User Interface Design Guidelines Arrangement in a Recommender System with Frame Ontology

Maxim Bakaev and Tatiana Avdeenko

Department of Economic Informatics, Novosibirsk State Technical University, Russia
maxis81@gmail.com, tavdeenko@mail.ru

Abstract. Design guidelines, which come from the extensive body of knowledge currently formed in HCI and usability engineering domains, remain poorly integrated. Guidelines and design patterns from various sources may contradict or duplicate each other, lack links to origins and justification, as well as contextual associations to concrete problems. The paper describes how the recommender system, developed to support interface design, resolves the issues of data integration and credibility via employing frame-based ontology model and guidelines "efficiency" evaluation algorithm based on fuzzy relations. Also, experimental investigation was carried out with 24 subjects of different age groups to assess the quality of the system work. The results suggests reasonable applicability of the proposed approaches, as the success rate for the website created with the system nearly doubled the one for the control group, and guidelines efficiencies were significantly higher for relevant target user groups.

Keywords: Data integration, frame ontology, metadata, fuzzy relations model.

1 Introduction

As the amount of data worldwide continues to grow exponentially, and the diversity of applications and devices is increasing, ensuring high quality of human-computer interaction (HCI) remains a vital problem. The practical adoption of usability engineering methods continues to be far from universal [1], while positive results of their employment are highly dependent on usability specialists' skills. One possible allocation of layers for interface design-related expert knowledge is the following:

1. Laws – high-level theoretical constructs or statements describing significant aspects of interaction or users.
2. Principles – more or less universal canons concerning design decisions or design process in general. Heuristics and design standards go into this layer as well.
3. Guidelines – pieces of practical advice on how to implement features for a successful interface or reminders about common pitfalls.

A recent strong trend in HCI is promotion of design patterns – approved model solutions for common design problems – as the next layer of knowledge, even more concrete than guidelines (see, e.g. [2, p.360]). However, it is still recognized that

H. Yu et al. (Eds.): DASFAA Workshops 2012, LNCS 7240, pp. 311–322, 2012.

"guidelines remain the most widely accepted form of presenting experience and knowledge" [3, p.160], so we in our paper chose to focus on guidelines and consider design patterns a sufficiently similar notion.

Among the layers of knowledge marked out above, guidelines constitute the most extensive and most used in interface design practice one. Both guidelines and design patterns [2, pp.365-367] should be applied on all stages of HCI engineering process, however certain problems concerned with their practical use are noted [4, pp.82-98]:

1. Average time spent to find and implement one guideline is 15 minutes.
2. Difficulties with correct interpretation of found guidelines by interface designers (in 30% of the cases).
3. Decreased guidelines utility due to obscurity of implied application context.

Indeed, nowadays design guidelines originate from a multitude of sources, so they routinely contradict or duplicate each other, lack theoretical justification as well as indication of appropriate application context, etc. [5, p.52]. Thus we undertook a review of HCI knowledge organization tools and the approaches they use to integrate guidelines or design patterns collected from various sources.

1.1 Design Guidelines Organization in Knowledge-Based Systems (KBS)

Our review of knowledge-based interface design support systems allows to suggest that the majority of them are indeed tools for working with guidelines. A Special Interest Group "Tools for Working with Guidelines" was established in the 1990s, when the works of J. Vanderdonckt and his co-authors set the basis for development of such systems [4] and subsequently proposed concrete implementations [6]. The result was the MetroWeb system, the detailed description of which is provided in [7], together with experimental data showing positive effect of its application in interface design. The guidelines in the system did include relations to model(s) and context, however the latter was understood as a set consisting of development stages, evaluation methods, etc., rather than as design context that would consider target user group, the product features and qualities, and so on [7 p.53]. On the whole, the researchers noted that designers were not engaged with the system and used it unenthusiastically, but even then it led to more user-centered approaches.

Another example is ontology-based system for organizing design patterns, whose core component was named BORE and which was in development since 1997, but currently seemingly disappeared from publications. The ontology implements a metamodel for design patterns, with attributes and formal relations such as "contains", "alternate to", "disjoint with", etc. [5, p.50]. Obviously, the problem of the patterns duplications and contradictions was recognized and attended to, but it remained unclear whether such strict associations are sufficient to adequately represent the domain, and if an expert supporting a working system would be able to manually maintain such relations.

