

HEALTH AUTHORITY ABU DHABI



SHAFAFIYA

Web Services Technical Definition

Version 1

Prepared By:

Injazat Data Systems

Date:

September - 2017

Revision History

Date	Version	Description	Author
03-07-2017	0.1	First Draft	Shafafiya Team
10-09-2017	1.0	First Release	Shafafiya Team

TABLE OF CONTENTS

-

REVISION HISTORY	2
1. INTRODUCTION	4
1.1. PURPOSE OF THIS DOCUMENT	4
2. SERVICE DESCRIPTIONS	5
2.1. UPLOAD TRANSACTIONS.....	5
2.1.1. <i>UploadTransaction Service</i>	5
2.2. DOWNLOAD TRANSACTIONS	6
2.2.1. <i>GetNewTransactions Service</i>	6
2.2.2. <i>DownloadTransactionFile Service</i>	7
2.2.3. <i>SetTransactionDownloaded Service</i>	7
2.2.4. <i>SearchTransactions Service</i>	8
2.3. DRG WEB SERVICES	10
2.3.1. <i>AddDRGToEClaim Service</i>	10
2.3.2. <i>GetDRGDetails Service</i>	11
2.4. PRIOR AUTHORIZATION TRANSACTIONS DOWNLOAD	12
2.4.1. <i>CheckForNewPriorAuthorizationTransactions Service</i>	12
2.4.2. <i>GetNewPriorAuthorizationTransactions Service</i>	13
2.5. CHECK FOR APPROVED TRADE DRUGS SERVICE	13
2.5.1. <i>GetPrescriptions Service</i>	13
2.6. NOTIFICATION SERVICE LINKS	14
2.6.2. <i>How to Register Your Notification Service Links</i>	15
2.7. WEB SERVICES RETURNED VALUES	15
2.8. WEB SERVICES ADDRESS	16
2.9. APPENDIX.....	17
2.9.1. <i>XML Format Download Transactions</i>	17
2.9.2. <i>Input XML Format</i>	17
2.9.3. <i>Output XML Format</i>	19
2.9.4. <i>Prescription.XSD</i>	19

1. Introduction

1.1. *Purpose of This Document*

This document contains the technical details about Shafafiya web services. This document shall be read by HAAD IT Operation, IT Security, HAAD partners willing to consume Shafafiya services and project manager.

Main audience includes:

- HAAD Business Users.
- HAAD Information Security team
- HAAD IT Operations team
- HAAD Project Manager
- Injazat Project Team.
- HAAD partners willing to consume Shafafiya services

2. Service Descriptions

2.1. Upload Transactions

This section provides details on Shafafiya Web Services related to transaction upload.

2.1.1. UploadTransaction Service

This web service is used to upload new transactions to the Shafafiya.

2.1.1.1. Service Signature

```
int UploadTransaction(string login,  
                     string pwd,  
                     byte[] fileContent,  
                     string fileName,  
                     out string errorMessage,  
                     out byte[] errorReport)
```

Attributes	Description
login	User login name to connect to the Shafafiya
pwd	User password to connect to the Shafafiya
fileContent	Content of the file to upload
fileName	Original file name; this file name will be used later to show users the available files for download
errorMessage	Output parameter that contains eventual error message if something goes wrong; web service can also send warnings using this parameter; the parameter may be null or empty string if the web service returns zero, otherwise the error message is expected;
errorReport	Output parameter that contains byte array with the content of error report eventually generated by Shafafiya Rule Validation during file upload; the parameter may be null

2.1.1.2. Service Output

The web service may return values described in the section *Web Services Returned Values*.

2.2. Download Transactions

This section provides details on Shafafiya Web Services related to transaction download.

2.2.1. GetNewTransactions Service

This web service is used to get a list of new transactions available on the Shafafiya. The service returns a list of transactions that haven't yet been marked as downloaded.

