



# Communiceren met Niki

## Handleiding voor een abonnee

Auteur:	Eelco Hoekstra, Dennis Tap, Marcel Mulder
Datum:	17 February 2012
Versie:	1.8.5
Niki:	3.19.0

## Inhoud

1	Inleiding.....	3
2	Globale beschrijving.....	4
3	Feeds.....	5
4	Projecten invoeren.....	7
4.1	Inleiding.....	7
4.2	Validatie.....	7
4.3	Beschrijving van de XML.....	8
4.3.1	Project algemeen.....	8
4.3.2	Autorisatie.....	9
4.3.3	Projectgegevens.....	9
4.3.3.1	Project stadia & datums.....	10
4.3.3.2	Projectbeschrijving.....	10
4.3.3.3	Actieregel.....	10
4.3.3.4	Woningsoorten.....	11
4.3.3.5	Inschrijvingen.....	11
4.3.3.6	Gekoppelde makelaars, opdrachtgevers en bestanden.....	11
4.3.3.7	Locatie.....	12
4.3.4	Woningtypes.....	14
4.3.5	Woningen.....	15
4.3.6	Bestanden.....	16
4.4	Kleine updates.....	18
4.4.1	Richtlijnen.....	18
5	Beveiliging.....	23
6	XML opsturen.....	24
7	Testen.....	25
8	Voorbeeldcode.....	26
8.1	Java.....	26
8.2	Visual Basic.....	27
9	SSL & niki.....	29
9.1	Java.....	29
9.2	Microsoft.....	30
10	Inschrijvingen.....	31
11	Omgevingen.....	32
11.1	Acceptatie en test omgeving.....	32
11.2	Live productie omgeving.....	32
11.3	Adressen.....	32



## 1 Inleiding

Voor de stichting LNP is een website ontwikkeld waar het verzamelde woningaanbod van verschillende projectopdrachtgevers wordt aangeboden. Dit woningaanbod wordt door de opdrachtgevers zelf, de abonnees, ingevoerd in de database van Niki. Dat kan op twee manieren.

- **Handmatig**

Door middel van een project administratie module kunnen projecten en woningen worden beheerd.

- **Geautomatiseerd**

Aanlevering van projecten in een gedefinieerd xml formaat.

Voor het handmatig beheer van projecten is een gebruikershandleiding beschikbaar voor de project administratie module. Dit document zal ingaan op alle communicatie die geautomatiseerd kan verlopen.



## 2 Globale beschrijving

Bij het centraal opslaan van projectdata van verschillende opdrachtgevers staan twee problemen centraal.

- *Hoe zorg je ervoor dat iedereen het over dezelfde gegevens heeft?*

Door deze gegevens centraal op te slaan in Niki en deze gegevens beschikbaar te stellen aan abonnee's door middel van xml-feeds. Dat zijn: **makelaars, opdrachtgevers, provincies, gemeenten, plaatsen, wijken, buurten, projectstatussen, woningstatussen, woningsoorten en kenmerken.**

- *Hoe behoudt de opdrachtgever het beheer over zijn eigen projecten?*

Door de opdrachtgever zelf een identifier te laten specificeren voor projecten en woningen. Aan de hand van deze identifier kan de opdrachtgever middels xml zijn gegevens beheren.



### 3 Feeds

Op de webserver van Niki zijn xml-files te vinden van de stamgegevens binnen Niki. In de bijlage zijn voorbeelden te vinden van deze xml-files. De gegevens zijn alleen benaderbaar met een geldig useraccount voor Niki. Het wachtwoord moet md5 encoded zijn.

Voorbeeld:

[https://xml.niki.nl/feed/province.do?user=\[username\]&password=\[password\]](https://xml.niki.nl/feed/province.do?user=[username]&password=[password])

De stamgegevens hebben als enige doel om de abonnee op te hoogte te stellen van de identifiers van deze gegevens. Deze identifiers worden namelijk gebruikt om bij invoer van projecten duidelijk te maken naar welk stamgegeven gerefereerd wordt.

Lijst van provincies.

<https://xml.niki.nl/feed/province.do>

Lijst van steden zoals in mei 2006 bekend bij CBS. Sindsdien up to date gehouden door de stichting LNP.

<https://xml.niki.nl/feed/city.do>

Naast de koppeling met een provincie kan een abonnee ook de gemeente te weten komen door het volgende mee te geven.

`community=true`

Lijst van makelaars.

<https://xml.niki.nl/feed/broker.do>

Een logo kan opgevraagd worden door een extra parameter mee te sturen:

`logo=true`

Alle door de LNP ingevoerde opdrachtgevers zijn hier te vinden. Dit kunnen abonnees zijn, maar dat hoeft niet.

