A Quality-of-Data Aware Mobile Decision Support System for Patients with Chronic Illnesses

Nekane Larburu[1(✉)], Boris van Schooten[1,2], Erez Shalom[3], Nick Fung[1], Marten van Sinderen[1], Hermie Hermens[1,2], and Val Jones[1(✉)]

[1] University of Twente, PO Box 217, Enschede, The Netherlands
{n.larbururubio, l.s.n.fung, m.j.vansinderen, v.m.jones}@utwente.nl, {b.vanschooten, h.hermens}@rrd.nl
[2] Roessingh Research and Development, Enschede, The Netherlands
[3] The Medical Informatics Research Center, Ben Gurion University of the Negev, Beer-Sheva, Israel
erezsh@bgu.ac.il

Abstract. We present a mobile decision support system (mDSS) which runs on a patient Body Area Network consisting of a smartphone and a set of biosensors. Quality-of-Data (QoD) awareness in decision making is achieved by means of a component known as the Quality-of-Data Broker, which also runs on the smartphone. The QoD-aware mDSS collaborates with a more sophisticated decision support system running on a fixed back-end server in order to provide distributed decision support. This distributed decision support system has been implemented as part of a larger system developed during the European project MobiGuide. The MobiGuide system is a guideline-based Patient Guidance System designed to assist patients in the management of chronic illnesses. The system, including the QOD-aware mDSS, has been validated by clinicians and is being evaluated in patient pilots against two clinical guidelines.

Keywords: Decision support · Computer-interpretable clinical guidelines · Knowledge representation for healthcare processes · Context-aware healthcare processes · Mobile process and task support in healthcare

1 Introduction

We present the design and implementation of a quality-aware mobile decision support system (mDSS) [1]. The mDSS forms part of a larger system developed during the IST MobiGuide project, in which a guideline-based Patient Guidance System (PGS) designed to assist patients in the management of chronic illnesses is researched, developed and evaluated. The MobiGuide PGS supports the patient and their medical team in adhering to best evidence as encapsulated in clinical practice guidelines. Moreover it supports communication between them, information sharing and shared decision making between patient and clinician. The goal is to support mobile, guideline-based monitoring and management, supporting independence whilst preserving safety.

D. Riaño et al. (Eds.): KR4HC/ProHealth 2015, LNAI 9485, pp. 126–139, 2015.
DOI: 10.1007/978-3-319-26585-8_9

The mDSS is part of a distributed Decision Support System (DSS). The knowledge-base of the distributed DSS is based on the knowledge encapsulated in Clinical Guidelines. The mobile part is implemented, along with other components, on a smartphone, as part of a Body Area Network (BAN) which performs patient monitoring by means of body-worn or mobile sensors and delivers guideline-based recommendations to patients via a smartphone interface.

We define a BAN as a body worn network of communicating devices, incorporating a processing platform (e.g. smartphone). In the case of a health BAN, the devices may include medical devices such as biosensors as well as general purpose devices (e.g. alarm buttons). BAN data such as measurements from biosensors may be processed locally on the BAN or sent to a remote system for processing, or a combination of the two. In 2001 we proposed the first application of BAN technology in healthcare [2] to support trauma care and home care. A number of health BANs for patient monitoring were prototyped and trialled during the IST Mobihealth project. In subsequent research health BAN applications were developed for a range of chronic conditions and BAN applications were augmented with context awareness [3]. Real time support for clinical guidelines was proposed in [4] and adaptive feedback, augmenting telemonitoring with teletreatment, was added [5].

In MobiGuide we extend mobile health research by distributing decision support functionality between the patient's mobile system and a fixed back-end system; a feature shared with ubiquitous healthcare systems such as [6, 7]. However, in MobiGuide we also incorporate clinical decision support based on clinical guideline knowledge and introduce quality of data awareness into the formalized clinical guidelines which form the knowledge bases of the distributed decision support systems.

Quality-of-Data (QoD) awareness is based on augmentation of clinical guidelines with quality information during knowledge engineering and by labelling data with quality labels at run time so that decision making can be informed by quality of clinical data. Technological context and the associated impact on quality of clinical data are handled by the Quality-of-Data Broker (QoD Broker), which runs on the BAN. The QoD aware mDSS can run standalone on the BAN if necessary but normally collaborates with the more advanced DSS system running on the back-end.

In MobiGuide we focus on two patient groups: patients with Atrial Fibrillation (AF) and pregnant women with Gestational Diabetes Mellitus (GDM). The knowledge bases of the AF and GDM applications are based respectively on clinical guidelines [8, 9]. The MobiGuide system is designed to be generic, hence any well formulated clinical guideline could be used as a basis for a MobiGuide application for another clinical condition, assuming the appropriate knowledge engineering effort to derive a Computer Interpretable Guideline (CIG) from the narrative guideline.