Our further enquiry, supplemented by a detailed review of the tools existing in HCI domain provided in [8], led us to the conclusion that the issue of design guidelines organization is yet to be fully resolved, and this is the main problem we address in our paper. To ensure proper guidelines arrangement, a recommender system in particular must:

1. Address the issue of guidelines integration from multitude of sources, providing means for resolving contradictions and identification of duplicates. This may be done via introduction of data "significance" determination mechanism and guidelines classification, both preferably maintained by the system.
2. Ensure data provenance in terms of both tracing the source of guideline and its justification, i.e. relation to higher-level knowledge, such as laws or principles.
3. Include guideline application context and be able to match it to the context in which recommendation is made. Thus the system must contain metadata on interface design domain that could be both assigned to guidelines and used to describe design context.

1.2 The Recommender System to Support Interface Design

In our project dedicated to the development of recommender system to support interface design, we came to the conclusion that the design context is primarily defined by target users attributes and requirements, mostly non-functional ones. The outcome of the system should be the set of human-readable guidelines relevant to the project context, with the addition of interface wireframe that together should be used by designer to construct the interface prototype (see Fig. 1).

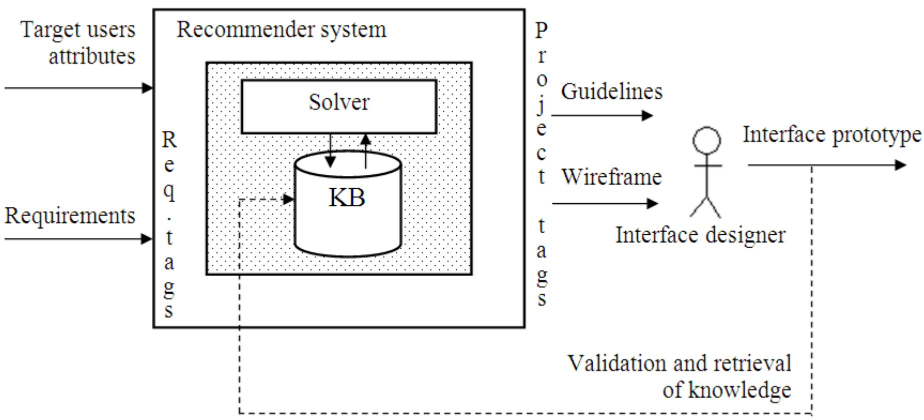

Fig. 1. General structure of the recommender system, input and output information

We propose to establish the recommender system on the knowledge base (KB) built upon ontology which model is described in the *Method* chapter of the paper. Among various possible ontology models, we developed the one based on such data structures as frames, which were first described by M. Minsky in the 1970s, as the ontology needs to combine domain specification and linguistic features. The former permits the classification of guidelines by assigning "tags" (domain concepts), while the latter would allow design context extraction directly from requirements specifications in natural language. In addition, the relationships peculiar to ontologies permit natural data provenance, as guidelines may be linked to both their sources and justifying knowledge. To evaluate the relative "significance" of guidelines, which is essential given their current multitude and poor organization, we also propose the model, which is based on fuzzy relations.

Although detailed description of the recommender system creation and application is beyond the scope of the current paper, which is dedicated to the proposed solutions for guidelines organization in the system, we briefly outline the development process in the chapter *The Recommender System Implementation*. The chapter *Experiment* describes the experimental investigation carried out to assess the feasibility of the proposed approaches to design guidelines arrangement.

2 Method

2.1 Knowledge Representation Model

Numerous researchers have been criticizing T. Gruber's initial definition of ontology as the "explicit specification of a conceptualization" – conceptualization being <D,R>, where D is domain and R are relations on the domain, – and working on its development (among them N. Guarino [9, p.5], one of the founders of ontology application in information science). Currently, the specification of ontology made a transition to a following form [10, p.180]:

$$O = \langle C, R, A, TDL, A_X \rangle, \quad (1)$$

where C is a set of classes corresponding to certain concepts of domain;

$R = \{R_I\} \cup \{R_P\} \cup \{R_A\}$ – a set of binary relations defined on classes C:

