Capturing Mental Models to Meet Users Expectations

Cristina Olaverri-Monreal*
Institute of Ergonomics
Dept. of Mechanical Engineering,
Technische Universität München
Boltzmannstraße 15,
85747 Garching, Germany
Email: olaverri@lfe.mw.tum.de

Joel Gonçalves
Institute of Ergonomics
Dept. of Mechanical Engineering,
Technische Universität München
Boltzmannstraße 15,
85747 Garching, Germany
Email: goncalves@lfe.mw.tum.de

Abstract— **User centered design implies knowledge of the future user's needs within the design phase of the product. A correspondence between both conceptual and mental models ensures a smooth operation in the expected manner that entails mastering all the functionality ensuring user satisfaction. Therefore, consideration of the user's mental model is crucial in designing a system consistent with user expectations. In this paper we present a tool to capture mental models by means of an In Vehicle Information System (IVIS). Through a collaborative approach the application enables IVIS users to design their own layered architecture while the data is simultaneously stored for further evaluation.**

Keyword-components; mental models; conceptual models; collaborative systems

I. INTRODUCTION

Computer systems that contain a large number of functionality in form of classes, functions and operations involve a degree of complexity that can prevent a smooth interaction [1]. In order to operate these systems effectively it is important that the users understand their purpose and capabilities. According to [2] mental models can be defined as the knowledge of how a system and its components work, their relationship, what are the internal processes and how they affect the components.

User centered design implies knowledge of the future user's needs within the design phase of the product. As a consequence, the intended context of use will be reflected in the representation of how a system should work. A correspondence between the product's conceptual and mental model ensures a smooth operation in the expected manner that entails mastering all the functionality ensuring user satisfaction. Therefore, it is crucial to take into account user's mental models to effectively design systems consistent with the users' expectations. This is particularly important for systems developed to be operated in a vehicular environment, such as In Vehicle Information Systems (IVIS) that can include different subsystems for entertainment and information. An optimal menu layout organization in IVIS user interfaces is essential to ensure a user friendly navigation that minimizes distraction from the road. Consequently, we focus on a software system that facilitates collaboration among different users and contribute therefore to extending the knowledge and understanding of interaction with technical systems presenting a tool to capture mental models by means of an IVIS. Our tool enables the representation of mental models by creating hierarchical structures from content generated by IVIS users. To validate the users' mental models, the designed prototypes are transferred to a further driving game framework where the their structures can be evaluated through feedback gathered from other platform users.

Our application enables the user creating his own layered architecture design being this way involved in a User-Centered Design (UCD) and development process from the beginning and having his needs, and difficulties considered. Following the requirements defined from the authors in [3] our approach uses generic methods to discover solutions to problems relying on other users suggestions as recommended in [4].

II. MENTAL MODELS

The idea of capturing shared mental models has been already introduced in previous work. For example in several studies related to team work, methods have been proposed to obtain and analyze shared mental models to find associations with performance within groups [5, 6]. In other works mental models related to information prioritization for different modules in Driver Information Systems have also been investigated [7].

The authors in [8] stated that shared mental models include belief and knowledge structures: knowledge structures describe what, how and why we know something and also what we think about a particular matter [9]. According to [10] there are three steps that need to be performed in order to measure mental models:

- obtaining the concepts that are inherent to the model;
- selecting the most appropriate representation technique to reproduce the content and structure of the model;

*Corresponding author, Member IEEE

 DOI: 10.1109/CISTI.2014.6877006

- analyzing the obtained data to investigate common cues in the mental models content.

To obtain mental models several techniques can be applied that include among others questionnaires, interviews, observations and card sorting [11]. This last technique was applied in the exploratory study in [12] where the authors investigated spatial mental models for a menu structure of a diabetes living assistant. They then related the models to the performance in the usage of a device and analyzed the results by several statistical methods including one-way ANOVA and multivariate analysis of variance.

