

**Implementation Guide
for the
ASAP Prescription Monitoring Program
Web Service Standard**

**Bidirectional Electronic Connections between Pharmacies, Prescribers,
and Prescription Monitoring Programs**

Version 1 • Release 1

January 2013

(1072013F)

American Society for Automation in Pharmacy

492 Norristown Road, Suite 160, Blue Bell, PA 19422

610/825-7783 (phone) 610/825-7641 (fax) www.asapnet.org

© Copyright 2013 American Society for Automation in Pharmacy. All rights reserved.

Table of Contents

| | |
|--|-----------|
| Introduction | 2 |
| Ad Hoc and Automated Polled Models | 2 |
| Web Service | 3 |
| Authentication | 3 |
| Supplied Credential Model | 3 |
| Ad Hoc Query | 4 |
| Detailed Ad Hoc Query | 4 |
| Reference Number Ad Hoc Query | 5 |
| Detailed Response to an Ad Hoc Query | 5 |
| Returning a Pick List | 7 |
| Automated Poll | 9 |
| Automated Poll Request | 9 |
| Detailed Alert Response to an Automated Poll | 9 |
| Reference Number Alert Response to an Automated Poll | 11 |
| Response to an Automated Poll When No Alerts Exist | 11 |
| Exceptions | 12 |
| Encryption | 12 |
| WSDL | 12 |
| Appendix | 16 |
| ASAP PMP Web Service Example Program to Implement the Standard | |

1 Introduction

It is the goal to have a method by which pharmacists and prescribers can more effectively communicate with prescription monitoring programs (PMPs). It is believed that more timely communications will have a positive impact on the reduction in diversion and abuse of prescription drugs.

To achieve this goal pharmacists and prescribers should be more easily informed of persons of interest. This can take one of two forms. The first allows the pharmacy management or EHR system to poll the PMP for alerts to providers (pharmacies/pharmacists and prescribers) that are then displayed directly into a pharmacy management or EHR system. Current practice is to send these alerts via fax, email, or regular mail. The second is to give the provider the option to do an ad hoc query of the PMP to retrieve information regarding recent activity on a person of interest (POI) directly from the pharmacy management or EHR system as opposed to using a Web portal.

It has been determined that the above options would better serve to increase use of the patient and prescriber information reported to PMPs. Therefore, a framework is needed that will allow an existing information system to be integrated with the information on file with the PMP.

To accomplish this ASAP has developed standards as explained in this document to facilitate the exchange of information between PMPs and providers.

2 Ad Hoc and Automated Polled Models

In the current environment, both ad hoc queries (or solicited requests) and unsolicited alerts are being employed. The ad hoc query model is when a physician or pharmacist would like to check on a person. This results in a query to the PMP. (In this document we use the term query to define this process.)

The unsolicited alert model is when a PMP sends out alerts to providers when a person of interest reaches the state's threshold for triggering such alerts. (In this document we refer to these as alerts.) A Web service is preferred for this communication framework. Since it is impractical for each PMP to maintain a list of pharmacies or prescribers with which it must communicate, it was decided that a polled model could be used for these alerts. This way the PMP can simply maintain a list of the alerts they would like to issue and the pharmacies and physicians can poll for this information on a periodic basis (typically daily).

In the automated polled model, the details provided on the person and prescriber will vary by state. This document includes examples of the details that can be sent but are not intended as the standard information that should be required. It is the structure of how the information is sent that the ASAP standard addresses. Same with queries activated by providers. State PMPs would determine which data elements are necessary in order to increase the probability of a match. If a pharmacy management or EHR system is programmed to send more information than a state requires to identify the person of interest, it can choose to ignore certain data elements in the query. For example, one state may find that including the person's phone number or ZIP code would increase the probability of a match, it will use these data elements or, if not, choose to ignore them.

When responding to the poll the PMP can include the details on the person of interest that triggered the alert. Optionally, the PMP can choose to respond with nothing more than a list of reference numbers. These reference numbers can then be used in a subsequent query to obtain the details on the POI.

Finally, when responding to an ad hoc query, the PMP may choose to respond with a list of possible matches and a value indicating the degree of certainty that a particular person is, indeed, the person being sought. This is commonly referred to as a "pick list" response. This two-step process functions as follows: The first step is the query transmitted with specifics on the person of interest. The state responds with the individuals that could be a match, assigning a weighting or probability of a match. The requestor would then select the person from the list provided that he or she wants information on and sends another request to the PMP for the prescription details on this person. The ad hoc query response provides for this functionality.

2.1 Web Service

Web services would be used to enable these communications.

The PMP will host a Web service with the following Web service actions:

| Action | Arguments | Type | Return Type |
|---|----------------|--------------|------------------|
| AdHocPMPRequest (Ad Hoc Query) | | | adhocPMPResponse |
| | userId | String | |
| | passwordDigest | base64binary | |
| | Nonce | string | |
| | timeStamp | dateTime | |
| | Request | PMPQuery | |
| | | | |
| PMPAlertAutomatedPoll (PMP Alerts) | | | PMPAlert |
| | userId | string | |
| | passwordDigest | base64binary | |
| | pharmacyId | string | |
| | Nonce | string | |
| | timestamp | dateTime | |

2.2 Authentication

Authentication may be completed using one of two approaches, namely the supplied credential model or the GFIPM Web Services System-to-System Profile, based on a trusted federation model. System vendors have agreed to support two authentication models: 1) where authentication happens at the pharmacy, for example, and 2) where authentication happens at the PMP.

2.2.1 Supplied Credential Model

In the supplied credential model the following data elements are used for authentication

| Element Name | Type | Value |
|-----------------------|--------------|---|
| userId | string | Unique name that identifies the user |
| passwordDigest | base64Binary | Base 64 encoded value of a SHA1 encryption of the nonce, the time stamp, and the password |
| Nonce | string | A one time use string used to create the password digest |
| Timestamp | dateTime | The date/time the password digest was created |

In this model, User ID and a Password Digest are included in the Web service call, along with a Nonce and a TimeStamp used to determine the password. The Nonce should be a unique string that is used only once. A good choice is a UUID field. The purpose of the Nonce and TimeStamp is to prevent “snoop and repeat” attacks.

Computation of the password digest is done by appending the Nonce, TimeStamp, and Password fields and performing a SHA-1 Hash on the result. The resulting hash is encoded in a base 64-array and passed to the PMP in the Web service call along with the Nonce and TimeStamp used to compute the password digest. On the host system the same calculation is performed using the expected password with the supplied Nonce and TimeStamp, and the resulting hash values are compared. Successful comparison means that the same password was used in each calculation, thereby verifying the supplied credential.

For example, given the following:

| | |
|------------------|--------------------------------------|
| Password | S3cur3P4\$\$w0rd |
| Nonce | 0F2ED1EA-2E78-48CC-9D22-C70A1FEB7615 |
| TimeStamp | 6/8/2012 at 12:59:00 PM |

Concatenating the nonce, timestamp, and password yields the following:

0F2ED1EA-2E78-48CC-9D22-C70A1FEB76152012-06-08T12:59:00ZS3cur3P4\$\$w0rd

And subsequent SHA-1 Hash yields the following password digest:

j2G3VF7YNkNnteuV2FVhEicMlhc=

2.3 Ad Hoc Query

The Ad Hoc Query can take one of two possible forms, a Detailed Request or a Reference Number Request. The Detailed Request would be used at the point of care when a provider would like to submit a query in reference to a specific patient or patients. Reference number queries are used when a reference number response is received from an automated poll.

The Request argument of the Ad Hoc Query is a Polymorphic XML argument, with the type of the object included as an attribute of the request argument. This will be demonstrated in the examples that follow. See the WSDL at the end of this document for specifics on the XML objects used.

2.3.1 Detailed Ad Hoc Query

In a Detailed Solicited Ad Hoc PMP Query, the specifics on the patient for which the pharmacist or prescriber is requesting information are included in the request. The following is an example of the SOAP request submitted by the provider.

```
<?xml version="1.0" encoding="UTF-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <AdHocPMPRequest xmlns="http://www.asapnet.org/pmprequest">
      <userId>user@somepharmacy.com</userId>
      <passwordDigest>ajJHM1ZGN11Oa05udGV1VjJGVmhFaWNNbGhjPQ==</passwordDigest>
      <nonce>0F2ED1EA-2E78-48CC-9D22-C70A1FEB7615</nonce>
      <ts>2012-12-07T14:12:47.8088824-05:00</ts>
      <req xsi:type="PMPDetailedQuery">
        <QueryDate>2012-12-07T00:00:00-05:00</QueryDate>
        <RequestDateRange>
          <DateRangeBegin>2012-01-01T00:00:00</DateRangeBegin>
          <DateRangeEnd>2012-07-01T00:00:00</DateRangeEnd>
        </RequestDateRange>
        <Patient>
          <BirthDate>1950-10-05T00:00:00</BirthDate>
          <Name>
            <GivenName>John</GivenName>
            <SurName>Doe</SurName>
          </Name>
        </Patient>
      </req>
    </AdHocPMPRequest>
  </s:Body>
</s:Envelope>
```

```

    <ContactInformation>
      <Phone>412-555-5555</Phone>
      <StreetAddress>123 Any Street</StreetAddress>
      <City>SomeCity</City>
      <LocationStateUsPostalServiceCode>PA</LocationStateUsPostalServiceCode>
      <LocationPostalCode>12345</LocationPostalCode>
    </ContactInformation>
    <Gender>M</Gender>
  </Patient>
</req>
</AdHocPMPRequest>
</s:Body>
</s:Envelope>

```

The “type” attribute of the <req> argument states that a detailed request is being submitted. Further, note that the <Patient> element provides the details of the patient for which the request is being submitted. Note: It is acceptable to use a “wild card” such as an asterisk in a truncated first name or middle name, if the PMP supports this.