R_I – binary transitive relation of inheritance, giving child classes both attributes and relations of the parent class (and the corresponding inverse relation);

R_P – binary transitive relations of inclusion (e.g. "has-part" – "is-part-of");

R_A – a finite set of other possible relations;

$A \subseteq C \times TDL \cup R \times TDL$ – a set of attributes for classes or relations;

$TDL = T \cup D \cup L$, correspond to set of attributes types ($T = \{$integer, string, …$\}$), data domains ($D = \{D_1, \ldots, D_n\}$) or constraints for the attributes values ($L = \{L_1, \ldots, L_m\}$);

A_X – a set of "axioms" that represent certain intrinsic knowledge of the modeled domain. As simple constraints for attributes types, values, etc. may be set in TDL, subsequently axioms should be some logical rules or state-independent information (S_I) of the domain (see [9, p.10] on state-independent and state-dependent information).

If an information system or a KBS must process textual resources, then it has to contain a thesaurus T_H – a structured vocabulary of terms corresponding to the domain, in one or several languages [10, pp.188-189]:

$$T_H = \langle T_R, R_T, A_T, A_{TX} \rangle, \quad (2)$$

where T_R is a set of terms that generally contain a specially defined subset T_{DEF}, with preferred (default) terms for a concept or relation;

$R_T = \{R_{TI}\} \cup \{R_{TS}\} \cup \{R_{TE}\} \cup \{R_{TA}\} \cup R_{TO}$ – a set of binary relations defined for T_H:

R_{TI} – relations connecting a term with more general term (and inverse relations);

R_{TS} – relations connecting terms from T_{DEF} and their less preferred synonyms;

R_{TE} – symmetrical relations of lexical equivalence for terms in different languages;

R_{TA} – a finite set of other possible relations between the terms;

R_{TO} – a relation connecting a thesaurus term to a class or relation of ontology;

A_T – a set of attributes describing the semantics of terms from T_R;

A_{TX} – a set of axioms describing the semantics of thesaurus T_H.

It is obvious that the specification of thesaurus complies with the one of ontology, and indeed their integration is a feasible method in information systems development, although approaches based on separation of vocabulary and domain ontology also exist [11]. There are also somehow intermediate solutions: for example in WordNet (http://wordnet.princeton.edu), the global lexical database, the nodes in thesaurus are not individual terms, but "synsets" – datasets that contain several synonymous terms and represent a distinct concept. However, WordNet is not concerned with relations between concepts except for linguistic ones, so it may hardly be considered a domain ontology (but possibly a language ontology).

We propose a specification of ontology based on frame model, as the following:

$$O_F = \langle F_C; F_R; F_A; F_E; I_F \rangle, \quad (3)$$

where $F_C = \langle N_C; T_R; a_C; r_C \rangle$ is a set of frames-concepts (correspond to Minsky's frames-prototypes). The frame name $N_C \in T_{DEF}$, i.e. it is the preferred term from thesaurus T_H, while the set T_R contains other terms, in different languages. There are also a set of frame-concept attributes $a_C \subseteq F_A$ and relations with other frames-concepts, $r_C \subseteq F_R$ (since a frame slot's value may be another frame);

$F_R = \langle N_R; R; a_R \rangle$ – a set of frames-relations linking concepts: N_R is the frame name, $a_R \subseteq F_A$;

$F_A = \langle N_A; A_F; TDL \rangle$ – a set of frames-attributes for concepts or relations: N_A is the frame name, $A_F = A \cup A_T$;

F_E – a set of frames-instances created based on frames-concepts (prototypes) and representing state-dependent knowledge of the domain, thus partially covering A_X;

I_F – a set of logical rules establishing semantic correctness of the domain (thus covering another fraction of A_X) or implementing procedural component in a KBS.

It should be noted that this approach removes the necessity of relations R_{TS}, R_{TE}, and R_{TO}, while T_{DEF} is formed automatically from the names of frames-concepts: $T_{DEF} = \{N_C\}$. Yet, $\{N_R\}$ and $\{N_A\}$ are not included in the thesaurus, so if these terms are deemed necessary, the corresponding frame-concept must be created. The purpose and substance of axioms, whose definition in literature is generally ambiguous, becomes more clear, and the set I_F may be also extended with rules, e.g. ones of production rule system that is a potent supplement to the naturally declarative frame model.

