Crowd-Sourcing Service Designs: Overview and Research Challenges

Nidhi Rajshree, Bikram Sengupta, and Nirmit V. Desai

IBM Research, Bangalore, India
{nidhi.rajshree,bsengupt,nirmit.desai}@in.ibm.com

Abstract. A service is supposed to embody the needs and interests of its providers as well as its consumers. However, service design has traditionally been the prerogative of service providers, often leading to services that provide unsatisfactory consumer experience. With increasing prevalence of open government initiatives and the advent of social computing in the enterprise, we posit that service design is set to become truly participatory, with the service provider tapping into the wisdom and creativity of the consumer "crowd", and the design of a service resulting from their collective ideation, brainstorming and refinement. This paper seeks to identify the research challenges in crowdsourcing service designs by way of proposing a new high-level framework and describing its application to an elaborate example of driver's license issuance service. The framework is a mix of the richness of crowd participation and the technical rigor afforded by formal analysis of service designs. The framework describes the components and how they come together, thereby leading us to the novel research challenges in realizing the components that should motivate further work on this topic.

1 Introduction

The goal of a service system is to create an experience that generates value for both the consumer and the provider of a service [13]. This suggests that service providers and consumers should collaborate closely to design services that serve their mutual interests. However, service systems have traditionally been designed by providers with limited participation from end users, often resulting in poor experiences when deployed. This is true for both public services as well as those delivered within an enterprise. Governments have found it difficult — or have lacked the motivation — to engage with diverse groups of citizens in a structured and constructive manner when designing public services. Even within an enterprise, hierarchical structures and centralized decision making have often meant that employees have had little influence on services they have to use. Services designed only from the perspective of the provider may ignore key consumer concerns and lead to unwieldy, overly-constrained processes that impose substantial burden on the end user instead of helping them.

Singh articulates the fundamental flaw in the current approaches of service designs and argues for the users and usages of a service to be playing a primary role in the design process [12]. Supporting this argument, service providers are increasingly accepting the necessity and the value of consumer involvement in the service design process. In many cases, governments are leading the way with open government initiatives, that seek to

A. Ghose et al. (Eds.): ICSOC 2012, LNCS 7759, pp. 203–214, 2013.
© Springer-Verlag Berlin Heidelberg 2013

enhance transparency, accessibility and responsiveness, often through increased citizen participation. This has motivated innovative research such as the COCKPIT project [5] under the EU FP7 program, that encourages citizen participation in the design of public services by offering multiple alternatives to them for their consideration and feedback. At the same time, we are also observing an increasingly young workforce instrumenting a deeper penetration of social computing within the enterprise[1], and rigid business processes are increasingly complemented with unstructured, team-based collaboration, particularly for complex decision making.

These trends suggest that we may be at the tipping point of *participatory service design*, where service providers seek to tap into the collective wisdom and creativity of the "crowd" of consumers, and the design process proceeds through their collective ideation, brainstorming, and refinement. "Crowdsourcing" [6] has emerged as a popular mechanism for fostering community participation, where placing a problem in the public domain along with an open call for responses and appropriate incentives, has sometimes yielded quick or effective solutions. For example, voting can be considered as an extremely simple form of crowdsourcing. The knowledge — the most popular choice — can be extracted trivially. In other crowdsourcing applications involving some form of computation or problem solving, rewards may be determined based on the volume of fine-grained tasks completed (e.g. tagging images), or by the selection of the "best" solution, from amongst the submitted responses, by the crowdsourcer (with a respondent only having visibility into his/her own solution) [8]. However, applying a standard crowdsourcing approach to the task of participatory service design throws up several challenges that we identify next. These challenges have also helped motivate a high-level framework for crowdsourcing service design outlined in this paper.

1.1 Challenges in Crowdsourcing Service Design

The design of a service as well as its evaluation are intellectually challenging tasks. A single design may have many dimensions to it and involve various trade-offs. Hence, the first challenge in crowdsourcing the design of a service is that it is non-trivial (and time consuming) to judge the overall quality or usefulness of any proposed solution.