Relying on the card sorting approach we developed a tool able to collect the appropriate data to assess the mental models captured, investigating similarities in the information structures performing the steps defined above, namely obtaining the model concepts, representing them and analyzing the results.

Our tool enables the representation of mental models by creating hierarchical structures from content generated by users, a common technique used to represent mental models [13]. This structured content enables mapping of different data through data mining techniques that allow for the investigation of the relationship between different models and the dependent and independent variables. A further quantitative analysis of the acquired data enables us to find similarities between different groups through cluster analysis or multi-dimensional scaling techniques.

To validate the users' mental models, the designed prototype is transferred to a further driving game platform. After evaluation the results can be of further use for the designer community to gather feedback from other users and finally incorporate this information into the final product design.

III. Information Architecture Automatic Tool (IAAT)

To enable the configuration of the preferred information architecture design of an IVIS, including the location of elements of lower hierarchy, we developed a Web-based tool so that it could be used in a global context and reach thus a large group of participants at a lower cost. Our Information Architecture Automatic Tool (IAAT) relies on the card sorting approach [14, 15] and allows the user to reflect upon how to perceive information and how to perform decisions in a vehicular context. It additionally reflects the characteristics of complex systems, such as heterogeneity of users' preferences by means of the system structure organization.

The IAAT tool was defined in a flexible way in order to enable different structures of configuration. Relying on Markup Language technologies such as the Extensible Markup Language (XML) that very efficiently represents menu hierarchies, we defined options and sub options hierarchically by tagging the document elements [16, 17]. The tool consisted of four architecture modules that we describe below and are illustrated in Fig. 1.

A. *IVIS Designer*

This application module enables the user to design his preferred IVIS. After having created the menus, an XML file that we named "Conceptual Model.xml" containing the tree structure was stored. This file was then used to further extend the information related to the IVIS design to all other modules. The "Model Validation.xsd" file contained the rules for the structure and content description of the XML file. An additional Extensible Stylesheet Language Transformation ("Local Transform.xslt") file included the rules to transform the original XML document into a different markup language document such as the "Preview.xhtml". This file generated the preview view of the selected structure in a web browser. This chosen format has the advantage to be interpreted by a multitude of applications that include also mobile web browsers, game engines or common User Interfaces (UI) development frameworks. We used the same markup language technology to provide our IVIS Designer application with a *What You See Is What You Get (WYSIWYG)* interface.

After the menu configuration, the *IVIS Designer* allowed exporting the "Conceptual Model.xml" file to several target applications such as the *IAAT Community Web Application* or further applications from other platforms such as the serious games based application to assess the ergonomics of IVIS IC-DEEP [18, 19] that provides a driving simulator environment to test the design in a more realistic context.

B. *IAAT Community Web Application*

This architecture component facilitated the publication of the conceptual models and allowed consulting the feedback other users provided. Similar to the IVIS Designer module, the IAAT Community Web application also uses XML based technologies in its workflow. The "Community.xslt" transforms the "Conceptual Model.xml" into the webpage "Community View.xhtml".

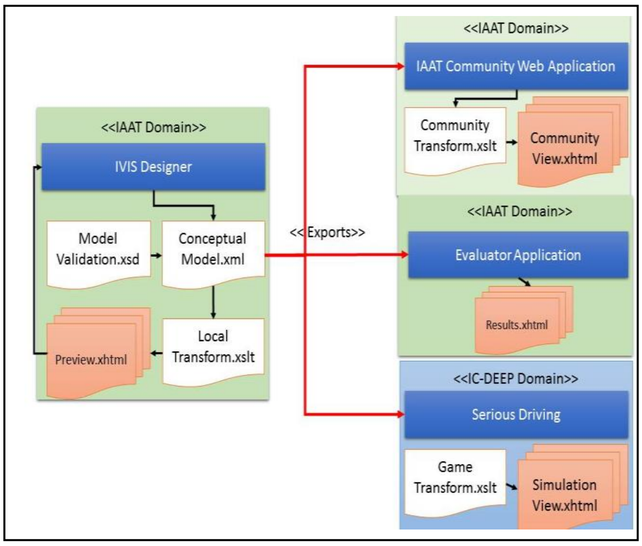

Figure 1. Architecture modules of the Information Architecture Information Automatic Tool

This webpage not only provides a visualization of the conceptual model but also of the UI components so other users can provide feedback in terms of rating, comments or even upload their own "Conceptual Model.xml" with the changes to the current version.

