

# ESMAC: A Web-Based Configurator for Context-Aware Experience Sampling Apps in Ambulatory Assessment

Anja Bachmann, Robert Zetsche, Till Riedel, Andrea Schankin, Michael Beigl  
Karlsruhe Institute of Technology (KIT)  
Pervasive Computing Systems  
Karlsruhe, Germany  
{bachmann, zetsche, riedel, schankin, michael}@teco.edu

Markus Reichert, Philip Santangelo, Ulrich Ebner-Priemer  
Karlsruhe Institute of Technology (KIT)  
Applied Psychology  
Karlsruhe, Germany  
markus.reichert@partner.kit.edu,  
{philip.santangelo, ulrich.ebner-priemer}@kit.edu

## ABSTRACT

In ambulatory assessment, psychologists apply experience sampling methods (ESM) on mobile devices to assess self-reports from subjects. One major challenge is to support domain experts to create ESM apps themselves without prior programming knowledge. When running ESM apps, subjects are prompted to answer self-reports time-triggered at fixed points in time or randomly. The compliance of the subjects often drops due to a high frequency of prompts or a high number of questions to be answered. We propose *ESMAC*, an open-source ESM app configuration system that is easy to use by non-programmers and able to create context-aware apps. Leveraging context-awareness can counteract a drop in compliance by prompting event-based only in situations of relevance (reducing the frequency) and by automatically assessing information (decreasing the number of questions). The *ESMAC* web interface for configuring ESM apps was evaluated with two psychologists. One of their configurations was deployed and evaluated in a preliminary user study with ESM subjects. Both experiments yielding good results using SUS and UEQ benchmarks. In addition, we analyzed the share of triggers and identified that 84% of all prompts were event- and not time-based. This emphasizes the relevance of event-triggers.

## Categories and Subject Descriptors

H.4 [Information Systems Applications]: Miscellaneous;  
J.4 [Social and Behavioral Sciences]

## Keywords

Mobile Sensing, Context Awareness, Experience Sampling, Ambulatory Assessment

## 1. INTRODUCTION

Ambulatory assessment refers to "the use of computer-assisted methodology for self-reports, behavior records, or physiological measurements, while the participant undergoes normal daily activities" [3]. Studies performed by psychologists show a need to assess subject information such as social interaction, physical movement or travel patterns [5]. Traditionally, this information is gathered using self-report questionnaires that are prompted on a timely basis, either random or time-triggered [4]. Though, self-reports often put a burden on the subjects due to their complexity and a high number of prompts.

This issue can be addressed analyzing sensor measurements and by prompting event-based, e.g. in case a threshold is exceeded such as "heart rate above 100bpm". What is left to do is to identify relevant events to trigger prompts. We propose a new frontend that easily allows non-developers to leverage smartphone sensors and context recognition methods to infer the occurrence of relevant events.

Though smartphones are promising candidates for context-aware assessment, many psychologists lack of programming knowledge. They have a need for user-friendly and easy-to-use platforms that allow them to create their own ESM apps. We propose *ESMAC*: a context-aware ESM app configurator for Android apps.

## 2. RELATED WORK

Platforms already exist that allow non-programmers to create their own ESM app that logs context information. We will focus on those that offer event-triggers in addition to time-triggers.

One of them is *Ohmage*<sup>1</sup>. It allows time- and location-triggered prompting of self-reports. Additionally, it collects sensor data from accelerometer, wifi, mobile radio cell and GPS. Ohmage allows users to adjust the event-triggers to enhance compliance. Though we prefer to give those privileges to the experts.

Another platform is *MyExperience*<sup>2</sup>. Experts can design ESM studies by choosing from a set of question types and by selecting sensors to be accessed, e.g. GPS, GSM or keystroke dynamics. In addition, experts can define event-triggers

<sup>1</sup><http://http://ohmage.org/>

<sup>2</sup><http://myexperience.sourceforge.net/>

based on additional, external sensors, e.g. self-reporting triggered by an increase of heart rate. Though, MyExperience lacks programming support. Users have to configure their study and triggers in XML.

