

Generation of Adaptive Mobile Applications Based on Design Patterns for User Interfaces [†]

Amani Braham ¹, Félix Buendía ^{2,*}, Maha Khemaja ³ and Faiez Gargouri ⁴

¹ ISITCOM, University of Sousse, 4000 Sousse, Tunisia; amanibraham@gmail.com

² Universitat Politècnica deValencia, 46022 Valencia, Spain

³ University of Sousse, 4000 Sousse, Tunisia; khemajamaha@gmail.com

⁴ University of Sfax, 3029 Sfax, Tunisia; faiez.gargouri@gmail.com

* Correspondence: fbuendia@disca.upv.es; Tel.: +34-682-347-392

[†] Presented at the 13th International Conference on Ubiquitous Computing and Ambient Intelligence UCAmI 2019, Toledo, Spain, 2–5 December 2019.

Published: 20 November 2019

Abstract: Generating mobile apps represents a big challenge in several areas, such as considering audience needs, adapting their user interfaces to such needs, dealing with design constraints or using different development technologies. The present work seeks to examine how design patterns can help to support the generation of this kind of adaptive mobile application. In particular, design patterns related to user interfaces are reviewed, and an ontology-based framework is proposed to manage their pattern descriptions and associated rules. Such a framework enables a more versatile and powerful organization of mobile interface items, as well as their adaptation to context changes and user requirements in specific scenarios. An example of adaptive mobile application has been developed to show the suitability of the proposed framework, and the application usability has been evaluated in terms of satisfaction, learnability, and efficiency.

Keywords: mobile applications; adaptive user interface; design patterns; ontology models; Usability tests

1. Introduction

The generation of mobile applications (i.e., apps) represents a big challenge in several areas addressing the analysis of user needs, the visual design of their components, the implementation details associated to these applications, or testing their usability. In general, these apps must deal with different user requirements according to specific profiles or take into account their preferences or special needs. Other surveys have studied how users interact with apps based on the effects of geographic aspects [1] or cultural factors [2] but attention has also been paid to the cognitive load for mobile applications from a design perspective [3]. The current research aims at highlighting the role of Adaptive User Interfaces (AUIs) in order to address issues such as the adaptation shortages when generating app interfaces or the difficulties in managing their complexity. AUIs can be viewed as systems that change their structures or contents according to diverse needs and user contexts [4]. In this sense, the rise of mobile devices and related applications demands increasing attention to the development of adaptive and personalized interfaces [5]. The main goal of this work consists in providing a framework able to show how design patterns can support the generation of this type of mobile application. Design patterns have been demonstrated to be useful in the design of adaptive user interfaces [6], but they need to be complemented with semantic models that contribute to managing user information and their surrounding context. This is the case for the current framework, which is derived from initiatives such as mobile ontology [7] to represent context-aware features, or

the mIO! proposal [8] that aims at defining contextual knowledge about how users can influence their interactions with mobile devices.

As mentioned before, cognitive load is another factor to take into account when designing or developing mobile apps. Reducing cognitive load contributes to better user interactions as it supports a more efficient management of visual memory design [9]. Therefore, a relevant aspect in the proposed framework consists of the selection of mobile design patterns oriented towards minimizing or varying overload during the use of adaptive apps along with capturing user attention and increasing their cognitive engagement [3]. Multiple usability studies have been performed to evaluate the impact of these mobile design factors in different areas such as commerce [10] or healthcare [11], and some of the mechanisms and techniques mentioned in these studies can be applied to test the selected design patterns. Moreover, recent publications [12,13] pay special attention to the necessity of including adapted interface designs and assistive mobile technology for several user profiles. Therefore, a usability evaluation of the generated mobile apps will be based on testing their adaptation among a set of users with different characteristics and needs. The remainder of the document is structured as follows. Section 2 mentions the past studies related to the use of design patterns in the generation of mobile apps and their adaptation capabilities. Section 3 describes the framework proposed in the current work to generate such adaptive apps, beginning with the basic components that underlie it and providing a global overview about its architecture. In section 4, an example of a mobile app is introduced to show the suitability of this framework, addressing several scenarios adapted to different user profiles. Section 5 presents an evaluation of this app based on usability dimensions, such as the degree of users' satisfaction and the learnability or time taken to complete some of the app tasks. Finally, some conclusions and further research direction are proposed.

2. Related Work

UI design patterns made their breakthrough in the HCI (Human Computer Interaction) domain, when Coram and Lee [14] introduced the first design pattern inspired by the need to enhance the way graphical designers can build user interfaces. In particular, the role of design patterns in mobile UI have been deeply studied in the context of featuring user experiences [15] or collecting and classifying these experiences in different problem areas [16]. Galleries and catalogues of UI mobile patterns have been developed [17], so there are a high number of available examples ready to be used. Several works that combine the use of models and design patterns have been proposed in the development of user interfaces. The PD-MBUI (Pattern-Driven and Model-Based User Interface) framework [18] provides a bridge between design patterns as a mechanism for generating UIs and model-driven software engineering methods to guide this generation process. A similar approach was pointed out in PaMGIS [19] for developing user interfaces for interactive ubiquitous applications. These approaches show the potential of pattern-based methods in terms of reducing complexity and reusing existing design knowledge. Despite the introduction of design patterns in these works, they lack mechanisms that enable the automatic instantiation of patterns addressed to adapt user interfaces at runtime. Alternative proposals have opted to deploy ontological models to generate adaptive user interfaces. Skillen et al. [20] proposed an ontology-based method for context-aware application personalization within mobile environments. Castillejo et al. [21] also contributed with an ontology-based model for supporting dynamic and adaptive user interfaces. The Egoki system [22] employed a user ontology model to automatically generate tailored interfaces using XML markup. However, most of these proposals are limited to ontological models that represent user information that could be applied in the adaptation process. In fact, they barely offer interface adaptation during user interactions. In this sense, Tsiporkova et al. [23] provided a semantic framework that supports adaptive multimodal interactions both at design and runtime by means of linking pattern specification and user modelling. The proposal of Kultsova et al. [24] was also addressed to highlight the role of ontological models, thereby guiding the use of interaction design patterns, but these were only triggered when users restarted the application that deployed them. In summary, there are few successful approaches combining design pattern-based methods and the adaptation capabilities of semantic models. The current work intends to overcome such limitations

by proposing a framework that joins together an ontological model of UI design patterns with the ontological representation of user profiles and the context surrounding them.