In terms of functionality, every community user must register first, whether for designing IVIS or for providing feedback to others' IVIS. After a new user registered in the website, providing personal information related to his nationality, native language, age, gender, etc. he was able to download the IVIS Designer tool.

All data, from both designers and feedback providers, is saved in a database. The database model is depicted in Fig. 2. The database was conceived to support multiple languages, multiple iterations in the design cycle, and store feedback from the community.

C. Evaluator Application

This module processed the "Conceptual Model.xml" file, creating the "Results.xhtml" as output for the results presentation. The file can then be accessed for further data processing and analysis.

D. Physical Architecture

Our approach relies on a client-server system model as depicted in Fig. 3. By default, some resources were already available for all projects.

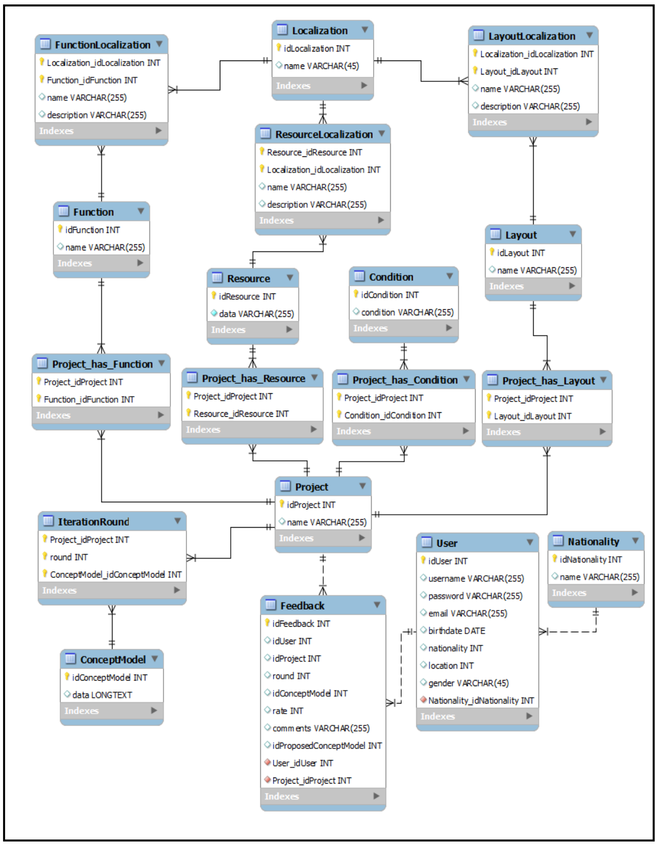

Figure 2. Database architecture

Next section explains in detail the functioning of the tool focusing on the user interface that enables designing the information architecture.

IV. IVIS DESIGNER DETAILS

A. Information Input

As explained in the previous section this architecture module enabled the user designing his preferred IVIS. To this end, a selection of IVIS functions and icons were displayed to the user. He could then create and label group categories to associate the functions in the most appropriate group. Let $\Phi = \{\varphi_1,\ldots,\varphi_f\}$ represent the set of all f features or functions the users can select, and $P=\{\rho_1,\ldots,\rho_r\}$ the set of all resources available (images, audio files, etc…). Additionally, we define $\Lambda = \{\lambda_1,\ldots,\lambda_l\}$ as the set of all admitted layouts (grid, vertical, horizontal, etc…), and $K = \{\kappa_1,\ldots,\kappa_k\}$ as the set of conditions the model must satisfy in order to make possible the design process. As a result, our design process input is described in the following manner: $I = (\Phi, P, \Lambda, K)$.

