

Assisted Composition of Services on Mobile Devices

Nikolay Mehandjiev^(✉), Lu Ning, and Abdallah Namoun

University of Manchester, Manchester, UK
n.mehandjiev@mbs.ac.uk

Abstract. Composing software services on mobile devices is especially challenging when attempted by non-programmers. In this paper we compare two alternative supporting strategies: using generic task templates and scripting together condition-response fragments. The first is exemplified by a prototype called ACOM (Assisted Composition on Mobiles), the second by a commercially available alternative called IFTTT (IF This Then That). The paper uses a comparative observational study to highlight the benefits and drawbacks of both approaches, and to derive lessons for their improvement.

Keywords: Assisted service composition · Mobile mash-ups · Mobile EUD

1 Introduction

The increased sophistication of mobile devices leads to extended scope of users' interactions with them. Users even create “mashups” by connecting web services to provide combined functionality or information. Providing effective support for such activities on mobile devices is still a developing research topic, with only a handful of user-centric systems reported in the literature (notably Puzzle [6] and MobiMash [7]).

Desktops tools claiming to support “mashups” exist, yet these tools are difficult to learn by non-programmers, requiring the understanding of advanced programming concepts such as loops [1]. An exception is Yahoo Pipes¹, which employs “pipe and filter” metaphor for connecting components through information pipes, yet its models are not scalable, the type of information to be processed is limited and users experience difficulties in localising faults [2]. Converting these tools to mobile platforms is far from trivial, facing a number of platform-specific challenges. Success depends on choosing an appropriate strategy and ensuring effective representations.

This paper presents our initial steps in this direction, exploring the effectiveness of two alternative support strategies for mobile service composition by end users: using generic task templates and scripting condition-response fragments. The second strategy is implemented in the commercial tool IFTTT (IF This Then That) [5], which enjoys a growing user community. We have a tool supporting the first strategy, called Assisted User Composition (AUC) [3], yet this is designed for desktop platforms.

In this paper we report on the conceptual adaptation of our AUC approach to mobile platforms, called *ACOM: Assisted Composition on Mobiles*; and on the results of an observational study comparing our adapted approach with IFTTT.

¹ <http://pipes.yahoo.com/pipes/>

2 Assisted Composition on Mobiles (ACOM)

Our Assisted User Composition tool [3] uses a set of templates organised in a taxonomy. Each template corresponds to a key generic activity of our users [4] and comprises a number of tasks. Once a user selects a template, the tool uses the information encoded in the template to select a set of services for each task in the template. These are listed underneath the task name in a tabular format, and the user is expected to select a service for each task. Using the semantic information encoded in the templates and in the actual services, the tool indicates all services which are incompatible with the ones selected so far. Further details are available elsewhere [3].

Adapting the interface ideas of this tool to the features of a mobile platform was the first step in our investigation. We decided to deconstruct the tabular layout into a set of tabbed panes, one tabbed pane for each column of the table, and to hide the taxonomy of templates once the initial selection has been made. The adapted approach was prototyped using the JustInMind prototype tool², with the result shown on the left in Figure 1. Once the service selection is complete, the prototype lists all selected services on a single screen as shown to the right in the figure.

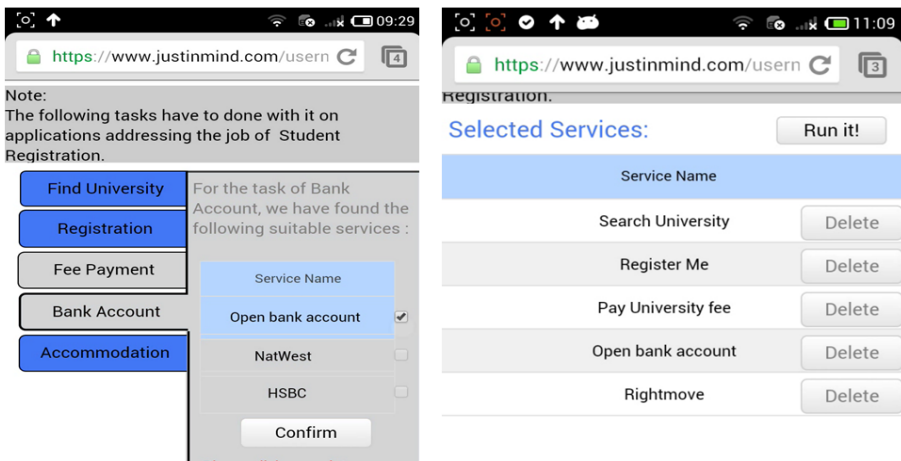


Fig. 1. Service Selection in ACOM (left) and list of selected services (right)

3 Experiments and Results

The aim of this study was to evaluate two alternative approaches to supporting end users when they try to compose services on a mobile device: using generic templates, exemplified by ACOM, and scripting together condition-response patterns, exemplified by IFTTT [5]. To that end we designed a within-subjects comparative observation study, where our participants were asked to complete two composition tasks of different complexity per tool. The task pairs across the tools were based on scenarios

² <http://www.justinmind.com/>

with similar complexity. Eight participants with average age of 23.4 years were selected for this experiment (four males and four females). All had low experience of similar tools, and only three of them had IT background. The participants were divided into two groups of four participants. Group 1 used ACOM first and then IFTTT, whilst Group 2, used IFTTT first and then ACOM. Participants alternated from different groups to reduce skewing due to learning by the moderator.

