|  |
| --- |
| < Supervisor Change Web Service API> |

Application Program Interface

Aricent:500300105

Revision 1.0

ARICENT™ CONFIDENTIAL

|  |
| --- |
| **Copyright** © **2006 Aricent Inc. All Rights Reserved.** No part of this document may be reproduced, stored in a retrieval system or transmitted, in any form, or by any means, electronic or otherwise, including photocopying, reprinting, or recording, for any purpose, without the express written permission of Aricent.  **Printed in \_\_\_\_\_\_\_\_**  **TRADEMARKS** ARICENT and THE ARICENT LOGO are trademarks of Aricent Inc. in the U.S. and other countries. The use of any of these trademarks without Aricent prior written consent is strictly prohibited. Other trademarks and trade names may be used in this document to refer to either the entities claiming the marks and names or their products. Aricent Inc. disclaims any proprietary interest in the trademarks and trade names other than its own.  **DISCLAIMER The** information in this book is provided “as is”, with no warranties whatsoever, including any warranty of merchantability, fitness for any particular purpose or any warranty otherwise arising out of any proposal, specification or sample. This document is provided for informational purposes only and should not be construed as a commitment on the part of Aricent. Information in this document is subject to change without notice.  **REQUESTS** For information or obtaining permission for use of material of this work, please submit a written request to: Corporate Marketing and Legal, 3460 Hill view Avenue, Palo Alto, Ca. 94304.  **DOCUMENT No.: ARICENT** |
| **ARICENT™ Ltd.** Plot 31, Electronic City Sector 18, Gurgaon – 122015 Haryana (INDIA) Tel: +91-124-2346666/2455555 Fax: +91-124-2455100/2455101 E-mail: info@aricent.com Visit us at: <http://www.aricent.com> |

Contents

1. 1. Introduction 1

1.1 Purpose 1

1.2 Scope 1

1.3 Audience 1

1.4 Document Organization 1

1.5 Acronyms and abbreviations 2

1. 2. Background 3

2.1 Overview 3

1. 3. Solution Architecture 5

3.1 Solution Overview 5

1. 4. Solution Architecture and Design 9

4.1 Web Methods 10

4.1.1 searchEmployee 10

4.1.2 getSCEmployeeDetails 11

4.1.3 updateSCEmployeeDetails 12

4.2 Logging and Tracing 15

4.3 Sequence Diagram 16

1. Alternate Architectures 17

Your Opinion Matters 19

Revision History 20

Figures

Figure 3‑1: Web Service Overview 5

Figure 4‑1: Solution Diagram 9

Figure 4‑2: SOAP request for searchEmployee 13

Figure 4‑3: SOAP response for EmployeeSearch 14

Figure 4‑4: Sequence Diagram 16

Tables

Table 1‑1: Chapter Description 1

Table 1‑2: Acronyms and abbreviations used in this template 2

Chapter

1

# Introduction

## Purpose

This document explains the Supervisor Change web service API. The web service will expose web methods that will be consumed by the clients. Methods will be created based on the application requirement.

## Scope

The scope of this document is to give description of Supervisor Change Tool web service API.

## Audience

Primarily this document is targeted for development & the testing team.

## Document Organization

The document is organized as follows:

Table ‑1: Chapter Description

| Chapter | Name | Description |
| --- | --- | --- |
| 1 | Introduction | Provides a brief project introduction |
| 2 | Background | Gives an overview of Supervisor Change web service API |
| 3 | Solution Architecture | Describes overview of solution approach |
| 4 | Flow Diagram | It explains application flow of Web Service. |
| 5 | Sequence Diagram | It explains the flow of logic involved in web service. |
| 6 | References | References used to create this document |
| 7 | Document History | Document History |

## Acronyms and abbreviations

The acronyms and abbreviations explained in Table 1‑2 are fundamental to the information in this document.

Table ‑: Acronyms and abbreviations used in this template

| Acronym | Description |
| --- | --- |
| SOAP | Simple Object Access Protocol |
| WSDL | Web Service Description Language |
| XML | Extensible Markup Language |
| HTTP | Hyper Text Transfer Protocol |
| XSD | XML Schema Definition |
|  |  |

Chapter

2

# Background

## Overview

Aricent IT has many internal applications. Supervisor change is one of the functionality in RM tool as a web application, which is used by the Manager to change the supervisor of a particular employee. For the same an enterprise mobile application is developed which will provide the same functionality via mobile and other smart devices and we have a web service as an interface, which will interact with the database (SQL) and provide functionalities required by the application.

This document describes the Supervisor Change web service API.

Chapter

3

# Solution Architecture

## Solution Overview

Web service will be designed to expose web methods based on the requirement. These web methods will be consumed by clients. Basically this web service will be developed on .NET framework. Web Service will receive a SOAP request and in return it will send back the corresponding SOAP response. Web service will perform required validations & business logics on each method call before placing the DB call. Web service will perform necessary DB operation based on the business logic.

**Service Consumer / Receiver**

**Service Provider / Sender**

**SOAP Req / Res**

**Web Service**

**Client**

**Data Service**

Figure 3‑1: Web Service Overview

**Web Service:**

A Web service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-process able format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards.