3. Framework Proposal

The current work presents a framework called AUIDP (Adaptive User Interface Design Patterns) that combines ontology-based models with UI design pattern methods for the generation of adaptive mobile applications.

3.1. Framework Fundamentals

The proposed framework is based on two main semantic models, first, a design pattern model that represents the best practices of UI development for users with specific requirements and secondly, a user profile model that comprises knowledge about users' information, such as their preferences, goals, or special needs. Most of the selected design patterns were proposed by Peissner et al. [6] or gathered from [25]. For describing UI design patterns, a well-known structure was adopted, in which every design pattern was identified with a name; this structure is composed of three principal elements: the context, problem, and solution. Table 1 outlines the structure of these patterns along with an example. After identifying an initial set of UI design patterns with a textual description, we elaborated an ontological model (see Figure 1) that represents the general knowledge of the selected design patterns, their corresponding items and features, as well as the condition of their use related to the user's restrictions and needs. Figure 1 also illustrates the relationships between the pattern items and the design solutions carried out for UI adaptation. Table 2 identifies the main ontological concepts (classes) that characterize the selected design patterns. The design pattern class denotes the main concept within this ontology. This class is divided into two sub-classes related to its category. The first category concerns patterns related to generic UI features, while the second category represents UI interaction patterns.

Table 1. UI design pattern structures.

Attribute	Description	Example
Name	The name of the design pattern	[Gallery DP]
Problem	The problem detected when the user interacts with an application	Users can have difficulties understanding text instructions
Condition	The assumption about how the selected design pattern can be used	The number of images has to fit the user's needs without overloading them
Solution	The solution designed to solve a problem	Image such as icons or pictures can support better user understanding

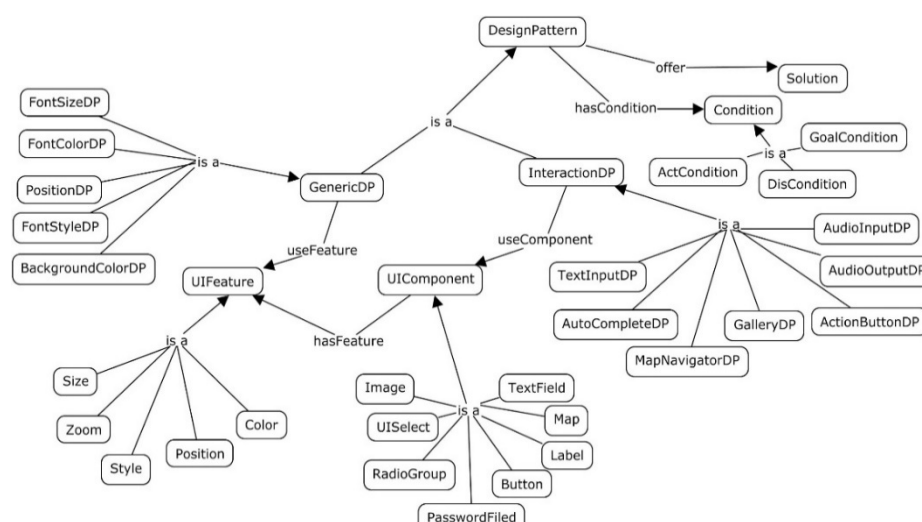


Figure 1. UI design pattern ontological model.

Table 2. Description of UI design pattern ontological concepts.

Concept	Description
DesignPattern	A kind of design solution that facilitates the development of user interfaces.
GenericDP	A category of design pattern that defines UI settings.
InteractionDP	A category of design pattern that defines UI components.
UIComponent	An interface element that may compose the UI.
UIFeature	A global setting that may encompass the UI variables.
Condition	A trigger for applying a design pattern. It can be related to the activity of a user (ActCondition), to the goal of a user (GoalCondition), or the disability of a user (DisCondition).

Figure 2 shows the user profile model, representing its semantics by means of an ontological model that describes the knowledge of users' profiles. In particular, their interests, goals, activities, current context circumstances, or disability features are included in this semantic model. A description of the main ontological concepts and relationships are reported in Table 3, which displays general information about them.

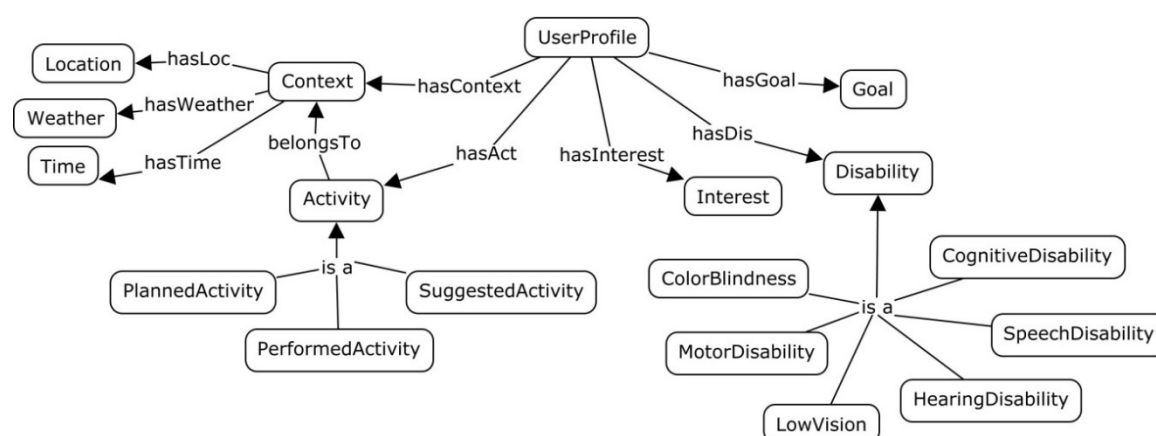


Figure 2. User Profile ontological model.

