Jordan University of science and technology

Department of Software Engineering

Graduation Project 1

BY

Nader Samir Alsouqi Ahmad Abdullah Alzamil

**Under supervision of DR. Hasan Alsheboul**

**Table of Contents**

SOA History………………………………………………………………………4

SOA Definition…………………………………………………………………...5

Motivation………………………………………………………………………..6

Problem Statement……………………………………………………………….7

Solution Plan……………………………………………………………………..8

About the Project ………………………………………………………………..9

Core Elements…………………………………………………………………...10

Functional Requirements………………………………………………………..11

This page intentionally left blank

**SOA History**

The idea of service-oriented architecture has been bouncing around the world of information technology for several years, however it emerged as a trend after several different pioneers had success with their own versions of what was essentially the same idea.

Early models provided more primitive definitions of what constituted an SOA.

Often the primary criteria was the involvement of a service registry and a discovery process. Since then, SOA has become a sophisticated and refined architectural model closely tied to a distinct design paradigm called service-orientation.

One of the first attempts to define an architecture that would allow monolithic to be decomposed into a number of discrete functions that could be accessed using sort of network api, was conceived by DEC in 1983 under project name “INFINET”.

A very successful adoption of service oriented architecture came in May 1995,

When wells Fargo bank completed a software project that allowed them to expand worldwide and became the first international bank in only 60 day.

The project succeeded with minimal cost and time because most of services needed to build the banking system has already been developed for another internal applications.

Note: some of the page contents are directly copied from “The 25 year’s history of SOA” article, credits goes to the original author Erik Townsend.

**SOA Definition**

Service Oriented Architecture (SOA) is a loosely-coupled architectural paradigm that allows building of Infrastructures for diverse application interaction and integration via services across different Platforms, domains of technology and locations.

SOA support transformation of solutions into a linked set of services that can be accessed over network.

Each service in SOA represent a self-contained unit of functionality, which can coalesce with other services to accomplish a specific task.

These services are interrelated through well-defined interfaces and contracts between the functional units of the application. The interface is defined in a manner that should be independent of hardware platform, operating system and programming language, to allow services to interact with each other in uniform and universal manner.

**Motivation**

The need to have a comprehensive, end to end architecture that works within and between enterprises had made the SOA preferred choice for IT and business organization to go to with their solutions.

SOA can help IT organization reduce their development and maintenance life cycle

Through the use of prebuilt services and encourage them to enhance their systems instead of replacing them.

And can also facilitate the process of establishing connections between business partners, customers and employees.

By treating elements of business process as modules to be reused and combined in a different ways, SOA provides organizations with extraordinary flexibility in addressing and prioritizing stakeholders needs.

By adopting SOA, businesses can realize lower costs, application sharing, flexibility, streamlined business processes, additional new revenue streams, and new business models.

**Araucaria**

A project that aims to develop a software system with the infrastructural components needed to allow its users to provide and consume software services over network based on SOA concepts.

**Project Goal**

Is to facilitate, organize, and standardize the adaption of service oriented architecture in software systems.

**Project Benefits**

* Dividing solution into a set of reusable modules.
* Facilitate service management and invocation.
* Encourage location transparency.
* Reduce the negative effects produced by centralized networks.
* Better parallelism in development.
* Better solution consistency and increased agility.
* Help IT and business organization to respond to market conditions in a much timelier manner.

**Problem Statement**

Software systems has always been a preferable approach for solving problems that occurs in many different fields. Some of these problems are simple, easy to solve and some of them are complicated and require sophisticated solutions to solve them.

But we can all agree that most of complicated problems are actually a group of sub-problems Coalescence together and represented in monolithic complicated problems.

Many problems have sub-problems in common, so providing a solution to sub-problem can be a part of solving other problems.

In software engineering we tend to divide solutions into sub-solution in order to reuse them to solve sub-problems that may reoccur. The result from these division can take many different shapes ranging from classes to libraries to services.

Our main concern in this project is services, providing solutions as services that can be consumed over network while maintaining software engineering quality attribute can be challenging due to the variation in software and hardware platforms, communication protocols and lack of standardization in this field.

**Solution Plan**

Realizing the flexibility and agility to respond to a rapidly changing environments SOA can give, made us realize the importance to provide a solid solution that satisfies stakeholder’s demands and grant them a competitive standing in their domain.

There are a lot of concepts and technologies that has been developed over the decades we can use in this project, however we will imply our knowledge and experience to develop a new protocols and technologies in order to take the solution to the subsequent level.

As we said before service is the key in this solution and defining a standard mechanism for publishing, managing, consuming and connecting services while providing a high level of flexibility and configurability to the stockholders to maintain a quality attributes in their solutions, will add integration and efficient architecting as values to the project.

**Core Elements**

Service provider: Agent who has the role of providing a service, which might then be consumed by other agents.

Service Consumer: Agent who has the role of consuming a service, which is provided by a service provider.

Composite Service: A type of services can request other services through the service consumer and work with them in cohesive manner.

Service Private Description: Service description file that has the necessary information application will need in order to invoke the service.

Service public Description: Service description file that has the necessary information service consumer will need in order to use the service and handle the result.

Service Static View: File contains implementation that is used to generate static output presentation to the service consumer.

Service Dynamic View: File contains implementation that is used to generate dynamic output presentation to the service consumer.

**Functional Requirements**

1. The application shall provide a public server provisioned for handling serves registration, deregistration and service lookup.
2. Service Provides should be able to register their services to a public server that is accessible to everyone over the network by providing the necessary information including the service name, input parameters needed to invoke the service, ip address of the server that provide the service and the port number.
3. Service providers shall be able to de-register their services from the server by providing the necessary information including the service name and the access key.
4. Service Consumers shall be able to look up for service by providing service name or service signature.
5. Service discovery server shall respond to the users service look up request and return the ip address and service signature of the requested service.