2.3.2 Reference Number Ad Hoc Query

In a reference number ad hoc query, the reference number is sent to the PMP in order to retrieve the associated details. A reference number can be returned in the response to an automated poll when sending out PMP alerts, or reference numbers can be returned in a pick list response to an ad hoc query. In either case, a reference number ad hoc query is used to retrieve the associated details.

```

<?xml version="1.0" encoding="UTF-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <AdHocPMPRequest xmlns="http://www.asapnet.org/pmprequest">
      <userId>user@somepharmacy.com</userId>
      <passwordDigest>ajJHM1ZGN11Oa05udGV1VjJGVmhFaWNNbGhjPQ==</passwordDigest>
      <nonce>0F2ED1EA-2E78-48CC-9D22-C70A1FEB7615</nonce>
      <ts>2012-12-07T14:50:31.003803-05:00</ts>
      <req xsi:type="PMPRefNoQuery">
        <QueryDate>2012-12-07T00:00:00-05:00</QueryDate>
        <RequestDateRange>
          <DateRangeBegin>2012-01-01T00:00:00</DateRangeBegin>
          <DateRangeEnd>2012-07-01T00:00:00</DateRangeEnd>
        </RequestDateRange>
        <RefNo>9876543</RefNo>
      </req>
    </AdHocPMPRequest>
  </s:Body>
</s:Envelope>

```

The “type” attribute of the <req> argument states that a reference number request is being submitted.

2.3.3 Detailed Response to an Ad Hoc Query

When responding to an ad hoc query, either a detailed query or a reference number query, the PMP will respond with an AdHocPMPRequestResponse object. Shown below is a detailed response to an Ad Hoc Query.

Note that not all possible fields are included in the example that follows. The WSDL contained in section 2.7 shows all possible data elements.

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <AdHocPMPRequestResponse xmlns="http://www.asapnet.org/pmprequest">
      <AdHocPMPRequestResult>
        <ResponseDate>2012-12-12T00:00:00-05:00</ResponseDate>
        <Details>
          <PMPDetailedResponse>

```

```

<Patient>
  <BirthDate>1950-10-07T00:00:00</BirthDate>
  <Name>
    <GivenName>John</GivenName>
    <SurName>Doe</SurName>
  </Name>
  <ContactInformation>
    <Phone>412-555-5555</Phone>
    <StreetAddress>123 Any Street</StreetAddress>
    <City>SomeCity</City>
    <LocationStateUsPostalServiceCode>PA</LocationStateUsPostalServiceCode>
    <LocationPostalCode>12345</LocationPostalCode>
  </ContactInformation>
  <DriversLicensID>12-345-678</DriversLicensID>
  <PassportID>12345678901234567890</PassportID>
  <SSN>123-45-6789</SSN>
  <Gender>M</Gender>
</Patient>
<PrescriptionDetails>
  <PharmacyDispenseInfo>
    <Pharmacy>
      <DEANumber>AB12345678</DEANumber>
      <Location>
        <Phone>123-456-7890</Phone>
        <StreetAddress>312 Another Street</StreetAddress>
        <City>SomeCity</City>
        <LocationStateUsPostalServiceCode>PA</LocationStateUsPostalServiceCode>
        <LocationPostalCode>12345</LocationPostalCode>
      </Location>
    </Pharmacy>
    <Prescriptions>
      <DispensingEventInfo>
        <Prescriber>
          <Name>
            <GivenName>Jack</GivenName>
            <SurName>Black</SurName>
          </Name>
          <DEANumber>CD234567890</DEANumber>
          <NPI>1234567890</NPI>
          <Location>
            <Phone>(800)555-1212</Phone>
            <StreetAddress>999 Practice Street</StreetAddress>
            <StreetAddress2>Suite 100</StreetAddress2>
            <LocationStateUsPostalServiceCode>PA</LocationStateUsPostalServiceCode>
            <LocationPostalCode>12876</LocationPostalCode>
          </Location>
        </Prescriber>
        <DispensingEvent>
          <DispenseDate>2012-01-01T00:00:00</DispenseDate>
          <WrittenDate>2012-01-01T00:00:00</WrittenDate>
          <DrugName>DrugName</DrugName>
          <Strength>20MG</Strength>
          <DosageForm>TAB</DosageForm>
          <Quantity>0</Quantity>
          <DaysSupply>0</DaysSupply>
          <RefillStatus/>
          <PartialFillIndicator>0</PartialFillIndicator>
          <PaymentType>CASH</PaymentType>
          <DateSold>0001-01-01T00:00:00</DateSold>
          <RxSerialNumber>
            <IssuingState>PA</IssuingState>
            <SerialNumber>987654321</SerialNumber>
          </RxSerialNumber>
          <PickupOrDropOffPerson>
            <Name>
              <GivenName>Jane</GivenName>
              <SurName>Doe</SurName>
            </Name>
            <IDQualifier>06</IDQualifier>
            <ID>123-456-789</ID>
          </PickupOrDropOffPerson>
        </DispensingEvent>
      </Prescriptions>
    </DispensingEventInfo>
  </PharmacyDispenseInfo>
</PrescriptionDetails>

```

```

        <RelationshipCode>03</RelationshipCode>
      </PickupOrDropOffPerson>
    </DispensingEvent>
  </DispensingEventInfo>
</Prescriptions>
</PharmacyDispenseInfo>
</PrescriptionDetails>
<Messages>
  <string>Free form message</string>
</Messages>
<Summary>
  <NumberOfPharmacies>1</NumberOfPharmacies>
  <NumberOfPrescribers>1</NumberOfPrescribers>
  <NumberOfPrescriptions>1</NumberOfPrescriptions>
</Summary>
</PMPDetailedResponse>
</Details>
</AdHocPMPRequestResult>
</AdHocPMPRequestResponse>
</soap:Body>
</soap:Envelope>

```

In the event, there is no match that can be found, the following would be returned by the PMP.

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <AdHocPMPRequestResponse xmlns="http://www.asapnet.org/pmprequest">
      <AdHocPMPRequestResult>
        <ResponseDate>0001-01-01T00:00:00</ResponseDate>
        <Details/>
      </AdHocPMPRequestResult>
    </AdHocPMPRequestResponse>
  </soap:Body>
</soap:Envelope>

```

Pick List: In some cases the PMP may return a list of possible matches to an ad hoc query. The intent would be for the provider's system to display that list and prompt for a selection from the list that matches the provider's criteria. The following is an example of such a response with a pick list.

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <AdHocPMPRequestResponse xmlns="http://www.asapnet.org/pmprequest">
      <AdHocPMPRequestResult>
        <ResponseDate>2012-12-12T00:00:00-05:00</ResponseDate>
        <Details>
          <PMPDetailedResponse>
            <Patient>
              <BirthDate>1950-10-05T00:00:00</BirthDate>
              <Name>
                <GivenName>John</GivenName>
                <SurName>Doe</SurName>
              </Name>
              <ContactInformation>
                <StreetAddress>123 Any Street</StreetAddress>
                <City>Some City</City>
                <LocationStateUsPostalServiceCode>PA</LocationStateUsPostalServiceCode>
                <LocationPostalCode>12345</LocationPostalCode>
              </ContactInformation>
            </Patient>
            <PickListDetails>
              <WeightingFactor>98</WeightingFactor>
              <RefNo>124441</RefNo>
            </PickListDetails>
          </PMPDetailedResponse>
        </Details>
      </AdHocPMPRequestResult>
    </AdHocPMPRequestResponse>
  </soap:Body>
</soap:Envelope>

```