The experiment involved three stages as follows:

- (1) Training stage, where participants completed user information forms, attended a tutorial and then completed a pre-observation questionnaire.
- (2) Composition stage, where participants performed first a simple and then a complex composition task whilst verbalizing their thoughts. Voice and interactions were captured on the device using a screen recording tool.
- (3) Rating stage, where the participants rated the adaptability and usability of the composition tool, as well as end user development experience.

The second and third stages were repeated for each of the two tools.

After the experiments, data were analyzed using thematic analysis for the qualitative observations and statistical analysis for the quantitative data.

3.1 Comments and Initial Impressions

After the training stage, participants were asked to make comments and talk about their initial impressions on the two composition tools, ACOM and IFTTT.

Regarding ACOM, participants thought that it seems to be an efficient and useful composition tool, with a simple appearance of the composition. However, participants also noted that there is no instruction to describe what each service is doing.

With regards to the IFTTT, participants stated that it has a more elegant interface than ACOM and it is easy to operate. However, there are too much choices and the function is more complex than ACOM. Participants also indicated that after the training session they believed they are aware of how to compose services and understand the concepts of trigger and action.

3.2 Design Strategies

Participants were able to understand the two ACOM scenarios, and to follow the instructions for both to completion. During the composition session for the simple scenario, participants were able to understand the instructions and information flow, and quickly completed the task without particular problems. However, when users composed services according to the complex scenario, they were confused between the name of the overall scenario and the name of one the tasks within. The lack of detailed information about each service was reiterated, users pointing out that they could not know what each service is used for. They were also able to delete a specific service quickly and accurately after composing services and then run the composition tool. The services that a participant composed are shown in Figure 1. Notably, participants with IT background spend less time on using ACOM than those without.

Similarly for IFTTT, all participants completed the simple task successfully and in a short time. The information flow for the complex scenario was easy to understand and users could find appropriate services according to the hints or keywords of required services. However, half of the participants pointed out that it was a little difficult for them to find the right services for tasks because of the large amount of services in IFTTT. Therefore, users spend too much time on looking for services. Furthermore, participants were confused by similar services, such as Android SMS and SMS, some selecting inappropriate services. When deleting services, they were also not able to find the “delete” button in a short time.

3.3 Pre-observation and Post-observation Questionnaires

The comments and observations from the previous sections are reflected in the pre-observation and post-observation questionnaires completed by the 8 participants.

Table 1. Common questions and their codes, version for ACOM

Q1	It is difficult to understand the notations used in ACOM.
Q2	It is difficult to understand the instructions for ACOM.
Q3	The interface of ACOM is concise and simple.
Q4	The function of ACOM is practical and useful.
Q5	It is easy to use ACOM.
Q6	It is easy to navigate ACOM.
Q7	It is easy to find the right services using ACOM.
Q8	It is easy to delete services using ACOM.
Q9	I feel confident using ACOM.
Q10	I feel confused using ACOM.
Q11	The interface of ACOM fits the screen of mobile.
Q12	It is a difficult task to compose/aggregate services for me.
Q13	It is time consuming to develop assisted composition applications on mobiles.
Q14	I know which services and modules to combine in order to develop my composite application.

The questions listed in Table 1 are answered through a 7-point Likert scale, where 7 is “strongly agree” and 1 is “strongly disagree”. The scores for questions formulated in a negative manner, have been inverted to allow comparison with the positive questions. Figure 2 shows average response values for questions similar across both tools.

ACOM scored better than IFTTT on the majority of the questions. The ones which are statistically significant using a two-tailed t-test at 90% are Q5 (ease of using the tool) and Q6 (ease of navigation). Overall, the participants preferred the concise and simple interface, the ease of understanding instructions and the navigation of ACOM. Also, participants did not find the development of applications using ACOM to be time consuming nor difficult. Finally, participants demonstrated that they knew more about service composition after this experiment and they showed strong interest in service composition and in learning more about EUD in the future.