This paper describes the mobile decision support system (mDSS), how QoD awareness is achieved via the QoD Broker, and how the mDSS collaborates with the back-end decision support system (BE DSS) to provide distributed decision support. Section 2 describes the knowledge engineering phase; specifically how guideline knowledge is transformed into a knowledge base and how quality information is

introduced at this stage. Section 3 presents the model for distribution of decision support between the BE DSS and the mDSS. Section 4 describes the QoD-aware mDSS and its relation to the back-end system. Sections 5 and 6 describe respectively the QoD Broker and the mDSS. Discussion and conclusions are found in Sect. 7.

2 Formalizing Clinical Guidelines

Clinical guidelines bring together the best and latest scientifically proven knowledge about how to manage and treat a particular condition and as such represent current medical consensus. They are developed by panels of top medical experts who review evidence from clinical trials and scientific literature in order to support evidence-based care. Most guidelines are written in natural language, however, and in order to integrate a guideline into an automated DSS the narrative guideline must be formalized to produce a Computer Interpretable Guideline (CIG). In the formalization step the guideline is analysed and carefully transformed into a semantically equivalent computer interpretable version expressed in a formal language such as Asbru [10].

Based on the knowledge acquisition methodologies of [11, 12], the guideline is first adapted to local practices and the tacit knowledge elicited from the narrative text, resulting in a local narrative consensus which is then marked up with semantic labels. This labelled, semi-structured text is then converted into a semi-formal representation which, in the MobiGuide project, takes the form of "parallel workflows" representing the sequence of tasks leading to clinical recommendations [13]. These workflows are then transformed into an executable form. As part of the analysis of the guideline, the narrative guideline and parallel workflows are also converted into a process model to identify possible options for distributing the required decision support functionality across the distributed DSS [14]. In this model, guideline knowledge is represented as a network of data flow processes, each of which encapsulates a separable portion of the guideline knowledge and represents, by definition, a unit operation that can be executed in parallel with the others. In this way, the model facilitates the identification of concurrent and similar tasks for distribution and allows a detailed exploration of the different possible distribution options.

In MobiGuide formalization is followed by two other steps during knowledge engineering: customization and personalization. These steps enable integration of context information and personalization of guidelines in order to improve effectiveness / efficacy of disease management whilst preserving patient safety by adding context awareness to the guideline and adapting it to the individual patient. The customisation step extends the CIG, for all patients, with different possible contexts; the personalisation step instantiates the customised CIG for an individual patient. During customization the CIG is extended with the possible contexts that could affect patient guidance. These contexts include personal context information, such as whether the patient has support at home or how their daily routine may change for example in holiday contexts or at social events such as weddings. As part of this step technological context information, expressed in terms of quality of data (QoD), is also added to the CIG.

Our definition of 'technological context' [15], which is aligned with Dey's [16], is the technological information, often expressed in terms of Quality of Service (QoS), provided by a collection of technological resources which characterize the treatment of a patient. The performance variations of technological resources (e.g. motion artefacts, battery level or poor internet connectivity) that characterize technological context affect the QoS of the system. As a result, the quality of the output of technological resources (i.e. the quality of the clinical data) will also be affected (see Sect. 5). Therefore, we augment the CIG with technological context information expressed in terms of QoD. This augmentation of the knowledge is performed by medical practitioners in collaboration with requirements engineers. First, requirements engineers prepare for each clinical variable (in each treatment) a "QoD effect table" that contains the five QoD dimensions that we adopted for our research (see Sect. 5). Medical practitioners determine via semi structured interviews the potential impact of each QoD dimension on treatment and how the treatment should be adapted to enhance patient safety. Requirements engineers include this information in the "QoD effect table". When the medical practitioners have validated the "QoD effect table" it is merged into the treatment scenarios and represented as data flow diagrams. Accordingly, the data flow diagrams cover the impact of QoD on different treatments. In case potential inconsistencies or conflicting conditions are encountered, medical practitioners modify the diagrams. Once the medical practitioners have validated the data flow diagrams, the information is incorporated into the formalized guideline. Subsequently these treatment adaptation mechanisms are validated with a live application of the telemedicine system that runs the executable QoD-aware CIG in the DSS. The resulting "customized" CIG defines how treatment is to be adapted for all patients according to the different possible contexts. Points where individual patient preferences can be taken into account are also specified in the customization step [17].

Personalization of the CIG takes place during a patient-physician encounter when they define together the concepts, specific to this individual patient, that will induce the contexts defined in the previous customization step and specify patient preferences (such as preferred timing of measurements). The resulting augmented CIG reflects the real state of the patient and allows him/her to receive decision-support suited to the context, based on the system's knowledge base which contains recommendations approved by physicians. These patient preferences can then be taken into account during CIG execution, enabling personalized recommendations to be delivered at appropriate times.