<https://xml.niki.nl/feed/developer.do>

Extra parameters zijn:

`logo=true` (toont ook de logo info)

`parent=true` (toont de hoofdopdrachtgever, indien aanwezig)

Hier kan een opdrachtgever zijn subopdrachtgevers opvragen zoals die bekend zijn bij de Niki.



<https://xml.niki.nl/feed/subdeveloper.do>

Lijst van mogelijke statussen die een woning kan hebben.

<https://xml.niki.nl/feed/housestatus.do>

Mogelijke verkoopstatussen van een project.

<https://xml.niki.nl/feed/projectstatus.do>

De woningsoorten zoals bekend bij Niki. Hierbij is tevens opgenomen welke attributen van een woning verplicht zijn per woningsoort.

<https://xml.niki.nl/feed/housemodel.do>

Verkoopcondities

<https://xml.niki.nl/feed/salecondition.do>

Huurcondities

<https://xml.niki.nl/feed/rentcondition.do>

Transactiecondities

<https://xml.niki.nl/feed/transactioncondition.do>

Woning kenmerken

<https://xml.niki.nl/feed/housecharacteristic.do>



## 4 Projecten invoeren

### 4.1 Inleiding

Een abonnee heeft totale controle over zijn projectgegevens binnen Niki. Hij kan projecten, woningtypen en woningen invoeren, wijzigen en verwijderen. Hiervoor kan de abonnee een xml opstellen aan de hand van een schema. Dit schema is te vinden in de bijlages. (Inp-schema-#.#.xsd).

De opgestelde xml moet voldoen aan een aantal eisen. Het invoeren van xml wijzigt een project in de database van Niki. Er wordt bij het wijzigen van projecten zorg voor gedragen dat de data in Niki consistent blijft, opdat projecten en woningen vindbaar blijven op de homepage. Hiertoe wordt elke wijziging gevalideerd.

Ingevoerde data wordt gevalideerd tegen de database, in plaats van op het niveau van de xml. Hierdoor is het niet noodzakelijk om bij wijzigingen in het project alle data van het project in zijn geheel op te sturen. Er kunnen dan ook *kleine updates* plaatsvinden. Verderop worden hier voorbeelden van getoond.

### 4.2 Validatie

De validatiemethode is afhankelijk van het stadium waarin het project zich bevindt.

- Project in concept: **naam, stad** en **status** verplicht.
- Project in voorbereiding: **naam, stad, status, beschrijving, geldige rijksdriehoek coördinaat** en de **datum** waarop het project in voorbereiding is geplaatst zijn verplicht, minimaal **1 afbeelding**, maximaal **6 afbeeldingen** en gekoppeld aan ten minste **1 woningsoort**.
- Project in verkoop/verhuur of 1 woningtype bevattend: **naam, stad, status, geldige rijksdriehoek coördinaat** en **datum verkoop/verhuur** zijn verplicht, minimaal **1 woningtype**.
  - Elk woningtype: **naam, beschrijving**, minimaal **1 afbeelding**, maximaal **6 afbeeldingen**, minimaal **1 woning**
  - Elke woning: **bouwnummer, woningsoort, status** en een geldige **prijs**

## 4.3 Beschrijving van de XML

### 4.3.1 Project algemeen

Een project bestaat uit verschillende elementen. Aan een project zijn woningtypes gekoppeld en aan elk woningtype zijn woningen gekoppeld. Elk project, woningtype en woning wordt geïdentificeerd aan de hand van een *externalId*, die de abonnee zelf specificeert.

Alleen de autorisatie is verplicht volgens het schema, omdat anders kleine updates niet gefaciliteerd kunnen worden.

```
<project externalId="zelfTeSpecificeren">

  <authorization ..../>    <!-- verplicht -->
  <projectdata ..../>

  <housetypes>
    <housetype externalId="htA" ..../>
    <housetype externalId="htB" ..../>
  </housetypes>

  <houses>
    <house externalId="hx"
externalHousetypeId="htA" .../>
    <house externalId="hy"
externalHousetypeId="htB" .../>
    <house externalId="hz"
externalHousetypeId="htB" .../>
  </houses>

</project>
```

### 4.3.2 Autorisatie

De abonnee heeft van de stichting LNP een gebruikersnaam en een wachtwoord gekregen. Deze zijn dezelfde als waarmee kan worden ingelogd in de project administratie tool. Het wachtwoord moet worden geconverteerd naar md5. Loginnaam en wachtwoord zijn verplicht.

