



# NUANCE

The experience speaks for itself™

***NDEV Mobile***

***HTTP Services for NDEV Mobile  
Clients***

## **Notice**

NDEV Mobile

HTTP Services for NDEV Mobile Clients

Copyright © 2011 - 2012 Nuance Communications, Inc. All rights reserved.

Published by Nuance Communications, Inc.

One Wayside Road, Burlington, Massachusetts 01803, U.S.A.

Last updated February 10, 2012.

Nuance Communications, Inc. provides this document without representation or warranty of any kind. The information in this document is subject to change without notice and does not represent a commitment by Nuance Communications, Inc. The software and/or databases described in this document are furnished under a license agreement and may be used or copied only in accordance with the terms of such license agreement. Without limiting the rights under copyright reserved herein, and except as permitted by such license agreement, no part of this document may be reproduced or transmitted in any form or by any means, including, without limitation, electronic, mechanical, photocopying, recording, or otherwise, or transferred to information storage and retrieval systems, without the prior written permission of Nuance Communications, Inc.

Nuance, the Nuance logo, and Nuance Vocalizer are trademarks or registered trademarks of Nuance Communications, Inc. or its affiliates in the United States and/or other countries. All other trademarks referenced herein are the property of their respective owners.

## Table of Contents

<b>1</b>	<b>HTTP client for dictation web service</b>	<b>1</b>
1.1	Requirements	1
1.1.1	Chunked Transfer Coding	1
1.1.2	HTTP Over TLS	1
1.2	Dictation	1
1.3	Dictation HTTP request message	2
1.3.1	Request query string	2
1.3.2	Request headers	3
1.3.3	Request message body	4
1.4	Dictation HTTP response message	5
1.4.1	Response headers	5
1.4.2	Response message body	5
1.4.3	Response HTTP status codes	5
<b>2</b>	<b>HTTP client for Text to Speech web service</b>	<b>7</b>
2.1	Requirements	7
2.1.1	HTTP Over TLS	7
2.2	Text to Speech (TTS)	7
2.3	TTS HTTP request message	8
2.3.1	Request query string	8
2.3.2	Request headers	10
2.3.3	Request message body	11
2.4	TTS HTTP response message	11
2.4.1	Response headers	11
2.4.2	Response message body	11
2.4.3	Response HTTP status codes	11
<b>3</b>	<b>Language support</b>	<b>13</b>
3.1	Current language support	13
3.2	Future language support	14

## 1 HTTP client for dictation web service

This document section describes how a HyperText Transfer Protocol (HTTP) client accesses NDev Mobile Platform web service to perform dictation based on an audio stream.

### 1.1 Requirements

This section outlines requirements.

#### 1.1.1 Chunked Transfer Coding

The HTTP client must support HTTP 1.1 (RFC2616) and chunked transfer coding ([www.w3.org/Protocols/rfc2616/rfc2616-sec3.html#sec3.6.1](http://www.w3.org/Protocols/rfc2616/rfc2616-sec3.html#sec3.6.1)).

Because the recorded audio stream length may not be known in advance, chunked encoding modifies the body of an HTTP message in order to transfer it as a series of chunks, each with its own size indicator.

For more information on chunked transfer coding, visit these links:

- [four.livejournal.com/887211.html](http://four.livejournal.com/887211.html)
- [developers.sun.com/mobility/midp/questions/chunking/](http://developers.sun.com/mobility/midp/questions/chunking/)
- [en.wikipedia.org/wiki/Chunked\\_transfer\\_encoding](http://en.wikipedia.org/wiki/Chunked_transfer_encoding)

#### 1.1.2 HTTP Over TLS

The HTTP client must support HTTP over TLS (also known as HTTPS).

HTTPS provides encrypted communication and secure identification with the NDEV web service.

For more information on HTTP over TLS, visit these links:

- HTTP Over TLS ([www.ietf.org/rfc/rfc2818.txt](http://www.ietf.org/rfc/rfc2818.txt))
- HTTP Secure ([en.wikipedia.org/wiki/HTTP\\_Secure](http://en.wikipedia.org/wiki/HTTP_Secure))

### 1.2 Dictation

The dictation web service receives an incoming voice stream from an HTTP client and provides the client with a transcription of what was spoken.