```

<Patient>
  <BirthDate>1950-10-05T00:00:00</BirthDate>
  <Name>
    <GivenName>John</GivenName>
    <SurName>Doe</SurName>
  </Name>
  <ContactInformation>
    <StreetAddress>123 South Any Street</StreetAddress>
    <City>Some City</City>
    <LocationStateUsPostalServiceCode>PA</LocationStateUsPostalServiceCode>
    <LocationPostalCode>12345</LocationPostalCode>
  </ContactInformation>
</Patient>
<PickListDetails>
  <WeightingFactor>90</WeightingFactor>
  <RefNo>124441</RefNo>
</PickListDetails>
</PMPDetailedResponse>
<PMPDetailedResponse>
  <Patient>
    <BirthDate>1950-10-05T00:00:00</BirthDate>
    <Name>
      <GivenName>John</GivenName>
      <SurName>Doe</SurName>
    </Name>
    <ContactInformation>
      <StreetAddress>30 Smith Road</StreetAddress>
      <City>Some City</City>
      <LocationStateUsPostalServiceCode>PA</LocationStateUsPostalServiceCode>
      <LocationPostalCode>02222</LocationPostalCode>
    </ContactInformation>
  </Patient>
  <PickListDetails>
    <WeightingFactor>20</WeightingFactor>
    <RefNo>124441</RefNo>
  </PickListDetails>
</PMPDetailedResponse>
<PMPDetailedResponse>
  <Patient>
    <BirthDate>1950-05-01T00:00:00</BirthDate>
    <Name>
      <GivenName>John</GivenName>
      <SurName>Doe</SurName>
    </Name>
    <ContactInformation>
      <StreetAddress>15A Barney Lane</StreetAddress>
      <City>Some City</City>
      <LocationStateUsPostalServiceCode>PA</LocationStateUsPostalServiceCode>
      <LocationPostalCode>02105</LocationPostalCode>
    </ContactInformation>
  </Patient>
  <PickListDetails>
    <WeightingFactor>15</WeightingFactor>
    <RefNo>124441</RefNo>
  </PickListDetails>
</PMPDetailedResponse>
<PMPDetailedResponse>
  <Patient>
    <BirthDate>1950-10-05T00:00:00</BirthDate>
    <Name>
      <GivenName>John</GivenName>
      <SurName>Doe</SurName>
    </Name>
    <ContactInformation>
      <StreetAddress>123 Any Street</StreetAddress>
      <City>Some City</City>
      <LocationStateUsPostalServiceCode>PA</LocationStateUsPostalServiceCode>
      <LocationPostalCode>12345</LocationPostalCode>
    </ContactInformation>
  </Patient>

```

```

    <PickListDetails>
      <WeightingFactor>75</WeightingFactor>
      <RefNo>124441</RefNo>
    </PickListDetails>
  </PMPDetailedResponse>
</Details>
</AdHocPMPRequestResult>
</AdHocPMPRequestResponse>
</soap:Body>
</soap:Envelope>

```

Note that, in this example, 5 possible matches to the query are returned. Each possible match includes a <PickListDetails> object. This object contains 2 elements, WeightingFactor and RefNo. WeightingFactor is an integer value between 1 and 100 that indicates the confidence that this person matches the criteria. RefNo is a string that can be used to pull back the details on the person that is considered a match.

2.4 Automated Poll

The Automated Poll is intended to support polling the PMP on a periodic basis, such as daily. The PMP can decide what information to communicate in its response. The information returned can take one of two forms. The PMP can either return detailed information on a person of interest, or it can return a reference number that the requestor would use in a subsequent ad hoc query to obtain the details on the person of interest.

2.4.1 Automated Poll Request

The following is an example of an Automated Poll Request where a provider is asking if there is anyone who reached the threshold that he or she should know about.

```

<?xml version="1.0" encoding="UTF-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <PMPAlertAutomatedPoll xmlns="http://www.asapnet.org/pmprequest">
      <userId>user@somepharmacy.com</userId>
      <passwordDigest>ajjHMI ZGN1IOa05udGV1VjJGVmhFaWNNbGhjPQ==</passwordDigest>
      <pharmacyId>1234567890</pharmacyId>
      <nonce>0F2ED1EA-2E78-48CC-9D22-C70A1FEB7615</nonce>
      <ts>2013-01-04T14:36:40.4771392-05:00</ts>
    </PMPAlertAutomatedPoll>
  </s:Body>
</s:Envelope>

```

2.4.2 Detailed Alert Response to an Automated Poll

This example shows a detailed response to a poll request. Note that the response is a polymorphic xml response. The “type” attribute of the “PMPResponse” object informs the caller of the type of response. Further note that the <Details> object holds a collection of <PMPResponse> objects. Multiple <PMPResponse> objects (i.e. multiple alerts) can be returned in response to a single poll. Further note that the <Messages> object is a collection of messages. Multiple messages can be returned at either the detail level, or overall.

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <PMPAlertAutomatedPollResponse xmlns="http://www.asapnet.org/pmprequest">
      <PMPAlertAutomatedPollResult>
        <ResponseDate>2012-12-12T00:00:00-05:00</ResponseDate>
        <Details>
          <PMPResponse xsi:type="PMPDetailedResponse">
            <Patient>
              <BirthDate>1950-10-07T00:00:00</BirthDate>
              <Name>
                <GivenName>John</GivenName>
                <SurName>Doe</SurName>
              </Name>
            </PMPResponse>
          </Details>
        </PMPAlertAutomatedPollResult>
      </PMPAlertAutomatedPollResponse>
    </soap:Body>
  </soap:Envelope>

```

```

    <ContactInformation>
      <StreetAddress>123 Any Street</StreetAddress>
      <City>SomeCity</City>
      <LocationStateUsPostalServiceCode>PA</LocationStateUsPostalServiceCode>
      <LocationPostalCode>12345</LocationPostalCode>
    </ContactInformation>
    <DriversLicensID>12-345-678</DriversLicensID>
    <PassportID>12345678901234567890</PassportID>
    <SSN>123-45-6789</SSN>
    <Gender>M</Gender>
  </Patient>
  <PrescriptionDetails>
    <PharmacyDispenseInfo>
      <Pharmacy>
        <DEANumber>AB12345678</DEANumber>
        <Location>
          <Phone>123-456-7890</Phone>
          <StreetAddress>312 Another Street</StreetAddress>
          <City>SomeCity</City>
          <LocationStateUsPostalServiceCode>PA</LocationStateUsPostalServiceCode>
          <LocationPostalCode>12345</LocationPostalCode>
        </Location>
      </Pharmacy>
      <Prescriptions>
        <DispensingEventInfo>
          <Prescriber>
            <Name>
              <GivenName>Jack</GivenName>
              <SurName>Black</SurName>
            </Name>
            <Location/>
          </Prescriber>
          <DispensingEvent>
            <DispenseDate>2012-01-01T00:00:00</DispenseDate>
            <WrittenDate>2012-01-01T00:00:00</WrittenDate>
            <DrugName>DrugName</DrugName>
            <Strength>20MG</Strength>
            <DosageForm>TAB</DosageForm>
            <Quantity>0</Quantity>
            <DaysSupply>0</DaysSupply>
            <RefillStatus/>
            <PartialFillIndicator>0</PartialFillIndicator>
            <PaymentType>CASH</PaymentType>
            <DateSold>0001-01-01T00:00:00</DateSold>
            <RxSerialNumber>
              <IssuingState>PA</IssuingState>
              <SerialNumber>987654321</SerialNumber>
            </RxSerialNumber>
            <PickupOrDropOffPerson>
              <Name>
                <GivenName>Jane</GivenName>
                <SurName>Doe</SurName>
              </Name>
              <IDQualifier>06</IDQualifier>
              <ID>123-456-789</ID>
              <RelationshipCode>03</RelationshipCode>
            </PickupOrDropOffPerson>
          </DispensingEvent>
        </DispensingEventInfo>
      </Prescriptions>
    </PharmacyDispenseInfo>
  </PrescriptionDetails>
  <Messages>
    <string>Free form message</string>
  </Messages>
</PMPResponse>
</Details>
<Messages>
  <string>Alert free form message 1</string>
  <string>Alert free form message 2</string>

```

```

    <string>Alert free form message 3</string>
  </Messages>
</PMPAlertAutomatedPollResult>
</PMPAlertAutomatedPollResponse>
</soap:Body>
</soap:Envelope>

```

2.4.3 Reference Number Alert Response to an Automated Poll

This example shows a reference number response to a poll request. Note that the response is a polymorphic xml response. The “type” attribute of the “PMPResponse” object informs the caller of the type of response. Further note that the <Details> object holds a collection of <PMPResponse> objects. Multiple <PMPResponse> objects (i.e. multiple alerts) can be returned in response to a single poll. Further note that the <Messages> object is a collection of messages. Multiple messages can be returned at either the detail level, or overall.