A more user-friendly ESM platform is *movisensXS*<sup>3</sup>. It offers additional wearable sensors such as activity, ECG or EDA sensors. Based on measurements from these sensors, study designers can select heart rate variability, level of activity and GPS location as event-triggers. Unfortunately, *movisensXS* is not free-to-use and not extensible. It does not leverage the capabilities of smartphones as many event-triggers are not yet offered.

We want to extend current systems and provide an easy but free to use ESM configuration platform that is extensible and offers a wider range of accessible smartphone sensors and, thereby, event-triggers.

### 3. THE ESMAC SYSTEM

#### 3.1 System Architecture

*ESMAC*<sup>4</sup> is an open-source, extensible *ESM App Configurator*. It combines the benefits of *movisensXS* (web interface for non-programmers to configure individual ESM apps) and MyExperience (XML-based configuration of event-triggers). It consists of three components: a web interface, an android app and an exchange format (see Figure 1). The *web interface* allows the study designer, further called user, to create mobile ESM applications by selecting question types and sampling strategies, which are either time-based, event-based or both. The *android app* adapts to the ESM configurations, i.e. by providing the configured study and by collecting data from selected sensors. To exchange data between the two applications we apply an XML-based *data exchange format* defined by a specific XML schema.

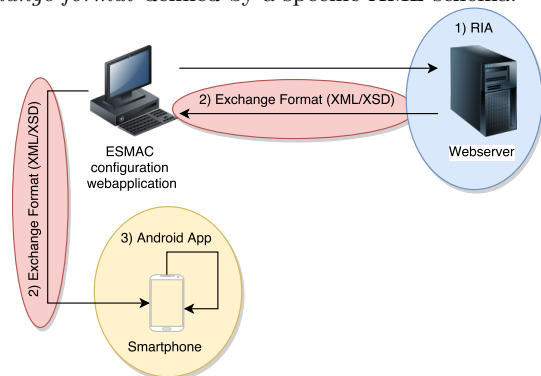


Figure 1: Core components of the ESMAC system.

To create an ESM application with ESMAC there are four required concepts: forms, rules, sensors to be logged and notification type. Similar to current systems and for a better overview, we split them into three views, one for forms and rules, respectively, and one joint view for sensors and notification type.

**The Form View** In the form view the user specifies the ESM study by selecting question types and posing questions. The following question types are available: open questions, single and multi answer questions, slider questions and Likert scales. In addition there are conditional questions which

<sup>3</sup><https://xs.movisens.com/>

<sup>4</sup><https://github.com/teco-kit/ESMAC>

are only displayed if an expected answer was given on the corresponding question.

**The Sampling View** The sampling view is based on a boolean algebra mechanism and specifies triggering rules. They can either be time-based, event-based or both. A rule consists of different sensor expressions. A sensor expression is a concatenation of a sensor type, a value type, an operator and a value. Available sensors are: time, call log, Bluetooth, notifications, user activity, display state, weather, accelerometer, GPS and ambient light. These sensors were selected based on an earlier online study [1].

Sensor expressions are combined using *and* or *or*. The resulting expressions can either be true or false. Two examples for sensor expressions are "weather: degree > 10" and "screen activity = true".

**The Sensor and Notification View** In the last view the sensors to be logged and the notification type are selected. The user can choose among a considerable amount of sensors [1]. To notify the subject the user selects one or a combination out of ring tone, vibration or notification LED. Additional settings such as maximum number of prompts per day and cool down time for a new prompt can be specified in this view as well. The user can select if voluntary self-reports shall be allowed or not.

All these configurations are transformed in a XML representation, downloaded by the user and transferred to the smartphone.

#### 3.2 Android App

The Android app validates and parses the XML representation, generates the specific UI and adapts to rule and sensor logging configurations in a dynamic manner. The app runs on all Android 4.4 and above smartphones. Figure 2 shows an example UI of an ESM study.

