Service Tailoring: Towards Personalized Homecare Services

Mohammad Zarifi Eslami, Alireza Zarghami, Brahmananda Sapkota, Marten van Sinderen

Information Systems Group,
University of Twente
Enschede, the Netherlands
```
{m.zarifi, a.zarghami, b.sapkota,
m.j.vansinderen}@ewi.utwente.nl
```

Abstract. Health monitoring and healthcare provisioning for the elderly at home have received increasingly attention. Since each elderly person is unique, with a unique lifestyle, living environment and health condition, personalization is an essential feature of homecare software services. Service tailoring, which is creating a new service to meet individual requirements may be achieved in a cost-effective and time-efficient manner if new services can be configured and composed from already existing services. In this paper, we propose an effective service tailoring process and architecture to personalize homecare services according to the individual care-receiver's needs. In addition, we present a scenario to highlight the need for service tailoring and to demonstrate the feasibility of the proposed approach.

1 Introduction

As computer networks are becoming widespread, the number of distributed services available over networks grows rapidly [1]. However, these services are usually designed for a general purpose, user or situation. In reality, different people have different requirements, therefore they prefer personalized services. The requirements for personalization reflect what the user wants, needs, and likes, all of which may depend on the context at hand (for example, location, physical characteristics of the environment, available resources, people nearby). Personalization also has an evolutional aspect as requirements –demands, needs and preferences– of users change over time. Services have to operate in a constantly evolving environment of people, content, electronic devices, and legacy systems [2].

Thus, application functionality provided to users as services should (1) be aligned with the uniqueness of each user's requirements, (2) evolve with changes in these requirements, and (3) take the dynamic context of the user into account. Ideally this would call for tailor-made services; however developing such services from scratch would be economically and technically infeasible. A better approach would be to reuse existing services, configure and compose them to satisfy the unique requirements of each individual user. *Service tailoring* is a way of creating a new

service to satisfy the specific requirements of an individual user. Although service tailoring is an essential feature in any application domain, we focus on homecare application services. Tailorability has been studied extensively in the context of specific technologies and applications [3-7], but those approaches are not suitable for homecare domain as homecare services have their own specifications such as high dynamicity of user's requirements and low level of user's technical skills [8].

To support well-being, health monitoring and independent living, there is an increasing tendency to provide homecare services for the elderly, especially in developed countries [9-11]. Several technological challenges concerning homecare applications have been previously studied, but our focus is to address the uniqueness of each elderly person by applying the service tailoring approach. Current homecare systems are generally stand-alone for treating specific diseases, assuming a 'standard' patient and in a static situation [8]. However, in reality each user is unique in the way he or she experiences or is affected by a disease or disability. This is not only because of each individual's mental and physical condition, but also because of the social and physical environment. Current automated homecare systems are often technology-driven. They can be difficult to use by non-technical users and difficult to change or adapt when new requirements arise. The key question therefore is: how can services be tailored to the requirements of the users in this domain, namely, care-receivers and caregivers?

In this paper, we propose a service tailoring approach for the homecare domain. The proposed service tailoring approach allows creating a new homecare service through composing and configuring existing services, based on the user's specific requirements. Also, earlier tailoring results can be incrementally changed through subsequent applications of tailoring to match evolving user's requirements.

In Section 2, we define a scenario to motivate service tailoring for homecare. We then define a generic service tailoring process in Section 3. This process is 'generic' in the sense that it is independent of the knowledge and skills of the person who does the tailoring. A tailoring architecture for implementing the tailoring process is proposed in Section 4. This architecture identifies the components of a possible service tailoring platform. In Section 5, by using a tailoring example applicable to our scenario, we show how the proposed service tailoring process and architecture aid in creating a personalized homecare service. Finally, we conclude our paper and discuss possible future work in Section 6.