2.2.1.1. Service Signature

```
int GetNewTransactions(string login,  
                      string pwd,  
                      out string xmlTransactions,  
                      out string errorMessage)
```

Attributes	Description
login	User login name to connect to the Shafafiya
pwd	User password to connect to the Shafafiya
xmlTransactions	Output parameter that contains xml with description of the new e-claim files; the format of this xml is described in the section XML Format .
errorMessage	Output parameter that contains eventual error message if something goes wrong; web service can also send warnings using this parameter; the parameter may be null or empty string if the web service returns zero, otherwise the error message is expected.

2.2.1.2. Service Output

The web service may return values described in the section *Web Services Returned Values*.

2.2.2. DownloadTransactionFile Service

The web service returns content of e-claim file.

2.2.2.1. Service Signature

```
int DownloadTransactionFile(string login,  
                           string pwd,  
                           string fileID,  
                           out string fileName,  
                           out byte[] file,  
                           out string errorMessage)
```

Attributes	Description
login	User login name to connect to the Shafafiya
pwd	User password to connect to the Shafafiya
fileID	ID of transaction file to download; this id is received from xml returned by <i>GetNewTransactions</i> web services
fileName	Output parameter that contains downloaded file name
file	File content as byte array returned by the web server to the caller
errorMessage	Output parameter that contains eventual error message if something goes wrong; web service can also send warnings using this parameter; the parameter may be null or empty string if the web service returns zero, otherwise the error message is expected

2.2.2.2. Service Output

The web service may return values described in the section *Web Services Returned Values*.

2.2.3. SetTransactionDownloaded Service

The web service is used to report the e-claim file as downloaded. User is supposed to call this web service to avoid receiving the same e-claim file using *GetNewTransactions* web service.

2.2.3.1. Service Signature

```
int SetTransactionDownloaded(string login,  
                             string pwd,
```

```
string fileID,  
out string errorMessage)
```

Attributes	Description
login	User login name to connect to the Shafafiya
pwd	User password to connect to the Shafafiya
fileID	ID of the file to set as downloaded; this id is received from xml file returned by <i>GetNewTransactions</i> web services; file id is GUID similar to '836a5d03-958f-415c-81c4-7ed99fea3c53'
errorMessage	Output parameter that contains eventual error message if something goes wrong; web service can also send warnings using this parameter; the parameter may be null or empty string if the web service returns zero, otherwise the error message is expected

2.2.3.2. Service Output

The web service may return values described in the section *Web Services Returned Values*.

2.2.4. SearchTransactions Service

The web service performs advanced search for sent and received transactions.

2.2.4.1. Service Signature

```
int SearchTransactions( string login,  
                      string pwd,  
                      int direction,  
                      string callerLicense,  
                      string ePartner,  
                      int transactionID,  
                      int transactionStatus,  
                      string transactionFileName,  
                      string transactionFromDate,  
                      string transactionToDate,  
                      int minRecordCount,  
                      int maxRecordCount,  
                      out string foundTransactions,  
                      out string errorMessage)
```

Attributes	Description
login	User login name to connect to the Shafafiya
pwd	User password to connect to the Shafafiya
direction	Flag to indicate if function will search for sent or received transactions; the parameter may have values 1 (sent only) or 2 (received only)

