User-controlled Form Adaptation by Unsupervised Learning

DIEGO ELOI, ALAA SAHRAOUI, and JEAN VANDERDONCKT, Université catholique de Louvain, Belgium
LUIS A. LEIVA, University of Luxembourg, Luxembourg

Forms are one of the most popular and widespread methods of interaction, yet they remain largely improper for adaptation. We argue that forms should be adapted to the user and their context in a controllable way, to minimize the potential negative effects of any adaptation. This can be accomplished with Scaler, a novel web environment for designing and deploying forms in which adaptation is initiated by the system and/or the user according to a vector profile, but always under the user's control, using two unsupervised learning methods: (1) A scoring function that ranks the most usable widgets for each data item on the form, balancing the input and preferences of the stakeholders involved in the form (i.e., user, designer, and developer). (2) A widget recommendation that contrasts the user's profile and those of all other users who have used the same form, whether they have modified it before or not. Our experiment with a car booking form shows that, after some interaction sessions (from 1 to 50 depending on the form field) and some user-controlled adaptations (from 3 to 29 depending on the field), the form design converged to a stabilized selection.

CCS Concepts: • **Social and professional topics → Software selection and adaptation**; • **Computing methodologies** → *Ranking*; • **Human-centered computing → Graphical user interfaces**; *Interactive systems and tools*; User interface design.

Additional Key Words and Phrases: Adaptivity, Adaptive user interfaces, Form design, User control.

ACM Reference Format:
Diego Eloi, Alaa Sahraoui, Jean Vanderdonckt, and Luis A. Leiva. 2024. User-controlled Form Adaptation by Unsupervised Learning. In *Adjunct Proceedings of the 2024 Nordic Conference on Human-Computer Interaction (NordiCHI Adjunct 2024), October 13–16, 2024, Uppsala, Sweden.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3677045.3685431

1 INTRODUCTION

Adapting user interfaces is an essential practice in user interface design to create intuitive, efficient, and user-friendly experiences [1, 3]. By tailoring *Adaptive User Interfaces* (AUIs) [12] to meet the specific needs [18] and preferences of users [21] in their contexts of use [33, 44], we can improve usability, user experience [30], and accessibility [39], leading to engagement and satisfaction [45]. AUIs have long been studied [12] and implemented [16], being convinced that they would achieve their objective by relying on adaptation rules involved either at design-time [2, 40] or at run-time [5], using for example grammars [11, 19], specification languages [8], model-based design approaches [5, 20, 30], model-driven approaches [2, 8, 27], and model-free approaches [35, 36]. This approach has nurtured the illusion that a sufficiently large number of adaptation rules will result in an AUI that demonstrates real accuracy [22] to adapt to the user's context [17, 30, 44], a real maximization of quality factors while minimizing potentially negative effects such as disruption [29]. Adding new rules no longer serves any purpose, except to lead to new exceptions, contradictions against previous rules, and complexity in managing them.

The UI adaptation process is a multi-factorial phenomenon: it is virtually impossible to know all the conditions under which a good adaptation rule applies to produce a positive result. Too many heterogeneous factors are intertwined to guarantee a positive outcome. For example, Lavie and Meyer [33] showed that the cost of an AUI can exceed the expected benefit. Only a few of these factors are known, such as personal traits and cognitive load [21], and too few to generalize their application. In particular, the end user may well expect an AUI, but it fails to do its job because neither preference nor performance are met [3]. The main cause of these shortcomings lies in the absence—or lack of consideration—of user feedback [53] and control [15, 28] throughout the adaptation life cycle. Yet, there are many opportunities within the user's control [9] where they could intervene to steer the process in the right direction (e.g., [16] and [37] identified four and seven adaptation stages, respectively) or, at the very least, avoid undesirable or inappropriate adaptations. Langley [32] has long recommended the use of Machine Learning (ML) so that the AUI learns how to adapt under the control of the end user, rather than imposing an adaptation whose outcome is not guaranteed. However, the practical implementation of ML is proving to be more complex than expected [4, 51, 54], since a large number of samples and participants are required [41], different characteristics must be mastered, and it is not clear which adaptation rules should evolve and how. To address these shortcomings, we propose SCALER (uSer- Control-led Adaptation of Forms by Unsupervised Learning), a web environment for deploying adaptive forms under the user's control thanks to two unsupervised learning methods: a scoring function and a widget recommendation. We ran a user study showing that progressive convergence stabilizes the adaptation to a favored selection. Our work can inform future research on AUIs.

2 RELATED WORK

In the area of AUIs, *adaptive forms* represent a proven approach to enhancing user experience [30] by dynamically adjusting form widgets based on the context of use [44], which incorporates the end user's profile and the associated interactive tasks, the platform or device used to interact, and the environment in which this interaction takes place [13, 40]. Adaptive forms have been used in many practical domains, such as accounting [2], healthcare [18], municipal administration [31], multimodal systems [5, 27, 50], mobile devices [43], and business processes [46]. ML has been used for adaptation in teaching [4] and multi-screen interaction [47].