2   Example scenario

To highlight the need for tailored homecare services, we use the following scenario:
"Jan and Linda are 74 and 67 years old respectively. Despite their age and having various medical conditions, they prefer to remain living in their own home. They need to take certain medicines at certain times. However, they suffer from Alzheimer's disease and may not remember when to take which medicines and how much. In this situation, a reminder service may help them remember the prescribed time and a

dispenser service may help to take the correct dosage of the right medicines. There may be situations in which the reminder and dispenser services may not be sufficient to ensure that Jan and Linda actually take their medicines. Jan, for example, sometimes ignores the reminder and does not take his medicine because he is too disoriented. In such situations, assistance or help from other (voluntary or professional) people is necessary. An alarm service may help detect such situations and trigger a request for external help. Moreover, Jan has a hearing problem and uses a hearing aid, and Linda cannot see well and so she must use glasses.

During the morning, they have their breakfast and Jan reads the newspaper on the screen while Linda listens to the radio through the multimedia system. (There are two multimedia interactive systems, one in the kitchen and one in the living room. These systems include TV, radio, reminder texts and news, which users select via a touchscreen). Then, they visit a park close to their home and meet their friends. They usually have their lunch in a nearby restaurant. They spend most of their evenings at home; Jan watching TV and Linda reading books. During the weekend, their children usually come to visit them and Jan and Linda have other things to do and so they have a different schedule than weekdays one. "

As this example scenario shows, Jan and Linda have individual requirements and preferences, and even the same person has different requirements and preferences depending on the current situation and time. For example, for the medicine reminder service, during the morning they prefer to have the reminder on the screen in the kitchen or from the radio. After breakfast, if they go out as usual, they prefer to receive the reminder on their mobile phone, and in the evening, Jan prefers to receive his reminder on TV while Linda wants to receive it through her wheelchair vibrator. Therefore, the desired services have to deliver their functionalities in various ways in response to changing requirements, preferences and circumstances.

3   Tailoring Process

For doing tailoring, we assume three distinct types of users depending on the individual knowledge and skill sets they possess: care-receiver (elderly people), caregiver (family doctor, nurse or relative), and service developer (someone proficient with the service-tailoring facility and the underlying technologies). A care-receiver has contextual knowledge about his or her own needs and physical environment, but probably possesses little domain or technical knowledge. A caregiver has domain knowledge about healthcare practices and procedures, but probably possesses little contextual and technical knowledge. Finally, a service developer has technical knowledge about service modeling and technology, but probably possesses little contextual and domain knowledge. Therefore, although the proposed tailoring process is common among the different types of users, they may need different interfaces (for tailoring) and may have various authoritative roles (levels of tailoring). However, actors may have distinct roles when they are interacting with the system: Jan, for example, is a care-receiver, but because of his knowledge in IT may also take on the role of service developer.

The tailoring process consists of six different steps. These steps are illustrated using BPMN notation in Figure 1, with each step having a corresponding activity. We do not show a data flow in the figure because we present the process focusing on tailoring platform view, regardless of its interaction with the user.
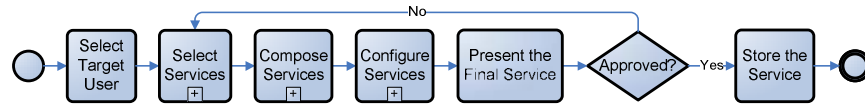
Fig. 1. Service tailoring process in BPMN notation

Of these steps, Steps 2, 3 and 4 are further refined into multiple activities. These six steps and their constituent sub-activities are explained below. To show the feasibility of the process and to make it easier to follow, each step is exemplified in Section 5.

Step 1: Selecting the Target User

Our service tailoring process starts with specifying for who the service is being created. The user data, such as age, their specific functional requirements and health status, is stored in a user profile. When a target user is selected in this step, the data stored in the corresponding user profile is used for tailoring the service in the other steps to suggest the candidate services, with proper configuration, to the user. For example, if Jan is chosen as a target user (care-receiver), the system knows that he has a hearing problem and will not offer him any services that use sound.