Here is an example of a dictation HTTP POST request and the response from the server:

**Request:**

```
https://dictation.nuancemobility.net/NMDPASrCmdServlet/dictation?app
Id=NMAID_FOO&appKey=525348e77144a9cee9a7471a8b67c50ea85b9e3eb377a3c2
a3a23dc88f9150eefe76e6a339fdb62b817595f53d72549d9ebe36438f8c2619846
b963e9f43a93&id=57349abd2390 HTTP/1.1
```

Transfer-Encoding: chunked

Content-Type: audio/x-pcm;bit=16;rate=8000

Accept: text/plain

Accept-Language: en-US Heading 1

800

[audio data]

4

[audio data]

0

**Response:**

HTTP/1.1 200 OK

Date: Tue, 31 Aug 2010 22:50:35 GMT

Content-Type: text/plain;charset=utf-8

Content-Language: en-US

Content-Length: 11

x-nuance-sessionid: 97bd6505-b7d6-420a-8eb7-7583036f7aa1

Hello world

## 1.3 Dictation HTTP request message

An HTTP POST request message is used by the client to access the dictation web service.

### 1.3.1 Request query string

A query string is the part of a Uniform Resource Locator (URL) that contains data to be passed to web applications. Three <key>=<value> pairs separated by ampersands must be set to access the NDEV Mobile web service:

- appId
- appKey (this needs to be formatted as a **128-byte** string for this interface to work.)
- id

For example:

**appId**=NMAID\_FOO&**appKey**=525348e77144a9cee9a7471a8b67c50ea85b9e3eb377a3c2a3a23dc88f9150eefe76e6a339fdb62b817595f53d72549d9ebe36438f8c2619846b963e9f43a93&**id**=57349abd2390

For more information on query strings, see these documents:

- Query strings: ([en.wikipedia.org/wiki/Query\\_string](http://en.wikipedia.org/wiki/Query_string))
- URI schemes: ([en.wikipedia.org/wiki/URI\\_scheme](http://en.wikipedia.org/wiki/URI_scheme))

### 1.3.1.1 appld and appKey

The application ID and application key provided by Nuance are used for authentication with the NDEV web service.

**The application key must be formatted as a 128-byte string.**

If the appld and appKey key-value pairs included with the HTTP POST request are missing, invalid, or not formatted properly, the NDEV web service will return an HTTP response with status code 401 Unauthorized (see 401 Unauthorized on page 5).

### 1.3.1.2 id

A unique ID is required to identify the device (for example, serial number, International Mobile Equipment Identity or IMEI) or the user.

## 1.3.2 Request headers

This section provides a list of headers to include in the HTTP POST request for dictation services.

### 1.3.2.1 Content-Type

The *Content-Type* header is used to specify the audio format from the client incoming voice stream. The supported formats for US English are:

Header Value	Description
audio/x-wav; codec=pcm; bit=16; rate=8000	8Khz 16bit PCM
audio/x-wav; codec=pcm; bit=16; rate=16000	16Khz 16bit PCM
audio/x-speex; rate=8000	Speex narrowband
audio/x-speex; rate=16000	Speex wideband
audio/amr	AMR
audio/qcelp	QCELP
audio/evrc	EVRC

For all other languages, the supported formats are:

Header Value	Description
audio/x-wav; codec=pcm; bit=16; rate=16000	16Khz 16bit PCM
audio/x-speex; rate=16000	Speex wideband

**Note:** Only the raw audio is to be sent, without the header.

#### 1.3.2.2 Accept

The *Accept* request-header field can be used to specify specific dictation result types that are acceptable for the response.

The supported formats are:

- application/xml
- text/plain

**Note:** Regardless of the value you use, dictation results are always returned as plain text. Responses that contain a list of possible dictation results will return each result as a text string separated by a new line. A future release of the NDEV HTTP Services Interface will support results returned in an XML format.

#### 1.3.2.3 Accept-Language

The *Accept-Language* request-header represents the language used to record the audio and restricts the natural language expected in the dictation result. For a list of supported languages and the language codes that can be assigned to this header, see the FAQ page on the NDEV Mobile website.

#### 1.3.2.4 Accept-Topic

The *Accept-Topic* request-header represents the type of topic used by the recognition engine.