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <PMPAlertAutomatedPollResponse xmlns="http://www.asapnet.org/pmprequest">
      <PMPAlertAutomatedPollResult>
        <ResponseDate>2012-12-12T00:00:00-05:00</ResponseDate>
        <Details>
          <PMPResponse xsi:type="PMPRefNoResponse">
            <RefNo>123456</RefNo>
          </PMPResponse>
          <PMPResponse xsi:type="PMPRefNoResponse">
            <RefNo>987654</RefNo>
          </PMPResponse>
        </Details>
        <Messages>
          <string>Alert free form message 1</string>
          <string>Alert free form message 2</string>
          <string>Alert free form message 3</string>
        </Messages>
      </PMPAlertAutomatedPollResult>
    </PMPAlertAutomatedPollResponse>
  </soap:Body>
</soap:Envelope>

```

2.4.4 Response to an Automated Poll When No Alerts Exist

In the situation when a PMP has no alerts to return, a result with an empty list should be returned. This response would look like the following:

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <PMPAlertAutomatedPollResponse xmlns="http://www.asapnet.org/pmprequest">
      <PMPAlertAutomatedPollResult>
        <ResponseDate>2012-12-12T00:00:00-05:00</ResponseDate>
        <Details />
        <Messages>
          <string>Optional message or messages detailing that no alerts exist</string>
        </Messages>
      </PMPAlertAutomatedPollResult>
    </PMPAlertAutomatedPollResponse>
  </soap:Body>
</soap:Envelope>

```

Note in the above example that the <Details> element contains no objects indicating that there are no current alerts.

2.5 Exceptions

Appropriate soap exceptions should be used to communicate an inability by the PMP to process the request due to systems being down, invalid arguments, internal errors, or other exception conditions. The following is an example of a soap exception.

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:Client</faultcode>
      <faultstring>Appropriate fault string</faultstring>
      <faultactor>http://localhost:49332/PMPRequestService.asmx</faultactor>
      <detail>
        <ErrorMessage MessageText="Descriptive message containing details of error condition."
xmlns="http://www.asapnet.org/pmprequest"/>
      </detail>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>
```

Note that the <details> node in the above example can certainly be modified, extended, or otherwise customized as is appropriate for the host system. This is simply an example of a typical soap exception.

2.6 Encryption

It is assumed that this Web service will be deployed on port 443 and will use secure http as a transport protocol, with encryption being provided by SSL.

2.7 WSDL

The following is the Web service definition language (WSDL) listing that defines the Web service.

```
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions xmlns:tm="http://microsoft.com/wsdl/mime/textMatching/" xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/" xmlns:tns="http://www.asapnet.org/pmprequest"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:s="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/" xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
targetNamespace="http://www.asapnet.org/pmprequest" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:types>
    <s:schema elementFormDefault="qualified" targetNamespace="http://www.asapnet.org/pmprequest">
      <s:element name="AdHocPMPRequest">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="userId" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="passwordDigest" type="s:base64Binary" />
            <s:element minOccurs="0" maxOccurs="1" name="nonce" type="s:string" />
            <s:element minOccurs="1" maxOccurs="1" name="ts" type="s:dateTime" />
            <s:element minOccurs="0" maxOccurs="1" name="req" type="tns:PMPQuery" />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:complexType name="PMPQuery">
        <s:sequence>
          <s:element minOccurs="1" maxOccurs="1" name="QueryDate" type="s:dateTime" />
          <s:element minOccurs="0" maxOccurs="1" name="RequestDateRange" type="tns:DateRange" />
        </s:sequence>
      </s:complexType>
      <s:complexType name="DateRange">
        <s:sequence>
          <s:element minOccurs="1" maxOccurs="1" name="DateRangeBegin" type="s:dateTime" />
          <s:element minOccurs="1" maxOccurs="1" name="DateRangeEnd" type="s:dateTime" />
        </s:sequence>
      </s:complexType>
    </s:schema>
  </wsdl:types>
```

```

<s:complexType name="PMPDetailedQuery">
  <s:complexContent mixed="false">
    <s:extension base="tns:PMPQuery">
      <s:sequence>
        <s:element minOccurs="0" maxOccurs="1" name="Patient" type="tns:Person" />
      </s:sequence>
    </s:extension>
  </s:complexContent>
</s:complexType>
<s:complexType name="Person">
  <s:sequence>
    <s:element minOccurs="1" maxOccurs="1" name="BirthDate" type="s:dateTime" />
    <s:element minOccurs="0" maxOccurs="1" name="Name" type="tns:PersonName" />
    <s:element minOccurs="0" maxOccurs="1" name="ContactInformation" type="tns:LocationInfo" />
    <s:element minOccurs="0" maxOccurs="1" name="DriversLicensID" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="PassportID" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="SSN" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="UniqueSystemID" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="MilitaryID" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="StateIssuedID" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="PermanentResidentCardID" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="TribalID" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="Gender" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="SpeciesCode" type="s:string" />
  </s:sequence>
</s:complexType>
<s:complexType name="PersonName">
  <s:sequence>
    <s:element minOccurs="0" maxOccurs="1" name="GivenName" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="MiddleName" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="SurName" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="NameSuffix" type="s:string" />
  </s:sequence>
</s:complexType>
<s:complexType name="LocationInfo">
  <s:sequence>
    <s:element minOccurs="0" maxOccurs="1" name="Phone" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="StreetAddress" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="StreetAddress2" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="StreetAddress3" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="City" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="LocationStateUsPostalServiceCode" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="LocationPostalCode" type="s:string" />
  </s:sequence>
</s:complexType>
<s:complexType name="PMPRefNoQuery">
  <s:complexContent mixed="false">
    <s:extension base="tns:PMPQuery">
      <s:sequence>
        <s:element minOccurs="0" maxOccurs="1" name="RefNo" type="s:string" />
      </s:sequence>
    </s:extension>
  </s:complexContent>
</s:complexType>
<s:element name="AdHocPMPRequestResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="AdHocPMPRequestResult" type="tns:adhocPMPResponse" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:complexType name="adhocPMPResponse">
  <s:sequence>
    <s:element minOccurs="0" maxOccurs="1" name="Messages" type="tns:ArrayOfString" />
    <s:element minOccurs="1" maxOccurs="1" name="ResponseDate" type="s:dateTime" />
    <s:element minOccurs="0" maxOccurs="1" name="Details" type="tns:ArrayOfPMPDetailedResponse" />
  </s:sequence>
</s:complexType>
<s:complexType name="ArrayOfString">
  <s:sequence>

```

```

    <s:element minOccurs="0" maxOccurs="unbounded" name="string" nillable="true" type="s:string" />
  </s:sequence>
</s:complexType>
<s:complexType name="ArrayOfPMPDDetailedResponse">
  <s:sequence>
    <s:element minOccurs="0" maxOccurs="unbounded" name="PMPDDetailedResponse" nillable="true" type="tns:PMPDDetailedResponse" />
  </s:sequence>
</s:complexType>
<s:complexType name="PMPDDetailedResponse">
  <s:complexContent mixed="false">
    <s:extension base="tns:PMPResponse">
      <s:sequence>
        <s:element minOccurs="0" maxOccurs="1" name="Patient" type="tns:Person" />
        <s:element minOccurs="0" maxOccurs="1" name="PrescriptionDetails" type="tns:ArrayOfPharmacyDispenseInfo" />
        <s:element minOccurs="0" maxOccurs="1" name="Messages" type="tns:ArrayOfString" />
        <s:element minOccurs="0" maxOccurs="1" name="PickListDetails" type="tns:PickListInfoDetails" />
        <s:element minOccurs="0" maxOccurs="1" name="Summary" type="tns:ResponseSummaryInfo" />
      </s:sequence>
    </s:extension>
  </s:complexContent>
</s:complexType>
<s:complexType name="PMPResponse" abstract="true" />
<s:complexType name="PMPRefNoResponse">
  <s:complexContent mixed="false">
    <s:extension base="tns:PMPResponse">
      <s:sequence>
        <s:element minOccurs="0" maxOccurs="1" name="RefNo" type="s:string" />
      </s:sequence>
    </s:extension>
  </s:complexContent>
</s:complexType>
<s:complexType name="ArrayOfPharmacyDispenseInfo">
  <s:sequence>
    <s:element minOccurs="0" maxOccurs="unbounded" name="PharmacyDispenseInfo" nillable="true" type="tns:PharmacyDispenseInfo" />
  </s:sequence>
</s:complexType>
<s:complexType name="PharmacyDispenseInfo">
  <s:sequence>
    <s:element minOccurs="0" maxOccurs="1" name="Pharmacy" type="tns:PharmacyInfo" />
    <s:element minOccurs="0" maxOccurs="1" name="Prescriptions" type="tns:ArrayOfDispensingEventInfo" />
  </s:sequence>
</s:complexType>
<s:complexType name="PharmacyInfo">
  <s:sequence>
    <s:element minOccurs="0" maxOccurs="1" name="NationalProviderID" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="NCPDPPProviderID" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="DEANumber" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="PharmacyName" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="Location" type="tns:LocationInfo" />
  </s:sequence>
</s:complexType>
<s:complexType name="ArrayOfDispensingEventInfo">
  <s:sequence>
    <s:element minOccurs="0" maxOccurs="unbounded" name="DispensingEventInfo" nillable="true" type="tns:DispensingEventInfo" />
  </s:sequence>
</s:complexType>
<s:complexType name="DispensingEventInfo">
  <s:sequence>
    <s:element minOccurs="0" maxOccurs="1" name="Prescriber" type="tns:Prescriber" />
    <s:element minOccurs="0" maxOccurs="1" name="DispensingEvent" type="tns:DispensingInfo" />
  </s:sequence>
</s:complexType>
<s:complexType name="Prescriber">
  <s:sequence>
    <s:element minOccurs="0" maxOccurs="1" name="Name" type="tns:PersonName" />
    <s:element minOccurs="0" maxOccurs="1" name="DEANumber" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="DEANumberSuffix" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="NPI" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="StateLicenseNumber" type="s:string" />
  </s:sequence>
</s:complexType>

```