As a result, in the knowledge engineering phase the knowledge-base of the DSS (the augmented CIG) is extended amongst others with recommendations adapted to variations in QoD. In the operational phase incoming data (e.g. patient data from sensors) is annotated with quality labels by the QoD Broker. Together these two enable the DSS to be QoD-aware, so that the safety of the patient can be enhanced even when technological disruptions occur.

At the end of this process, the resulting (augmented) CIG is a formalized, customized, personalized and QoD aware version of the guideline. The two augmented CIGs for AF and GDM in MobiGuide are documented in [17].

3 Distributed Decision Support

In order to provide decision support to the patient anytime anywhere, the MobiGuide system incorporates, amongst other components, two decision support systems: one on the patient's smartphone (the mDSS) and one on the back-end server (BE-DSS). Although the BE-DSS, based on a continuous guideline application engine [18], has more resources available than a smartphone to perform complex data processing, it is dependent on a reliable mobile communications infrastructure for receiving patient data acquired by the BAN; this may not always be available. Therefore, by distributing some functionality to the patient's smartphone, the resources of which may be too limited for some complex decision support, the MobiGuide distributed DSS supports real-time operation independent of the network environment with the mobile part providing data input, basic processing, feedback, and guidance even if the network is temporarily unavailable. Furthermore, delegating processing to patients' mobile systems supports scalability of the service to large patient populations by processing raw bulk data locally and providing only the necessary summaries to the BE-DSS. For example, heart rate and physical activity level data are processed entirely locally on the BAN.

The mechanism for distributing knowledge and processing responsibilities between the mDSS and the BE DSS is known as Projection. In order to determine how to delegate parts of the decision support to the mDSS, several factors are considered:

- The actor of the decision (patient or physician), since it is more appropriate for the mDSS to provide decision support to patients only;
- The temporal horizon of future recommendations, whether they are alerts, for example, which require immediate patient attention and should therefore be performed by the mDSS, or longer-term decisions which are less dependent on reliable connectivity and can, as a result, be performed by the BE-DSS despite potential intermittent loss of connectivity;
- The data and knowledge resources needed for the decision compared to the resources available on the smartphone;
- The need for data stored in the PHR (Personal Health Record), which may not be accessible outside the hospital due to security and privacy considerations;
- The dependencies between different parts of the decision support, which can be identified, for example, by modelling the guideline as a network of concurrent processes (Sect. 2); and
- A consideration of where a potential personalization of the guideline should reside.

These principles need to be considered by the knowledge engineer and expert physicians during the knowledge specification phase. Once these factors are decided, the BE-DSS delegates procedural knowledge to the mDSS by creating and sending procedural directives called projections which incorporate the delegated procedural knowledge and the contextual information from the PHR (e.g. patient preferences and clinical history) that are required to interpret the raw BAN data. A procedural projection is a simplified decision procedure that can run stand-alone on the mDSS to handle decisions of part of the GL, typically for time spans ranging from days to months. It may eventually be replaced by another projection if the mDSS signals

exceptional circumstances and/or the BE-DSS decides to change the procedure. Projections contain mostly fixed schedules (such as measurement or medicine schedules), along with the conditions which could trigger schedule changes, such as QoD related conditions (see Fig. 2). The mDSS executes the schedules stand-alone until one of the change triggers occur. Then it sends a signal, a callback, to the BE-DSS which triggers the BE-DSS to change the projections if necessary. By a judicious choice of procedural knowledge to project to the mDSS, callbacks will occur at relatively low frequency, thus reducing the risks and effects associated with loss of connectivity to the back-end. While optimal functioning of the system does require the network to be available at regular intervals, the systems designed to degrade gracefully if the network is unavailable for longer periods of time.

Projections are subdivided into unit projections, which can run as parallel processes. The BE-DSS can send multiple new unit projections plus a directive to stop previously running projections in a single message. The main directives comprising a projection are detailed in Sect. 6, but typically, a projection contains a declarative part, which tags items in the mDSS database according to certain criteria, and a procedural part, which is usually a wait loop which triggers on a particular event or time. Figure 1 shows an example projection which represents a condition in the GDM guideline (two abnormal blood glucose measurements within one week) which triggers a recommendation to change the blood glucose measurement schedule. Although not explicitly shown in Fig. 1, projections may also include details of the clinical effects of quality of data. Data with insufficient quality may, for example, be tagged differently by being given a different ID and may, as a result, trigger different procedures.

```
unitProjection("20095","2 abnormal measurements in past week") {
annotateTemporal("or", [
"event.getNumber(4985)>=150",
"event.getNumber(4986)>=150",
"event.getNumber(4987)>=150",
"event.getNumber(4988)>=150"
], "abnormal_BG", "date");
while (true) {
waitTemporalQuery("count >= 2", "abnormal_BG", "8 calendardays");
callback("5111", "2 abnormal values in BG were found in your measurements in
the past week, system is calculating another schedule for you"); } }
```

Fig. 1. Example projection. The annotateTemporal statement defines the condition under which a record or set of records is annotated with a particular tag. In this case, it tags a set of events as abnormal_BG if one or more blood glucose (BG) measurements over 150 occur in one calendar day. The number 4985-4988 represent BG measurements at particular times of the day with quality higher than “very low”. The wait loop at the bottom waits for at least two abnormal_BGs to occur within 8 calendar days, then sends a callback.

