DESIGN AND IMPLEMENTATION OF LOCATION AND SITUATION BASED SERVICES FOR A PERVASIVE MOBILE ADVENTURE GAME

Eileen Kuehn, Juergen Sieck

University of Applied Sciences Berlin,
Treskowallee 8, 10318 Berlin, Germany,
Eileen.Kuehn@HTW-Berlin.de / J.Sieck@HTW-Berlin.de, http://inka.htw-berlin.de

Abstract: *This paper describes the design and implementation of a location and situation based pervasive mobile adventure game named Sportix. The prototype uses different types of sensor data – including 3D acceleration data and XPS – to determine the current position and activity of the player. Depending on the firm classification, data and quests are retrieved accordingly from a central information system and visualised for the player on the mobile device. By solving quests and performing sports the player is able to participate in the game.*

Keywords: *Pervasive game, location based services, situation based services, context, sensor networks, mobile phone*

1. INTRODUCTION

In contrast to a traditional game that takes place in the virtual world, pervasive games combine the physical and virtual world by integrating information and communication technology of mobile devices to produce new kinds of game play experiences. They have received continuously increasing attention and have become a popular field of investigation.

Regular games are played in particular spaces at certain times by certain players, whereas recent pervasive games are characterised by the following typical features:
- local dependence;
- mobility;
- connectivity;
- public interaction.

The most pervasive games are based on location-aware services. So the position of the user is the most prominent feature that is examined. However, the use of the player's position is mostly restricted to the outdoor area as the implementation is based on the usage of GPS sensors. Therefore the availability of GPS signals is supposed to participate in the game, otherwise the functionality cannot be assured. This is contradictory to the concept of pervasive computing: the paradigms demand the ease-of-use and the invisible integration of technology into everyday life [1]. Technical details should be hidden from the user so that their actions are not affected.

Furthermore, adaptations regarding the current situation or the environment by sensory perceptions

are difficult to find. The game REXplorer[1] for example uses GPS and additionally the acceleration sensor of a mobile device to realise guided tours. However, the acceleration sensors are only used to recognise predefined gestures to play the game and interact with game characters. Accordingly, the different kinds of sensors are not used to complement one another to assure or increase the quality of data, they are autonomously integrated to support the game in different ways.

This paper investigates the possibility of enriching pervasive games through the provision of an automated context recognition and adaptation to the current position and activity of the user as well as the coupling to a centralised information system delivering context-aware data. The system developed uses the different available sensors of a mobile phone (at least two) and therefore has a versatile and complementary view of the user's environment. If one sensor fails for example relevant data can still be delivered to the user as there is still one available sensor able to provide its data. Therefore the user does not even have to realise the sensor loss and in this way the usability of pervasive games can be increased.

The development and use of location and context sensitive services are the main characteristics of the new type of games. The key technologies for these applications are wireless communication, determination of the position, collection of sensor

---

1 See the website http://wiki.caad.arch.ethz.ch/Research/REXplorer for more details

data and the identification of the actual situation. For example, with the help of the data from GPS and the 3D acceleration sensor we can recognise the activity of use.

The location and situation based pervasive adventure game Sportix currently runs on an iPhone 3G and uses XPS to determine the current position by GPS, cellular positioning services or Wireless Positioning System (WPS[2]) and a 3D accelerometer as well as a server to which the mobile device connects over WiFi. The communication is realised by using standardised interfaces in terms of web services enabling a loose coupling and extensibility.

2. THE DESIGN AND IMPLEMENTATION

Compared to traditional games in virtual worlds, pervasive games require the physical movements and social interactions between the players. The pervasive game uses the real world as the playing field and profits from the advantages and possibilities of the mobile devices, the wireless networks, the sensor systems and the virtual world [2].

This opens a new genre of mobile computer games. The players are doing common activities and the game Sportix introduced in this paper analyses their movements and gives points for the different activities, dependent on the intensity of activity and its duration.

2.1. PRINCIPLE OF GAME

Sportix is a location and situation based pervasive sport adventure game. The player has a small companion (a Sportix) on his phone that is too lazy to be able to go home. With the help of athleticism the player has to collect so-called activity points to get ahead with the game. They represent the current fitness level of the Sportix and can be earned by doing sports, solving quests or challenging his teammates. Once he earned enough activity points, the Sportix can go home.