```

    <s:element minOccurs="0" maxOccurs="1" name="Location" type="tns:LocationInfo" />
  </s:sequence>
</s:complexType>
<s:complexType name="DispensingInfo">
  <s:sequence>
    <s:element minOccurs="1" maxOccurs="1" name="DispenseDate" type="s:dateTime" />
    <s:element minOccurs="1" maxOccurs="1" name="WrittenDate" type="s:dateTime" />
    <s:element minOccurs="0" maxOccurs="1" name="PrescriptionNumber" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="DrugName" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="Strength" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="DosageForm" type="s:string" />
    <s:element minOccurs="1" maxOccurs="1" name="Quantity" type="s:decimal" />
    <s:element minOccurs="1" maxOccurs="1" name="DaysSupply" type="s:decimal" />
    <s:element minOccurs="0" maxOccurs="1" name="RefillStatus" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="PartialFillIndicator" type="s:string" />
    <s:element minOccurs="1" maxOccurs="1" name="PaymentType" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="RxNormCode" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="ElectronicPrescriptionReferenceNumber" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="ElectronicPrescriptionOrderNumber" type="s:string" />
    <s:element minOccurs="1" maxOccurs="1" name="DateSold" type="s:dateTime" />
    <s:element minOccurs="0" maxOccurs="1" name="RxSerialNumber" type="tns:PrescriptionSerialNoInfo" />
    <s:element minOccurs="0" maxOccurs="1" name="PickupOrDropOffPerson" type="tns:PickupPersonInfo" />
  </s:sequence>
</s:complexType>
<s:complexType name="PrescriptionSerialNoInfo">
  <s:sequence>
    <s:element minOccurs="0" maxOccurs="1" name="IssuingState" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="SerialNumber" type="s:string" />
  </s:sequence>
</s:complexType>
<s:complexType name="PickupPersonInfo">
  <s:sequence>
    <s:element minOccurs="0" maxOccurs="1" name="Name" type="tns:PersonName" />
    <s:element minOccurs="0" maxOccurs="1" name="IDQualifier" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="ID" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="RelationshipCode" type="s:string" />
  </s:sequence>
</s:complexType>
<s:complexType name="PickListInfoDetails">
  <s:sequence>
    <s:element minOccurs="1" maxOccurs="1" name="WeightingFactor" type="s:int" />
    <s:element minOccurs="0" maxOccurs="1" name="RefNo" type="s:string" />
  </s:sequence>
</s:complexType>
<s:complexType name="ResponseSummaryInfo">
  <s:sequence>
    <s:element minOccurs="1" maxOccurs="1" name="NumberOfPharmacies" type="s:int" />
    <s:element minOccurs="1" maxOccurs="1" name="NumberOfPrescribers" type="s:int" />
    <s:element minOccurs="1" maxOccurs="1" name="NumberOfPrescriptions" type="s:int" />
  </s:sequence>
</s:complexType>
<s:element name="PMPAlertAutomatedPoll">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="userId" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="passwordDigest" type="s:base64Binary" />
      <s:element minOccurs="0" maxOccurs="1" name="pharmacyId" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="nonce" type="s:string" />
      <s:element minOccurs="1" maxOccurs="1" name="ts" type="s:dateTime" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="PMPAlertAutomatedPollResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="PMPAlertAutomatedPollResult" type="tns:PMPAlert" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:complexType name="PMPAlert">

```



```

    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="ResponseDate" type="s:dateTime" />
      <s:element minOccurs="0" maxOccurs="1" name="Details" type="tns:ArrayOfPMPResponse" />
      <s:element minOccurs="0" maxOccurs="1" name="Messages" type="tns:ArrayOfString" />
    </s:sequence>
  </s:complexType>
  <s:complexType name="ArrayOfPMPResponse">
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="unbounded" name="PMPResponse" nillable="true" type="tns:PMPResponse" />
    </s:sequence>
  </s:complexType>
</s:schema>
</wsdl:types>
<wsdl:message name="AdHocPMPRequestSoapIn">
  <wsdl:part name="parameters" element="tns:AdHocPMPRequest" />
</wsdl:message>
<wsdl:message name="AdHocPMPRequestSoapOut">
  <wsdl:part name="parameters" element="tns:AdHocPMPRequestResponse" />
</wsdl:message>
<wsdl:message name="PMPAlertAutomatedPollSoapIn">
  <wsdl:part name="parameters" element="tns:PMPAlertAutomatedPoll" />
</wsdl:message>
<wsdl:message name="PMPAlertAutomatedPollSoapOut">
  <wsdl:part name="parameters" element="tns:PMPAlertAutomatedPollResponse" />
</wsdl:message>
<wsdl:portType name="PMPRequestServiceSoap">
  <wsdl:operation name="AdHocPMPRequest">
    <wsdl:input message="tns:AdHocPMPRequestSoapIn" />
    <wsdl:output message="tns:AdHocPMPRequestSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="PMPAlertAutomatedPoll">
    <wsdl:input message="tns:PMPAlertAutomatedPollSoapIn" />
    <wsdl:output message="tns:PMPAlertAutomatedPollSoapOut" />
  </wsdl:operation>
</wsdl:portType>
<wsdl:binding name="PMPRequestServiceSoap" type="tns:PMPRequestServiceSoap">
  <soap:binding transport="http://schemas.xmlsoap.org/soap/http" />
  <wsdl:operation name="AdHocPMPRequest">
    <soap:operation soapAction="http://www.asapnet.org/pmprequest/AdHocPMPRequest" style="document" />
    <wsdl:input>
      <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="PMPAlertAutomatedPoll">
    <soap:operation soapAction="http://www.asapnet.org/pmprequest/PMPAlertAutomatedPoll" style="document" />
    <wsdl:input>
      <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
</wsdl:binding>
<wsdl:binding name="PMPRequestServiceSoap12" type="tns:PMPRequestServiceSoap">
  <soap12:binding transport="http://schemas.xmlsoap.org/soap/http" />
  <wsdl:operation name="AdHocPMPRequest">
    <soap12:operation soapAction="http://www.asapnet.org/pmprequest/AdHocPMPRequest" style="document" />
    <wsdl:input>
      <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
      <soap12:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="PMPAlertAutomatedPoll">
    <soap12:operation soapAction="http://www.asapnet.org/pmprequest/PMPAlertAutomatedPoll" style="document" />
    <wsdl:input>

```

