

8-5-2011

uService – Enabling user-driven fitness services on-the-go

Alexandra Chapko

German Research Center for Artificial Intelligence, Alexandra.Chapko@dfki.de

Andreas Emrich

German Research Center for Artificial Intelligence, Andreas.Emrich@dfki.de

Marc Gräßle

German Research Center for Artificial Intelligence, Marc.Graessle@dfki.de

Tatjana Feldmann

German Research Center for Artificial Intelligence, Tatjana.Feldmann@dfki.de

Dirk Werth

German Research Center for Artificial Intelligence, dirk.werth@dfki.de

See next page for additional authors

Follow this and additional works at: http://aisel.aisnet.org/amcis2011_submissions

Recommended Citation

Chapko, Alexandra; Emrich, Andreas; Gräßle, Marc; Feldmann, Tatjana; Werth, Dirk; Tacke, Jürgen; Flake, Stephan; and Rust, Carsten, "uService – Enabling user-driven fitness services on-the-go" (2011). *AMCIS 2011 Proceedings - All Submissions*. 391.
http://aisel.aisnet.org/amcis2011_submissions/391

Authors

Alexandra Chapko, Andreas Emrich, Marc Gräßle, Tatjana Feldmann, Dirk Werth, Jürgen Tacke, Stephan Flake, and Carsten Rust

uService – Enabling user-driven fitness services on-the-go

Alexandra Chapko

German Research Center for Artificial Intelligence
Alexandra.Chapko@dfki.de

Andreas Emrich, Marc Gräble

German Research Center for Artificial Intelligence
[Andreas.Emrich][Marc.Graessle]@dfki.de

Tatjana Feldmann, Dirk Werth

German Research Center for Artificial Intelligence
[Tatjana.Feldmann][Dirk.Werth]@dfki.de

Jürgen Tacke, Stephan Flake

Orga Systems GmbH
[jtacken][sflake]@orga-systems.com

Carsten Rust

Morpho e-Documents
carsten.rust@morpho.com

ABSTRACT

User-centric developments are on the move, both in business and information technology. Especially in the mobile environment, personalization is of great importance. The concept of the mobile service super prosumer is core of the research carried out in the project uService. It aims at providing a platform for the creation, discovery and consumption of mobile services on-the-go. Tailoring fitness applications for mobile phones to the respective needs of patients and hobby runners is the main goal of uService's application scenario uRun. This use case enables hobby runners to create personalized, context-aware versions of mobile services tailored to their specific personal preferences, health conditions and training plans. Based on these features, a semantic knowledge base allows for the context-aware search and recommendations for service creation, discovery and usage. Compared to traditional consumer-centric health care systems, the depicted approach will be more personalized and context-aware and will empower physicians and personal trainers to act more timely than before.

Keywords

user-generated mobile services, health and fitness mobile application, mobile super prosumer, Web 2.0 and Web 3.0 enabled technologies

INTRODUCTION

People are increasingly looking for mobile products and services that allow them to track, compete, socially interact, and share data related to health and fitness. Furthermore, social interaction, challenge and reward systems are considered future trends for mobile health applications (Ashbrook, 2010). In the following a brief overview of mobile running applications and Web 2.0 technologies used in the domain of Health and Fitness is provided. Thereafter the recommendation mechanism developed in the project and the incorporation of semantic knowledge is described. Application of the recommendation mechanism is described by means of the uRun use case scenario. The paper ends with a conclusion and outlook section.

STATE-OF-THE-ART

Providers of mobile running applications can be differentiated according to software and hardware providers. The former develop mobile applications for mobile devices, whereas the scope of the application is limited by the technical features of the mobile device it is running on, e.g. whether a camera or GPS sensor is available. The most common providers for mobile running applications are Apple, Google and Nokia. It must be noted that the reviewed applications are not platform-independent, but usually are bound to certain mobile operating systems. Hardware providers offer mobile running applications connected to a certain device with different sensors. As a result the software application is optimally fitted to the hardware device. In general, the most common providers are manufacturers of sports goods such as Adidas, Nike or Polar. As a last category there are hybrid running applications, i.e. hardware and software providers team up to provide applications that run on a mobile device which in turn is connected to a another device of a sports good manufacturer. Apple is a software provider who offers different running applications in form of so-called apps. One of the best known applications is the free available "Smartrunner" (Smartrunner, 2010). In addition to running, "Smartrunner" also covers cycling, hiking and inline skating activities. The application provides an overview of average and maximum speed, distance travelled and calories burned. Furthermore, it enables a map view of the planned route. The travelled route can then be submitted to social networks like Facebook and Twitter, making it accessible for friends.