Different types of activities are classified on the mobile device by analysing available sensor data. The current implementation of the game application can determine four different activities (e.g. Fig. 1):

- resting;
- walking;
- jogging;
- cycling.

Depending on the intensity of the activity the player can earn a varying amount of activity points. They are calculated by using the metabolic equivalent of task (MET). The metabolic equivalent is used as a measuring unit that expresses the energy

costs of a physical activity. The values for common tasks given in [3] have been adapted to meet the needs of the game.
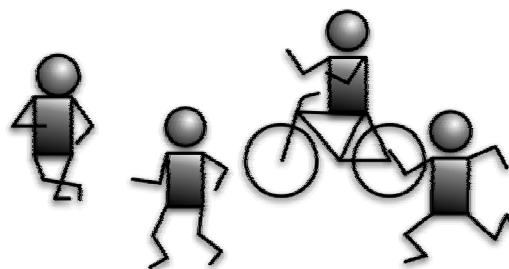

Fig. 1 – Graphical representation of activities in the game Sportix

In Table 1 the calculations for the amount of activity points is specified. Depending on the condition, either the MET base value is used or the calculated value. If the player is jogging with a velocity of more than 7 km/h the activity points are equal to the speed. Otherwise the base MET value represents the progress.

By performing sports and moving to specific locations that have been defined in the content management system of the server the player is also able to uncover parts of the story and special quests enabling them to earn quite more points.

In addition to the quests and tasks that can be found the personal benefits of each player play a major role for the ongoing motivation. Every time a personal record has been achieved a new quest is created on the server and located at the player's current position. These quests can be found and accepted by all players around that position. If they beat that record, their statistics replace the current data and a new quest is created. The original owner of the record is informed about that event and gets the chance to beat the new record.

Table 1. Overview of calculations for activity points depending on the activity

Activity	Condition	MET	Calculation
Resting	true	1	1
Walking	true	1	Speed
Jogging	Speed > 7 km/h	7	Speed
Cycling	Speed > 16 km/h	4	(Speed - 4)/2

Furthermore, the server can automatically generate challenges by identifying players with a similar profile.

Besides an automatic challenge generation, users can directly invite a friend for a competition by defining a combination of rules that have to be solved by the players: the kind of activity, the

duration, the distance that has to be covered or the minimum or maximum speed over time. Sportix evaluates the activities and statistical data, allocates the activity points and determines the winner of a competition. The winner is rewarded with some additional activity points.

2.2. ARCHITECTURE

Situation and context-aware systems can be implemented in many ways. The approach depends on the requirements and conditions as the location of sensors (local or remote), the amount of users or the available resources of the used devices.

A very prominent feature of pervasive games is the mobility. Additionally all game relevant data should be stored on a central instance that can be accessed by mobile devices. Therefore the architecture requires the implementation of a client/server infrastructure. Thus the design of the prototype is based on a loosely coupled client/server architecture that uses a 3-tier architecture concerning the separation of model, view and control. It is composed by a game and an administration application. The game application is designed for being achievable on a mobile device whereas the administration application represents the central instance and offers appropriate functionality to the client.

2.2.1. GAME APPLICATION

The recognition, processing and storage of context are central features of pervasive games. Context-awareness enables the automated adaptation to a user's situation and the automated recognition of their feasible intention. This is done by analysing and merging different types of raw sensor data being offered by the iPhone SDK, i.e. the data of the 3D acceleration sensor and location information through the CoreLocation API.

The sensor data are known as physical data as it is retrieved by physically existent sensors on the mobile device. It is the most frequently used type of sensor and nowadays many hardware sensors are available being capable of capturing almost any physical data.

On current mobile devices there is only a small subset of sensors available that are able to measure light, the visual context, audio, motion and acceleration, location or touch. Concerning the way data is captured, sensors can be classified in three groups [6].

Besides physical sensors there are also logical and virtual sensor data that can be accessed and used to evaluate the situation of the user. Virtual sensor data can be retrieved from software applications or services. For example, it is possible to determine the weather not only by using thermometers but also by a virtual sensor, e.g. by accessing information using a web service. Whereas logical sensors make use of several information sources and combine physical and virtual sensors with additional information in order to solve higher tasks. On this way the location of a user can be classified as urban or suburban by evaluating the number of available WiFi networks in the neighbourhood.