Second, the scale of responses, e.g., for a public service impacting a large population, or even a business service within a large enterprise - may make it infeasible for the service provider to process and evaluate each response, determine which ones are valid as per the domain constraints, and then evaluate the "goodness" of the valid ones.

Third, we would like the design process to be truly participatory, involving iterations where respondents have visibility into each other's designs and can suggest enhancements to the same. This further exacerbates the problem of scale as new designs emerge across iterations. Also, the "best" design (by whatever evaluation criteria is used) may involve a creative composition of ideas from multiple designs, and it is infeasible for participants to manually generate all possible compositions and determine the best. Note that a "best" design must be adjudged so from the perspective of all the stake holders, including the end users. Hence, choice of what variant to execute cannot

[1] http://www.gartner.com/it/page.jsp?id=1470115

be made by end users of a service unilaterally, e.g., citizens in the driver's license case. This is why it is necessary to arrive at the single "best" design.

Finally, richer the collaboration among contributors, the harder it is to define incentives. Crowdsourcing depends on the participation from the crowd. From the perspective of rationality, there must be an incentive for the crowd to participate. If contributors collaborate closely, it is difficult to trace an idea to a contributor and assign credit. But unless credit is attributed to in a fair manner, the crowd may lose interest to participate.

1.2 Contributions

In this paper, we propose a high-level framework to support service design crowdsourcing, motivated by the above challenges. The novelty of the framework lies in its ability to tap into the richness of crowd participation, while suitably tempering it through formal reasoning and analysis to filter out invalid submissions, automatically generate new designs by including "good" aspects from multiple existing submissions, and help focus the attention of the crowd on the most promising design variants. The combination of crowdsourcing and formal reasoning ensures that we can handle scale without sacrificing on the originality of human thinking. It is beyond the scope of this paper to present a concrete realization of the framework - instead, we discuss the various challenges it brings up and possible ways of addressing them, with the goal of fostering further research into the area. We find that the main challenges are in (a) representing the high-level service designs and criteria for comparing alternative detailed designs, (b) generating new detailed designs and assimilating modifications to existing ones such that the new designs include "good" features of all contributed designs, and (c) incentivising contributors eventhough they collaborate closely and have visibility to each others' designs. A case study involving the collaborative design of a driving license issuance service is used to illustrate what the framework is envisioned to achieve without delving into how this can be achieved today.

The rest of the paper is organized as follows. Section 2 presents the case study on the public service of driving licenses. Section 3 describes our framework for crowdsourcing process models. Section 4 applies our framework to the case study of driving licenses and steps through the method for arriving at the *best* model for delivering the driving license service. Section 5 presents the research challenges in realizing our framework and suggests potential approaches. We summarize the related works in Section 6 and conclude in Section 7.

2 Case Study: Driving License Service

The Transport Department of India provides two kinds of services to its citizenry, namely, services related to motor vehicles and those related to individuals. The former includes services such as inspection of vehicles in the incident of an accident, transfer of vehicles from one owner to another, and issuance of fitness certificates to motor vehicles. The individual services include issuance of learner's license, granting of international driving permit and issuance of driver's license. Our case study is based on the Driver's License issuance service. The Motor Vehicle Act of 1988 and its subsequent

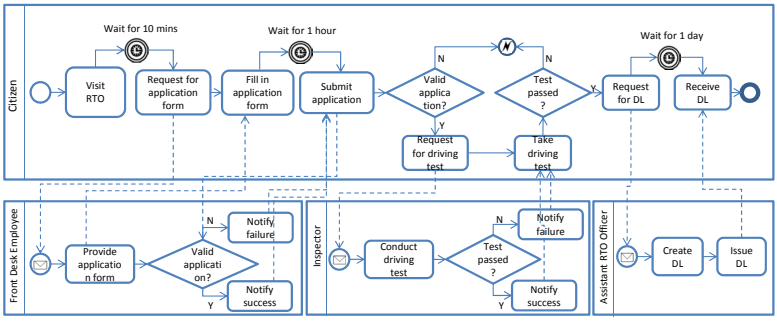

Fig. 1. Case study: Driving license services in India

amendments stipulate the national official licensing process. Individuals must first obtain a temporary license, which grants the right to practice driving under the supervision of a licensed individual.