```
<soap12:body use="literal" />
</wsdl:input>
<wsdl:output>
  <soap12:body use="literal" />
</wsdl:output>
</wsdl:operation>
</wsdl:binding>
<wsdl:service name="PMRequestService">
  <wsdl:port name="PMRequestServiceSoap" binding="tns:PMRequestServiceSoap">
    <soap:address location="http://localhost:49332/PMRequestService.asmx" />
  </wsdl:port>
  <wsdl:port name="PMRequestServiceSoap12" binding="tns:PMRequestServiceSoap12">
    <soap12:address location="http://localhost:49332/PMRequestService.asmx" />
  </wsdl:port>
</wsdl:service>
</wsdl:definitions>
```

Appendix

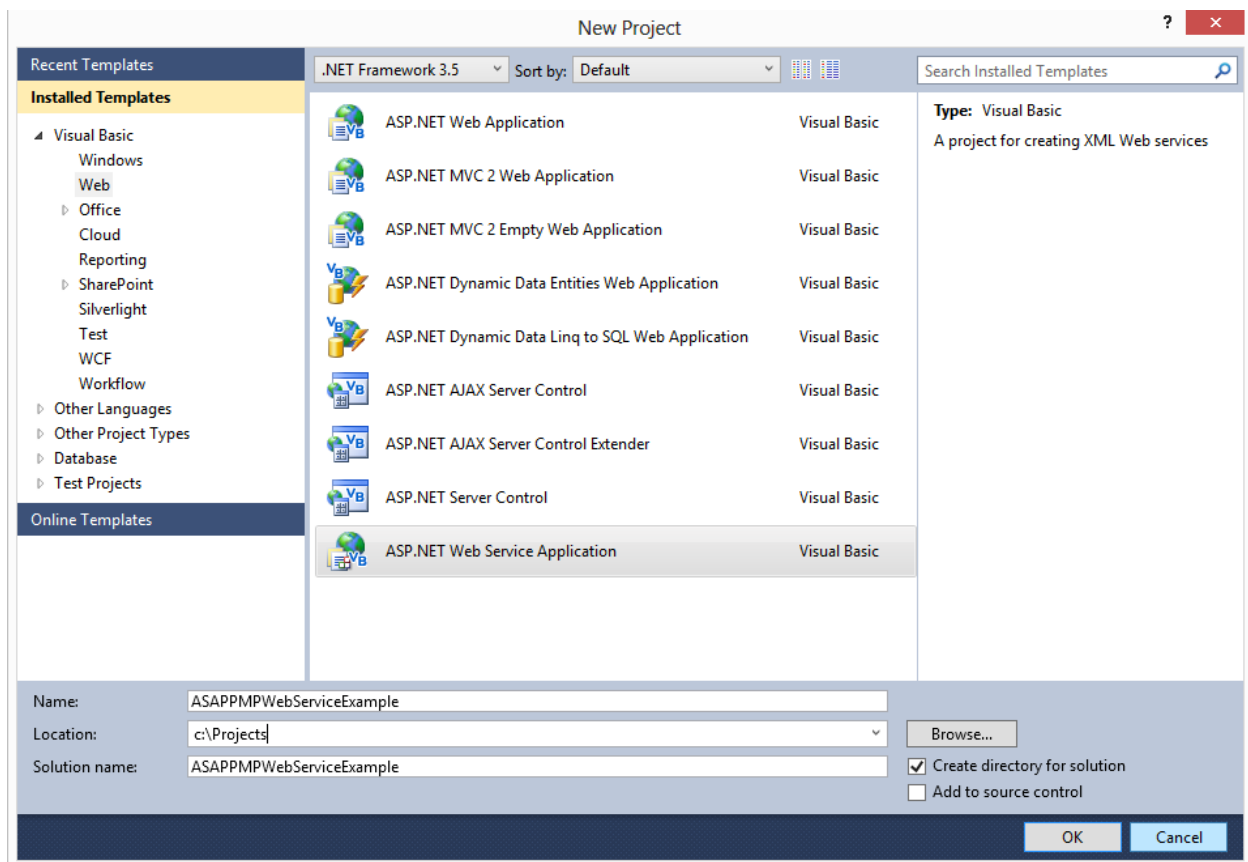
ASAP PMP Web Service Example Program to Implement the Standard

3 Introduction

The following steps detail how to build a sample application that implements the ASAP PMP Web Service, as well as a small sample application to call that service. To follow along with these samples you will need Microsoft Visual Studio 2010.

4 Building the Web Service

Create a new project using Visual Studio. Change the .NET Framework to 3.5 and select ASP .NET Web Service Application under the Visual Basic — Web templates, as shown in the following screenshot.



Change the project name, solution name, and location as desired.

Create the following files and add the source code shown.

File Name: adhocPMPResponse.vb

```
Imports System.Collections.Generic

Public Class adhocPMPResponse
    Public Property Messages As List(Of String)
    Public Property ResponseDate As Date
    Public Property Details As List(Of PMPDetailedResponse)
End Class
```

File Name: DateRange.vb

```
Public Class DateRange
    Public Property DateRangeBegin As Date
    Public Property DateRangeEnd As Date
End Class
```

File Name: DispensingEventInfo.vb

```
Public Class DispensingEventInfo
    Public Property Prescriber As Prescriber
    Public Property DispensingEvent As DispensingInfo

    Public Sub New()
        Prescriber = New Prescriber
        DispensingEvent = New DispensingInfo
    End Sub
End Class
```

File Name: DispensingInfo.vb

```
Public Class DispensingInfo
    Public Property DispenseDate As Date
    Public Property WrittenDate As Date
    Public Property PrescriptionNumber As String
    Public Property DrugName As String
    Public Property Strength As String
    Public Property DosageForm As String
    Public Property Quantity As Decimal
    Public Property DaysSupply As Decimal
    Public Property RefillStatus As String
    Public Property PartialFillIndicator As String
    Public Property PaymentType As String
    Public Property RxNormCode As String
    Public Property ElectronicPrescriptionReferenceNumber As String
    Public Property ElectronicPrescriptionOrderNumber As String
    Public Property DateSold As Date
    Public Property RxSerialNumber As PrescriptionSerialNoInfo
    Public Property PickupOrDropOffPerson As PickupPersonInfo

    Public Sub New()
        RxSerialNumber = New PrescriptionSerialNoInfo
        PickupOrDropOffPerson = New PickupPersonInfo
    End Sub
End Class
```

File Name: LocationInfo.vb

```
Public Class LocationInfo
    Public Property Phone As String
    Public Property StreetAddress As String
    Public Property StreetAddress2 As String
    Public Property StreetAddress3 As String
    Public Property City As String
    Public Property LocationStateUsPostalServiceCode As String
    Public Property LocationPostalCode As String
End Class
```

File Name: Person.vb

```
Public Class Person
    Public Property BirthDate As Date
    Public Property Name As PersonName
    Public Property ContactInformation As LocationInfo
    Public Property DriversLicensID As String
    Public Property PassportID As String
    Public Property SSN As String
    Public Property UniqueSystemID As String
    Public Property MilitaryID As String
    Public Property StateIssuedID As String
    Public Property PermanentResidentCardID As String
    Public Property TribalID As String
    Public Property Gender As String
    Public Property SpeciesCode As String

    Public Sub New()
        Name = New PersonName
        ContactInformation = New LocationInfo
    End Sub
End Class
```

File Name: PersonName.vb

```
Public Class PersonName
    Public Property GivenName As String
    Public Property MiddleName As String
    Public Property SurName As String
    Public Property NameSuffix As String
End Class
```

File Name: Pharmacy.vb

```
Public Class PharmacyInfo
    Public Property NationalProviderID As String
    Public Property NCPDPPProviderID As String
    Public Property DEANumber As String
    Public Property PharmacyName As String
    Public Property Location As LocationInfo

    Public Sub New()
        Location = New LocationInfo
    End Sub
End Class
```

File Name: PharmacyDispenseInfo.vb

```
Public Class PharmacyDispenseInfo
    Public Property Pharmacy As PharmacyInfo
    Public Property Prescriptions As List(Of DispensingEventInfo)

    Public Sub New()
        Pharmacy = New PharmacyInfo
        Prescriptions = New List(Of DispensingEventInfo)
    End Sub
End Class
```

File Name: PickListInfoDetails.vb

```
Public Class PickListInfoDetails
    Public Property WeightingFactor As Integer
    Public Property RefNo As String
End Class
```

File Name: PickupPersonInfo.vb

```
Public Class PickupPersonInfo
    Public Property Name As PersonName
    Public Property IDQualifier As String
End Class
```

```

    Public Property ID As String
    Public Property RelationshipCode As String

    Public Sub New()
        Name = New PersonName
    End Sub
End Class

```

File Name: PMPAlert.vb

```

Public Class PMPAlert
    Public Property ResponseDate As Date
    Public Property Details As List(Of PMPResponse)
    Public Property Messages As List(Of String)
End Class

```

File Name: PMPDetailedQuery.vb

```

Public Class PMPDetailedQuery : Inherits PMPQuery
    Public Property Patient As Person
End Class

```

File Name: PMPDetailedResponse.vb

```

Public Class PMPDetailedResponse
    Inherits PMPResponse
    Public Property Patient As Person
    Public Property PrescriptionDetails As List(Of PharmacyDispenseInfo)
    Public Property Messages As List(Of String)
    Public Property PickListDetails As PickListInfoDetails
    Public Property Summary As ResponseSummaryInfo
End Class

```

File Name: PMPQuery.vb

```

Public Class PMPQuery
    Public Property QueryDate As Date
    Public Property RequestDateRange As DateRange
End Class

```

File Name: PMPRefNoQuery.vb

```

Public Class PMPRefNoQuery : Inherits PMPQuery
    Public property RefNo As String
End Class

```

File Name: PMPRefNoResponse.vb

```

Public Class PMPRefNoResponse
    Inherits PMPResponse
    Public Property RefNo As String
End Class

```

File Name: PMPRequestService.asmx

```

Imports System.Web.Services
Imports System.Web.Services.Protocols
Imports System.ComponentModel
Imports System.Xml

```

```

Imports System
Imports System.Web.Services.Description
Imports System.Xml.Schema
Imports System.Xml.Serialization

```

```

' To allow this Web Service to be called from script, using ASP.NET AJAX, uncomment the following line.
' <System.Web.Script.Services.ScriptService()> _
<System.Web.Services.WebService(Namespace="http://www.asapnet.org/pmprequest")> _

```

```

<System.Web.Services.WebServiceBinding(ConformsTo:=WsiProfiles.BasicProfile1_1)> _
<ToolboxItem(False)> _
Public Class PMPRequestService
    Inherits System.Web.Services.WebService

    <WebMethod()>
    <XmlInclude(GetType(PMPDetailedQuery))>
    <XmlInclude(GetType(PMPRefNoQuery))>
    Public Function AdHocPMPRequest(ByVal userId As String, ByVal passwordDigest As Byte(), ByVal nonce As
String, ByVal ts As DateTime, ByVal req As PMPQuery) As adhocPMPResponse
        Dim rsp As adhocPMPResponse
        Dim PickListResponse As Boolean, EmptyResponse As Boolean, ThrowException As Boolean

        rsp = New adhocPMPResponse
        rsp.Details = New List(Of PMPDetailedResponse)