Other fee-based apps like “Runtastic” (Runtastic, 2010) offer the same basic functions as “Smartrunner”. Runtastic has - besides of recording time, speed, distance, altitude and the route - additional functions like a compass, real-time statistics and voice feedback. The voice feedback is used to motivate the runners to keep their target in mind. Collaboration is the main objective of the project “Jogging over distance” (Jogging over distance, 2010). Information about speed and time of the jogger is sent to a computer via a Smartphone. Since each runner is wearing a headset, runners can communicate and motivate each other, even if they are at different locations.

A product of a hardware provider is the sports mobile “athlosoft RUNNER” (Athlosoft, 2010). Using a heartbeat belt, the user can monitor the distance, speed, step frequency, step length and heart rate. The logged training data can be connected to a server using Google Earth. Therefore, the training track is transmitted to the Internet, making it visible to others in real-time. Furthermore, the route can also be transferred to Internet-enabled mobile phones, so that other users can track the path of the runners and join them if they want, even when they are already on the go.

A hybrid application is the “Nike + iPod Sport Set” (Nike + iPod, 2010). Here, a sensor in the Nike running shoe records the running data and sends it to an iPod. The user can define goals such as duration, speed and running track via the iPod. In addition, the user can select a motivational song on his iPod, which is automatically played when the speed drops below a certain level. All information can be sent to friends via Facebook and other social networks.

An activity tracking device with a rather medical focus is the BodyMedia Sensewear Armband (BodyMedia, 2011). Being worn the whole day over several weeks the armband collects all information about physical activity, sleep efficiency and, by means of an online system, food intake of the user. The aggregated data is transmitted to a physician in an evaluation-friendly format, such as Excel or .cvs. Monitoring the patient’s everyday activity, the physician is able to support the patient in organizing his/her lifestyle in a healthier way, help to set realistic goals and to achieve them and hence prevent from obesity or illness.

Collaboration, flexibility, a pre-eminence of content creation over content consumption and interactivity are considered the most prominent features of the social web (Kamel Boulos and Wheeler, 2007). In this context, Barsky et al. (Barsky and Purdon, 2006) consider amongst others wikis, blogs, and the user comment functionality as social networking enabled technologies relevant in the context of health care application. For instance, British Medical Journal Rapid Responses and a user comment functionality at Patient.co.uk offer the opportunity to record your experience as a patient and for the latter even rate the experience entries of others, thus extending the peer rating functionality with a reputation management system. In terms of wikis, wikisurgery (WikiSurgery, 2011) is an example of an social networking enabled technology which aims at collaboratively building a surgical encyclopedia for surgeons and patients. Examples of health related blogs are DrugScope DrugData Update blog (DrugScope, 2011) or the TRIP database blog (TRIP, 2011), which is a clinical search engine for clinicians. In terms of social networking sites, LibraryThing Medicine Group (LibraryThings, 2011) is an interesting example, where users with an interest in books pertaining to medicine and medical science share content and ideas.

An example of a social community offering several social web enabled technologies is the mental health social network HealthyPlace (HealthyPlace, 2011). Kamel Boulos et al. (Kamel Boulos et al., 2007) list several more examples of social networking enabled technologies in terms of health applications for instant messaging and virtual meetings or online social gaming. Eysenbach et al. (Eysenbach, Powell, Engelsakis, Rizo and Stern, 2004) conducted a review of studies on health related virtual communities and support groups and their effects of online peer to peer interactions. The results lead to the conclusion that no robust evidence exists on the effects on health and social outcomes of computer based peer to peer communities and electronic self support groups. Eysenbach et al. point out that due to the growing number of virtual communities and support groups more quantitative studies are needed in addition to qualitative studies on this topic.

An American study on social media and health conducted by the Pew Internet & American Life Project (Fox and Jones, 2009) found out that collaborative filtering mechanisms of the social web are a main feature users look for when using web 2.0 technologies, since they search for “just-in-time-someone-like-me”. On the other hand creation of health content is rather low compared to the consumption of health content. The survey furthermore found out that social networking sites are used only sparingly for health queries and updates and that there is a surge of interest in information about exercise and fitness.