In this paper, we study the process wherein after 30 days (and within 180 days) of the issuance of the temporary license, the individual may apply for a permanent license. Figure 1 depicts a BPMN process model for this process. Note that BPMN is chosen for illustration and is not a suggested representation for addressing the challenge of crowdsourcing service designs. The applicant must visit the Regional Transport Office (RTO), obtain and fill out an application form and get it validated by the front desk. The application requires supporting documents such as proof of age, proof of residence, a recent passport size photo, and his or her temporary license. If valid, a driving test with an inspector is scheduled and later conducted after a wait. If the test is passed, a driving license may be requested by the applicant and issued by the Assistant RTO. Either invalid application or a test failure results in the rejection of the driving license application. Although this service involves multiple stake holders, we focus on the applicant's perspective in the following discussions. Note that Figure 1 shows simply the current way of achieving the goal of issuing driver's licenses; there can be better ways and the citizens may have ideas for the same. We will revisit this in Section 4.

3 Proposed Framework

Motivated by the challenges identified in Section 1.1, we now propose a high-level framework to support crowdsourcing of service design as shown in Figure 2.

Our approach begins with the service designer (working on behalf of the service provider), who comes up with a abstract specification of a service that may include the goals of the service, some of the key high-level tasks, the constraints that define validity of contributed detailed designs, and criteria for comparing multiple designs. This specification is then published to the "crowd", and design ideas are sought. The crowd may include contributors who submit detailed designs for the entire service or parts thereof, and a potentially larger community of users of the service being designed,

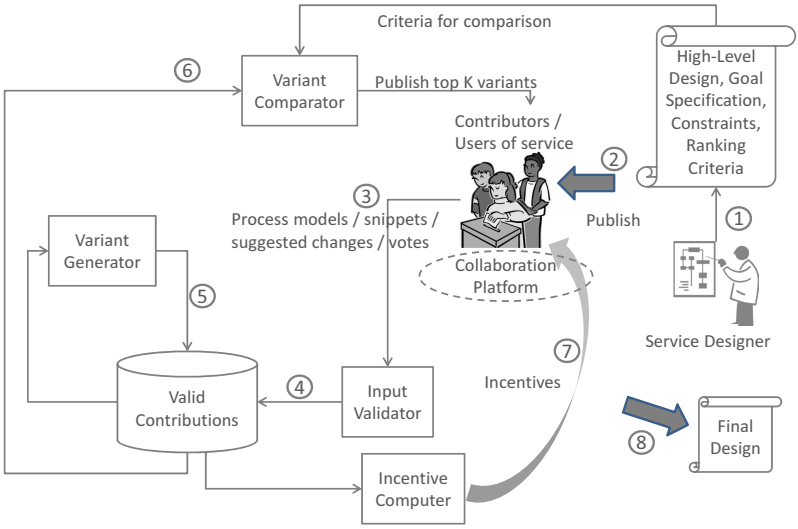

Fig. 2. Framework for crowdsourcing process models

who can review the contributed designs, suggest changes or enhancements, and rate the designs. Our framework includes a collaboration platform through which in-context discussions on service designs may be supported. With a sufficiently large crowd we will encounter the challenge of managing the scale of contributions - ensuring that invalid submissions do not attract undue attention, while good ideas from the many valid submissions are identified and highlighted for further consideration. This is where the formal reasoning and analysis afforded by our framework can play an important role, as we describe next.