B. System State

The iterative nature of the design process is captured by the list H, where each element is defined as $h_i = (N,E,F,FR,NL)$. The current state is composed by the N set of nodes and E set of edges expressing both the special structure of the mental model. Specifically, each node $v = (name, \phi)$ has a name associated, and a set of ϕ features or IVIS functions, being $\phi \subseteq \Phi$. If in the resulting information

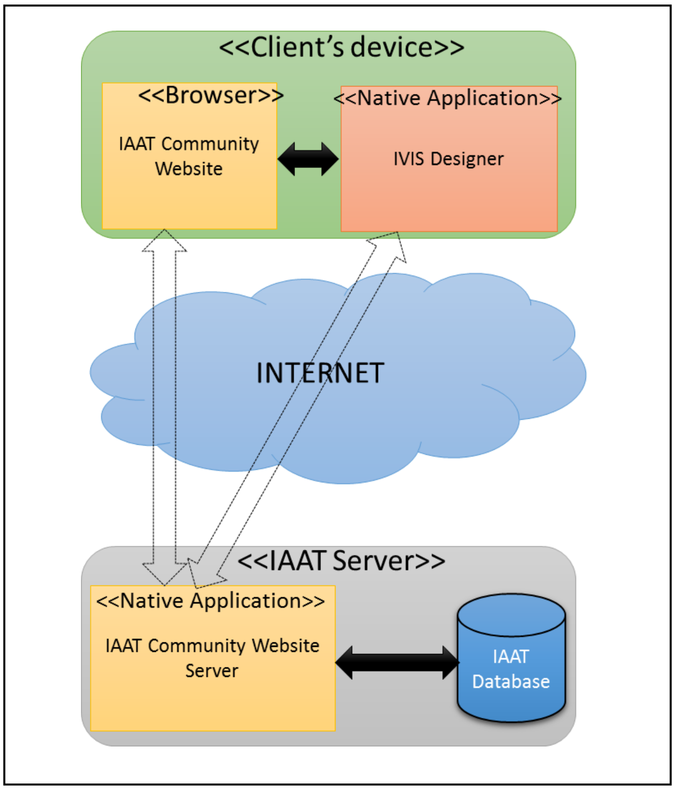

Figure 3. IAAT physical architecture

structure the user has only selected a feature, the selected node represents a leaf or terminal node in the tree structure. On the contrary, if several features have been selected then the nodes will be menus including children nodes. F is a set of features not yet used in N that serves mainly for verification purposes in case of a slow collaborative work. Finally, the two last sets FR and NL define the aesthetic aspects of the mental model consisting the FR set of the (φ_m, ρ_n) pair and representing the assignment of resources to features (e.g. radio icon to represent a FM feature). The NL set (v_x, λ_y) describes the layout of each node.

C. System Interactions

We define several different ways of collaboration for users that design the information structure and users that give their feedback in the same project.

Users can provide their feedback regarding a certain project through ratings in the following form: (s_t, rate, comment) being the s_t state the number $_t$.

Similar to the system's state definition in the previous subsection, we define the interaction for submitting a new state version with h = (N,E,F,FR,NL). The only difference is that the user's role in this case is the one of a "super user" for the reason that when he/she submits a version, the current iteration ends and a new begins.

We also provide an alternative way to directly address specific issues of designed structures by sending short messages with implemented changes (s'_t, comment) that the user can be simply accept and incorporate to his structure design.