The current supported values are: *Dictation* and *WebSearch*. If the value assigned to the header is invalid, the status code 406 Not Acceptable is returned.

#### 1.3.2.5 Transfer-Encoding

Unless the total length of the voice input stream is known, *Transfer-Encoding: chunked* is a mandatory header in the HTTP POST request.

#### 1.3.2.6 Content-Length

The *Content-Length* entity-header field indicates the size of the voice input stream if it is known in advance.

When *Content-Length* is specified, the *Transfer-Encoding* header field must not be used.

### 1.3.3 Request message body

The message-body of the HTTP POST message is used to carry the voice input stream associated with the request. The message-body differs from the entity-body only when *Transfer-Encoding: chunked* has been applied (see [tools.ietf.org/html/rfc2616#section-3.6.1](http://tools.ietf.org/html/rfc2616#section-3.6.1)).

The presence of a message-body in a request is signaled by the inclusion of a *Content-Length* or *Transfer-Encoding* header field in the request's message-headers.

*Transfer-Encoding: chunked* is the preferred way to send the voice input stream and should be sent in chunks of ~260 ms of audio duration (that is, 4160 bytes of PCM 16 bits 8 kHz) when the user speaks.

When *Transfer-Encoding: chunked* is not used, the message body includes the complete voice input stream recorded with the Content-Length.

**Note:** The HTTP client interface does not provide capture of the audio stream. It is the responsibility of the developer using this interface to build the audio capture application layer that will interact with this HTTP interface.

## 1.4 Dictation HTTP response message

An HTTP POST response message is sent to the client. The response message contains status information about the completion of the request.

### 1.4.1 Response headers

This section provides a list of headers specified by the server for a dictation HTTP POST response message.

#### 1.4.1.1 Content-Type

See [Accept \(above\)](#) for supported values.

**Note:** Nuance only supports UTF-8 for the character set.

#### 1.4.1.2 Content-Language

See [Accept-Language \(above\)](#) for supported values.

#### 1.4.1.3 x-nuance-sessionid

This header is used as a session ID from the server to identify the transaction with the dictation web service. It can be used to retrieve the call log associated with a dictation request.

### 1.4.2 Response message body

The message-body of the HTTP POST response is used to carry the dictation result of what was spoken. The format is specified by the *Content-Type* header.

### 1.4.3 Response HTTP status codes

This section describes the status codes of the HTTP response.

#### 1.4.3.1 2xx Success

The 2xx class of status codes indicates that the action requested by the client was received, understood, accepted, and processed successfully.

##### 1.4.3.1.1 200 OK

Standard response for a successful dictation HTTP POST request.



#### **1.4.3.1.2 204 No Content**

The server successfully processed the request, but is not returning any content.

#### **1.4.3.2 4xx Client Error**

The 4xx class of status codes is intended for cases in which the client seems to have erred.

##### **1.4.3.2.1 400 Bad Request**

The request cannot be fulfilled due to bad syntax.

##### **1.4.3.2.2 401 Unauthorized**

Used when authentication is possible but has failed or not yet been provided.

##### **1.4.3.2.3 403 Forbidden**

The request was a legal request, but the server is refusing to respond to it.

##### **1.4.3.2.4 404 Not Found**

The dictation resource could not be found but may be available again in the future.

##### **1.4.3.2.5 405 Method Not Allowed**

A request was made of a dictation resource using a request method not supported; for example, using GET instead of POST

##### **1.4.3.2.6 406 Not Acceptable**

The dictation resource is only capable of generating content not acceptable according to the Accept headers sent in the request.

##### **1.4.3.2.7 408 Request Timeout**

The server timed out waiting for the request.

##### **1.4.3.2.8 410 Gone**

Indicates that the resource requested is no longer available and will not be available again.

##### **1.4.3.2.9 413 Request Entity Too Large**

The request is larger than the server is willing or able to process.

##### **1.4.3.2.10 414 Request-URI Too Long**

The URI provided was too long for the server to process.

##### **1.4.3.2.11 415 Unsupported Media Type**

The request entity has a media type which the server or resource does not support.

#### **1.4.3.3 5xx Server Error**

The 5xx class of status codes indicates that the server failed to fulfill an apparently valid request.

#### **1.4.3.3.1 500 Internal Server Error**

A generic error message given when no specific error message is suitable.