The designs submitted by the crowd are passed through an *Input Validator* that automatically checks the validity of contributed designs, based on the formal specification provided by the service designer. Many designs that do not meet the necessary constraints are dropped at this step, and only the valid contributions are persisted with. The next key step is to analyze the existing designs and look for ways in which new variants can be generated by combining useful aspects of these designs. This is handled by the *Variant Generator*. This is a critical step since it involves (a) identifying the useful aspects that can be extracted out, using some notion of value that these aspects may bring (e.g. based on the evaluation criteria defined by the designer) (b) understanding which aspects can be combined and in what order, using a formal semantics for composition that also respects the high-level specification. Finally, the *Variant Comparator* evaluates all the generated variants using the known evaluation criteria and publishes the top K variants back to the crowd for further review and discussions. To ensure systematic progress, we expect alternate time-bound cycles of crowd participation (to review and elicit new ideas) and formal analysis (to filter, compose and prioritize these ideas for the next round of discussion). The process continues till the final design is arrived at, a selection that may be made by the service designer/provider at an appropriate point in

time, taking into account the relative merits of the candidate designs (ordered by Variant Comparator), and crowd feedback (e.g. through voting and discussions).

Finally, the participants in service design crowdsourcing need to be rewarded. This will be handled by an *Incentive Computer* that will rank the participants based on the relative strength of their contributions, as determined by it from the value of the designs submitted by or contributed to by each participant, during the crowdsourcing exercise.

4 Applying the Framework to Driver's License Service

This section describes how our framework can be applied to the driver's license case study. Although we do not propose any concrete implementation of the components of our framework, the following discussion should help one understand the overall flow on an actual example, while Section 5 will discuss the key research challenges to be addressed.

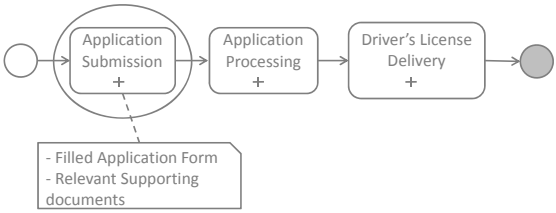

Fig. 3. A high level process published by the service designer

Let us assume that the Transport Department would like to seek ideas from citizens to improve the design of the driver's license issuance service. To align citizen interaction with the core components of the existing design, the designer decomposes it into sub-processes, namely, Application Submission, Application Processing and Driver's License Delivery. Figure 3 shows these subprocesses.

For the purposes of illustration, let us focus on the Application Submission sub-process. As shown in Figure 4, the designer has also provided a high-level subprocess for Application Submission along with the following constraints.

1. The citizen must apply for the Driver's License in person by visiting the RTO office.
2. A set of supporting documents is needed along with the application.
3. If the application and the supporting documents are not valid, the application for license will be rejected.

The contributors create variants of the detailed process from scratch or by modifying variants submitted by others. As shown in Figure 5 the first crowdsourced variant (Variant 1) for the Application Submission sub-process focuses on removing the citizens' pain point of waiting in a long queue before applying for a driver's license. Additional steps are added, without violating the constraints, to take an online appointment before submitting the application in order to avoid the wait.