4 The Quality-of-Data Aware mDSS

The focus of the paper is on the mobile part of the MobiGuide system, specifically the mDSS and its interaction with QoD Broker. This section focuses on the mDSS and the influence of QoD on the decisions output by the mDSS.

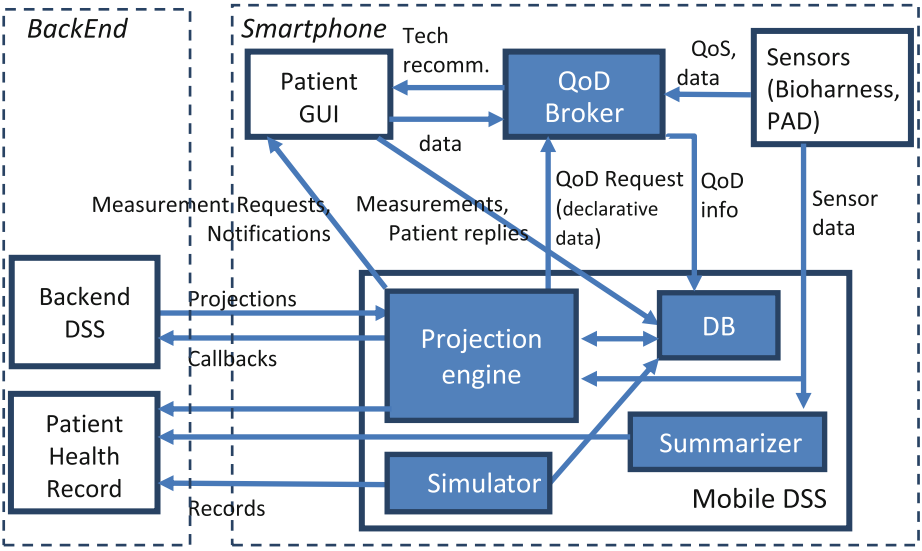

Fig. 2. Simplified architecture of the mobile QoD aware mDSS

Figure 2 shows the components comprising the mDSS and their main interaction with other components on the smartphone and the back-end. Communication proceeds through a piece of middleware called the BAN service, which handles authentication, network communication, and sensor handling. Furthermore, in case of temporary absence of connectivity, the BAN service can temporarily store all messages that are to be exchanged between the back-end and the smartphone in a queue. The mDSS components are:

- The projection engine, a scripting engine which takes care of starting, interpreting, and stopping projections. It receives projections from the BE-DSS and outputs callbacks for the BE-DSS, measurement requests to Patient GUI, and records for storage in the databases.
- The local database which stores records relevant for the mDSS functioning emanating from the projection engine, GUI or QoD Broker.
- The simulator, which enables a sequence of events to be simulated system-wide. It emits events which are then stored in the databases, namely the local database as well as the personal health record, and reacted to by the mDSS and BE-DSS.

- The summarizer, which produces summaries of AF monitoring sessions which are stored in the personal health record. Clinical abstractions include average heart rate and AF episode information.

5 The Quality-of-Data Broker

The Quality-of-Data (QoD) Broker is the component responsible for translating technological context information into QoD. Our design approach, aligned with that of Berti et al. [19], began with determination of the QoD dimensions relevant to the clinical application. The second step was to incorporate algorithms for quantifying QoD and finally we determined together with clinical specialists which is the relative QoD, since QoD requirements may diverge depending on the context (e.g. the specifics of treatment and patient condition).

QoD plays a major role in healthcare [20, 21]. The dimensions used to quantify QoD, sometimes termed QoD attributes, are highly diverse. They do not refer only to 'accuracy' or 'correctness' [22–25], but also to other quality attributes relevant for the user, aligned with 'fitness for use' theory [26, 27]. Based on literature and requirements of the clinical application, we express QoD according to five dimensions [15]:

