a Table 1 with the evaluated ranges of pain are provided. It will serve as a guide for the patient at the time of selecting the level of the pain before performing the exercises.

TABLE 1. Pain scale guide.

Pain intensity	Description
≤ 6	I have no pain or little pain; however, I can perform out the activities such as: getting up, walking, climbing stairs.
$= 7$	I have pain in the hip, but I can perform with a difficulty the activities such as: getting up, walking, climbing stairs.
≥ 8	I have a hip pain that does not allow me to perform the activities such as: lift the leg, walk, climb stairs.

The patient requires to select at least one option for each question to answer this survey. In the first and third question the patient can choose just one option while in the second one can choose multiple ones. The first and the third one can have three possible options while the second one can have seven possible options, as show in Table 2. Thus, a total of $3 * 7 * 3 = 63$ instances are possible, as shown in Appendix A. Then, the options were validated manually with the physiotherapist as a ground truth criteria.

TABLE 2. Possible options for the skin question.

O1	O2	O3
True	True	False
False	True	False
False	False	True
False	True	True
True	True	False
True	False	True
True	True	True

After that, the answers are processing to allow or deny the continuation of the therapeutic program by considering the following exclusion conditions as criteria for the answer validation. This criteria was provided by the physiotherapist as a categorical value. A simple Recursive Partitioning And Regression Trees (CART) algorithm [36], was run over all the answers to create a simple binary classification tree. Each root node represents a single input exclusion condition and a split point on that condition. The leaf nodes of the binary tree contain an output Boolean variable which is used to make a prediction. Code is presented in Appendix B. This is important in medicine because pain is may be a warning sign of serious health conditions, so, for this reason it necessary to inform the physiotherapist that a physical examination of the patient may be required.

As a result, the most relevant exclusion conditions were identified by using *entropy* in order to built a classification decision tree model. *Entropy* is an information-theoretic measure of the uncertainty contained in a set, due to the presence of more than one possible classification [37]. The resulting model (Figure 10) represents visually the relevant exclusion conditions to determine if the patient can or cannot start a rehabilitation session.

The *Resources* package describes the structure of the learning resources (LR). LRs can be created by the physiotherapists in three formats: video, audio or text document.

FIGURE 9. Preliminary survey.

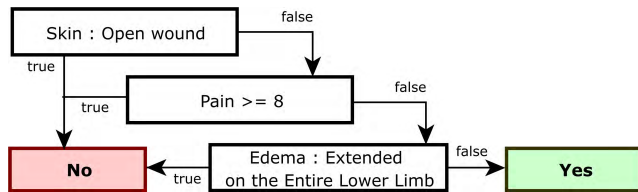


FIGURE 10. Decision tree to validate the preliminary survey.

The patients can choose the most appropriate for them considering their health condition.

The *Exercises* package manages sets of exercises with session objects. Each session object has two stages: active and inactive (Figure 11). It is active if the patient is able to perform the set of exercises and it is inactive if the patient is not able to perform it. The exercises are handled by imitation of the composite design pattern [38]. Exercises can be defined with a set of instructions and several sequences by the physiotherapist. Each exercise requires a set of instructions, a set of series, a number of repetitions and a rest time to be determined.

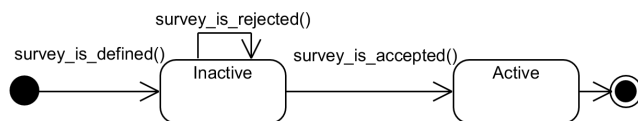


FIGURE 11. State machine for session object.

In order to perform an exercise routine, the patient has to check the instructions and start a set of series as it is shown in Figure 12. Series have some trials. In each trial, a start command is displayed to guide the patient. The body movements tracked and used to animate a virtual avatar. As previously mentioned, the data of the skeleton and the video are processed for the real time evaluation (Figure 4). When, a patient finishes an exercise, a stop command is shown also in the patient's screen. This process is repeated until the patient completes all the trials of the series. Then, the average statistics are presented.

Four types of exercises are currently available: “hip abduction”, “slow flexion of hip and knee”, “hip extension”, and “frontward, sideways, backward” steps sequence. In Hip abduction, the leg's movement is evaluated in the sagittal plane by identifying one working angle and three compensation angles (Figure 13). In slow flexion of hip and knee, the leg's movement is evaluated by identifying two working angles and four compensation angles (Figure 14). In hip extension, the leg's movement is evaluated by identifying the frontal and sagittal compensation (Figure 15). In frontward, sideways, backward, the leg's movement is evaluated by identifying the frontal and sagittal compensation (Figure 16).

The *API REST* package works with the exercises packages and contains the intelligent modules to assess the quality of the movements performed by the patients and also, their emotions. It has a couple of RESTful Web Services to process

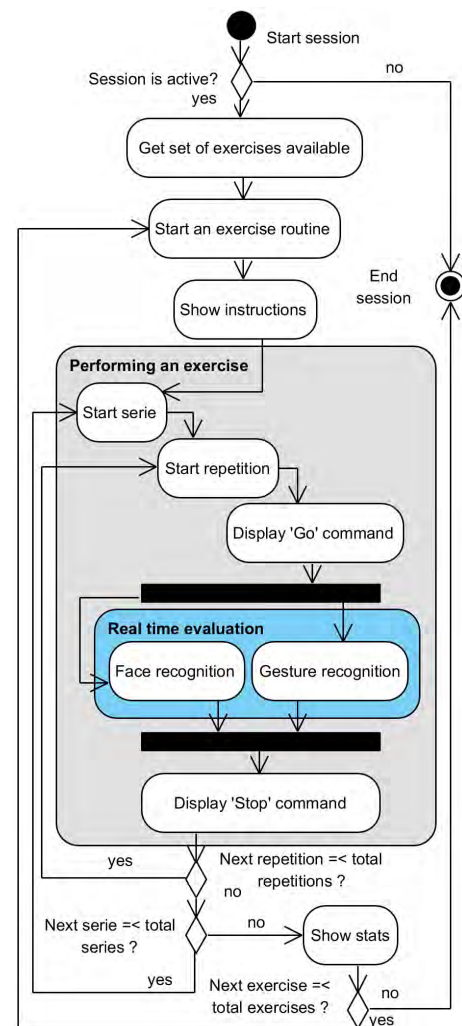
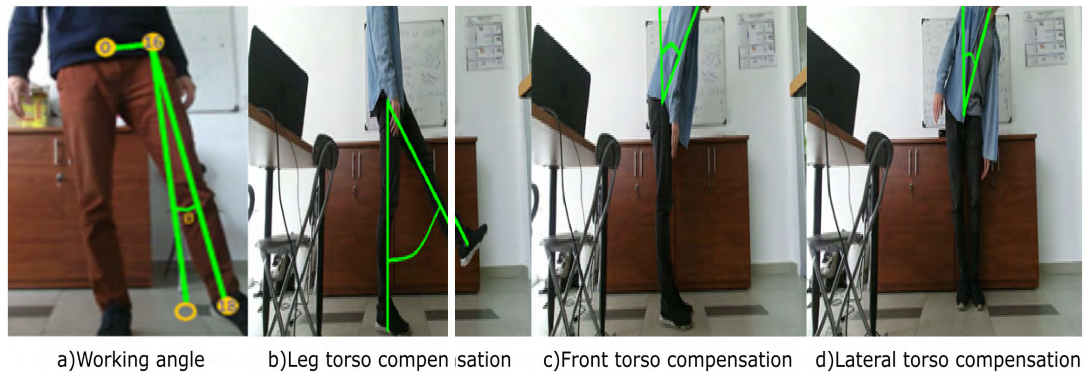
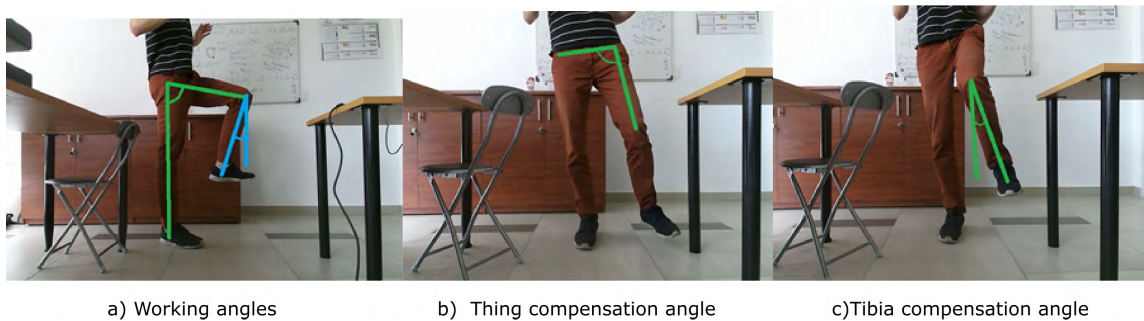
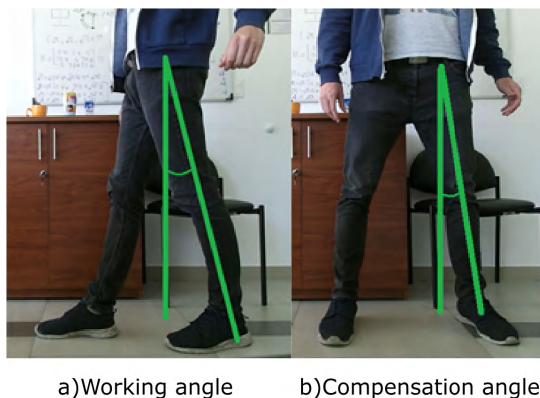