DynaForms [24], MasterMind [19] and FormGen [11] all defined a context-free grammar to specify adaptive forms, in which form fields are described using tuples (records), variants, lists, and basic predefined types to be mapped to widgets according to selection rules [6, 49]. Although these selection rules can be modified and extended at design-time, they cannot produce adaptive results at run-time, therefore preventing the form from benefiting from adaptivity gains. Similarly, AdapForm [7] promotes a form definition language that designates the structure and constraints upon acceptable input, and a software architecture that continuously validates and adapts the form. While the state of the form is kept persistently on the server side, the system ensures that all forms are valid and type safe, thereby making them helpful [8]. Other approaches use an XML-based specification of the adaptive form [8], such as xForms [27], and structured models [2, 5]. While these specifications and models are interpreted at run-time, therefore making forms rather flexible for adaptation, they do not involve any form of user control during adaptation. As a workaround, animated transitions [15] let the user see and understand the adaptivity process. However, this study suggests that end-users no longer request any such animated transition after gaining trust.

PowerForms [10] exploits JavaScript to validate user input on the client side in HTML forms: it generates an interactive form that combines static HTML form and a PowerForms specification that performs continuous validation as the form is filled. Adaptive forms should leverage data-driven personalization to enhance the user experience. By analyzing the user's interaction history, preferences, and behavior patterns, forms can be adapted to present relevant fields and options in a more personalized way [14, 38]. Adaptive forms often embed real-time feedback mechanisms to guide users as they fill out fields. Immediate validation and error messages help users correct errors on the fly, preventing frustration and enhancing the user experience. Advanced techniques, such as predictive text and contextual hints, can also be used to assist users in providing accurate information, further streamlining the process. To this end, the integration of ML into adaptive forms (using multi-class classification [51], spatio-temporal structure learning [34], or reinforcement learning [54]) represents a significant advancement, as ML can analyze user behavior in real-time to predict and suggest accurate [22] adaptation of fields, auto-complete information, and adapt the complexity of the layout based on the user's expertise level. For example, a novice user might see a simplified form, while an expert user receives a more detailed version [2]. In sum, adaptive forms existed for a while, but without user control, ML techniques have recently been considered to support form adaptation.

3 USER-CONTROLLED FORM ADAPTATION

Fig. 1 shows a system walkthrough of Scaler on a car rental UI, a W3C reference case study [42]: ① the user creates and edits the specifications of form fields in terms of data type, format, domain of values, input method (e.g., by typing or selecting), and constraints. Using these specifications, Scaler automatically selects relevant Abstract Interaction Objects [49] that can be reviewed and edited by a designer based on design experience or the end user based on preferences; ② the resulting form is rendered in HTML5. To make the form adaptive, Scaler enables the end user to specify a scoring function ③ and/or to rely on a recommendation mechanism ④, using two unsupervised learning methods that consider three profiles: *anonymous users* (since they are not authen-ticated, they can provide Scaler with feedback and adapt the form to their purpose, thereby feeding the global system, but their adaptation will be lost for them), *identified users* (since they are authenticated, they benefit from all unsupervised learning to adapt their forms), and *administrators* (they are authenticated designers, developers, or super-users who can create, edit, and delete any form). An administrator can also assign a role to any other user, such as “writer”, “translator”, or “manager”.

3.1 Unsupervised Learning by Scoring

Scaler calculates a score for adapting each widget corresponding to any form field to the context of use, then selects and displays the widget with the highest score. If the user wants to change the selected widget (via the Change widget push button), the list of other widget proposals is sorted in descending order of their score based on four sub-scores:

(1) *Score of Change* (*SC*): a score assigned to a widget chosen by the end-user when it was not the default widget (i.e., it was not the widget with the highest score).
(2) *Score of Unchange* (*SU*): a score assigned to a widget chosen by the end user when it was the default widget (i.e., the widget that received the highest score).
(3) *Score of Admin* (*SA*): a score assigned to a widget chosen by the administrator as the default choice, assuming that the selection is not randomly performed.
(4) *Score Global* (*SG*): an overall score assigned to all widgets of this type when any of them is selected. For this purpose, a list of potential widgets is assigned to any Abstract Interaction Object (AIO), which corresponds to one type (see ② in Fig. 1).

To determine these scores, we define three system variables for every widget of every form:

(1) *Number of Changes* (C): the number of times a widget has been chosen by the end-user when it was not the default widget.
(2) *Number of Unchanges* (U): the number of times a widget has been chosen by the end-user when it was already the default widget.
(3) *Number of Global Choices* (G): the number of times a widget of this type has been chosen by end users.