By combining different sensors a versatile and complementary view of the current environment can be evaluated. Therefore the current implementation is based on physical sensors offering acceleration and positioning data as well as logical sensors providing data about the connectivity state. Generally our implementation of the pre-processing and extraction of features is done on the mobile device. In this way the costs for playing the game can be minimised as the communication with the server does not have to be maintained.

The different steps for identifying the current situation are shown in Fig. 2. It visualises a layered conceptual architecture for the implemented system.
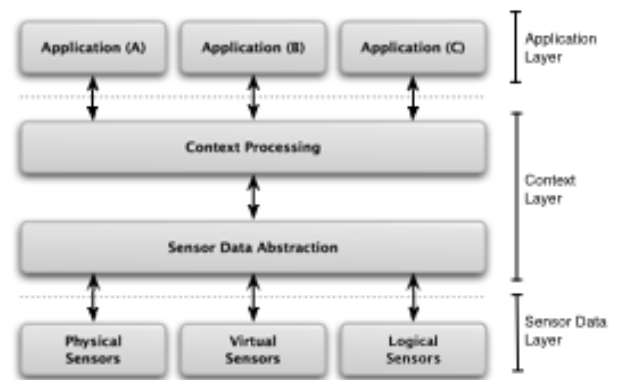
Fig. 2 – Basic architecture of implemented context recognition

The architecture of the context recognition system consists of four different layers and augments layers for detecting and using context by adding the context processing and reasoning functionality. The first layer, the sensor data layer, consists of a collection of different sensors: physical, virtual and logical sensors. It is responsible for providing access to different services and components by using appropriate drivers for physical sensors and APIs for virtual and logical sensors. It offers basic information and data.

As the visualisation of different components in the sensor layer indicates, context can be specified by different sources of information. Therefore the context layer has to abstract from the heterogeneity.

As the system consists of more than one context data source, the single context atoms can be combined to high-level information in the third layer

– titled context processing. The different services and sensor data are analysed and features extracted to determine the context. This process is also called aggregation or composition. In the last step these data can be accessed via a public interface by the application layer to offer context specific adaptations. Applications may gain access in two different ways: synchronous and asynchronous. When using the synchronous mode the client is polling the data for changes. The asynchronous mode works via notifications. Each client subscribes for specific events and gets notified as soon as data gets available.

The CoreLocation API of the iPhone SDK enables the access to the user's current position being determined by cellular positioning services, GPS and WPS. The combination of different services for the localisation enables our implementation to use the position in very different scenarios.

As already mentioned before classical games mostly depend on the availability of GPS. GPS is able to provide very accurate location results under certain circumstances – mainly in rural areas where you have a clear line of sight. But it will not work indoor and its accuracy tends to degrade significantly when the receiver is located under shelted area or near high-rise buildings. This fact implicates the absence of positioning information if the player is actively playing the game in the indoor area. But the requirements for the game Sportix depict the ease-of-use and invisible integration in everyday life. The game must not be constrained by missing positioning data. Moreover the user should also be able to do indoor sports by working out at a gym or by going upstairs and downstais. Therefore the exclusive usage of GPS was excluded during the conceptual design of the prototype.

The evaluation of different technologies showed that Skyhook Wireless' XPS that combines the benefits of GPS, Cell Tower triangulation and Wi-Fi Positioning fitted our requirements best. XPS has a portable architecture and "can be configured to run across a broad range of mobile operating systems, devices and platforms" and the iPhone already has it built-in (http://www.skyhookwireless.com).

The Wi-Fi Positioning System (WPS) determines the location by analysing the current fingerprint of currently available access points. It performs best where GPS is weakest. This means in urban areas and indoors. Next to WPS and GPS Cell Tower Triangulation is used as a coverage fallback when neither GPS nor WPS is available. In this cases location results with only 200 – 1000 meter accuracy can be reached.

Because of this technology the Core Location framework of the iPhone SDK establishes a good basis for the implementation. It reports the location and updates the positioning information as it receives new or improved signals.

