Identifying User eXperiencing factors along the development process: a case study

Marco Winckler
ICS-IRIT
Université Paul Sabatier
winckler@irit.fr

Cédric Bach
ICS-IRIT
Université Paul Sabatier
cedric.bach@irit.fr

Regina Bernhaupt
ICS-IRIT, Univ. Paul Sabatier
RUWIDO
Regina.Bernhaupt@ruwido.com

ABSTRACT

Currently there are many evaluation methods that can be used to assess the user interface at different phases of the development process. However, the comparison of results obtained from methods employed in early phases (e.g. requirement engineering) and late phases (e.g. user testing) of the development process it is not straightforward. This paper reports how we have treated this problem during the development process of a mobile application called Ubiloop aimed at supporting incident reporting in cities. For that purpose we have employed semi-directive requirement interviews, model-based task analysis, survey of existing systems and user testing with high fidelity prototypes. This paper describes how we have articulated the results obtained from these different methods. Our aim is to discuss how the triangulation of methods might provide insights about the identification of UX factors.

Author Keywords

Incident reporting systems, UX factors, development process

ACM Classification Keywords

H.5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

General Terms

Human Factors; Design; Measurement.

1 INTRODUCTION

Incident reporting is a very well-known technique in application domains such as air traffic management and health, where specialized users are trained to provide detailed information about problems. More recently, this kind of technique has been used for crisis management such as the hurricane Katrina [1]. Such self-applications are aimed to be accessible by the general public with a minimum or no training. In the context of the project Ubiloop, we are investigating the use of mobile technology for allowing citizens to report urban incidents in their neighborhood that might affect the quality of their environment. We consider urban incidents as any (micro)events, perceived by a citizen, that might affect the quality of his urban environment (e.g. hornet nest, potholes, broken bench, tags,…). By reporting incidents, citizens can improve the quality of life by influencing the quality of their environment. Figure 1 illustrates the overall scenario of our case study.

Figure 1. Overview of incident reporting with Ubiloop: users report incidents like potholes, tagging, or broken street lamps to the local government using a mobile phone application.

Despite the fact that incident reporting systems using mobile technology are becoming more common, little is known about its actual use by the general population and which factors affect the user experience when using such system. In order to investigate which user experience factors must be taken into account when designing the interface of mobile application for incident reporting, we have employed several evaluations methods (including semi-directive requirement interviews, model-based task analysis, survey of existing systems and user testing with high fidelity prototypes) along the development process of the application Ubiloop (developed in the context of the eponym project). Hereafter we report how, using several evaluation methods, it was possible to:

- Identify which (and in what extension) UX factors affect mobile incident reporting systems;
- Associate UX factors and artifacts that are aimed to support the design and implementation of systems;
- Determine how users value the incident reporting systems (in terms of UX factors) in both early and late phases of the development process.

The first two sections of this papers provide an overview of the development process (section 2) and the methods employed (section 3) in the Ubiloop project. Then, at section 4 we describe how we have articulated the results in order to provide a bigger picture of UX factors and artifacts used during the development process. Finally we discuss the results and lessons learned.

2 OVERVIEW OF THE PROCESS

We have followed a user centered design approach. Our first goal was to identify how user experience factors are important to the users when they are performing tasks such as reporting, monitoring and sharing with other citizen's information about urban incidents. We firstly address the following dimensions: perceived quality of service, awareness of perceived user involvement with reported incidents, perceived effects of mobile technology for reporting incidents, trust, privacy, perceived usefulness, usability and satisfaction with incident reporting systems in urban contexts. These dimensions are articulated around four main research questions:

- How citizens perceive and describe urban incidents as part of their quality of life?
- How does the choice of communication to digitally report incidents in a mobile context influence the overall user experience? If so, what dimensions of user experience are important for such an incident reporting application?
- How does social awareness affect the user experience when interacting with incident reporting systems?
- What contextual factors are important for incident reporting and which interaction techniques better assist user in reporting incidents?

These questions were investigated along the development process by the means of different evaluation methods as shown by Table 1.

Table 1. Methods employed during the development process of the application Ubiloop.

Design phase	Methods employed
Requirement analysis	Survey of existing applications Semi-directive requirement interviews Model-based tasks analysis
Design	Prototyping
Evaluation	User testing

Figure 2 shows the articulation between methods and artifacts produced. Notice that the dashed arrows indicate the relationships ensuring cross-consistency between artifacts and results obtained from the methods employed.