- Accuracy: degree of correctness at which the attentive phenomenon is represented by the data. For example, if the heart rate (HR) sensor is not properly placed and the data is noisy, the accuracy of the monitored HR will be 'poor'.
- Dependability: degree of certainty that data is available (or complete), and can be used for meaningful decisions regardless of speed or accuracy. An example of 'poor' dependability is when it is not possible to measure HR due to the sensor unavailability due to lack of battery or when data connectivity is poor and data cannot be transmitted to the point of decision.
- Timeliness: time interval to transport data from source to destination. For example, HR data may contain a 'significant' delay for making a treatment decision on time due to data processing or transmission delay. This may lead to 'poor' timeliness and increase treatment risk if the patient needs to be notified immediately.
- Cost: amount of money required to obtain data for the decision-making process. Cost is a quality dimension that is addressed in very few QoD literature studies [28, 29], but is an important QoD dimension since it may affect other QoD dimensions, such as timeliness [29]. Ballou et al. [28] studied the tradeoff between cost and other QoD dimensions and found that 'in a majority of the cases the best solution in terms of error rate is the worst in terms of cost'. Moreover, medical practitioners attest the significance of cost in telemedicine systems, since it may influence treatment guidance. For example, if the patient pays more for roaming data than with Wi-Fi, when Wi-Fi is not available extra cost is needed, leading to 'poor' cost. Besides, if the roaming option is not chosen by the patient due to the additional cost, data will not be transmitted immediately, implying additional data delay; otherwise, data can be transmitted immediately, but at higher (i.e. poorer) cost.
- Quality of Evidence: degree of conformance with guidelines, rules of certification/legislation bodies and evidence based medicine (e.g. controlled trials). This is aligned

with [24]. Hence, it indicates how reliable the source of information is. For example, 'poor' Quality of Evidence (QoEvidence) of HR data is defined when an HR sensor does not hold the CE certificate that guarantees output data quality.

Notice that, depending on the context, the impact of the QoD dimensions may vary. For example, if the patient has a fixed cost for roaming data, cost may not have much influence. However, if the patient is abroad, data roaming cost to transmit clinical data to the BE DSS may need to be considered.

In order to compute clinical data quality, using the layering technique [30], we developed a conceptual model which defines the functional and non-functional relation between technological and clinical concepts. The non-functional relation is based on the functional relation and defines relations between QoD of technological variables, such as raw data, and QoD of meaningful clinical variables relevant to the treatment, such as HR. The non-functional relation also includes computational models used to determine the impact of QoS of technological resources on QoD. As described in [15], these computational models consist of transfer functions (f_i), such as mathematical functions or graph-based mapping functions. These transfer functions are used to provide QoD based on QoS and previous QoD: $QoD_i = f_i(QoS_i, QoD_{i-1})$. For example, the input data (D_{i-1}) of a technological resource, such as a HR processor, may be an electrocardiogram (ECG) with Signal to Noise ratio (SNR) equal to 0.7 dB. The SNR is an attribute that characterizes the Accuracy of the ECG. The HR processor manufacturer may have a SNR robustness graph, which shows how robust the HR process is against noise. Based on a graph-based mapping function, we can obtain the Sensitivity (Se) and Specificity (Sp) values of the output HR. From these values and applying a simple mathematical function, the scalar value of accuracy is computed: Accuracy = Se × 0.5 + Sp × 0.5 (e.g. Accuracy = 85 %). Other examples to compute QoD dimensions are shown in [15]. These QoD dimension values need to be translated to a clinically meaningful quality grades by applying the Relative Quality of Data step.

Relative QoD is a relevant concept in QoD that emphasizes the importance of taking into account the consumer's viewpoint to judge QoD based on a "fitness for use" study [26]. In our research, we applied this concept by stratifying the scalar values of the QoD dimensions from the computational models into one of four quality grades: High, Medium, Low, Very Low [15, 30]. These grades are adapted from [24]. This stratification model is based on the medical practitioners' interpretation of the scalar QoD values [15], considering also additional technological information. For example, a scalar value of HR clinical variable with accuracy = 50 % may be due to a noisy ECG signal, where the R peaks of the ECG used to compute HR are not easily identified. Additionally, the medical practitioners considered the context of the application (e.g. outdoors physical exercise treatment) and the user condition (e.g. persistent AF patient) to determine each QoD grade. Hence, with the support of the QoD expert, the medical practitioners determined for each parameter (e.g. HR) the clinically relevant ranges of QoD dimensions (e.g. Accuracy) to be mapped onto each QoD grade in each context. For example, Accuracy = 85 % may correspond to a 'Medium' quality grade in a specific case (Table 1), while in a different context, this value may be mapped to 'Low'. In order to calculate this relative QoD information aligned to the context and user,

Table 1. Stratification model example for HR_{mon} accuracy [15]

Clinical variable HR_{mon}	
Scalar ranges	Grade value
[0 %, 69.9 %]	Very Low
[70 %, 79.9 %]	Low
[80 %, 94.9 %]	Medium
[95 %, 100 %]	High

QoD Broker needs treatment declarative data, which contains the necessary context and user information. This information is provided by the mDSS.