Production system supplementing the frame model may be presented in the following simplified form:

$$I_F = \langle N_I; L_{HS}; R_{HS} \rangle, \quad (4)$$

where N_I is name assigned to the rule, to ensure at least some order in naturally non-structured production system;

$L_{HS} = U(F_C; F_R; F_A; F_E; S_D)$ – "left-hand side", a logical expression U. To define whether it is true, inference engine assesses certain values in the current frames condition or in state-dependent data (S_D);

$R_{HS} = V(F_C; F_R; F_A; F_E; S_D)$ – "right-hand side", a set V of certain operations (actions) performed on frames or S_D if U in L_{HS} turned out to be true.

2.2 Guidelines Efficiency Evaluation Model

Due to considerable amount of existing design guidelines, we suggest they must not only be classified with tags and associated with design context, but also ranked by relative significance. Such an “efficiency” index may be evaluated based on the information obtained via explicit data collection, common for recommender systems, and in accordance with the following algorithm.

Let the number of guidelines (G) in the system equals N (thus G = $\{g_1, g_2, \ldots g_N\}$), and each is assigned efficiency index ($a_n \in [0;1]$, default is 0.5), while the number of stored interface designs (I) equals to R (thus I = $\{i_1, i_2, \ldots, i_R\}$). We can define a binary fuzzy relation GI:

$$GI = \{< g_n, i_r >, \mu_{GI}(< g_n, i_r >)\}, \tag{5}$$

where $<g_n,i_r>$ is a tuple of two elements, each of which is taken from the corresponding set: $g_n \in G$, $i_r \in I$. The membership function $\mu_{GI}(<g_n,i_r>)$ is continuous in the range [0;1] and semantically corresponds to the extent (generally evaluated by an expert) to which interface design i_r conforms to guideline g_n.

The system also stores several possible interface quality metrics (Q), whose total number is M (thus Q = $\{q_1, q_2, \ldots, q_M\}$). Let us define another binary fuzzy relation, IQ:

$$IQ = \{< i_r, q_m >, \mu_{IQ}(< i_r, q_m >)\}, \tag{6}$$

where $<i_r,q_m>$ is a tuple of two elements, each of which is taken from the corresponding set: $i_r \in I$, $q_m \in Q$. The membership function $\mu_{IQ}(<i_r, q_m>)$ is continuous in the range [0;1] and corresponds to interface design i_r quality rating on metric q_m. The quality ratings may be obtained via various usability engineering methods: usability testing (e.g. success rate), survey (e.g. subjective satisfaction), etc. As $\mu_{IQ}(<i_r,q_m>) \in [0;1]$, in some cases normalization of quality ratings would be required.

Then, based on the above, we need to define a fuzzy relation GQ meaning guideline g_n effect on enhancing quality metric q_m. We can do that via GI and IQ fuzzy relations composition with product-averaging, so that membership function will be the following:

$$\mu_{GQ}(< g_n, q_m >) = \underset{i \in I}{average}\{\mu_{GI}(< g_n, i_r >) * \mu_{IQ}(< i_r, q_m >)\}. \tag{7}$$

Considering so far the relative importance of all M quality metrics being equal, we can finally calculate efficiency index for each of N guidelines:

$$a_n = \underset{q \in Q}{average}\{\mu_{GQ}(< g_n, q_m >)\}, n = \overline{1, N}. \tag{8}$$

The values a_n may be then normalized so that their mean $\overline{a}$ = 0.5.

3 The Recommender System Implementation

3.1 Development Tools

Among existing ontology editors, we chose Protégé-Frames created within Stanford University (http://protege.stanford.edu), as it fully supports frame-based ontologies in the form that we proposed in (3). The developed ontology was imported into CLIPS (C Language Integrated Production System, http://clipsrules.sourceforge.net) that allows efficient construction of expert systems and incorporates capabilities for both logic and procedural programming, supplemented by object-oriented paradigm (OOP). The prototype of the system was made available for online access at http://clips.vgroup.su.