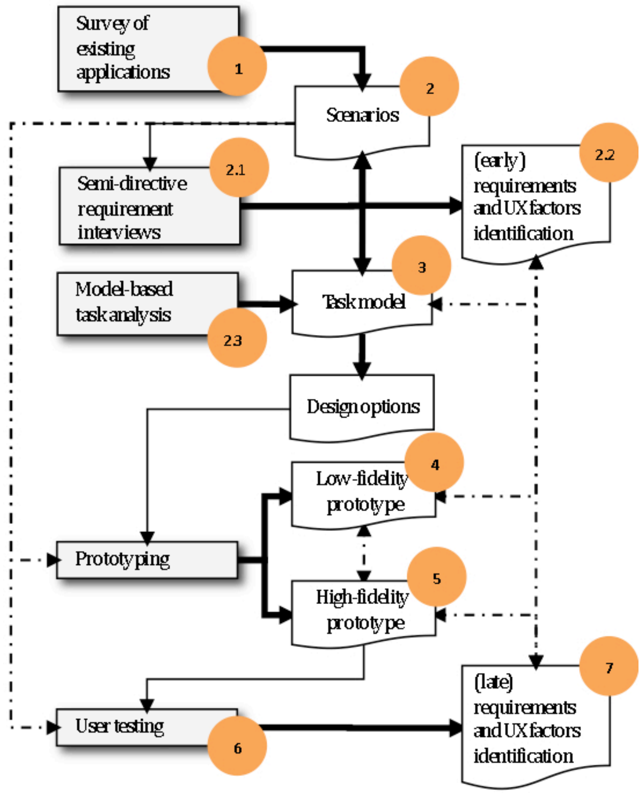

Figure 2 Articulation between artifacts and methods employed. Thick lines indicated artifacts produced; thinner lines indicate input for the method; dashed lines are used to show compatibility checking between artifacts.

In more general terms, this design process started by (1) benchmarking existing applications in order to provide a coverage of the application domain. From this step we have extracted (2) generic and representative scenarios that were used to organize an (2.1) interview with (18 potential) future end-users of the Ubiloop application. These requirement interviews allowed us to identify new scenarios (some of them not covered by existing applications), expectations (that we name here early requirements) and (2.2) UX factors that are associated to the scenarios. By (2.3) analyzing a set of 120 scenarios it was possible to identify a task pattern that was then specified by using a task-model notation. This (3) task model was used to check the coherence of the design with respect to the previously identified scenarios. Then, design options supported by the task model were (4) (5) prototyped and subsequently tested with end users. During (6) user testing, we have assessed (7) UX factors that were then compared with those collected earlier during the (2.2) interviews.

3 METHODS EMPLOYED AND MAIN FINDINGS

In this section presents the methods and key findings.

3.1 Survey of Existing Systems

In order to analyze the actual support provided by existing applications, we conducted an analysis of existing services for incident reporting in urban contexts. This study focused on the front office (i.e. reporter tools). Applications for incident reporting were first identified from the set of tools ranked by Web search engines (i.e. google.com). Then, only those that were available for remote testing were selected for further analysis.

Fifteen applications were selected covering international reporting services. What we found to be specific for the area of incident reporting is the broad diversity of features for reporting urban incidents (more than 340). Nonetheless, these incident reports seem to share similar characteristics which can be used for helping users to locate on the user interface the service that better suits to the type of incident s/he wants to report in a given context of use. Despite the fact that these applications address the same problem of reporting incidents in urban context using mobile technology, none of them was implemented following the same scenario; which might be explained by cultural difference that affect the user experience with this kind of applications. For example, in some countries the identity of the citizen reporting the incident is always mandatory whilst in other countries it was mainly optional or only requested in specific types of incidents (that could be perceived as denunciation).

From the analysis of existing systems we have extracted a set of generic and representative scenarios that should be supported by our application. We could not find in the literature any work describing UX factors addressing this specific application domain.

3.2 Semi-directive requirement interviews

In order to understand users expectations and requirements for the future system, two series of semi-directed interviews were conducted. The first one, called *general interview*, focused on how users perceive their environment and how they formulate general requirements for reporting incidents using a smartphone. The second one, called *scenario-based interview* was designed to investigate how users react to different situations that would be subject of an incident report. Each series of interviews involved nine participants.