Table 3. Description of user profile ontological concepts.

Concept	Description
User	A user can be elderly or disabled (for which the application is intended).
Activity	An activity can be a suggested pursuit, a planned task, or a performed action.
Goal	A goal that can be achieved by a user, e.g., entertainment.
Interest	A specific location or service that a user may find interesting.
Context	A context is the environment surrounding a user, which is composed of location, weather, or time.
Disability	A disability is an impairment that can affect a user.

3.2. Framework Architecture

The AUIDP framework aims at using previously defined ontological concepts and associated rules to generate user interfaces according to specific circumstances and selected design patterns. It is based on a client–server architecture, which is divided into three components: (i) an expert knowledge base, (ii) an automatic UI design pattern selection engine, and (iii) a UI adaptation and generation engine. These components interact to generate user interfaces adapted to user needs and the contextual circumstances surrounding them, as illustrated in Figure 3.

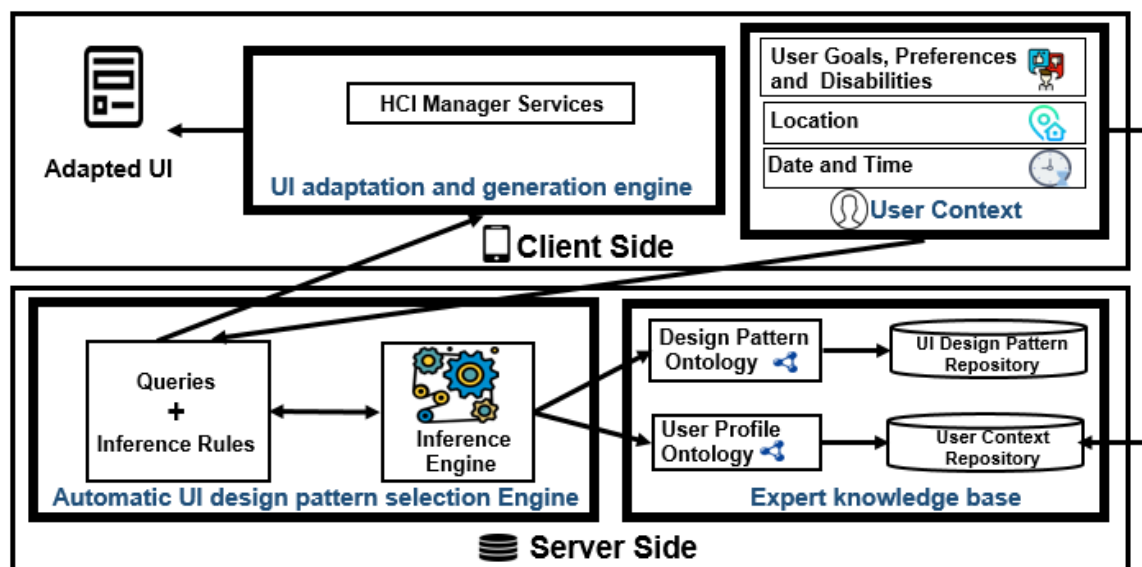


Figure 3. Framework architecture.

As stated before, the proposed framework incorporates ontological models for UI design patterns and the user profile. These models and the corresponding repository are accessed through the framework's expert knowledge base. The Automatic UI design pattern selection engine takes the user's current context as its input and then automatically selects a more suitable UI design pattern after querying the ontological models that are contained in the expert knowledge base. When this component tries to select such design patterns, it applies rules by making use of the rule-based reasoning component. The UI design patterns derived from this engine are divided into two main categories: generic patterns and interaction patterns:

- Generic patterns provide global UI setting items, such as increasing contrast, changing colors for color-blind users, or maximizing the font size for users with low vision capabilities.
- Interaction patterns specify the UI items needed in every interaction situation, such as applying an audio input modality for users with motor disabilities or selecting icons to schedule a daily activity event.

Each category fulfills a specific task in the UI adaptation process. Thus, selecting the right UI design pattern for each context of use is one of the framework's main contributions. It has, as well, a crucial impact on the efficiency of the UI adaptation process. Consequently, this component provides a real time reasoning module to meet timing requirements. It uses a set of rules in combination with the UI design pattern ontology and the user profile ontology to decide which appropriate UI design pattern should be applied, as well as its corresponding UI items and features. Figure 4 shows the rule schema that has two parts: first, a condition that controls the circumstances under which the rule can be triggered and, second, an action part that represents the resulting decision about the selected design pattern when the condition is checked.

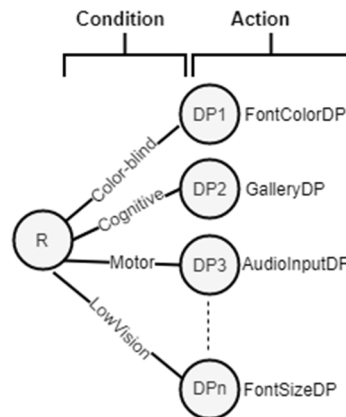


Figure 4. Partial list of rules for selecting UI design patterns.