D. System Dynamics

To complete the formal definition of our system, we present the interaction workflow as an algorithm as follows.

```
1   while(verify(K,H,R,C) == false) {
2       turn := sizeof(H);
3       s := H[turn];
4       s' := handleCommunity(s,C);
5       s' := handleIndividual(s', i);
6       H += s';
7       wait(K);
8       R := getRates(turn+1);
9       C := getCommunityRecommendations(turn+1);
10  }
```

In line 1, the verify (K,H,R,C) function proofs if the design process can be stopped comparing the constraints defined in K (e.g. number of interactions, threshold of positive/negative ratings, etc.) with the history of the states H, the list of ratings R, and the community recommendations C.

Lines 4 and 5 handle the interaction between the community and the individual user. Community contributions are associated to the state s. Therefore, users can only submit their individual contributions after having considered the community feedback. Line 6 adds a new state to the list of previous states. The statement in line 7 represents the waiting time until one of these cases applies.

- A subset of K is fulfilled, so the process resumes (e.g. at least 30 people voted).
- The user designing the information structure manually interrupts the waiting process and resumes his turn.

Finally, lines 8 and 9 respectively execute the acquisition ratings and recommendation interactions.

V. APPLICATION SCENARIOS

As the IAAT has a contest character pre-defined information structures from different users compete in efficiency with each other. Therefore the target audience is engaged in the project and in providing data to be further studied. Our platform allows the collection of relevant data through the user configuration of IVIS preferred information architecture. Through data mining techniques the data sets are computationally processed to discover patterns so that they can be later transformed into structures for further analysis through methods like cluster analytical approaches. Mental models can then be inferred from structures similarities patterns and be grouped into types regarding their complexity. Our application can be applied in the following scenarios:

A. Use Case Scenario 1

After having the users generated their own IVIS design the feasibility of the approach can be tested taking into account usability, and also the intended driving environment where the application is to be used.

For this purpose we determine the metrics to measure driving performance. As in any driving scenario, the driver needs to keep his eyes on the road and we select as a parameter the time to perform a given task with the IVIS, for example, regulating the temperature in the vehicle. The "Conceptual Model.xml" file contains the number of interaction steps to access a certain function and provided us with relevant information. The file is then exported to the IC-DEEP simulator platform where the designed IVIS is reproduced in a display. Test subjects can then test the menu structure design in a realistic driving environment that also measures driving performance and satisfaction with the system.

B. Use Case Scenario 2

It has been shown in previous research that a detailed study of cultural differences can be used to decide to what extent subsequent measures related to adaptation of a certain product to a specific target culture need to be taken. In the particular case of IVIS, cultural preferences in the interaction with the user interface could lead to a navigational ease decrease through an insufficiently localized human machine interface depending on the culture [20]. Based on this, a further example scenario is a system that considers different cultural backgrounds.

To achieve this behavior, test subjects with different cultural backgrounds design their own IVIS. Using data mining techniques to obtain cultural similarity patterns the ideal system for a certain target culture is then evaluated and recommended.

VI. CONCLUSION AND FUTURE WORK

In this paper we presented a system to capture mental models through the creation of hierarchical structures by users. The framework enables the user creating his own layered architecture design being this way involved in a User-Centered Design (UCD) and development process from the beginning and having his needs, and difficulties considered. The data acquired reflects the mental models of the users and will extend the knowledge and understanding of interaction with technical systems. Future work aims to collect and evaluate data pertinent to this realm of study.

REFERENCES

[1] A. A. Turner, *Mental Models and User-Centered Design*, DTIC Document, 1990.

[2] J. M. Carroll, J. R. Olson, and N. S. Anderson, *Mental models in human-computer interaction: Research issues about what the user of software knows*: National Academies, 1987.

[3] G. Fischer, "Communication requirements for cooperative problem solving systems," *Information Systems,* vol. 15, no. 1, pp. 21-36, 1990.

[4] C. Olaverri-Monreal, and J. Gonçalves, "Collaborative System to Investigate Mental Models: the Information Architecture Automatic Tool (IAAT). Proceedings International Conference on Collaboration Technologies and Systems, Minneapolis, Minnesota, USA., 2014.

