# Sample userInputSpec.xml

## Sample userInputSpec.xml

Here's an example 3 panel userInputSpec.xml file. You specify the use of this XML document inside your install definition in the resource section like this (assuming your ant build.xml uses a property called build.dir):

```
<resources>
  <res id="userInputSpec.xml"
src="@{build.dir}/userInputSpec.xml" />
</resources>
```

**userInputSpec.xml**

```
<userInput>
  <!-- Install -->
    <panel order="0">
        <createForPack name="Main
Application" />
        <field type="title" txt="Import
Keystores" bold="true" size="1" />
        <field type="divider" align="top"/>
        <!-- Keystore -->
        <field type="staticText"
align="left" txt="Existing SSL keystore to
import:"/>
        <field type="file" align="left"
variable="existing.ssl.keystore">
          <spec txt="" size="25" set=""/>
        </field>
        <field type="space"/>
        <!-- Truststore -->
```

```
        <field type="staticText"
align="left" txt="Existing SSL truststore to
import:"/>
        <field type="file" align="left"
variable="existing.ssl.truststore">
          <spec txt="" size="25" set=""/>
        </field>
        <field type="space"/>
        <!-- Signing Keystore -->
        <field type="staticText"
align="left" txt="Existing signing keystore
to import:"/>
        <field type="file" align="left"
variable="existing.signing.keystore">
          <spec txt="" size="25" set=""/>
        </field>
        <field type="space"/>
        <field type="divider"
align="bottom"/>
    </panel>
    <panel order="1">
      <!-- Validate access to keystores with
information from last panel -->
        <createForPack name="Main
Application" />
        <field type="title" txt="SSL
Keystore Settings" bold="true" size="1" />
        <field type="divider"
align="bottom"/>
        <!-- Skip validation in case
customer has something wrong they want to
```

```
fix later -->
        <field type="check" align="left"
variable="skip.keystore.validation">
          <spec txt=" Skip keystore password
validation (not recommended)" size="25"
true="true" false="false" set="false"/>
        </field>
        <field type="divider" align="top"/>
        <!-- Keystore -->
        <field type="combo"
variable="existing.ssl.keystore.type">
          <spec txt="Keystore type:"
id="existing.ssl.keystore.type">
            <choice txt="JKS" value="JKS"
set="true"/>
            <!--
            <choice txt="PKCS12"
value="PKCS12"/>
            -->
          </spec>
        </field>
        <field type="space"/>
        <field type="text" align="left"
variable="keystore.key.alias">
            <spec txt="Keystore Key Alias:"
size="25" set="alias-1"/>
        </field>
        <field type="password" align="left"
variable="keystore.password">
          <spec>
            <pwd txt="Keystore Password:"
```

```
size="25" set=""/>
            <pwd txt="Retype Password:"
size="25" set=""/>
          </spec>
          <validator
class="com.izforge.izpack.panels.userinput.v
alidator.PasswordEqualityValidator"
txt="Both keystore passwords must match."
id="key for the error text"/>
          <validator
class="com.izforge.izpack.panels.userinput.v
alidator.PasswordKeystoreValidator"
txt="Could not validate keystore with
password and alias provided." id="key for
the error text">
            <param name="keystoreFile"
value="${existing.ssl.keystore}"/>
            <param name="keystoreType"
value="${existing.ssl.keystore.type}"/>
            <param name="keystoreAlias"
value="${keystore.key.alias}"/>
            <param name="skipValidation"
value="${skip.keystore.validation}"/>
          </validator>
        </field>
        <field type="space"/>
        <!-- Truststore -->
        <field type="combo"
variable="existing.ssl.truststore.type">
          <spec txt="Truststore type:"
id="existing.ssl.truststore.type">
```

```
            <choice txt="JKS" value="JKS"
set="true"/>
            <!--
            <choice txt="PKCS12"
value="PKCS12"/>
            -->
          </spec>
        </field>
        <field type="space"/>
        <field type="password" align="left"
variable="truststore.password">
          <spec>
            <pwd txt="Truststore Password:"
size="25" set=""/>
            <pwd txt="Retype Password:"
size="25" set=""/>
          </spec>
          <validator
class="com.izforge.izpack.panels.userinput.v
alidator.PasswordEqualityValidator"
txt="Both truststore passwords must match."
id="key for the error text"/>
          <validator
class="com.izforge.izpack.panels.userinput.v
alidator.PasswordKeystoreValidator"
txt="Could not validate keystore with
password and alias provided." id="key for
the error text">
            <param name="keystoreFile"
value="${existing.ssl.truststore}"/>
            <param name="keystoreType"
```

```
value="${existing.ssl.truststore.type}"/>
