Article

An Event-Driven Service Oriented Architecture Approach for E-Governance Systems

Kardan Journal of Engineering and Technology
1 (1) 1–10

Kardan Publications
Kabul, Afghanistan
DOI: 10.31841/KJET.2021.1
https://kardan.edu.af/Research/Kardan_journal_of_engineering_and_technology.aspx#

Majid Ayoubi
Ali Rehman Shinwari
Mohammad Sabaoon Khan

Abstract

The use of Information and Communication Technology (ICT) in industries have brought a significant increase in productivity and reliability of the systems, especially in Electronic-Governance. Meanwhile, the governmental organizations try to automate and accelerate the services for the citizens using ICT. The Event-Driven architecture has brought new essences of service automation empowered by Complex Event Processing (CEP) engines. In this architectural design approach, we proposed a novel Event-Driven architecture supported by integrated service bus to provide diverse system integration for the heterogeneous data of multiple types of organizations. The system is automated using CEP engine which is taking the data from different modules for analysis. The architecture evaluation shows that our model is comprised of the characteristics needed for a noble architectural model.

Keywords: *Information & Communication Technology, Service Oriented Architecture, E-Governance, Event Detection, Complex Event Processing*

Mr. Majid Ayoubi, Mr. Ali Rehman Shinwari and Mr. Mohammad Sabaoon Khan *are Assistant Professors at Faculty of Engineering and Technology, Kardan University, Kabul, Afghanistan.*

Introduction

E-governance or Electronic governance is to use the Information and Communication Technology (ICT) in order to enhance the governance in both the public and private sectors [1]. The e-governance service delivery systems adopted by different countries are: Intra Governmental System (G2G): this system enables different governmental bodies to communicate efficiently and carry out a task with the help of online electronic means, Government and Business (G2B and B2G): the purpose of this structure of the services is to establish the services 'bridge between the governmental bodies and businesses, Government services to Citizens (G2C & C2G): This service delivery system is based on government-citizen relationship. The citizens are represented as a customer who is supposed to use the services of the government. This system contains various types of Customer Relationship Management processes. Services such as educational documents attestations, online passport application, property registration, applying for different sort of certificates, tax payment, and etc, and Government and Foreign affairs (G2X): there are many governmental services that can be offered online for foreigners, it could be foreign trade, tourism, cultural and academic exchange programs [2].

In order to deliver the abovementioned services, there should be an integrated, compatible, reusable, autonomous, loosely coupled, and interoperable systems. The traditional e-governance peer to peer systems caused many other complex problems. In addition, a unified environment is needed in order to efficiently integrate all heterogeneous public information system and business processes. Therefore, the governments shall provide a single login portal for access to all its E-governance services [3].

Service-oriented architecture makes the distributed application development easier, rapid and cost-effective. SOA uses services as a basic block, even in a heterogeneous environment. Services are autonomous, platform-independent and they can be described, published and discovered. Services also can be assembled in a dynamic way to develop massive, interoperable, and evolving systems [4].

Service-Oriented Architecture (SOA) is an architectural approach that is used to provide better services to users [5]. Normally, services are carrying some functional portions of a specific process. The composability of the SOA services enables this architecture to provide boundless service delivery in integrated environments. The traditional stand-alone or central architectures cannot meet the business and organizations' requirements.

The reusability feature of SOA makes this approach as most adaptive and flexible.

Event-Driven Architectures is promoting service functionality to provide more productive intelligent and automated services [6]. The SOA 2.0 or Event-Driven SOA combines the Event-Driven architectures which are proactive with the SOA which provides the environment for service orchestration. The Event-Driven Architectures are used in modern software systems to provide actions based upon the specific criteria matched.

In this work, we proposed an Event-driven Service Oriented Architecture that is driven by automated events. In this architecture, events are empowered by Complex Event Processing (CEP) engine. In addition, the proposed system consists of a security server, query dispatcher, and integrated service bus. The CEP module makes the proposed system different from other SOA based systems. CEP engine is responsible for the system functionality.