During the *general interview*, participants were prompted to report about: how they perceive places and their environment; negative experiences in terms of environmental quality; personal involvement with problems; preferred system design; and dimensions they think important.

In the *scenario-based interview*, participants were introduced to 7 scenarios (one at once, in random order) and then asked to explain how they would envisage reporting incidents using a smartphone. The scenarios included to report a *broken street lamp*, a *pothole*, a *missing road sign*, a *bulky waste*, a *hornet nest*, a *tag/graffiti*, and a *broken bench in a park*. These incidents were selected from the set of scenarios supported by existing applications. Moreover, each scenario was designed to highlight a specific point, for example: a *broken lamp* points out an incident that is difficult to illustrate with a picture, whilst a *hornet nest* focus on the perceived danger. Every interview included a short questionnaire on demographics and technology usage. All sessions were recorded and then transcribed by a French native speaker. The transcriptions were analyzed accordingly to the grounded theory approach [3][6]. A corpus of 92 240 words was analyzed resulting in 11 classes/codes with 1125 segments of text. The coding was supported by the MaxQDA 10 software [8].

The interviews provided two key pieces of information: i) scenarios for reporting incidents, which can be associated to a task that must be supported by the system; and, ii) qualitative attributes that could be interpreted as UX factors associated to the given scenario. For an example, let assume the following segment given by participant P2: *"…Besides going to report your* [own] *idea, you could ask if there are other ideas* [proposed by other]... [that are] *close to your home..."* From this passage, the participant clearly states a UX factor (*stimulation* as described by Hassenzahl [4]) that could influence him to perform the task (*report* [an incident]). These two requirements interviews provided evidence for identifying the following UX dimensions: *visual & aesthetic experience, emotions* (related to negative experience of the incident and positive experience to report it – joy / pride), *stimulation*, *identification (through their personality, their own smartphone, their sensibility to specific incidents)*, *meaning and value*, and *social relatedness/co-experience*.

3.3 Model-Based Task Analysis

From the analysis of existing applications and interviews we have identified 120 possible scenarios that could be generalized as a user task pattern consisting of: *(1) to detect the incident, (2) to submit an incident report* and *(3) to follow up on an incident report*. This pattern was modeled using the task notation HAMSTERS [6] which feature a hierarchical graph decomposing complex tasks into more simple ones as shown by Figure 4. Tasks are depicted accordingly to the actors involved in the task execution (i.e. user, system or both). It also integrates operators for describing dependencies between tasks (i.e. order of execution). As this task model does not impose any particular design for the system it can accommodate all the scenarios identified during the analysis of existing applications. By modeling user tasks it was possible to identify aspects such as optional/mandatory tasks associated to incident reporting, inner dependencies between tasks, as well as pre- and post-conditions associated to tasks execution.

3.4 Prototyping

In previous work [2] we have found that information related to incidents includes: **what** the incident is about, **when** it occurs, **where** it is located, **who** identifies the incident and the expected **outcomes** leading to its solution. These dimensions include optional and mandatory elements that characterize incidents. For example, the dimension **what** can include a combination of either a textual description, a picture of the incident, or just an indication of the incident category. Based on these early findings and the generic task model described above we developed a low-fidelity and then a high-fidelity prototype (see Figure 3). The prototype takes full benefits of currently embedded technology available in smartphones such as video camera and global positioning systems (GPS). GPS makes the user's task of locating incidents easier and photos attached to the description of incidents provide contextual information and in some situation might be used as evidence of its occurrence.

a) b) c)

Figure 3 Ubiloop protoyope featuring: a) main menu page; b) textual description of incident; c) location on an interactive map.

The user interface of the Ubiloop prototype supports all the user tasks previously identified. The prototype was also designed to support the early requirements expressed by users. Moreover, the prototype was designed to create a positive user experience that could be also inferred from the results of the semi-directive requirement interview. For example, to enhance the UX factor experience we deploy the prototype in *a smartphone* (whose technology is perceived as a stimulation for using the application), we include *categories of incident* (as users said they are more likely to report an incident if they could see example of categories on incidents) and allow users *to see reported incidents in the neighborhood* (as suggested by the participant 2, see section 3.2).