Via de *owner* kan een abonnee een subabonnee specificeren/wijzigen voor het opgvoerde project. De op te geven identifier correspondeert met de identifiers uit de subdeveloper-feed.

```
<authorization>
  <loginname>username</loginname>
  <md5password>940bb....daac64f8645a7</md5password>
  <owner id="12"/>
</authorization>
```

### 4.3.3 Projectgegevens

Een project heeft om te beginnen een naam en een locatie. Zolang het project niet gepubliceerd is geweest (te vinden op de [www.niki.nl](http://www.niki.nl)) kan de stad worden aangepast.

```
<projectdata>
  <name>Project Voorbeeld</name>
  <location>
    <city id="81"/>
    <Xcoordinate>150000</Xcoordinate>
    <Ycoordinate>200000</Ycoordinate>
  </location>
  ...
```

#### 4.3.3.1 Project stadia & datums

Een project kent verschillende stadia. Via *dateInPreparation*, *dateSale* en *dateDone* kan worden aangegeven wanneer het project in voorbereiding, in verkoop/verhuur of in archief wordt gezet. Als geen van deze datums verstreken is, dan is het project in concept.

Projecten kunnen ook direct in verkoop/verhuur of zelfs in archief worden geplaatst.

Met de tag *startSale* wordt aangegeven wanneer de start verkoop van een project plaatsvindt. Deze datum wordt gebruikt op [www.niki.nl](http://www.niki.nl) bij projecten in voorbereiding.

```
...  
<startSale>20061101</startSale>  
<dateInPreparation>20060201</dateInPreparation>  
<dateSale>20060301</dateSale>  
<dateDone>20060901</dateDone>  
...
```

#### 4.3.3.2 Projectbeschrijving

Elk project kan worden voorzien van een onderstaande elementen. De disclaimer is puur tekst en komt terug op de printpagina van een project of een woningtype. De beschrijving is verplicht voor projecten in voorbereiding.

De beschrijving kan de volgende HTML elementen bevatten:

**a - b - strong - i - em - ul - u - li - br**

Er zijn twee mogelijkheden om dit in de xml mee te geven. Omsloten door een CDATA tag zonder escaping of zonder CDATA tags, maar met escaping (bijvoorbeeld: de < en > vervangen door &lt; en &gt;)

```
...  
<description>Description of Project A</description>  
<url>http://www.example.com</url>  
<disclaimer>Disclaimer of Project A</disclaimer>  
...
```

#### 4.3.3.3 Actieregel

De actieregel maakt het mogelijk om prominent op de detailpagina een wervende tekst te plaatsen. Bijvoorbeeld: "Kom de modelwoning bezoeken op de Open Dag op 21 augustus"

De actieregel is project gebonden en wordt bij het toekomstig aanbod en alle koop en huur woningtypes van een project getoond.

De actieregel heeft een verval datum. Na het bereiken van deze datum wordt de actieregel gewist.

```
...  
<promoTextLine>Kom de modelwoning bezoeken op de  
Open Dag op 21 augustus<promoTextLine>  
<promoTextDueDate>20100822</promoTextDueDate>  
...
```

#### 4.3.3.4 Woningsoorten

Bij een project in voorbereiding is het verplicht aan te geven welke woningsoorten het project zal bevatten (voor projecten in andere stadia is het niet verplicht). De identifiers corresponderen met de identifiers uit de housemodel feed.

```
...  
<housemodels>  
  <housemodel id="1"/>  
  <housemodel id="2"/>  
  <housemodel id="3"/>  
</housemodels>  
...
```

#### 4.3.3.5 Inschrijvingen

Een bezoeker van niki kan interesse tonen in projecten en woningtypes. De abonnee kan opgeven of en hoe zij deze wil ontvangen. De interesses worden gemaïld naar de opgegeven emailadressen. Hierbij kan een abonnee kiezen tussen html mailtjes (standaard) of in xml formaat.

```
...  
<doSubscription>true</doSubscription>  
<emailaddresses>  
  <emailaddress>  
    interesse.ontvanger2@opdrachtgever.nl  
  </emailaddress>  
  <emailaddress type="xml">  
    xml-interest-reader@opdrachtgever.nl  
  </emailaddress>  
</emailaddresses>  
...
```

#### 4.3.3.6 Gekoppelde makelaars, opdrachtgevers en bestanden

Aan een project kunnen betrokken partijen worden gekoppeld door in de brokers en



developers tags aan te geven welke makelaars en opdrachtgevers meewerken aan dit project. De identifiers kunnen worden bepaald aan de hand van de broker en developer feeds.