Step 2: Selecting the Services

Originally, the web was designed for human-human communication, but later it also become a machine understandable information space [12]. This is also true for web services. The goal of designing web services was not only to interact with humans, but also with other applications [13]. This means that data exchanged among services carry semantics. So the semantics has been used to provide machine-understandable information and enable services or frameworks to exploit machine-understandable information to facilitate interoperability. The semantics help systems to automate various aspects such as discovery, invocation and composition. Moreover, utilizing semantic similarities among components and services improves adaptability in selection. For adding semantics to our tailoring process, we exploit ontology. The ontology aids the tailoring framework to understand user requirements and thereby discover and suggest services which suit the best to these requirements.

Creating a new integrated service by a nontechnical user is a difficult task: the user needs to know which existing individual services do what. To make the job easier for the user, a goal-based method [14, 15] can be used. Goal-based methods have been used in different areas of computer science to identify stakeholder's objectives, discover requirements for software systems and guide the system's behavior. In service-oriented computing goals can be used to express users' requirements.

The second step includes five activities as shown in Figure 2:

Fig. 2. Step 2: Selection of services

In this step, first user specifies care-receiver's goals (requirements and preferences). The tailoring platform utilizes the goal ontology for homecare domain to *analyze user goals and preferences*. The aim of using a goal ontology is to minimize possible terminological heterogeneities due to autonomously created goals (user requirements) and tasks (existing services). Besides a goal ontology, a task ontology is needed to associate existing services with their support functions (tasks). Once a user asks for a service (determine his or her requirements), the tailoring platform matches the user goal (requirement) with the proper task existed in the task ontology. Then it tries to find and select the services associated with that task and suggests it to the user. Therefore, a goal based method can be used to help users in service requests. In this manner the users have a way of expressing what they want at a higher abstraction level.

As shown in Figure 3, the user goals are more objective and nonfunctional, which is easier to be understood by a non-technical user. Whereas tasks are functional but not concrete, and existing services are more concrete.
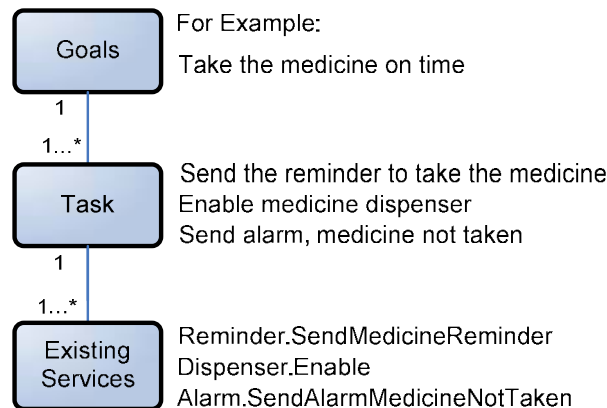

Fig. 3. How user goals wind up to the system services

After identifying the *user goals and preferences,* for *presenting the candidate services,* the service tailoring platform finds and suggests several suitable services. To do so, the service tailoring platform utilizes a repository of services and a task ontology which depicts functionalities provided by these services. Each goal in the goal ontology is associated with a set of tasks in the task ontology. Therefore, after identifying the goal, the tailoring platform finds the most suitable tasks and then suggests relevant services to the user.

The tailoring platform also uses the recommendations besides the goal based method to suggest certain services to the user. The recommendations, using other care-receivers' information to suggest certain existing services which are used by them, in a similar context [16]. As an example, assume that Ben is another elderly person who lives at his own home with similar requirements as Jan. They both use medicine dispenser service. If Ben uses an alarm service with the medicine dispenser, the tailoring platform of Ben recommends the use of alarm service to Jan's tailoring platform through the network of homecare applications. Then the alarm service will be recommended in the service selection step. However, services in homecare must be selected carefully; any recommendation must be confirmed by an authorized user as will be explained in Step 5.