To get access to different sensors we implemented an open-source, extensible *sensor library* which uses an observer principle to access sensors in an energy-efficient manner. After a relevant value has changed the evaluation mechanism checks each rule. The subject is prompted in case a rule is true, the maximum number of prompts per day is not yet reached and the time difference to the last prompt exceeds the cool down time. A prompt is carried out by an Android notification.

### 4. EVALUATION

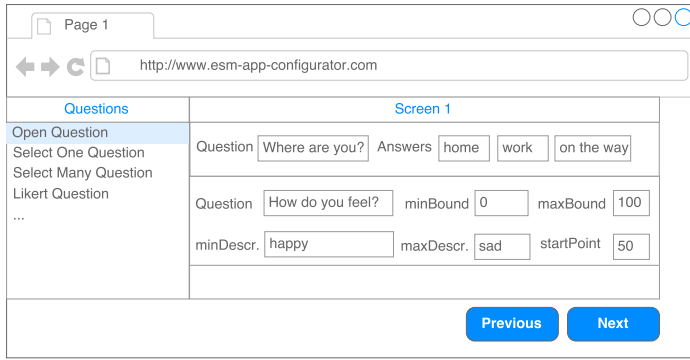
The evaluation of the system was divided into two parts according to the two main components of the system: the web interface and the Android app. In both cases, ESMAC was tested against the *movisensXS* platform as their architectures and functionalities are similar and as *movisensXS* is the most prominent and still applied state of the art approach.

#### 4.1 Evaluation of the Web Interface

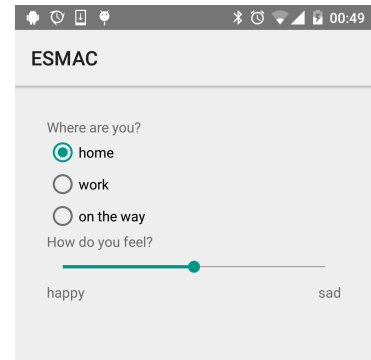
##### *Study Design.*

The web interface was evaluated by two ambulatory assessment experts who participated in the sensor selection survey (see [1]) and who used *movisensXS* before. We confirmed with them that all tasks could be performed efficiently on both platforms.

The experts were asked to design a short ESM study that aims at assessing the emotion of students in everyday life during lecture time for three days. The conditions were



(a) Web Interface



(b) Android App

Figure 2: Visualization of a configuration within an exemplary web interface’s question frame and within the Android app.

counterbalanced, i.e. one expert started using movisensXS and ESMAC afterwards, the second expert the other way round. To measure the usability, user experience, and mental workload of the experts, three standardized questionnaires namely System Usability Scale (SUS) [2], User Experience Questionnaire (UEQ) [8], and NASA Task Load Index (NASA-TLX) [7] were handed out. In addition, we handed out a free text questionnaire to receive qualitative feedback.

### Results.

Both systems achieved an average SUS score of over 90 points, ESMAC 95 and movisensXS 93.7, and are located under the best 99.9999% of all applications [10].

Concerning user experience, we reviewed the results in each of the six dimensions of the UEQ as summarized in Table 3. Schrepp et. al provide a benchmark to classify UEQ average scores [11]. Based on this benchmark, both movisensXS and ESMAC show *excellent* results in the dimensions attractiveness, perspicuity, efficiency, dependability and stimulation. For both apps the results are fairly equal. ESMAC scores slightly better in attractiveness, perspicuity, efficiency and stimulation. Though it has marginally lower values in the dimensions of dependability and novelty what is probably caused by the additional effort to define event rules within ESMAC.

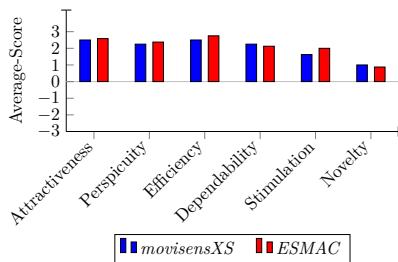


Figure 3: UEQ results for both web interfaces.