FIGURE 12. Performing an exercise routine algorithm.

the skeleton and the images frames with artificial intelligence algorithms in each repetition.

In the first service, the skeleton is processed to evaluate if the patient is exercising correctly and the result is sent back as a json structure with a set of angles. The evaluation of the movement quality is carried out by an AI algorithm based on the DTW method. Details of this algorithm are presented in [39]. This algorithm processes the skeleton data and provides the user with a discriminative evaluation of the accuracy of different body angles as a feedback result. Each angle has a type, a DTW metric which refers how close the movement is to the reference movement, a textual message and a color light. The light refers to the feedback rating scale and can be 1=red, 2=yellow or 3=green. For Example, if a patient performed a *hip abduction*, a feedback is returned by the system in a json format (Listing 1) and presented to the user for a graphic way (Figure 4) at the end of the repetition.

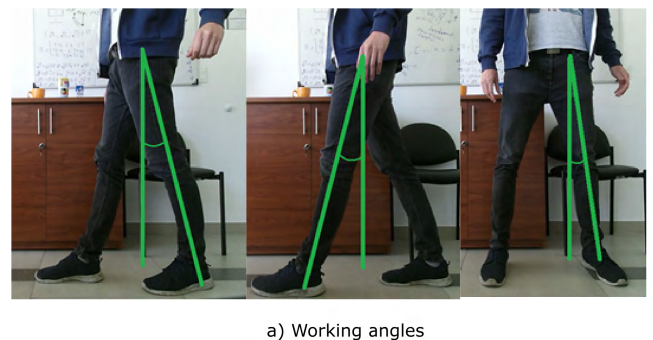
The recognition module compares the DTW cost of each described angle (ROM and compensations) to an empirically defined threshold (Figure 17). If the distance calculated

**FIGURE 13.** Hip abduction exercise.**FIGURE 14.** Slow Flexion of Hip and Knee exercise.**FIGURE 15.** Hip extension exercise.

by the DTW is higher than the threshold, the angle of the movement is classified as wrong. Since different DTW run in parallel (one for each specific working and compensation movements), it is possible to provide the patients a detailed feedback of their errors.

In the second service, the image frame is processed every second to determine if the patient feels pain. A Support Vector Machine (SVM) algorithm was trained to detect the face and to process it based on the following criteria:

- (i) **0** - the patient does not feel pain.
- (ii) **1** - the patient feels pain.

**FIGURE 16.** Frontward, sideways, backward exercise.

- (iii) **2** - the patient face cannot be processed.
- (iv) **3** - the patient emotion cannot be detected due to other factors. For instance, the quality of the image frames.

As a result, a JSON format (see Listing 2) is obtained at the end of a repetition. This information is presented to the patient in two graphical representations. The first representation display an array with the values 0-3 to specify the result of recognition based on the criteria explained above while the patient performs the exercises (see Figure 21). The second display an array of values from 0-100 with the intensity pain level (see Figure 22).

The intensity of the pain suffered by the patient is carried out by an emotion recognition program written in python.

```

1  {"Movement": "Hip Abduction",
2   "Threshold": 50.0, "angles":
3   [
4     {"type": "angusefull",
5      "DTW": 232.3711172813716,
6      "message": "Bad ROM",
7      "light": 1},
8     {"type": "angcompensationleg1",
9      "DTW": 9.994705773630727,
10     "message": "No leg compensation",
11     "light": 3},
12    {"type": "angcompensationtorso1",
13     "DTW": 5.888015792606099,
14     "message": "No frontal torso compensation",
15     "light": 3},
16    {"type": "angcompensationtorso2",
17     "DTW": 923.1000110715414,
18     "message": "Sagittal torso compensation",
19     "light": 1}
20  ]}

```

Listing 1. Result JSON Structure's example.

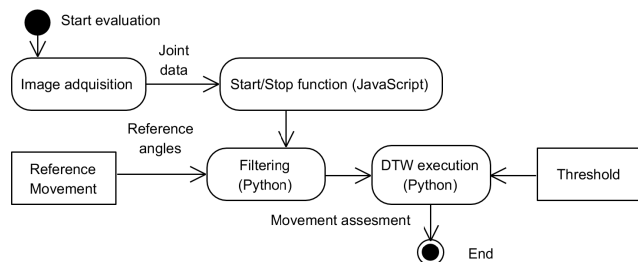


FIGURE 17. Scheme of the Data Processing of Gesture Recognition.

```

1  {
2   "id": 1,
3   "pain": [2,2,2,2,1,1,1,1],
4   "pain_intensity": [0,0,0,0,45,50,70,75],
5  }

```

Listing 2. Gesture recognition result JSON Structure's example.

A machine learning approach based on a SVM algorithm was trained on several datasets to predict the emotion of a given face picture. The Female Facial Expression (JAFPE)⁵ database [40] has been mainly used to training our algorithm, but it has been complemented by the Caucasian, Asian, Hispanic, and Black people datasets from the tarlab stimulus repository,⁶ used in [41]. The datasets are made of facial expression images. We have converted these images to vectors of the same size coded on the gray scale, and we had retrieved the emotion linked with each image. A python algorithm retrieves the video from a webcam, applies a face

⁵The Japanese Female Facial Expression (JAFPE) Database. It contains 213 images of 7 facial expressions (6 basic facial expressions + 1 neutral) posed by 10 Japanese female models. Each image has been rated on 6 emotion adjectives by 60 Japanese subjects. <http://www.kasrl.org/jaffe.html>

⁶Stimulus images courtesy of Michael J. Tarr, Center for the Neural Basis of Cognition and Department of Psychology, Carnegie Mellon University, <http://www.tarlab.org/>. Funding provided by NSF award 0339122. http://wiki.cnb.cmu.edu/Face_Place

detection with the Haarcascades code, and predicts the emotion of the detected face. Preliminary results on a reduced number of participants show that the emotion recognition algorithm is able to discriminate pain from no pain with 100% accuracy [42].

Figure 18 shows a representation of an individual performing an exercise of Hip abduction. In panel A we observe the position of the individual. Likewise, in panels B and C we observe the corresponding avatar from several viewing perspectives. The blue lines of panels B and C represent the reference angle (or threshold) that the individual should maintain or reach when performing a Hip Abduction movement. In the case of panel B we observed that the individual has not reached the objective in relation to the angle of movement of his left leg. Also, in panel B we observed that the individual has made a sagittal torso compensation. Panel C of also shows the compensations that the individual has made in relation to the Frontal torso and Leg.

The differences between the thresholds and a movement made by the individual, for each angle of interest, are represented by Figure 19. The lines in blue represents the threshold defined by the physiotherapist, while the bar values represent the result of the movement for each angle of interest in a performed exercise. Each bar color follows a linear variation from green to red depending on its value. To be intuitive, it becomes greener when the range of movement reached by the Sagittal, Frontal and leg compensation are inferior or equal of the thresholds indicated by the physiotherapist for each movement. Bars color becomes redder when the differences increase the thresholds. Another criterion is applied to the working angle because the physiotherapist establishes a range for movement (ROM). Thus, two blue lines should be represented for the working angle (ROM), one for the low threshold and another for the high threshold. In this case, the bar color becomes greener when the working angle is in a high and a low threshold. This means that the individual that perform a Hip Abduction movement has reached his goal. Otherwise, bar color becomes redder when the movement amplitude is outside the limits. This indicates that the individual objectives do not respect the limits established by the physiotherapist.