Later, for *selecting services*, the caregiver selects his desired services from the set of services suggested in the previous step by the tailoring platform. In this way, the user selects explicitly desired services. However, the tailoring platform may also *select additional services* because of service dependencies. Some services may need or recommend other services' functionalities to work, as a precondition. This means that –based on user-selected services– the system adds or removes certain other relevant services. For the second part, we use the service bundling techniques [17, 18]. In a service bundle, dependencies between services are utilized to help a user to select and discover services automatically and without the user intervention.

Step 3: Composing the Services

The third step includes five activities as shown in Figure 4:

Fig. 4. Step 3: Composing of services

In this step, the tailoring platform composes the chosen services to satisfy the user requested requirements. This composition is done based on the given *user goals and stored preferences* in the user profile. Then this *composite service(s) will be presented* to the user through our tailoring platform. Predefined service compositions, stored in the Composition repository by the service developer. To suggest the proper composition, we can exploit the Event-Condition-Action (ECA) rules. An ECA rule has the general syntax on event, if condition, do actions. The event part specifies when the rule should be triggered, which, in our case, can be the user goals. The condition part specifies the conditions of this event, which, in our case, can be again the user goals or preferences (configuration of services). The action part states the actions to be performed automatically if the condition holds. In our case, this action can be the suggested composition to the user. Therefore, ECA rules can be expressed as conditions for decision making and can be used for configuration of services or composition [1, 19].

After presenting the suggested compositions, the *user should select one* of them. The user can also *edit the desired composition*, for example, changing the candidate services in the composition or their order in the suggested composition. To allow user to edit the composition, we use a mashup-like technique [20, 21]. Unlike developer-centric composition techniques, mashup provides a user-friendly graphical approach to compose services and applications.

Step 4: Configuring the Services

In general, to configure the services two different methods can be used: rule based and learning user behavior. The rule based method is necessary because it may not be possible to derive rules for new users or applications from user behavior. On the other hand, learning user behavior by learning user preferences (in different contexts) from history information is also necessary. It might be difficult to define a generic rule that is applicable to every user. Therefore, learning the user behavior can be used to update the predefined rules. We use both methods in our process.

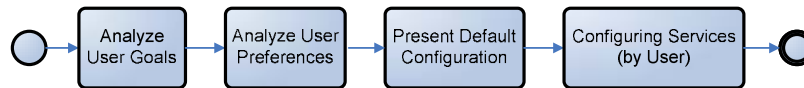The fourth step includes four activities as shown in Figure 5:

Fig. 5. Step 4: Configuring the services

The services selected in the previous step are presented to the user based on the *default configuration* derived from *user preferences* as indicated in the service profile and *user goals*. However, the user is also allowed to *(re)configure the basic services* to specify his or her current preferences and needs. To configure the services, user can use a configuration interface.

Step 5: Presenting the Final Service

In this step, the final newly created service will be presented to the user. After this step, the user confirms whether the service meets his or her requirements. If the user does not approve any of the suggested compositions, the process returns to Step 2.

Step 6: Storing the Service

Finally, once the user has approved the suggested service, the description of this newly created service will be stored for later use. Also, user preferences in the user profile may change because of configuration of the services by the user.