3.5 User testing

A user testing with high-fidelity prototype was designed to explore how users report urban incidents with Ubiloop. The study was held at the campus of the University of Toulouse during the summer 2012. Thinking aloud protocol was used during the experiment. Users were asked to wear glasses embedding a video recording system, so that it was possible to determine where they were looking at whilst using the prototype. The recording apparatus also included a logging system and a screen recorder embedded into the smartphone.

Users were trained during 5 minutes on how to report a simple incident (i.e. a *Broken street lamp*) with a smartphone embedding Ubiloop. Participants were then asked to follow a predefined route in the campus and any report incidents found in the way. The route was populated with tags prompting users to report fake incidents that refer to the scenarios presented in section 3.2. In addition to these predefines tags, users were free to report any other incident he could see in the campus (and the route had many real incidents such as *potholes, tags, public light open during day*…). In addition to these tasks users were asked to fill in a demographic questionnaire, an AttrakDiff questionnaire [5] and a debriefing interview.

Nineteen participants, ranging from 21 to 52 years old, took part in the experiment. All participants successfuly complete the tasks. The analysis of data concerning UX factors took into account the answers provided by the AttrakDiff questionnaire, the users tasks and the comments provided by users whilst performing the tasks. Again user's comments were transcribed and analyzed accordingly to the grounded theory approach. At this time the segments were coded accordingly to the actual tasks performed by the users during the experiment.

One of the findings is that all UX factors previously identified during the semi-directed requirement interviews (see section 3.2), were reported again during the user testing. Nonetheless, due to space reasons, we illustrate the description of findings to two factors, *stimulation* and *identification to incident*, that we have found out to be key UX factors to engage the process of reporting (when user decides to report the incident s/he identified in the environment).:

- *Stimulation* was evaluated during the user testing through a question of the post-test interview: "Did you discover some incidents on the University campus that you could report with the prototype?" This UX factor can was also detected during thinking aloud technique and the Attrakdiff questionnaire.
- *Identification to incident* was evaluated with another question of the post-test interview: "Are the incidents you declared during the experiment candidates to be really reported by you to the Ubiloop service"?

Furthermore, the evaluation of Identification to incident reveals that a strong proportion of UT participants declare to be ready to report some of the mandatory incidents (90 % for Broken bench and Hornet nest; 75% for the Broken street lamp; and 45% for the Heap of rubble). And individuals are mainly ready to declare the incidents they spontaneously discover during the experiment (according that the declaration is easy to perform and useful). In other words, the applications seem to be able to increase both *Stimulation* and *Identification to incident*.

4 TRIANGULATION OF METHODS

To answer the research questions on what user experience (UX) dimension should be taken into account when designing incident reporting systems for urban contexts; we have triangulated the results of the three methods used in this work, as follows:

- During semi-directive requirement interviews users expressed requirements and expectation for reporting incidents by the means of personal stories that were interpreted as possible scenarios of use. These scenarios were then used to revise our original task model for incident reporting systems.
- By using a model-based task analysis, it was possible to remove ambiguities present in the discourse of participants and then to formalize users' requirements. Moreover, model-based task analysis provided an accurate description of user tasks. This step is extremely important for future development of incident reporting systems. As described in [7], tasks models do not only improve the understanding of user tasks but they also can be used to assess if an incident reporting system was effectively implemented to support the specified set of user tasks.
- In order to make sure that tasks identified in the semi-directive requirement interviews and model-based tasks analysis are representative we compare them with a survey of existing systems. The results confirm that our analysis is exhaustive because our task model covers all tasks supported by surveyed systems and these systems do not implement any task that is not described in the task model.
- The analysis of transcripts of semi-directive requirements interviews also supported the identification of UX dimensions associated to user scenarios. By combining UX dimensions and user scenarios it was possible to extrapolate the results in a single task model as shown by Figure 4 where user tasks are decorated with UX dimension (e.g. [ID] for identification] so that the above could be read as follows: *"I am passing by at this park every Sunday and this bench has not been repaired for weeks [ID]. It is time now to report that, so it will get fixed. It is not really a problem or unsafe, but the bench is simply not usable in the current state [MV]. [: detect/recognize the incident:]. It seems important now to make sure that the appropriate person is informed about that bench [CX], I think I should use the application to report the incident, because I want to be a good citizen [ID]. I think it is a good idea to send them a photo so they can see that the bench is really broken and that the wood has to be replaced. And when they see the photo they see that it is really there and so they will not need my contact information to have a proof that the broken bench really exists. [MV] [:describe the incident:]"*. This example shows how user tasks are interrelated to the UX dimensions.
- The prototypes were building accordingly to the task models. Once implemented, the prototype was cross-checked in order to make sure that it can effectively support the scenarios early identified. Thus, every presentation unit (ex. screens and widgets) can be easily associated with an element of the task model. By extrapolation with the results from requirement interviews we could extrapolate a tuple consisting of *user interface elements* + *user tasks* + *UX factors*.