USERVICE RECOMMENDATION MECHANISMS

In the following the search and recommendation mechanisms are described for the discovery of user-generated mobile services. However, the approach is transferable to any kind of object which is described with the applied description language. This results in terms of the uRun scenario in several search indices for the different objects of the scenario, i.e. services, routes, actions based on vital parameters, products from the health and fitness domain, etc. Each object will have an

ontology describing and relating relevant information about the individual object. An overall domain ontology will connect the different object ontologies and enable reasoning over the complete knowledge base of the uRun application scenario.

Query Formulation

The query mechanism is different from “classic” approaches that rely on a set of search terms. We use the concept of “query extension” (Pretschner and Gauch, 1999) to expand the query by adding user-related and contextual information. By this means, higher quality of search and recommendations results is achieved and a personalized and context-aware mechanism is provided.

Whereas search queries are actively defined by user, for recommendations, there are no explicit search terms. However, there are user’s profile information and contextual data sent by the mobile terminal. The definition of the query is sufficient for both “pull” and “push” recommendations. For “push” recommendations, a background service continually monitors the user’s current situation and provides proactive recommendations when a certain “threshold of change” is reached. For “pull” recommendations, the process is similar to a search, however, without any search terms. In this case, the results are suggested by the system based on detected user interests and preferences.

In the following, we consider a query term t_i to be the atomic part of a search query q :

$$q = (t_1, t_2, \dots, t_n)$$

Figure 1. Formula 1

Furthermore, we define a query term t_i as follows:

$$t_i = \langle c, n, v, w \rangle$$

Figure 2. Formula 2

c = category of the query term; i.e. “keyword”, “context-information”, or for service creation a “input/output block”

n = unique name of the query term within a category within a single query; (e.g. “name” for the name of a user”, or “name” as the name of the service)

v = value of the parameter (e.g. the actual name of the user, his current location, the actual search term, etc.)

w = weight of the query term within the query

The weight of the query term can be defined by the user or determined by the system. A more detailed description on how these weights are actually defined and used, is provided in the following.

As the query formulation shows, all kind of requests from mobile device can be handled by this generic query formulation. The category can be used to describe, whether we have a user-entered search term, automatically gathered context information or personal preferences of the user. When a query is formulated, weights are allocated to the different query terms. This can be done explicitly by users in an advanced search interface or implicitly by the system. Some basic considerations are listed below:

- Search terms: If search terms are defined, i.e. we have an actual search and not a proactive recommendation, they should have a rather high weight.
- User profile information: users can define weights for different items in their user profile. When executing a query, the user profile information is matched to the other query terms in order to re-rank the profile information according to their relevance.
- Context information: Context data includes complex information such as vital parameter of the user and environment variables. The very generic query formulation as shown above also allows for such scenarios.

Search index

The search index maps the generic meta data description structure to object instances. The generic meta data description structure comprises a set of ontology classes that have been used throughout the description of the objects. In case the object

is a user-generated service, when it is created, all possible combinations of ontological representations will be mapped to the concrete service reference. For instance, a user creates a service named “Foodball” which is about the best cooking recipes for watching a football match. The most obvious ontological representation for the service “Foodball” in the search index will be $OR_1 = \{\text{football, cooking}\}$, while the term OR stands for ontological representation. However, since football and cooking are part of an ontology, the different super-classes of Football and Cooking have to be considered as well for referencing this service (see Figure 3).

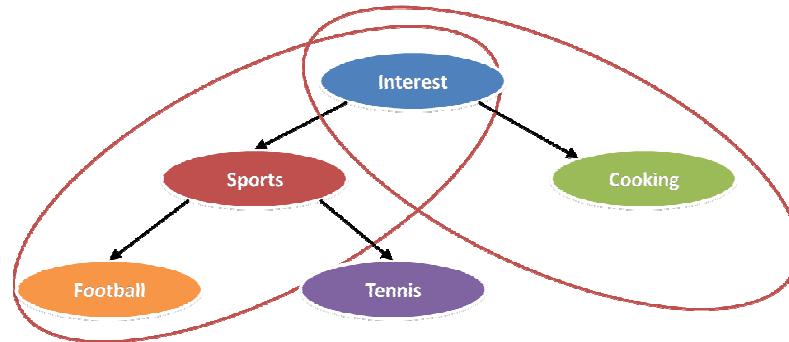


Figure 3. Inheritance hierarchy

Football and Cooking are both subclasses of the class Interest. Whereas cooking is the direct subclass of Interest, Football has the super-class Sports. Therefore, another reference needs to be stored in the search index, namely $OR_2 = \{\text{sports, cooking}\}$. Furthermore, the single values need to be part of the reference list as well, $OR_3 = \{\text{football}\}$ and $OR_4 = \{\text{cooking}\}$. The creation of references stops when a common super-class of Football and Cooking is reached, i.e. in this example when Interest is reached.

