STANDARD INTERFACE SPECIFICATION

Web Service Implementation

Specifications

Table of contents

1. Revision history

Version	Date	Revised by	Comments
1.0	12.3.05	Brad Barnett	Document created
2.0	12.4.07	Marek Bohacz	Reformatted

2. Related documents

Document	File location
N/A	N/A

3. Introduction

3.1. Background

PageUp People is a web based recruitment solution, allowing people to apply for jobs online, and be screened online by an organisation. PageUp provides a generic Web Service interface for various integration points into the PageUp People system.

3.2. Document purpose

The purpose of this document is to detail the method used to interface with PageUp People via its generic Web Service interface.

3.3. Document scope

This document is a detailed API that covers the methods used when sending data to, or retrieving data from PageUp People using its generic Web Service interface. It is intended to be used as a guide to accessing PageUp's Web Service infrastructure and should be used in conjunction with the relevant detailed interface specification for the particular integration point required to be accessed.

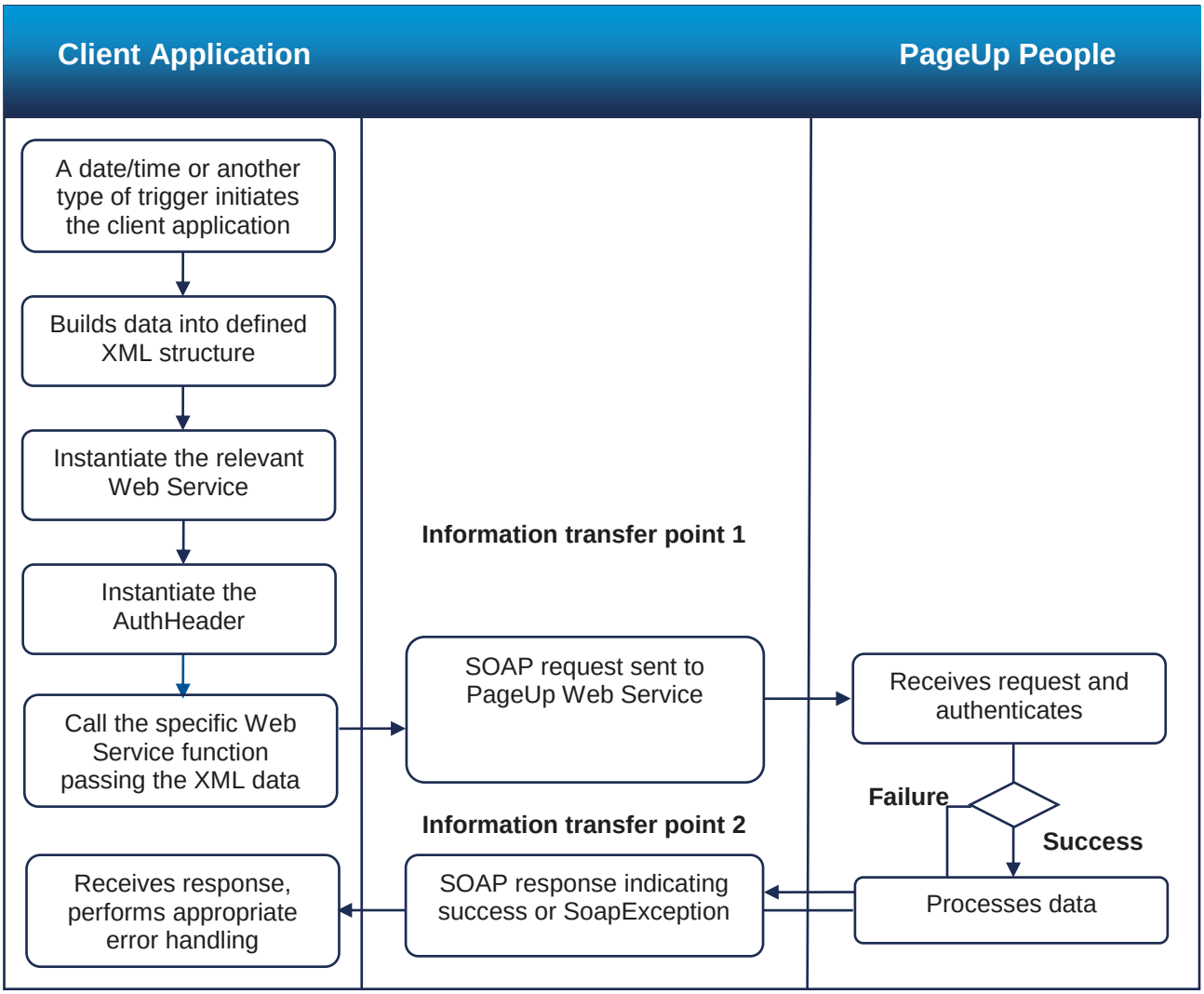
Acronyms and definitions

The table below defines abbreviations and terminology specific to this document

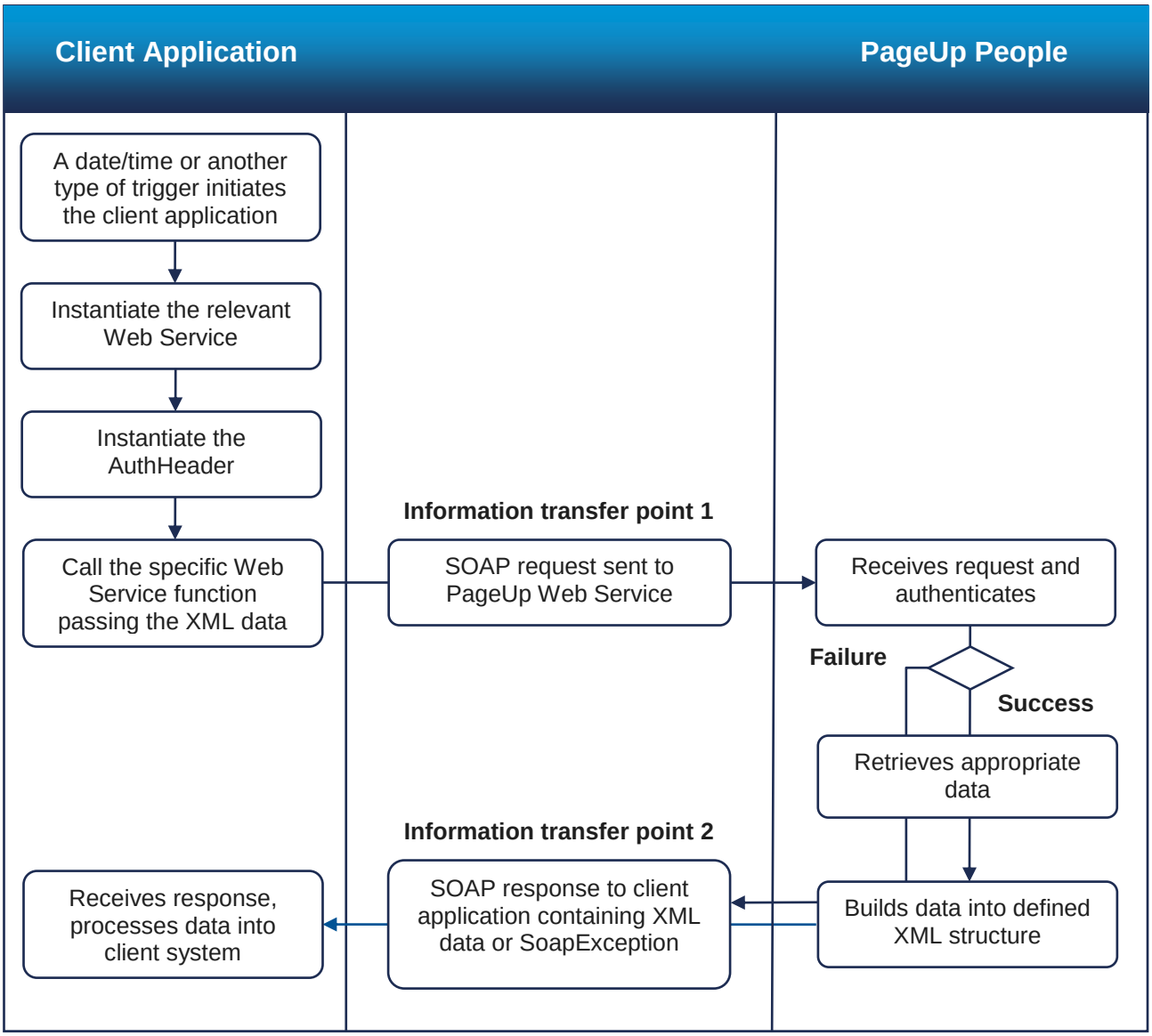
Technical term	Definitions
SOAP	Simple Object Access Protocol, the protocol that drives Web Services
XML	Extensible Markup Language is a standard for creating markup languages which describe the structure of data for data exchange.
WSDL File	Web Services Description Language is the standard format for describing a Web Service. A WSDL definition describes how to access a Web Service and what operations it will perform
Client application	The clients program that will access the PageUp Web Service interface when sending data to or retrieving data from PageUp People.