#### **1.4.3.3.2 501 Not Implemented**

The server either does not recognize the request method, or it lacks the ability to fulfill the request.

#### **1.4.3.3.3 503 Service Unavailable**

The server is currently unavailable (because it is overloaded or down for maintenance). Generally, this is a temporary state.

#### **1.4.3.3.4 504 Gateway Timeout**

The server was acting as a proxy and did not receive a timely response from the upstream server.

#### **1.4.3.3.5 505 HTTP Version Not Supported**

The server does not support the HTTP protocol version used in the request.

## **2 HTTP client for Text to Speech web service**

This document section describes how a HyperText Transfer Protocol (HTTP) client accesses the NDEV Mobile web service to perform a Text to Speech (TTS) conversion.

### **2.1 Requirements**

This section outlines the requirements for an HTTP client for a TTS web service.

#### **2.1.1 HTTP Over TLS**

The HTTP client must support HTTP over TLS (also known as HTTPS).

HTTPS provides encrypted communication and secure identification with the NDEV web service.

For more information on HTTP over TLS, visit these links:

- HTTP Over TLS ([www.ietf.org/rfc/rfc2818.txt](http://www.ietf.org/rfc/rfc2818.txt))
- HTTP Secure ([en.wikipedia.org/wiki/HTTP\\_Secure](http://en.wikipedia.org/wiki/HTTP_Secure))

### **2.2 Text to Speech (TTS)**

The TTS web service receives a text message from a HTTP client and provides the client with an audio stream of the text sent.

Here is an example of a TTS HTTP POST request and the response from the server:

#### **Request:**

```
https://tts.nuancemobility.net:443/NMDPTTSCmdServlet/tts?appId=NMAID_FOO&appKey=25348e77144a9cee9a7471a8b67c50ea85b9e3eb377a3c2a3a23dc88
```

```
f9150eefe76e6a339fdb62b817595f53d72549d9ebe36438f8c2619846b963e9f43
a9&id=57349abd2390&ttsLang=en_US HTTP/1.1
```

```
Content-Type: text/plain
```

```
Accept: audio/x-wav
```

```
Hello world
```

### Response:

```
HTTP/1.1 200 OK
```

```
Date: Tue, 31 Aug 2010 22:50:35 GMT
```

```
x-nuance-sessionid: 97bd6505-b7d6-420a-8eb7-7583036f7aa1
```

```
-----
Response content length: 804
```

```
Chunked?: 0
-----
```

```
..x.....#...|. ....:.....0.4.....p.....f.....k.;.
.....D.9...).*. ....D.....N...). \...}.....+...{...
|. ....
.T.'8.M.}...O...{. `...h.B.x.....O.....6.....W...T...F.,.....`
.j.....`.$.....G...{...m...`...p....."....`l./.....i...h....
...m...7.....q.....D.]'.].X...2.....g.D.....
...l...o.A.U.....5.B.9...~....._.....B.....A...q...J.

.
... \.....o...../."...N.-
.P.L.....Q.B...Y.=.#.....>.Q.].....:'. ....A...6...u.....R..
...}.....w...g...I.m.....
..._F.J...'.0.Q.A.....~.y...[.....;-
.#.....(.5.c.).....W.....~.x.....n
.....3.N.H.4.....
```

## 2.3 TTS HTTP request message

An HTTP POST request message is used by the client to access the TTS web service.

### 2.3.1 Request query string

A query string is the part of a Uniform Resource Locator (URL) that contains data to be passed to web applications. To access the NDEV Mobile web service, set <key>=<value> pairs, separated by ampersands, for the following:

- appld
- appKey (this needs to be formatted as a **128-byte** string for this interface to work.)
- id

- {ttsLang|voice} Provide either a language or specific voice talent to use. If both are present, voice will override ttsLang.