In general, the algorithm for the creation of the search index works as follows. At first the meta data and its permutations for describing the new object provided by the creator are stored in the search index. In the example above this would be OR_1 , OR_3 and OR_4 . Having identified the location of the meta data terms in the ontologies, the inheritance hierarchy is taken into account for creating all possible combinations of ontology representations for the new service (in the example this refers to OR_2). As this could lead to millions of ontology representations for a single service we do not only limit it by the inheritance hierarchy, but also by using a generalization depth, which signifies, how often a given class can be inferred to its respective super-class.

Pre-processing of the search query

The simple task for query matching is to perform string matching between submitted query term(s) and ontological concepts and metadata descriptions. However, this will limit the content discovery to a direct matching method and will be keyword and vocabulary dependent. Using look up tables or lexical chains will support finding other variations of a query term and also extending the domain of the search for a user query. The search engine needs to perform an advanced “label matching” in searching the ontology and metadata descriptions. This will extend simple string matching mechanism with an advanced solution which is able to look for other variations of a keyword (or a set of keywords) and is also able to utilise distance measurements to match similar concepts in ontology and metadata descriptions. Lexical chains, Latent Semantic Analysis (LSA) (Landauer, Foltz and Laham, 1998), distance measurement functions used in traditional text analysis mechanisms (e.g. Cosine Distance (Garcia, 2006)) employing common vocabularies for synonyms such as WordNet (WordNet, 2011) and other similar methods can be utilised to support this task.

The possible duplicates have to be consolidated and the associated weights have to be merged. For instance, if the keywords “football” and “soccer” in a search query map to the same concept in a domain ontology, they will be merged into one representation. According to that, the weight of the merged ontology should be higher according to the weights of the respective search terms.

Filtering

From a process view the filtering can be illustrated as in Figure 4. The filtering is triggered with a query and starts the matching of each search term. Having ontologies for each search term an overall ontological representation of the query can be built. This ontological representation is used to query the instance space where each service instance is matched to an

ontological representation. At this stage, two types of results are possible: either a list of service instances is found which match the ontological representation of the query. In this case the filtering process ends; Or no services instances are found so that the instance cache has to be queried again with a reduced ontology set. Reduction is conducted by removing the term which has the lowest weight..

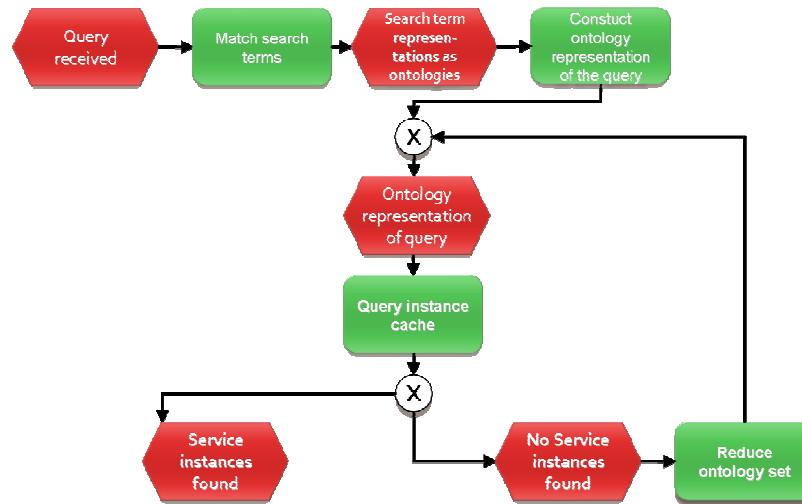


Figure 4. Filtering Process

The filtering process described above mainly aims at the discovery of service instances which match the overall search query. For some use cases we need to extend this mechanism with item-based evaluations that have even to be performed throughout the filtering.

For instance, a service might not be valid anymore or the user might not have access rights to the service. Another example is a service which provides information that is outside the physical range of the user. The role of a user has to be taken into account for the second filtering round, whether the user wants to create, provide or consume a service.

As these interpretations already contain some information which are needed throughout the ranking, these intermediate results should be cached for the ranking later on. Depending on the search scenarios we have (location-based, time-dependent search, etc.), we need a configuration which describes, which parameters should be evaluated throughout filtering.

For all the different aspects that need an evaluation while filtering, this information needs also to be included in the table that maps ontology representations to service ids. For time there could be an additional attribute that can be queried throughout the filtering.

