



HAL
open science

Mediating Citizen-Sourcing of Open Government Applications – A Design Science Approach

Mai Abu-El Seoud, Ralf Klischewski

► **To cite this version:**

Mai Abu-El Seoud, Ralf Klischewski. Mediating Citizen-Sourcing of Open Government Applications – A Design Science Approach. 14th International Conference on Electronic Government (EGOV), Aug 2015, Thessaloniki, Greece. pp.118-129, 10.1007/978-3-319-22479-4_9 . hal-01412243

HAL Id: hal-01412243

<https://hal.science/hal-01412243>

Submitted on 8 Dec 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Mediating Citizen-Sourcing of Open Government Applications – a Design Science Approach

Mai Abu-El Seoud and Ralf Klischewski

German University in Cairo
11835 New Cairo City, Egypt
{mai.abuelseoud, ralf.klischewski}@guc.edu.eg

A design science approach is followed to develop architectural blueprints for implementing platforms to source open government applications from citizens. Zachman framework is initially used as a guide to categorize and develop the artefacts. After designing the blueprints, their usefulness is demonstrated through prototype implementation, and their potential for problem solution is evaluated from the development perspective as well as communicated to governmental peers. Contributions to research and practice include a set of blueprints covering the top levels of a platform's enterprise architecture, a reusable sourcing platform prototype, set of validated test cases for following up the implementation process, as well as success factors and lessons learned from the government perspective.

1 Introduction

Since the introduction of the open government concept in 2009, many governments started to follow this path. However, intermediate monitoring revealed that governments still need to exert much more effort to utilize the emerging technologies in the service of open government [35], one of which citizen-sourcing, among other Government 2.0 technology supports [37, 3]. Citizen-sourcing is defined as the adoption of crowdsourcing principles and technologies in the public sector [24], and crowdsourcing is well-established in the private sector as an instrument for supporting customer participation and knowledge sharing [14].

The focus of this research is on enabling citizen-sourcing of open government applications because (a) applications are needed to harness the benefit of available open data [35], and (b) governments mostly do not have the capacity to develop such applications to serve citizens' needs. While citizen-sourcing could be the solution to address this shortage, the research gap is perceived as the lack of sufficient knowledge of how to efficiently and effectively develop citizen-sourcing platforms that enable citizens to share self-developed web applications to support open government. In this regard, the research question is: Which architectural artefacts are needed to develop an open government application sourcing platform? Accordingly, this research aims to develop, validate, demonstrate, test, and communicate the appropriate architectural

artefacts that are required and could be easily replicated for developing open government application sourcing platforms. As being concerned with developing new innovative artefacts and testing them in real world context, a design science approach has been followed. After discussing the extant literature on citizen-sourcing and open government applications (section 2), the applied design science process is introduced and subsequently the main activities and results of each phase are presented (section 3). Finally, the conclusion (section 4) summarizes the findings and points to future research.

2 Citizen-sourcing of Open Government Applications

Private organizations always seek for new innovative approaches that help in improving their customers' engagement and participation [2, 25]. Recently, governments worldwide are following similar strategies in attempting to enhance citizen participation and collaboration. Moreover, the unremitting evolution of ICTs has changed the way governments interact with their citizens [9]. The concept of citizen-sourcing relates to and integrates several other well-established concepts [16, 27]:

- *Citizen engagement* reveals the main objective of citizen-sourcing [24, 28]. Citizen engagement is not only about governments being transparent and gaining support from citizens in return, but it is also concerned with changing the traditional and often insufficient ways of exchanging information between governments and citizens [22, 8]. Moreover, with the acceleration of citizens' demands, governments have started to realize the importance of adopting a citizen-centric approach to understand citizen demands towards government transformation [26, 16].
- *Crowdsourcing* as a term has been first coined by Jeff Howe [18, 19] as the act of an institution or a company to perform a function or a task that was once performed by the employees. It provides a number of benefits to companies such as the ability to gain access to a very huge community of current workers in the requested field, save cost and time, enhance quality of the provided solution or idea [28, 20]. The task should be outsourced to an undefined network of people which is referred to as the crowd. Accordingly, Hilgers and Ihl [16] defined citizen-sourcing as the governmental adoption of the crowdsourcing principles and techniques in the public sector. Citizen-sourcing is seen as a new approach that (re-)shapes the relationship between governmental agencies and citizens based on evolving practices from the private sector [24, 16].
- *Open government* as a term dates back to the initiative of U.S. President Obama having announced a new era of participation, transparency, and collaboration in 2009. Since then widely adopted, this concept embraces new approaches to further involve citizens and external parties in governmental processes [16]. Yet, the ambiguity regarding open government expectations largely remains, and definitions in the literature (e.g. [5, 37]) mostly rely on three principles which are: availability of regularly updated governmental information for citizens (transparency); citizen engagement across all the levels of government, non-profit organizations, and businesses (collaboration); and improving effectiveness in government and enhance