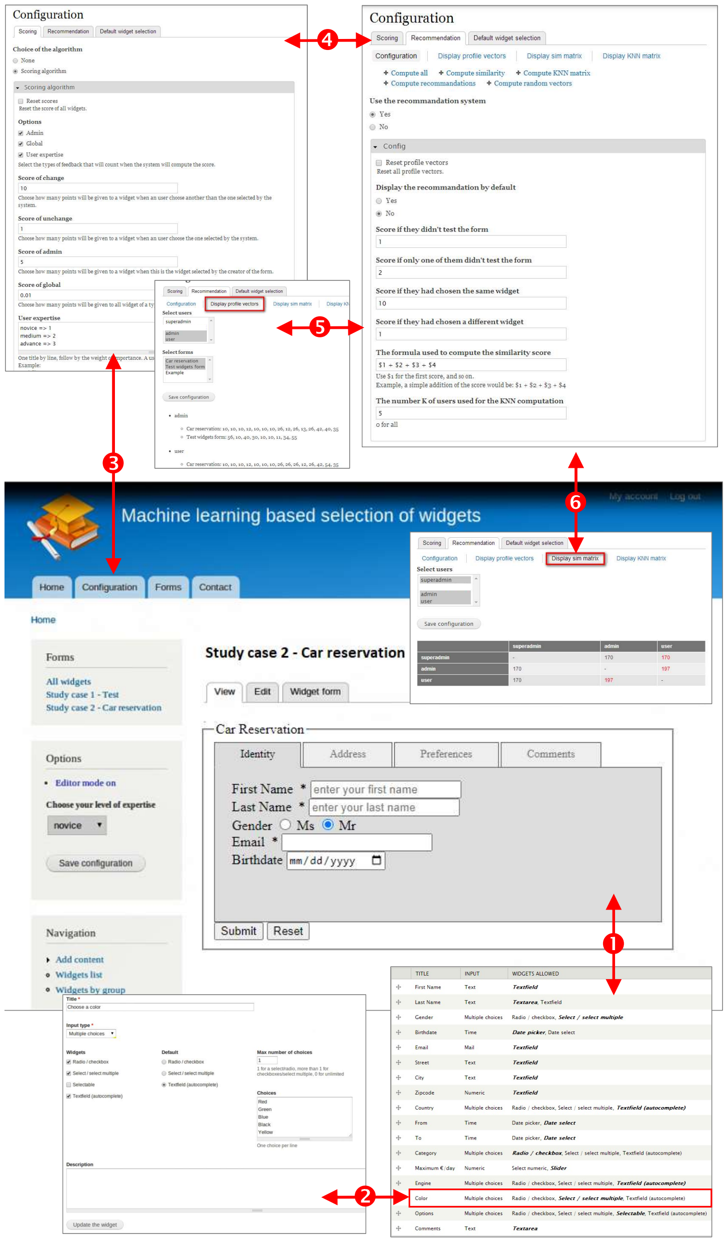
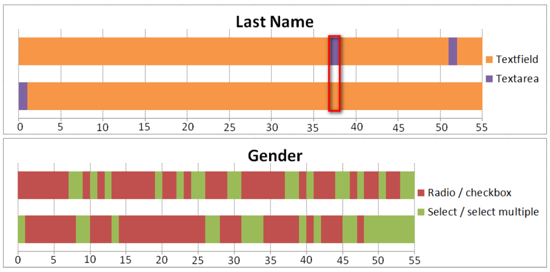


Fig. 1. Scaler system walkthrough: ① editing the specifications of form fields and direct rendering, ② connecting to an abstract interaction object, ③ specifying a scoring function, ④ defining the recommendation mechanism, ⑤ displaying the vector profile of a user, and ⑥ accessing the similarity matrix.

$$\text{scoring}\,(w, form) = \begin{cases} C \times SC + U \times SU + G \times SG & \text{if Admin=false and Global=false} \\ C \times SC + U \times SU & \text{if Admin=false and Global=true} \\ C \times SC + U \times SU + f(w, SA) & \text{if Admin=true and Global=false} \\ C \times SC + U \times SU + G \times SG + f(w, SA) & \text{if Admin=true and Global=true} \end{cases}$$

Fig. 2. Definition of the scoring function.



Selected vs. Pred.	Textfield	Textarea	Total
Textfield	52	1	53 (96.4%)
Textarea	2	0	2 (3.60%)
Total	54	1	52 (94.5%)

Selected vs. Pred.	Radio	Select	Total
Radio	21	15	36 (65.5%)
Select	14	5	19 (34.5%)
Total	35	20	26 (47.3%)

Fig. 3. Some scarf plots and confusion matrices of our case study.

Since these variables are specifically tailored to every widget of every form, their values and treatments can largely vary from one form to another. Scaler captures the context of use (e.g., the user ID or the height and width of the screen) and the system variables in a hidden text area included in each form: all actions and events are captured via W3C Window Object and W3C Navigator Object, A *scoring function*, also referred to as an objective function, is used in optimization problems to evaluate the performance or the quality of a model's predictions or solutions [41]. We define the Scaler scoring function in Fig. 2 depending on the parameters selected (③ in Fig. 1), where $f(w, SA) = SA$ if the widget w was the default widget selected by Scaler or the designer or 0 otherwise. The end user then (un)checks the parameters of this scoring function depending on the adaptation needs and preferences.