Ranking

After the filtering process, the services are ranked according to their relevance described by the submitted query. The ranking includes an evaluation of the different terms in a query. It uses the different aspects of service descriptions, user profiles and also implicit feedback e.g. from the service usage to evaluate the relevance of a service in comparison to the given query.

The ranking function is a combination of weights of the different search terms and implicit and explicit user feedback. $r(q_s)$ describes the overall ranking result of a service s which was returned as a result for query q . At first the search terms of query q are considered in the ranking function. The sum $\sum_{i=1} r(t_i)$ describes the overall evaluation of the different search terms t_i . Each search term is evaluated by its assigned weight and additional parameters. For instance, search term t_1 specifies that the desired services should provide information not older than an hour. If the service s meets this requirement the Boolean parameter 1 is assigned, if the services s does not meet this requirement the Boolean parameter 0 is assigned. $r(t_i)$ in this particular example is 1 multiplied with the weight assigned to term t_1 for the first case and 0 in case service s does not meet the requirement. Hence, $\sum_{i=1} r(t_i)$ adds the different evaluations of each search term t_i of query q . $r(s)$ represents the feedback about the service provided by the users. Here, different aspects of feedback are considered, e.g. collaborative or hybrid mechanisms. $r(or)$ is feedback about the description of the service, i.e. whether it was helpful. $r(inst)$ and $r(ext)$ are statistical parameters obtained from the knowledge warehouse. $r(ext)$ evaluates the context of a user which can be vital parameters or routes matching a users fitness profile. $\gamma, \delta, \varepsilon, \phi, \lambda$ enable a variation of the relevance of the different terms of the ranking

function and thus, a variation of the order of services in the result list. Furthermore, parameters $\gamma, \delta, \varepsilon, \varphi, \lambda$ have to be adjusted in relation to the user role. For instance, if the user searches for services for consumption γ will be assigned a higher value than φ . In the reverse conclusion, φ is assigned a higher value, if the user acts as services creator.

$$r(q_s) = \sum_i r(t_i) + \gamma \cdot r(s) + \delta \cdot r(or) + \varepsilon \cdot r(inst) + \varphi \cdot r(ext) + \lambda \cdot r(cxt)$$

Figure 5. Formula 2

$r(q_s)$ = Ranking of service s which was returned as result of the filtering process for search query q

$r(t_i)$ = Evaluation of the search term t_i

$r(s)$ = Feedback about service s

$r(or)$ = Feedback about service description of service s which is represented via its different ontologies

$r(inst)$ = Feedback about how often service s was installed

$r(ext)$ = Feedback about how often service s was extended to a new services

$r(cxt)$ = Feedback about how service s matches the context of a user (e.g. vital parameters)

$\gamma, \delta, \varepsilon, \varphi, \lambda$ = Parameter to emphasize or reduce relevance of function terms of ranking equation

Overall the ranking and filtering mechanisms follow a modular design approach which makes them easily extendable with additional filters and ranking sub-functions. Semantic knowledge is used to enrich a query and a search index entry of a service with additional semantic information. The problem of incorporating information which is very little related to the initial query term or service description is tackled by introducing weights which indicate how relevant the added semantic information is.

URUN – A USE CASE SCENARIO

In the uRun scenario, people use the uService framework as a mobile running assistant. As described above different recommendations are provide to the user, to support him before, during and after the training. Furthermore, users can create their own custom mobile services on the mobile device for gathering data from different sources during their runs and for making this information accessible to others (e.g. for doctors or personal trainers) in a secure manner. Moreover, the framework can be used for networking aspects, e.g. for getting in touch with like-minded people or professional fitness service providers. Thus, individual support by professional trainers or medical surveillance is possible. In the following, an example scenario is described with graphical interface mock-ups.

The uRun Scenario

Robert is a runner, who has recently registered for uRun. He is living in Paderborn. On his way to work in the bus, he remembers that he wanted to look for a running partner. He activates the uRun framework on his mobile and starts a search ((a), Figure 6). The results do not match his needs, but an interesting advertisement appears on the screen, promising that Robert can easily build his own search service for a small fee ((b), Figure 6). The advertisement is not sent randomly to Robert, but is matched to Roberts unsuccessful search and user profile. Robert decides to give it a try and starts creating his own service ((c), (d), (e) Figure 6). Henry, another uRun user is on a business trip in Paderborn and does not know the city very well. With the uService platform, he wants to find out whether there are any interesting running events he could join during his stay. When looking for running events he finds the Robert's uService ((f), (g), (h), Figure 3). Since Rober's running services is described with the terms "running partner" and "search" and the system knows form the semantic knowledge base that a running event contains running partners, the Recommendation Engine returns based on the current location of Henry, Roberts's service. (a), Figure 4 shows a targeted advertisement about a running shoe which is the favorite brand of the user and input options for the training target. (b), Figure 4 shows GPS based navigation for the running route with monitoring and recording of different sensor values like temperature, speed and pulse rate. The context-related subfunction of the ranking function is able to recommend different actions to the user based on the current context parameters. (c) Figure 7 shows a recommendation for an equipment rental service offered by Henry's hotel in Paderborn.