Figure 20 depicts the amplitude of the movement of the working angle performed by an individual in a session of eight series with eight repetitions by series for a Hip Abduction exercise (see the all working angle results in Appendix 4). Note that according to the Working Angle represented by the Figure 19 the first repetition for the 'Trial nro. 1' is 31.6 degrees. Blue lines represent the thresholds. Green bar for represents that the individual has reached the goal because the amplitude of the movement is within the range of movement (ROM). The red bars indicate that the amplitude of the movement is outside limits indicated by the physiotherapist. Finally, graphs with similar conditions are produced for the web-based platform for each other interest angles of the angles of example depicted in Figure 18.

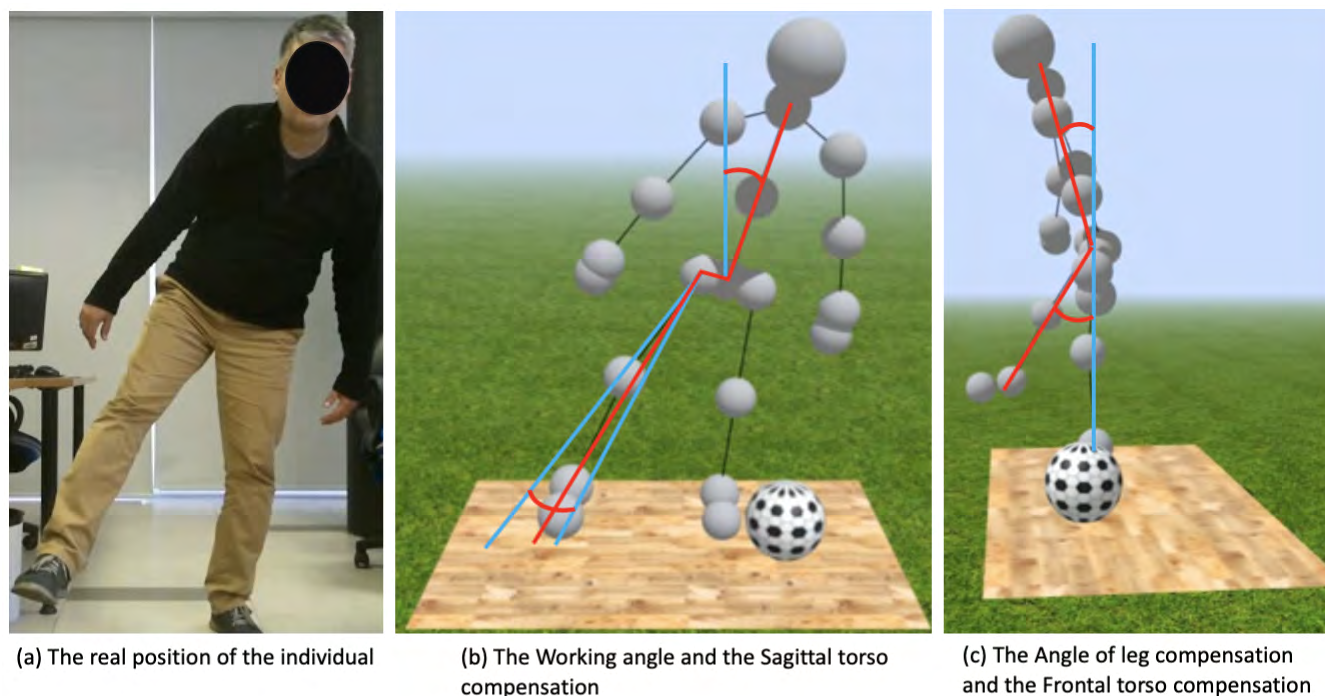


FIGURE 18. An example of a Hip abduction exercise. Blue lines depicts the thresholds. (a) Screen shot with the real position of the individual. (b) The avatar showing the Sagittal torso compensation and the working angle of the individual. (c) Other angle of the avatar showing the Frontal compensation and the angle of the Leg compensation.

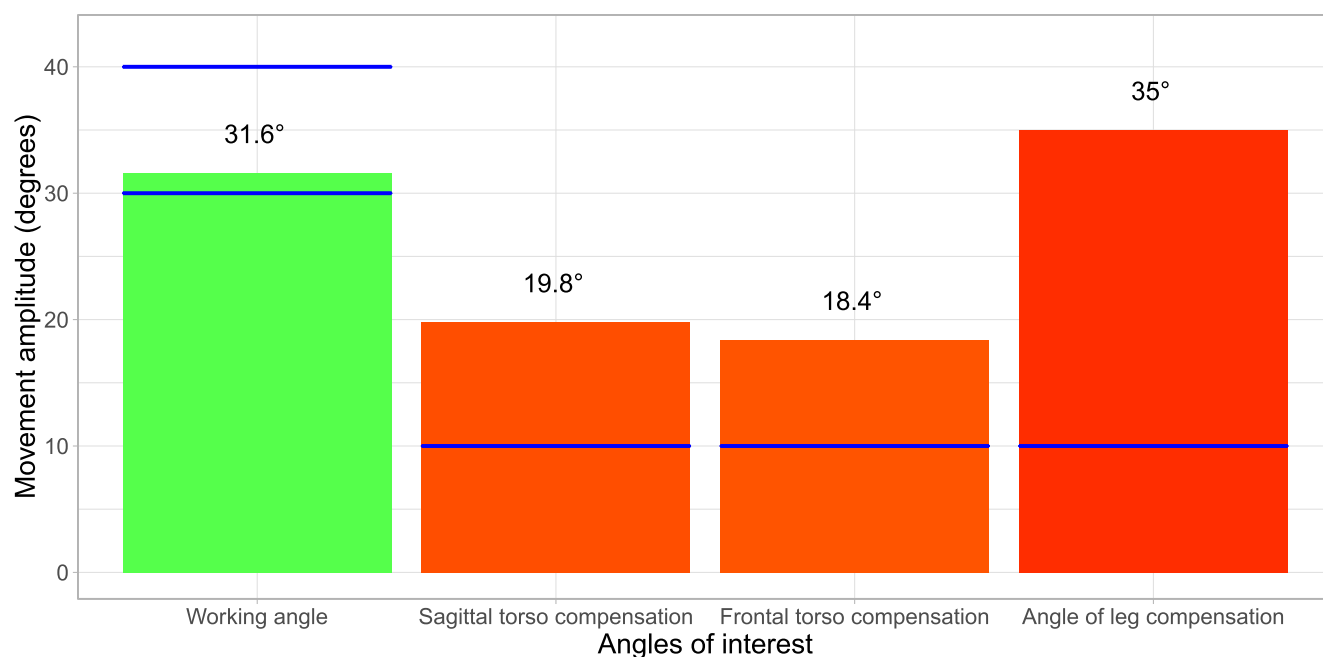


FIGURE 19. Angles of interest in a Hip abduction exercise. Blue lines represent the thresholds. Green bar for the working angle represents that the individual has reached the goal because the amplitude of the movement is within the range of movement (ROM). The red bars indicate that the individual has exceeded the limit indicated by the respective threshold.