The UI adaptation and generation engine component is in charge of generating adaptive interfaces that fit users' needs. It takes as its input a set of UI design patterns that are retrieved from the UI design pattern selection engine and, then, adapts them according to the observed needs. Figure 5 shows a sequence diagram displaying the UI adaptation process within the AUIDP framework. Initially, an interaction event is produced between the user and the interface. This event triggers the adaptation engine when the user's context changes. Afterward, when the UI design pattern selection engine receives a request from the adaptation engine to obtain relevant patterns, it calls the inference engine. Thus, the inference engine is able to access and query the expert knowledge base that includes design pattern ontology and user profile ontology and their corresponding repositories. After receiving the inference results, the UI design pattern selection engine sends the relevant design patterns to the adaptation engine. This engine adapts the UI components and features that compose the user interface. The results of such adapted UI components and features are displayed as a final interface for the user.

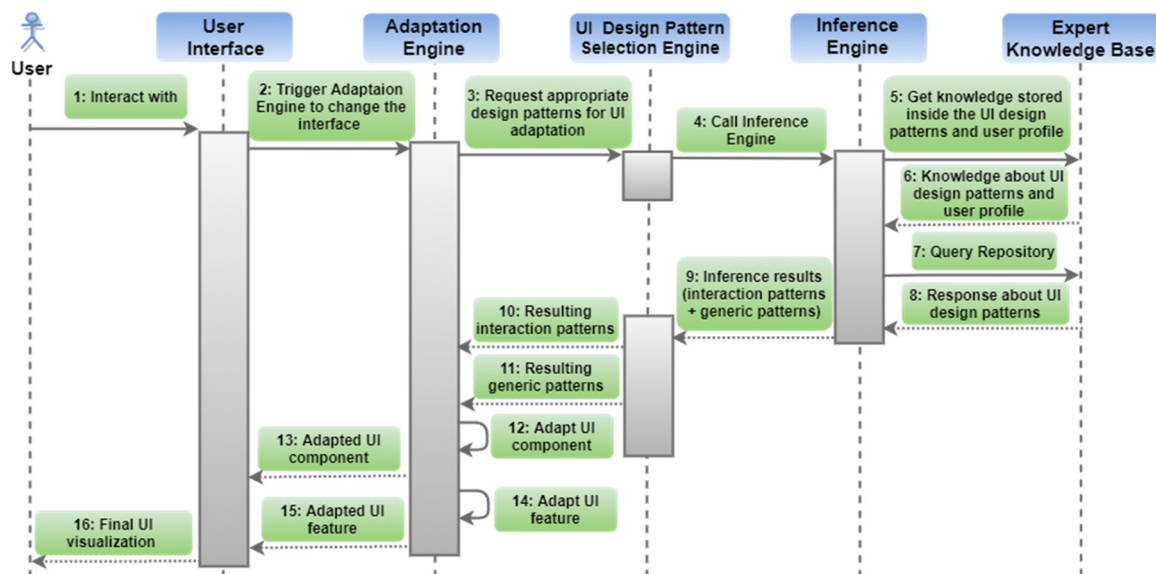


Figure 5. Sequence diagram representing the UI adaptation process.

4. Application of the Framework

In order to bring our research closer to practice, an application of the AUIDP framework is described in this section. Hence, we have developed a prototype for a mobile application called APA (Adaptive Personal Assistant), which is a location-based/assistant app for elderly people or people with special needs. It is able to track users' current locations and weather conditions around them,

locate points of interest for these users, notify them and display nearby restaurants at lunch or dinner time, and provide a list or an agenda of planned events. First, a short review of the prototype's implementation is presented. A second stage outlines the ontological models and rules used to support the expert knowledge base of the framework in this particular case. Then, a case study is introduced with a description of several user scenarios, which illustrate the use of various UI design patterns and highlight the adaptation of user interfaces to changing user requirements.

4.1. Implementation Features

The APA prototype was implemented by using three main technology platforms (i) the IONIC mobile application development tool, (ii) Node.js (as a JavaScript run-time environment for server-side scripting), and (iii) Apache Tomcat as a server for RESTful Web services. Moreover, additional tools and libraries were used to process the ontology models addressed in this framework:

- Protégé as an editor to create ontology models.
- Apache Jena Fuseki for handling OWL ontologies, reasoning with OWL data sources, and processing SPARQL queries.
- The REST Country API to gather general information about countries where the user can stay.
- Zomato API to retrieve restaurant or food shop information.
- Leaflet for displaying an interactive map that allows the user to situate a location.
- DBpedia as an external ontology to retrieve information about specific places to be visited by users.

4.2. Ontology Models and Rules

The starting point when using the AUIDP framework is the development of the two ontological models (the UI design pattern and the user profile model), based, respectively, on the descriptive patterns and user's characterization. These ontological models were created using the Protégé editor and implemented in OWL notation. Figure 6a highlights an excerpt of the first ontological model displaying the main UI design pattern ontology class. Similarly, Figure 6b shows the class hierarchy related to the user model. Once these models are defined, a set of specific rules can be provided to ensure the generation of the final UI display. Then, an example of rules is presented according to users with special needs, as well as the adaptive interface results of applying such reasoning rules.