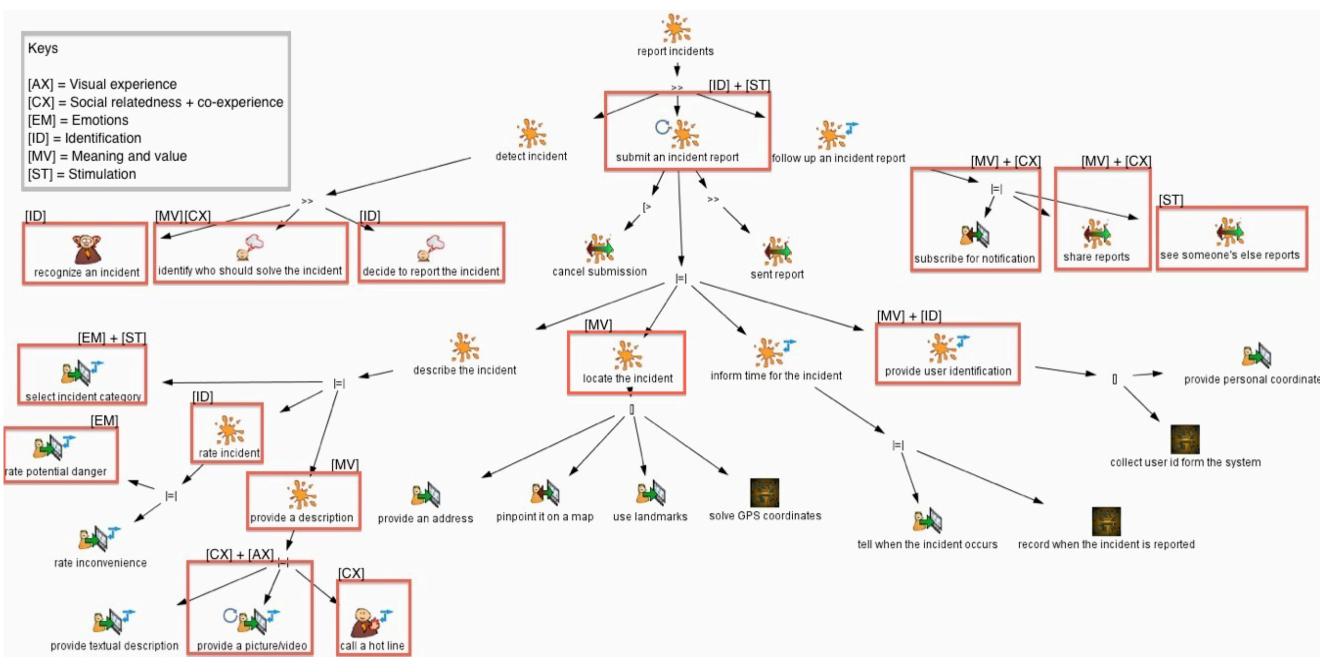

Figure 4 Generic task and most important UX dimensions for each sub-task.

- During user testing is was possible to identify UX factors during the execution of the tasks with the prototype. It is interesting to notice that the scenarios supported by the prototype were the same used during interviews so it was possible to correlate the results found in early and later phases of the development process. Thus, we have found that the UX factor *stimulation* reported during interviews to the tasks *to find incidents* occurred again when the users use the prototype to complete the same task. This confirms the value of early identification of UX factors with requirement interviews. Moreover, when counting the number of segments of user testing reporting the UX factor *stimulation*, we have found that this factor is more frequent and even distributed along tasks. We also have compared the categories of incident reported by users during the thinking aloud and during the debriefing; we have found that the distributions of incidents across categories are more important in the requirement phase (72 citations/42 categories) than in user testing (80 cites/19 categories). Indeed, during the requirement interview participants had difficulties to identify/remember urban incidents whilst during user testing participants had more ease to identify incidents along the route of the experiment.