Figure 21 shows how the pain criteria obtained by the execution of the facial recognition algorithm when a patient had a rehabilitation session. The first trial is aligned to the

pain feature in Listing 2 considering a range of 0 to 3 as it was described previously. The levels 0 and 1 refers to the pain levels while the levels 2 and 3 are not higher levels of

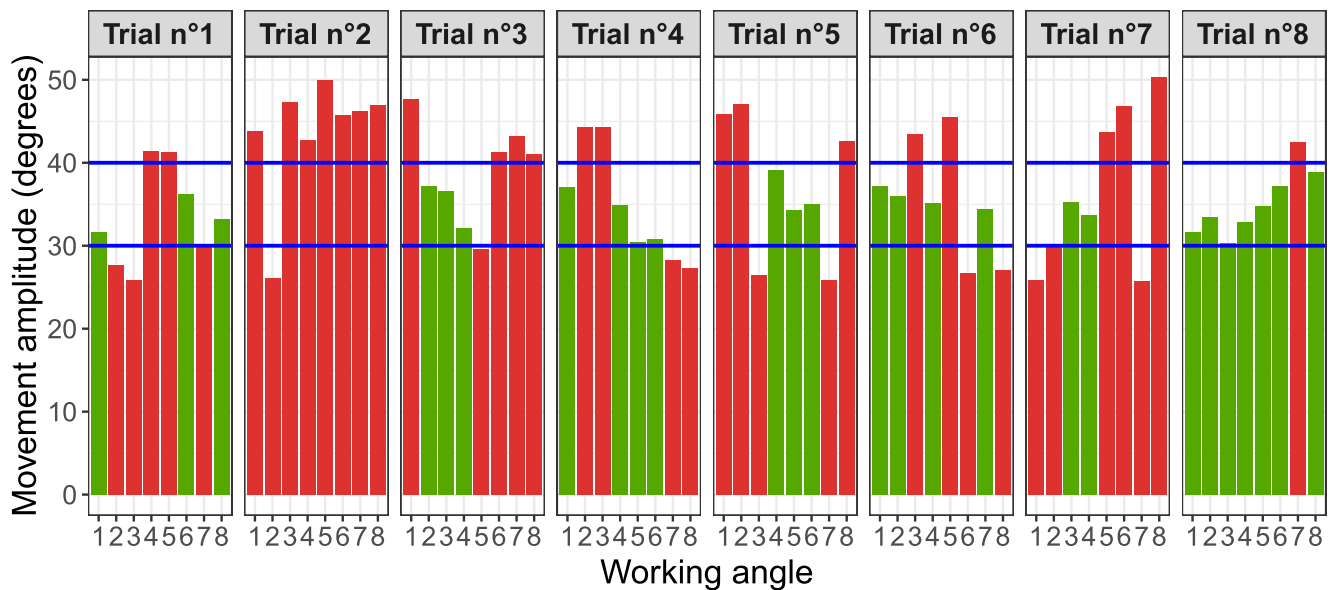


FIGURE 20. Movement amplitude of the Working angle performed for a session of eight series of Hip abduction exercise. Blue lines represent the thresholds. Green bar for represents that the individual has reached the goal because the amplitude of the movement is within the range of movement (ROM). The red bars indicate that the amplitude of the movement is outside limits indicated by the physiotherapist.

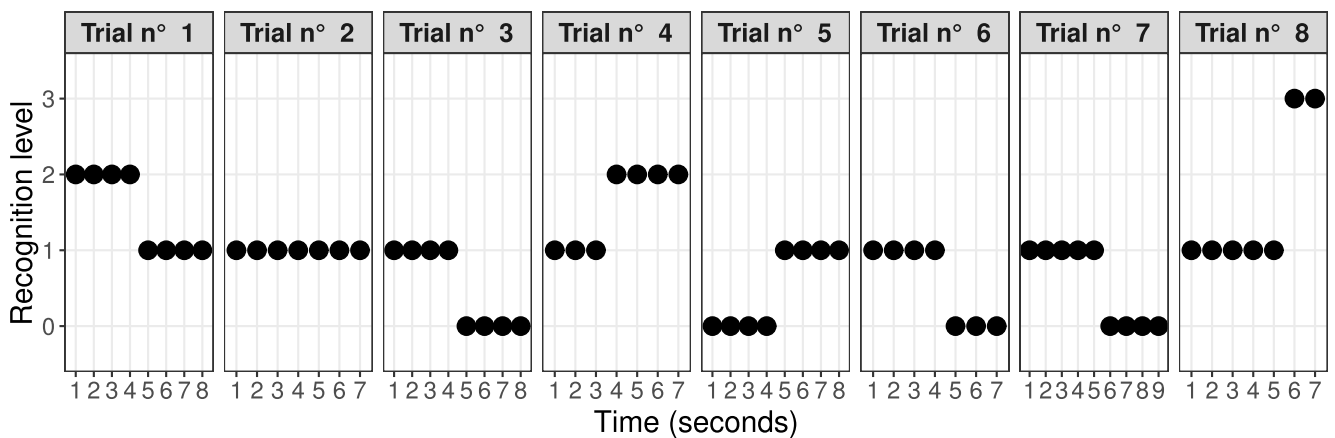


FIGURE 21. Pain recognition during the performance of movements (to note both intra and inter-trial alterations of pain level). (0) No pain. (1) Pain. (2) No face detected. (3) System error. The levels 2 and 3 are erroneous identifications of emotions and are not higher levels of pain.

pain because they are erroneous identifications of emotions. In this example, it is possible to observe several criteria:

- 1) Sometimes the pain is not detected. For example, for movements from 1 to 4 of trial 1 the system does not detect the patient's face. This situation can be by a poor position of the patient
- 2) The system can detect an emotion other than pain. In this case it is simply interpreted as "no pain" (See the results of the movements from 5 to 8 in trial 3)
- 3) The system can return errors when transmission failures occur. This is mainly due to failures in the structure of the frame (See the last two movements of trial 8)

The patient can spend more time in some trials than in others so the time between each trial can vary. The pain measurement is also useful for the physiotherapists because it gives them an idea of how much the patient suffered when performing the exercise.

As a complement, an intensity value of the pain is also obtained, see Figure 22. This value represents an estimation levels from 0 to 100% of the pain suffered by the patients when the pain is recognized by the system. For the pain recognition in the movements between 6 and 8 of the first trial of Figure 21, the intensities of the pain started at an intensity value of 45% and increased to 75%. This confirms that the pain appeared in movement 5 and its intensity increased throughout the trial. Note also that the first trial is aligned to the *pain_intensity* feature in Listing 2. The values of the intensity are over the 50% in most of the trials. However, the graph shows that the patient tends to feel less pain throughout the trials.

Additionally, an example of the evolution of the pain intensity over the therapeutic program for a determined individual is presented into the Figure 23. It is expected that the average value of pain intensity will decrease throughout the weeks.

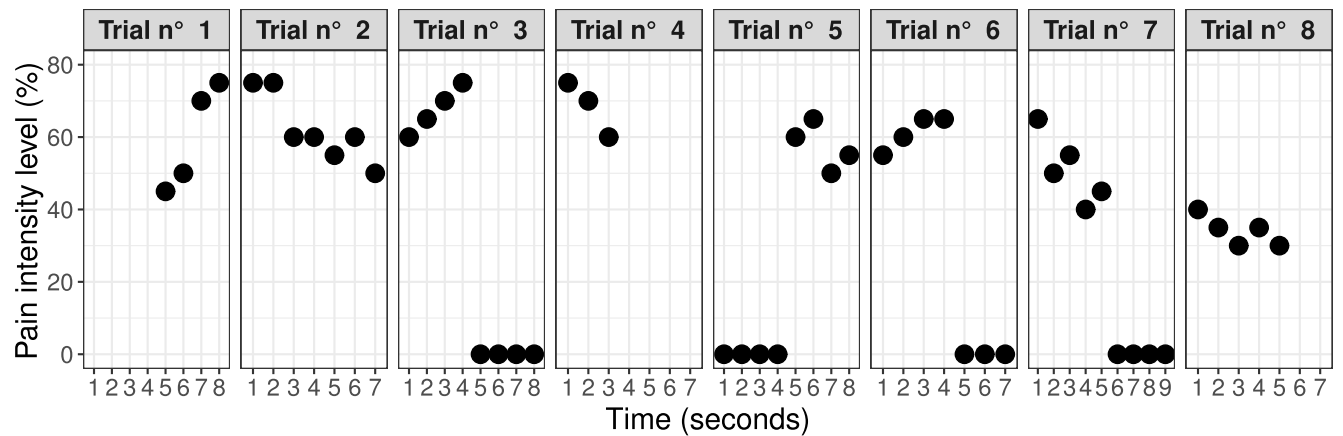


FIGURE 22. Intensity pain level during the movement.