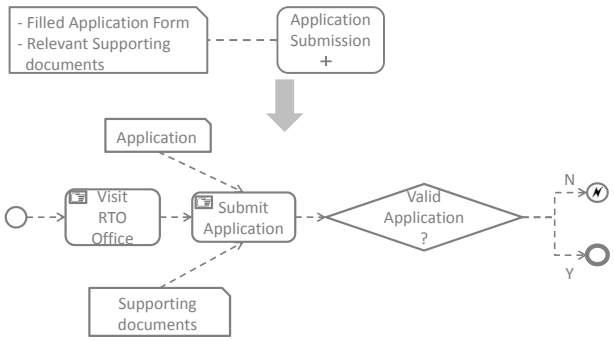

Fig. 4. An elaboration of the the application submission sub-process

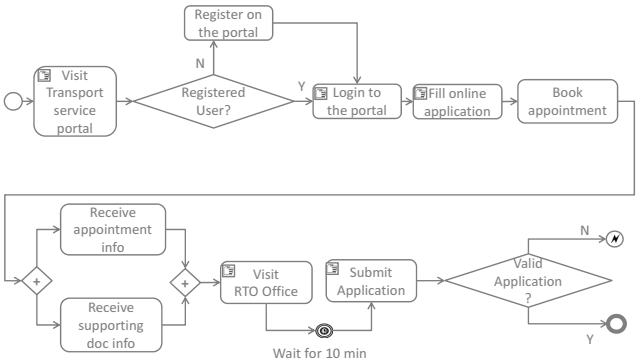

Fig. 5. Variant 1: An online portal is used to make appointments

Next, as shown in Figure 6, a citizen modifies Variant 1 to remove online booking and replaces it with a phone booking. However, just replacing the mode of appointment is not sufficient. With the phone booking, additional tasks are added where citizen receive an SMS reminder for the booked appointment. Also since the application form is no more filled online, the citizen has to fetch the form and fill it manually. The newly created variant (Variant 2) with the phone appointments is shown in Figure 7.

The Variant Generator identifies that the addition of an SMS reminder is compatible with Variant 1, since the prerequisite for receiving an SMS reminder is a booked appointment, which also exists in Variant 1. Hence the Variant Generator creates a new variant (Variant 3) merging Variant 1 and Variant 2, as shown in Figure 8. However, other modifications such as requesting and filling application form in RTO office are not compatible with the Variant 1 and hence does not lead to new variants.

Another citizen prefers to simply visit the RTO office without appointments. Hence, she changes Variant 2 with manual submission of the application as shown in Figure 9. These changes result in the creation of a new variant (Variant 4) as shown in Figure 10.