3.2 Unsupervised Learning by Recommendation

The second method consists of applying the k-nearest neighbors (k-NN) algorithm, a non-parametric, unsupervised learning classifier using proximity to make predictions about the grouping of an individual object or 'data point'. We assume this is the case for any user who will be near other users of the same form, either in similar or different contexts of use. This method is typically used to classify an object with respect to a class of objects among its k nearest neighbors [26] The underlying assumption in k-NN is that similar objects can be found near each other. In 1-NN, its simplified version with k=1, the candidate object is simply assigned to the class of the closest neighbor. Scaler uses a 1-NN to determine suitable recommendations for each user (④ in Fig. 1). For this purpose, a user vector profile is created to express their preferences for interaction and for the available widgets to investigate similarities between users (⑤ in Fig. 1): an indexed vector referring to a set of available widgets where "0" indicates that an existing widget has never been used, "1" indicates a widget used, and "-1" is used to indicate new widgets.

A similarity matrix is dynamically created to identify common user profiles (⑥ in Fig. 1). In this matrix, the value of a point (i, j) is the Euclidean distance between i and j. Once the similarity index is calculated and the k nearest profile is determined, a prediction value is computed by widget, as a recommendation for the user. The prediction score of a widget j to a user i is calculated as the sum of the widget frequencies for users, weighted by the similarity of the users a_{pj} (Fig. **??**). Similarly to the scoring function presented in the previous section, the weights representing the various types of users can be adjusted to reflect their relative importance. For example, if the designer's choice is considered the most important, its corresponding weight can be increased by decreasing the weight for another user or developer.

4 USER STUDY

We conducted a user study with 55 randomly selected participants (50 male, 5 female) from our institution's mailing lists, Participants were 18–50 years old (M=24 years) and had different expertise levels in using forms (novice, intermediate, and expert). After a brief introduction to the study, the participants were instructed to use the car rental UI to complete a reservation, allowing them to adapt the forms according to their preferences. Participants were granted user access to Scaler so that they could adapt the forms from their own browser. Fig. 3 shows the results of the adaptation process as a scarf plot [52] (left) and a confusion matrix (right). For example, the Last name field was subject to selection to a textfield or a textarea: one user adapted a textfield to a textarea and two users used a textarea without changing it. Similar figures are obtained for all form fields. We observe a progressive convergence that stabilises the adaptation on the most preferred selection of widgets: after some adaptation and some time, users no longer required any adaptation.

5 CONCLUSION AND FUTURE WORK

We have presented Scaler, a web environment for designing and deploying forms in which adaptation is initiated by the system and/or the user under the user's control, using two unsupervised learning methods: a scoring function and a recommendation algorithm. While the end user is indeed able to control the adaptation process, namely by choosing the learning method and parameterizing it, further experimentation is desired to investigate which method they prefer and how they determine the weights according to their preferences, which is a multi-factorial and elusive problem. The future of adaptive forms and AUIs lies in further advancements in ML, such as with reinforcement learning [48, 54], with an increased emphasis on privacy and security [25], and the seamless integration of multimodal interaction (e.g., voice, touch, and gestures). As these technologies evolve (for example, Gaspar-Figueiredo et al. [23] used EEG to select adaptive graphical menus), AUIs will become more sophisticated, capable of providing highly adaptive, context-aware, and inclusive experiences. In conclusion, adaptive forms in graphical user interfaces represent a significant leap forward in user experience design provided that an appropriate mechanism for user-control is incorporated. By dynamically adjusting to the user's context, preferences, and behavior, these forms enhance usability, accessibility, and efficiency. As technology continues to advance, in particular, with the progress of machine learning, the potential for even more intelligent and intuitive adaptive forms will certainly grow, further transforming the landscape of form interaction.

ACKNOWLEDGMENTS

This work is supported by the European Innovation Council Pathfinder-Awareness Inside challenge "Symbiotik" project (1 Oct. 2022-31 Dec. 2026) under Grant no. 101071147. We thank Nicolas Foret for his contributions to this work.

REFERENCES

[1] Silvia Abrahão, Emilio Insfrán, Arthur Sluÿters, and Jean Vanderdonckt. 2021. Model-based intelligent user interface adaptation: challenges and future directions. *Software and Systems Modelling* 20, 5 (2021), 1335–1349. https://doi.org/10.1007/s10270-021-00909-7

[2] Pierre A. Akiki, Arosha K. Bandara, and Yijun Yu. 2014. Adaptive Model-Driven User Interface Development Systems. *Comput. Surveys* 47, 1 (2014), 9:1–9:33. https://doi.org/10.1145/2597999

[3] Victor Alvarez-Cortes, Victor H. Zarate, Jorge A. Ramirez Uresti, and Benjamin E. Zayas. 2009. Current Challenges and Applications for Adaptive User Interfaces. In *Human-Computer Interaction*, Inaki Maurtua (Ed.). IntechOpen, London, UK, Chapter 3, 49–68. https://doi.org/10.5772/7745