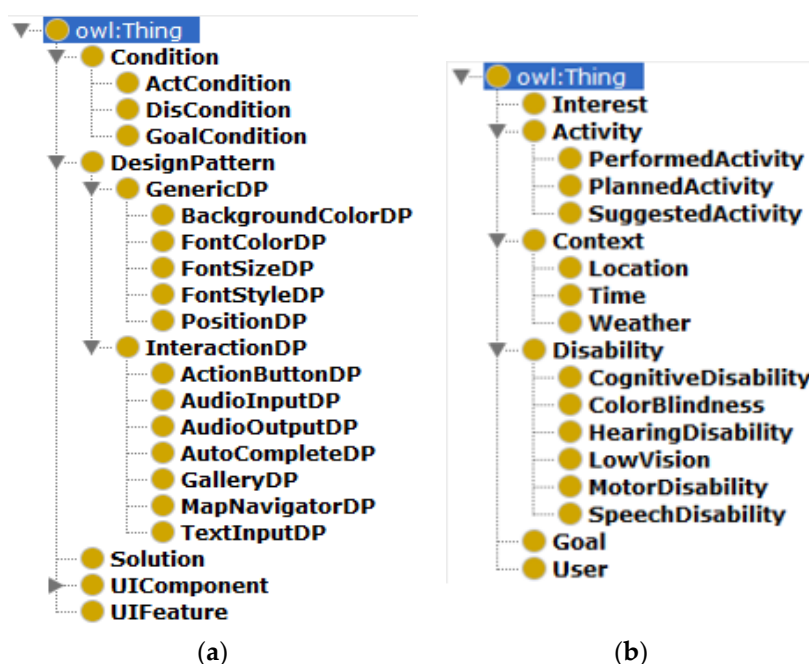


Figure 6. Ontological models representing the UI design pattern and the user profile. (a) UI design pattern, (b) user profile.

In the rule example displayed in Figure 7, we assume that the user has a low vision disability and needs to perform the following task: read information about his points of interest. In this case, the reasoning component is invoked to trigger the right rule (“rule1”) for selecting the required UI design patterns. Applying this “rule 1” to the ontology model of Figure 6, the right part would be indicated as a “result1”, to which a design pattern can be applied with the right level of map zoom.

<p>[rule1:</p> <pre>(?Goal rdf:type uni:GoalCondition) (?Goal uni:conditionName 'viewInterest') (?Dis rdf:type uni:DisCondition) (?Dis uni:conditionName 'lowVisionSevere') (?DP1 rdf:type uni:FontSizeDP) (?DP1 uni:DPName 'fontSizeMaxDP') (?sol rdf:type uni:Solution) (?feature1 rdf:type uni:UIFeature) (?component1 rdf:type uni:Map) (?feature1 uni:isAssociatedTo ?component1) (?DP1 uni:useFeature ?feature1) -> (?sol uni:applyDP ?DP1) (?feature1 uni:featureValue '14')</pre> <p>]</p>	<p>result1: [</p> <pre>{ "Goal": " viewInterest", "Dis": " lowVisionSevere", "DPName": "fontSizeMaxDP", "Solution": "Increase the zoom of the map", "UIComponentName": "map", "UIFeatureName": "zoomIn", "UIFeatureValue": "14" }</pre> <p>]</p>
--	--

Figure 7. Example of adaptation rule.

4.3. Use-case Scenario

In order to check the suitability of the proposed framework, a case study is presented to demonstrate how the APA prototype can deal with different user case scenarios. Figure 8a shows the first APA screenshot, which allows users to sign in or sign up in the application by using different alternative methods. Figure 8b displays the “Welcome” screenshot that introduces some of the main features. To highlight the adaptation of various UI items to the user profile changes, several application scenarios are considered and described in the following paragraphs.

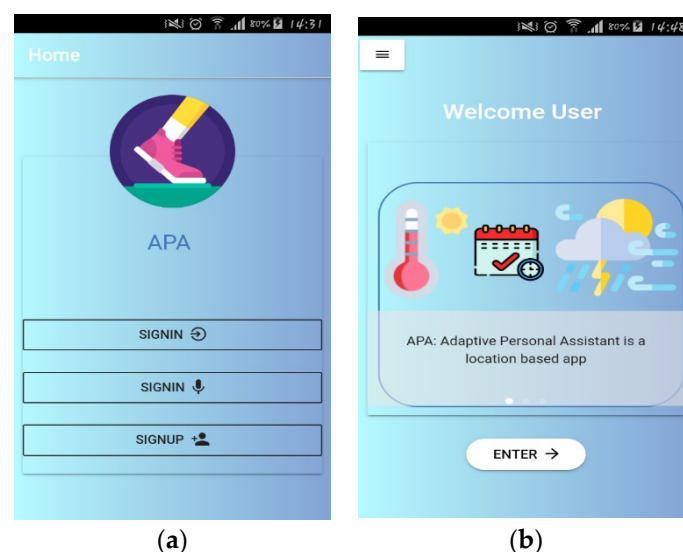


Figure 8. APA prototype screenshots. (a) home landing page, (b) welcome page.

The first scenario defines a case in which a color-blind user (Anna) is interested in visiting two points of interest (for instance, a museum and a hotel). After signing up and entering her specific information and interests, Anna connects to the application by typing her username and password. While connecting to APA, the user interfaces display items that are adapted, for example, to deciding which color or contrast display level can be selected in order to assure a suitable user experience. Once the UI display is generated, Anna can access interfaces with dark contrast and black and white images. Figure 9 illustrates two UI items corresponding to a welcome interface and search place interface generated by the current framework for this scenario.