The mental workload measured by raw NASA-TLX was 24.58 for ESMAC and 25.83 for movisensXS. Neither of them tends to underload or overload [6].

Both applications show a great performance in usability, user experience and mental workload. This shows that ESMAC is comparable with the state of the art and that an additional event rule configuration influences neither usability, user experience nor mental workload.

In the free text questionnaire one expert noted that ESMAC increases the possibilities of performing ESM studies due

to its possible combinations of sensor sources and event-triggers. This emphasizes the relevance of event-triggers.

## 4.2 Evaluation of the Android App

### Study Design.

We evaluated a representative app configuration which was created in the first study within an in-field user study. The ESM configuration was as follows:

- **Questions: 6-Point Likert Scales**
    - How do you feel? energetic - tired
    - How do you feel? tense - relaxed
    - How do you feel? good mood - bad mood
  - **Rules**
    - 17 randomized time triggers from 08:00 to 23:59
    - Number of Bluetooth devices  $\geq 2$  (\*)
    - Call status = answered (\*)
    - User Activity = walking, running, on bicycle, in vehicle (\*)
    - Notifications of WhatsApp, Facebook (\*)
- (\*ESMAC only due to its event-triggers)

We recruited 10 subjects (7 male, 3 female) with an average age of 23.6 years. The subjects were divided into two groups with counterbalanced conditions. Group 1 (subject 1-5) first experienced the ESMAC app and afterwards the movisensXS app – each for three days – and group 2 (subject 6-10) vice versa. After finishing each condition, the subjects were handed a SUS, a UEQ and a free text questionnaire. At the end of the study another free text questionnaire was handed out to compare the applications.

### Results.

Both SUS and UEQ scores differ significantly between the two apps. ESMAC reaches an average SUS score of 83.5 whereas movisensXS only reaches an average score of 74.25.

The UEQ scores are visualized in Table 4. ESMAC dominates every scale with the highest difference in the attractiveness scale. In addition, only the perspicuity dimension for both applications and efficiency dimension for ESMAC reach the *excellent* class of the UEQ benchmark [11].

We investigated the average response time (time between prompt and answer), response rate (how many prompts did the subjects react to) and answering time (time between reacting to the notification and submitting the self-report). The reaction time of movisensXS is significantly low (10.21s)

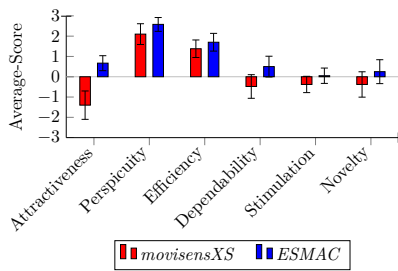


Figure 4: UEQ results for both web interfaces.

whereas subjects took more time responding to ESMAC prompts (783.87s). Both lie in the required time limit of 20 to 30 minutes [12]. The higher reaction times of ESMAC are caused by ESMAC allowing prompts at night as well and the WhatsApp event-triggers. Many subjects received messages in the night that were answered in the morning, causing higher response times. Only considering day-time prompts, ESMAC’s response rate is reduced to 420.58s.

The average response rate of prompts for movisensXS is 76.72%, ESMAC only reaches 46.12%. Low values are again often caused by nightly prompts which were ignored by a sleeping subject and deleted from the notification view once a new prompt came in. Considering day-time prompts only, ESMAC’s response rate is 64.07%. In addition, movisensXS has a more obtrusive notification type. ESMAC only uses a single notification whereas movisensXS uses a repeated alarm clock. There is a clear trade-off between user acceptance and processing rate or reaction time, respectively.

For ESMAC we were able to measure the answering time as well. The mean is 42.68s. Some outliers, e.g. one response time of 1.15h, distort the results. Such high response times are often caused by subjects opening the app, getting distracted and finishing the self-report much later. Hence, we also investigated the median value which was only 12.18s – a fair answering time for 3 questions.