In our case, concepts (F_C) of the imported frame-based ontology corresponded to OOP classes and frames-instances (F_E) – to objects, while the slots (F_A) and relationships (F_R) were reflected as data fields and their types. The I_F component is based on CLIPS rules, whereas state-dependent information, such as context of a particular project, is represented with unordered facts.

3.2 The Frame Ontology and Knowledge Base

The ontology design process combined top-down and bottom-up approaches and was carried out in several iterations. The current version of the system incorporated more than 150 frames-concepts, with such classes as *Design property, HCI engineering task, HCI knowledge, Interface design, Interface element, Requirement, Target user, User attribute, Website, Website element*, etc. at the top level.

Fig. 2 outlines how the HCI knowledge structure was represented in the ontology, showing related classes, their attributes and relations. So, *Guideline* may be related to its *Source*, justification (*Law*, *Principle*, *Finding*) and assigned classification tags.

To extend the ontology into KB, we added more than 300 instances representing domain knowledge, them mostly being design guidelines that were collected from various sources ([12], [13], [14], etc.) by human experts. Assigning domain concepts (tags) to guidelines was also carried out by experts, although in the future versions of the system this process may be supplemented by automated information extraction as well.

3.3 The Application of the Recommender System

One of the practical tasks the system was applied for was the design of the official website for the People's Faculty of Novosibirsk State Technical University (NSTU) – a department that provides "computer literacy" courses for senior citizens. The specified target user attributes were reflected in the system, which then mined the project requirements for ontology terms ($T_{DEF} \cup T_R$) and produced a set of concepts describing the design context that was matched against web design-related guidelines stored in the KB. The output of the system was the ordered set of about 100 guidelines and the interface wireframe, which were used by the recommender system user to create the interface design prototype. The final version of the People's Faculty official website, improved based on results of the experiment described below, was open at http://nf.assoc.nstu.ru.

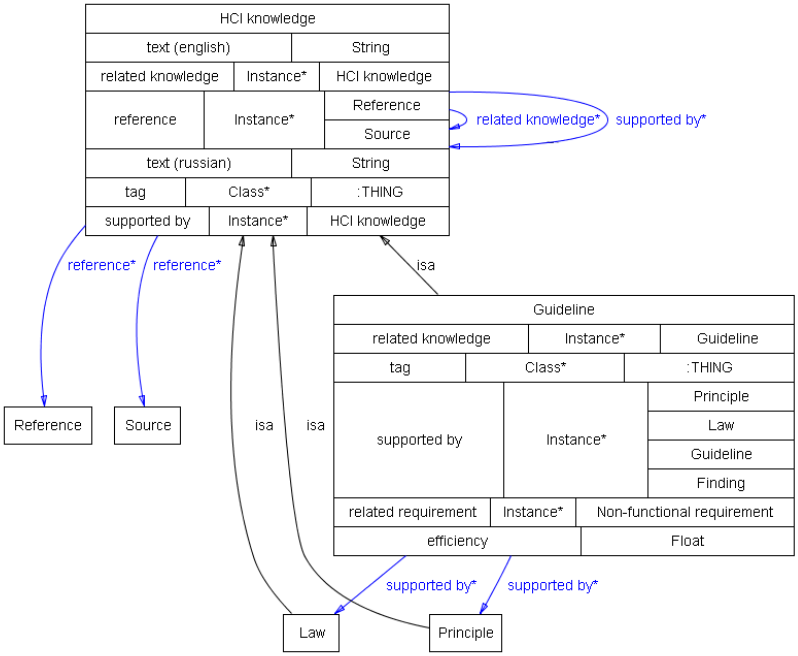

Fig. 2. HCI knowledge representation in the frame-based ontology

4 Experiment

4.1 The Experiment Description

Based on the output provided by the recommender system, designer developed the first version of the website interface, which was used in the experimental investigation together with 5 other existing websites. The goal of the experiment was twofold: a) to assess the quality of the recommender system-aided design for the target users and b) to explore if the proposed guideline efficiency evaluation model produces eloquent results. As the target user group for the People's Faculty website was senior citizens, the control group websites (#1, #2, #3, and #4) were selected from similar third-party websites also having seniors as one of target user groups. Website #5 was also a functioning one, created about 6 months before the experiment with the participation of a usability specialist affiliated with the authors of the current paper.