The sequence diagram shown in Figure 6 summarizes our discussion about the service tailoring process. It shows the interaction between the 'user' and the 'tailoring platform'. Although the 'Service repository' is a part of tailoring platform, because of its importance we presented it separately from the tailoring platform in Figure 6.
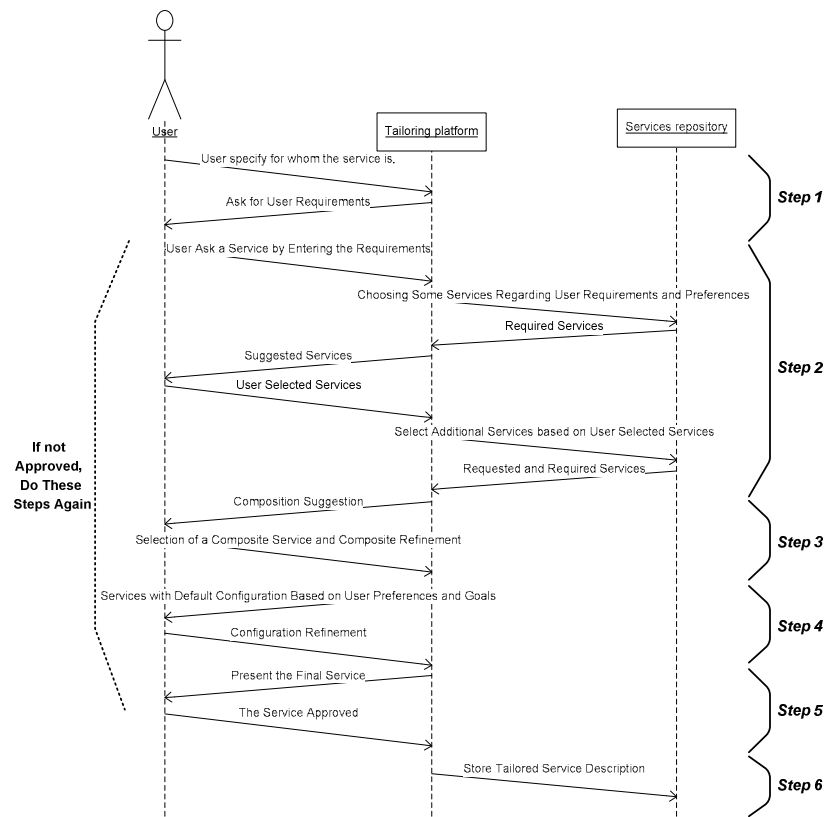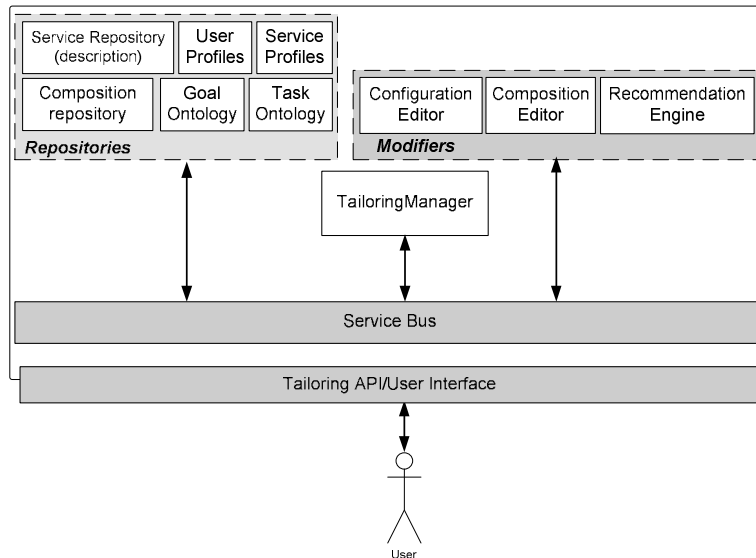
Fig. 6. Sequence diagram of the service tailoring process

4 Tailoring Architecture

To support the tailoring process described in Section 3, we identify the required components and propose a tailoring architecture for homecare services. The tailoring architecture, as presented in Figure 7, has three categories of components: *TailoringManager*, *Repositories* and *Modifiers*. These components and their relationships are described separately in the remainder of this section.

Service Repository (description) User Profiles Service Profiles

Composition repository Goal Ontology Task Ontology

Repositories

Configuration Editor Composition Editor Recommendation Engine

Modifiers

TailoringManager

Service Bus

Tailoring API/User Interface

User

Fig. 7. Service tailoring architecture

- **TailoringManager.** This component is responsible for receiving and resolving the user requests by utilizing goal and task ontology, suggesting existing services to the user by utilizing service repository, making initial configuration of services by utilizing user profile and service profiles, suggesting composition by utilizing composition repository, and storing the final services in the service repository.