decision-making quality through citizen engagement (participation). Challenges in open government implementation and evaluation are manifold and still subject to research.

While there are many efforts directed to serve open government initiatives worldwide, most of these initiatives have been restricted to tackling the ability of the governments to present their data to the public in the form of open data such as Data.gov platform [7]. Hence, it was claimed that governments need to direct more efforts in utilizing the emerging technologies in the service of open government [35]. Moreover, to be able to make best use of open data, either web or mobile applications are needed to harness the benefit of the availability of this available open data. Notably, the Obama administration gave citizen-sourcing superiority among other Government 2.0 technologies [27, 3]. Yet, there is still very limited research directed towards utilizing citizen-sourcing in an open government context [31, 28].

From the research perspective no unique definition of open government applications has been coined on the basis of which their scope and expected deliverables could be identified. Hence, for the time being, we refer back to the definition of web-based applications. A web-based application can be considered as an application that is developed to be executed in a web-based environment [12, 4], i.e. enables information processing functions to be initiated remotely from a web browser and executed partly on a web server, application server, and/or database server [4]. Based on this, an open government application is considered as a web-based application that serves the purpose of open government by serving its above mentioned principles. In the current research, open government applications are envisioned to be provided through a platform-based citizen-sourcing process.

3 A Design Science Approach for Platform Development

The platform development assumes the following scenario: The citizen-sourcing process starts by an application request in the form of an open call. The requester can be representatives from the government, non-government organization, or citizens. This request is then published on an online platform that acts as an intermediate between the requesters and providers. The skilled citizens are the providers who submit applications that fulfil the published requests. An intermediate in the form of an online platform facilitates the process between the requesters and providers. Citizens can also act as testers and evaluate the submitted applications by reviewing them, posting possible enhancements to the developers, providing a rating etc. After testing the uploaded web applications, the requesters can choose applications that match best the call requirements and proceed with setting these applications into productive mode either on the intermediate or on another platform.

Based on this scenario figure 1 depicts the main actors, concepts, and their interrelations as perceived within the frame of this research. Since the research gap is concerned with the lack of knowledge with regards to efficiently and effectively developing open government application sourcing platforms, the creation of the appropriate architectures as reusable blueprints is considered essential (samples of architecture

blueprints are depicted lower part; see section 3.1 for more details). Since the research is aiming for developing new artefacts to solve a given problem, a design science methodology was found to be the most appropriate.