callerLicense	<p>License of the caller; if the parameter is NULL, it is not used as search criteria; otherwise the parameter value must be in the list of licenses your Shafafiya login is associated with; caller license is used as follows:</p> <ul style="list-style-type: none"> - if direction= 1 (sent only), then caller license is used to find transactions sent by the caller; - if direction = 2 (received only), then caller license is used to find transactions received by the caller;
ePartner	<p>License of the ePartner; if parameter is NULL, it is not used in search criteria; otherwise ePartner license is used as follows:</p> <ul style="list-style-type: none"> - if direction = 1 (sent only), then ePartner license is used to find transactions sent to the ePartner; - if direction = 2 (received only), then ePartner license is used to find transactions received from the ePartner;
transactionID	<p>ID of transactions to search for; the parameter may contain more than one license as sum of below :</p> <ul style="list-style-type: none"> 2 - Claim Submission transaction only 4 - Person Register transaction only 8 - Remittance Advice transaction only 16 - Prior Request transaction only 32 - Prior Authorization transaction only <p>if this parameter is -1, then all transactions will be searched for</p>
transactionStatus	<p>Flag to indicate if function will search for new transactions or transactions already downloaded from the Shafafiya ; the parameter may have values 1 (new only) or 2 (already downloaded only)</p>
transactionFileName	<p>Any part of the transaction file name to search for; for example, string 'may' will return back all transactions that have word 'may' in their names, e.g. 'ClaimSubmissionMay2010.zip'; if the parameter is NULL, it is not used as search criteria</p>
transactionFromDate	<p>Transaction date from which the e-claim files will be found; the parameter may not be in the future; date must be provided as a string in a format 'dd/MM/yyyy hh:mm:ss'; if time is missed, the zero is used by default; if the parameter is NULL, it is set to 100 days before transactionToDate</p>
transactionToDate	<p>Transaction date until which the e-claim files will be found; the parameter may not be in the future; date must be provided as a string in a format 'dd/MM/yyyy hh:mm:ss'; if time is missed, the zero is used by default; if the parameter is NULL, it is set to current date</p>
minRecordCount	<p>Minimum number of records in the transaction file to search for. If the parameter is greater than zero, it must be less or equal to the value in the parameter maxRecordCount; if the parameter is -1, it is not used as search criteria</p>
maxRecordCount	<p>Maximum number of records in the transaction file to search for; if the parameter is greater than zero, it must be greater or</p>

	equal to the value in the parameter minRecordCount; if the parameter is -1, it is not used as search criteria
foundTransactions	Output parameter that contains xml with description of the found e-claim files; the format of this xml is described in the section XML Format ; note that this web service will return maximum 500 records
errorMessage	Output parameter that contains eventual error message if something goes wrong; web service can also send warnings using this parameter; the parameter may be null or empty string if the web service returns zero, otherwise the error message is expected

2.2.4.2. Service Output

The web service may return values described in the section *Web Services Returned Values*.

2.3. DRG Web Services

This section provides details on Shafafiya Web Services related to DRG.

2.3.1. AddDRGToEClaim Service

This web service is used to generate new e-claim file with DRG using the existing Claim Submission e- claim file.

2.3.1.1. Service Signature

```
int AddDRGToEClaim(string login,
                   string pwd,
                   byte[] originalFileContent, string
                   originalFileName, decimal
                   baseRate,
                   out byte[] drgFileContent, out
                   string drgFileName,
                   out byte[] auditFileContent, out
                   byte[] reportFileContent, out string
                   errorMessage,
                   out byte[] errorReport)
```

Attributes	Description
login	User login name to connect to the Shafafiya
pwd	User password to connect to the Shafafiya
originalFileContent	Content of the original e-claim file to use for DRG calculation; note that only Claim Submission transaction is supported for DRG calculation
originalFileName	Original e-claim file name
baseRate	Default rate in AED to be used for DRG net price calculation in all claims in the e-claim file; if baseRate is negative or zero, then the default base rate is used (the default base rate is 8,500 AED as per October 1 st , 2010)
drgFileContent	Output parameter that contains new e-claim file with generated DRG
drgFileName	Output parameter that contains name of the new e-claim file with generated DRG
auditFileContent	Output parameter that contains audit file generated by DRG Grouper; this file may be used for troubleshooting and understanding why certain DRG code is generated; the file content is returned back as zipped file
reportFileContent	Output parameter that contains report file generated by DRG Grouper; this file may be used for troubleshooting and understanding why

	certain DRG code is generated; the file content is returned back as zipped file
errorMessage	Output parameter that contains eventual error message if something goes wrong; web service can also send warnings using this parameter; the parameter may be null or empty string if the web service returns zero, otherwise the error message is expected
errorReport	Output parameter that contains byte array with the content of error report eventually generated by Shafafiya Rule Validation during file upload; the parameter may be null

2.3.1.2. Service Output

The web service may return values described in the section *Web Services Returned Values*.

2.3.2. GetDRGDetails Service

This web service is used to generate proper DRG code with respective weight.