Table 1 illustrates an example of the stratification model for the scalar values of Accuracy QoD dimension of heart rate monitoring (HR_{mon}) clinical variable. In this example, the context is an outdoors physical exercise treatment for permanent Atrial Fibrillation (AF) patients. As shown in Table 1, scalar values ranges, in this case from 0 % to 100 %, are mapped to one of the quality grades as specified by the cardiologist.

QoD Broker acquires treatment declarative data from the mDSS and QoS information and clinical data from sensors and patient GUI to compute QoD information. Additionally, QoD Broker provides technological recommendations to the patient via the GUI (Fig. 2) to improve clinical data quality, so that treatment efficacy and patient safety can be optimized. For example, QoD Broker may ask the patient to re-enter a data value when an error is detected, or it may advise the patient to charge the smartphone battery before physical exercise to pre-empt battery failure during exercise therapy. In this way the QoD Broker not only detects low quality data, but can also, in some cases, pre-empt collection of low quality data and ensure capture of higher quality data. The QoD Broker is implemented in the mobile part of the MobiGuide system to acquire QoS information from the technological resources. The mDSS processes the clinical data and its QoD. This enhances the safety of the mDSS recommendations, since its knowledge is based on the QoD-aware CIG. The QoD information is also stored in the back-end PHR, so that it can be used by the medical practitioner and by the BE-DSS to support a QoD-aware decision making process.

As discussed by Weber et al. [27], a QoD method is needed to design better health information systems. Their study focuses on Data Quality by contract (DQbC), which applies pre-conditions (data input constraints) and post-conditions (assurances of the output data), and compares the data with other data sources to quantify the quality. Our approach focuses on the QoD for an autonomous mobile patient guidance system. However the DQbC design theory and method is applicable in our model once the data is stored in the PHR.

6 The Mobile Decision Support System

The mDSS component is an Android service which communicates with other components by subscribing and publishing to the appropriate channels provided by the BAN service middleware. The mDSS functions as a sort of communication hub within

the mobile device, so a substantial part of the mDSS consists of message handling and passthrough mechanisms, making it a suitable host for the data simulator (see below).

The most interesting part of the mDSS is the projection engine. The projections were developed in several steps. First, semi-formal projections were specified according to the overall BE-DSS CIG specification. These were then developed into fully executable specifications using a projection language that was designed to be simple, yet flexible, powerful and generic. The projection language is based on JavaScript for execution using the Rhino scripting engine, which was chosen for its technical suitability: it runs on Android and enables full processing state save/restore by means of its built-in Continuation mechanism.

High-level functions were developed for enabling concise specification of guidelines. These functions principally operate on the local database, whose entries consist of time stamped events with one or more values and annotations attached to each. The most important functions are the following:

- Annotations. An annotation statement specifies a condition under which an event should be annotated with a particular annotation, or a set of events according to particular conditions within the set.
- Temporal queries. This involves specifying a calculation over time, such as a sum of values or count of events occurring within a specified time window. The projection can be made to wait for (trigger on) a temporal query. To ensure that the system does not re-trigger on the same condition again, the events that led up to the trigger are tagged so that they are no longer considered for the next temporal query. Additionally, a refractory period can be defined that specifies how long the trigger will remain inactive after triggering.
- Calendar queries. In some cases, the system reacts to events in the user's calendar, in particular if risky events like operations are planned in the near future.
- Periodic wait. The system can wait for a particular weekday or time to occur. A start and end date can also be specified.
- Event functions. Events from the database can be queried, manipulated, and stored.
- Message functions. Several functions exist for sending specific types of messages, such as patient notifications, measurement requests, and callbacks.

Apart from the projection engine, the mDSS also contains a summarizer component which summarises the BAN data streams according to the clinicians' requirements, thus mitigating any problem of information overload from the raw data. In the MobiGuide project, it was decided in consultation with the clinicians that summaries are needed for the AF application, specifically for the streaming heart rate and R-R interval data derived from the BioHarness™ sensor. As recommended by the AF guideline [17], patients should wear this sensor regularly during daily living, and whenever an AF symptom is felt, and for each monitoring session, the Summarizer computes the standard deviation of the R-R intervals every minute as well as the average, minimum and maximum heart rate detected during the whole session. Furthermore, the Summarizer receives data concerning episodes of irregular heart rate from the AF detector software running in the BAN and computes from it the total proportion of time in which the heart rate is irregular as well as the average, minimum and maximum heart rate of each irregular heart rate episode.

To enable rigorous testing, we designed a data simulator, which provides a generic means for testing the MobiGuide system against different scenarios and on data spanning potentially long time periods. A simulation scenario is subdivided into multiple steps which can be started from the GUI, allowing the user to interact with the system between steps.