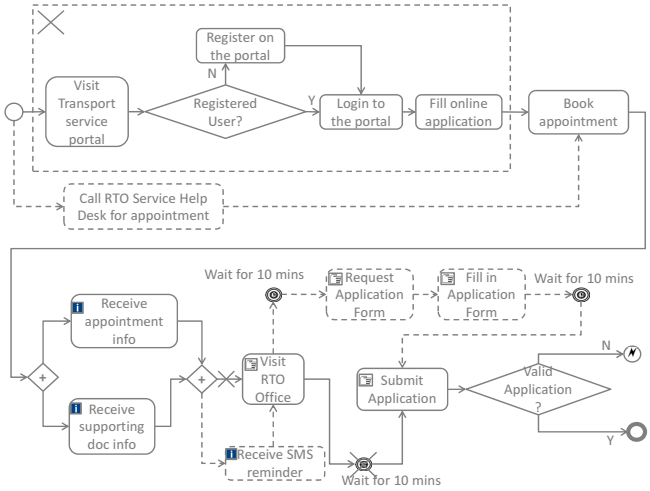

Fig. 6. Changes to Figure 5 to make appointments over phone

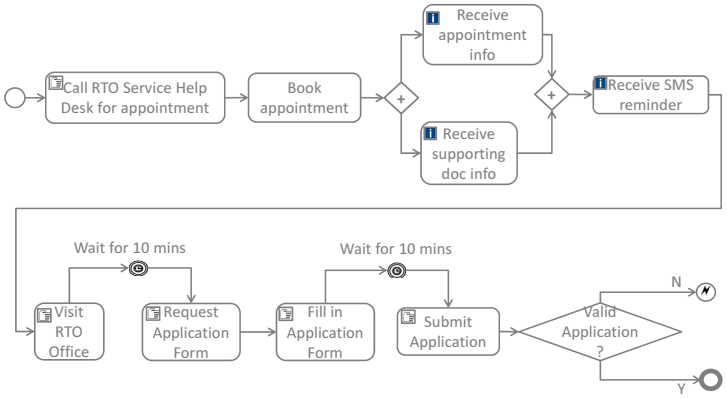

Fig. 7. Variant 2: Service help desk is used to make appointments over phone

Once these four variants are arrived at, the Variant Comparator compares these variants based on certain criteria set by the service designer such as cost, time, convenience, transparency, and susceptibility to corruption. Variants 2 and 3 provide the maximum information back to the citizen and maybe rated higher on transparency than other variants. Variant 3 on the other hand is the least time consuming and Variant 4 is completely manual, hence more prone to corruption. Variant 3 may be perceived as the most suitable one, but the service designer may also choose to retain Variant 2 as an option for citizens who do not have access to the internet for online booking. Additionally the crowd can also vote for their favorite variant. Once the new design is finalized, all the

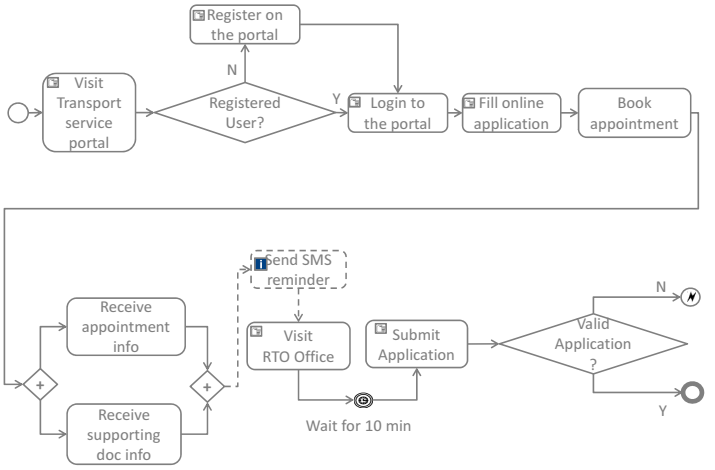

Fig. 8. Variant 3: Variant 2 improved with SMS reminder step suggested in Figure 6

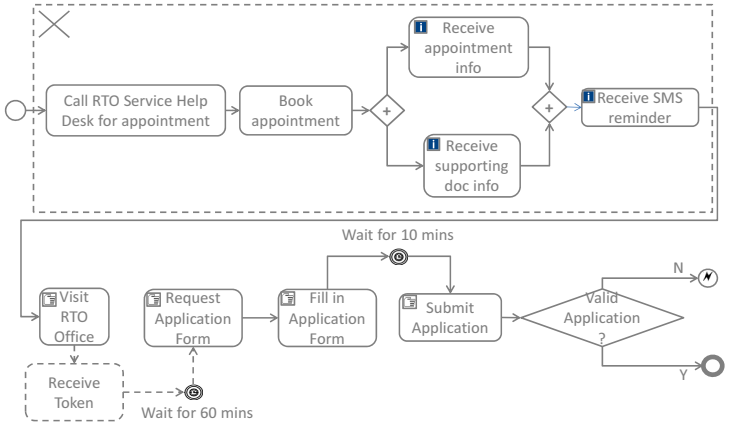

Fig. 9. Changes to Figure 7 to visit without appointments

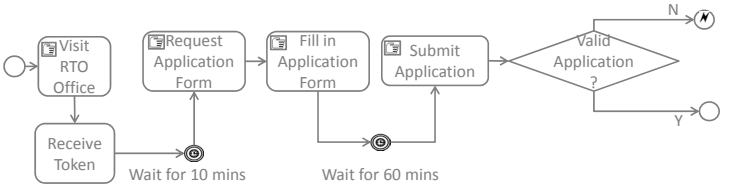

Fig. 10. Variant 4: Visit RTO office without application

contributors to the design are identified and rewarded appropriately, based on the relative merits of their contributions, as determined by the Incentive Computer.

5 Research Challenges

Given the exposition in the previous section, we summarize the broader research challenges that arise as well as hint at the techniques that may help address them.