- **Repositories:**
  - **Service repository.** The service repository is a repository of existing services which stores detailed information (for example, in the form of WSDL) about several patient-neutral homecare services.
  - **User profile.** This component stores user information. It has two parts: static part which keeps the static information of user such as name and gender, and dynamic part which keeps the evolving information of the user such as user specific requirements.
  - **Service profile.** This component stores the default configuration of the services.
  - **Composition repository.** This is the repository of processes which keeps different composition of services.
  - **Goal ontology.** The Goal ontology is used for helping users to specify their desired goals.
  - **Task ontology.** The tasks are supported functionalities of existing services which are defined in the Task ontology. In the Task ontology, each functionality offered by of a service is associated to one or more tasks (e.g., reminding, alarming, etc.). The service developer defines these goals and tasks ontology.

- **Modifiers:**
  - **Configuration editor.** This component supports the user (caregiver or service developer) in defining and alerting the service configurations. These configurations can be stored in the service profiles repository.
  - **Composition editor.** This component support user (caregiver or service developer) to define processes. These processes are stored in the composition repository.
  - **Recommendation engine.** This component recommends services to the users (caregiver or service developer) in service selection and composition steps. To do so, we assume all the instances of our homecare systems are registered in a central server and they can make inquiry about the services which are used by each of these instances anonymously. Assuming the central server, enables all of these recommendation engines to maintain a list of similar anonymous users and their utilized services, by exploiting the information preserved in the user and service profiles.

5 Run through Example

In this section, we show the support provided by the proposed tailoring process and architecture for meeting the requirements derived from the example scenario presented in Section 2.

"Assume that Jan is prescribed to take certain medicines at certain times. Nancy, his caregiver, wants to create a service to assist Jan in taking his medicine at right times."

We follow the service tailoring process step by step to see how Nancy creates the desired service for Jan:

Step 1: Nancy, by choosing Jan's profile, specifies through the tailoring interface that a new service to be created is for Jan. By receiving this information, the tailoring platform looks at Jan's profile and finds his specific requirements and preferences, for example, Jan has a hearing problem and will not be able to use any services or devices which utilize sound as a means to convey messages to him.

Step 2: As shown in Figure 8, Nancy sends a request to the tailoring platform for a service which can help Jan to take his medicine at right times. For this purpose, she chooses a goal among the ones listed by the system and presented in the screen. Next, she chooses sub-goals to refine or finally define her goal. For example, from the goals list, she chooses *reminder*, and then from the new sub-goals' list she chooses *reminder for medicine*. The tailoring platform, after receiving and analyzing this request and considering the fact that Jan can not use sound related services, suggests a list of services to Nancy. She chooses a *reminder service* which will send a reminder to Jan such that he will not forget the prescribed time to take a medicine. Further, she selects a *dispenser service* which will release right medicines such that Jan can take the right amount of the right medicines. After selecting the reminder and dispenser services by Nancy, the tailoring platform adds alarm service as an additional service

to the user selected services. As reminder and dispenser services are bundled with the *alarm service*, which helps to detect hazard situations (in this case ignoring the reminder and not taking the medicines) and to trigger a request for help.
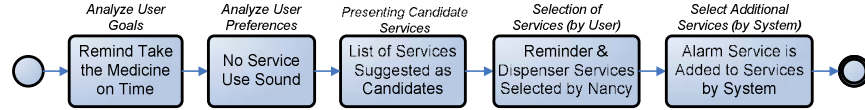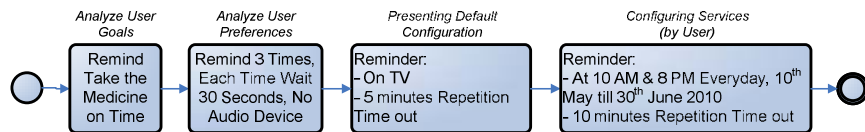

