Voice-Based Virtual Assistants for User Interaction Modeling

Marco Brambilla(✉) and Davide Molinelli

Dipartimento di Elettronica, Informazione e Bioingegneria,
Politecnico di Milano, Milano, Italy
{marco.brambilla,davide.molinelli}@polimi.it

Abstract. In this work, we propose a virtual assistant that allows building models by means of voice commands. To demonstrate the generality of the approach, we describe three alternative strategies that apply voice-based support at three levels of detail: a fully-guided strategy; a pattern-based strategy; and an element-based strategy. We describe our implementation experience with the development of a design assistant covering the three strategies described above for OMG's IFML (Interaction Flow Modeling Language), in the context of user interaction design, including the integration with the Amazon Alexa assistant. We report our results that show how the assistant can bring advantages in terms of productivity.

1 Introduction

The design of the user interaction model represents one of the most delicate steps in the whole process of building and implementing a software system [2]. The objective of the work is focused towards the introduction of a form of artificial intelligence that could transform a passive support into an active assistance, able to communicate and guide the designer, understanding his commands, providing advice and building models automatically [4] . Other works aimed at this objective providing vocal assistant approaches for software designers [6,7].

In this paper we propose an approach and implementation of **a voice-based virtual assistant for model-driven development of user interactions and user interfaces**. While our solution is general and independent from the modeling language, to demonstrate the feasibility and advantages of the approach, the paper describes an implementation upon software models specified with IFML, the Interaction Flow Modeling Language [3,5]. The paper also reports our preliminary results that show how the assistant can bring advantages in terms of productivity.

2 Voice-Based Modeling Assistant

In the context of Model-Driven Development, a voice assistant can support designers with a different level of expertise, in a non-invasive manner, with the

M. Brambilla et al. (Eds.): ICWE 2021, LNCS 12706, pp. 530–533, 2021.
https://doi.org/10.1007/978-3-030-74296-6_45

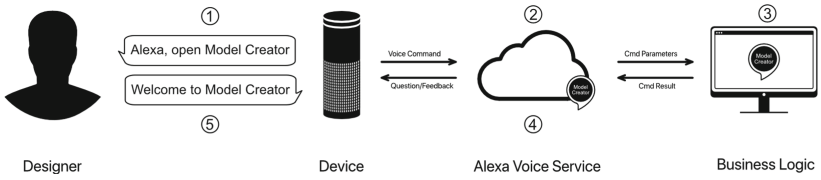

Fig. 1. Overview of the voice assistant communication flow.

purpose of improving productivity and also the quality of the resulting models. Since modeling could be applied at different phases of the development and by different designers profiles, an assistant should be flexible and able to adapt to the trade-off between speed and precision in the design. The voice assistant can follow or lead the design process by interacting vocally with the designer, asking questions, reacting to his answers, and taking different decisions on the subsequent steps to follow in a dialogue based on these answers.

In order to offer different levels of support to the designer, we propose three different levels of assistance: a *fully-guided requirement-based strategy*, that produces complete designs starting from requirements, without the need of looking into the modeling at all; a *pattern-based strategy*, that lets designers specify solutions by selecting patterns and combining them together, thus obtaining quick and optimal designs; and an *element-based strategy*, that lets designers specify precise modeling structures with fine-grained granularity.

In this research we implemented *Model Creator*, an *Amazon Alexa* application (called *skill* in Alexa parlance) developed using the Alexa Developer Console, a suite made available by Amazon to define the *intents* of applications that make use of the voice assistant to execute tasks.

The implementation goes through a multi-step dialogue (see Fig. 2) and it progressively acquires all the information about the context and requirements, guiding the designer through a set of alternatives, so as to reach a point where a complete model of the application can be automatically generated (by exploiting also the template-based approach when needed).

The skill aims at supporting the design of IFML models, and integrates with *IFMLEdit.org*, an open-source, online framework for the specification of IFML models and the generation of code for web and mobile applications [1].

An intent specified by the user during the design process can be represented by a single interaction or a longer dialogue composed by multiple requests and answers. In the first case, the designer asks for the execution of a simple command, while, in the second case, the request is complex and the voice assistant demands the progressive acquisition of information. The reason is that any intent can require the fulfillment of mandatory and optional parameters. When the number of parameters is high, it is difficult to provide all the required values, by means of a single sentence. Moreover, some parameters may become mandatory, depending on the value assumed by other parameters. Therefore, the interaction is divided into multiple steps.

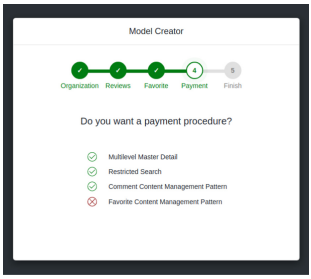

Fig. 2. In the guided mode, the voice assistant acquires the specifications through a multi-turn dialog before generating a complete model of the corresponding application.