We also investigated the share of trigger types to investigate the power of event-triggers (see Table 1). 84.47% of all prompts are event-triggered which shows the effectiveness and importance of the event-based approach.

	Notifi- cations	Blue- tooth	User Activity	Call	Time
Share	42.92%	29.22%	11.57%	0.76%	15.53%

Table 1: Share of all trigger sources.

## 5. CONCLUSIONS

Smartphones are a ubiquitous, wearable sampling system that can be used for ambulatory assessment [9]. By now, self-reports are provided using ESM apps running on the subject’s own smartphone. Study designers without programming experience have a need for an easy-to-use platform that supports them in creating their own ESM app. Especially in ambulatory assessment studies there is a need for event-triggers based on embedded sensor measurements. So far, related work only covered one of those aspects.

We propose ESMAC, the ESM app configurator, which offers (1) a web interface for non-programmers to configure their own context-aware ESM apps and (2) an Android app that adapts to the configurations from the web interface. Both were evaluated compared to the state of the art system movisensXS in terms of usability and user experience, achieving similar or higher values in SUS and UEQ. We also

identified that a study designer has to find a trade-off between user experience and response time. An analysis of prompt triggers showed that 84.47% of them were event-triggered, thus confirming the importance of event-triggers.

A next step is the inclusion of machine learning into the system to allow further interpretation of sensor data to derive more contexts and provide additional event-triggers. Due to their rising number, wearables such as smartwatches are good candidates for complementary sensor sources.

## 6. ACKNOWLEDGEMENTS

The authors would like to acknowledge the funding by the German Federal Ministry of Education and Research (BMBF) under the project Software Campus and the micro-project ”MARC<sup>2</sup>” (Project No. 01IS12051).

## 7. REFERENCES

- [1] A. Bachmann, R. Zetzsche, T. Riedel, M. Beigl, J. Stumpp, M. Reichert, P. Santangelo, and U. Ebner-Priemer. Identification of relevant sensor sources for context-aware ESM apps in ambulatory assessment. 2015. [to appear].
- [2] J. Brooke. Sus-a quick and dirty usability scale. *Usability evaluation in industry*, 189(194):4–7, 1996.
- [3] J. Fahrenberg, M. Myrtek, K. Pawlik, and M. Perrez. Ambulatory assessment—monitoring behavior in daily life settings: A behavioral-scientific challenge for psychology. *European Journal of Psychological Assessment*, 23(4), 2007.
- [4] C. D. Fisher and M. L. To. Using experience sampling methodology in organizational behavior. *Journal of Organizational Behavior*, 33(7):865–877, 2012.
- [5] A. Grunerbl, A. Muaremi, V. Osmani, G. Bahle, S. Ohler, G. Troster, O. Mayora, C. Haring, and P. Lukowicz. Smartphone-based recognition of states and state changes in bipolar disorder patients. *IEEE Journal of Biomedical and Health Informatics*, 19(1):140–148, 2015.
- [6] P. A. Hancock. A dynamic model of stress and sustained attention. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 31(5):519–537, 1989.
- [7] S. G. Hart and L. E. Staveland. Development of NASA-TLX (task load index): Results of empirical and theoretical research. *Advances in psychology*, 52:139–183, 1988.
- [8] B. Laugwitz, T. Held, and M. Schrepp. Construction and evaluation of a user experience questionnaire. *HCI and Usability for Education and Work*, 2008. p. 63-76.
- [9] G. Miller. The smartphone psychology manifesto. *Perspectives on Psychological Science*, 7(3), 2012.
- [10] J. Sauro and J. R. Lewis. Quantifying the user experience : practical statistics for user research, 2012.
- [11] M. Schrepp, S. Olschner, and U. Schubert. User experience questionnaire benchmark praxiserfahrungen zum einsatz im business-umfeld. *Usability Professionals*, 13, 2013.
- [12] C. N. Scollon, C.-K. Prieto, and E. Diener. Experience sampling: promises and pitfalls, strength and weaknesses. In *Assessing well-being*. Springer, 2009. p. 157–180.