**Service Consumers and Providers:**

The purpose of a Web service is to provide some functionality on behalf of its owner -- a person or organization, such as a business or an individual.

A requester entity is a person or organization that wishes to make use of a provider entity's Web service. It will use a requester agent to exchange messages with the provider entity's provider agent.

The provider entity is the person or organization that provides an appropriate agent to implement a particular service.

**Service Description:**

The mechanics of the message exchange are documented in a Web service description (WSD). The WSD is a machine process able specification of the Web service's interface, written in WSDL. It defines the message formats, data types, transport protocols, and transport serialization formats that should be used between the requester agent and the provider agent. It also specifies one or more network locations at which a provider agent can be invoked, and may provide some information about the message exchange pattern that is expected. In essence, the service description represents an agreement governing the mechanics of interacting with that service.

**SOAP:**

Simple Object Access Protocol is a protocol specification for exchanging structured information in the implementation of Web Services in computer networks. It relies on Extensible Markup Language (XML) for its message format, and usually relies on other Application Layer protocols, most notably Hypertext Transfer Protocol (HTTP) and Simple Mail Transfer Protocol (SMTP), for message negotiation and transmission.

A SOAP message is an ordinary XML document containing the following elements:

* An Envelope element that identifies the XML document as a SOAP message
* A Header element that contains header information
* A Body element that contains call and response information
* A Fault element containing errors and status information

**Data Service:**

Data as a Service brings the notion that data quality can happen in a centralized place, cleansing and enriching data and offering it to different systems, applications or users, irrespective of where they were in the organization or on the network. As such, Data as a Service solution provide the following advantages:

* Agility – Customers can move quickly due to the simplicity of the data access and the fact that they don’t need extensive knowledge of the underlying data. If customers require a slightly different data structure or has location specific requirements, the implementation is easy because the changes are minimal.
* Cost-effectiveness – Providers can build the base with the data experts and outsource the presentation layer, which makes for very cost effective user interfaces and makes change requests at the presentation layer much more feasible.
* Data quality – Access to the data is controlled through the data services, which tends to improve data quality because there is a single point for updates. Once those services are tested thoroughly, they only need to be regression tested if they remain unchanged for the next deployment.

Chapter

4

# Solution Architecture and Design

Windows Machine

**Web Service Provider**

(.NET 4.0 Framework, IIS 6.0+, SOAP,C# ADO.NET)

**Web Service**

**Method 1**

**Method 2**

**Method n**

**Internet / Intranet**

**Consumer**

**SOAP Req / Res**

**Data Base**

SQL

Figure 4‑1: Solution Diagram

The figure above depicts the logical flow/interaction between the consumer and web service

## Web Methods

### searchEmployee

**Web Method Detailed Description**

**Summary:** The web method fetches the employee ID and name of the employee from the database based on the search filters (Name) .There are two categories of search based on the flag value which is an input parameter.

1. Flag = 1 : Search result will contain all employee names that matches search characters(Name) and are placed under the manager( who is applying for the supervisor change).
2. Flag = 2 : Search result will contain all employee names that matches search characters(Name) and whose grade is E6 & above.

In Case of success : returns EmpId and EmpName along with status flag = 0 & message = “success”.

In Case of failure : returns Null values for EmpId and EmpName along with status flag = 1 & message = “No records found”.

**Signature:** searchEmplyoee (String Name, String Flag, String MgrId)

**Input parameters:** String EmpName, String Flag, String MgrEmpNo(optional).

**Output parameters:** String EmpId , String EmpName, String StatusFlag, String StatusMessage.

### getSCEmployeeDetails

**Web Method Detailed Description**

**Summary:** The web method fetches the details of the employee from the database based on the search filters employee ID and manager ID(new supervisor)

In Case of success : returns employee details along with status flag = 0 & message = “success”.

In Case of failure : returns Null values for employee details along with status flag = 1 & message = “No records found”.

**Signature:** getSCEmployeeDetails (String EmpId, String New\_MgrId)

**Input parameters:** String EmpId, String New\_MgrId.

**Output parameters:** All the field value required by the application.

### updateSCEmployeeDetails

**Web Method Detailed Description**

**Summary:** The web method updates the details of the employee in the database and returns status flag/message for the operation performed.

In Case of success : Status Flag = 0 and Status Message = “success”

In Case of failure : Status Flag = 1 and Status Message = “error”

**Signature:** updateSCEmployeeDetails (required parameters)

**Input parameters:** String EmpId, String EmpName, String MgrId, … (required parameters).

**Output parameters:** String StatusFlag, String StatusMessage .

**Soap Request : searchEmployee**

Figure 4‑2: SOAP request for searchEmployee

**Soap Response: searchEmployee**

Figure 4‑3: SOAP response for EmployeeSearch

## Logging and Tracing

Logging is used to record information about a program's execution for debugging and testing purposes. There are many logging tool for.Net eg: C#logger, Nlog, logger.NET, log4net etc. we would be using log4net tool for Supervisor Change Tool API development.