As for guidelines, 32 of them were elected for the experiment, on the basis of being relevant for all the 6 websites. They were divided into two types: 15 of them, which had associated tag *Elder user* in the KB, were assigned "for seniors" type (*Sen*), while all the others became of "general" type (*Yng*). Accordingly, younger participants were also employed in the experiment, so that the guidelines efficiency could be evaluated for the two subject groups independently, based on their corresponding quality metrics. The hypothesis is that guidelines of *Sen* type should have higher efficiency for senior group, while *Yng* type guidelines will have higher evaluations for younger one.

Subjects. Senior subjects were recruited among the recent graduates of computer literacy courses for seniors provided by the People's Faculty. The sampling was not random, as higher priority was given to graduates with more intense online experience, which was deemed necessary to better simulate the current and future senior population online, or even the alumni themselves in a few months from the graduation date. In total, there were 11 senior subjects (2 male, 9 female) in the experiment, whose age ranged from 58 to 71 years (mean = 62.5, SD = 4.1).

Also, there were 13 younger subjects (4 male, 9 female), recruited among undergraduate students of NSTU's Business Faculty, aged from 17 to 19 years (mean = 17.8, SD = .69). All the participants took part in the experiment voluntary and provided informed consent after reading through the tasks and learning the instructions.

Procedure. The experimental design and settings were quite typical for a user testing session, and there were in total 12 specially developed tasks, for a planned experimental session duration of 90 minutes. For each of the attempted tasks, success rate was measured by the instructor, and 0 was assigned for completely failed tasks, 0.3 – for tasks involving major errors possibly requiring support from the instructor, 0.7 – for tasks involving minor errors possibly requiring encouragement from the instructor, and 1 – for successfully completed tasks (similar approach was proposed by J. Nielsen in [15]). After completing all the tasks with all the 6 websites, the subjects were also asked to evaluate their overall impression of the websites by ranking them on a scale from 1 (worst) to 5 (best).

To evaluate guidelines efficiency separately for the two subject groups, their respective subjective impressions (normalized) were used as quality ratings $\mu_{IQ}(<i_r,q_m>)$. Success rate quality metric was not used in guideline efficiency calculation, as, expectedly, its values obtained in the experiment were stably higher for younger participants. The values for the membership function $\mu_{GI}(<g_n,i_r>)$, i.e. the degrees of interfaces correspondence to guidelines, were provided by human experts.

Thus, independent variables in the experiment were website group (third-party website, expert designer's website or website developed with recommender system), subject group, and guideline type. Dependent variables were task success rates, user subjective impressions of the websites, and evaluated guidelines efficiencies.

4.2 The Experiment Results

In total, 262 tasks were attempted by the participants and the overall mean success rate was 63.4% for the target group of senior participants. Their mean success rate for the control group of websites (#1, #2, #3 and #4) was 40.8%, while for the website developed with the system (#6) it ran up to 85.9%, however for the website developed with a usability specialist (#5) the success rate was even higher, 86.4%. Table 1 provides information on the experimental websites, together with respective success rates and subjective impressions (SI).

Table 1. Experimental websites descriptions and obtained quality metrics values

Website ID	Website description	Success rate seniors	SI seniors, Mean (SD)	SI younger, Mean (SD)
#1	http://pensionerki.ru A forum for pensioners.	23.2%	3.86 (.69)	3.18 (1.25)
#2	http://npfraiffeisen.ru A non-state pension fund.	20.0%	3.29 (1.25)	4.00 (.77)
#3	http://euro-kurses.ru A business education center.	82.2%	4.29 (.76)	4.30 (.82)
#4	http://moscow.apteka.ru An online medical shop.	52.0%	4.20 (.45)	3.64 (.81)
#5	http://vgroup.ru A web development company.	86.4%	4.33 (.52)	3.64 (1.43)
#6	http://nf.assoc.nstu.ru The People's Faculty.	85.9%	4.50 (.84)	4.50 (.71)

4.3 The Guidelines Efficiency Evaluation

Efficiencies for the 32 guidelines employed in the experiment were evaluated according to the proposed model, for the two groups of subjects independently (Table 2).