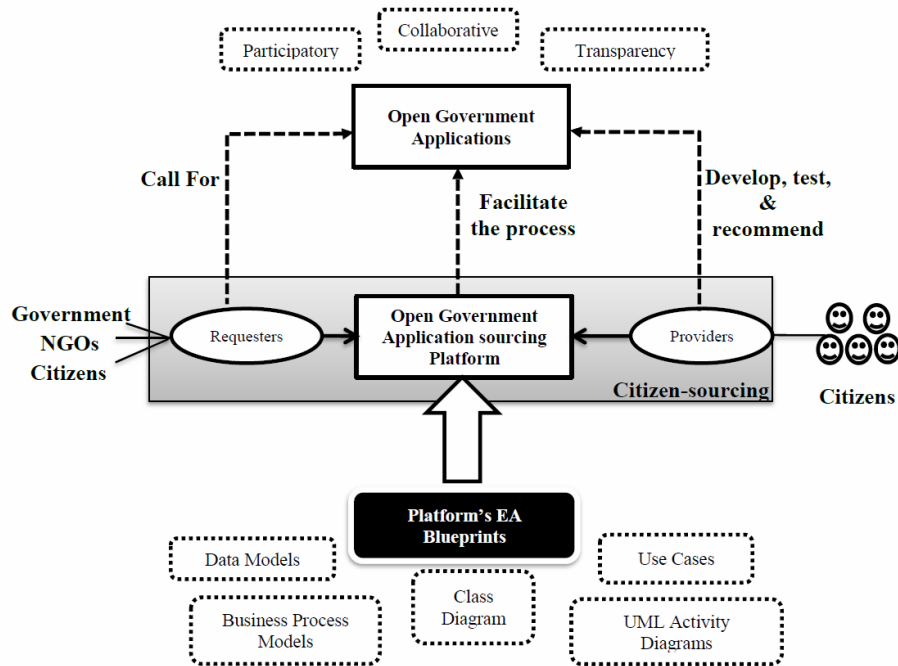


Fig. 1. Research model

The literature has put forward several schema variations of the design science processes from which the widely cited approach by Peffers et al. [29] has been adopted in the current research. This design science process consists of six main phases with the different possible entry points for the research: identifying problem and motivation, defining objectives for a solution, design and development, demonstration, evaluation, and communication. This research has started the nominal process from the second phase as the research problem has been identified from previous literature and governmental documents on the status of open government (see section 2). For each of the remaining five phases the main activities (stretching from April 2014 until January 2015) as well as the results are described in the following subsections.

3.1 Defining Objectives for a Solution

Objectives for a solution should be inferred from the problem definition. The problem is perceived as (1) the governmental necessity to utilize new technologies, especially citizen-sourcing, in the favour of open government, (2) the need to make use of the available open data through web applications, (3) the need of enhancing the relationship between citizens and their governments through enhancing citizen engage-

ment (see section 2). All of this can be achieved by engaging citizens in the development of open government web applications. Using a citizen-sourcing process will not only help governments achieve a new level of citizens' engagement but also provide governments with a wide range of developed and tested web applications at minimal cost.

Artefact development should be based on theories and/or constructs. Research has frequently emphasized the importance of high-level integration, alignment, and coordination between different architectures in an organization, as various architectures are needed to manage the complexity of large information systems [1, 13, 15, 23]. Even though the to-be developed intermediary platforms are not *a priori* assumed to be part a larger organizational context, using an enterprise architecture framework is considered an appropriate approach to ensuring the integration of all system's components. Enterprise architectures transform the broader principles, capabilities, and goals defined in the strategies into systems and processes that enable the enterprise to realize these goals [34]. From various available enterprise architecture frameworks, the Zachman framework [38] was chosen to provide guidance for identifying relevant information systems' artefacts. The framework offers a logical structure by which organization can follow to ensure the information flow architectures are integrated with their business units [11]. Zachman framework ensures accurate and consistent results and enables the description of the architecture from the view point of every stakeholder respectively. The core of the Zachman framework consists of a two dimensional matrix of which the six columns represent various abstractions (data, process, location, people, time, and motivation) and the six rows indicate different actor perspectives (planner, owner, designer, builder, programmer, and user) [38].

One of the authors has acted as the planner, owner, and designer of the online platform covering the first three top levels of Zachman framework, following the recommendation to elaborate on the columns in a top-down approach. A variety of artefacts have been proposed to satisfy Zachman framework, from which the approach of Pereira and Sousa [30] has been employed for identifying the artefacts of this research. Most of these proposed deliverables have been developed except for those strongly requiring contextualization (i.e. all artefacts related to location as well as to people and motivation from the designer's view).

3.2 Blueprint Design and Development

This phase is mainly concerned with developing and designing the artefacts. Each blueprint is developed following certain design notations, for example unified modelling language (UML) for activity diagrams, crow's foot notation for entity relationship diagram, and business process modelling notation (BPMN) for business process modelling. Requirements were elaborated from analysing existing crowdsourcing processes, extended through applying open government principles as far as feasible. See table 1 for the type and number of blueprints developed, and figure 2 depicts a sample of the developed blueprints from the owner's view: the entity relationship diagram, one of the business process models, and the app developer's use case diagram (all artefacts are available on request).