- Before the participants of the requirements interviews had strong difficulties to identify, remember or imagine urban incidents. It's not the case (or less the case) when users can interact with the mobile application.
- Others examples come from the responses to the post-test interview question about the Stimulation factor.

  "I never thought to report this kind of incident [a public garbage with a broken top] before [to use the application], but that true this is would be quickly a serious problem of squalor."

  "That's funny because this application gives me the opportunity to discover my own environment with a new eye."

5 DISCUSSION AND LESSONS LEARNED

Unfortunately, we don't have room for providing a comprehensive description of the results collected by the different methods. Nonetheless, the results given in this paper illustrate that UX factors can be detected both in early and later phases of the development process. Moreover, in some extend, such results can be correlated.

One of the challenges was to determine the importance of UX factors when they are collected in different phases of the development process. In the present work we have been using a simple counting method (number of segments) and distribution of UX factors across users' tasks. Using this simple method we found some differences that require further analyses. Nonetheless, it prompts by a case where it would be interesting to have quantitative metrics of UX for comparing them.

It is important to associate the identifying UX factors with the artifacts used to the design. In our study, we have found that scenarios and task models works as a *lingua franca* for mapping user requirements and UX factors. However, it is worthy to notice that this might be specific to a certain types of interactive systems that can be successfully described by tasks models. We can just wonder if this approach could work in application domain such as game where user activity is harder to represent by the means of task models. Further work is required to determine if other design artifacts and evaluation methods can also be used to provide such as articulation.

We have deliberated performing the user testing with high-fidelity prototypes. We have found in the requirement interviews that the use of the device smartphone is *per se* a stimulating element. For the purpose of the project, it was more important to test the high-fidelity prototype in a situation of mobility than a paper-based mockup. However it would be interesting to assess the impact of mockups on the identification of UX factors.

Acknowledgement

This work is part of the Ubiloop project partly funded by the European Union. Europe is moving in France Midi-Pyrenees with the European Regional Development Fund (ERDF). Genigraph and e-Citiz are partner of this work.

REFERENCES

1. Moynihan, D. P. (2007). From Forest Fires to Hurricane Katrina: Case Studies of Incident Command Systems. IBM Center for the Business of Government.
2. Bach, C., Bernhaupt, R., Winckler, M. Mobile Incident Reporting in Urban Contexts: Towards the Identification of Emerging User Interface Patterns. In 5th IFIP's WG 13.2 Workshop PUX. Lisbon, Portugal, September 5th 2011.
3. Glaser, B.G., Strauss, A. L. (1967) The discovery of Grounded Theory: Strategies for qualitative research, Adline: Chicago.
4. Hassenzahl, M. The thing and I: understanding the relationship between user and product. In Funology: From Usability to Enjoyment, M. Blythe, C. Overbeeke, A. F. Monk, & P. C. Wright Eds Dordrecht: Kluwer, 2003, pp. 31-42.
5. Hassenzahl, M. (2002). The effect of perceived hedonic quality on product appealingness. International Journal of Human-Computer Interaction, 13, 479-497
6. Lazar, J., Feng, J. H., Hochheiser, H. Research methods in Human-Computer interaction, John Wiley & Sons: UK, 2010.
7. Martinie, C., Palanque, P., Winckler, M. Structuring and Composition Mechanisms to Address Scalability Issues in Task Models. In Proc. of INTERACT, (3) 2011, pp. 589-609. Springer LNCS.
8. MaxQDA [online] software available: www.maxqda.com
9. Matyas, S., Kiefer, P., Schlieder, C., Kleyer, S. (2011). Wisdom about the Crowd: Assuring Geospatial Data Quality Collected in Location-Based Games. Entertainment Computing – ICEC 2011 Lecture Notes in Computer Science, 2011, Vol. 6972/2011, 331-336.
10. Goodchild, M. F. (2007). Citizens as sensors: the world of volunteered geography. GeoJournal, 69(4), 211–221. doi:10.1007/s10708-007-9111-y.