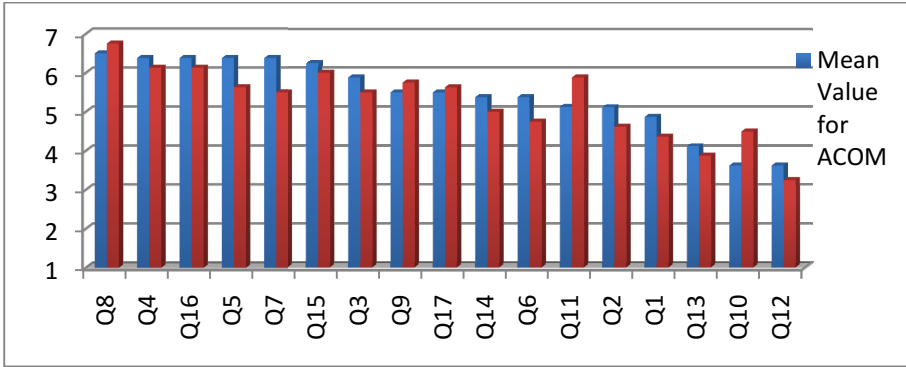


Fig. 2. Mean values of answers (7 is best) in decreasing order for ACOM

In addition to common questions, there were a few questions specific for one of the tools. The mean of the score for Q18ACOM (The application produced by ACOM is helpful to accomplish a student registration) and the mean of the score for Q22ACOM (ease of understanding the final list of services) for ACOM are nearly 7. It means that participants thought ACOM was helpful for the job of student registration and the final selected service list was also easy to understand. The mean value of Q18IFTTT (limitation of functions in IFTTT) is 4, which means participants’ opinion for this question is neutral. They thought the functions in IFTTT are sufficient for general use.

3.4 Post-observation Comments and Suggestions

After observation, participants completed a post-observation questionnaire, where they commented on features they like most and least, and provide suggestions for improvement in Table 2.

Table 2. Suggestions for Improvements

ACOM	IFTTT
Adapt UI for mobile devices, for example enlarge the buttons and checkboxes	Simplify the way applications appear to end users to avoid confusion
Improve the security of the system to protect the privacy of the users	Add history recording, especially for searches and results
Add more descriptions for each service on the screen	
Improve the aesthetic design of user interface	

Table 3 shows the results from the thematic analysis of participants’ comments on ACOM and IFTTT. In summary the comments relate to:

User Interface: Both interfaces were considered simple and elegant.

Notations: Participants expressed weak preference for the ACOM notation, finding both notations easy to understand but lacking details about the services

Navigations: Participants thought the two composition tools were easy to use and navigate without programming skills. Most participants thought that the navigation of IFTTT is better and easier than ACOM.

Functions: The functions of both tools were perceived as useful, yet ACOM had limited services and scope of use, whilst IFTTT had too many services, making the choice of service too slow. Besides, both tools were found lacking a good searching function. The number of comments suggests the functions in ACOM were better regarded than those in IFTTT.

Table 3. Times of mentioning aspects of the tools

Themes	ACOM	IFTTT
User Interface	7 (5 positive and 2 negative)	4 (4 positive)
Notations	2 (1 positive and 1 negative)	2 (2 negative)
Navigations	5 (3 positive and 2 negative)	6 (6 positive)
Functions	9 (7 positive and 2 negative)	12 (7 positive and 5 negative)

4 Summary and conclusions

The constraints of this study meant only eight participants were used, focusing analysis on qualitative comments. This initial feedback will be used to develop an operational ACOM prototype to be evaluated in a future study with more participants.

Despite its limitations, the current study achieved its objectives in comparing two alternative approaches to service composition on mobiles and highlighting requirements for further improvement. For example, clear information about each service would help users deal with a large amount of unfamiliar services. The findings are generally in line with our earlier work on mental models [3], e.g. confirming end user difficulties in following data dependencies and other dependencies between services. Other interesting requirements were the need for search history, the preference for larger icons and simpler service composition interfaces. The participants were able to perform the simple composition tasks without the help of the moderator, and provided an overall positive opinion about the ease of use of ACOM, and about the usefulness of such tool in the task which is supported by it.

References

1. Zang, N., Rosson, M.B., Nasser, V.: Mashups: who? what? why?. In: Proceedings of CHI, Florence, Italy (2008)
2. Jones, M., Churchil, E.: Conversations in developer communities: a preliminary analysis of the yahoo! pipes community. In: Proceedings of the 4th International Conference on Communities and Technologies, PA, USA (2009)
3. Mehandjiev, N., Lecue, F., Wajid, U., Namoun, A.: Assisted service composition for end users. In: Proceedings of ECOWS 2010, pp. 131-138. IEEE CS, Washington, DC (2010). doi=10.1109/ECOWS.2010.30

4. Sutcliffe, A.: *Domain Theory: Patterns for Knowledge and Software Reuse*. L. Erlbaum Assoc. Inc., Hillsdale (2002)
5. IFTTT ‘Put the internet to work for you. - IFTTT’ (2015). <http://ifttt.com> (accessed: March 13, 2015)
6. Danado, J., Paternò, F.: Puzzle: a visual-based environment for end user development in touch-based mobile phones. In: Winckler, M., Forbrig, P., Bernhaupt, R. (eds.) HCSE 2012. LNCS, vol. 7623, pp. 199–216. Springer, Heidelberg (2012)
7. Cappiello, C., Matera, M., Picozzi, M., Caio, A., Guevara, M.T.: Mobimash: end user development for mobile mashups. In: *Proceedings of the WWW 2012 Companion*, pp. 473–474. ACM, NY. doi=10.1145/2187980.2188083