After developing the initial set of the blueprints, an expert panel has been conducted consisting of four academics from the Information Systems field (one associate professor and three senior assistants) as well as three software developers working in the software development industry. Although the participants were selected upon convenience, it was ensured that they all have sufficient knowledge and experience regarding development and use of architectural models.

Table 1. Summary of developed artefacts

Level	Abstractions	Current Research deliverables
Level 1 – Objectives/Scope Planner’s View	Data (What)	List of important data
	Function (How)	List of important business process
	People (Who)	List of stakeholders – Stakeholders’ analysis
	Time (When)	List of important events
	Motivation (Why)	List of the business goals
Level 2 – Business Conceptual Model Owner’s View	Data (What)	Entity Relationship Diagram (ERD)
	Function (How)	Business Process Models (5)
	People (Who)	Use case diagrams (5)
	Time (When)	Event process chain Diagrams (5)
	Motivation (Why)	Platform’s rules
Level 3 – System Model Designer’s View	Data (What)	Class diagram
	Function (How)	Activity Diagrams (17)
	Time (When)	State diagram

The participants were first presented with the project scope, all the developed artefacts, and the notations used. Subsequently they were asked to review the compatibility of the artefacts with Zachman framework, the clarity and correctness (logic) of the artefacts, and the artefacts notations, as well as to suggest ways to enhance the artefacts and recommend new artefacts that might be missing.

Feedback was collected through one-to-one sessions with each panel expert. All feedback has been taken into consideration and accordingly some of the artefacts have been modified and enhanced. Most of the artefacts (such as the activity diagrams, use case diagrams) have only been slightly altered to improve the wording to be clearer for readers. However, the entity relationship diagram was significantly changed in terms of adding more entities and relationships between them for a better database design. Since the development of the class diagram considers the entity relationship diagram as an input, it was also modified with the new changes. For the framework compatibility, the feedback showed that the presented artefacts are compatible with Zachman’s description of each cell.

3.3 Demonstration: Blueprints Used for Prototyping

In this stage, the artefacts are utilized in a suitable context to solve one or more instances of the originally identified problem. According to [29], this could take the form of experiments, case study, or any appropriate form. In this research, a simplified version of the platform has been developed based on the enhanced version of the designed artefacts from the previous stage, now used as blueprints. A prototype is considered a simple version that implements the core features of the intended system