Besides that big advantage of combining three technologies for positioning there are some characteristics and quirks that should be mentioned: As the framework records timestamp values at the beginning of each location query positioning data is often being delivered in the wrong order. So we are not able to assume that we receive the best possible location. Because of the combination of different positioning technologies to get a location fix the usage is not constrained to the outdoor area. In contrast to that benefit the highly varying quality and accuracy of the position determined by these technologies has significant negative impacts on the results and context recognition itself. The accuracy ranges from 9 metres up to about 2.000 metres. Typically the positioning data determined by GPS is the best available data, provided that the signal is available. By interpreting the vertical accuracy of the delivered position the differentiation between the different techniques becomes possible – for example WPS and the Cell Tower Triangulation have a negative value whereas GPS always provides a value greater than 0.

The positioning data is especially used to calculate the current velocity of the user. The usage of extremely fluctuating data would have an impact on the measured distance and the calculated speed. This is shown in Fig. 3. The varying accuracy generally means an inappropriate position for the user's current location. This has an impact on the presumed activity of the user. As a consequence the activity points could not be calculated correctly and the game could be fooled by exploiting the fact of inaccurate positioning data.
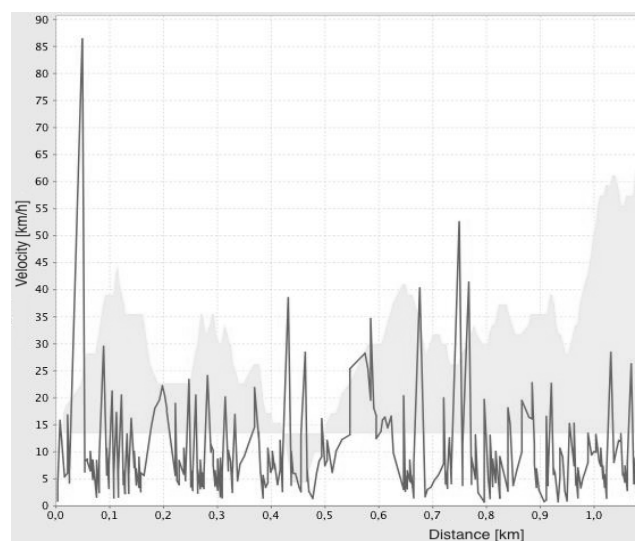
Fig. 3 – Influence of fluctuative positioning data on calculated velocity

To optimise these data the extracted and merged features are also filtered and therefore a chain of different filter types has been developed.

The positioning data is filtered using three different components:

- a general filter determining whether current GPS-only values are accepted by the context recognition system;
- a dynamic threshold filter that takes care of accepting only values that have the correct accuracy for optimal, but always available positioning data,
- a specific filter determining if the calculated speed meets the needs of the current kind of sport.

The values not being filtered are directly used to calculate the users current speed as they assure a good quality of initial calculations in the majority of cases. The calculated speed by itself cannot be used to clearly determine the user's current activity, but it enables the constraint of possible activities: a pace of about 11 km/h can be assumed to be a fast run or a slow cycle.

Next to the speed and position of the user the data of the 3D accelerometer is used. Different features such as

- mean;
- standard deviation;
- energy;
- correlation;

are used to determine the activity more precisely. The usefulness of these features has already been demonstrated in prior work (see [5] for more details).

In the current implementation a decision tree is used to map the features that were deduced by raw sensor data to the currently supported activities. It is based on thresholds that were retrieved by measuring the performance data of a group of sportsmen. As a basis of measurement it was defined that the mobile device has to be carried in the hand of the player. In this way, characteristic data for standing, walking, running and cycling was gathered (see Fig. 4). In this way the current activity is determined based on the combination of aggregated features of speed and acceleration data.

As a last step these results are monitored over a fixed period of two seconds before visually presenting the current activity to the player. Fig. 5 for example illustrates the activity standing and the current duration.
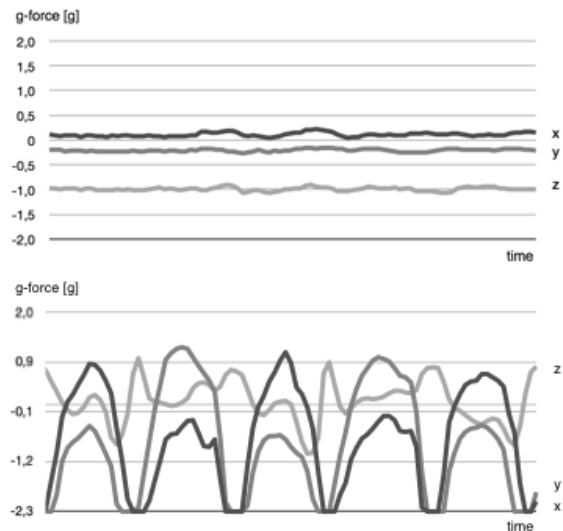
Fig. 4 – Characteristic data delivered from accelerometer for the activities 1) standing and 2) running