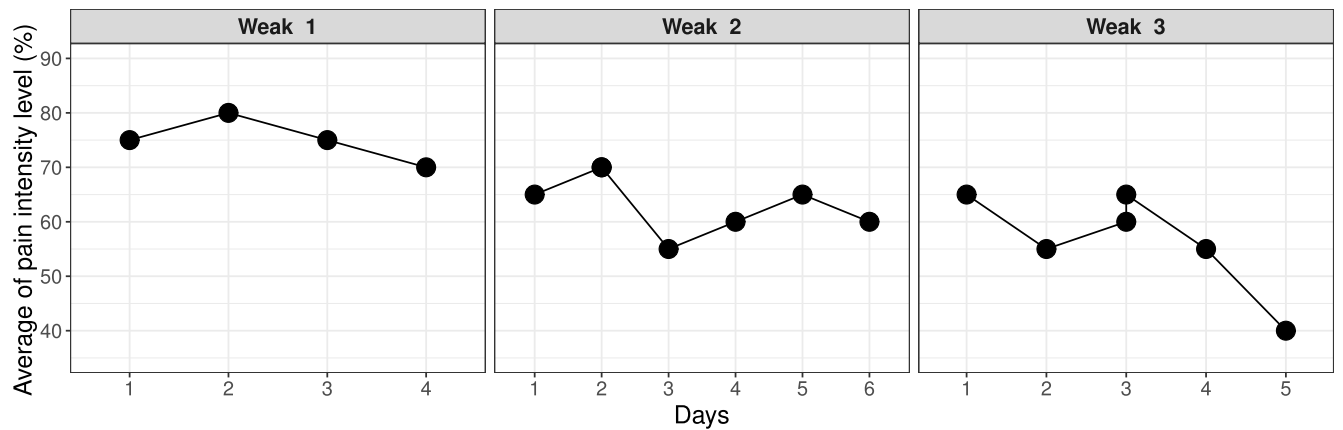


FIGURE 23. Example of the evolution of the pain intensity over the therapeutic program for a determined individual.

This information is also important for the physiotherapists as an indicator to evaluate the effectiveness of a therapeutic program.

The *Messages* package describes the components used to ensure the communication between patients and physiotherapists. Communication is key in a rehabilitation process, so it is important to provide the patients with a way to ask and consult the physiotherapists at any time. Also, it allows the physiotherapist to be aware about the patient's feedback emotional status and expected recovery celerity.

VI. INTERACTION BETWEEN THE COLLECTIONS OF OBJECTS

This section describes three UML sequence diagrams to explain how the collections of objects interact to accomplish *patient tasks* over time. These diagrams are key UML artifacts for modeling the behavior of software and are used to describe the internal system operation from the implementation point of view [34]. Figure 24 shows the sequence diagram to learn an exercise. Patients get the learning resources, then they

select a resource and the type of document. Finally, the information is given to the patients.

Figure 25 describes the sequence diagram to perform an exercise. The patients take a survey. The answers are processed to determine if the patient is authorized or not to perform the planned therapeutic session. In the case of having the authorization, the patient can start a session, the defined exercises are gotten, and an exercise is chosen to be performed. Figure 26 shows the sequence diagram to send a message. Notifications are given from the patient by the message object.

VII. DISCUSSION

This section describes some benefits and limitations of using the purposed ePHoRt architecture to support the tele-rehabilitation process. The main advantage is its modularity and flexibility, to guarantee a configurable and a scalable platform considering an architecture based in a MVC framework and a RESTful API. So, it could be extended to other types of rehabilitation exercises by increasing the

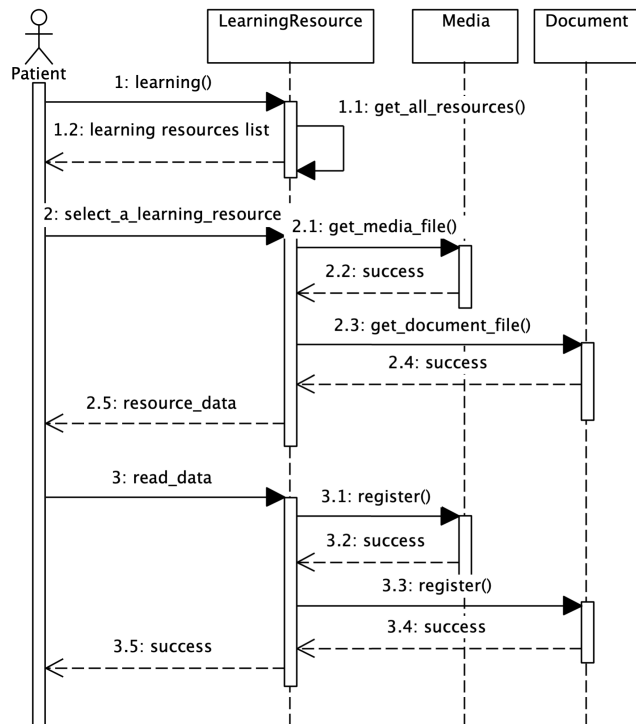


FIGURE 24. Learn an exercise sequence diagram.

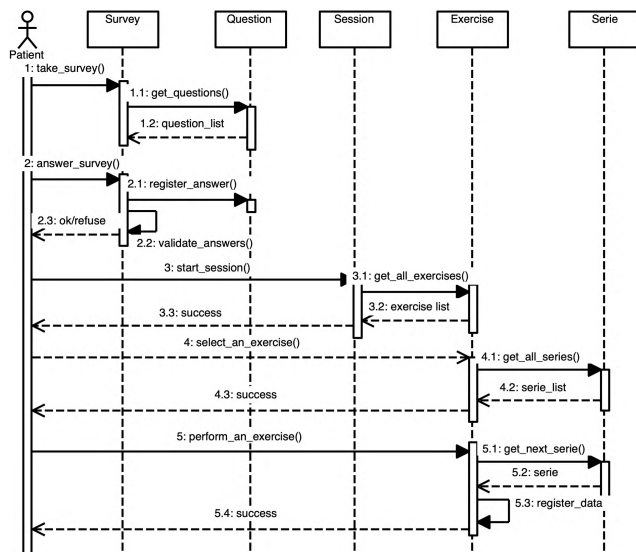


FIGURE 25. Perform an exercise sequence diagram.

resources in the CORE and by adding other AI algorithms to evaluate them with the REST API.

A meta-model is also included as a complement as the central part of the architecture. It provides the principal concepts like classes, relations and attributes. It captures how the information flows between the components and its helpful to understand how the problem was abstracted and the vocabulary terms that were considered in this domain.

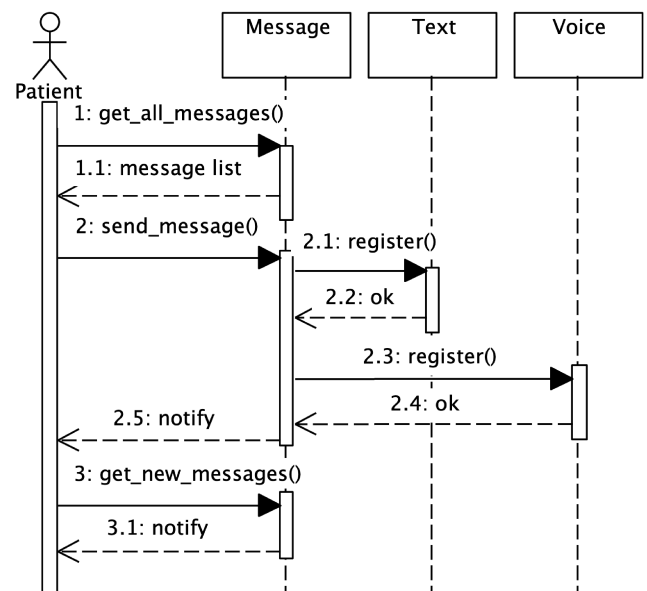


FIGURE 26. Send a message sequence diagram.

Thus, additional functionality could be re-engineered upon the proposed conceptual model.

The platform also incorporates gestural recognition to support the movements quality and the facial recognition to measure the intensity of pain that a patient suffers when performing the exercises. The intensity measure also allows the physiotherapist to evaluate the effectiveness of the therapeutic program.

The architecture considers only a Kinect device for capturing movements and emotions. It does not consider situations in which occlusions may occur due to the fact that the patient has a limited space to carry out the exercises. Likewise, the recognition of gestures is not accurate if an element that was not during the recording is entered in the image. Generally, this is the typical situation when a person who is not the one who performs the exercises appears in the Kinect' field of vision.