This paper, generally, incorporates to combine new technologies such as, SOA and Event-Driven Architecture with new security systems for SOA, to provide, automated, secure and highly integrated software system for E-Governance.

Paper is organized as followed: Section II covers the related work; section III addresses the proposed architecture; section IV explores the system evaluation; section V and VI covers Conclusion and future work respectively.

2 Related Work

SEDA-SOA represented by Ayoubi et al. [7] brings forward a novel event-driven software architecture using enterprise service bus for software integration. Their work is scaled using staged event-driven architecture. We utilized their architecture to model an approach for e-government system. The SEDA-SOA architecture is based on SOA and it is not application specific. In their architecture, the security of the system is not considered.

Roy et al. [8] proposed a software architecture which is connected to the internet cloud. The architecture consists of three main modules, the Citizen side, the Government side, and the Bank side. The architecture is supported by the internet cloud for secure data transmission and storage. The overall architecture is secured using firewalls which are installed on each connection point with the other modules. The Cloud-Governance or c-governance module is providing the basic citizen's services after the

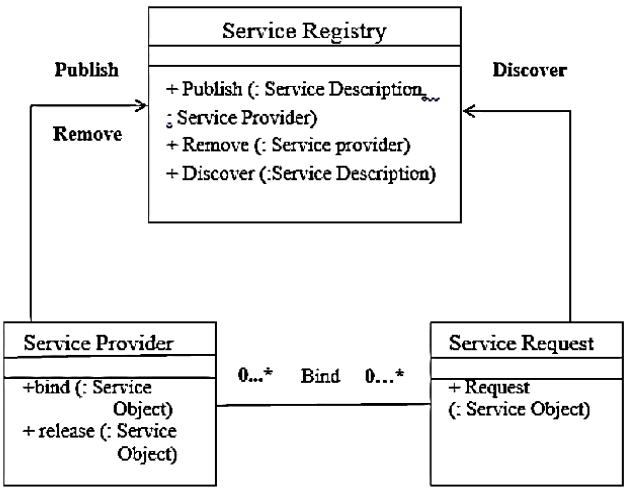
successful verification of the citizen. Their architectural system is well-designed for security and maintenance of the government data. They did not consider the system automation which is vital for current e-government systems.

An e-government integrated system with SOA is proposed by Gitau et al. [9] They have observed different problems of Kenya's e-government systems and suggested an e-Citizen system which can provide single portal for government services. After boarder discussion on software integration, they concluded that the Service Oriented Architecture can provide high capability for integration of heterogeneous software systems. In the e-Citizen system, the information of citizens is confirmed by Integrated Population Registration service. Afterwards, the user requests for specific government service. Their work is providing an SOA based architecture to enable software integration for the Kenya's e-government systems.

G. Behara [10] proposed "Next Generation Reference Architecture for Connected Government". This reference architecture is an architectural design which interconnects frameworks of business processes, data, applications and technology for the Government Transformations. They have identified the common pitfalls of the current government architectures. Pitfalls includes lack of flexibility and scalability, and the use of legacy systems. The government service transformation elements are used to boost the quality of life of citizens through revolutions of processes and technologies.

Recently, SOA has become famous by its adaptability with the integrated software systems. The SOA can integrate the heterogeneous and distributed software systems to provide a single representation of software services. The SOA has three main software service components which includes, 1. Service Registry, 2. Service Provider and 3. Service Request [5]. In SOA the service can be defined as "an encapsulated unit of functionalities" [5]. There is interaction between different software parties in the Service Oriented Architecture. The service provider publishes the description of the services and provides the implementation details for the service consumers. The service consumer utilizes the services available at the service registry by requesting the service from the service provider, Fig 1.