De eerste makelaar is automatisch ook de directievoerend makelaar.

Als een abonnee ervoor kiest om de status van verkoop (*showStatus*) te tonen dan wordt op [www.niki.nl](http://www.niki.nl) een lijst getoond met verkoopstatus van een project. Anders alleen een samenvatting.

De verkoopstatus (*saleStatus*) correspondeert met de projectstatus feed. Deze wordt getoond bij het detail resultaat van een project in voorbereiding.

De verwerking van files wordt verderop omschreven.

```
...

<brokers>
  <broker id="1"/>
  <broker id="2"/>
</brokers>

<developers>
  <developer id="3"/>
</developers>

<showStatus>true</showStatus>
<isRentProject>>false</isRentProject>
<saleStatus id="1"/>

<files ... />

</projectdata>
```

#### 4.3.3.7 Locatie

Om de locatie te kunnen tonen in Google Maps en om de buurt en wijk te bepalen is het verplicht om coördinaten mee te geven. De coördinaten kunnen opgegeven worden in rijksdriehoek formaat of in GPS formaat. Er wordt gecontroleerd of de opgegeven coördinaat binnen Nederland valt.

Rijksdriehoek maakt gebruik van de elementen Xcoordinate en Ycoordinate voor respectievelijk de x en y coördinaat. De opgegeven waardes zijn in meters.

```
...

<location>
  <city id="81"/>
```

```
    <Xcoordinate>150000</Xcoordinate>
    <Ycoordinate>200000</Ycoordinate>
  </location>

  ...
```

GPS maakt gebruik van de elementen latitude en longitude. De opgegeven waarden zijn in graden volgens het stelsel ETRS89. Wees er bewust van dat o.a. Google Maps gebruik maakt van WGS84. De verschillen zijn echter minimaal waardoor omrekenen tussen beide stelsels niet noodzakelijk is.

```
    ...

  <location>
    <city id="81"/>
    <latitude>52.156478</latitude>
    <longitude>5.390014</longitude>
  </location>

  ...
```



### 4.3.4 Woningtypes

Een woningtype kent een naam en een beschrijving. De beschrijving moet voldoen aan dezelfde eisen als de beschrijving van een project.

De verwerking van files wordt verderop omschreven.

```
<housetype externalId="xyhtAA">
  <name>housetypenaamAA</name>
  <description>
    Description of HouseType AA
  </description>
  <files .../>
</housetype>
```

### 4.3.5 Woningen

Een woning wordt gekoppeld aan een woningtype. Bij invoer moet minimaal de *status*, het *buildid* en *housemodel* worden opgegeven. Als de woning actief is moet ook een correcte prijs worden opgegeven.

Bij het project wordt aangegeven of het project koop of huurwoningen bevat. Afhankelijk daarvan geeft de abonnee de tag *rent* of de tags *minprice* en *maxprice* op. Verkooprijzen kennen een prijsbereik, maar als het verschil nul is dan wordt op [www.niki.nl](http://www.niki.nl) maar 1 prijs getoond.

Afhankelijk van het woningsoort kunnen enkele van de volgende attributen verplicht gesteld worden. (*groundsurface*, *livingsurface*, *content*, *servicecosts* en *roomcount*). Deze afhankelijkheid kan bepaald worden vanuit de housemodel feed.

De status van een woning is *Te Koop*, *In Optie*, *Verkocht*, *Te Huur*, *Verhuurd*. De identifiers zijn te bepalen aan de hand van de housestatus feed. Tevens kunnen er nog enkele optionele attributen worden toegevoegd aan de woning: adres, huisnummer, huisnummer toevoeging en postcode (*street*, *number*, *numberAddition*, *zip*). In tegenstelling tot de beschrijvingen van woningtypes en projecten is de beschrijving van een woning normale tekst zonder opmaak.

Naast de prijs zijn ook de voorwaarden waaronder een huis verkocht/verhuurd worden op te geven. *saleCondition* zal standaard ingevuld zijn met waarde *1* = vrij op naam. *rentCondition* staat standaard op *3* = per maand. De id's van de verschillende *sale*- en *rentConditions* zijn op te vragen via de feed.

Wanneer een woning verkocht of verhuurd is kan er extra informatie meegestuurd worden over de daadwerkelijke koop/huur transactie. De *transactionPrice*, *transactionCondition* en *dateSigned* worden gebruikt om aan te geven voor welke prijs de woning verkocht is, onder welke voorwaarden en op welke datum de koop getekend is. De verschillende *transactionCondition* ids zijn op te vragen via de feed.