'*****
' The following booleans control the behavior of the sample program. Only one should be true
' at a time.
' If none of the following are true then a detailed response will be returned.
PickListResponse = True
EmptyResponse = False
ThrowException = False
'*****

    If TypeOf (req) Is PMPDetailedQuery Then
        Dim dq As PMPDetailedQuery
        dq = req
        Debug.Print(dq.Patient.Name.GivenName)
    End If

    If ThrowException Then
        Dim doc As New System.Xml.XmlDocument
        Dim node As System.Xml.XmlNode = doc.CreateNode(XmlNodeType.Element,
            SoapException.DetailElementName.Name, SoapException.DetailElementName.Namespace)
        Dim details As System.Xml.XmlNode = doc.CreateNode(XmlNodeType.Element, "ErrorMessage",
            "http://www.asapnet.org/pmprequest")
        Dim attr As XmlAttribute = doc.CreateAttribute("MessageText")
        attr.Value = "Descriptive message containing details of error condition."
        details.Attributes.Append(attr)
        node.AppendChild(details)
        Dim se As New SoapException("Fault occurred", SoapException.ClientFaultCode,
            Context.Request.Url.AbsoluteUri, node)
        Throw se
    ElseIf EmptyResponse Then
        'do nothing
    ElseIf PickListResponse Then
        rsp.ResponseDate = Today
        Dim dr As New PMPDetailedResponse

        dr.Patient = New Person
        dr.Patient.BirthDate = "05/10/1950"
        dr.Patient.Name.GivenName = "John"
        dr.Patient.Name.SurName = "Doe"
        dr.Patient.ContactInformation.StreetAddress = "123 Any Street"
        dr.Patient.ContactInformation.City = "Some City"
        dr.Patient.ContactInformation.LocationStateUsPostalServiceCode = "PA"
        dr.Patient.ContactInformation.LocationPostalCode = "12345"
        dr.PickListDetails = New PickListInfoDetails
        dr.PickListDetails.WeightingFactor = 98
        dr.PickListDetails.RefNo = 1234567
        rsp.Details.Add(dr)

        dr.Patient = New Person

```

```

dr.Patient.BirthDate = "05/10/1950"
dr.Patient.Name.GivenName = "John"
dr.Patient.Name.SurName = "Doe"
dr.Patient.ContactInformation.StreetAddress = "123 South Any Street"
dr.Patient.ContactInformation.StreetAddress2 = "Suite 1"
Debug.Print(dr.Patient.ContactInformation.StreetAddress)
dr.Patient.ContactInformation.City = "Somewhere"
dr.Patient.ContactInformation.LocationStateUsPostalServiceCode = "PA"
dr.Patient.ContactInformation.LocationPostalCode = "02145"
dr.PickListDetails = New PickListInfoDetails
dr.PickListDetails.WeightinFactor = 90
dr.PickListDetails.RefNo = 2134569
rsp.Details.Add(dr)

dr.Patient = New Person
dr.Patient.BirthDate = "05/10/1950"
dr.Patient.Name.GivenName = "John"
dr.Patient.Name.SurName = "Doe"
dr.Patient.ContactInformation.StreetAddress = "30 Smith Rd."
dr.Patient.ContactInformation.City = "Burlington"
dr.Patient.ContactInformation.LocationStateUsPostalServiceCode = "PA"
dr.Patient.ContactInformation.LocationPostalCode = "02222"
dr.PickListDetails = New PickListInfoDetails
dr.PickListDetails.WeightinFactor = 20
dr.PickListDetails.RefNo = 1133319
rsp.Details.Add(dr)

dr.Patient = New Person
dr.Patient.BirthDate = "05/10/1950"
dr.Patient.Name.GivenName = "John"
dr.Patient.Name.SurName = "Doe"
dr.Patient.ContactInformation.StreetAddress = "15A Barney Lane"
dr.Patient.ContactInformation.City = "Smithfield"
dr.Patient.ContactInformation.LocationStateUsPostalServiceCode = "PA"
dr.Patient.ContactInformation.LocationPostalCode = "02103"
dr.PickListDetails = New PickListInfoDetails
dr.PickListDetails.WeightinFactor = 15
dr.PickListDetails.RefNo = 1133666
rsp.Details.Add(dr)

dr.Patient = New Person
dr.Patient.BirthDate = "05/01/1950"
dr.Patient.Name.GivenName = "John"
dr.Patient.Name.SurName = "Doe"
dr.Patient.ContactInformation.StreetAddress = "123 South Any Street"
dr.Patient.ContactInformation.City = "Some City"
dr.Patient.ContactInformation.LocationStateUsPostalServiceCode = "PA"
dr.Patient.ContactInformation.LocationPostalCode = "12345"
dr.PickListDetails = New PickListInfoDetails
dr.PickListDetails.WeightinFactor = 75
dr.PickListDetails.RefNo = 124441
rsp.Details.Add(dr)
Else
rsp.ResponseDate = Today()
Dim dr As New PMPDetailedResponse

dr.Patient = New Person
dr.Patient.BirthDate = "10/7/1950"
dr.Patient.Name.GivenName = "John"
dr.Patient.Name.SurName = "Doe"
dr.Patient.ContactInformation.Phone = "412-555-5555"
dr.Patient.ContactInformation.StreetAddress = "123 Any Street"
dr.Patient.ContactInformation.City = "SomeCity"
dr.Patient.ContactInformation.LocationStateUsPostalServiceCode = "PA"
dr.Patient.ContactInformation.LocationPostalCode = "12345"
dr.Patient.Gender = "M"
dr.Patient.DriversLicensID = "12-345-678"
dr.Patient.PassportID = "12345678901234567890"
dr.Patient.SSN = "123-45-6789"

```



```

Dim pdi As New PharmacyDispenseInfo
pdi.Pharmacy.DEANumber = "AB12345678"
pdi.Pharmacy.Location.Phone = "123-456-7890"
pdi.Pharmacy.Location.StreetAddress = "312 Another Street"
pdi.Pharmacy.Location.City = "SomeCity"
pdi.Pharmacy.Location.LocationStateUsPostalServiceCode = "PA"
pdi.Pharmacy.Location.LocationPostalCode = "12345"

Dim dei As New DispensingEventInfo
dei.Prescriber.Name.GivenName = "Jack"
dei.Prescriber.Name.SurName = "Black"
dei.Prescriber.DEANumber = "CD234567890"
dei.Prescriber.NPI = "1234567890"
dei.Prescriber.Location.StreetAddress = "999 Practice Street"
dei.Prescriber.Location.StreetAddress2 = "Suite 100"
dei.Prescriber.Location.LocationStateUsPostalServiceCode = "PA"
dei.Prescriber.Location.LocationPostalCode = "12876"
dei.Prescriber.Location.Phone = "(800)555-1212"

dei.DispensingEvent.DispenseDate = "1/1/2012"
dei.DispensingEvent.WrittenDate = "1/1/2012"
dei.DispensingEvent.DrugName = "DrugName"
dei.DispensingEvent.Strength = "20MG"
dei.DispensingEvent.DosageForm = "TAB"
dei.DispensingEvent.PartialFillIndicator = "0"
dei.DispensingEvent.PaymentType = "CASH"
dei.DispensingEvent.RefillStatus = ""
With dei.DispensingEvent.RxSerialNumber
    .IssuingState = "PA"
    .SerialNumber = "987654321"
End With
With dei.DispensingEvent.PickupOrDropOffPerson
    .Name.GivenName = "Jane"
    .Name.SurName = "Doe"
    .IDQualifier = "06"
    .ID = "123-456-789"
    .RelationshipCode = "03"
End With
dei.DispensingEvent.PickupOrDropOffPerson.Name.SurName = "Doe"
pdi.Prescriptions.Add(dei)
dr.PrescriptionDetails = New List(Of PharmacyDispenseInfo)
dr.PrescriptionDetails.Add(pdi)

dr.Summary = New ResponseSummaryInfo
dr.Summary.NumberOfPharmacies = 1
dr.Summary.NumberOfPrescribers = 1
dr.Summary.NumberOfPrescriptions = 1

dr.Messages = New List(Of String)
dr.Messages.Add("Free form message")

rsp.Details.Add(dr)
End If
Return rsp
End Function

<WebMethod(>
<XmlInclude(GetType(PMPDetailedResponse))>
<XmlInclude(GetType(PMPRefNoResponse))>
Public Function PMPAlertAutomatedPoll(ByVal userId As String, ByVal passwordDigest As Byte(), ByVal
    pharmacyId As String, ByVal nonce As String, ByVal ts As DateTime) As PMPAlert
Dim rsp As New PMPAlert

rsp.ResponseDate = Today

```

```

'*****
' The following booleans control the behavior of the sample program. If true a detailed response

```

```

' will be returned. If false, a reference number response will be returned.
Dim detailedReponseReturn As Boolean = False
'*****