Actually, the real problem is the partial occlusion of certain joints during the execution of the movements; for instance, a hip occlusion by the knee during an slow exercise. Nevertheless, when the exercises or the patients require an extra element (i.e. a chair for support), it can be included as long as this element was also present during the recording phase. We would therefore recommend developing a set of guidelines covering these factors prior to wider scale uptake of Kinect technology for motion capture.

Moreover, in the same way as stated in [6], we are convinced that Kinect can perform better in an uncluttered environment which has implications for the deployment of the technology into the patient's home where space and furnishing is dictated.

The size and silhouette of people are other aspects that influence the recognition of the patient's movements. Obese

TABLE 3. List of possible instances for the preliminary survey.

No	Pain			Skin			Edema			Result
	<= 6	= 7	>= 8	Open wound	Sensitive skin	High local skin temperature	Extended on the Entire Lower Limb	Relative Reduced and Local	Very Reduced and Local	
1	True	False	False	True	False	False	True	False	False	False
2	True	False	False	True	False	False	False	True	False	False
3	True	False	False	True	False	False	False	False	True	False
4	True	False	False	False	True	False	True	False	False	False
5	True	False	False	False	True	False	False	True	False	True
6	True	False	False	False	True	False	False	False	True	True
7	True	False	False	False	False	True	True	False	False	False
8	True	False	False	False	False	True	False	True	False	True
9	True	False	False	False	False	True	False	False	True	True
10	True	False	False	True	True	False	True	False	False	False
11	True	False	False	True	True	False	False	True	False	False
12	True	False	False	True	True	False	False	False	True	False
13	True	False	False	True	False	True	True	False	False	False
14	True	False	False	True	False	True	False	True	False	False
15	True	False	False	True	False	True	False	False	True	False
16	True	False	False	False	True	True	True	False	False	False
17	True	False	False	False	True	True	False	True	False	True
18	True	False	False	False	True	True	False	False	True	True
19	True	False	False	True	True	True	True	False	False	False
20	True	False	False	True	True	True	False	True	False	False
21	True	False	False	True	True	True	False	False	True	False
22	False	True	False	True	False	False	True	False	False	False
23	False	True	False	True	False	False	False	True	False	False
24	False	True	False	True	False	False	False	False	True	False
25	False	True	False	False	True	False	True	False	False	False
26	False	True	False	False	True	False	False	True	False	True
27	False	True	False	False	True	False	False	False	True	True
28	False	True	False	False	False	True	True	False	False	False
29	False	True	False	False	False	True	False	True	False	True
30	False	True	False	False	False	True	False	False	True	True
31	False	True	False	True	True	False	True	False	False	False
32	False	True	False	True	True	False	False	True	False	False
33	False	True	False	True	True	False	False	False	True	False
34	False	True	False	True	False	True	True	False	False	False
35	False	True	False	True	False	True	False	True	False	False
36	False	True	False	True	False	True	False	False	True	False
37	False	True	False	False	True	True	True	False	False	False
38	False	True	False	False	True	True	False	True	False	True
39	False	True	False	False	True	True	False	False	True	True
40	False	True	False	True	True	True	True	False	False	False
41	False	True	False	True	True	True	False	True	False	False
42	False	True	False	True	True	True	False	False	True	False
43	False	False	True	True	False	False	True	False	False	False
44	False	False	True	True	False	False	False	True	False	False
45	False	False	True	True	False	False	False	False	True	False
46	False	False	True	False	True	False	True	False	False	False
47	False	False	True	False	True	False	False	True	False	False
48	False	False	True	False	True	False	False	False	True	False
49	False	False	True	False	False	True	True	False	False	False
50	False	False	True	False	False	True	False	True	False	False
51	False	False	True	False	False	True	False	False	True	False
52	False	False	True	True	True	False	True	False	False	False
53	False	False	True	True	True	False	False	True	False	False
54	False	False	True	True	True	False	False	False	True	False
55	False	False	True	True	False	True	True	False	False	False
56	False	False	True	True	False	True	False	True	False	False
57	False	False	True	True	False	True	False	False	True	False
58	False	False	True	False	True	True	True	False	False	False
59	False	False	True	False	True	True	False	True	False	False
60	False	False	True	False	True	True	False	False	True	False
61	False	False	True	True	True	True	True	False	False	False
62	False	False	True	True	True	True	False	True	False	False
63	False	False	True	True	True	True	False	False	True	False

or very small people can hardly be detected. We are convinced that we could avoid such limitations with other forms of low cost data acquisition. Likewise, we found an interesting challenge with the incorporation of a filtering stage after the acquisition because the low cost motion capture device will provide us with a noisy raw signal.

Other data acquisition mechanisms like red conventional cameras, Webcams or inertial sensors could provide us a maximum precision. For this reason, throughout the development of the architecture, we have incorporated the use of the Kinect in an independent module. However, other ways to solve our current issues will be replaced by another technology in a future.

The manipulation of the interface while patient perform the exercises is done by gestural commands. Hand gesture recognition is a useful solution to suspend and to operate the remote interface. However, the patient can get tired quickly by the fact of having to raise his hand for a long time and repetitively. For this reason, the incorporation of another interaction modality like vocal commands is required into the architecture when patients perform their exercises.

The execution interface of the exercises considers only an interactive scenario as support for the rehabilitation process. We evidence the need to add to the architecture a serious game package to increase the flexibility and modularity of the platform and considering more versatile scenarios. The separation of the interactive scenarios also allows us to increase the performance and the use of other technologies like Three.js or Unity 3D for the development of serious games. A user-friendly 3d environment will help the patient and to feel more comfortable while he performs the therapeutic plan. Additionally, new changes and functionalities will be possible to incorporate thanks to the proposed architecture.

VIII. CONCLUSION

This paper presents a flexible, a scalar and a modular software architecture for the ePHoRt platform. The proposed architecture considers technological and medical aspects. The main contribution of this work is to serve as a conceptual guideline for the development of tele-rehabilitation platforms. Web-based platforms for home rehabilitation are of interest not only for the patients and their families, but also for public health since the use of technology saves infrastructure costs.

Current work consists of: (1) finalizing the development of the platform from the physiotherapist's perspective. With the next development, we plan to include emerging dashboards to facilitate decision making by physiotherapists. Once the development is finished, we must (2) conduct usability tests to measure the performance (e.g. time, efficiency) of both patients and physiotherapist when using the platform, and (3) to carry out accessibility and cognitive tests (e.g. mental workload) in order to offer an inclusive tele-rehabilitation platform.

Future efforts contemplate replacing the Kinect with conventional low-cost cameras and/or inertial sensors to explore

TABLE 4. Working angle in a session of eight series of a Hip abduction exercise.

No	Trial	Repetition	Working Angle (°)	diffWA _{Th} (°)
1	1	1	31.6	8.4
2	1	2	27.67	12.33
3	1	3	25.82	14.18
4	1	4	41.36	-1.36
5	1	5	41.34	-1.34
6	1	6	36.27	3.73
7	1	7	29.97	10.03
8	1	8	33.16	6.84
9	2	1	43.82	-3.82
10	2	2	26.13	13.87
11	2	3	47.27	-7.27
12	2	4	42.73	-2.73
13	2	5	49.97	-9.97
14	2	6	45.8	-5.8
15	2	7	46.26	-6.26
16	2	8	46.91	-6.91
17	3	1	47.66	-7.66
18	3	2	37.24	2.76
19	3	3	36.59	3.41
20	3	4	32.13	7.87
21	3	5	29.56	10.44
22	3	6	41.33	-1.33
23	3	7	43.25	-3.25
24	3	8	41.08	-1.08
25	4	1	37.08	2.92
26	4	2	44.25	-4.25
27	4	3	44.36	-4.36
28	4	4	34.89	5.11
29	4	5	30.39	9.61
30	4	6	30.83	9.17
31	4	7	28.28	11.72
32	4	8	27.32	12.68
33	5	1	45.9	-5.9
34	5	2	47.11	-7.11
35	5	3	26.41	13.59
36	5	4	39.1	0.9
37	5	5	34.24	5.76
38	5	6	35	5
39	5	7	25.81	14.19
40	5	8	42.62	-2.62
41	6	1	37.22	2.78
42	6	2	35.94	4.06
43	6	3	43.41	-3.41
44	6	4	35.1	4.9
45	6	5	45.56	-5.56
46	6	6	26.72	13.28
47	6	7	34.39	5.61
48	6	8	27.04	12.96
49	7	1	25.85	14.15
50	7	2	29.82	10.18
51	7	3	35.26	4.74
52	7	4	33.69	6.31
53	7	5	43.72	-3.72
54	7	6	46.89	-6.89
55	7	7	25.78	14.22
56	7	8	50.27	-10.27
57	8	1	31.66	8.34
58	8	2	33.47	6.53
59	8	3	30.32	9.68
60	8	4	32.82	7.18
61	8	5	34.75	5.25
62	8	6	37.17	2.83
63	8	7	42.44	-2.44
64	8	8	38.88	1.12

other ways to solve our current motion capture limitations. The latter technology will solve problems such as the occlusion. Also, we thought about exploring the possibility of integrating vocal and gestural commands to the platform for a more ecological interaction. In addition, with the incorporation of other mechanisms for acquiring patients' movements, we must evaluate the performance on people with different morphologies.