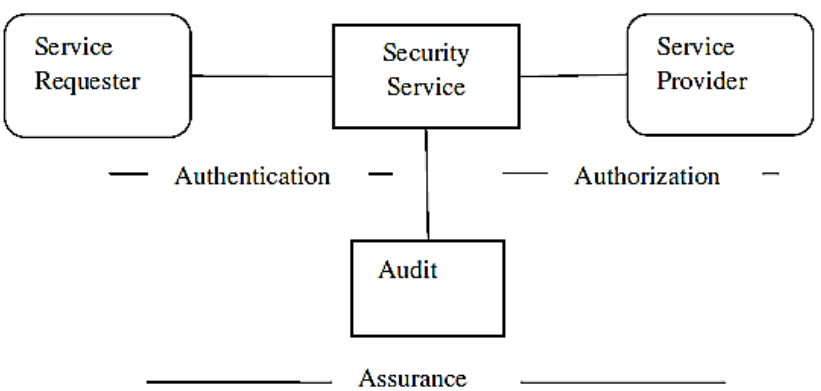
Figure 1: Three Principles of SOA Services



Source: [5]

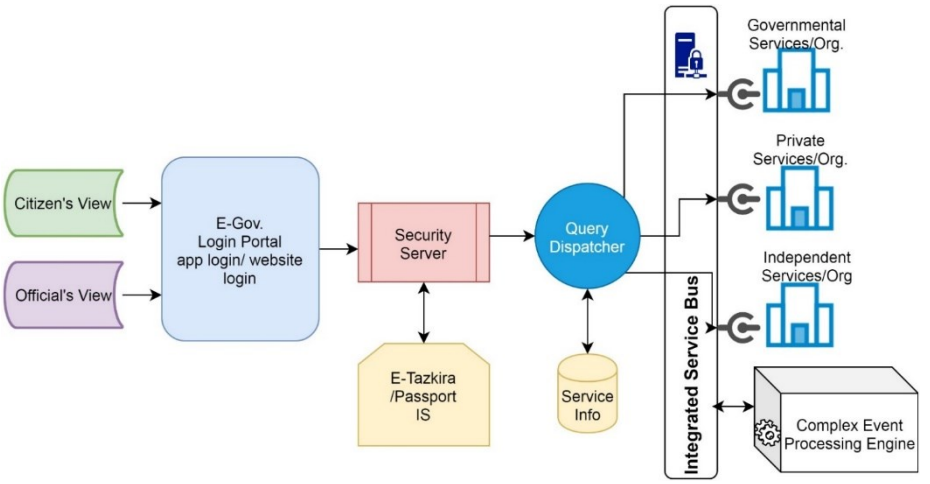
Saudi Arabia launched the e-governance program in 2005 [11]. They named the program as Yesser, and the objective of the program was to provide services to the public. A wide range of services are provided through that project. The architecture of Yesser has been categorized into three layers that are Front-end layer, Middle layer, and Back-end layer. The front-end layer contains the user interfaces such as, portals, informative and operational websites. The middle layer uses interaction toolkits to build the communication line between e-government services and users. The integration bus enables e-governance infrastructure's interoperability. The back-end layer is the hardware infrastructure containing clusters, servers, midrange systems, and mainframe systems that provides data services to the user.

Figure 2: The Security Server Architecture with SOA



Source: [12]

Figure 3: The Event-Driven SOA architecture for E-Governance



Source: Author's Compilation

Oldooze Karimi [12] proposed a model to enhance the security of Service-Oriented Architectures. He suggests that the use of combined security approach can ensure a protected SOA. In opposite to the central systems, the SOA systems are distributed and integrated. These systems demand different security features. He emphasizes that, for heterogeneous systems, each end connection should be secured using the recent web services security standards such as Security Assertion Markup Language (SAML) or WS-Security for exchanges of the messages. He proposed a secure architecture for SOA, which comprises of a secure server between the service requesters and service providers, Figure 2. We utilized this approach to enhance the security of our architecture.

3 Proposed Software Architecture

3.1 Citizen/Officials View

This module refers to the types of system users who are having different kinds of services and accessibility to the resources. Citizen's View is designed for normal citizens to have e-administration facilities. They can access different public, private or independent organizations' services. The Officials' View refers to the system login by the government officials which in turn need to verify the requests of the citizens and provide a digital signature for it.