2.3.2.1. Service Signature

```
int GetDRGDetails(string login,
                  string pwd,
                  string xmlClaimInput,
                  out string xmlDRGDetails,
                  out byte[] auditFileContent,
                  out byte[] reportFileContent,
                  out string errorMessage)
```

Attributes	Description
login	User login name to connect to the Shafafiya
pwd	User password to connect to the Shafafiya
xmlClaimInput	XML string that contains input data required for DRG grouping; refer to the section Input Xml Format for more details
xmlDRGDetails	XML string that contains grouped DRG code with respective weight; this parameter is null if at least one claim from the xmlClaimInput cannot be grouped; refer to the section Output Xml Format for more details
auditFileContent	Output parameter that contains audit file generated by DRG Grouper; this file may be used for troubleshooting and understanding why certain DRG code is generated; the file content is returned back as zipped file

reportFileContent	Output parameter that contains report file generated by DRG Grouper; this file may be used for troubleshooting and understanding why certain DRG code is generated; the file content is returned back as zipped file
errorMessage	Output parameter that contains eventual error message if something goes wrong; web service can also send warnings using this parameter; the parameter may be null or empty string if the web service returns zero, otherwise the error message is expected

2.3.2.2. Service Output

The web service may return values described in the section *Web Services Returned Values*.

2.4. Prior Authorization Transactions Download

This section provides details on Post Office Web Services related to Prior Request and Prior Authorization transactions download.

2.4.1. CheckForNewPriorAuthorizationTransactions Service

This web service is used to check if new Prior Request or Prior Authorization transactions are available on the Shafafiya. The service returns back a reply if at least one such transaction exists.

2.4.1.1. Service Signature

```
int CheckForNewPriorAuthorizationTransactions(string login,
                                             string pwd,
                                             out string errorMessage)
```

Attributes	Description
login	User login name to connect to the Shafafiya
pwd	User password to connect to the Shafafiya
errorMessage	Output parameter that contains eventual error message if something goes wrong; the parameter may be null or empty string if the web service returns zero or positive value, otherwise the error message is expected.

2.4.1.2. Service Output

The web service may return values described in the section *Web Services Returned Values*.

2.4.2. GetNewPriorAuthorizationTransactions Service

This web service is used to get a list of new Prior Request or Prior Authorization transactions available on the Shafafiya. The service returns back a list of transactions that haven't yet been downloaded.

2.4.2.1. Service Signature

```
int GetNewPriorAuthorizationTransactions(string login,  
                                       string pwd,  
                                       string foundTransactions,  
                                       out string errorMessage)
```

Attributes	Description
login	User login name to connect to the Shafafiya
pwd	User password to connect to the Shafafiya
foundTransactions	Output parameter that contains xml with description of the new e-claim files with Prior Request and Prior Authorization transactions; the format of this xml is described in the section XML Format ;
errorMessage	Output parameter that contains eventual error message if something goes wrong; the parameter may be null or empty string if the web service returns zero or positive value, otherwise the error message is expected

2.4.2.2. Service Output

The web service may return values described in the section *Web Services Returned Values*.

2.5. Check for Approved Trade Drugs Service

This section provides details on Shafafiya Web Services related to checking for approved trade drugs in electronic prescription.

2.5.1. GetPrescriptions Service

This web service is used to check Prescription transaction that contains a list of approved trade drugs available for dispensing. Currently this service is not in use.

2.5.1.1. Service Signature

```
int GetPrescriptions(string login,
                    string pwd,
                    string payerID,
                    string memberID,
                    out byte[] prescription,
                    out string errorMessage)
```

Attributes	Description
login	User login name to connect to the Shafafiya
pwd	User password to connect to the Shafafiya
payerID	License of payer the member belongs to
memberID	Patient insurance card number
prescription	Output parameter that contains Prescription transaction that contains approved drugs; Prescription transaction is sent as string
errorMessage	Output parameter that contains eventual error message if something goes wrong; the parameter may be null or empty string if the web service returns zero, otherwise the error message is expected.

2.5.1.1. Service Output

The web service may return values described in the section *Web Services Returned Values*. Get prescription output will be based on [Prescription.XSD](#).