7 Discussion and Conclusions

The components described here have been implemented as part of the MobiGuide system, which is being evaluated against the AF and GDM guidelines. Using a participatory design approach, medical domain experts validated the domain knowledge and system functionality during system design and development. Patient user as well as clinician user participation during the design trajectory was also a priority, with patient focus groups and surveys used to gain feedback from patients on the concepts, the design and the perceived value of the service. Regarding impact of Quality of Data, the medical practitioners understood the potential negative implications of degradations of technological context and determined that the inclusion of data quality awareness has the potential to improve patient safety and treatment effectiveness.

The MobiGuide system components have undergone unit and integration testing as well as a pre-pilot testing phase. The pre-pilot study was performed with volunteers in order to verify that the system functionalities run according to the medical requirements and successful results were obtained. Amongst other things, these tests confirmed the technical feasibility of providing QoD-aware guideline-based decision support to patients via a semi-autonomous system running on their smartphones. Currently, as a further step in the clinical and technical evaluation, the MobiGuide system is being piloted on patients in Spain and Italy. The GDM pilot site is Corporacio Sanitaria Parc Tauli de Sabadell near Barcelona in Spain and the pilot site for AF patients is Fondazione Salvatore Maugeri Clinica del Lavoro e della Riabitazione in Pavia, Italy.

For the University of Twente, the research conducted in MobiGuide together with our partners has extended our research into health BAN applications, amongst others, by incorporating clinical decision support based on clinical guideline knowledge into the BAN application, by distributing decision support functionality between the patient's mobile system and a fixed back-end system via a projection mechanism, and by introducing quality of data awareness to BAN applications.

The research on clinical decision support in the context of evidence-based medicine has produced new modelling approaches to be applied in the analysis of guideline knowledge and generic mechanisms (the projection model and language) for distributing clinical knowledge and decision support functionality. The QoD research demonstrates that the approach applied not only succeeds in detection of data quality problems (thus enabling pre-emption of adverse effects of poor data quality) but also enhancement of clinical data quality through identification and corrective action where certain technological resource problems are identified.

Acknowledgments. The MobiGuide project (http://www.mobiguide-project.eu/) has received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no. 287811.

References

1. Shalom, E., Shahar, Y., Goldstein, A., Ariel, A., Sheinberger, M., Fung, N., Jones, V., Van chooten, B.: Implementation of a distributed guideline-based decision support model in MobiGuide framework. In: Riaño, D., Lenz, R., Miksch, S., Peleg, M., Reichert, M., ten Teije, A. (eds.) KR4HC/ProHealth 2015. LNCS (LNAI), vol. 9485, pp. 111–125. Springer, Heidelberg (2015)
2. Jones, V.M., Bults, R.G.A., Konstantas, D.M., Vierhout, P.A.M.: Healthcare PANs: Personal Area Networks for trauma care and home care. In: Fourth International Symposium on Wireless Personal Multimedia Communications (WPMC) (2001)
3. Jones, V.M., Huis in't Veld, R., Tonis, T., Bults, R.B., van Beijnum, B., Widya, I., Vollenbroek-Hutten, M., Hermens, H.: Biosignal and Context Monitoring: Distributed Multimedia Applications of Body Area Networks in Healthcare. In: Proceedings of the 2008 IEEE 10th International Workshop on Multimedia Signal Processing, (MMSP 2008), pp. 820–825 (2008)
4. Hermens, H., Jones, V.: Extending remote patient monitoring with mobile real time clinical decision support. In: Veltink, P., Eberle, W. (eds.) 4th Annual Symposium of the IEEE-EMBS Benelux. University of Twente, The Netherlands (2009)
5. Jones, V., Hermens, H., et al.: Predicting feedback compliance in a teletreatment application. In: 2010 3rd International Symposium on Applied Sciences in Biomedical and Communication Technologies (ISABEL), pp. 1–5. IEEE (2010)
6. Touati, F., Tabish, R.: U-healthcare system: state-of-the-art review and challenges. J. Med. Syst. **37**, 9949 (2013)
7. Hommersom, A., Lucas, P., Velikova, M., Dal, G., Bastos, J., Rodriguez, J., Germs, M., Schwietert, H.: MoSHCA - my mobile and smart health care assistant. In: 2013 IEEE 15th International Conference on e-Health Networking, Applications Services (Healthcom), pp. 188–192 (2013)
8. Circulation, Journal of the American Heart Association. Guidelines for the Management of Patients With Atrial Fibrillation: A Report of the American College of Cardiology Foundation/American Heart Association Task Force on Practice Guidelines 123, e269–e367 (2011)
9. Rigla, M., Tirado, R., Caixàs, A., Pons, B., Costa, J.: Gestational Diabetes Guideline CSPT. MobiGuide Project Deliverable (FP7-287811) Version 1.0, 12/02/2013
10. Shahar, Y., Miksch, S., Johnson, P.: The Asgaard project: A task-specific framework for the application and critiquing of time-oriented clinical guidelines. Artif. Intell. Med. **14**(1), 29–51 (1998)
11. Shahar, Y., Young, O., Shalom, E., Galperin, M., Mayaffit, A., Moskovitch, R., Hessing, A.: A framework for a distributed, hybrid, multiple-ontology clinical-guideline library, and automated guideline-support tools. J. Biomed. Inform. **37**, 325–344 (2004)
12. Shalom, E., Shahar, Y., Taieb-Maimon, M., Bar, G., Yarkoni, A., Young, O., Martins, S.B., Vaszar, L., Goldstein, M.K., Liel, Y., Leibowitz, A., Marom, T., Lunenfeld, E.: A quantitative assessment of a methodology for collaborative specification and evaluation of clinical guidelines. J. Biomed. Inform. **41**, 889–903 (2008)