The dynamic infrastructure and the mobility of the player imply that information and technical features do not have to be available at all times. The challenge of connectivity and sensor loss requires a robust game design. The context being determined by the positioning data as well as the information gained by the acceleration sensor can be used to draw a conclusion concerning the activity of the user.

Therefore the context can also be determined if only one of those components is available. Because of the missing view on the environment if only a single sensor is used the result cannot be guaranteed by the game Sportix.

Fig. 5 – Sportix action resting and list of available quests

Besides the handling of sensor loss, there have to be various caching techniques able to deal with

different connectivity states or incoming calls on the mobile device. Especially the constraints of the iPhone and the missing functionality of running applications in the background is a significant limitation. To handle these issues game and activity relevant data are written to the hard drive every minute. In this way important data can be restored after an interruption of the application. One possible and common interruption is an incoming call: the application is finished by the iPhone OS and will be restarted after the call is finished. If no data were saved by Sportix the current activity, game progress and points the player gained during its current session would be lost. So it is an important part to save and restore the game and context relevant data.

Furthermore, in areas relevant to security an enormous reliability of the system has to be reached. Depending on the fact of whether the user trusts the system and its handling of the player decides to use the system.

2.2.2. ADMINISTRATION APPLICATION

Next to the game application that runs on a mobile device there is also an administration application offering the possibilities of a content management system to create, manage and delete game relevant contents and a user administration. It represents the central instance of the implemented architecture and offers the game contents and functionalities for different clients via web services.

With the help of the administration application users in the role of an author or administrator are able to modify game relevant data. New quests or parts of the story can be created by defining a name of the game element and descriptive text. In addition context relevant data can be added that specifies the situation or position when it will become accessible to the player. This can be done by adding a specific kind of activity, duration, speed or position for example.

Next to the possibility to edit quests, the storyline of the game or user specific data, the content management system offers the opportunity to monitor the current state of the players. Authorised people can view a list of players, each player's rank inside the game and their current game progress (see Fig. 6.1). Currently active players are additionally linked to a detail page that allows the players current position to be visualised on a map. Fig. 6.2 for examples visualises the integration of Google Maps for showing the player's position.
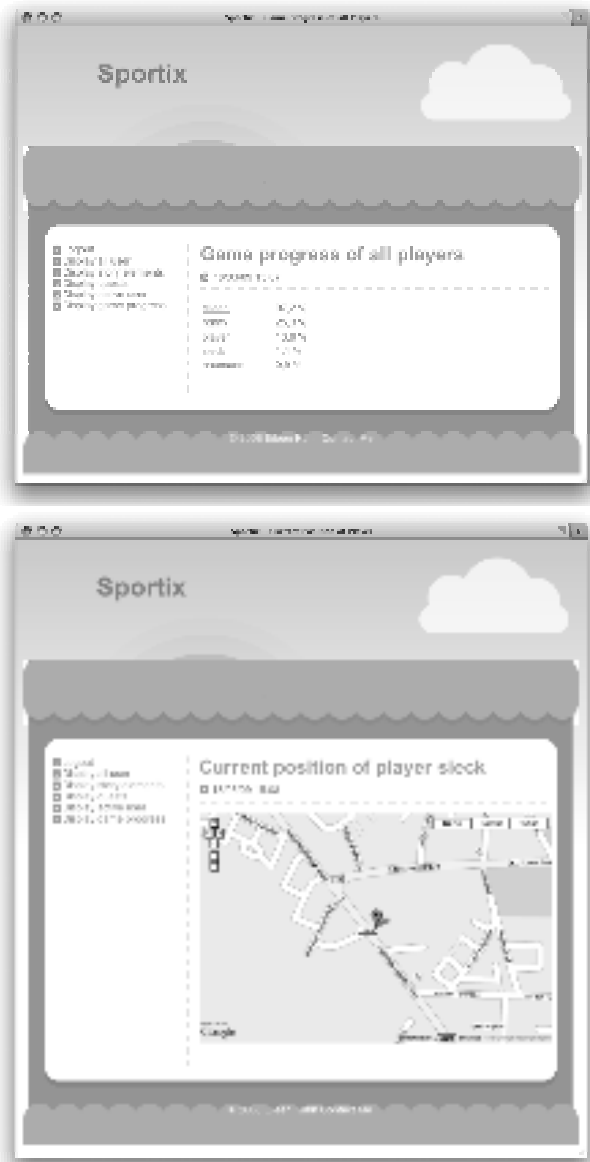
Fig. 6 – Screenshots of the content management system showing 1) the game progress and 2) the current position of the selected player