The Apache log4net library is a tool to help the programmer output log statements to a variety of output targets. log4net is a port of the Apache log4j framework to the Microsoft® .NET runtime.

**Log level**

The following defines the log levels and messages in log4net, in decreasing order of severity:

1. **OFF**

The highest possible rank and is intended to turn off logging.

1. **FATAL**:

Severe errors that cause premature termination. Expect these to be immediately visible on a status console.

1. **ERROR**:

Other runtime errors or unexpected conditions. Expect these to be immediately visible on a status console.

1. **WARN:**

Use of deprecated APIs, poor use of API, 'almost' errors, and other runtime situations that is undesirable or unexpected, but not necessarily "wrong". Expect these to be immediately visible on a status console.

1. **INFO**:

Interesting runtime events (startup/shutdown). Expect these to be immediately visible on a console, so be conservative and keep to a minimum.

1. **DEBUG**:

Detailed information on the flow through the system. Expect these to be written to logs only.

1. **TRACE**:

More detailed information. Expect these to be written to logs only.

The log file would be in txt format and the file size/rolling shall be configured in web.config of Asp.Net applications.

## Sequence Diagram

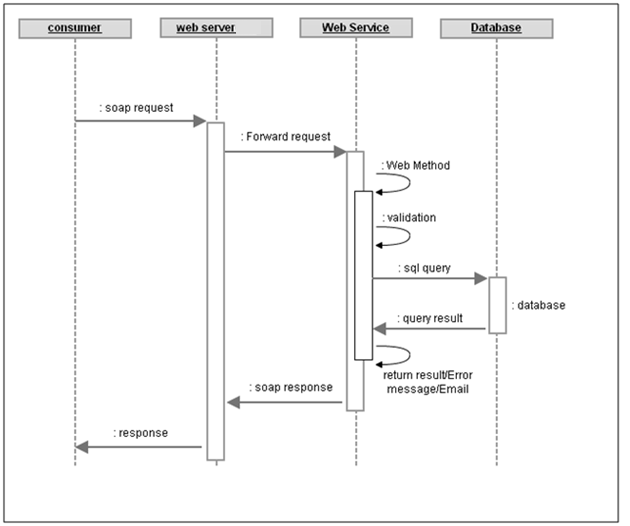
****

Figure 4‑4: Sequence Diagram

Appendix

A

###### Alternate Architectures

None.

Your Opinion Matters

Dear Reader:

We aspire to provide you with important information, delivered efficiently. If this guide has been informative, useful, and effective, your satisfaction is our reward. If not, tell us how it can be improved. Your valuable input will help us understand your information needs better.

Please fill up the form provided below and mail it to us at the address provided on the Copyright page.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Did you find the INFORMATION provided in this guide:** | | | | | | | |
| Complete | Y |  | N | Accurate | Y |  | N |
|  |  |  |  |  |  |  |  |
| Relevant | Y |  | N | Well-organised | Y |  | N |
| Other: | | | | | | | |
|  | | | | | | | |
| **Did you find the LANGUAGE used in this guide:** | | | | | | | |
| Easy to understand | Y |  | N | Correct | Y |  | N |
|  |  |  |  |  |  |  |  |
| Consistent | Y |  | N | Professional | Y |  | N |
| Other: | | | | | | | |
|  | | | | | | | |
| **Did you find the TONE used in the delivery:** | | | | | | | |
| Appropriate | Y |  | N | Professional | Y |  | N |
| Other: | | | | | | | |
|  | | | | | | | |
| **Did you find the ILLUSTRATIONS provided in the guide:** | | | | | | | |
| Relevant | Y |  | N | Easy to understand | Y |  | N |
| Other: | | | | | | | |
|  | | | | | | | |
| **Did you find the PAGE LAYOUT:** | | | | | | | |
| Aesthetically pleasing | Y |  | N | Crowded | Y |  | N |
| Other: | | | | | | | |
|  | | | | | | | |

What improvements do you recommend for this document?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Document No.

Thank you for participating in our quest for total customer satisfaction.

Revision History

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Author: | | | 1st Reviewer | | | Kumaresan Ettiyannan | | | | |
| Hari Haran Venugopal | | | 2nd Reviewer | | |  | | | | |
| 3rd Reviewer | | |  | | | | |
|  | | | | | | | | | | |
| Document Status: | | Initial | | | Internal | Draft | Approved | | Obsolete | Other |
| X | | |  |  |  | |  |  |
|  | | | | | | | | | | |
| Version | Date | | | Author | | | | Comments | | |
| 1.0 | 06/07/2012 | | | Hari Haran Venugopal | | | | Initial Draft | | |
|  |  | | |  | | | |  | | |
|  |  | | |  | | | |  | | |
|  |  | | |  | | | |  | | |
|  |  | | |  | | | |  | | |
|  |  | | |  | | | |  | | |
|  |  | | |  | | | |  | | |
|  |  | | |  | | | |  | | |
|  |  | | |  | | | |  | | |
|  |  | | |  | | | |  | | |
|  |  | | |  | | | |  | | |
|  |  | | |  | | | |  | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Rev | Date of Issue | Author | Approver | Scope |
|  |  |  |  |  |