Two examples of query strings to access the TTS web service:

**appId=NMAID\_FOO&appKey=525348e77144a9cee9a7471a8b67c50ea85b9e3eb377a3c2a3a23dc88f9150eefe76e6a339fdb62b817595f53d72549d9ebe36438f8c2619846b963e9f43a93&id=57349abd2390&ttsLang=en\_US**

**appId=NMAID\_FOO&appKey=525348e77144a9cee9a7471a8b67c50ea85b9e3eb377a3c2a3a23dc88f9150eefe76e6a339fdb62b817595f53d72549d9ebe36438f8c2619846b963e9f43a93&id=57349abd2390&voice=Samantha**

For more information on query strings, see these documents:

- Query strings: ([en.wikipedia.org/wiki/Query\\_string](http://en.wikipedia.org/wiki/Query_string))
- URI schemes: ([en.wikipedia.org/wiki/URI\\_scheme](http://en.wikipedia.org/wiki/URI_scheme))

#### 2.3.1.1 appId and appKey

The application ID and application key provided by Nuance are used for authentication with the Nuance Mobile Speech Platform (NMSP) web service. The NMSP web service returns an HTTP response with status code 401 Unauthorized if the appId and appKey key-value pairs with the HTTP POST request are missing or invalid.

#### 2.3.1.2 id

A unique ID is required to identify a device (for example, serial number, International Mobile Equipment Identity or IMEI) or a user.

#### 2.3.1.3 ttsLang

Defines the TTS language. For example, for North-American English the value is: *enus*.

The TTS language definition depends on the supported languages from the NMSP web service. If you include the *ttsLang* and the (optional) *voice* parameter in the request query string, then the *voice* parameter will override the *ttsLang* parameter.

For a list of currently supported languages and associated language codes you can assign to the *ttsLang* parameter, see the FAQ page on the NDEV website. As well, for a discussion of languages that will be supported in the future, see the FAQ page.

#### 2.3.1.4 voice

The type of voice used to generate the TTS audio response. For example, for North-American English there are two voice types:

- "Samantha"
- "Tom"

The *voice* parameter identifies the language that is used to generate the audio response. If you include the *voice* parameter in the request query string, it will override the *ttsLang* parameter.

For a list of currently supported languages and associated language codes you can assign to the *voice* parameter, see the FAQ page on the NDEV website. As well, for a discussion of languages that will be supported in the future, see the FAQ page.

## 2.3.2 Request headers

The headers to include in the HTTP POST request are described below.

### 2.3.2.1 Content-Type

The *Content-Type* header specifies the type of text used for audio speech generation. The supported formats are:

- `text/plain` — plain text
- `message/external-body` — the URI where the text file is located.

**Note:** The default value is `text/plain` if the header is missing or invalid.

### 2.3.2.2 Accept

The *Accept* request header field can be used to specify the response audio format for the outgoing audio stream sent to the client. The supported formats are:

Header Value	Description
<code>audio/x-wav; codec=pcm; bit=16; rate=8000</code>	8Khz 16bit PCM. Defaults to a Microsoft .wav file.  A raw stream of 16-bit signed integers at 8000 samples per sec. suitable for streaming into an audio I/O stream without saving to a file, and then invoking a media player. PCM audio streams do not have any sort of file-header— just the raw audio data. If you save them to a file and then try to load them into a media player, you have to explicitly tell the media player the data format.
<code>audio/x-wav; codec=pcm; bit=16; rate=22000</code>	22Khz 16bit PCM. Defaults to a Microsoft .wav file.
<code>audio/x-speex; rate=8000</code>	Speex narrowband

audio/x-speech;rate=16000	Speex wideband
audio/amr	AMR
audio/qcelp	QCELP
audio/evrc	EVRC

**Note:** Only the raw audio is returned, no header.

The default value is 8Khz 16bit PCM if the header is missing.

### 2.3.3 Request message body

The message-body of the HTTP POST message is used to carry the text that is transferred to speech, depending on the *Content-Type*.

## 2.4 TTS HTTP response message

The TTS response message is an HTTP POST that is sent to the client. The response message contains status information about the completion of the TTS request.

### 2.4.1 Response headers

The headers specified by the server for a TTS HTTP POST response message are described below.

#### 2.4.1.1 Content-Type

See Accept on page 4.

**Note:** Nuance only supports UTF-8 for the character set.

x-nuance-sessionid

This header is used as a session ID from the server to identify the transaction with the TTS web service. It can be used to retrieve the call log associated with a TTS request.

### 2.4.2 Response message body

The message-body of the HTTP POST response contains the TTS result of the text provided in the request.

### 2.4.3 Response HTTP status codes

This section describes the status codes of the HTTP response.