Table 2. Normalized guidelines efficiencies for younger (*Yng*) and senior (*Sen*) subjects

Guideline			Efficiency	
N	Text	Type	*Yng*	*Sen*
1	Adequately large font size, possibly with option to increase size	*Sen*	**.278**	.276
2	Uncondensed text and without long fragments in italic style	*Sen*	.517	**.519**
3	High-contrast for reading text (dark on white or beige background w/out pattern)	*Yng*	**.604**	.585
4	Senior-friendly colors (avoiding blue, green and violet tones)	*Sen*	**.500**	.492
5	Increased size for interface elements	*Sen*	.269	**.272**
6	Increased or non-existent time-outs (e.g. when filling web-forms)	*Sen*	.737	**.744**
7	Avoiding dynamic/moving elements in interface, especially in navigation	*Sen*	.722	**.730**
8	Most of webpage text visible without scrolling	*Sen*	.433	**.449**
9	Pop-up windows avoided or at least not implying scrolling	*Yng*	**.595**	.585
10	Search of adequate quality and with errors correction	*Yng*	.341	**.346**
11	Minimum amount of data input and strict formats, max use of default values	*Yng*	**.551**	.544
12	Concise web-pages with a small number of topics covered on one page	*Sen*	.513	**.515**
13	"Flat", rather than "deep", website hierarchy	*Sen*	.638	**.649**
14	Avoiding ads, especially animated or interface-like banners	*Yng*	**.599**	.597
15	Highlighted hyperlinks and consistency of their style throughout the website	*Yng*	**.419**	.408
16	Visited hyperlinks change color, preferably to more "worn-out"	*Yng*	.217	**.232**
17	Explained hyperlinks (in text or with titles), rather than "click here"	*Yng*	.247	**.251**
18	Dedicated explanation on how to use the website (e.g. Help chapter)	*Sen*	.353	**.368**
19	Consistent placement (layout) of interface elements on web pages	*Yng*	**.558**	.553
20	Simpler and familiar navigation interface (e.g. with tabs)	*Sen*	**.452**	.451
21	Avoiding multimedia materials or at least providing alternate text	*Sen*	.699	**.716**
22	Providing alternate text for all images used as interface elements	*Sen*	.613	**.615**
23	All the website content in HTML (no .pdf, .doc or .zip files)	*Sen*	.589	**.610**
24	Avoiding hyperlinks leading to the same page (anchors)	*Yng*	**.605**	.583
25	More "personal" address to website visitor in texts	*Yng*	**.421**	.409
26	Classical design (clear, orderly, pleasant, etc.) rather than "expressive"	*Sen*	.656	**.668**
27	Saturated (with less grayish) primary and secondary colors in design	*Yng*	**.473**	.459
28	Non-radical difference in primary and secondary colors hues	*Yng*	**.379**	.376
29	Trust-enhancing details about company, certifications, awards, etc.	*Yng*	**.449**	.439
30	Avoiding "marketese" text style ("new", "best", "unique", etc.)	*Yng*	.545	**.552**
31	Adequate quality of text typography, no misspellings or misprints	*Yng*	.704	**.705**
32	High-quality and high-resolution photos	*Yng*	**.324**	.301
	Mean:		.500	.500

Further, for each guideline we calculated E_d, the difference between *Yng* and *Sen* efficiencies, – it is negative if guideline is more significant in interface design for senior users rather than younger ones and positive otherwise. Employed ANOVA test showed highly significant effect of guideline type on E_d ($F_{1,30}$=15.2; p=.001), with overall means of -.007 and .0062 for senior and younger subject groups respectively.

5 Conclusion

The recommender system was developed to support interface design for modern diverse applications and websites via providing ordered set of context-dependent guidelines and interface wireframe. The system is founded upon knowledge base storing human-readable guidelines, which is built from ontology combining domain model and thesaurus used for information extraction from software project requirements.

One of the major tasks when developing the system was organization of design guidelines, which currently come from multitude of sources and are poorly integrated. The employment of knowledge representation model incorporating frame ontology with metadata allowed classification of guidelines via assigning domain concepts to them and their subsequent intersection with similar tags describing the design context. The frame ontology model also could naturally support data provenance, in particularly guidelines relations to their origin and justification in other layers of HCI knowledge (laws, principles or findings). Further, fuzzy relations-based algorithm for guideline efficiency index evaluation was proposed, which can also contribute to automated identification of duplicating guidelines in the knowledge base or providing credible recommendation in case of contradictions.