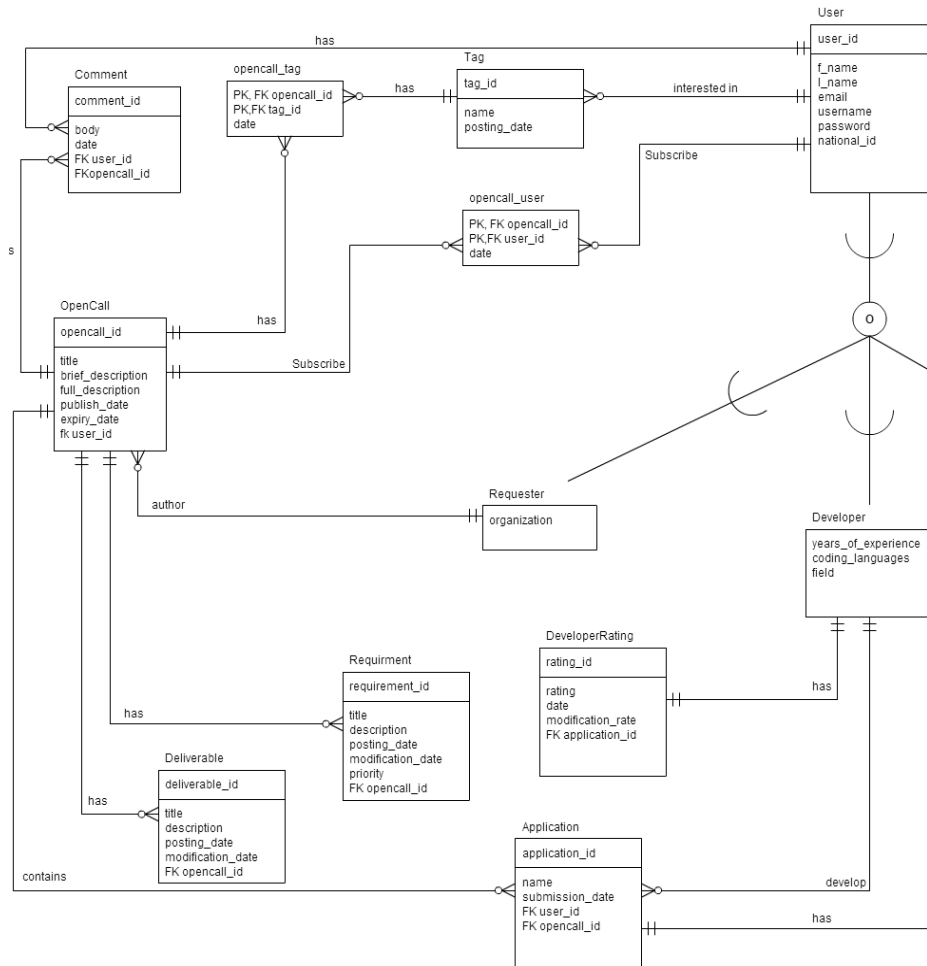
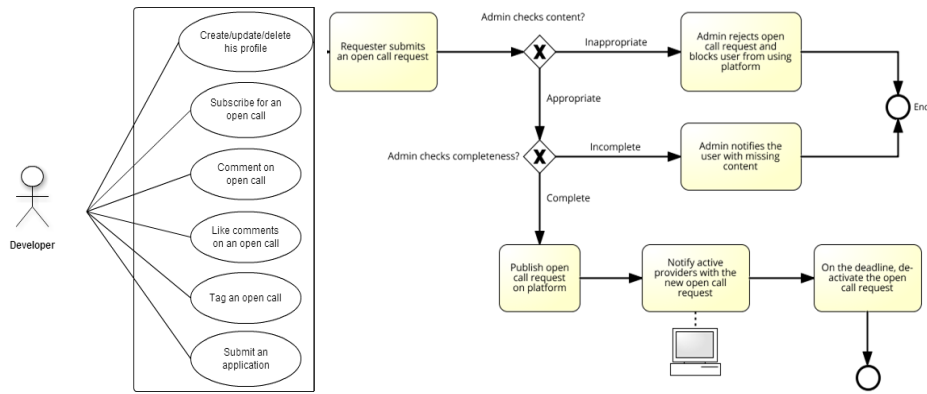


Fig. 2. Sample blueprints: app developer's use cases, process of call request, ERD (cropped)

to be developed quickly and to be used for early evaluation [32]. According to [33], prototypes take the architectures, designs, features, and functionalities of the system as an input and demonstrate them through the implementation of the prototype.

The prototype development has been outsourced to a computer science graduate who has been working in a reputable software development company for two years. The developer was chosen based on his qualifications, development skills, availability, and willingness to develop the platform. The development process took about six weeks. The developer was provided with all the enhanced artefacts. All the core features are implemented except some supporting features such as the 'forgot password'. The outcome of this phase is an implemented prototype based on the blueprints. The prototype has some limitations such as it only runs on Firefox internet browser, the user interface is very basic and simple, and the uploaded applications should be coded with the same language as the prototype. Two dummy applications were developed to be used in the prototype testing.

3.4 Evaluation of Effectiveness and Efficiency

The artefacts in use should be observed and measured for their effectiveness and efficiency to solve the problem. In this phase, a comparison between the objectives of the solution and the actual results should take place. At the end of this phase, the researcher can choose either to iterate back to the design and development phase to improve the artefacts or to continue to the last phase and leave the improvements to future research and projects [29].

In general, effectiveness is the extent to which the stated objectives are met. To be able to test the effectiveness of the blueprints in developing open government application sourcing platforms, the researcher has initially generated 110 test cases to test each blueprint through the prototype. Each blueprint was used to generate success and failure test cases to be used for evaluation by a testing committee. For example, using an activity diagram, test cases are generated that ensure the prototype is following the same sequence as the design etc.