2.6. Notification Service Links

When user sends e-claim transaction to Shafafiya Server, the server can notify the receiving party about new transaction waiting for download using the Notification Service.

2.6.1.1. Workflow Steps

The web service may return values described in the section *Web Services Returned Values*.

1. Payers and providers can specify http or https link that Shafafiya will call when new transaction is received. One link is used for Claim Submission and Remittance Advice transactions. Another link is used for Prior Request and Prior Authorization transactions.
2. Shafafiya calls payer's link when any provider sends new transaction for respective payer.
3. Shafafiya calls provider's link when the payer sends new transaction for respective provider.
4. Payers and providers are free to specify any hardcoded Notification Service Links, either http or https.

5. Notification Service Link for Claims and Notification Service Link for Requests can be the same link if you make sure your internal logic differentiates claims and requests.
6. Shafafiya will not accept dynamic links. All links must be provided as hardcoded web addresses.
7. No login and password are accepted to call this Link.
8. If payer or provider web site is down, Shafafiya will fail to call Notification Service Links. It is advised for payer and providers to check Shafafiya for new transactions every time their system is up and running again after down time.
9. If Notification Service Link is down permanently during 7 calendar days, these links are removed from Shafafiya database without any notice. It is responsibility of respective payer and provider to make sure their services are up and running.

2.6.2. How to Register Your Notification Service Links

You can register your notification service links by sending e-mail to Technical support e-mail: shafafiyasupport@haad.ae and providing Notification Service Link for Claims and Notification Service Link for Requests.

2.7. Web Services Returned Values

All Shafafiya web services share the return value, so that client application can implement one handler to process the result of web service execution and share it with all web service calls. The supported returned values are shown below:

Code	Description
0	Operation is successful
1	e-claim transaction validation succeeded with warnings
2	no new prior authorization transactions are available for download
3	Member has no approved trade drugs, hence Prescription transaction is not returned
4	DRG grouping is performed using default patient gender and age (female 21 years old)
-1	Login failed for the user
-2	e-claim transaction validation is failed with errors
-3	Required input parameter for the web service is empty, or null, or contains invalid value
-4	Unexpected error occurred
-5	if difference between date from and date to parameters is longer than 100 days
-6	The specified file is not found
-7	Transaction is not supported
-8	DRG Grouper is busy serving other requests; if you get this error code please try to call the web service again in 5-10 minutes
-9	Error occurred while running DRG Grouper
-10	if no search criteria is found

2.8. Web Services Address

Shafafiya web services are available on this address:

Web Services Address	https://shafafiya.haad.ae/v2/webservices.asmx	Production
Web Services WSDL	https://shafafiya.haad.ae/v2/webservices.asmx?WSDL	Production
Web Services Address	https://shafafiyapte.haad.ae/v2/webservices.asmx	Staging
Web Services WSDL	https://shafafiyapte.haad.ae/v2/webservices.asmx?WSDL	Staging

2.9. Appendix

2.9.1. XML Format Download Transactions

Parameter xml Transaction in the web service is expected to have the following format:

```
<Files>
  <File FileID='31830' FileName='MyFile1.xml' SenderID='MF2057'
ReceiverID='A001' TransactionDate='11/04/2010 10:44:03' RecordCount='15'
IsDownloaded='True' />
  <File FileID='31824' FileName='MyFile2.xml' SenderID='MF2102'
ReceiverID='A001' TransactionDate='12/04/2010 18:12:10' RecordCount='8123'
IsDownloaded='False' />
  <File FileID='31821' FileName='MyFile3.xml' SenderID='MF2058'
ReceiverID='A001' TransactionDate='31/03/2010 16:59:31' RecordCount='6930'
IsDownloaded='True' />
</Files>
```

XML Fields Description:

FileID– unique file id that will be used later to download the file;

FileName– original file name as it was uploaded by the sender;

SenderID– e-claim file sender;

ReceiverID– e-claim file receiver;

TransactionDate– value from e-claim file's TransactionDate node;

RecordCount– value from e-claim file's TransactionDate node;

IsDownloaded– True if file has been already downloaded from the Shafafiya ; False otherwise. Note that this attribute is provided only for the web service *SearchTransactions*.

