A Reference Architecture for Participatory Budgeting in an eGovernment Landscape

Achim Reiz[1] and Michael Fellmann[1]

[1] *Rostock University, 18051 Rostock, Germany*

Abstract

Participatory Budgeting (PB) puts a part of a city's public budget directly in the responsibility of its constituents – most notably, the citizens – who can make proposals on how to spend the money and vote on the proposals they like best. This type of resource allocation can increase democratic understanding and the efficiency of public spending. However, integrating or adding such IT-supported PB applications into eGovernment systems is far from a trivial task as a PB should not be isolated from the rest of the governmental IT systems but integrated. Thus, the complex system landscapes of the municipalities have to be touched.

Reference architectures can guide the creation of future-proof software. Hence this paper is meant to assist the implementation of PB into an existing eGovernment landscape by proposing a PB reference architecture. It shall support administrations who plan to implement a PB by proposing a possible system architecture and integration scenarios. As PB is part of an eGovernment system, it also presents a state-of-the-art eGovernment IT architecture on which other applications can build.

Keywords

Participatory Budgeting, Reference Architecture, eGovernment

1. Introduction

Participatory Budgeting (PB) puts a part of the public budget directly in the hands of the constituents. The citizens can craft proposals for spending the budget, which are later voted on by the community, with the proposals receiving the most votes getting implemented. This democratic process of allocating parts of the budget increases the inclusion of otherwise overlooked groups, triggers debates on how to spend the budget, increases the public's understanding of governmental processes, and thus strengthens the quality of democracy. Further, PB improves state performance by increasing accountability [1].

In recent years, the idea of PB spread worldwide and increasingly took hold in administrations worldwide. Moreover, as PB initiatives often use specialized software ([2] provides an overview of existing solutions), these systems often need to be integrated into the larger IT infrastructure landscape. At best, these PB systems work seamlessly with other eGovernment (eGov) applications, sharing data and logins. Such interoperable applications need software architecture management to abstract the complexity to a controllable amount [3]. However, creating software architectures from scratch is not a trivial task, as many systems need to interact, and the design choices directly influence the corresponding systems. Reference architectures can support design decisions and mitigate the complexities. They offer blueprints that can be adapted and integrated for individual use cases and support the creation of high-quality concrete architectures [4].

While some research papers describe actual instantiations of eGov-architectures, neither a general reference architecture for PB nor eGov systems are available. This lack leaves aspiring administrations in need of such systems on their own. Even though their needs and challenges are often similar, every administration needs to create its own IT architecture, which leads to unnecessary heterogeneity and

BIR 2022 Workshops and Doctoral Consortium co-located with 21th International Conference on Perspectives in Business Informatics Research (BIR 2022)

EMAIL: achim.reiz@uni-rostock.de (A. 1); michael.fellmann@uni-rostock.de (A. 2)
ORCID: 0000-0003-1446-9670 (A. 1); 0000-0003-0593-4956 (A. 2)

costs. Our research aims to fill this gap. This paper's goal is to derive a technical reference architecture for these PB applications into a broader governmental IT landscape. At first, we identified already existing PB initiatives and gathered the state-of-the-art in eGovernment architectures based on a systematic literature analysis. In the next step, the core features of a PB were integrated into a universal service-oriented eGov architecture. While the current focus of the architecture is on participatory systems, we believe that the architecture can also be used to integrate other kinds of eGov systems.

2. LITERATURE SEARCH

A systematic literature analysis first gathered the current state-of-the-art. The analysis is grounded in the methodology for systematic literature reviews in information systems research proposed by Barbara Kitchenham [5]. The search focuses on the main research question: "Which architectures were developed that are relevant for the IT support of PBs?" Papers were included as relevant if they presented an ICT-related architecture for PB or related fields (#1) or architecture where a PB can be integrated (#2). Thus, we initially queried for PB-specific architectures (#1.1, #1.2) and software architectures that are similar regarding their overall expected components (#1.3). Later, we included a more general view of eGovernment architectures (#2.1, 2.2) and integrated the PB-related aspects into the eGovernment view. We excluded architectures that are domain-specific without relevance for PB, literature reviews, or such that are unreadable (e.g., due to low resolution). PB is an increasingly global phenomenon. Thus, researchers from different regions and continents published results. The search was performed using the metasearch engine Scopus, which offers extensive coverage of peer-reviewed research output from various sources.