A test case is a specific input including procedures that the tester will do to test the software at hand [21]. The main purpose of the test cases is to detect the faults in the software against the design and requirements. This is considered a black box testing as it is concerned with the external functionalities of the platform, not the coding design and style [6]. Accordingly, the test cases were validated by a certified software tester (chosen based on convenience) who works in the website testing field for almost three years. The tester was provided with all the developed artefacts plus the generated test scenarios and asked to validate the wording of the test cases, the test cases in accordance with the artefacts, the classification of the test cases, and to identify missing test cases.

After the validation, four new test cases were added to the initial pool to result in 114 cases to be executed. Upon execution 105 passed test cases were found as well as 9 failed cases. Accordingly, the pass rate is calculated to identify the percentage of passed test cases, reflecting the quality of the implemented software and how far it meets the objectives and designs. Since a pass rate of the total test cases above 90% is

claimed to be acceptable for many software development projects [6], a pass rate of more than 92% in this research is considered satisfactory.

Second, efficiency is doing the work with fewer resources. The resources utilized to develop the prototype (e.g. time consumed to understand and use the blueprints) are identified and a set of three efficiency measurable variables are used to evaluate efficiency. A simple questionnaire (to be answered by the prototype developer) had been developed to test each of these variables using a 5-point Likert scale. The prototype developer agreed that the blueprints meet the clarity, time, and accuracy standards; this confirms a positive feedback after using the blueprints as a base for implementing the application sourcing platform.

3.5 Communication: Architecture Review by the Egyptian Government

The problem, the developed artefacts, and their solution capability should also be communicated to researchers and other relevant audiences (professionals, other stakeholders). This research has been communicated through presenting the research problem, artefacts, prototype, and evaluation results to selected citizens, including academics, and representatives of the ministry that is responsible of electronic government in Egypt with the main objective to employ ICTs in implementing and improving e-government services. As the Egyptian government has started to consider open government and recently announced an open data initiative,¹ feedback from these stakeholders was considered essential. The vice minister has been contacted and interviewed as well as a project manager and the software quality assurance manager who are mainly interested in improving the electronic services which the Egyptian government offers to the citizens. The interviews were mainly directed towards the applicability of such technology within the Egyptian context and whether the government can consider the architectures as a first step of implementing citizen-sourcing to develop open governments web applications or not.

During the interviews, the representatives from the Egyptian government have shown serious interest in such projects, highlighting that this is what they actually need now in terms of citizens engagement and participation. However, Egypt is still at a very early stage having an open data initiative. After some discussion about the readiness of the government to consider the research artefacts, it was agreed that some prerequisites should be met before implementing such technology: (a) current regulations prevent the government from acquiring software solutions except from a few contracted software vendors – this would have to change when implementing the application sourcing platform as the providers will be citizens expected not to belong to one of these vendors; (b) the availability of governmental open data for the citizens is considered a first step before implementing this application sourcing platform – yet multiple efforts are still needed to achieve this in the near future; (c) the availability of skilled citizens who can develop and test the open government applications should be ensured – this can be achieved through establishing partnerships with different university, research centres, and software development start-ups.

¹ http://www.egypt.gov.eg/english/general/Open_Gov_Data_Initiative.aspx

4 Conclusion

After reviewing the extant literature on the related concepts and identifying the research gap, this research has developed the needed architectures for developing a platform for sourcing open government applications from citizens. Subsequently, these architectures were validated through an expert panel upon which the architectures were enhanced. These architectures were then demonstrated by implementing a prototype of the online application sourcing platform by a third party. Test cases had been developed and applied to evaluate the effectiveness of the blueprints based on the implemented prototype, and a simple questionnaire was conducted to evaluate the blueprints' efficiency in developing the application sourcing platform. Finally, the research has been communicated namely to governmental representatives, leading to identification of success factors of platform utilization.

The contribution of this research is both practical as well as theoretical. Firstly, a set of validated, demonstrated, tested, and communicated software architectures for developing open government application sourcing platforms is provided as blueprints for reuse. Secondly, the prototype developed based on these architectures can be further enhanced and used by governments or other stakeholders to adopt citizen-sourcing for open government applications. And the set of validated test cases can be used as a base for generating more test cases for testing a fully functioning platform. Moreover, the government feedback may serve as input for developing action plans and roadmaps when implementing such citizen-sourcing projects. As contribution to e-government research can be considered the adoption of Zachman's framework to develop and categorize application sourcing platforms architectures as well as the systematic approach of utilizing architectures to bring out an e-government solution intended for replication.