Representing goals and constraints. The challenge is in capturing what the service designer considers as the high-level purpose of the service. To aid the automatic analysis of the variants, a formal specification of such goals is a necessity. Although the goal-based or commitment-based modeling techniques such as Tropos [3] and Amoeba [4] may come in handy, it is not clear how without a common vocabulary, the crowd and the designer would agree on the semantics of such goals and constraints. Relying on the crowd to manage the vocabulary as the contributions flow in may be a natural solution and needs to be studied.

Representing criteria for evaluation. The challenge here is in defining rules for ranking the variants. This is nontrivial because the criteria may conflict with each other, e.g., cost of the service and the comfort for the citizens. Further, the criteria should be such that the service designs would support evaluation of the chosen criteria, e.g., comfort level afforded by a variant. While multi-objective optimization is not new, some of the important criteria such as satisfaction and comfort are not easy to evaluate. A possible way forward maybe to combine a voting-based evaluation of such "soft" criteria with an automatic evaluation of the machine-friendly criteria such as the average time taken to deliver driver's licenses. Are there generalized rules of thumb that can help us compare variants without dependeing on the crowd for voting?

Representing service designs. Apart from the challenge of having a common vocabulary for a formal representation, we need to enable attribution of the contributed elements of service designs to the contributors. For example, for the single purpose of allowing phone appointments, several changes had to be made in Figure 7. While the newly added elements can be attributed to this contributor, many of the additions are entailed due to the basic purpose and are not novelties in themselves, e.g., having to request an application form now that no application was submitted online. How do we differentiate novel additions from additions required by the novelties?

Generating variants. How shall the variant generator create new variants that combine compatible features from multiple variants? Although service composition has received a lot of attention [9], the challenge here is in the scale of the state space of all possible variants that a typical planner or reasoner would have to deal with. A way forward is to develop a dominance relation between variants so that new variants need to be generated only from the set of variants that is not dominated by any other variant in the set. Such a relation should ideally be based on the criteria of evaluation. However, some of the important criteria are evaluated only by votes from the crowd. Are there other effective ways of pruning the state space?

Incentivizing contributors. How shall we assign credit to the contributors? And what structure of incentives would make this an individually rational mechanism so that

participating in it derives a greater utility for the contributors than skipping? Although mechanism design has been an active area with many significant results on mechanisms with individual rationality [10], the core prolem here is in assigning credit for the novelties contributed by a contributor. Naturally, the basis for assigning credit should be the value of the contributed variants. What would be the loss in value had this contributor not participated? Answering this question is nontrivial because the contributors build on top of each others' ideas in an iterative manner and new models of computing long term values may be needed.

6 Related Work

Recent years have seen a spike in interest in collaborative approaches to solving knowledge-oriented problems. Brabhan discusses many case studies of successful crowdsourcing in problem solving [1]. Notable examples include highly technical and scientific problems solved through wisdoms of crowd.

Participatory design is an established approach to design, that actively involves all stakeholders in the design process. Though the term has been used in a variety of fields, it is relatively new in service design [11]. Hartman *et al.* propose an early approach to participatory service designs [5]. However, this work either limits the participation of contributors to specific stages of design or is incapable of scaling when the number of contributors grows large. COCKPIT (Citizens Collaboration and Co-creation in Public Service Delivery) [5] is one such method where governments seek citizens' deliberation in service design and decision making. The methodology limits citizens' participation to voting their preferred design amongst the existing ones identified by the service designer. This technique does not allow citizens to freely specify their preferred way of experiencing the service. Unlike our approach it suffers from lack of true collaboration and co-creation of service value between the government and the citizen.