[5] L. A. DeChurch, and J. R. Mesmer-Magnus, "Measuring shared team mental models: A meta-analysis," *Group Dynamics: Theory, Research, and Practice,* vol. 14, no. 1, pp. 1, 2010.

[6] F. W. Kellermanns, S. W. Floyd, A. W. Pearson, and B. Spencer, "The contingent effect of constructive confrontation on the relationship between shared mental models and decision quality," *Journal of Organizational Behavior,* vol. 29, no. 1, pp. 119-137, 2008.

[7] C. Olaverri-Monreal, C. Lehsing, N. Trubswetter, C. A. Schepp, and K. Bengler, "In-vehicle displays: Driving information prioritization and visualization." Proceedings IEEE Intelligent Vehicles Symposium, Gold Coast, Australia, pp. 660 – 665., 2013.

[8] J. A. Cannon‐Bowers, and E. Salas, "Reflections on shared cognition." *Journal of Organizational Behavior,* vol. 22, no. 2, pp. 195-202, 2001.

[9] A. P. Banks, and L. J. Millward, "Differentiating Knowledge in Teams: The Effect of Shared Declarative and Procedural Knowledge on Team Performance," *Group Dynamics: Theory, Research, and Practice,* vol. 11, no. 2, pp. 95, 2007.

[10] M. Matteson, "Capturing shared mental models: An approach for bona fide groups," *Journal of Librarianship and Information Science,* pp. 0961000613499362, 2013.

[11] J. Langan-Fox, S. Code, and K. Langfield-Smith, "Team mental models: Techniques, methods, and analytic approaches," *Human Factors: The Journal of the Human Factors and Ergonomics Society,* vol. 42, no. 2, pp. 242-271, 2000.

[12] A. C. Valdez, M. Ziefle, F. Alagöz, and A. Holzinger, "Mental models of menu structures in diabetes assistants," *Computers Helping People with Special Needs*, pp. 584-591: Springer, 2010.

[13] S. Mohammed, L. Ferzandi, and K. Hamilton, "Metaphor no more: A 15-year review of the team mental model construct," *Journal of Management,* vol. 36, no. 4, pp. 876-910, 2010.

[14] J. Nielsen, "Card sorting to discover the users' model of the information space," *Obtained from: http://www. useit. com/papers/sun/cardsort. html*, 1995.

[15] A. Nawaz, "A Comparison of Card-sorting Analysis Methods.", Proceedings 10th Asia Pacific Conference on Computer Human Interaction, p. 583-92, 2012.

[16] C. M. Sperberg-McQueen, C. Huitfeldt, and A. Renear, "Meaning and interpretation of markup," *Markup Languages Theory and Practice,* vol. 2, no. 3, pp. 215-234, 2000.

[17] C. Olaverri-Monreal, K. Bengler, M. Breisinger, and C. Draxler, "Markup Languages and Menu Structure Transformation during the Internationalisation process of Driver Information Systems," *Localisation Focus,* vol. 9, no. 1, pp. 4-12, 2010.

[18] J. Goncalves, R. J. Rossetti, and C. Olaverri-Monreal, "IC-DEEP: A serious games based application to assess the ergonomics of In-Vehicle Information Systems.", Proceedings IEEE, 15th Intelligent Transportation Systems Conference, Anchorage, AK, USA, pp. 1809 – 1814., 2012.

[19] P. R. Alves, J. Gonçalves, R. J. Rossetti, E. C. Oliveira, and C. Olaverri-Monreal, "Forward Collision Warning Systems Using Heads-Up Displays: Testing Usability of Two New Metaphors.", Proceedings IEEE Intelligent Vehicles Symposium Workshops, Gold Coast, Australia. pp. 1 – 6., 2013.

[20] C. Olaverri-Monreal, and K. Bengler, "Impact of cultural diversity on the menu structure design of Driver Information Systems: a cross-cultural study," *IEEE Intelligent Vehicles Symposium,* Baden Baden, Germany, pp. 107-112., 2011.