[4] Karl-Emil Kjær Bilstrup, Magnus Høholt Kaspersen, Matilde Fjeldsø Larsen, Niels Olof Bouvin, and Marianne Graves Petersen. 2022. The Best of Both Worlds: Designing a Tiered Hybrid Interface for Teaching Machine Learning in K-9 Education. In *Proceedings of the Nordic Human-Computer Interaction Conference* (Aarhus, Denmark) *(NordiCHI '22)*. Association for Computing Machinery, New York, NY, USA, 51:1–51:12. https://doi.org/10.1145/3546155.3546156

[5] Marco Blumendorf, Grzegorz Lehmann, and Sahin Albayrak. 2010. Bridging models and systems at runtime to build adaptive user interfaces. In *Proceedings of the 2nd ACM SIGCHI Symposium on Engineering Interactive Computing System* (Berlin, Germany) *(EICS '10)*, Noi Sukaviriya, Jean

Vanderdonckt, and Michael Harrison (Eds.). Association for Computing Machinery, New York, NY, USA, June 19–23, 2010. https://doi.org/10.1145/1822018.1822022

[6] François Bodart and Jean Vanderdonckt. 1994. On the Problem of Selecting Interaction Objects. In *Proceedings of BCS International Conference on Human-Computer Interaction – People and Computers IX* (Glasgow, UK) *(HCI '94)*, Gilbert Cockton, Stephen W. Draper, and George R. S. Weir (Eds.). Cambridge University Press, Cambridge, UK, 163–178. https://doi.org/10.1017/CBO9780511600821.013

[7] Morten Bohøj, Niels Olof Bouvin, and Henrik Gammelmark. 2011. AdapForms: A Framework for Creating and Validating Adaptive Forms. In *Web Engineering*, Sören Auer, Oscar Díaz, and George A. Papadopoulos (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 105–120. https://doi.org/10.1007/978-3-642-22233-7_8

[8] Morten Bohøj, Niels Olof Bouvin, and Henrik Gammelmark. 2012. A Framework for Interactively Helpful Web Forms. *Journal of Web Engineering* 11, 1 (2012), 1–22. http://www.rintonpress.com/xjwe11/jwe-11-1/001-022.pdf

[9] Sarah Bouzit, Gaëlle Calvary, Joëlle Coutaz, Denis Chêne, Eric Petit, and Jean Vanderdonckt. 2017. The PDA-LPA Design Space for User Interface Adaptation. In *Proc. of the 11th IEEE Int. Conf. on Research Challenges in Information Science* (Brighton, UK) *(RCIS '17)*. IEEE Press, Hoboken, New Jersey, USA, 353–364. https://doi.org/10.1109/RCIS.2017.7956559

[10] Claus Brabrand, Anders Møller, Mikkel Ricky, and Michael I. Schwartzbach. 2000. PowerForms: Declarative client-side form field validation. *World Wide Web* 3 (2000), 205–214. https://doi.org/10.1023/A:1018772405468

[11] Alfons Brandl and Gerwin Klein. 1999. FormGen: A Generator for Adaptive Forms Based on EasyGUI. In *Proceedings of the 8th International Conference on Human-Computer Interaction, Human-Computer Interaction: Ergonomics and User Interfaces* (Munich, Germany) *(HCI International '99, Vol. 1)*, Hans-Jörg Bullinger and Jürgen Ziegler (Eds.). Lawrence Erlbaum, 1172–1176. https://www.researchgate.net/publication/221098264_FormGen_A_Generator_for_Adaptive_Forms_Based_on_EasyGUI

[12] Dermot Browne, Peter Totterdell, and Mike Norman (Eds.). 1990. *Adaptive User Interfaces*. Academic Press, London, UK. https://doi.org/doi/10.5555/130348

[13] Gaëlle Calvary, Joëlle Coutaz, David Thevenin, Quentin Limbourg, Laurent Bouillon, and Jean Vanderdonckt. 2003. A Unifying Reference Framework for Multi-target User Interfaces. *Interacting with Computers* 15, 3 (2003), 289–308. https://doi.org/10.1016/S0953-5438(03)00010-9

[14] Juan Cruz-Benito, Andrea Vázquez-Ingelmo, José Carlos Sánchez-Prieto, Roberto Therón, Francisco José García-Peñalvo, and Martín Martín-González. 2018. Enabling Adaptability in Web Forms Based on User Characteristics Detection Through A/B Testing and Machine Learning. *IEEE Access* 6 (2018), 2251–2265. https://doi.org/10.1109/ACCESS.2017.2782678

[15] Charles-Eric Dessart, Vivian Genaro Motti, and Jean Vanderdonckt. 2011. Showing User Interface Adaptivity by Animated Transitions. In *Proceedings of the ACM Symposium on Engineering Interactive Computing Systems* (Pisa, Italy) *(EICS '11)*. Association for Computing Machinery, New York, NY, USA, 95–104. https://doi.org/10.1145/1996461.1996501