APPENDIX A POSSIBLE OPTIONS FOR THE PRELIMINARY SURVEY

Table 3 presents the all possible instances to validate the preliminary questionnaire based the pain, skin and edema. All instances were manually validated by the physiotherapists considering their experience to determine if the patient can or cannot continue with the therapeutic program.

APPENDIX B SCRIPT IN LANGUAGE R TO CREATE THE SIMPLIFIED DECISION TREE

The instances presented in the previous Appendix A are used as ground truth to train a model with the *Classification and Regression Trees (CART)* and to create a simple binary decision tree. The decision tree was used later in the program to validate the patient' answers and determine if the he can or cannot continue with the therapeutic program in real time. If the output was true, the patient could continue with the therapeutic program without problems. However, If the output was false, the patient cannot continue with the therapeutic program and the physiotherapist will receive a notification so he can communicate with the patient or plan an urgent face-to-face meeting.

```
library("rpart")
library("rpart.plot")
dt<-read.csv(file="posible_instances.csv", sep=",")
head(dt)
dt1<-dt[, 1:9]

rpart.tree=rpart(dt$decision~., data=dt1, method
                 ="class", minsplit = 2,
                 cp = 0.0000001)
rpart.plot(rpart.tree)
dt$pred <- predict(rpart.tree, dt1, type = "class")
```

APPENDIX C EXAMPLE OF A WORKING ANGLE IN A HIP ABDUCTION EXERCISE

Table 4 presents a sample of instances of a session composed of eight series in Hip abduction exercise. A serie has a set of 8 trials with 8 repetitions where the working angle is record in order to determine the $diffWA_{Th}$. The $diffWA_{Th}$ is the difference between the working angle and a threshold of 40° .

REFERENCES

- [1] L. Bass, P. Clements, and R. Kazman, *Software Architecture in Practice*. Reading, MA, USA: Addison-Wesley, 2003.
- [2] S. Foote, *Learning to Program*. Reading, MA, USA: Addison-Wesley, 2014.
- [3] H. Andrew and T. David, *The Pragmatic Programmer: From Journeyman to Master*. Redwood City, CA, USA: Addison Wesley, 2000.
- [4] F. Buschmann, K. Henney, and D. C. Schmidt, *Pattern-Oriented Software Architecture: On Patterns and Pattern Languages*, vol. 5. Hoboken, NJ, USA: Wiley, 2007.
- [5] D. Antón, A. Goñi, A. Illarramendi, J. J. Torres-Unda, and J. Seco, "KiReS: A Kinect-based telerehabilitation system," in *Proc. IEEE 15th Int. Conf. e-Health Netw., Appl. Services (Healthcom)*, Oct. 2013, pp. 444–448.
- [6] D. Anton, M. Nelson, T. Russell, A. Goni, and A. Illarramendi, "Validation of a Kinect-based telerehabilitation system with total hip replacement patients," *J. Telemed. Telecare*, vol. 22, no. 3, pp. 192–197, 2016.
- [7] G. Kurillo, J. J. Han, A. Nicorici, and R. Bajcsy, "Tele-MFAsT: Kinect-based tele-medicine tool for remote motion and function assessment," in *Proc. MMVR*, 2014, pp. 215–221.
- [8] Y. Rybarczyk, J. K. Deters, A. A. Gonzalvo, M. Gonzalez, S. Villarreal, and D. Esparza, "ePHoRt project: A Web-based platform for home motor rehabilitation," in *Proc. World Conf. Inf. Syst. Technol.* Cham, Switzerland: Springer, 2017, pp. 609–618.
- [9] J. L. P. Medina and J. Vanderdonckt, "A tool for multi-surface collaborative sketching," in *Proc. 3rd Int. Workshop Interacting Multi-Device Ecologies*, 2016. [Online]. Available: <http://cross-surface.com/iss2016/>
- [10] Y. Ghanam and S. Carpendale, "A survey paper on software architecture visualization," Univ. Calgary, Calgary, AB, Canada, Tech. Rep., 2008. doi: 10.11575/PRISM/30491.
- [11] R. L. Cordero and I. R. Salavert, "Analyzing styles of the modular software architecture view," in *Proc. Eur. Conf. Softw. Archit.* Berlin, Germany: Springer, 2007, pp. 275–278.
- [12] J. J. Rivas, F. Orihuela-Espina, L. Palafox, N. Berthouze, M. del Carmen Lara, J. Hernández-Franco, and E. Sucar, "Unobtrusive inference of affective states in virtual rehabilitation from upper limb motions: A feasibility study," *IEEE Trans. Affect. Comput.*, to be published.
- [13] R. Calvo, S. D'Mello, J. Gratch, A. Kappas, J. F. Cohn, and F. De La Torre, "Automated face analysis for affective computing," Oxford Univ. Press, Oxford, U.K., Jul. 2014, pp. 131–150. [Online]. Available: <https://www.oxfordhandbooks.com/view/10.1093/oxfordhb/9780199942237.001.0001/oxfordhb-9780199942237-e-020>
- [14] A. Kleinsmith and N. Bianchi-Berthouze, "Affective body expression perception and recognition: A survey," *IEEE Trans. Affect. Comput.*, vol. 4, no. 1, pp. 15–33, Jan. 2013.
- [15] Z. Hammal and J. F. Cohn, "Towards multimodal pain assessment for research and clinical use," in *Proc. Workshop Roadmapping Future Multimodal Interact. Res. Including Bus. Opportunities Challenges (RFMIR)*, New York, NY, USA, 2014, pp. 13–17. [Online]. Available: <http://doi.acm.org/10.1145/2666253.2666257>
- [16] Z. Hammal and J. F. Cohn, "Automatic, objective, and efficient measurement of pain using automated face analysis," in *Social and Interpersonal Dynamics in Pain*. Cham, Switzerland: Springer, 2018, pp. 121–146.
- [17] T. Olugbade, N. Berthouze, N. Marquardt, and A. Williams, "Human observer and automatic assessment of movement related self-efficacy in chronic pain: From exercise to functional activity," *IEEE Trans. Affect. Comput.*, to be published.
- [18] M. P. Jensen, J. A. Turner, J. M. Romano, and L. D. Fisher, "Comparative reliability and validity of chronic pain intensity measures," *Pain*, vol. 83, no. 2, pp. 157–162, 1999.
- [19] P. Kersten, A. A. Küçükdeveci, and A. Tennant, "The use of the visual analogue scale (VAS) in rehabilitation outcomes," *J. Rehabil. Med.*, vol. 44, no. 7, p. 609, 2012.
- [20] A. M. Boonstra, H. R. S. Preuper, M. F. Reneman, J. B. Posthumus, and R. E. Stewart, "Reliability and validity of the visual analogue scale for disability in patients with chronic musculoskeletal pain," *Int. J. Rehabil. Res.*, vol. 31, no. 2, pp. 165–169, 2008.
- [21] J. E. Ware, Jr., and C. D. Sherbourne, "The MOS 36-item short-form health survey (SF-36): I. Conceptual framework and item selection," *Med. Care*, vol. 30, no. 6, pp. 473–483, 1992.
- [22] N. Bellamy, G. Wells, and J. Campbell, "Relationship between severity and clinical importance of symptoms in osteoarthritis," *Clin. Rheumatol.*, vol. 10, no. 2, pp. 138–143, Jun. 1991. doi: 10.1007/BF02207652.
- [23] V. Zampelis, E. Ornstein, H. Franzén, and I. Atroshi, "A simple visual analog scale for pain is as responsive as the WOMAC, the SF-36, and the EQ-5D in measuring outcomes of revision hip arthroplasty," *Acta Orthopaedica*, vol. 85, no. 2, pp. 128–132, 2014.