The assistant is implemented as an Alexa skill that realizes a new voice-based front-end for the modeling editor. A communication flow between the skill and the modeling editor enables the voice-driven design process. This flow consists of the following steps, also depicted in Fig. 1: (1) The designer wakes up Alexa; (2) Alexa collects the voice stream containing the user request and sends it to the Alexa Voice Service that recognizes the command; (3) The skill is launched; (4) The modeling tool communicates back the result to the Alexa Cloud Service, that, in turn, elaborates the voice answer stream and sends it back to Alexa; (5) Alexa uses the speakers to communicate a feedback on the executed command and awaits the formulation of new commands or answers. For each new command (aka. intent) formulated by the designer, the process is repeated: the voice stream is sent to the Alexa Voice Service that identifies the intent and forwards the request to the modeling tool, which in turn executes the required operations on the model editor, and provides the respective feedback.

Some examples of usage of the assistant have been recorded in a video that is available online at: https://www.youtube.com/watch?v=00HoMz9Tq0A. For instance, the *Fully-guided Requirement-based Assistant* covers some typical business requirements and based on the decision of the designer at the requirement level we obtain a complete model of the application. The application scenarios covered in the current prototype include: e-commerce platforms, social network applications, blogs, and crowdsourcing systems. During the interaction process, the voice assistant initially asks for the scenario of interest and consequently changes the formulation of the next questions, starting a dialogue towards the ultimate intent of the user. At the end of the interaction, the assistant generates the complete and configured model that best fits the user's specifications. The progresses of the multi-turn dialog are visually managed and kept under control by means of a wizard panel (Fig. 2).

Table 1. Operations necessary to develop models with different assistants.

	No assistant or Element-based assistant					Pattern-based assistant (including configuration)						Fully-guided assistant (including configuration)					
	I	C	R	B	Total	I	C	R	B	Total	Saving	I	C	R	B	Total	Saving
E-commerce	200	97	927	219	1443	33	30	249	50	362	74.91%	1	0	249	0	250	82.67%
Blog	142	70	632	127	971	24	19	170	23	236	75.69%	1	0	170	0	171	82.38%
Social network	124	64	546	113	847	41	26	227	37	331	60.92%	1	0	227	0	228	73.08%
Crowdsourcing	107	49	478	101	735	22	10	139	6	177	75.91%	1	0	139	0	140	80.95%
Mean	-	-	-	-	-	-	-	-	-	-	71.85%	-	-	-	-	-	79.77%

3 Evaluation

We evaluated our approach by asking some designers to use the assistant in different configurations and application scenarios: an e-commerce application, a social network, a blog, and a crowdsourcing platform. Developers performed various kinds of operations:*Insertions (I)*, *Connections (C)*, *Refinements (R)*, and *Bindings (B)*, i.e., links between IFML elements. Table 1 reports the total number of operations needed to build the model with the corresponding type of assistant, and the percentage of operations saved using the *Pattern-based support* and the *Fully Guided support* with respect to the basic *Element-based support*. On average, the *Pattern-based support* allows to save around 71% of operations, while the *Fully-guided support* can reach 79% of saving.

References

1. Bernaschina, C., Comai, S., Fraternali, P.: IFMLEdit.org: model driven rapid prototyping of mobile apps. In: 4th International Conference on Mobile Software Engineering and Systems, pp. 207–208. IEEE Press (2017)
2. Brambilla, M., Cabot, J., Wimmer, M.: Model-Driven Software Engineering in Practice. 2nd ed. Morgan & Claypool, San Rafael (2017)
3. Brambilla, M., Fraternali, P.: Interaction Flow Modeling Language: Model-Driven UI Engineering of Web and Mobile Apps with IFML. Morgan Kaufmann, Burlington (2014)
4. Cabot, J., Clarisó, R., Brambilla, M., Gérard, S.: Cognifying model-driven software engineering. In: Seidl, M., Zschaler, S. (eds.) STAF 2017. LNCS, vol. 10748, pp. 154–160. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-74730-9_13
5. OMG: Interaction Flow Modeling Language (IFML), version 1.0. http://www.omg.org/spec/IFML/1.0.2015.
6. Soares, F., Araújo, J., Wanderley, F.: VoiceToModel: an approach to generate requirements models from speech recognition mechanisms. In: Proceedings of the 30th Annual ACM Symposium on Applied Computing, pp. 1350–1357. ACM (2015)
7. Stephan, M.: Towards a cognizant virtual software modeling assistant using model clones. In: 41st International Conference on Software Engineering: New Ideas and Emerging Results, pp. 21–24. IEEE Press (2019)