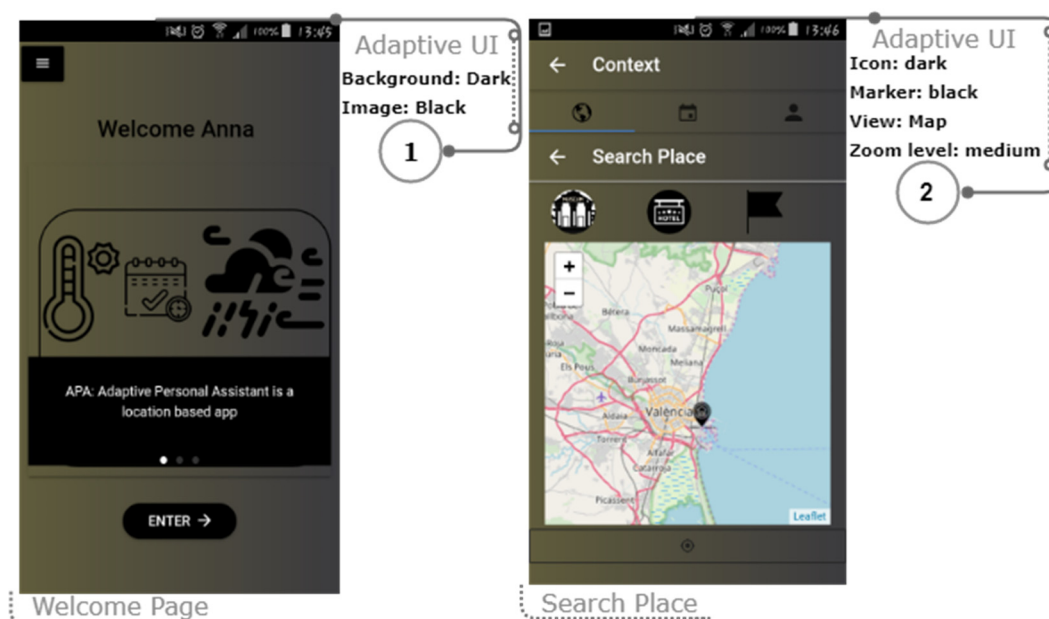


Figure 9. Scenario case adapted to color-blind users.

In the second scenario, a user named Alex with a low vision level is considered. Alex can use APA to get his current location, locate a set of points of interest (e.g., a hotel, museum, or airport), as well as schedule an event. Once connected, adaptation rules are carried out to deploy icons with the right size, thereby allowing users like Alex to schedule an event that fits their interests in a comfortable way. Figure 10 shows the UI items generated in this second scenario.

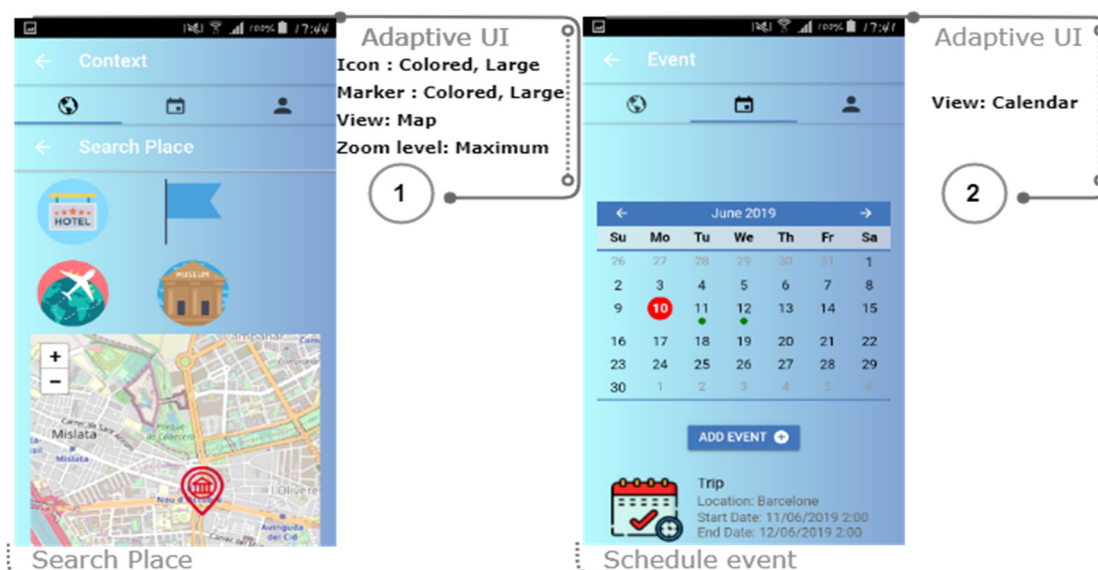


Figure 10. Scenario case adapted to users with low vision levels.

5. Evaluation Results

The evaluation of the AUDIP framework was based on testing the usability of mobile adaptive applications that could be generated under the principles of this framework. In this case, the APA application presented in the Case Study section was evaluated using three of the usability dimensions proposed by Nielsen [26]: learnability, which refers to how well users can work with the APA, satisfaction, by asking users how pleasant their experience with the application is, and efficiency, which is measured by the task completion time. To this end, two main instruments were used. Firstly, a questionnaire was conducted to assess the satisfaction and learnability dimensions. Secondly, a task to schedule events was proposed to measure the time required by users to complete it. The mentioned questionnaire was formulated using question samples extracted from two main sources: (i) the System Usability Scale questionnaire (SUS) [27] and (ii) the Questionnaire for User Interaction Satisfaction (QUIS) [28]. Table 4 shows the list of questions, which were divided into two categories corresponding to the dimensions of Satisfaction (5 questions from Q1 to Q5) and Learnability (3 questions from Q6 to Q8). Each question could be answered using a scoring scale from 0 to 4 (0 = To a very small extent, 1 = To a small extent, 2 = Somewhat, 3 = To a large extent, and 4 = To a very large extent). Eight persons participated in the evaluation (aged from 21 to 54 years), each with different user profiles. The Cronbach's alpha coefficient value was computed to measure the reliability of the proposed questionnaire. In terms of the Satisfaction dimension, the value of Cronbach's alpha was 0.71 which can be considered acceptable. Concerning the Learnability dimension, the alpha Cronbach's coefficient was higher, with a value of 0.91, which is considered an excellent measure of the questionnaire's reliability. In general, answers to questions related to user satisfaction were scored with a medium–high value. The first question obtained a mean value of 2.75, indicating a high appreciation of the application's easiness. The readability of texts (Q2) reached the top score. Participants were also asked about the content color display (Q3) and the size of the interface components (Q4), with a good valuation of both questions. Otherwise, the matching between the provided user interfaces with their needs (Q5) received a lower score. Questions with the highest scores were located in the Learnability dimension (Q6–8). These scores confirmed that the application tasks could be performed in a straightforward manner, and the time to learn the application was estimated as short.