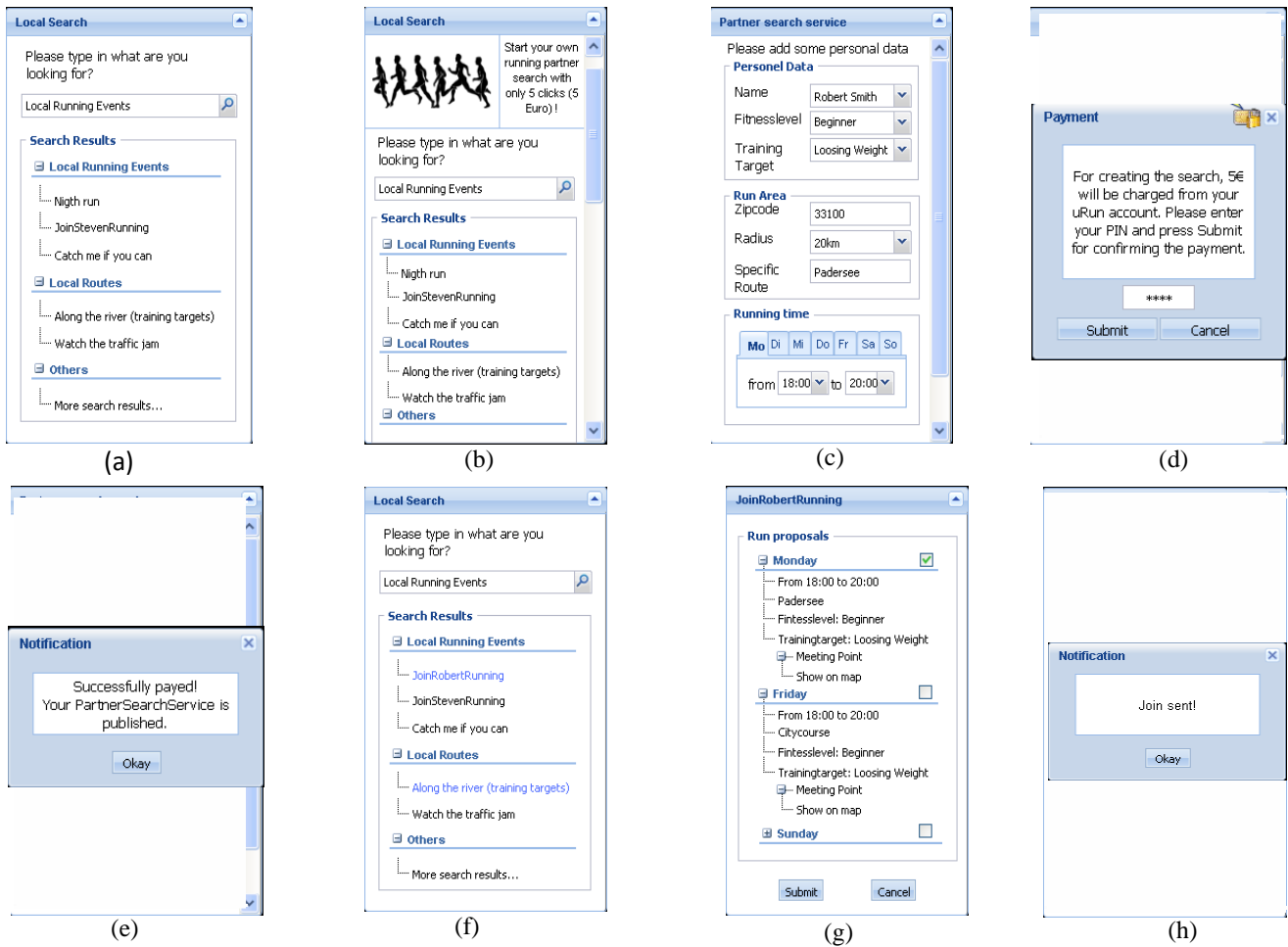


Figure 6. Robert and Henry - uRun use case scenario

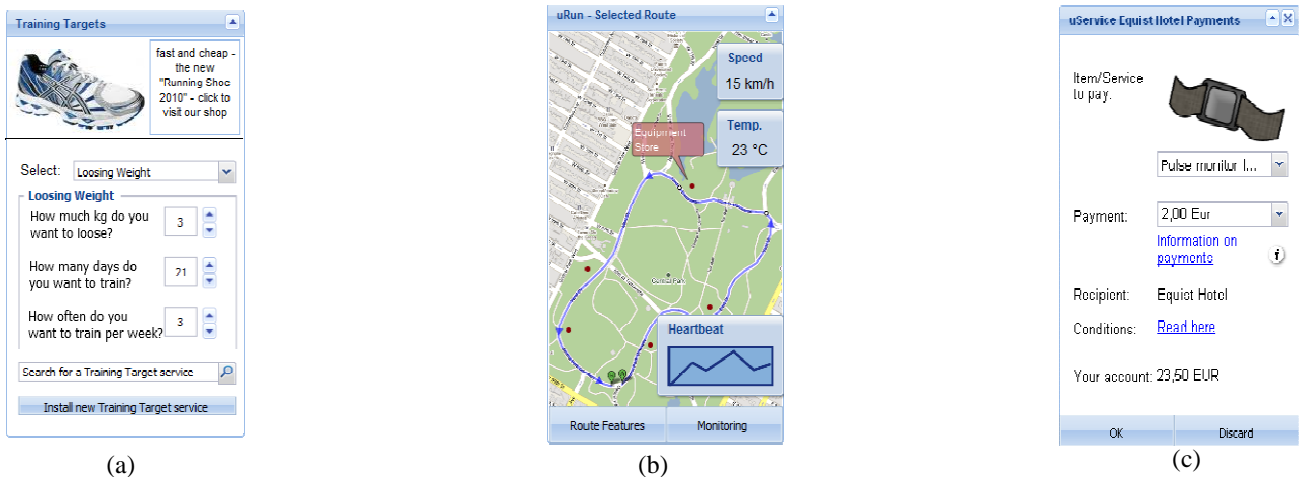


Figure 7. Other uRun use cases

CONCLUSION AND OUTLOOK

The project uService develops a framework for user-generated mobile services and will evaluate the framework with a scenario about health and fitness mobile application. The goal is to combine Web 2.0 and Web 3.0 enabled technologies and recommendation mechanisms to provide a user-centered, proactive system which can support a user anywhere and at anytime with staying healthy. In the following different aspects of the uService system are discussed:

Expert vs. layperson knowledge

As described in the state-of-the-art section mobile running applications and social web technologies applied in the health domain provide the possibility to collect information about health topics provided by professionals. In addition, individual experiences mostly in form of blogs can be provided by individuals. Quality of the former is ensured since information is provided and supervised by professionals. However, information is usually more general and more suitable to get a general understanding of a health issue. Information about a particular health situation or issue is more likely to be found in the latter case. However, information sources are usually laypersons, so that quality and thus usability of the information is questionable. A combination of information sources consisting of laypersons with similar experiences (i.e. using collaborative experiences) and professionals promises to improve the quality of health information.

Proactivity vs. reactivity