[16] Hartmut Dieterich, Uwe Malinowski, Thomas Kuhme, and Matthias Schneider-Hufschmidt. 1994. State of the Art in Adaptive User Interfaces. In *Adaptive User Interfaces Principles and Practice*, M. Schneider-Hufschmidt, T. Kuhme, and U. Malinowski (Eds.). Elsevier Science Publishers, Amsterdam, Chapter 10, 13–48. https://www.elsevier.com/books/adaptive-user-interfaces/schneider-hufschmidt/978-0-444-81545-3

[17] Mateusz Dubiel, Bereket Abera Yilma, Kayhan Latifzadeh, and Luis A. Leiva. 2022. A Contextual Framework for Adaptive User Interfaces: Modelling the Interaction Environment. *CoRR* abs/2203.16882 (2022). https://doi.org/10.48550/arXiv.2203.16882 arXiv:2203.16882

[18] Mahboubeh Eslami, Mohammad Firoozabadi, and Elaheh Homayounvala. 2018. User Preferences for Adaptive User Interfaces in Health Information Systems. *Universal Access in Information Society* 17, 4 (Nov. 2018), 875–883. https://doi.org/10.1007/s10209-017-0569-1

[19] Martin R. Frank and Pedro A. Szekely. 1998. Adaptive Forms: an interaction technique for entering structured data. *Knowl. Based Syst.* 11, 1 (1998), 37–45. https://doi.org/10.1016/S0950-7051(98)00058-6

[20] Elizabeth Furtado, Vasco Furtado, Wilker Bezerra Silva, Daniel William Tavares Rodrigues, Leandro da Silva Taddeo, Quentin Limbourg, and Jean Vanderdonckt. 2001. An Ontology-Based Method for Designing Multiple User Interfaces. In *Proceedings of Int. Workshop on Multiple User Interfaces (MUI' 01)*. https://www.researchgate.net/publication/2567741_An_Ontology-Based_Method_for_Universal_Design_of_User_Interfaces

[21] Krzysztof Z. Gajos and Krysta Chauncey. 2017. The Influence of Personality Traits and Cognitive Load on the Use of Adaptive User Interfaces. In *Proceedings of the 22nd International Conference on Intelligent User Interfaces* (Limassol, Cyprus) *(IUI '17)*. Association for Computing Machinery, New York, NY, USA, 301–306. https://doi.org/10.1145/3025171.3025192

[22] Krzysztof Z. Gajos, Katherine Everitt, Desney S. Tan, Mary Czerwinski, and Daniel S. Weld. 2008. Predictability and Accuracy in Adaptive User Interfaces. In *Proceedings of the ACM Conference on Human Factors in Computing Systems* (Florence, Italy) *(CHI '08)*. Association for Computing Machinery, New York, NY, USA, 1271—-1274. https://doi.org/10.1145/1357054.1357252

[23] Daniel Gaspar-Figueiredo, Silvia Abrahão, Emilio Insfrán, and Jean Vanderdonckt. 2023. Measuring User Experience of Adaptive User Interfaces using EEG: A Replication Study. In *Proceedings of the 27th International Conference on Evaluation and Assessment in Software Engineering* (Oulu, Finland) *(EASE 23)*. Association for Computing Machinery, New York, NY, USA, 52–61. https://doi.org/10.1145/3593434.3593452

[24] Andreas Girgensohn, Beatrix Zimmermann, Alison Lee, Bart Burns, and Michael E. Atwood. 1995. *Dynamic Forms: An Enhanced Interaction Abstraction Based on Forms*. Springer US, Boston, MA, 362–367. https://doi.org/10.1007/978-1-5041-2896-4_60

[25] Amr Gomaa. 2022. Adaptive User-Centered Multimodal Interaction towards Reliable and Trusted Automotive Interfaces. In *Proceedings of the ACM International Conference on Multimodal Interaction* (Bengaluru, India) *(ICMI '22)*. Association for Computing Machinery, New York, NY, USA, 690–695. https://doi.org/10.1145/3536221.3557034

[26] Trevor Hastie, Robert Tibshirani, and Jérome Friedman. 2009. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* Springer New York, New York, USA. https://doi.org/10.1007/978-0-387-84858-7

[27] Mikko Honkala and Mikko Pohja. 2006. Multimodal interaction with xforms. In *Proceedings of the 6th International Conference on Web Engineering* (Palo Alto, California, USA) *(ICWE '06)*, David Wolber, Neil Calder, Christopher H. Brooks, and Athula Ginige (Eds.). Association for Computing Machinery, New York, NY, USA, 201–208. https://doi.org/10.1145/1145581.1145624

[28] Eric Horvitz. 1999. Principles of Mixed-Initiative User Interfaces. In *Proceeding of the ACM Int. Conf. on Human Factors in Computing Systems* (Pittsburgh, PA, USA, May 15-20) *(CHI '99)*. Association for Computing Machinery, New York, NY, USA, 159–166. https://doi.org/10.1145/302979.303030