Per woning kunnen kenmerken in de vorm van *houseCharacteristics* worden toegevoegd. De verschillende kenmerken zijn op te vragen via de housecharacteristic feed.

```
<house externalId="h1" externalHousetypeId="ht1">
  <status id="1"/>
  <housemodel id="3"/>
  <active>true</active>

  <rent>300</rent>
  <pricemin>250000</pricemin>
  <pricemax>250000</pricemax>

  <buildid>hAAA</buildid>
  <description>
    Description of this house
  </description>

  <groundsurface>120</groundsurface>
```

```

<livingsurface>100</livingsurface>
<content>400</content>
<servicecosts>20</servicecosts>
<roomcount>4</roomcount>

<saleCondition id="1"/>
<transactionPrice>250000</transactionPrice>
<transactionCondition id="2"/>
<dateSigned>20081211</dateSigned>

<houseCharacteristics>
  <houseCharacteristic id="1"/>
  <houseCharacteristic id="2"/>
</houseCharacteristics>
</house>

```

### 4.3.6 Bestanden

Mogelijkheden:

- Projecten
  - Maximaal 6 afbeeldingen (type *image*)
  - 1 plattegrond (pdf) (type *mappdf*)
  - 1 brochure (pdf) (type *brochure*)
  - Optioneel een logo (type *logoimage*)

*Wordt getoond op de printpagina van het factsheet (ook van woningtypes)*

- Woningtypes
  - Maximaal 6 afbeeldingen (type *image*)
  - 1 plattegrond (pdf) (type *mappdf*)
  - 1 brochure (pdf) (type *brochure*)

```

<file type="brochure">
  <binary>
    <mimetype>application/pdf</mimetype>
    <data>
EhACEIQAhCEAIQhACEIQAhCEAIQhACEIQAhCEAIQhACEIQAhCEAIQhACEIQAh
hDFWERFDCEIQdhCEAIQhACEI/9j/4AAQSkZJRgAB.....
.....
..... IQhACEIQAhCEAIQhACEIQAhCEB//Z
    </data>

```

```
        </binary>
    </file>
    <file type="image">
        <reference>
            <mimetype>image/gif</mimetype>
            <uri>http://link.to/image.gif</uri>
        </reference>
    </file>
```

Een file kan op twee manieren worden doorgegeven. Per referentie (een uri) of base64 gecodeerd. Elke pdf moet mimetype *application/pdf* hebben, anders volgt er een foutmelding. Geaccepteerde mimetypes voor afbeeldingen: [*image/jpeg, image/png, image/gif*]

De brochures en plattegronden worden ter download aangeboden op de detailpagina van [www.niki.nl](http://www.niki.nl).

Afbeeldingen moeten minimaal *1024 pixels breed danwel 768 pixels hoog* zijn. Ingevoerde afbeeldingen (en logo's) worden geconverteerd naar 4 afmetingen mits de afbeelding al minimaal dat formaat had. De breedte/hoogte verhouding blijft gehandhaafd.

*1024x768 – 640x480 – 232x172 – 126x94*

Een file mag niet groter zijn dan 5 megabyte.

## 4.4 Kleine updates

De beschreven wijze van communiceren biedt de mogelijkheid om kleine wijzigingen op de projecten in Niki uit te voeren. Zo kan 1 of meerdere woningtypes of woningen worden aangepast, terwijl de rest onaangeroerd blijft.

### 4.4.1 Richtlijnen

#### Wijzigen van de projectdata

Geef onderstaande xml op. Alleen de files missen op project niveau. Er zijn geen woningtypes of woningen gespecificeerd.

```
<project externalId="zelfTeSpecificeren">

  <authorization .... />
  <projectdata>
    <name ... />
    <location .... />
    <startSale .... />

    <dateInPreperation .... />
    <dateSale .... />
    <dateDone .... />

    <description .... />
    <url .... />
    <disclaimer .... />

    <housemodels .... />
    <doSubscription .... />
    <emailaddresses .... />

    <brokers .... />
    <developers .... />

    <showStatus .... />
    <isRentProject .... />
    <saleStatus .... />
  </projectdata>
</project>
```

## Wijzigen/toevoegen van afbeeldingen van een project

Geef alle files opnieuw op.

```
<project externalId="zelfTeSpecificeren">
  <authorization ..../>
  <projectdata>
    <files>
      <file ../>
      <file ../>
    </files>
  </projectdata>
</project>
```