3.2 E-Government Login Portal

The login portal authenticates the users by the username and passwords followed by their security code for the E-Tazkira identification. The portal is designed to provide an interface for all the services available by different organizations. The E-Tazkira information is checked and

verified by the database to ensure the identity and to proceed with the request further.

3.3 Security Server

For Service Oriented Architecture as being highly distributed and integrated, the vulnerability of security issues is high. To ensure the security of the architecture, we provided a single point security server [12] to provide unique security certificates for the whole journey of request traveling throughout the system.

3.4 Query Dispatcher and Service Information

Using the service information from the service database, the query dispatcher identifies different organizations' services corresponding user requests. The service information database maintains the portals based on the user's request.

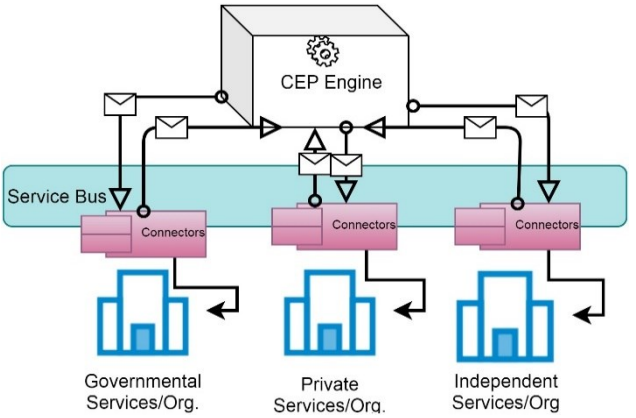
3.5 Integrated Service Bus

This module works as a backbone channel for the exchange of information between different organizations and the Complex Event Processing Engine. The SOA systems' data is highly distributed and is heterogeneous. It is needed for each service to have a specific connector for the built-in security mechanisms such as Security Assertion Markup Language SAML. SAML works as an XML framework for exchanging authenticated and authorized information used for the web services [12]. Beside the connectors' security consideration, the Bus needs to unify the content to be understandable for the rest of components and to the CEP engine. This feature provides interoperability for the services. The service bus, in turn, converts the retrieved data to the unique data type and submits it to the CEP engine for event detection and automated service delivery.

3.6 Complex Event Processing (CEP) Engine

The CEP engine takes the functional responsibility of the e-government architecture. The service bus provides the information to the CEP engine to infer the events which are getting generated from the complex patterns [11]. These events are suggesting further circumstances. The engine is used to analyze different types of information provided by the service bus, from multiple organizational services. The service bus retrieves the raw data or submits event results to the other components. The CEP engine provides automated service delivery as it is enriched by artificial intelligence. The result of the CEP engine is published in the right-time/ real-time, Figure 4.

Figure 4: CEP Engine vs Organizations' Interactions



Source: Author's Compilation

4 Model Discussion

To evaluate software architecture, there are plenty of techniques on hand. The famous type of evaluation is called the Architecture Tradeoff Analysis Method (ATAM) [13]. These methods are observing the architecture in different stages such as early-stage, while the architecture is not implemented or after the implementation of it. Professionals suggest that the early-stage evaluation of a software system decreases the implementation risks of it. There are different aspects in which software can be evaluated.

In the ATAM software evaluation method, the following characteristics of the software are evaluated: Performance, Reliability, Availability, Security, Modifiability, Portability, Functionality, Variability, Sub-set ability, and Conceptual Integrity.

Our proposed architecture provides the majority of these characteristics in the best way possible, because the architecture is designed based on SOA principles. The followings are further discussions on different characteristics.

4.1 Security

The proposed architecture provides a single point security server to provide security certificates for requests. The connectors of the integrated service bus are also using security mechanisms according to their application requirements. The two-sided security of the architecture maintains a high level of trust in our proposed work.

4.2 Portability and Variability

The architecture is based on SOA, which is well-known for its ability to be portable and variable. This architecture can be easily modified for extension.