2.9.2. Input XML Format

Example of the input xml string is presented below.

```
<?xml version="1.0" encoding="utf-8" ?>
<Claims>
  <Claim>
    <ID>26-S-0011</ID>
    <MemberBirthDate>29/09/1980</MemberBirthDate>
```

```

    <MemberGender>1</MemberGender>
    <EncounterEndType>1</EncounterEndType>
    <EncounterStart>21/05/2012 14:20</EncounterStart>
    <EncounterEnd>28/05/2012 16:15</EncounterEnd>
    <PrincipalDiagnosis>003.21</PrincipalDiagnosis>
    <SecondaryDiagnosis>005.9</SecondaryDiagnosis>
    <SecondaryDiagnosis>V23.41</SecondaryDiagnosis>
    <AdmittingDiagnosis>E829.4</AdmittingDiagnosis>
    <AdmittingDiagnosis>930.8</AdmittingDiagnosis>
    <ActivityCode>84295</ActivityCode>
    <ActivityCode>86140</ActivityCode>
  </Claim>
</Claims>

```

The input xml has the following nodes and elements. Note that all below elements are mandatory if not mentioned otherwise:

1. Claims– main parent node in the xml. Only one such node is allowed in xml.
2. Claim– claim node that contains all claim data required for DRG grouping. There may be one or more claims in the xml.
3. ID– claim ID. This node has the same meaning and rules as node ClaimID in the original Claim Submission transaction.
4. MemberBirthDate– birth date of the member in a format dd/MM/yyyy.
5. MemberGender– gender of the member. 0 means female, 1 means male.
6. EncounterEndType– end type of the encounter. This node has the same meaning and rules as node EncounterEndType in the original Claim Submission transaction.
7. EncounterStart– start date of the encounter. This node has the same meaning and rules as node EncounterStart in the original Claim Submission transaction. The value is provided in a format dd/MM/yyyy hh:mm.
8. EncounterEnd– end date of the encounter. This node has the same meaning and rules as node EncounterEnd in the original Claim Submission transaction. The value is provided in a format dd/MM/yyyy hh:mm.
9. PrincipalDiagnosis– principal diagnosis node. Every claim must have one and only one principal diagnosis node.
10. SecondaryDiagnosis– secondary diagnosis node. There might be zero or more secondary diagnoses per claim.

11. AdmittingDiagnosis– admitting diagnosis node. There might be zero or more admitting diagnoses per claim.
12. ActivityCode– CPT code of the procedure. This node has the same meaning and rules as node ActivityCode in the original Claim Submission transaction for CPT codes.

2.9.3. Output XML Format

Example of the output xml string is presented below.

```
<?xml version="1.0" encoding="utf-8" ?>
<Claims>
  <Claim>
    <ID>26-S-0011</ID>
    <DRGCode>011132</DRGCode>
    <DRGWeight>0.4511401078</DRGWeight>
  </Claim>
</Claims>
```

The output xml has the following nodes and elements. Note that all below elements are mandatory if not mentioned otherwise:

1. Claims– main parent node in the xml. Only one such node is allowed in xml.
2. Claim– claim node that contains claim data with grouped DRG. Number of claim nodes in output xml is the same as number of claim nodes in the input xml, assuming all claims are grouped successfully.
3. ID– claim ID. This node has the same meaning and rules as the respective node in the input xml.
4. DRGCode– DRG code that is generated by DRG Grouping software.
5. DRGWeight– weight for the DRG as per respective Mandatory Tariff file.