## Wijzigen/toevoegen van een woningtype

Bij het toevoegen van een nieuw woningtype moet ook tenminste één woning worden toegevoegd. Echter, als het gaat om een wijziging van een bestaand woningtype dan is het toevoegen van een nieuwe woning niet vereist.

```
<project externalId="zelfTeSpecificeren">
  <authorization ..../>
  <housetypes>
    <housetype externalId="xx">
      <name ... />
      <description .../>
    </housetype>
  </housetypes>
</project>
```

## Wijzigen/toevoegen van afbeeldingen aan een woningtype

Geef alle files opnieuw op.

```
<project externalId="zelfTeSpecificeren">

  <authorization ..../>
  <housetypes>
    <housetype externalId="xx">
      <files>
        <file ../>
        <file ../>
      </files>
    </housetype>

    <!-- eventueel meerdere woningtypes -->
  </housetypes>

</project>
```

## Wijzigen van de status van een woning

```
<project externalId="zelfTeSpecificeren">

  <authorization ..../>
  <houses>
    <house externalId="xx" externalHousetypeId="yy">
      <status id="2"/>
    </house>

    <!-- eventueel meerdere woningen -->
  </houses>

</project>
```

## Toevoegen (of aanpassen) van een woning met minimale attributen

### Een huurwoning opvoeren

```
<project externalId="zelfTeSpecificeren">

  <authorization ..../>
  <houses>
    <house externalId="xx" externalHousetypeId="yy">
      <status id="1"/>
      <housemodel id="1"/>

      <rent>860</rent>
      <buildid>a</buildid>

      <!-- afhankelijk van het woningmodel ook
           andere attributen verplicht -->
    </house>

    <!-- eventueel meerdere woningen -->
  </houses>

</project>
```

### Verwijderen van een woning

```
<project externalId="zelfTeSpecificeren">

  <authorization ..../>
  <houses>
    <house externalId="xx" externalHousetypeId="yy"
           action="delete">
  </houses>

</project>
```

### Verwijderen van een woningtype

```
<project externalId="zelfTeSpecificeren">
  <authorization ..../>
  <housetypes>
    <housetype externalId="xx" action="delete">
  </housetypes>
</project>
```

### Verwijderen van een project

```
<project externalId="zelfTeSpecificeren">
  <authorization ..../>
  <projectdata action="delete"/>
</project>
```

### Controle behouden: *het hele project opnieuw*

Een abonnee kan het gehele project opnieuw aanbieden. Alle data die bij niki bekend is betreffende het project en die niet in de xml wordt genoemd zal worden verwijderd.

```
<project externalId="zelfTeSpecificeren" handleAsNew="true">
  <authorization ..../>
  <projectdata .../>
  <housetypes .../>
  <houses .. />
</project>
```



## 5 Beveiliging

Het domein xml.niki.nl is alleen beveiligd bereikbaar. Om een abonnee toegang te geven tot dit domein zijn twee acties noodzakelijk:

1. De abonnee zal zijn ip-nummer(s) moeten opgeven. Deze worden in dan in de firewall geaccepteerd.
2. Om met xml.niki.nl te kunnen communiceren heeft een abonnee een ondertekend certificaat nodig.

Om certificaten te ondertekenen is een certificaten autoriteit opgezet. Het root certificaat van de autoriteit is bijgevoegd bij dit document. Dit certificaat wordt gebruikt om de clients en server certificaten mee te ondertekenen en kan indien gewenst worden opgenomen als een vertrouwd certificaat in browsers en in andere applicaties.

Om een ondertekend certificaat te produceren, zijn de volgende gegevens noodzakelijk:  
*Common Name, Organisational Unit, Plaats, Provincie, Emailadres*

De abonnee ontvangt vervolgens twee files:

### 1)abonnee.crt

Het ondertekende certificaat met de verstrekte gegevens. Het certificaat is in x.509 formaat en PEM encoded.

### 2)abonnee.p12

De public en private key van het certificaat in PKCS#12 formaat. Deze file kan geïmporteerd worden in een browser en in de software om te communiceren met xml.niki.nl.



## 6 XML opsturen

- 1) De abonnee stelt per in te voeren of up te daten project een xml samen.
  1. Grootte per xml maximaal 200MB.
  2. De xml moet UTF-8 gecodeerd zijn en voldoen aan het laatste schema.
  3. De eerste regel van de xml starten op de eerste kolom:  
`<?xml version="1.0" encoding="UTF-8"?>`
- 2) Elke xml kan worden gepost naar
  1. <https://xml.niki.nl/parser.do> met als post parameter xml (kleine letters)
  2. De xml moet hierbij urlencoded worden.
  3. De content length van de post moet worden opgegeven.
  4. De content encoding moet application/x-www-form-urlencoded zijn.
- 3) De xml wordt vervolgens in een transactie in de database verwerkt. Het post-request levert als resultaat een http return code:
  - 200:** verwerking geslaagd
  - 400:** verwerking niet geslaagd met daarbij vermeld de reden(en)
  - 403:** autorisatie mislukt
  - 500:** onverwachte fout
  - 503:** als er geen projecten kunnen worden ingevoerd op dit moment.