[29] Bowen Hui, Grant Partridge, and Craig Boutilier. 2009. A Probabilistic Mental Model for Estimating Disruption. In *Proceedings of the 14th International Conference on Intelligent User Interfaces* (Sanibel Island, Florida, USA) *(IUI '09)*. Association for Computing Machinery, New York, NY, USA, 287–296. https://doi.org/10.1145/1502650.1502691

[30] Jamil Hussain, Anees Ul Hassan, Hafiz Syed Muhammad Bilal, Rahman Ali, Muhammad Afzal, Shujaat Hussain, Jae Hun Bang, Oresti Banos, and Sungyoung Lee. 2018. Model-based Adaptive User Interface based on Context and User experience Evaluation. *Journal of Multimodal User Interfaces* 12, 1 (2018), 1–16. https://doi.org/10.1007/s12193-018-0258-2

[31] Pieternel Kuiper and Betsy van Dijk. 2009. Adaptive Municipal Electronic Forms. In *Handbook of Research on E-Transformation and Human Resources Management Technologies: Organizational Outcomes and Challenges*, Tanya Bondarouk, Huub Ruel, Karine Guiderdoni-Jourdain, and Ewan Oiry (Eds.). IGI Global, Hershey, PA, USA, 1116–133. https://doi.org/10.4018/978-1-60566-304-3.ch007

[32] Pat Langley. 1997. Machine Learning for Adaptive User Interfaces. In *Proceedings of 21st Annual German Conference on Artificial Intelligence, Advances in Artificial Intelligence 97* (Freiburg, Germany) *(Lecture Notes in Artificial Intelligence, Vol. 1303)*, Gerhard Brewka, Christopher Habel, and Bernhard Nebel (Eds.). Springer, Berlin, Heidelberg, 53–62. https://doi.org/10.1007/3540634932_3

[33] Talia Lavie and Joachim Meyer. 2010. Benefits and costs of adaptive user interfaces. *International Journal of Human-Computer Studies* 68, 8 (2010), 508–524. https://doi.org/10.1016/j.ijhcs.2010.01.004

[34] Hosub Lee, Youngsang Choi, and Yeojin Kim. 2011. An adaptive user interface based on Spatiotemporal Structure Learning. In *Proceedings of IEEE Consumer Communications and Networking Conference* (Las Vegas, NV, USA) *(CCNC'11)*. IEEE Computer Society, Los Alamitos, CA, USA, 923–927. https://doi.org/10.1109/CCNC.2011.5766642

[35] Luis A. Leiva. 2011. Restyling website design via touch-based interactions. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services* (Stockholm, Sweden) *(MobileHCI '11)*. Association for Computing Machinery, New York, NY, USA, 599–604. https://doi.org/10.1145/2037373.2037467

[36] Luis A. Leiva. 2012. Automatic web design refinements based on collective user behavior. In *CHI '12 Extended Abstracts on Human Factors in Computing Systems* (Austin, Texas, USA) *(CHI EA '12)*. Association for Computing Machinery, New York, NY, USA, 1607–1612. https://doi.org/10.1145/2212776.2223680

[37] Víctor López-Jaquero, Jean Vanderdonckt, Francisco Montero Simarro, and Pascual González. 2007. Towards an Extended Model of User Interface Adaptation: The ISATINE Framework. In *Proceedings of the Joint Working Conferences on Engineering Interactive Systems, EIS'07-EHCI'07-DSV-IS'07-HCSE'07* (Salamanca, Spain) *(Lecture Notes in Computer Science, Vol. 4940)*, Jan Gulliksen, Morten Borup Harning, Philippe A. Palanque, Gerrit C. van der Veer, and Janet Wesson (Eds.). Springer, 374–392. https://doi.org/10.1007/978-3-540-92698-6_23

[38] Uwe Malinowski. 1993. Adjusting the presentation of forms to users' behavior. In *Proceedings of the 1st International Conference on Intelligent User Interfaces* (Orlando, Florida, USA) *(IUI '93)*. Association for Computing Machinery, New York, NY, USA, 247–249. https://doi.org/10.1145/169891.170016

[39] Mahdi H. Miraz, Maaruf Ali, and Peter S. Excell. 2021. Adaptive user interfaces and universal usability through plasticity of user interface design. *Computer Science Review* 40 (2021), 100363. https://doi.org/10.1016/j.cosrev.2021.100363

[40] Vivian G. Motti and Jean Vanderdonckt. 2013. A Computational Framework for Context-aware Adaptation of User Interfaces. In *Proceedings of the 7th IEEE Int. Conf. on Research Challenges in Information Science (RCIS '13)*. 1–12. https://doi.org/10.1109/RCIS.2013.6577709

[41] Jeffrey Nichols. 2013. Using the Crowd to Understand and Adapt User Interfaces. In *Proceedings of the ACM SIGCHI Symposium on Engineering Interactive Computing Systems* (London, United Kingdom) *(EICS '13)*. Association for Computing Machinery, New York, NY, USA, 1–2. https://doi.org/10.1145/2494603.2480344