Limitations of time and other resources have caused also several limitations in research such as: limited number of high-level expert panellists; the prototype was only implemented for Firefox internet browser; the efficiency questionnaire was simple and not validated nor piloted; only dummy application were uploaded ("sourced") and no sourced application was set into production mode; only one developer was recruited to actually use the artefacts to implement the prototype. Especially the last limitation raises concerns about generalization of the efficiency evaluation results.

Beyond overcoming these limitations, future research is suggested to address the following questions: How to ensure security of the submitted applications as well as property rights? How to integrate the developed applications with the already available governmental infrastructure? How to publish the open government applications on the platform and link them with governmental public databases? What kind of open government tests are appropriate as pre-defined test cases executable on the application sourcing platform? And on the infrastructure level: what is the role of architectures in developing, disseminating, and managing e-government solutions in a distributed environment? After all, the application sourcing platform is only an intermediary which could be replicated in many countries and contexts. However, what the principles of open government mean in a given context and what quality benchmarks are to be applied, this should be subject to an on-going government-citizen dialogue.

5 References

1. Barros, O., Julio, C.: Enterprise and process architecture patterns. *Business Process Management Journal*, 17(4), 598-618 (2011).
2. Boudreau, K.: Open platform strategies and innovation: granting access vs. devolving control. *Management Science Journal*, 56 (10), 1849–1872 (2010).
3. Brabham, D. C.: Using crowdsourcing in Government. Collaborating across Boundaries Series, IBM Center for the Business of Government. (2013). <http://www.businessofgovernment.org/sites/default/files/Using%20Crowdsourcing%20In%20Government.pdf>
4. Bruno, V., et al.: Characteristics of web applications that affect usability: a review. In: 17th Australia conference on Computer-Human Interaction: Citizens Online: Considerations for Today and the Future, pp. 1-4. Computer-Human Interaction Special Interest Group (CHISIG) of Australia (2005).
5. Chun, S. A., et al.: Government 2.0: Making connections between citizens, data and government. *Information Polity* 15(1), 1-9 (2010).
6. Desikan, S., Ramesh, G.: *Software testing: principles and practice*, Pearson Education: India (2006).
7. Ding, L., et al.: Data-gov Wiki: Towards Linking Government Data. In AAI Spring Symposium: Linked data meets artificial intelligence 10 (2010).
8. Dobos, Á. Jenei, Á. Citizen Engagement as a Learning Experience. *Procedia-Social and Behavioral Sciences* 93, 1085-1089 (2013).
9. Dutil, P. A., et al.: Rethinking government-public relationships in a digital world: Customers, clients, or citizens? *Journal of Information Technology and Politics*, 4(1), 77–90 (2007).
10. Executive Office of the President Office, Office of Management and Budget. (2009). Memorandum for the Heads of executive departments and agencies: Open Government Directive, Report No 20503. http://www.whitehouse.gov/sites/default/files/omb/assets/memoranda_2010/m10-06.pdf
11. Ertaul, L., Sudarsanam, R.: Security planning using Zachman framework for enterprises. In: Proceedings of EURO mGOV 2005, 153-162 (2005).
12. Finkelstein, A. C. et al.: Ubiquitous web application development-a framework for understanding (2002). <http://www0.cs.ucl.ac.uk/staff/A.Finkelstein/papers/uwa.pdf>
13. Frankel, D. S. et al. The Zachman framework and the OMG's model driven architecture. A White Paper by the Business Process Trends (2003).
14. Geiger, D. et al.: Managing the crowd: Towards taxonomy of crowdsourcing processes. In: 17th American Conference on Information Systems, Detroit, Michigan August 4-7 (2011).
15. Gregor, S. et al.: Enterprise architectures: enablers of business strategy and IS/IT alignment in government. *Information Technology and People* 20(2), 96-120 (2007).
16. Hilgers, D., Ihl, C. (2010). Open Governance and Citizensourcing: Applying the idea of Open Innovation to the public sector. *International Journal of Public Participation*, 4(1), 67-88.
17. Houy, C. et al.: Empirical research in business process management—analysis of an emerging field of research. *Business Process Management Journal* 16(4), 619-661(2010).
18. Howe, J.: The rise of Crowdsourcing. *Wired Blog Network: Crowdsourcing* (2006a) <http://www.wired.com/wired/archive/14.06/crowds.html>
19. Howe, J.: Crowdsourcing: A definition. *Wired Blog Network: Crowdsourcing* (2006b). http://crowdsourcing.typepad.cpm/cs/2006/06/crowdsourcing_a.html