## 7 Testen

	Testomgeving	Live
Feeds	<a href="http://acc.niki.nl/xml/feed/">http://acc.niki.nl/xml/feed/</a>	<a href="https://xml.niki.nl/feed/">https://xml.niki.nl/feed/</a>
Parser	<a href="http://acc.niki.nl/xml/parser.do">http://acc.niki.nl/xml/parser.do</a>	<a href="https://xml.niki.nl/parser.do">https://xml.niki.nl/parser.do</a>
Handmatige upload	<a href="http://acc.niki.nl/xml/">http://acc.niki.nl/xml/</a>	<a href="https://xml.niki.nl/">https://xml.niki.nl/</a>

Bij het testen van de upload aan de hand van het tekstvlak is het verstandig om **Firefox** te gebruiken als testbrowser. Als de parser namelijk een fout ontdekt bij toetsing dan wordt een http status 400 teruggegeven met de foutmeldingen als tekst daarbij opgenomen. Internet explorer laat deze informatie achterwege terwijl Firefox deze wel toont.

Er moet overigens worden opgemerkt dat een xml file die voldoet aan het schema niet per se succesvol verwerkt zal worden. Aangezien er ook gedeelten van een project kunnen worden ge-update, is het niet altijd noodzakelijk om alle informatie mee te sturen. Een ge-update project moet echter blijven voldoen aan de minimale criteria zoals gesteld in het functioneel ontwerp. Deze toetsing vindt pas plaats nadat de xml is ontvangen en kan leiden tot foutmeldingen.



## 8 Voorbeeldcode

### 8.1 Java

```
String xml = "....";
HttpClient client = new HttpClient();
PostMethod method = new PostMethod("http://acc.niki.nl/xml/parser.do");

method.setParameter("xml", xml);

int httpStatus = client.executeMethod(method);
String response = method.getResponseBodyAsString();
```



## 8.2 Visual Basic

### Inlezen van een file

```
Dim strPhysicalPath As String = " TestFile.xml"
Dim byt() As Byte

Dim objFS As FileStream =
    File.Open(strPhysicalPath, FileMode.Open, FileAccess.Read)
Try
    ReDim byt(objFS.Length)
    objFS.Read(byt, 0, Convert.ToInt32(objFS.Length))
    strXml = System.Text.Encoding.UTF8.GetString(byt)
Catch ex As Exception
    Throw ex
Finally
    objFS.Close()
End Try
```

### Posten van een xml file

```
Private Function PostXml(ByVal url As String, ByVal xmlstr As String)
    as string
    Try
        Dim HttpReq As New MSXML2.XMLHTTP
        Dim objXMLDocument As New MSXML2.DOMDocument50
        Dim str As String

        objXMLDocument.async = False
        objXMLDocument.loadXML(xmlstr)

        HttpReq.open("POST", url, False)

        HttpReq.setRequestHeader
            ("Content-Type", "application/x-www-form-urlencoded")

        str = "xml=" & xmlstr
        HttpReq.send(str)

        Return HttpReq.responseText()

    Catch ex As Exception
        Throw ex
    End Try
End Sub
```



## Posten van een xml file (nogmaals)

```
Sub test()  
  Dim XMLHTTPRequest As New XMLHTTP30  
  Dim xml As String  
  
  Open "C:\LNPXML\8590119.XML" For Binary As #1  
  
  xml = Space(LOF(1))  
  Get #1, , xml  
  Close #1  
  
  xml = "xml=" & xml  
  
  XMLHTTPRequest.Open "POST", "http://acc.niki.nl/xml/parser.do", False  
  XMLHTTPRequest.setRequestHeader  
    "Content-type:", "application/x-www-form-urlencoded"  
  XMLHTTPRequest.setRequestHeader "Content-length:", "" & Len(xml)  
  XMLHTTPRequest.send xml  
  
  Debug.Print XMLHTTPRequest.Status  
End Sub
```



## 9 SSL & niki

### 9.1 Java

Om Java SSL te laten praten hebben wij het volgende gedaan, ons hierbij baserende op de httpclient van apache. <http://hc.apache.org/httpcomponents-client/index.html>

Het is de bedoeling dat je een protocol implementeert voor SSL, door een *ProtocolSocketFactory* te gebruiken en de applicatie te vertellen dat alle https verbindingen met je eigen implementatie gemaakt moeten worden.