4. The Web Service process

4.1. Overview of the Web Service process when sending data to PageUp People



4.2. Overview of the Web Service process when retrieving data from PageUp People



5. Web Service authentication

All PageUp People Web Services require authentication using a username and password combination that will be provided during implementation. Authentication is performed using SOAP headers attached to each Web Service (SOAP) request. PageUp's implementation of this functionality is via an AuthHeader object that inherits from the .Net frameworks SOAPHeader object.

5.1. Authentication using the Microsoft.Net based client

- Add the reference to the WSDL file provided in the documentation for the Web Service into your VS.Net project
- Modify the sample C# code below to call and authenticate with the Web Service

```
using ExampleProjectName.ExampleWebServiceReference

...

...


GenericWebService oGenWebService = new GenericWebService();

AuthHeader oAuthHeader = new AuthHeader();


oAuthHeader.UserName = "yourUsername";

oAuthHeader.Password = "yourPassword";


oGenWebService.AuthHeaderValue = oAuthHeader;

string sResult = oGenWebService.WSAuthenticationExample();
```

In the example above, the function call to the WSAuthenticationExample() function can be substituted for the appropriate function call within the specific Web Service your are calling.

6. Error reporting

Errors will be returned to the client in the form of SoapExceptions conforming to the SOAP specifications. Details of the error, including the error number and message, can be found within the detail element of the SoapException object. For example, the following contains the XML of the detail element of an exception

```
<detail>
    <Error>
     <ErrorNumber>-100</ErrorNumber>
     <ErrorMessage>Access denied.</ErrorMessage>
    </Error>
</detail>
```

caused due to incorrect login details:

Specifications

To check for errors when calling the PageUp Web Service, you can wrap your code around a try {} catch {} and then check for SoapExceptions. The following C# code shows how to catch and handle these SOAP exceptions:

```
using System.Web.Services.Protocols;
using System.Xml;
...
...

try
{
      //Call PageUp Web Service
      ...
      ...
}
catch (SoapException soapEx)
{
      //Catch exceptions from the Web Service

      //Load the Detail element of the SoapException object
      XmlDocument oXMLDocument = new XmlDocument();
      oXMLDocument.LoadXml(soapEx.Detail.OuterXml);

      //get the error node from the XML
      XmlNode oXMLNode =
      XMLDocument.DocumentElement.SelectSingleNode("Error");

      //get the error number from the XML
      string sErrorNumber =
      oXMLNode.SelectSingleNode("ErrorNumber").InnerText;

      //get the error message from the XML
      string sErrorMessage =
oXMLNode.SelectSingleNode("ErrorMessage").InnerText;

}
catch (Exception ex)
{
      //catch non-soap exceptions
```

Specifications

The table below shows a list of the standard PageUp error codes and their meanings. These error codes exist across all PageUp Web Services:

Error number	Description
-100	Could not log in. The username or password provided in the AuthHeader was incorrect.
-101	An exception occurred on the PageUp side. PageUp is automatically notified of these errors.
-102	The XML sent to PageUp was missing some mandatory information. The XML should be validated against its WSDL before being sent to PageUp.