2.9.4. Prescription.XSD

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:tns="http://www.haad.ae/DataDictionary/CommonTypes"
  elementFormDefault="qualified" version="2.0" id="ElectronicPrescription"
  xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:xsd="undefined">
```

```

<xs:import schemaLocation="CommonTypes.xsd"
namespace="http://www.haad.ae/DataDictionary/CommonTypes"/>
<xs:element name="Electronic.Prescription">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Header">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="SenderID" type="tns:HeaderSenderID"/>
            <xs:element name="ReceiverID" type="tns:HeaderReceiverID"/>
            <xs:element name="TransactionDate" type="tns:HeaderTransactionDate"/>
            <xs:element name="RecordCount" type="tns:HeaderRecordCount"/>
            <xs:element name="DispositionFlag" type="tns:HeaderDispositionFlag"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="Authorization">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="ID" type="tns:AuthorizationID"/>
            <xs:element name="IDPayer" type="tns:AuthorizationIDPayer"/>
            <xs:element name="MemberID" type="tns:AuthorizationMemberID"/>
            <xs:element name="PayerID" type="tns:AuthorizationPayerID"/>
            <xs:element name="DateOrdered" type="tns:AuthorizationDateOrdered"/>
            <xs:element name="Start" type="tns:AuthorizationStart"/>
            <xs:element name="End" type="tns:AuthorizationEnd"/>
            <xs:element minOccurs="0" name="Comments"
type="tns:AuthorizationComments"/>
            <xs:element minOccurs="0" name="Encounter">
              <xs:complexType>
                <xs:sequence>
                  <xs:element name="FacilityID"
type="tns:EncounterFacilityID"/>
                  <xs:element name="Type" type="tns:EncounterType"/>
                </xs:sequence>
              </xs:complexType>
            </xs:element>
            <xs:element minOccurs="0" maxOccurs="unbounded"
name="Diagnosis">
              <xs:complexType>
                <xs:sequence>
                  <xs:element name="Type"
type="tns:DiagnosisType"/>
                  <xs:element name="Code"
type="tns:DiagnosisCode"/>
                </xs:sequence>
              </xs:complexType>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```

                <xs:element minOccurs="0" maxOccurs="unbounded" name="GenericDrug">
                    <xs:complexType>
                        <xs:sequence>
                            <xs:element name="ID"
                                type="tns:ActivityID"/>
                            <xs:element name="Code"
                                type="tns:ActivityCode"/>
                            <xs:element name="Quantity"
                                type="tns:ActivityQuantity"/>
                            <xs:element minOccurs="0"
                                name="DenialCode" type="tns:ActivityDenialCode"/>
                            <xs:element minOccurs="5"
                                maxOccurs="6" name="GenericDrugObservation">
                                    <xs:complexType>
                                        <xs:sequence>
                                            <xs:element
                                                name="Type" type="tns:ObservationType"/>
                                            <xs:element
                                                name="Code" type="tns:ObservationCode"/>
                                            <xs:element
                                                name="Value" type="tns:ObservationValue"/>
                                            <xs:element
                                                name="ValueType" type="tns:ObservationValueType"/>
                                        </xs:sequence>
                                    </xs:complexType>
                                </xs:element>
                            <xs:element minOccurs="1"
                                maxOccurs="unbounded" name="TradeDrug">
                                    <xs:complexType>
                                        <xs:sequence>
                                            <xs:element
                                                name="ID" type="tns:ActivityID"/>
                                            <xs:element
                                                name="Code" type="tns:ActivityCode"/>
                                            <xs:element
                                                name="Quantity" type="tns:ActivityQuantity"/>
                                            <xs:element
                                                name="Net" type="tns:ActivityNet"/>
                                        </xs:sequence>
                                    </xs:complexType>
                                </xs:element>
                            <xs:element minOccurs="0" name="TradeDrugObservation">
                                    <xs:complexType>
                                        <xs:sequence>
                                            <xs:element name="Type" type="tns:ObservationType"/>
                                            <xs:element name="Code" type="tns:ObservationCode"/>
                                        </xs:sequence>
                                    </xs:complexType>
                                </xs:element>
                        </xs:sequence>
                    </xs:complexType>
                </xs:element>
            </xs:sequence>
        </xs:complexType>
    </xs:element>

```

```
<xs:element name="Value" type="tns:ObservationValue"/>
<xs:element name="ValueType" type="tns:ObservationValueType"/>
</xs:sequence>
</xs:complexType>
                                        </xs:element>
                                </xs:sequence>
                        </xs:complexType>
                </xs:element>
                </xs:sequence>
                </xs:complexType>
        </xs:element>
        </xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>
```