Fig. 8. Step2: Selection of services for the example

Step 3: As shown in Figure 9, the tailoring platform returns a set of possible compositions, based on user goals and preferences. For example, because Jan prefers to have reminder three times, the composition will change in such a way that the reminder service sends the reminder three times. The tailoring platform proposes different compositions such as P1 and P2, because all these compositions can satisfy user's requirements and preferences. This means that the composite service P1 first sends a reminder message to Jan and then enables the dispenser to let him take the medicine, whereas the composite service P2 first enables the dispenser and then sends the reminder message to Jan. Then, Nancy chooses P2 as a one of the possible compositions. After choosing P2, she may want to modify it, for example, because the medicine is pain killer and ignoring of taking it by Jan is not a hazard situation, so Nancy removes the alarm service from the P2.
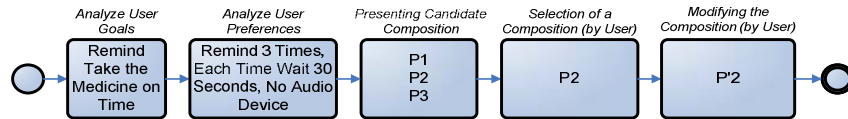
Fig. 9. Step 3: Composing the services for the example

Step 4: As shown in Figure 10, the tailoring platform returns three selected services with default configuration to Nancy. These default configurations are done based on the user goals and preferences. For example, Jan prefers a reminder to him be repeated three times, allow 5 minutes for him to react, deliver this reminder as text and deliver it on the TV. So the default configuration for the reminder will be to send the output to the URI of TV and repeat each 5 minutes if the user does not respond. Then Nancy adds other configurations, such as reminding Jan to take the medicines from 10th May to 30th June 2010, everyday at 10 AM & 8 PM. She can also edit the default configuration. She changes, for example, the repeating time of the reminder from 5 to 10 minutes.

Fig. 10. Step 4: Configuration of services for the example

Step 5 and 6: The tailoring platform presents the final tailored service to Nancy, and if she approves it the description of the service will be stored. The default

configuration of the reminder service also will be changed, from 5 to 10 minutes repetition time out. If Nancy does not approve the final service from any of suggested compositions, the process returns to the Step 2 till Nancy approves the tailored service.

6 Conclusion and outlook

Since every care-receiver is unique, personalization is one of the main concerns of homecare services. To achieve this, a service tailoring approach for homecare, with a corresponding process and supporting architecture, is proposed. Our service tailoring approach assumes the availability of basic care-related services which can be used as building blocks in the tailoring process.

The service tailoring process defines the flow of actions involving the user of the service tailoring platform to define a personalized homecare service for a care- receiver. Although we show the feasibility of the proposed service tailoring process, it is not claimed to be the only or most ideal solution for achieving service tailoring. In this paper, we describe a process and architecture for achieving personalized homecare services but further investigation is required to:

- **Developing a 'generic' service tailoring process.** The service tailoring process is 'generic' in the sense that it is independent of the knowledge and skills of the user of the service tailoring platform. We foresee that the service tailoring platform should offer different interfaces to optimally support different types of users. Moreover, the service tailoring process should be tested to identify whether the service tailoring process has the right 'genericity', and can be extended and specialized for different user types.

- **Decreasing privacy risks of recommendations.** Our service tailoring process may use recommendations from care-receivers. However, one of the main drawbacks of recommendations is privacy risks. To protect the privacy of care-receivers, distributed recommendation methods such as exploiting peer-to-peer networks would be interesting as our future work [22].

References

1.   Fujii, K. and T. Suda, *Semantics-based context-aware dynamic service composition.* ACM Trans. Autonom. Adapt. Syst., 2009. **4**(2): 1-31.