Table 1
Development of the Search String for the Systematic Literature Analysis

#	Search String	Results	Relevant
1.1	*TITLE-ABS-KEY((“participatory budget*” OR pb) architecture AND reference)*	24	–
1.2	*TITLE-ABS-KEY(("participatory budget*") AND architecture)*	9	[6]
1.3	*TITLE-ABS-KEY(("participatory budget*" OR "idea management" OR "innovation management") AND architecture)*	125	[7]
2.1	*TITLE-ABS-KEY((egov* OR e-gov*) reference architecture)*	89	[8,9]
2.2	*TITLE-ABS-KEY((egov* OR e-gov*) AND architecture AND framework)*	440	[10–19]

3. DEVELOPING THE REFERENCE ARCHITECTURE

The identified architectures differentiate the core elements like user interfaces, application servers, or data storage. While some research, like [10] and [14], do not further organize the underlying architecture, most of the other papers ([8,11–13,15–19]) propose a layered structure. This kind of architecture can encapsulate various levels of separated functionality, with each higher layer using the functionality provided by the lower levels. Layered structures allow for easier standardization due to the abstract definition of standardized interfaces and tasks [3]. The layered architectural structure is thus also inherited for the newly created reference architecture (cf. Fig. 1).

The first layer (A*ccess Layer*) of the eGov reference architecture concerns access to governmental services. It addresses the various devices that the end-users utilize. While [16] and [19] also propose additional channels like the telephone, digital tv, call center, and teleconferencing, our approach is limited to serving a website through a PC, a mobile device like a smartphone or tablet, or a stationary kiosk computer that is available, e.g., at a government site. The inclusion of additional mentioned channels like a call center would add high cost and complexity to the eGov/PB system. The advantage of the proposed reduction to mobile kiosks and PCs is that setting up a state-of-the-art website that is also mobile-enabled is sufficient to roll out the application to the targeted devices.