13. Sacchi, L., Fux, A., Napolitano, C., Panzarasa, S., Peleg, M., Quaglini, S., Shalom, E., Soffer, P., Tormene, P.: Patient-tailored workflow patterns from clinical practice guidelines recommendations. In: MEDINFO 2013, pp. 392–396 (2013)
14. Fung, N.L.S., Widya, I., Broens, T., Larburu, N., Bults, R., Shalom, E., Jones, V., Hermens, H.: Application of a conceptual framework for the modelling and execution of clinical guidelines as networks of concurrent processes. Procedia Comput. Sci. **35**, 671–680 (2014)
15. Larburu, N., Bults, R.G., Widya, I., Hermens, H.J.: Quality of data computational models and telemedicine treatment effects. In: 2014 IEEE 16th International Conference on e-Health Networking, Applications and Services (Healthcom), pp. 364–369 (2014)
16. Dey, A.: Understanding and using context. Pers. Ubiquit. Comput. **5**(1), 4–7 (2001)
17. Goldstein, A., Quaglini, S., Sacchi, L., Panzarasa, S., Napolitano, C., Larburu, N., Bults, R., Rigla, M., García-Sáez, G.: MobiGuide Deliverable D4.1: CIGs KB (2014)
18. Shalom, E., Shahar, Y., Parmet, Y., Lunenfeld, E.: A multiple-scenario assessment of the effect of a continuous-care, guideline-based decision support system on clinicians compliance to clinical guidelines. Int. J. Med. Inform. **84**(4), 248–262 (2015)
19. Berti, L.: Quality and recommendation of multi-source data for assisting technological intelligence applications. In: Bench-Capon, T.J.M., Soda, G., Tjoa, A.M. (eds.) DEXA 1999. LNCS, vol. 1677, pp. 282–291. Springer, Heidelberg (1999)
20. Berner, E.S.: Clinical decision support systems: state of the art. AHRQ Publication (09-0069), 4–26 (2009)
21. McCormack, J.L., Ash, J.S.: Clinician perspectives on the quality of patient data used for clinical decision support: a qualitative study. In: AMIA Annual Symposium Proceedings, vol. 2012, p. 1302. American Medical Informatics Association
22. Chengalur-Smith, I.N., Ballou, D.P., Pazer, H.L.: The impact of data quality information on decision making: an exploratory analy-sis. IEEE Trans. Knowl. Data Eng. **11**(6), 853–864 (1999)
23. Wand, Y., Wang, R.Y.: Anchoring data quality dimensions in ontological foundations. Commun. ACM **39**(11), 86–95 (1996)
24. Guyatt, G.H., Oxman, A.D., Kunz, R., Falck-Ytter, Y., Vist, G.E., Liberati, A., Schunemann, H.J., GRADE Working Group: Going from evidence to recommendations. BMJ **336**(7652), 1049–1051 (2008)
25. Widya, I., Stap, R.E., Teunissen, L.J., Hopman, B.F.: On the end-user QoS-awareness of a distributed service environment. In: van Sinderen, M., Nieuwenhuis, L.J.M. (eds.) PROMS 2001. LNCS, vol. 2213, pp. 222–237. Springer, Heidelberg (2001)
26. Wang, R.Y., Strong, D.M.: Beyond accuracy: what data quality means to data consumers. J. Manage. Inf. Syst. **12**(4), 5–33 (1996)
27. Weber, J.H., Price, M., Davies, I.: Data quality by contract – towards an architectural view for data quality in helath information systems. In: Prohealth (2015)
28. Ballou, D.P., Pazer, H.L.: Cost/quality tradeoffs for control procedures in information systems. Omega **15**, 509–521 (1987)
29. Widya, I., van Beijnum, B. J., Salden, A.: QoC-based optimization of end-to-end M-Health data delivery services. In: 14th IEEE International Workshop on Quality of Service, IWQoS 2006, pp. 252–260 (2006)
30. Larburu, N., Widya, I., Bults, R.G., Hermens, H.J.: Making medical treatments resilient to technological disruptions in telemedicine systems. In: 2014 IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI), pp. 285–288 (2014)