2. Di Nitto, E., et al., *A journey to highly dynamic, self-adaptive service-based applications.* Automated Software Engineering, 2008. **15**(3-4): 313-341.
3. Gehlert, A., et al., *Towards Goal-Driven Self Optimisation of Service Based Applications*, in *1st European Conf. on Towards a Service-Based Internet*. 2008, Springer-Verlag. pp. 13-24.
4. Hielscher, J., et al., *A Framework for Proactive Self-adaptation of Service-Based Applications Based on Online Testing*, in *1st European Conf. on Towards a Service-Based Internet*. 2008, Springer-Verlag. pp. 122-133.
5. Choi, O. and S. Han, *Personalization of Rule-based Web Services.* Sensors, 2008. **8**(4): 2424-2435.
6. Kumanayaka, O. and N. Ranasinghe, *Ontology based Web Service Personalization*, in *Intl. Conf. on Information and Automation, ICIA. .* 2006. pp. 69-74.
7. Kazhamiakin, R., et al., *Having Services "YourWay!": Towards User-Centric Composition of Mobile Services*, in *First Future Internet Symp., FIS*. 2009, Springer-Verlag. pp. 94-106.
8. Zarifi Eslami, M. and M. Van Sinderen. *Flexible home care automation: adapting to the personal and evolving needs and situations of the patient.* in *3rd Intl. Conf. on Pervasive Computing Technologies for Healthcare, PervasiveHealth. .* 2009. London, UK. pp. 1-2.
9. Korhonen, I., J. Parkka, and M. Van Gils, *Health monitoring in the home of the future.* Engineering in Medicine and Biology Magazine, IEEE, 2003. **22**(3): 66-73.
10. White, C.C., et al. *Improving Healthcare Quality through Distributed Diagnosis and Home Healthcare*. in *1st Transdisciplinary Conf. on Distributed Diagnosis and Home Healthcare, D2H2. .* 2006. pp. 168-172.
11. Kim, Y.-J., et al., *Hallym Jikimi 3rd system: web-based monitoring for u-health care service*, in *4th Intl. Conf. on Persuasive Technology*. 2009, ACM. pp. 1-5.
12. Berners-Lee, T., *Semantic Web Roadmap*. 1998. Available at: http://www.w3.org/DesignIssues/Semantic.html.
13. Papazoglou, M.P. and D. Georgakopoulos, *Introduction.* Communications of the ACM, Special section: Service-oriented computing 2003. **46**(10): 24-28.
14. Kangkang, Z., L. Qingzhong, and S. Qi. *A Goal-driven Approach of Service Composition for Pervasive Computing*. in *1st Intl. Symp. on Pervasive Computing and Applications*. 2006. pp. 593-598.
15. da Silva Santos, L.O.B., et al. *Towards a Goal-Based Service Framework for Dynamic Service Discovery and Composition*. in *Information Technology: Sixth Intl. Conf. on New Generations ITNG '09*. 2009. pp. 302-307.
16. Manikrao, U.S. and T.V. Prabhakar, *Dynamic Selection of Web Services with Recommendation System*, in *Intl. Conf. on Next Generation Web Services Practices*. 2005, IEEE Computer Society. 5 pp.
17. Gordijn, J., S. de Kinderen, and R. Wieringa. *Value-driven Service Matching*. in *16th IEEE Intl.Conf. on Requirements Engineering, RE '08. .* 2008. pp. 67-70.
18. Jun, H., et al., *Personalized Active Service Spaces for End-User Service Composition*, in *IEEE Intl. Conf. on Services Computing, SCC '06*. 2006. pp. 198-205.
19. Wang, F. and K.J. Turner, *Towards personalised home care systems*, in *1st intl. conf. on Pervasive Technologies Related to Assistive Environments*. 2008, ACM. pp. 1-7.
20. Xuanzhe, L., et al., *Towards Service Composition Based on Mashup*, in *IEEE Congress on Services*. 2007. pp. 332-339.
21. Nan, Z. and M.B. Rosson, *What's in a mashup? And why? Studying the perceptions of web-active end users*, in *IEEE Symp. on Visual Languages and Human-Centric Computing, VL/HCC* 2008. pp. 31-38.
22. Schafer, J.B., et al., *Collaborative filtering recommender systems*, in *The adaptive web: methods and strategies of web personalization*. 2007, Springer-Verlag. pp. 291-324.