20. Iren, D., Bilgen, S.: *Methodology for Managing Crowdsourcing in Organizational Projects* (2012).
21. Keyvanpour, M. R. et al.: A Classification Framework for Automatic Test Case Generation Techniques for web applications. *Journal of Information Processing and Management* 4(3), 26-39. (2013).
22. Klein, W. R.: Building consensus. In: Ch. J. Hoch, L.C. Dalton, & S. S Frank (Eds.), *the practice of local government planning* (423-438). Washington D.C.: ICMA (2000).
23. Lankhorst, M.: *A language for enterprise modeling*. Lankhorst, M. (3rd Edition) *Enterprise Architecture at Work* (pp. 75-114). Berlin Heidelberg: Springer (2013).
24. Lukensmeyer, C. J., Torres, L. H.: *Citizensourcing: Citizen Participation in a networked nation*. In K. Yang & E. Bergrud (Eds.), *Civic engagement in a network society* (207–233). Charlotte: Information Age Publishing (2008).
25. Majchrzak, A., Malhotra, A.: Towards an information systems perspective and research agenda on crowdsourcing for innovation. *The Journal of Strategic Information Systems* 22(4), 257-268 (2013).
26. Medeni D.T. et al.: The Demand for Development of E-Government Service and Gateway in Turley Taking Citizens Perceptions and Suggestions into Accountl. In: M. A. Shareef, V. Kumar, U. Kumar & Y. K. Dwivedi (1st Eds), *Stakeholder Adoption of E-Government Services: Driving and Resisting Factors* (116-135). Hershey: Information Science Reference (2010).
27. Nam, T.: Suggesting frameworks of citizen-sourcing via Government 2.0. *Government Information Quarterly* 29(1), 12-20 (2012).
28. Nam, T.: Citizens' attitudes toward open government and government 2.0. *International review of administrative sciences* 78(2), 346-368 (2012).
29. Peffers, K. et al.: A design science research methodology for information systems research. *Journal of management information systems* 24(3), 45-77 (2007).
30. Pereira, C. M., Sousa, P.: A method to define an Enterprise Architecture using the Zachman Framework. *Proceedings of the ACM symposium on Applied computing* (pp. 1366-1371). ACM (2004).
31. Prest, E.: *Citizensourcing* (Doctoral dissertation, Central European University) (2012). www.etd.ceu.hu/2012/prest_emma.pdf
32. Sabale, R. G., Dani, A. R.: Comparative Study of Prototype Model For Software Engineering With System Development Life Cycle. *Journal of Engineering* 2(7), 21-24 (2012).
33. Scacchi, W.: *Process models in software engineering*. *Encyclopedia of software engineering* (2001). from <http://www.ics.uci.edu/~wscacchi/Papers/SE-Encyc/Process-Models-SE-Encyc.pdf>
34. Tamm, T. et al.: How does enterprise architecture add value to organizations? *Communications of the Association for Information Systems* 28(1), 141-168 (2011).
35. The White House: *The Obama's Administration Commitment to Open Government: Status Report* (2011). http://www.whitehouse.gov/sites/default/files/opengov_report.pdf
36. Whitla, P.: Crowdsourcing and its application in marketing activities. *Contemporary Management Research* 5(1), 15-28 (2009).
37. Yu, H., Robinson, D.: *The New Ambiguity of Open Government*. (2012). http://papers.ssrn.com/sol3/papers.cfm?abstract_id=2012489
38. Zachman, J.: A framework for information systems architecture. *IBM systems Journal*, 26(3), 276-292 (1987).