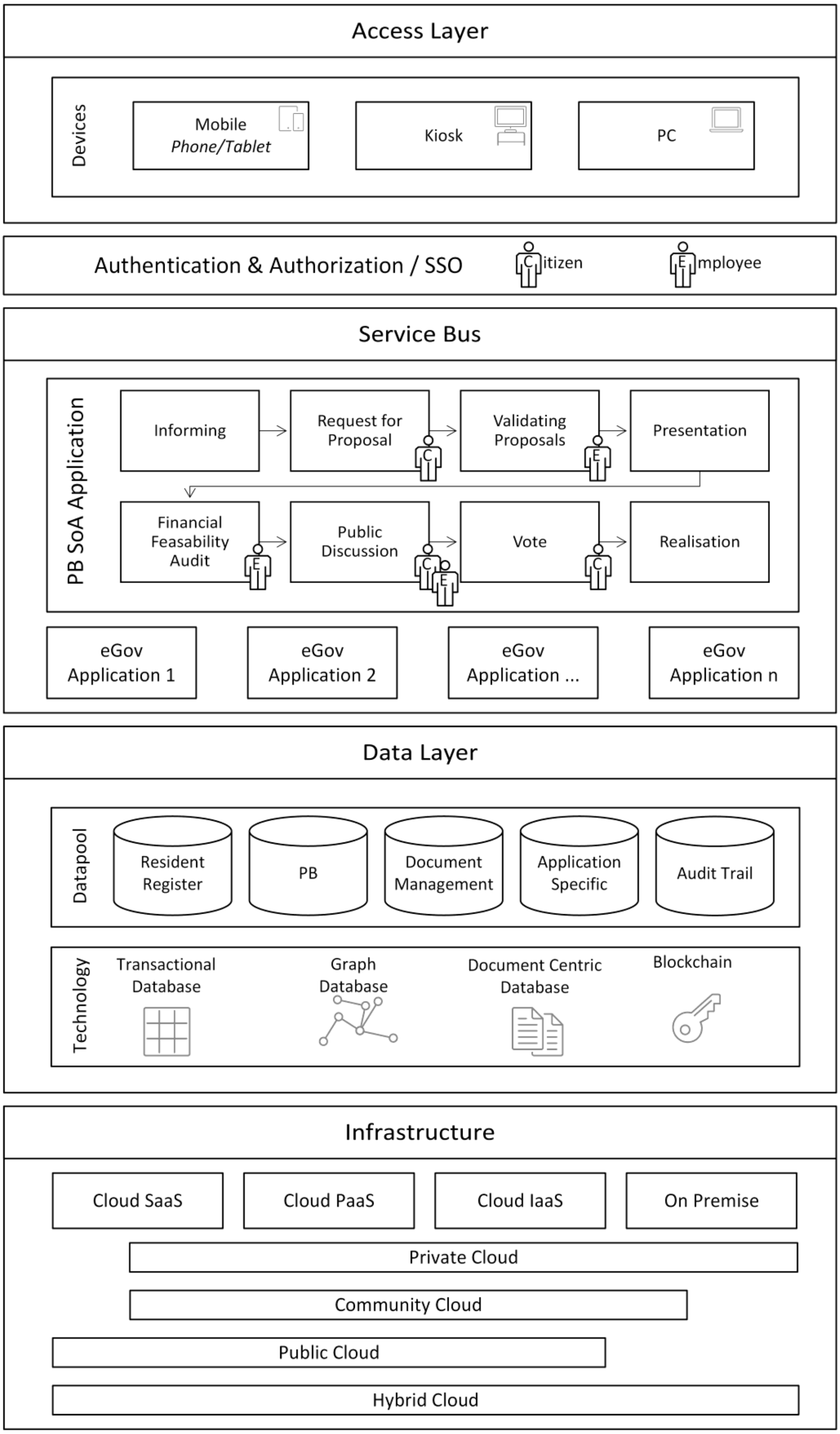

Figure 1: Proposed Reference Architecture for a Participatory Budgeting Application in an eGovernment Systems Landscape

The next layer manages the access control through *Authentication and Authorization.* It builds upon the single sign-on (SSO) proposed by [13,16] and shall enable the use of all governmental applications with just one login. It, thus, is a prerequisite for the one-stop paradigm proposed by [14,15,17,19] (having all necessary resources in one place), enabling proper rights management. There are many forms one can implement security through authentication. Depending on the implementing municipality's legal, technical, and cultural background, one can use login credentials, two-factor authentication, ID cards, and more. Also, the requirements for participation vary. While some initiatives might only require a simple registration with a username and password, others might postulate a residency. After successful authentication, a user is either authorized as a "Citizen" or "Employee" and can access the parts of the applications that are not publicly available.

The layer *Service Bus* comprises the business logic of the eGov system. As seen in the literature analysis, most of the more recent eGov architectures ([6,8,12,13,15–18]) build upon a Service-oriented Architecture (SoA)[2] to manage the complexity and ensure a high degree of modularity. Almost all other proposals are not far from the SoA paradigm (cf. interchangeable Java applications [11], portal solution [19]). Thus, the proposed reference architecture for PB also builds upon the SoA paradigm.

As a result, the PB software in our architecture is just one of many applications that utilize the same interfaces as the other eGov applications. Our architecture depicts an archetypical PB process based on [20]. The human pictogram marked with "C" or "P" indicates that authorization is required for the respective functionalities. In the presented process, only registered "citizens" are allowed to cast a vote, hand in proposals, and participate in discussions, while registration as a government employee is necessary to validate and audit the proposals and moderate the discussions.

As already stated, PB initiatives differ widely depending on the jurisdictional and cultural backgrounds. The instantiation of the application of the PB process is, thus, always bound to local customization. A detailed description of possible functions along the exemplary process of such a system is given in [20].

The *Data Layer* is the gateway for accessing the databases and storing the documents needed for the applications to fulfill their tasks. The layer is proposed by [7,10,11,13–15]. The data can originate from systems like an ERP, the central governmental resident register, a PB database, or other application-specific databases. These databases then build upon the technologies like relational-like databases, document-centric NoSQL databases, or even Blockchains, e.g., for ensuring secure voting. An audit trail provides additional security by logging access to security-sensitive data [16].

The *Infrastructure Layer* is responsible for provisioning the computational resources and networking infrastructure of the services. The papers upon this reference architecture is built regard this layer as the provision of servers, networks, data centers, hardware, and more [8,19]. The full or partial virtualization of these parts of infrastructure through the use of cloud providers can reduce IT costs while simultaneously improving reliability. A recent literature study on the effects of cloud computing for eGov outlined its potential [21]. Even though no paper was identified as relevant for eGov architectures that included a cloud infrastructure, this might be due to the often older literature and the recent emergence of cloud technologies, combined with the hesitant adaption of new technologies in governmental branches. As cloud computing and the (at least partially) infrastructure virtualization is expected to rise in the upcoming years [22], we included this technology in the reference architecture as a possible deployment model.