Table 4. Statistical results of the Mean, Standard Deviation, and Cronbach's alpha coefficients.

Usability Dimension	Question	Mean	SD	Cronb. Coeff.
Satisfaction	Q1 This application is easy to use	2.75	0.82	0.71
	Q2 The texts/characters on the interface are readable	4	0	
	Q3 The colors used in the application provide better content visualization/display	2.62	1.11	
	Q4 The size of the interface items can be considered right/suitable	2.87	0.78	
	Q5 The interface items fit with my current context or needs	2.37	1.11	
Learnability	Q6 The application is easy to learn	3.25	0.96	0.91
	Q7 Application tasks can be achieved in a straightforward manner	2.62	1.11	
	Q8 The time required for learning the application can be considered short	3.25	1.08	

The second evaluation instrument consisted of measuring the task completion time (Efficiency dimension). In this case, two different user interfaces for scheduling a certain event were proposed to the evaluation participants. Figure 11a shows a screenshot displaying a traditional interface to introduce event information by means of text forms, while the rest of the images in Figure 11 show screenshots that represent an adaptive user interface with a selection of event icons (Figure 11b), times, and dates displayed in graphical formats (Figure 11c), or the choice of location places (Figure 11d). The average of task completion time in the traditional UI was about 102 s. With the use of an adaptive UI, the average time lowered to 45 s, such that participants were able to complete their task in a shorter time, thereby reducing the required cognitive load.

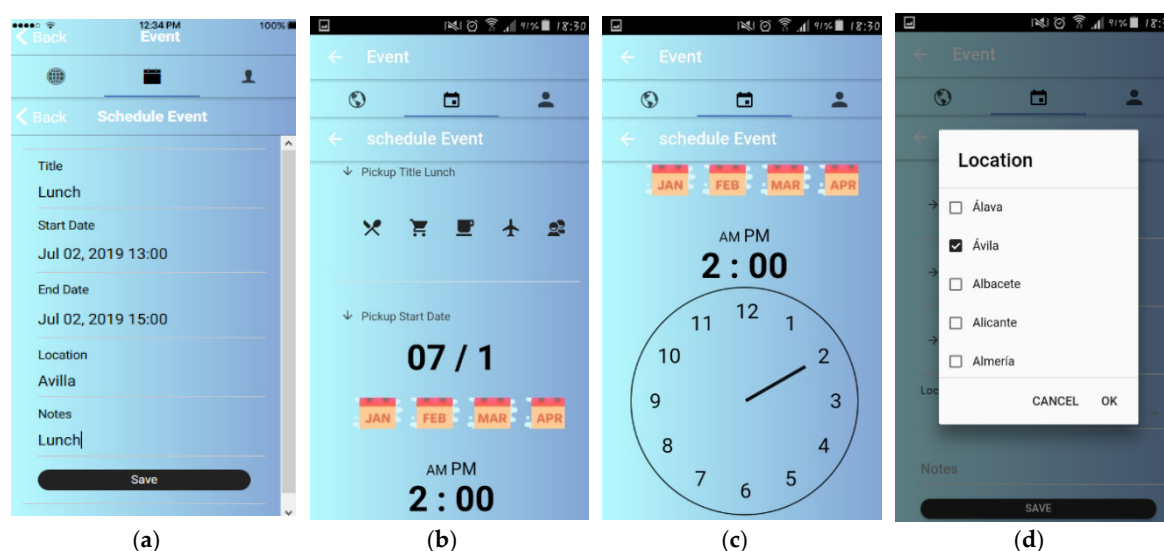


Figure 11. UI interfaces for event scheduling. (a) Traditional interface, (b) event setup, (c) time configuration, (d) location selection.

6. Conclusions

The current work has presented a framework called AUIDP (Adaptive User Interface Design Patterns) that combines ontology-based models with UI design pattern methods to support the generation of adaptive mobile applications. This framework allows a dynamic selection of relevant UI design patterns according to user requirements. A case study of the proposed framework has been provided in the form of a mobile app called APA (Assistant Personal App), whose adaptability features have been tested in several scenarios. Two of these scenarios have been shown in this work, demonstrating the framework's versatility to deal with different user profiles. Moreover, an evaluation of its usability has been performed by measuring some aspects related to satisfaction, learnability, and efficiency dimensions. Future work will focus on improving usability features by gathering new collections of UI design patterns and extending their ontological models. In addition, the technical feasibility of the proposed framework will be tested by collecting developer feedback.

References

1. Lim, S.L.; Bentley, P.J.; Kanakam, N.; Ishikawa, F.; Honiden, S. Investigating country differences in mobile app user behavior and challenges for software engineering. *IEEE Trans. Softw. Eng.* **2015**, *41*, 40–64.
2. Peltonen, E.; Lagerspetz, E.; Hamberg, J.; Mehrotra, A.; Musolesi, M.; Nurmi, P.; Tarkoma, S. The hidden image of mobile apps: Geographic, demographic, and cultural factors in mobile usage. In Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services, Barcelona, Spain, 3–6 September 2018; p. 10.
3. Bhandari, U.; Chua, W.Y.; Neben, T.; Chang, K. Cognitive Load and Attention for Mobile Applications: A Design Perspective. In Proceedings of the International Conference on Human-Computer Interaction, Toronto, ON, Canada, 17–22 July 2016; pp. 278–284.
4. Schneider-Hufschmidt, M.; Malinowski, U.; Kuhme, T. *Adaptive User Interfaces: Principles and Practice*; Elsevier Science Inc.: New York, NY, USA, 1993.
5. Nivethika, M.; Vithiya, I.; Anantharshika, S.; Deegalla, S. Personalized and adaptive user interface framework for mobile application. In Proceedings of the International Conference on Advances in Computing, Communications and Informatics (ICACCI), Mysore, India, 22–25 August 2013; pp. 1913–1918.
6. Peissner, M.; Häbe, D.; Janssen, D.; Sellner, T. MyUI: Generating accessible user interfaces from multimodal design patterns. In Proceedings of the 4th ACM SIGCHI symposium on Engineering Interactive Computing Systems, Copenhagen, Denmark, 25–26 June 2012; pp. 81–90.