4.3 Reliability and Availability

The availability of the distributed systems is difficult to maintain, but if any of the related components is failed, it does not affect the whole system and only the failed component will not be able to continue.

4.4 Conceptual Integrity

It is nearly impossible for the complex tasks to be based on a single machine. Instead, the task or service can be divided into many sub-systems which in turn all the sub-systems unify their process output. On the other hand, this leads to a consistency problem. Our proposed architecture maintains the consistency of information by the CEP engine which is frequently engaged to different connected entities such as government or private services. CEP engine works as a master controller over the different services, Figure 4.

5 Conclusion

In this paper, we have approached a new software architectural model to provide event-driven e-government services. The users can log in from different perspectives as citizens and officials. The security server makes sure to provide the access certificates to the right user. The Complex Event Processing engine monitors the system automation supported by an integrated service bus for interoperability. The model discussion shows that our proposed system can provide the characteristics required for a novel architecture defined by the software evaluation methods.

6 Future work

Whereas, our system can provide the characteristics of a well-designed software architecture, it is also responsive to provide good scalability for the user requests. Due to increased number of users, the performance of the architecture may not maintain to be real-time and to tackle humongous amount of user requests. The architecture can be designed to provide higher measures of scalability to support exceedingly number of users' requests.

References

[1] Thornton, "Sugar cane to base oil: Another game changer," *Fuels and Lubes International,* vol. 20, no. 4, pp. 32-36, 2014.

[2] Rani ,G. ,Chakraverty, S., "Survey of E-Governance Systems with focus on Development Approaches and Interface Quality," *International Journal of InterScience Management Review,* vol. 2, no. 2, pp. 34-42, 2012.

[3] Zakaria I.Saleh, Rand A. Obeidat, Yaser Khamayesh, "A Framework for E-Government based on Service Oriented Architecture for Jordan," *Information Engineering and Electronic Businesses,* vol. 3, no. 2, pp. 1-10, 2013.

[4] Papazoglou, M. P. , Traverso, P., Dustdat, S., Leymann, F., "Service Oriented Computing: a research roadmap.," *Internatioinal Journal of Cooperative Information Systems,* vol. 4, no. 2, pp. 200-234, 2008.

[5] Ayoubi, M., Ravichandra S., "SEDA-SOA: A Scalabale Event-Driven Context-Aware Service Oriented Architecture," in *ICCNT*, Banglore, India, 2018.

[6] Rama Krushna Das, Manas Ranjan Patra, "A Service Oriented Architecture for E-Governance Systems," *International Journal of Information Technology Convergence and Services,* vol. 2, no. 2, pp. 12-18, 2010.

[7] A. A. Al-Rashedi, "E-Governmenet based on Cloud Computing and Service Oriented Architecture," *International Journal of Computer and Electrical Engineering,* vol. 6, no. 3, pp. 201-207, 2014.

[8] R. K. Das, "Service Oriented Design approach for E-Governance Systems," *International Journal of Technology Convergence and Services,* vol. 3, no. 3, pp. 1-11, 2013.

[9] Abdul Mujeeb Mohmand, Aimal Marjan, Amirzai Sangin, "Developing E-Governance in Afghanistan," in *International Conference in E-governance systems*, Beijing, China, 2010.

[10] Nirmala Sugirtha Rajini, T. Bhuvaneswari, "Mapping Service Oriented Architecture and E-Governance," in *International Journal of Computer Applications*, Mumbai, India, 2012.

[11] O. Karimi, "Security Model for Service Oriented Architecture," *Advanced Computing, International Journal,* vol. 2, no. 4, pp. 48-56, 2011.

[12] D. Wang, "A Novel Complex Evenent Processing Engine for Intelligent Data Analysis," *International Journal on Network Systems,* vol. 3, no. 3, pp. 45-58, 2016.

[13] Rick Kazman, Mark Klein, Paul Celements, ATAM: Method for Architecture Evaluation, Pittsburg PA: Carnegie Mellon Software Engineering Institute, 2000.