The approaches mentioned above of applying social web enabled technologies all described approaches which are reactive and mainly monitor the health status or activities of a user. Despite the fact that there are examples where user can share the collected data with others, the system does not provide any recommendations on how to use the information which was collected about a user. Here, uService provides a hybrid approach for a system, which on the one hand gives professionals remote access to a user, but also automatically reacts to certain data.

User-induced motivation vs. system-induced motivation

Motivational aspects are reduced to the application of social web technologies which are assumed to generate the necessary motivation just because they are social web technologies. By enabling a user not only to create content but also to create health and fitness services it is expected to produce a higher motivation for users to use the mobile assistance system. Mobile user generated services will allow users to develop services tuned to their actual needs providing tailor-made services that will satisfy a Long Tail demand, i.e. small but focused services demanded by a small group of users. As a result, the collective creativity is expected to create services which can motivate every user to work on improving his/her individual health.

Fitness application vs. health application

As described above, the survey of the PWE Internet and American Life Project observed a surge of interest in information about exercise and fitness. In terms of the demographical change, prevention is becoming more and more important so that platforms such as uService promise to have a greater impact on an aging society, since the focus is on preventing health issues rather than managing them. Nonetheless, the system is suitable to serve all kinds of individual health profiles, making it suitable for all types of users.