If detailedReponseReturn Then
    Dim dr As New PMPDetailedResponse
    dr.Patient = New Person
    dr.Patient.BirthDate = "10/7/1950"
    dr.Patient.Name.GivenName = "John"
    dr.Patient.Name.SurName = "Doe"
    dr.Patient.ContactInformation.StreetAddress = "123 Any Street"
    dr.Patient.ContactInformation.City = "SomeCity"
    dr.Patient.ContactInformation.LocationStateUsPostalServiceCode = "PA"
    dr.Patient.ContactInformation.LocationPostalCode = "12345"
    dr.Patient.Gender = "M"
    dr.Patient.DriversLicensID = "12-345-678"
    dr.Patient.PassportID = "12345678901234567890"
    dr.Patient.SSN = "123-45-6789"

    Dim pdi As New PharmacyDispenseInfo
    pdi.Pharmacy.DEANumber = "AB12345678"
    pdi.Pharmacy.Location.Phone = "123-456-7890"
    pdi.Pharmacy.Location.StreetAddress = "312 Another Street"
    pdi.Pharmacy.Location.City = "SomeCity"
    pdi.Pharmacy.Location.LocationStateUsPostalServiceCode = "PA"
    pdi.Pharmacy.Location.LocationPostalCode = "12345"

    Dim dei As New DispensingEventInfo
    dei.Prescriber.Name.GivenName = "Jack"
    dei.Prescriber.Name.SurName = "Black"

    dei.DispensingEvent.DispenseDate = "1/1/2012"
    dei.DispensingEvent.WrittenDate = "1/1/2012"
    dei.DispensingEvent.DrugName = "DrugName"
    dei.DispensingEvent.Strength = "20MG"
    dei.DispensingEvent.DosageForm = "TAB"
    dei.DispensingEvent.PartialFillIndicator = "0"
    dei.DispensingEvent.PaymentType = "CASH"
    dei.DispensingEvent.RefillStatus = ""
    With dei.DispensingEvent.RxSerialNumber
        .IssuingState = "PA"
        .SerialNumber = "987654321"
    End With
    With dei.DispensingEvent.PickupOrDropOffPerson
        .Name.GivenName = "Jane"
        .Name.SurName = "Doe"
        .IDQualifier = "06"
        .ID = "123-456-789"
        .RelationshipCode = "03"
    End With
    dei.DispensingEvent.PickupOrDropOffPerson.Name.SurName = "Doe"
    pdi.Prescriptions.Add(dei)
    dr.PrescriptionDetails = New List(Of PharmacyDispenseInfo)
    dr.PrescriptionDetails.Add(pdi)

    dr.Messages = New List(Of String)
    dr.Messages.Add("Free form message")

    rsp.Details = New List(Of PMPResponse)
    rsp.Details.Add(dr)
Else
    Dim rnr As New PMPRefNoResponse
    rnr.RefNo = "123456"

    rsp.Details = New List(Of PMPResponse)
    rsp.Details.Add(rnr)

    rnr = New PMPRefNoResponse
    rnr.RefNo = "987654"

```

```

        rsp.Details.Add(rnr)

    End If
    rsp.Messages = New List(Of String)
    rsp.Messages.Add("Alert free form message 1")
    rsp.Messages.Add("Alert free form message 2")
    rsp.Messages.Add("Alert free form message 3")

    Return rsp
End Function
End Class

```

File Name: PMPResponse.vb

```

Imports System.Web
Imports System.Web.Services
Imports System.Web.Services.Protocols
Imports System.ComponentModel

Public MustInherit Class PMPResponse

End Class

```

File Name: Prescriber.vb

```

Public Class Prescriber
    Public Property Name As PersonName
    Public Property DEANumber As String
    Public Property DEANumberSuffix As String
    Public Property NPI As String
    Public Property StateLicenseNumber As String
    Public Property Location As LocationInfo

    Public Sub New()
        Name = New PersonName
        Location = New LocationInfo
    End Sub
End Class

```

File Name: PrescriptionSerialNoInfo.vb

```

Public Class PrescriptionSerialNoInfo
    Public Property IssuingState As String
    Public Property SerialNumber As String
End Class

```

File Name: ResponseSummaryInfo.vb

```

Public Class ResponseSummaryInfo
    Public Property NumberOfPharmacies As Integer
    Public Property NumberOfPrescribers As Integer
    Public Property NumberOfPrescriptions As Integer
End Class

```

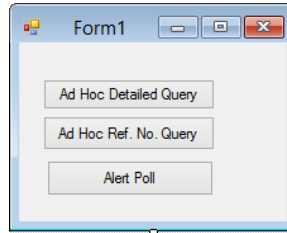
Compile and test the web service.

5 Building the test program

Add a new project to the solution. The type of the project should be a Windows Forms Application.

Add a reference to the Web service created in the previous project. Name this Web service "pmp."

Create a new form in the application that looks like this:



Place the following code in the form.

```
Public Class Form1
```

```

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button1.Click
    Dim p As New pmp.PMPRequestServiceSoapClient
    Dim dq As New pmp.PMPDetailedQuery
    dq.QueryDate = Today()
    dq.RequestDateRange = New pmp.DateRange
    dq.RequestDateRange.DateRangeBegin = CDate("1/1/2012")
    dq.RequestDateRange.DateRangeEnd = CDate("7/1/2012")
    dq.Patient = New pmp.Person
    dq.Patient.Name = New pmp.PersonName
    dq.Patient.Name.SurName = "Doe"
    dq.Patient.Name.GivenName = "John"
    dq.Patient.BirthDate = "10/5/1950"
    dq.Patient.Gender = "M"
    dq.Patient.ContactInformation = New pmp.LocationInfo
    dq.Patient.ContactInformation.StreetAddress = "123 Any Street"
    dq.Patient.ContactInformation.City = "SomeCity"
    dq.Patient.ContactInformation.LocationPostalCode = "12345"
    dq.Patient.ContactInformation.LocationStateUsPostalServiceCode = "PA"
    dq.Patient.ContactInformation.Phone = "412-555-5555"
    Dim b() As Byte
    Dim rsp As pmp.adhocPMPResponse
    b = System.Text.Encoding.UTF8.GetBytes("j2G3VF7YNkNnteuV2FVhEicM1hc=")
    Try
        rsp = p.AdHocPMPRequest("user@somepharmacy.com", b, "0F2ED1EA-2E78-48CC-9D22-C70A1FEB7615",
            Now(), dq)

        For Each dtl As pmp.PMPDetailedResponse In rsp.Details
            If dtl.Patient Is Nothing Then
                MessageBox.Show("No match")
            Else
                MessageBox.Show(dtl.Patient.Name.GivenName)
                'Do whatever you like with the response...
            End If
        Next

        Catch ex As Exception
            MessageBox.Show("Exception thrown by web service:" + ex.Message)
        End Try
    End Sub

Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button2.Click
    Dim p As New pmp.PMPRequestServiceSoapClient
    Dim rq As New pmp.PMPRefNoQuery
    rq.QueryDate = Today()
    rq.RefNo = "9876543"
    rq.RequestDateRange = New pmp.DateRange
    rq.RequestDateRange.DateRangeBegin = CDate("1/1/2012")
    rq.RequestDateRange.DateRangeEnd = CDate("7/1/2012")
    Dim b() As Byte
    b = System.Text.Encoding.UTF8.GetBytes("j2G3VF7YNkNnteuV2FVhEicM1hc=")
    Dim rsp As pmp.adhocPMPResponse
    rsp = p.AdHocPMPRequest("user@somepharmacy.com", b, "0F2ED1EA-2E78-48CC-9D22-C70A1FEB7615",
        Now(), rq)

```

```

For Each dtl As pmp.PMPDetailedResponse In rsp.Details
    If dtl.Patient Is Nothing Then
        MessageBox.Show("No match")
    Else
        MessageBox.Show(dtl.Patient.Name.GivenName)
        'Do whatever you like with the response...
    End If
Next
End Sub

Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button3.Click
    Dim p As New pmp.PMPRequestServiceSoapClient
    Dim b() As Byte = System.Text.Encoding.UTF8.GetBytes("j2G3VF7YNkNnteuV2FVhEicMlhc=")
    Dim rs As pmp.PMPAlert = p.PMPAlertAutomatedPoll("user@somepharmacy.com", b, "1234567890",
        "0F2ED1EA-2E78-48CC-9D22-C70A1FEB7615", Now())

    For Each r As pmp.PMPResponse In rs.Details
        If TypeOf r Is pmp.PMPDetailedResponse Then
            Dim dr As pmp.PMPDetailedResponse = r
            MessageBox.Show("detailed response for " & dr.Patient.Name.GivenName & " " &
                dr.Patient.Name.SurName)
        Else
            Dim rnr As pmp.PMPRefNoResponse = r
            MessageBox.Show("refno response " & rnr.RefNo)
            'Do whatever you like with the response...
        End If
    Next
End Sub

End Class

```

Compile the entire solution and execute the test program.