- [24] D. A. Delgado, B. S. Lambert, N. Boutris, P. C. McCulloch, A. B. Robbins, M. R. Moreno, and J. D. Harris, "Validation of digital visual analog scale pain scoring with a traditional paper-based visual analog scale in adults," *JAAOS Global Res. Rev.*, vol. 2, no. 3, p. e088, 2018.
- [25] Y. Rybarczyk, J. K. Deters, C. Cointe, and D. Esparza, "Smart Web-based platform to support physical rehabilitation," *Sensors*, vol. 18, no. 5, p. 1344, 2018.
- [26] F. Paternò, *Model-Based Design and Evaluation of Interactive Applications*, 1st ed. Berlin, Germany: Springer-Verlag, 1999.
- [27] Y. Rybarczyk and D. Vernay, "Educative therapeutic tool to promote the empowerment of disabled people," *IEEE Latin Amer. Trans.*, vol. 14, no. 7, pp. 3410–3417, Jul. 2016.
- [28] I. Khan, A. Saffari, and H. Bischof, "Tvgraz: Multi-modal learning of object categories by combining textual and visual features," in *Proc. AAPR Workshop*, 2009, pp. 213–224.
- [29] D. Riehle, "Composite design patterns," *ACM SIGPLAN Notices*, vol. 32, no. 10, pp. 218–228, 1997.
- [30] S. Jaiswal and R. Kumar, *Learning Django Web Development*. Birmingham, U.K.: Packt, 2015.
- [31] G. C. Hillar, *Django RESTful Web Services: The Easiest Way to Build Python RESTful APIs and Web Services with Django*. Birmingham, U.K.: Packt, 2018.
- [32] N. Tisch. (2017). *Kinectron—A Realtime Peer Server for Kinect 2*. [Online]. Available: <https://kinectron.github.io/docs/server.html>
- [33] B. Jakobs, *Mastering Bootstrap 4: Master the Latest Version of Bootstrap 4 to Build Highly Customized Responsive Web Apps*. Birmingham, U.K.: Packt, 2018.
- [34] O. Nikiforova, J. Sejans, and A. Cernickins, "Role of UML class diagram in object-oriented software development," *Sci. J. Riga Tech. Univ., Comput. Sci.*, vol. 44, pp. 65–74, Jan. 2011.
- [35] B. Unhelkar, *Software Engineering With UML*. Boca Raton, FL, USA: CRC Press, 2017. [Online]. Available: <https://books.google.com.ec/books?id=LQZDDwAAQBAJ>
- [36] R. A. Berk, "Classification regression trees (CART)," in *Statistical Learning from a Regression Perspective*. Cham, Switzerland: Springer, 2016, pp. 129–186. doi: [10.1007/978-3-319-44048-4_3](https://doi.org/10.1007/978-3-319-44048-4_3).
- [37] M. Bramer, *Principles Data Mining* (Undergraduate Topics in Computer Science). London, U.K.: Springer, 2016. [Online]. Available: <https://books.google.com.ec/books?id=7j94DQAAQBAJ>
- [38] S. M. Yacoub, H. H. Ammar, S. Yacoub, and H. Ammar, *Pattern-Oriented Analysis and Design: Composing Patterns to Design Software Systems*. Reading, MA, USA: Addison-Wesley, 2004.
- [39] Y. Rybarczyk, J. K. Deters, A. A. Gonzalo, D. Esparza, M. Gonzalez, S. Villarreal, and I. L. Nunes, "Recognition of physiotherapeutic exercises through DTW and low-cost vision-based motion capture," in *Proc. Int. Conf. Appl. Hum. Factors Ergonom.* Cham, Switzerland: Springer, 2017, pp. 348–360.
- [40] M. Lyons, S. Akamatsu, M. Kamachi, and J. Gyoba, "Coding facial expressions with Gabor wavelets," in *Proc. 3rd IEEE Int. Conf. Autom. Face Gesture Recognit.*, Apr. 1998, pp. 200–205.
- [41] G. Righi, J. J. Peissig, and M. J. Tarr, "Recognizing disguised faces," *Vis. Cognition*, vol. 20, no. 2, pp. 143–169, 2012.
- [42] Y. P. Rybarczyk, L. Leconte, J.-L. P. Medina, K. J. Vargas, P. Acosta-Vargas, and W. D. E. Yanez, "Design of a kinematic and emotional assessment module for the tele-rehabilitation platform," *Int. J. Model. Optim.*, vol. 9, no. 2, pp. 92–96, 2019.



JORGE-LUIS PÉREZ-MEDINA received the M.S. degree in information system and the Ph.D. degree in computer science from the Joseph Fourier University of Grenoble, France, in 2006 and 2010, respectively. He was responsible of a Human–Computer Interaction course with the Université catholique de Louvain, Belgium, from 2015 to 2016.

He is currently a Professor and a Researcher with the Intelligent and Interactive Systems Laboratory (SI2 Lab), Universidad de Las Américas, Quito, Ecuador. Throughout his career, he developed a deep knowledge of computer science and human–computer interaction. His interests in scientific research has been promoted by the desire of facilitating the work of IT managers and models' designers by helping them in choosing methods, models and modeling environments adapted to their specific needs.



KARINA BEATRIZ JIMENES-VARGAS received the master's degree in computer science. She collaborates in a tele-rehabilitation project for the Intelligent and Interactive Systems Laboratory (SI2-Lab), Universidad de Las Américas, Quito, Ecuador, as a Research Assistant. Her current research interests include software development, artificial intelligence, and data mining.



LOUIS LECONTE received the bachelor's degree in electrical engineering from ENS Paris-Saclay, in 2016, where he is currently pursuing the master's degree. From 2014 to 2016, he has done two years of Classe préparatoire in Collège Stanislas, Paris, France. He is currently doing an Internship at the CoCoLab, Université de Montréal (UdeM), Montreal, Canada. He is currently a Normalien Civil Servant of the Ecole Normale Supérieure (ENS), Paris-Saclay, France. His research interests include signal processing and artificial intelligence.



SANTIAGO VILLARREAL received the bachelor's degree in computer science and the master's degree in web intelligence from Jean Monnet University, Saint-Étienne, France, in 2007 and 2010, respectively. He is currently pursuing the Ph.D. degree with the Université Catholique de Louvain (UCL), Belgium. He has been a Professor with the Universidad de Las Américas, teaching software development, since 2013. He has also been collaborating with the Research Department with the ePHoRT project, since 2016. His current research interests include AI, multi-agent systems (M.A.S.), DISTRIBUTED Constraint Satisfaction Problem (DisCSP), movement processing, and ETC.



YVES RYBARCZYK received the Ph.D. degree in robotics from the University of Evry, France, in 2004. From 2007 to 2015, he was an Assistant Professor with the Department of Electrical Engineering, Nova University of Lisbon, Portugal. He is currently a Professor and the Head of the Intelligent and Interactive Systems Laboratory (SI2-Lab), Universidad de Las Américas, Quito, Ecuador. He has participated in several projects on the modelling and development of complex and interactive systems (eHealth and environmental sciences). He has authored more than 70 scientific papers and book chapters. His research interests include human–machine interaction and artificial intelligence.



JEAN VANDERDONCKT received the M.Sc. degree in mathematics, the M.Sc. degree in computer science, and the Ph.D. degree from the University of Namur, Belgium, in 1987, 1989, and 1997, respectively. He was a Visiting Associate Professor with Stanford University, in 2000, and is a permanent Invited Professor with the Polytechnic University of Valencia. He is currently a Full Professor with the Louvain School of Management, Université catholique de Louvain, Belgium, where he has been leading the Louvain Interaction Laboratory, since 1998. His research interests include human–computer interaction (HCI), engineering interactive computing systems (EICS), intelligent user interfaces (IUI), usability engineering, and software engineering. He is a Conference Co-Chair of the IEEE RCIS'2019. He is an Associate Editor of *ACM Transactions on Interactive Intelligent Systems* (TiiS) and *Knowledge and Information Systems* (KAIS). He is also the Co-Editor-in-Chief of the Springer Series of Human–Computer Interaction and the Springer Briefs in Human–Computer Interaction.

...