7. Villalonga, C.; Strohbach, M.; Snoeck, N.; Sutterer, M.; Belaunde, M.; Kovacs, E.; Droegehorn, O. Mobile ontology: Towards a standardized semantic model for the mobile domain. In Proceedings of the International Conference on Service-Oriented Computing, Vienna, Austria, 17–20 September 2007; pp. 248–257.
8. Poveda-Villalón, M.; Suárez-Figueroa, M.C.; García-Castro, R.; Gómez-Pérez, A. A context ontology for mobile environments. In Proceedings of the Workshop on Context, Information and Ontologies (CIAO), Lisbon, Portugal, 11 October 2010.
9. Mazza, D. Reducing Cognitive Load and Supporting Memory in Visual Design for HCI. In Proceedings of the CHI Conference Extended Abstracts on Human Factors in Computing Systems, Denver, CO, USA, 6–11 May 2017; pp. 142–147.
10. Hussain, A.; Mkpojiogu, E.O. *Usability Evaluation Techniques in Mobile Commerce Applications: A Systematic Review*; AIP Publishing: New York, NY, USA, 2016; Volume 1761, p. 20049.
11. Zahra, F.; Hussain, A.; Mohd, H. *Usability Factors of Mobile Health Application for Chronic Diseases*; AIP Publishing: New York, NY, USA, 2016; Volume 1761, p. 20108.
12. Kalimullah, K.; Sushmitha, D. Influence of design elements in mobile applications on user experience of elderly people. *Procedia Comput. Sci.* **2017**, *113*, 352–359.
13. Petrovčič, A.; Rogelj, A.; Dolničar, V. Smart but not adapted enough: Heuristic evaluation of smartphone launchers with an adapted interface and assistive technologies for older adults. *Comput. Hum. Behav.* **2018**, *79*, 123–136.
14. Coram, T.; Lee, J. Experiences—A pattern language for user interface design. In Proceedings of the Joint Pattern Languages of Programs Conferences PLOP, Allerton Park, IL, USA, 4–6 September 1996.
15. Ballard, B. *Designing the Mobile User Experience*; John Wiley & Sons: New York, NY, USA, 2007.
16. Nilsson, E.G. Design patterns for user interface for mobile applications. *Adv. Eng. Softw.* **2009**, *40*, 1318–1328.
17. Neil, T. *Mobile Design Pattern Gallery: UI Patterns for Smartphone Apps*; O'Reilly Media Inc.: Sebastopol, CA, USA, 2014.
18. Seffah, A. *Patterns of HCI Design and HCI Design of Patterns: Bridging HCI Design and Model-Driven Software Engineering*; Springer: New York, NY, USA, 2015.
19. Engel, J.; Martin, C.; Forbrig, P. A Concerted Model-driven and Pattern-based Framework for Developing User Interfaces of Interactive Ubiquitous Applications. In Proceedings of the Workshop on Large-Scale and Model-Based Interactive Systems: Approaches and Challenges, Duisburg, Germany, 23–26 June 2015; pp. 35–41.
20. Skillen, K.L.; Chen, L.; Nugent, C.D.; Donnelly, M.P.; Burns, W.; Solheim, I. Ontological user profile modeling for context-aware application personalization. In Proceedings of the International Conference on Ubiquitous Computing and Ambient Intelligence, Vitoria-Gasteiz, Spain, 3–5 December 2012; pp. 261–268.
21. Castillejo, E.; Almeida, A.; Lopez-De-Ipina, D. Ontology-Based Model for Supporting Dynamic and Adaptive User Interfaces. *Int. J. Hum.-Comput. Interact.* **2014**, *30*, 771–786.
22. Gamecho, B.; Miñón, R.; Aizpurua, A.; Cearreta, I.; Arrue, M.; Garay-Vitoria, N.; Abascal, J. Automatic Generation of Tailored Accessible User Interfaces for Ubiquitous Services. *IEEE Trans. Hum.-Mach. Syst.* **2015**, *45*, 1–12.
23. Tsiporkova, E.; Hristoskova, A.; Tourwé, T.; Stevens, T. Semantic Modelling in Support of Adaptive Multimodal Interface Design. In Proceedings of the IFIP Conference on Human-Computer Interaction, Cape Town, South Africa, 2–6 September 2013; pp. 627–634.
24. Kultsova, M.; Potselico, A.; Anikin, A.; Usov, O. An Ontological Representation of Interface Patterns in Context of Interface Adaptation for Users with Special Needs. In Proceedings of the 9th International Conference on Information, Intelligence, Systems and Applications (IISA), Zakynthos, Greece, 23–25 July 2018; pp. 1–5.
25. Punchoojit, L.; Hongwarittorn, N. Usability studies on mobile user interface design patterns: A systematic literature review. *Adv. Hum.-Comput. Interact.* **2017**, *2017*, doi:10.1155/2017/6787504.
26. Nielsen, J. The usability engineering life cycle. *Computer* **1992**, *25*, 12–22.
27. Brooke, J. SUS-A quick and dirty usability scale. *Usability Eval. Ind.* **1996**, *189*, 4–7.
28. Norman, K.L.; Shneiderman, B.; Harper, B.; Slaughter, L. *Questionnaire for User Interaction Satisfaction*; University of Maryland: College Park, MD, USA, 1998.