The content management system is only one important part of the administration application. Besides this, it offers different web services being used by the game application. The server offers functionalities for client devices to register new players, verify its login data, to transmit the current context and activity, its current position or to request context specific data. To enable the implementation of a large variety of devices the implementation of the web services relies on the Extensible Markup Language (XML) by using SOAP over HTTP as its message format.

The server provides the data being relevant to the client by analysing the current context and activity profile of the specific player. Depending on the results, lists of story elements, quests and players can be transmitted.

Moreover, challenges that can additionally be delivered to clients are periodically created by the server. Depending on players being active at the moment when a challenge will be generated, the server calculates the similarity of user profiles by checking their preferred kind of sports, average speed and duration. Active players are determined by checking the timestamp of the last transmission. If it occurs within a time span of 30 minutes the player is remembered to be active. Due to the fact that a scan of statistical data for all active players cannot be accepted in terms of complexity and delay, the current implementation is based on the 1/e-law of best choice (see [7] for more details). One third of the active players are tested against one randomly chosen player. The best matching player is assumed to be the counterpart during that phase of analysis. After that, the remaining players are checked until one is found, being better than the currently selected opponent (see Fig. 7). By this means the best applicant is selected with a probability of 1/e. After having found two players with similar behaviour and experience a challenge is generated, based on the preceding calculations. These challenges can be retrieved by the clients and solved synchronously as well as asynchronously.

The synchronous mode is started when both players accept the challenge at the same time. In this mode the player obtains some feedback about the current data of its counterpart in near real-time. If the challenge is started in the asynchronous mode one player solves the task as best as possible and the other has to beat the given time, speed and duration. Handling challenges in this way also enables an offline mode for the game: if no connection can be established during a competition the data is transmitted as soon as it becomes available and the challenge will be handled asynchronously.

3. CONCLUSION AND FUTURE WORK

This paper presents a location and situation based pervasive adventure game named Sportix, which combines the results of physical and logical sensor data to determine the player's current activity and position. In Sportix, players can find and solve quests by acting in a casual way, i.e. the application automatically determines and adapts to their location and activity by evaluating the 3D acceleration sensors and position data from the iPhone 3G.

The implemented prototype has been tested in performance trials as well as in initial field trials. Even though tests are ongoing, the preliminary tests demonstrated that the current implementation provides good results if all sensor data are available.
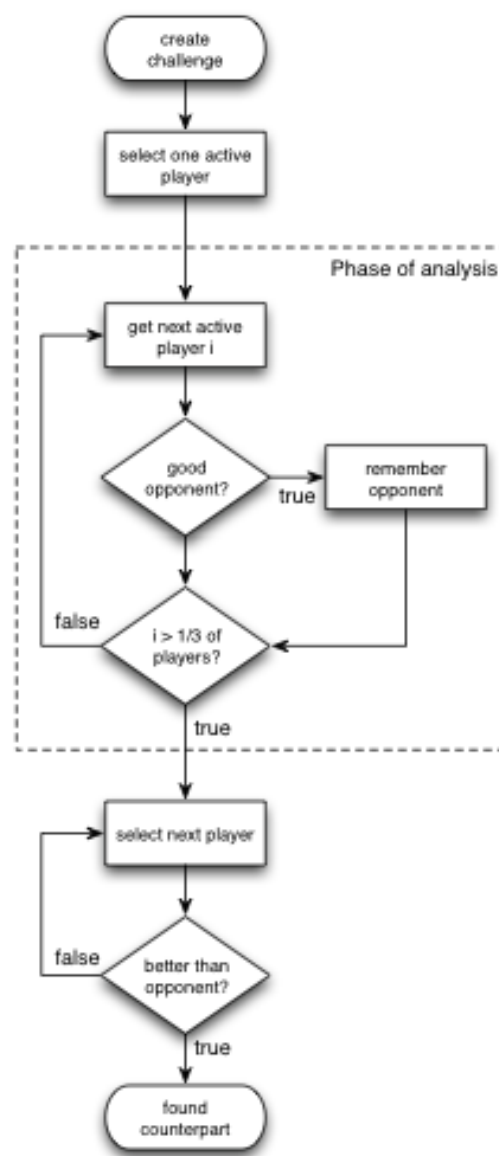