The proposed infrastructure layer is based on the NIST definition for cloud computing [23]. It differentiates four deployment models: In a public cloud, a provider shares computational resources with the general public, and the servers are on the premises of the cloud providers. In a private cloud, the virtualization infrastructure is run entirely for (and also possibly by) a single organization. The community cloud provides a shared infrastructure for exclusive use by selected users (e.g., a governmental cloud run by the state for administrations). The hybrid cloud combines two or more infrastructures, e.g., private and public or community and public. This combination allows the execution of non-sensitive tasks in a public cloud while sensitive data remains within the premises.

[2] In a SoA, the PB application is merely one service of many in the government. Even though it can be integrated with other services, it has a high degree of independence within the system. (As an example, it is not necessarily bound to the same programming language as the web portal.) It just needs the interfaces to get integrated with the already existing system landscape.

There are three service models regarding the cloud virtualization level: In a Software as a Service (SaaS) model, the organization orders the whole software from a provider and pays per use (e.g., per active user). The provider takes care of the provisioning, licensing, and updating of the services. Even though some applications can also be installed in a private cloud and managed through a provider, these services normally run off-premise. The provider handles everything except the installed software in a Platform as a Service (PaaS) model. The managed services include the operating system, runtime environments, and servers. In an Infrastructure as a Service model (IaaS), the provider handles the servers, networks, and virtualization infrastructure. Here, the consumer controls everything from the operating system to the application. However, eGov applications can also run On-Premise without a cloud virtualization infrastructure. Please note that the decision for or against running the infrastructure in the cloud is not binary. It is also possible to run just specific, new applications on this virtualization infrastructure. Most likely, some SaaS application is already in use, like Microsoft Office 365 or Zoom.

4. Conclusion

Managing complex IT architectures is a tedious and complex task, as many systems with many interfaces need to interact and share data. Without structuring, these systems can uncontrollably grow out of control, making them costly and hard to maintain. Reference architectures can support the structuring tasks by providing blueprints. This report presented a reference architecture based on existing literature for integrating a PB application into an eGov landscape. It is targeted at administrations who plan to implement a PB initiative. The reference architecture shall enable these administrations to (1) assess their current eGov architecture, (2) identify the technologies needed to create a PB, (3) identify possible integrations for a PB application into the existing landscape, and (4) adapt the proposed reference architecture to their individual needs.

Even though the systematic literature analysis of this paper ensures that the built reference architecture is based on the already existing body of knowledge, it is not yet validated in practice. Thus, future research directions should be concerned with validating the proposed reference architecture in real-life examples.

5. Acknowledgments

The authors are part of the EU-funded EmPaci project (Empowering Participatory Budgeting in the Baltic Sea Region). This project is part of the European Regional Development Fund.

6. References

[1] A. Shah, Participatory budgeting, World Bank, Washington DC, 2007.

[2] A. Reiz, M. Fellmann, L.-M. Sinervo, A. Heinonen, Selecting the Right Software for Supporting Participatory Budgeting in Local Government – Reviewing Suitable Solutions, in: Electronic Governance and Open Society, St. Petersburg, Russia, 2021.

[3] L. Khalid, Software Architecture for Business, Springer International Publishing, Cham, 2020.

[4] S. Angelov, J.J.M. Trienekens, P. Grefen, Towards a Method for the Evaluation of Reference Architectures: Experiences from a Case, in: R. Morrison, R. Morrison, D. Balasubramaniam, K. Falkner (Eds.), Software architecture: Second European conference, ECSA 2008, Paphos, Cyprus, September 29-October 1, 2008 ; proceedings, Springer, Berlin, Heidelberg, 2008, pp. 225–240.

[5] B. Kitchenham, Procedures for Performing Systematic Literature Reviews. Joint Technical Report, 2004.

[6] C. Alfaro, J. Gomez, J.M. Lavin, J.J. Molero, A configurable architecture for e-participatory budgeting support, JeDEM 2 (2010) 39–45. https://doi.org/10.29379/jedem.v2i1.18.

[7] M.Z. Murah, Z. Abdullah, R. Hassan, M. Abu Bakar, I. Mohamed, H. Mohd Amin, Kacang Cerdik: A Conceptual Design of an Idea Management System, IES 6 (2013). https://doi.org/10.5539/ies.v6n6p178.