Mobile vs. stationary

Health and fitness applications on-the-move guarantee an increased value for users. The ability to upload data and receive recommendations anywhere in combination with a scenario about running – an activity which can be carried out almost everywhere - provides the necessary flexibility to a user to stay health and well. Gathering additional information about the current location of the user (e.g. running route in the vicinity or healthy food restaurant) which can support a user to continue with his/her individual recovery or fitness plan are of particular importance for a society on-the-move. In addition, access via stationary devices is ensured as well.

An aspect which has not been discussed in detail is the security of the collected data of a user. In this context, it is planned to use an approach based on SIM card to ensure the secure transfer of sensitive user data.

Overall, the depicted idea can implement intelligent, context-aware assistance in various application contexts, taking into account several types of complex characteristics, such as vital parameters of runners or running route profiles.

REFERENCES

1. Ashbrook, D. (2010). Mobile Health and Fitness. <http://rndbackyard.vodafone.com/2010/02/mobile-health-and-fitness>, last visited Dec 7th, 2010.
2. Smartrunner. (2010). <http://www.smartrunner.de/pages>, last visited Dec 7th, 2010.
3. Runtastic (2010). <http://runtastic.com>, last visited Dec 7th, 2010.
4. Jogging over distance (2010). http://exertioninterfaces.com/jogging_over_a_distance, last visited Dec 7th, 2010.
5. Athlosoft. (2010). http://www.athlosoft.com/shop2/product_info.php?info=p18_athlosoft-runner.html last visited Dec 7th, 2010.
6. Fitbit. (2010). <http://www.fitbit.com>, last visited Dec 7th, 2010.
7. Nike + iPod. (2010). <http://www.apple.com/ipod/nike/run.html>, last visited Dec 7th, 2010.
8. BodyMedia (2011) <http://sensewear.bodymedia.com>, last visited Feb 27th 2011
9. Kamel Boulos, M.N. and Wheeler, S. (2007) *Health Information & Libraries Journal*, 24, 1, March, 2–23.
10. Barsky, E. and Purdon, M. (2006) Introducing Web 2.0: social networking and social book marking for health librarians, *Journal of Canadian Health Library Association*, 27, 65–7.
11. WikiSurgery (2011). http://www.wikisurgery.com/index.php?title=Main_Page, last visited Feb 27th 2011.
12. DrugScope (2011). http://drugscope.blogspot.com/2011_02_01_archive.html, last visited Feb 27th 2011.
13. TRIP (2011). <http://blog.tripdatabase.com> last visited Feb 27th 2011.
14. LibraryThing (2011). <http://www.librarything.com/groups/medicine>, last visited Feb 27th 2011.
15. HealthyPlace (2011). <http://www.healthyplace.com>, last visited Feb 27th 2011.
16. Eysenbach, G., Powell, J., Engelsakis, M., Rizo, C. and Stern, A. (2004) Health Related Virtual Communities and Electronic Support Groups: Systematic Review of the Effects of Online Peer to Peer Interactions, *BMJ*, 328 : 1166 doi: 10.1136/bmj.328.7449.1166, May 13.
17. Fox, S. and Jones S. (2009) The Social Life of Health Information, *Pew Internet & American Life Project*, June, Washington D.C. http://classweb.gmu.edu/gkreps/721/18._Fox-_The-Social-Life-of-Health-Information%5B2%5D.pdf last visited Feb 27th 2011.
18. Pretschner, A. and Gauch, S. (1999) Ontology based personalized search, *Proceedings of 11th IEEE International Conference on Tools with Artificial Intelligence (ICTAI)*, November 8-10, 391-398.
19. Landauer, T., Foltz, P. W. and Laham, D. (1998). Introduction to Latent Semantic Analysis, *Discourse Processes*, 25, 259–284, available at: <http://lsa.colorado.edu/papers/dp1.LSAintro.pdf>, last visited Feb 27th 2011.
20. Garcia, E. (2006) Cosine Similarity and Term Weight Tutorial: An Information Retrieval Tutorial on Cosine Similarity Measures, Dot Products and Term Weight Calculations, available at: <http://www.miislita.com/information-retrieval-tutorial/cosine-similarity-tutorial.html>, last visited Feb 27th 2011.
21. WordNet (2011) <http://wordnet.princeton.edu>, last visited Feb 27th 2011.