Hierbij helpt apache ons. Ze hebben een stuk code geschreven voor client-side - server-side SSL communicatie. Zoek op de term *AuthSSLProtocolSocketFactory* op <http://jakarta.apache.org/commons/httpclient/sslguide.html>

De *ProtocolSocketFactory* moet worden bediend met client en trusted keystores. Hoe je die kan maken vind je in de javadoc van de code van Apache. De tool Keystore Explorer (<http://www.lazgosoftware.com/kse/>) kan hierbij erg behulpzaam blijken.

De voorbeeldcode blijft ongewijzigd, er is alleen een https protocol geregistreerd.



## 9.2 Microsoft

Het client certificates moet geïnstalleert worden in de personal folder van de desbetreffende gebruiker (raoul of administrator, etc)

Wanneer je vanuit code het geëxporteerde certificaat aanroep moet dit ook onder de rechten zijn van de hierboven genoemde gebruiker. Dit is dus geen probleem wanneer je een winApp.exe gebruikt.

Bij het uitvoeren van een winApp.exe kan je aangeven als welke gebruiker de .exe gerund moet worden. Bij een webApp is dit een heel ander verhaal. Bij een Soap webservice kan een gebruiker met behulp van "impersonation" aanroepen. (probleem opgelost). Bij een client certificates is dat niet mogelijk. Echter is er wel een oplossing.

Met behulp van de "Windows HTTP Services Certificate Configuration Tool" kan je namelijk rechten verlenen aan een personal - client certificate.

Dit zodat de webuser (aspnet user of je application pool) rechten heeft op het client certificate.

Meer informatie over deze oplossing is te vinden in de volgende artikelen:

Probleem <http://support.microsoft.com/kb/817854/>

Oplossing <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnnetsec/html/secneth13.asp>



## 10 Inschrijvingen

Een abonnee kan ervoor kiezen om bezoekers van de site de optie te bieden om interesse te tonen in zijn project(en). Bezoekers vullen dan een formulier in. Dit formulier wordt opgeslagen in de database. Via de project administratie applicatie zijn hier excel uitdraaien van te maken.

Een abonnee kan er ook voor zorgen dat inschrijvingen realtime worden verzonden naar een emailadres. Er kan dan gekozen worden om de inschrijvingen in xml op te sturen. Deze xml's voldoen aan een schema. Dit is bijgevoegd bij dit document.

### **subscription-#.#.xsd**

Het schema waaraan ge-emailde inschrijvingen voldoen.

De emails worden verstuurd met content-type *text/xml* en content encoding *UTF-8*. Er zijn voorbeelden beschikbaar.

Voor het geautomatiseerd verwerken van inschrijvingen is ook een webservice beschikbaar. Deze webservice wordt apart beschreven met de documentatie in het downloadbare bestand *beschrijving-webservice.zip*



## **11 Omgevingen**

Niki maakt gebruik van twee omgevingen. Op beide omgevingen is xml invoer applicatie beschikbaar.

Gebruikers accounts moeten voor beide omgevingen afzonderlijk aangemaakt worden en zijn niet uitwisselbaar.

Er is geen garantie dat stamgegevens op beide omgevingen overeenkomen.

Het is mogelijk dat de acceptatieomgeving over een nieuwere versie beschikt dan de productieomgeving.

### ***11.1 Acceptatie en test omgeving***

De acceptatie en testomgeving wordt gebruikt voor het testen en accepteren van nieuwe Niki releases. Daarnaast stelt de stichting LNP deze omgeving beschikbaar aan deelnemers en derden voor het testen van aan Niki gekoppelde applicaties. Vanwege het karakter van deze omgeving is hij beschikbaar op basis van best effort.

Locatie xml invoer applicatie

Voorbeel formulier: <https://acc.niki.nl/xml/>

Parser: <https://acc.niki.nl/xml/parser.do>

### ***11.2 Live productie omgeving***

De live productie omgeving bestaat uit de www.niki.nl website en diverse gekoppelde sites. Deze omgeving is redundant uitgevoerd en is daardoor vrijwel permanent beschikbaar.

Locatie xml invoer applicatie

Voorbeeld formulier: <https://xml.niki.nl>

Parser: <https://xml.niki.nl/parser.do>

### ***11.3 Adressen***

De Niki servers opereren binnen de volgende IP adresrange:

87.233.15.0/25 (87.233.15.0 – 87.233.15.127)