[8] BA. Baheer, D. Lamas, S. Sousa, Towards Development of a Reference Architecture for E-government, in: Proceedings of the 11th International Conference on Theory and Practice of Electronic Governance, Galway Ireland, ACM, New York, NY, 2018, pp. 640–643.

[9] E. Tambouris, E. Kaliva, M. Liaros, K. Tarabanis, A reference requirements set for public service provision enterprise architectures, Softw Syst Model 13 (2014) 991–1013. https://doi.org/10.1007/s10270-012-0303-7.

[10] S.S. Hilabi, F.L. Gaol, T. Matsuo, Enterprise Architecture Design For Regional Parliament Information Systems Using the Standard Government and Architecture Application (SAGA) Framework, ICIS Express letters 15 (2021) 178–181. https://doi.org/10.24507/icicel.15.02.173.

[11] R.N. Goddy-Worlu, C.K. Ayo, V.O. Geteloma, A sustainable ubiquitous engagement platform for open government Implementation, J. Phys.: Conf. Ser. 1378 (2019) 42037. https://doi.org/10.1088/1742-6596/1378/4/042037.

[12] A. Abdullah, A.-K. Zuhoor, N. Kraiem, Y.A. Jamoussi, Enhanced eGovemment integration framework for higher interoperability in eGovernment initiatives, in: 2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT-2017), Kannur, IEEE, Piscataway, NJ, 2017, pp. 1322–1329.

[13] M. Al-Husban, C. Adams, Connected Services Delivery Framework, in: Z. Mahmood (Ed.), Emerging mobile and Web 2.0 technologies for connected e-government, IGI Global, Hershey, Pennsylvania (701 E. Chocolate Avenue, Hershey, Pa., 17033, USA), 2014, pp. 50–75.

[14] V. Ciobanu, D. Popescu, C. Dobre, F. Pop, V. Cristea, The EU-Services Directive: An E-framework to Optimize Public Administration, in: 2012 Third International Conference on Emerging Intelligent Data and Web Technologies (EIDWT 2012), Bucharest, Romania, IEEE, Piscataway, NJ, 2012, pp. 336–341.

[15] W. Yanyan, Research on E-government Technical Framework Systems, in: 2010 International Conference on E-Business and E-Government, Guangzhou, China, IEEE, 5/7/2010 - 5/9/2010, pp. 707–710.

[16] B. Chakravarti, V. Varma, An enterprise architecture framework for building service oriented e-governance portal, in: 2008 IEEE Region 10 conference, Hyderabad, India, IEEE, Piscataway, NJ, 2008, pp. 1–6.

[17] F. Corradini, F. de Angelis, C. Ercoli, A. Polzonetti, A Services Management System for Small and Disadvantaged Communities, in: 2007 29th International Conference Information Technology Interfaces, Cavtat, Croatia, IEEE Service Center, Piscataway, NJ, 2007, pp. 477–482.

[18] D. Yang, Y. Han, J. Xiong, eGovGrid: A Service-Grid-Based Framework for E-Government Interoperability, in: W. Wang (Ed.), Integration and innovation orient to e-society: Seventh IFIP International Conference on e-Business, e-Services, and e-Society (I3E2007), October 10-12, Wuhan, China, Springer, New York, NY, 2007, pp. 364–372.

[19] Z. Ebrahim, Z. Irani, E-government adoption: architecture and barriers, Business Process Mgmt Journal 11 (2005) 589–611. https://doi.org/10.1108/14637150510619902.

[20] Rostock University, Feature Repository & Recommendations: Group of activities 4.1: Output 3. Project: "Empowering Participatory Budgeting in the Baltic Sea Region – EmPac", Rostock, Germany, 2020.

[21] M. Fathey, F. Olayah, Ali, Abdullah, Gazem, Nadhmi A., The Effect of Cloud Computing Adoption on the Sustainability of E-Government Services: A Review, International Journal of Advanced Science and Technology (IJAST) 29 (2020) 2636-2642.

[22] Statista, Worldwide information technology (IT) infrastructure spending breakdown (by value) from 2014 to 2024, by deployment type, 2021. https://www.statista.com/statistics/486586/it-infrastructure-spending-forecast-by-type/ (accessed 9 August 2021).

[23] P.M. Mell, T. Grance, The NIST definition of cloud computing, National Institute of Standards and Technology, Gaithersburg, MD, 2011.