As demonstrated by Brabhan crowdsourcing is an appropriate model for citizen engagement in public administration, many governments are soliciting citizens' involvement in collectively solving problems at a city, state or national level [2]. With our approach the wisdoms of a crowd can be effectively tapped and utilized. Another more collaborative approach to crowd source workflow designs is displayed by Turkomatic [7,8] Turkomatic is a tool that recruits crowd workers (a.k.a. *turkers*) to aid requesters in planning and solving complex jobs, with the aim of identifying the best split of a complex knowledge task, e.g., writing an essay. The purpose here to ensure that the division of a complex problem into sub-problems aids the workers in solving the sub-problems effectively. Turkomatic's continuous price-divide-solve loop approach that asks workers to recursively divide complex steps into simpler ones, solve them, and later merge them, though is a truly collaborative, unlike our approach is not scalable. This is done with some manual moderation from the task requester to ensure quality of a split and a merge. James Surowiecki examines several cases of crowd wisdom at work, where he states that the wisdom of crowds in not derived from averaging solutions rather from aggregating them [14]. In our approach, we reconfirm the same by showing that the best solution need not necessarily be suggested by one contributor but by smartly aggregating solutions from different contributors in the crowd.

7 Conclusions

In this paper, we have introduced the problem of crowdsourcing service designs, which will enable true participatory design of services taking into account the needs and interests of both providers and consumers of a service. We have identified the challenges involved, and then proposed a high-level framework to address the same. The novelty of the framework lies in combining the richness of crowd participation with the technical rigor afforded by formal reasoning and analysis of service design models. The approach was illustrated through a case study involving a driver's license issuance service. We also discussed the research challenges that need to be addressed to realize the approach. These challenges will also guide our future work on this topic.

References

1. Brabham, D.: Crowdsourcing as a model for problem solving an introduction and cases. Convergence: The International Journal of Research into New Media Technologies 14(1), 75–90 (2008)
2. Brabham, D.: Crowdsourcing the public participation process for planning projects. Planning Theory 8(3), 242–262 (2009)
3. Bresciani, P., Perini, A., Giorgini, P., Guinchiglia, F., Mylopolous, J.: Tropos: An agent-oriented software development methodology. Autonomous Agents and Multi-Agent Systems 8(3), 203–236 (2004)
4. Desai, N., Chopra, A., Singh, M.: Amoeba: A methodology for modeling and evolving cross-organizational business processes. ACM Transactions on Software Engineering and Methodology (TOSEM) 19(2), 1–45 (2009)
5. Hartman, A., Jain, A.N., Ramanathan, J., Ramfos, A., Van der Heuvel, W.-J., Zirpins, C., Tai, S., Charalabidis, Y., Pasic, A., Johannessen, T., Grønsund, T.: Participatory Design of Public Sector Services. In: Andersen, K.N., Francesconi, E., Grönlund, Å., van Engers, T.M. (eds.) EGOVIS 2010. LNCS, vol. 6267, pp. 219–233. Springer, Heidelberg (2010)
6. Howe, J.: The rise of crowdsourcing. Wired Magazine 14(6), 1–4 (2006)
7. Kulkarni, A., Can, M., Hartmann, B., et al.: Collaboratively crowdsourcing workflows with turkomatic. In: Proc. CSCW (2012)
8. Kulkarni, A.P., Can, M., Hartmann, B.: Turkomatic: automatic recursive task and workflow design for mechanical turk. In: Proceedings of the Human Factors in Computing Systems, pp. 2053–2058 (2011)
9. Rao, J., Su, X.: A Survey of Automated Web Service Composition Methods. In: Cardoso, J., Sheth, A.P. (eds.) SWSWPC 2004. LNCS, vol. 3387, pp. 43–54. Springer, Heidelberg (2005)
10. Saijo, T.: Incentive compatibility and individual rationality in public good economies. Journal of Economic Theory 55(1), 203–212 (1991)
11. Sangiorgi, D., Clark, B.: Toward a participatory design approach to service design. In: Participatory Design Conference, pp. 148–151 (2004)
12. Singh, M.: Self-renewing applications. IEEE Internet Computing 15(4), 3–5 (2011)
13. Spohrer, J., Maglio, P., Bailey, J., Gruhl, D.: Steps toward a science of service systems. Computer 40(1), 71–77 (2007)
14. Surowiecki, J.: The wisdom of crowds. Anchor (2005)