            <param name="skipValidation"
value="${skip.keystore.validation}"/>
          </validator>
        </field>
        <field type="divider"
align="bottom"/>
    </panel>
    <panel order="2">
      <!-- Validate access to signing
keystore with information from last panel
-->
        <createForPack name="Main
Application" />
        <field type="title" txt="Signing
Keystore Settings" bold="true" size="1" />
        <field type="divider"
align="bottom"/>
        <!-- Skip validation in case
customer has something wrong they want to
fix later -->
        <field type="check" align="left"
variable="skip.keystore.validation">
          <spec txt=" Skip keystore password
validation (not recommended)" size="20"
true="true" false="false" set="false"/>
        </field>
        <field type="divider" align="top"/>
        <!-- Keystore -->
        <field type="combo"
variable="existing.signing.keystore.type">
```

```
          <spec txt="Keystore type:"
id="existing.signing.keystore.type">
            <choice txt="JKS" value="JKS"
set="true"/>
            <!--
            <choice txt="PKCS12"
value="PKCS12"/>
            -->
          </spec>
        </field>
        <field type="space"/>
        <field type="password" align="left"
variable="signing.keystore.password">
          <spec>
            <pwd txt="Keystore Password:"
size="20" set=""/>
            <pwd txt="Retype Password:"
size="20" set=""/>
          </spec>
          <validator
class="com.izforge.izpack.panels.userinput.v
alidator.PasswordEqualityValidator"
txt="Both signing keystore passwords must
match." id="key for the error text"/>
          <validator
class="com.izforge.izpack.panels.userinput.v
alidator.PasswordKeystoreValidator"
txt="Could not access keystore with password
provided." id="key for the error text">
            <param name="keystoreFile"
value="${existing.signing.keystore}"/>
```

```
            <param name="keystoreType"
value="${existing.signing.keystore.type}"/>
            <param name="skipValidation"
value="${skip.keystore.validation}"/>
          </validator>
        </field>
        <field type="space"/>
        <!-- Signing Key 1 -->
        <field type="text" align="left"
variable="first.signing.keystore.key.alias">
            <spec txt="First Signing Alias:"
size="20" set="alias-1"/>
        </field>
        <field type="password" align="left"
variable="first.signing.password">
          <spec>
            <pwd txt="First Signing
Password:" size="20" set=""/>
            <pwd txt="Retype Password:"
size="20" set=""/>
          </spec>
          <validator
class="com.izforge.izpack.panels.userinput.v
alidator.PasswordEqualityValidator"
txt="Both first signing key passwords must
match." id="key for the error text"/>
          <validator
class="com.izforge.izpack.panels.userinput.v
alidator.PasswordKeystoreValidator"
txt="Could not validate keystore with
password and first signing alias provided."
```

```
id="key for the error text">
            <param name="keystoreFile"
value="${existing.signing.keystore}"/>
            <param name="keystoreType"
value="${existing.signing.keystore.type}"/>
            <param name="keystorePassword"
value="${signing.keystore.password}"/>
            <param name="keystoreAlias"
value="${first.signing.keystore.key.alias}"/
>
            <param name="skipValidation"
value="${skip.keystore.validation}"/>
          </validator>
        </field>
        <field type="space"/>
        <!-- Signing Key 2 -->
        <field type="text" align="left"
variable="second.signing.keystore.key.alias"
>
            <spec txt="Second Signing
Alias:" size="20" set="crate-cmd-alias"/>
        </field>
        <field type="password" align="left"
variable="second.signing.password">
          <spec>
            <pwd txt="Second Signing
Password:" size="20" set=""/>
            <pwd txt="Retype Password:"
size="20" set=""/>
          </spec>
          <validator
```

```
class="com.izforge.izpack.panels.userinput.v
alidator.PasswordEqualityValidator"
txt="Both second signing key passwords must
match." id="key for the error text"/>
          <validator
class="com.izforge.izpack.panels.userinput.v
alidator.PasswordKeystoreValidator"
txt="Could not validate keystore with
password and second signing alias provided."
id="key for the error text">
            <param name="keystoreFile"
value="${existing.signing.keystore}"/>
            <param name="keystoreType"
value="${existing.signing.keystore.type}"/>
            <param name="keystorePassword"
value="${signing.keystore.password}"/>
            <param name="keystoreAlias"
value="${second.signing.keystore.key.alias}"
/>
            <param name="skipValidation"
value="${skip.keystore.validation}"/>
          </validator>
        </field>
        <field type="divider"
```

```
align="bottom"/>
    </panel>
</userInput>
```