The system was built as a web application, with the core programmed in CLIPS language and the ontology developed in Protégé-Frames editor, and the intermediate online version was made available at http://clips.vgroup.su. About 150 guidelines were extracted from various sources and saved into the knowledge base of the system, which allowed its application in a practical project – the development of the official website for the People's Faculty of NSTU (http://nf.assoc.nstu.ru).

The results of the recommender system use suggest its applicability to support interface design activities, mostly to save specialists' time and avoid serious drawbacks in design. Such objective interface quality metric as success rate was 85.9% for the developed website, compared to only 40.8% for the control group. Website created with a usability specialist, however, predictably achieved even higher value of 86.4%.

Further, subjective impressions of the websites from the two subject groups, senior and younger users, were used to evaluate the guidelines efficiencies according to the proposed algorithm. The results of these values statistic analysis suggest that guidelines efficiencies were significantly higher for their relevant target user groups. The findings confirm the applicability of the recommender system and of the proposed solutions for guidelines arrangement, which could be probably also considered for complex data integration in other domains.

References

1. Bygstad, B., Ghinea, G., Brevik, E.: Software development methods and usability: Perspectives from a survey in the software industry in Norway. Interacting with Computers 20(3), 375–385 (2008)
2. Borchers, J.O.: A pattern approach to interaction design. AI & Soc. 15(4), 359–376 (2001)
3. Koukouletsos, K., Khazaei, B., Dearden, A., Ozcan, M.: Creativity and HCI: From Experience to Design in Education. IFIP, vol. 289, pp. 159–174. Springer, Boston (2009)
4. Vanderdonckt, J.: Development milestones towards a tool for working with guidelines. Interacting with Computers 12, 81–118 (1999)
5. Henninger, S., Ashokkumar, P.: An Ontology-Based Infrastructure for Usability Design Patterns. In: Proc. Semantic Web Enabled Software Engineering (SWESE), Galway, Ireland, pp. 41–55 (2005)
6. Furtado, E., Furtado, V., Soares, S.K., Vanderdonckt, J., Limbourg, Q.: KnowiXML: A Knowledge-Based System Generating Multiple Abstract User Interfaces in UsiXML. In: Proc. of 3rd Int. Workshop on Task Models and Diagrams for User Interface Design, TAMODIA 2004, pp. 121–128. ACM Press, New York (2004)
7. Chevalier, A., Fouquereau, N., Vanderdonckt, J.: The Influence of a Knowledge-Based System on Designers' Cognitive Activities: a study involving Professional Web Designers. Behaviour & Information Technology 28(1), 45–62 (2009)
8. Dearden, A., Finlay, J.: Pattern languages in HCI: A critical review. Human-Computer Interaction 21(1), 49–102 (2006)
9. Guarino, N.: Formal Ontology in Information Systems. In: Proceedings of FOIS 1998, Trento, Italy, pp. 3–15. IOS Press, Amsterdam (1998)
10. Zagorulko, Y.A., Borovikova, O.I.: Models and methods for building information systems based on ontologies (in Russian). In: Marchuk, A.G. (ed.) Sistemnaya Informatika 2011, pp. 175–207. SO RAN, Novosibirsk (2011)
11. Zagorulko, Y.A., Sidorova, E.A.: Document analysis technology in information systems for supporting research and commercial activities. Journal Optoelectronics, Instrumentation and Data Processing 45(6), 520–525 (2009)
12. Coyne, K.P., Nielsen, J.: Web usability for senior citizens. The Nielsen Norman Group, Fremont (2002)
13. Nielsen, J., Snyder, C., Molich, R., Farrell, S.: E-Commerce User Experience. Nielsen Norman Group (2001)
14. U.S. Department of Health and Human Services. Research-based web design & usability guidelines, `http://www.usability.gov/pdfs/chap.html`
15. Nielsen, J.: Success Rate: The Simplest Usability Metric. Alertbox, February 18 (2001), `http://www.useit.com/alertbox/20010218.html`