[42] Fabio Paternò, Carmen Santoro, Lucio Davide Spano, and Dave Raggett. 2014. *Model-Based User Interface (MBUI) - Task Models, W3C Working Group Note.* Technical Report. World-Wide Web Consortium, Cambridge, Massachusetts, USA. https://www.w3.org/TR/task-models/

[43] Pertti Repo. 2004. Facilitating user interface adaptation to mobile devices. In *Proceedings of the Third Nordic Conference on Human-Computer Interaction* (Tampere, Finland) *(NordiCHI '04)*. Association for Computing Machinery, New York, NY, USA, 433–436. https://doi.org/10.1145/1028014.1028088

[44] Anil Shankar, Sushil J. Louis, Sergiu Dascalu, Linda J. Hayes, and Ramona Houmanfar. 2007. User-Context for Adaptive User Interfaces. In *Proceedings of the 12th International Conference on Intelligent User Interfaces* (Honolulu, Hawaii, USA) *(IUI '07)*. Association for Computing Machinery, New York, NY, USA, 321–324. https://doi.org/10.1145/1216295.1216357

[45] Makram Soui, Khaled Ghédira, and Mourad Abed. 2015. Evaluating User Interface Adaptation using the Context of Use. *International Journal of Adaptive, Resilient and Autonomic Systems (IJARAS)* 6, 1 (2015), 1–24. https://doi.org/10.4018/IJARAS.2015010101

[46] Kênia Sousa, Hildeberto Mendonça, Jean Vanderdonckt, Els Rogier, and Joannes Vandermeulen. 2008. User Interface Derivation from Business Processes: A Model-Driven Approach for Organizational Engineering. In *Proceedings of the 2008 ACM Symposium on Applied Computing* (Fortaleza,

Ceara, Brazil) *(SAC '08)*. Association for Computing Machinery, New York, NY, USA, 553–560. https://doi.org/10.1145/1363686.1363821

[47] Hua-Zhe Tan, Wei Zhao, and Hai-Hua Shen. 2018. Adaptive user interface optimization for multi-screen based on machine learning. In *Proceedings of IEEE 22nd International Conference on Computer Supported Cooperative Work in Design* (Nanjing, China) *(CSCWD '18)*. 743–748. https://doi.org/10.1109/CSCWD.2018.8465348

[48] Kashyap Todi, Gilles Bailly, Luis Leiva, and Antti Oulasvirta. 2021. Adapting User Interfaces with Model-based Reinforcement Learning. In *Proceedings of the ACM CHI Conference on Human Factors in Computing Systems (CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 573, 13 pages. https://doi.org/10.1145/3411764.3445497

[49] Jean Vanderdonckt and François Bodart. 1993. Encapsulating knowledge for intelligent automatic interaction objects selection. In *Proceedings of jointly organised ACM Conference on Human Aspects in Computing Systems CHI'93 and IFIP TC13 International Conference on Human-Computer Interaction* (Amsterdam, The Netherlands), Bert Arnold, Gerrit C. van der Veer, and Ted N. White (Eds.). Association for Computing Machinery, New York, NY, USA, 424–429. https://doi.org/10.1145/169059.169340

[50] Jean Vanderdonckt, Gaëlle Calvary, Joëlle Coutaz, and Adrian Stanciulescu. 2008. *Multimodality for Plastic User Interfaces: Models, Methods, and Principles.* Springer Berlin Heidelberg, Berlin, Heidelberg, 61–84. https://doi.org/10.1007/978-3-540-78345-9_4

[51] Anuradha Welivita and Tharindu Ranathunga. 2016. *A Machine Learning Approach to Generate Adaptive User Interfaces.* Technical Report. 99X Technology, Walukarama Road, Colombo, Sri Lanka. https://doi.org/10.13140/RG.2.2.17446.22082

[52] Chia-Kai Yang and Chat Wacharamanotham. 2018. Alpscarf: Augmenting Scarf Plots for Exploring Temporal Gaze Patterns. In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems* (<conf-loc>, <city>Montreal QC</city>, <country>Canada</country>, </conf-loc>) *(CHI EA '18)*. Association for Computing Machinery, New York, NY, USA, 1–6. https://doi.org/10.1145/3170427.3188490

[53] Enes Yigitbas, André Hottung, Sebastian Mansfield Rojas, Anthony Anjorin, Stefan Sauer, and Gregor Engels. 2019. Context- and Data-driven Satisfaction Analysis of User Interface Adaptations Based on Instant User Feedback. *Proc. of ACM Human-Computer Interaction* 3, EICS (2019), 19:1–19:20. https://doi.org/10.1145/3331161

[54] Lamia Zouhaier, Yousra Ben Daly Hlaoui, and Leila Ben Ayed. 2021. A Reinforcement Learning Based Approach of Context-driven Adaptive User Interfaces. In *Proceedings of IEEE 45th Annual Computers, Software, and Applications Conference* (Madrid, Spain) *(COMPSAC '21)*. IEEE Press, Los Alamitos, USA, 1463–1468. https://doi.org/10.1109/COMPSAC51774.2021.00217