Fig. 7 – Activity diagram for choosing matching opponents

Some issues are especially related to the iPhone 3G. The positioning data are frequently not available or offer very bad accuracies of about more than 300 metres. For an accurate determination of the activity the prototype needs at least an accuracy of about 50 metres. As a fallback the acceleration data is used to determine the activity with a limited rate of recognition.

The basis for an apparently reliable and accurate location and situation dependent mobile application has been developed. Future work will focus on investigation of unsupervised recognition algorithms as well as the extension of the context framework by using additional sensors. Possible additional sensors as radar systems or RFID are able to boost the accuracy of the geographical context and its derived data such as the movement of the user or his speed

within buildings. Furthermore the context can be expanded by using other classes of sensors, e.g. web services about weather or wear links enabling a broad variety of situation-based applications or any other application which can take advantage of using sensor networks.

A first step towards the implementation of a real pervasive infrastructure is to enable ad hoc networks between single players and different devices. As a result, social interaction and communication can be distinguished in the principle of game and will help the player to get in contact with other players nearby. Players could perform sports in common groups or in competition with each other. Additionally, the game could operate without a server by utilising the ad hoc network between different players whilst interchanging game relevant data.

Furthermore the flexible infrastructure of the system facilitates the implementation of more classes of devices using the centralised services of the administration application.

4. ACKNOWLEDGMENT

5. REFERENCES

[1] Roy Want. Remembering Mark Weiser: Chief technologist, Xerox PARC, *IEEE Personal Communications* (2000), pp 8-10.

[2] Carsten Magerkurth, Adrian D. Cheok, Regan L. Mandryk, Trond Nilsen. Pervasive games: bringing computer entertainment back to the real world. *Comput. Entertain. 3* (2005). Nr. 3, p. 4 – ISSN 1544-3574.

[3] B. E. Aintwort et al. Compendium of physical activities: An update of activity codes and MET intensities, *Medicine and Sciences in Sports and Exercise* 32 (2000). pp 489-516.

[4] Research group "Information and Communication Systems", HTW Berlin, http://inka.htw-berlin.de.

[5] L. Bao, S. Intille. Activity recognition from user-annotated acceleration. *Proceedings of the 2nd International Conference on Pervasive Computing* (2004). pp 1-17.

[6] J. Indulska, P. Sutton. Location management in pervasive systems, *CRPITS'03: Proceedings of the Australasian Information Security Workshop* (2003), pp 143-151.

[7] Franz T. Bruss. *Strategien der besten Wahl* (2004), pp 102-104.

Eileen Kühn graduated as an engineer for computer science in 2009 from the University of Applied Sciences (HTW) Berlin, Germany. Scince 2009 she is part of the POSEIDON project team in the research group INKA at the HTW where she currently develops position and context-aware services by using active and passive RFID technologies.

Her research interests are in ubiquitous and pervasive computing, context-aware services, applied algorithms and information retrieval.

Prof. Dr. Jürgen Sieck received his degree in mathematics in 1981 and his PhD in computer science in 1989 from the Humboldt University Berlin, Germany. Scince 1994 he is a professor of computer science at the University of Applied Sciences (HTW) Berlin and head of the Research group INKA. He teaches algorithms and data structures, multimedia, virtual reality and telecommunication. Juergen Sieck also taught and researched abroad, he was guest professor at the Johannes Kepler University Linz, Austria, visiting professor at the Old Dominion University Norfolk, USA, at the ICSI in Berkeley, USA and at the Monash University Melbourne, Australia.

His research interests are in multimedia, computer graphics, virtual reality and wireless communication.