#### 2.4.3.1 2xx Success

The 2xx class of status codes indicates that the action requested by the client was received, understood, accepted, and processed successfully.

#### **2.4.3.1.1 200 OK**

Standard response for successful TTS HTTP POST request.

#### **2.4.3.1.2 204 No Content**

The server successfully processed the request, but is not returning any content.

#### **2.4.3.2 4xx Client Error**

The 4xx class of status code is intended for cases in which the client seems to have erred.

##### **2.4.3.2.1 400 Bad Request**

The request cannot be fulfilled due to bad syntax.

##### **2.4.3.2.2 401 Unauthorized**

Used when authentication is possible but has failed or not yet been provided.

##### **2.4.3.2.3 403 Forbidden**

The request was a legal request, but the server is refusing to respond to it.

##### **2.4.3.2.4 404 Not Found**

The TTS resource could not be found but may be available again in the future.

##### **2.4.3.2.5 405 Method Not Allowed**

A request was made of a TTS resource using a request method not supported; for example, using GET instead of POST.

##### **2.4.3.2.6 406 Not Acceptable**

The TTS resource is only capable of generating content not acceptable according to the Accept headers sent in the request.

##### **2.4.3.2.7 408 Request Timeout**

The server timed out waiting for the request.

##### **2.4.3.2.8 410 Gone**

Indicates that the resource requested is no longer available and will not be available again.

##### **2.4.3.2.9 413 Request Entity Too Large**

The request is larger than the server is willing or able to process.

##### **2.4.3.2.10 414 Request-URI Too Long**

The URI provided was too long for the server to process.

##### **2.4.3.2.11 415 Unsupported Media Type**

The request entity has a media type which the server or resource does not support.

### 2.4.3.3 5xx Server Error

The 5xx class of status codes indicates that the server failed to fulfill an apparently valid request.

#### 2.4.3.3.1 500 Internal Server Error

A generic error message, given when no more specific message is suitable.

#### 2.4.3.3.2 501 Not Implemented

The server either does not recognize the request method, or it lacks the ability to fulfill the request.

#### 2.4.3.3.3 503 Service Unavailable

The server is currently unavailable (because it is overloaded or down for maintenance). Generally, this is a temporary state.

#### 2.4.3.3.4 504 Gateway Timeout

The server was acting as a proxy and did not receive a timely response from the upstream server.

#### 2.4.3.3.5 505 HTTP Version Not Supported

The server does not support the HTTP protocol version used in the request.

## 3 Language support

This section addresses the current and future languages supported by the NDEV web service and the codes used by NDEV to represent those languages.

The language codes used by the NDEV web service are derived from two standards published by the International Standards Organization (ISO):

- ISO-3166-1 — *Codes for the representation of names of countries and their subdivisions – Part 1: Country codes*
- ISO-639 — *Codes for the representation of names of languages*

### 3.1 Current language support

The FAQ page on the NDEV website lists the codes for currently supported languages. Use these codes to assign a value to:

- The Accept-Language header in the HTTP POST request for dictation services (see Accept-Language on page 4)
- The ttsLang and voice parameters in the request query string to access text-to-speech (TTS) web services (see Request query string on page 8).

The codes to identify a language to the NDEV web service are a combination of these ISO country and language codes:

- ISO 3166-1, Alpha-2, two-letter country codes.



- ISO 639-1, Alpha-2, two-letter language codes.

### 3.2 Future language support

The plan for future versions of the NDEV web service is to support the languages that will be listed on the FAQ page of the NDEV website. The codes to identify languages in future versions of the NDEV web service will be a combination of these ISO codes:

- ISO 639-2, Alpha-3, three-letter language code (lower-case).
- ISO 3166-1, Alpha-3, three-letter country code (upper-case).

In future versions of the NDEV web service, the language and country codes are separated by a hyphen. Future versions of the NDEV web service will support the four-letter language codes used in the current version of the NDEV web service.

Each language categorized by the ISO 639-2 standard can have one or both of the following three-letter codes:

- A bibliographic code (ISO 639-2/B), derived from the English name for the language. All languages have an ISO 639-2/B code.
- A terminological code (ISO 639-2/T), derived from the native name for the language. Only a few languages have an ISO 639-2/T code. In cases where the language has both types of code, usually the ISO 639-2/T code is used.